Computer Methods in Applied Mechanics and Engineering 418 (2024) 116470

Contents lists available at ScienceDirect

Computer Methods in Applied Mechanics and Engineering

journal homepage: www.elsevier.com/locate/cma

Finite element-based invariant-domain preserving approximation of hyperbolic systems: Beyond second-order accuracy in space[☆]

Jean-Luc Guermond[a,*], Murtazo Nazarov[b], Bojan Popov[a]

[a] Department of Mathematics, Texas A&M University 3368 TAMU, College Station, TX 77843, USA
[b] Division of Scientific Computing, Department of Information Technology, Uppsala University, Uppsala, Sweden

ARTICLE INFO

Keywords:
Hyperbolic systems
Riemann problem
Invariant domain
High-order method
Limiting
Finite element method

MSC:
65M60
65M12
35L65
35L45

ABSTRACT

This paper proposes an invariant-domain preserving approximation technique for nonlinear conservation systems that is high-order accurate in space and time. The algorithm mixes a high-order finite element method with an invariant-domain preserving low-order method that uses the closest neighbor stencil. The construction of the flux of the low-order method is based on an idea from Abgrall et al. (2017). The mass flux of the low-order and the high-order methods are identical on each finite element cell. This allows for mass preserving and invariant-domain preserving limiting.

1. Introduction

This paper is concerned with the approximation of hyperbolic systems in conservation form using finite elements of degree two and higher. In particular, the paper proposes answers the some questions raised over the years 2014 to 2016 in the PhD theses of [1, §3.3.2.1] and [2, §4.2.2], where it was observed that the low-order invariant-domain preserving method proposed in [3] was not robust with respect to the polynomial degree. Building on ideas developed in [4] (see Propositions 3.1 and 3.2 therein), we propose here a variation of the invariant-domain preserving method from [3] that behaves better as the polynomial degree increases. In particular, the low-order method is based on the closest neighbor stencil.

To avoid distracting details regarding boundary conditions, we consider the Cauchy problem posed over a space domain $D \subset \mathbb{R}^d$ and a time interval $[0, T]$ with $T > 0$:

[☆] This material is based upon work supported in part by the National Science Foundation, USA grants DMS2110868, by the Air Force Office of Scientific Research, USAF, USA, under grant/contract number FA9550-23-1-0007, and by the Army Research Office, USA under grant/contract number W911NF-19-1-0431. The second author is funded by the Swedish Research Council (VR), Sweden under grant number 2021-04620. The third author is supported by the European Union-NextGenerationEU , through the National Recovery and Resilience Plan of the Republic of Bulgaria, project No BG-RRP-2.004-0008.

[*] Corresponding author.
E-mail address: guermond@tamu.edu (J.-L. Guermond).

https://doi.org/10.1016/j.cma.2023.116470
Received 27 April 2023; Received in revised form 26 August 2023; Accepted 19 September 2023
Available online 16 October 2023

$$\begin{cases} \partial_t \boldsymbol{u} + \nabla\cdot \mathbb{f}(\boldsymbol{u}) = \boldsymbol{0}, & \text{for } (\boldsymbol{x},t)\in D\times(0,T), \\ \boldsymbol{u}(\boldsymbol{x},0) = \boldsymbol{u}_0(\boldsymbol{x}), & \text{for } \boldsymbol{x}\in D. \end{cases} \tag{1.1}$$

The dependent variable takes values in $\mathbb{R}^m$, $m \geq 1$. We assume that (1.1) has an invariant domain $\mathcal{A}\subset\mathbb{R}^m$. This means that if $\boldsymbol{u}_0$ takes values in $\mathcal{A}$ almost everywhere in D (in the absence of perturbations due to the boundary conditions), then any admissible solution to the Cauchy problem also takes values in $\mathcal{A}$ almost everywhere in $D\times(0,T)$. We assume that $\mathbb{f}:\mathcal{A}\rightarrow\mathbb{R}^{m\times d}$ is Lipschitz. We say that a numerical approximation of (1.1) is invariant-domain preserving if it leaves $\mathcal{A}$ globally invariant.

There is a vast literature on finite volume and discontinuous Galerkin methods describing techniques that are third- and higher-order accurate in space and are invariant-domain preserving (see e.g., [5–7] for the finite volume literature, and [8–10] for the discontinuous Galerkin literature). By comparison, the continuous finite element literature on this topic is sparse. The objective of this paper is to propose some solutions to fill this gap. Some of the arguments presented in this paper find their root in [4] and have some similarities with the residual distribution method developed in [11–13], [14].

Our starting point is the technique described in the following series of papers [3,15], [16–18]. The main idea behind [3,15–17] consists of combining two methods in the spirit of the flux transport corrected methodology of [19,20] (see also [21,22], and the literature therein for other finite element extensions on this idea): A low-order method that is invariant-domain preserving serves as gate keeper to limit a high-order method which may not be invariant-domain preserving but is somewhat entropy consistent. One important property of most of the methods mentioned above is that they do not have theoretical upper limits on the polynomial degree of the space approximation to be invariant-domain preserving. In principle, the method from [3,15–17] can be implemented with any polynomial degree. This is indeed true, but as observed in the PhD theses of [1, §3.3.2.1] and [2, §4.2.2], the low-order invariant-domain preserving method, which is the gate keeper of the technique, is not robust with respect to the polynomial degree; more precisely, the CFL number that is required to maintain the invariant-domain property decreases very fast as the polynomial degree increases, and the method becomes more and more diffusive as the polynomial degree increases. This phenomenon is also reported in [23, §3.3]. It is also shown in Quezada De Luna [2, S4.2.2] and Anderson et al. [23, §3.3] that Bernstein finite elements behave far better than Lagrange elements in this respect. We have worked on this problem since the observations made in Alrashed [1] and Quezada De Luna [2]. It was clear from the start that a hierarchical decomposition of the space approximation had to be done, but the main roadblock on the way that made progresses slow was to have the low-order method and the high-order method to carry exactly the same mass. A solution to this problem for fourth-order finite differences is proposed in [24]. It is shown therein that the high-order fluxes can be recombined in a conservative manner on the low-order stencil. Building on an original idea from [4], we have found a reasonable solution to the conservation problem for finite elements in 2019, and it is the objective of this paper to expose this solution. In addition to have the stencil of the low-order method only depend on the next neighbors, the key idea is to slightly modify the fluxes of the low-order method so that it carries exactly the same mass as the high-order method while still being conservative and consistent (in the spirit of the so-called residual distribution technique by [4]). The proposed technique is robust with respect to the polynomial degree and is exactly mass preserving on patches.

The paper is organized as follows. We describe the low-order and the high-order space approximation setting in Section 2. The method that is high-order accurate in space is described in Section 3. The low-order method is described in Section 4. The limiting operation ensuring that the method combining the high-order approximation and the low-order one is invariant-domain preserving is described in Section 5. The proposed approach involves a node-based limiting and a cell-based limiting. The method is numerically illustrated in Section 6. Technicalities are collected in the Appendices A to C.

2. Space approximation

The goal of this section is to describe the setting for the space approximation. We restrict ourselves to continuous finite elements. We denote by $\mathbb{P}_k$ and $\mathbb{Q}_k$ the (real) vector spaces composed of the d-variate polynomials of degree at most k and of partial degree at most k, respectively.

2.1. Motivation

One important property of the finite-element-based invariant-domain preserving low-order technique introduced in [3,15,16] is that nowhere in the theory there is a theoretical upper limit on the polynomial degree. In principle, the method can be implemented with any polynomial degree. But as observed in the theses of [1, §3.3.2.1] and [2, §4.2.2], the low-order invariant-domain preserving method is not robust with respect to the polynomial degree. The key reason is that the size of the stencil of the method grows with the polynomial degree. This in turn makes the CFL number that is required to maintain the invariant-domain property decrease as the polynomial degree increases, and the method becomes more and more diffusive. This phenomenon is numerically illustrated in Fig. 4 in Section 4.1. The purpose of the paper is to introduce a hierarchical decomposition of the space approximation to address this problem akin to what is done in [4,8,13,25]. We focus in this paper on continuous Lagrange finite elements.

2.2. High-order finite element setting

Using Ciarlet's notation, we consider a (high-order) reference Lagrange or Bernstein finite element $(\widehat{K},\widehat{P}^{\mathrm{H}},\widehat{\Sigma}^{\mathrm{H}})$. The superscript $^{\mathrm{H}}$ is meant to remind us that the vector space $\widehat{P}^{\mathrm{H}}$ is composed of high-order polynomials. The shape functions of the reference element $(\widehat{K},$

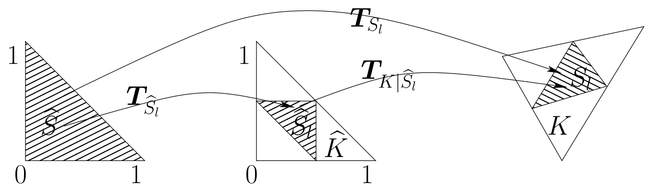


Fig. 1. Let $K \in \mathscr{T}_h$ and $S_l \in \mathscr{T}^{\mathrm{s}}_K$. $\boldsymbol{T}_{\hat{S}_l}$ maps the reference element $\widehat{S}$ to $\widehat{S}_l \subset \widehat{K}$. $\boldsymbol{T}_{K|\hat{S}_l}$ (restriction of $\boldsymbol{T}_K$ to $\widehat{S}_l$) maps $\widehat{S}_l$ to $S_l \subset K$. Then $\boldsymbol{T}_{S_l} := \boldsymbol{T}_{K|\hat{S}_l} \circ \boldsymbol{T}_{\hat{S}_l}$.

$\widehat{P}^{\mathrm{H}}, \widehat{\Sigma}^{\mathrm{H}})$ are denoted $\{\widehat{\theta}^{\mathrm{H}}_n\}_{n\in\hat{\mathscr{N}}}$. When $\{\widehat{\theta}^{\mathrm{H}}_n\}_{n\in\hat{\mathscr{N}}}$ are Lagrange shape functions, the corresponding reference Lagrange nodes are denoted $\{\widehat{\boldsymbol{a}}_n\}_{n\in\hat{\mathscr{N}}}$ (here $\widehat{\mathscr{N}}$ is the index set enumerating the reference shape functions). When $\{\widehat{\theta}^{\mathrm{H}}_n\}_{n\in\hat{\mathscr{N}}}$ are Bernstein shape functions, the domain nodes are denoted $\{\widehat{\boldsymbol{a}}_n\}_{n\in\hat{\mathscr{N}}}$. Recall that $dim\widehat{P}^{\mathrm{H}} = \mathrm{card}(\widehat{\mathscr{N}})$.

Let $(\mathscr{T}_h)_{h\in\mathscr{H}}$ be a shape-regular sequence of matching meshes, where $\mathscr{H}$ is a countable set with 0 as unique accumulation point; the index h refers to the typical meshsize of the cells in $\mathscr{T}_h$. We then introduce the continuous high-order finite element space

$$P^{\mathrm{H}}(\mathscr{T}_h) := \left\{ v \in C^0(D;\mathbb{R}) \middle| v_{|K} \circ T_K \in \widehat{P}^{\mathrm{H}}, \quad \forall K \in \mathscr{T}_h \right\}, \tag{2.1}$$

where $T_K : \widehat{K} \to K$ is the bijective geometric transformation that maps the reference element $\widehat{K}$ to the current element K. For the sake of simplicity, we assume in the entire paper that the geometric transformations are affine. The high-order approximation in space of the solution of (1.1) will be done with the space $\boldsymbol{P}^{\mathrm{H}}(\mathscr{T}_h) := (P^{\mathrm{H}}(\mathscr{T}_h))^m$. The global shape functions in $P^{\mathrm{H}}(\mathscr{T}_h)$ are denoted by $\{\varphi^{\mathrm{H}}_i\}_{i\in\mathscr{V}}$. Recall that these functions form a basis of $P^{\mathrm{H}}(\mathscr{T}_h)$, i.e., $dim(P^{\mathrm{H}}(\mathscr{T}_h)) = \mathrm{card}(\mathscr{V})$. We denote by $\mathrm{j} : \mathscr{T}_h \times \widehat{\mathscr{N}} \longrightarrow \mathscr{V}$ the connectivity array, which we recall is defined such that

$$\varphi^{\mathrm{H}}_{i|K} := \begin{cases} \widehat{\theta}^{\mathrm{H}}_n \circ \boldsymbol{T}_K^{-1} & \text{if there exists}(n,K) \in \widehat{\mathscr{N}} \times \mathscr{T}_h \text{s.t.} i = \mathrm{j}(n,K) \\ 0 & \text{otherwise.} \end{cases} \tag{2.2}$$

For all $i \in \mathscr{V}$ and all $K \in \mathscr{T}_h$, we set

$$\mathscr{I}(i) := \left\{ j \in \mathscr{V} \middle| \varphi^{\mathrm{H}}_i \varphi^{\mathrm{H}}_j / \equiv 0 \right\}, \qquad \mathscr{I}(K) := \left\{ i \in \mathscr{V} \middle| \varphi^{\mathrm{H}}_{i|K} / \equiv 0 \right\}, \tag{2.3a}$$

$$\mathscr{T}(i) := \left\{ K \in \mathscr{T}_h \middle| \varphi^{\mathrm{H}}_{i|K} / \equiv 0 \right\}. \tag{2.3b}$$

We refer to $\mathscr{I}(i)$ as the stencil of the shape function φ^{H}_i. We introduce the nodes $\{\boldsymbol{a}_i\}_{i\in\mathscr{V}}$ in D such that $\boldsymbol{a}_i = T_K(\widehat{\boldsymbol{a}}_n)$ for all $K \in \mathscr{T}(i)$, where $n \in \widehat{\mathscr{N}}$ is such that $i = \mathrm{j}(n,K)$. Recall that $\varphi^{\mathrm{H}}_j(\boldsymbol{a}_i) = \delta_{ij}$.

Finally, for all $i \in \mathscr{V}$ and all $K \in \mathscr{T}(i)$, we define the following quantities which play an important role in the rest of the paper:

$$m_{ij} := \int_D \varphi^{\mathrm{H}}_i(\boldsymbol{x})\varphi^{\mathrm{H}}_j(\boldsymbol{x})\,\mathrm{d}x, \boldsymbol{c}^{\mathrm{H}}_{ij} := \int_K \varphi^{\mathrm{H}}_j \nabla \varphi^{\mathrm{H}}_i\,\mathrm{d}x, \tag{2.4a}$$

$$m_j := \sum_{i\in\mathscr{V}} m_{ij} = \int_D \varphi^{\mathrm{H}}_j(\boldsymbol{x})\,\mathrm{d}x, m^K_j := \int_K \varphi^{\mathrm{H}}_j(\boldsymbol{x})\,\mathrm{d}x. \tag{2.4b}$$

Assumption 2.1. For all $i \in \mathscr{V}$ we have $m_i \geq 0$.

Notice that Assumption 2.1 holds if $m^{\hat{K}}_n := \int_{\hat{K}} \widehat{\theta}^{\mathrm{H}}_n\,\mathrm{d}x \geq 0$ for all $n \in \widehat{\mathscr{N}}$. In particular, this property holds for Lagrange finite elements on triangles up to degree 3 in two dimensions. It holds for all the Lagrange finite elements on quadrangles and hexahedrons. It holds for all Bernstein finite elements.

2.3. Multiscale structure

We now assume for simplicity that the reference element can be subdivided into sub-cells by connecting the reference Lagrange (or domain) points $\{\widehat{\boldsymbol{a}}_n\}_{n\in\hat{\mathscr{N}}}$ so that each sub-cell contains exactly the same number of Lagrange (or domain) points. Let $\{\widehat{S}_l\}_{l\in\hat{\mathscr{L}}}$ be the enumerated collection of sub-cells in question, and let us denote $\mathscr{T}^{\mathrm{s}}_{\hat{K}} := \{\widehat{S}_l\}_{l\in\hat{\mathscr{L}}}$ the sub-mesh of $\widehat{K}$ thus formed (here $\widehat{\mathscr{L}}$ is the index set corresponding to the enumeration in question). The superscript $^{\mathrm{s}}$ is meant to remind us that we are dealing with subdivided entities (cells or degrees of freedom). For all $l \in \widehat{\mathscr{L}}$, we introduce the index set $\widehat{\mathscr{N}}_l \subsetneq \widehat{\mathscr{N}}$ so that $\{\widehat{\boldsymbol{a}}_n\}_{n\in\hat{\mathscr{N}}_l}$ is the set of the reference Lagrange (or domain) points that belong to $\widehat{S}_l$.

To make the above construction more precise, we assume that there exists a reference cell $\widehat{S}$ so that for each sub-cell $\widehat{S}_l \in \mathscr{T}^{\mathrm{s}}_{\hat{K}}$ there

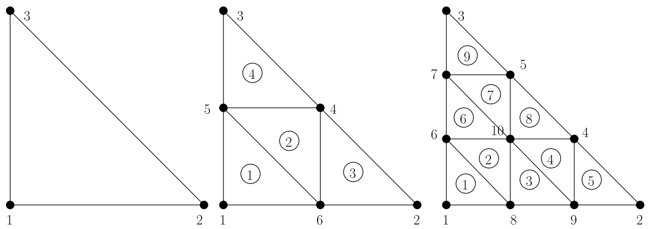


Fig. 2. Degrees of freedom (black dots) and sub-triangle enumeration (circled numbers) for two-dimensional $\mathbb{P}_1$, $\mathbb{P}_2$, and $\mathbb{P}_3$ Lagrange elements.

exists an affine geometric transformations $\boldsymbol{T}_{\widehat{S}_l}$ so that $\widehat{S}_l = \boldsymbol{T}_{\widehat{S}_l}(\widehat{S})$; see Fig. 1. (It happens in general that $\widehat{S} = \widehat{K}$, but this is not necessary; for instance, one could subdivide the unit square into triangles. See [14] where mixed subdivisions are considered.) We finally assume that there exists a set of points in $\widehat{S}$, say $\{\widehat{z}_{n^{\mathrm{s}}}\}_{n^{\mathrm{s}}\in\widehat{\mathscr{N}}^{\mathrm{s}}}$, so that for all $n \in \widehat{\mathscr{N}}$ there exist $l \in \widehat{\mathscr{L}}$ and $n^{\mathrm{s}} \in \widehat{\mathscr{N}}^{\mathrm{s}}$ so that $\widehat{\boldsymbol{a}}_n = \boldsymbol{T}_{\widehat{S}_l}(\widehat{z}_{n^{\mathrm{s}}})$. We formalize this property by introducing the reference hierarchical connectivity array $\widehat{\mathrm{j}} : \widehat{\mathscr{N}}^{\mathrm{s}} \times \widehat{\mathscr{L}} \to \widehat{\mathscr{N}}$ such that $\boldsymbol{T}_{\widehat{S}_l}(\widehat{z}_{n^{\mathrm{s}}}) = \widehat{\boldsymbol{a}}_{\widehat{\mathrm{j}}(n^{\mathrm{s}},l)}$ for all $n^{\mathrm{s}} \in \widehat{\mathscr{N}}^{\mathrm{s}}$ and all $l \in \widehat{\mathscr{L}}$. In this paper we adopt the increasing vertex-index enumeration; that is, $(n_1 \le n_2) \Leftrightarrow (\widehat{\mathrm{j}}(n_1,l) \le \widehat{\mathrm{j}}(n_2,l))$ for all $n_1, n_2 \in \widehat{\mathscr{N}}^{\mathrm{s}}$ and all $l \in \widehat{\mathscr{L}}$. We finally assume that the above construction satisfies the following property:

Assumption 2.2. There exists a polynomial space $\widehat{P}^{\mathrm{L}}$ and a set of linear forms $\widehat{\Sigma}^{\mathrm{L}}$ such that $\mathbb{P}_1 \subset \widehat{P}^{\mathrm{L}}$ and $(\widehat{S}, \widehat{P}^{\mathrm{L}}, \widehat{\Sigma}^{\mathrm{L}})$ is a Lagrange finite element based on the Lagrange nodes $\{\widehat{z}_{n^{\mathrm{s}}}\}_{n^{\mathrm{s}}\in\widehat{\mathscr{N}}^{\mathrm{s}}}$, i.e., $\widehat{\sigma}_{n^{\mathrm{s}}}(\widehat{p}) = \widehat{p}(\widehat{z}_{n^{\mathrm{s}}})$, for all $\widehat{\sigma}_{n^{\mathrm{s}}} \in \widehat{\Sigma}^{\mathrm{L}}$. We denote by $\{\widehat{\theta}^{\mathrm{L}}_{n^{\mathrm{s}}}\}_{n^{\mathrm{s}}\in\widehat{\mathscr{N}}^{\mathrm{s}}}$ the corresponding reference shape functions.

This assumption means that the cells obtained after subdivision allow for piecewise linear interpolation.

Examples of subdivisions and enumerations for the unit simplex in two dimensions are shown in Fig. 2. For instance, for the triangular $\mathbb{P}_3$ finite element shown in the rightmost panel of Fig. 2 we have $\widehat{\mathrm{j}}(1,1) = 1, \widehat{\mathrm{j}}(2,1) = 6, \widehat{\mathrm{j}}(3,1) = 8$, and $\widehat{\mathrm{j}}(1,3) = 8, \widehat{\mathrm{j}}(2,3) = 9, \widehat{\mathrm{j}}(3,3) = 10$. The arrays $\widehat{\mathrm{j}}$ for $\mathbb{P}_2$ and $\mathbb{P}_3$ finite elements in two dimensions are given in Table 8. (Notice that we do not impose any restriction on the sign of the determinant of the geometric transformations $\boldsymbol{T}_{\widehat{S}_l}$, $l \in \widehat{\mathscr{L}}$.)

2.4. Low-order finite element space

All the mesh cells K in $\mathscr{T}_h$ are subdivided as explained Section 2.3; i.e., all the sub-cells of K are images by $\boldsymbol{T}_K$ of the sub-cells $\{\widehat{S}_l\}_{l\in\widehat{\mathscr{L}}}$. The collection of the sub-cells of K (see Fig. 3) is denoted

$$\mathscr{T}^{\mathrm{s}}_K := \{\boldsymbol{T}_K(\widehat{S}_l)\}_{l\in\widehat{\mathscr{L}}}. \tag{2.5}$$

We denote by $\mathscr{T}^{\mathrm{s}}_h$ the mesh obtained by subdividing all the cells. The subdivision process guarantees that the sequence $(\mathscr{T}^{\mathrm{s}}_h)_{h\in\mathscr{H}}$ is shape-regular. Referring to Fig. 1 for an illustration, we introduce the following notation for all $S_l \in \mathscr{T}^{\mathrm{s}}_K$:

$$\boldsymbol{T}_{S_l} := \boldsymbol{T}_{K|\widehat{S}_l} \circ \boldsymbol{T}_{\widehat{S}_l}. \tag{2.6}$$

Recalling that $(\widehat{S}, \widehat{P}^{\mathrm{L}}, \widehat{\Sigma}^{\mathrm{L}})$ is a Lagrange finite element (see Assumption 2.2), we define the corresponding low-order Lagrange finite element space

$$P^{\mathrm{L}}(\mathscr{T}^{\mathrm{s}}_h) := \left\{ v \in C^0(D;\mathbb{R}) \middle| v_{|S} \circ \boldsymbol{T}_S \in \widehat{P}^{\mathrm{L}}, \quad \forall S \in \mathscr{T}^{\mathrm{s}}_h \right\}. \tag{2.7}$$

We introduce the connectivity array $\mathrm{j}^{\mathrm{L}} : \widehat{\mathscr{N}}^{\mathrm{s}} \times \mathscr{T}^{\mathrm{s}}_h \to \mathscr{V}$ defined by setting

$$\mathrm{j}^{\mathrm{L}}(n^{\mathrm{s}}, S_l) := \mathrm{j}(\widehat{\mathrm{j}}(n^{\mathrm{s}}, l), K), \tag{2.8}$$

for all $n^{\mathrm{s}} \in \widehat{\mathscr{N}}^{\mathrm{s}}$, all $K \in \mathscr{T}_h$, and all $S_l \in \mathscr{T}^{\mathrm{s}}_K$. We denote by $\{\varphi^{\mathrm{L}}_i\}_{i\in\mathscr{V}}$ the global Lagrange shape functions of $P^{\mathrm{L}}(\mathscr{T}^{\mathrm{s}}_h)$. The enumeration is done so that

$$\varphi^{\mathrm{L}}_{i|S} := \begin{cases} \widehat{\theta}^{\mathrm{L}}_{n^{\mathrm{s}}} \circ \boldsymbol{T}_S^{-1} & \text{if there exists}(n^{\mathrm{s}}, S) \in \widehat{\mathscr{N}} \times \mathscr{T}^{\mathrm{s}}_h \text{s.t.} i = \mathrm{j}^{\mathrm{L}}(n^{\mathrm{s}}, S) \\ 0 & \text{otherwise.} \end{cases} \tag{2.9}$$

Lemma 2.3. *For all $i, j \in \mathscr{V}$, we have $\varphi^{\mathrm{L}}_i(\boldsymbol{a}_j) = \varphi^{\mathrm{H}}_i(\boldsymbol{a}_j) = \delta_{ij}$.*

Proof. This is a consequence of (2.2), (2.8), and (2.9). □

We are going to make use of the following notation for all $T \in \mathscr{T}^{\mathrm{s}}_h$, all $i \in \mathscr{V}$ and all $K \in \mathscr{T}(i)$:

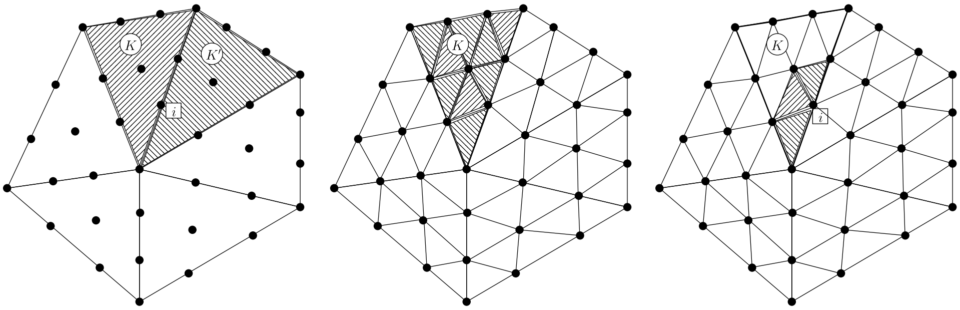


Fig. 3. Left panel: Definition of $\mathscr{T}(i)$ (shaded cells). Center panel: Definition of $\mathscr{T}^{\rm s}_K$ (shaded sub-cells). Right panel: Definition of $\mathscr{T}^{\rm s}_{K,i}$ (shaded sub-cells).

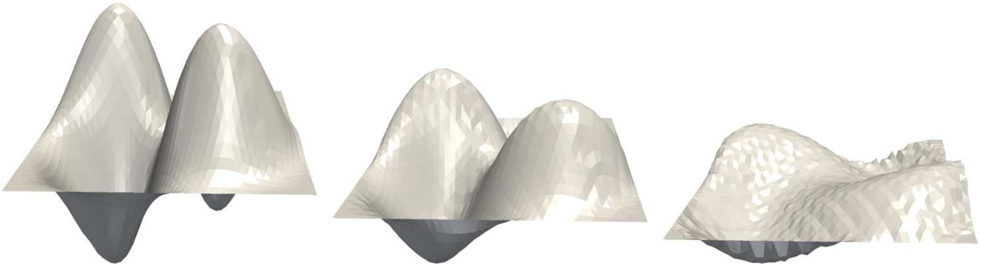


Fig. 4. Loss of robustness of the low-order solution w.r.t. the polynomial degree when using the full high-order stencil. Left: $\mathbb{P}_1$, 1927 nodes; center: $\mathbb{P}_2$, 1945 nodes; Right: $\mathbb{P}_3$, 1888 nodes.

$$\mathscr{I}^{\rm L}(i) := \left\{ j \in \mathscr{V} \middle| \varphi^{\rm L}_i \varphi^{\rm L}_j \not\equiv 0 \right\}, \qquad \mathscr{I}^{\rm L}(S) := \left\{ i \in \mathscr{V} \middle| \varphi^{\rm L}_{i|S} \not\equiv 0 \right\}, \tag{2.10a}$$

$$\mathscr{T}^{\rm s}_{K,i} := \left\{ S \in \mathscr{T}^{\rm s}_K \middle| \varphi^{\rm L}_{i|S} \not\equiv 0 \right\}. \tag{2.10b}$$

Notice that $\mathscr{I}^{\rm L}(S) = {\rm j}^{\rm L}(\widehat{\mathscr{N}}^{\rm s}, S)$ and $\mathscr{T}^{\rm s}_{K,i} = \{S \in \mathscr{T}^{\rm s}_K | \exists n^{\rm s} \in \widehat{\mathscr{N}}^{\rm s}, \quad {\rm j}^{\rm L}(n^{\rm s}, S) = i\}$. The notation introduced above is illustrated in Fig. 3.

For all $K \in \mathscr{T}_h$ and $S \in \mathscr{T}^{\rm s}_K$, we set $P^{\rm H}_K := \{\widehat{p} \circ \boldsymbol{T}_K^{-1} | \widehat{p} \in \widehat{P}^{\rm H}\}$ and $P^{\rm L}_S := \{\widehat{p} \circ \boldsymbol{T}_S^{-1} | \widehat{p} \in \widehat{P}^{\rm L}\}$. We then define the local low-order Lagrange interpolation operator $\Pi^{\rm L}_S : P^{\rm H}_K \to P^{\rm L}_S$ as follows:

$$\Pi^{\rm L}_S\left(u_{h|K}\right) = \sum_{n^{\rm s} \in \widehat{\mathscr{N}}^{\rm s}} {\sf U}_{{\rm j}^{\rm L}(n^{\rm s},S)} \varphi^{\rm L}_{{\rm j}^{\rm L}(n^{\rm s},S)|S}, \qquad \forall u_h := \sum_{i \in \mathscr{V}} {\sf U}_i \varphi^{\rm H}_i \in P^{\rm H}(\mathscr{T}_h). \tag{2.11}$$

Then we define the corresponding global low-order Lagrange interpolation operator $\Pi^{\rm L}_h : P^{\rm H}(\mathscr{T}_h) \to P^{\rm L}(\mathscr{T}^{\rm s}_h)$ by $\Pi^{\rm L}_h(u_h)_{|S} := \Pi^{\rm L}_S(u_{h|K})$ for all $K \in \mathscr{T}_h$ and all $S \in \mathscr{T}^{\rm s}_K$. Notice that (2.9) implies that

$$\Pi^{\rm L}_h(u_h) = \sum_{i \in \mathscr{V}} {\sf U}_i \varphi^{\rm L}_i, \qquad \forall u_h := \sum_{i \in \mathscr{V}} {\sf U}_i \varphi^{\rm H}_i \in P^{\rm H}(\mathscr{T}_h). \tag{2.12}$$

3. High-order method

For completeness, we introduce in this section the high-order approximation of (1.1). No originality is claimed here for we essentially paraphrase [3, §3.2]. The low-order method is introduced in Section 4.

3.1. High-order update

To properly describe the forward Euler time stepping technique, we denote by t^n the current time level, $n \in \mathbb{N}$, and let τ^n be the current time step; i.e., $t^{n+1} = t^n + \tau^n$. The time step may vary at each time level, but to simplify the (already heavy) notation we are going to drop the super-index ^{n} and use the symbol τ for the time step. We denote by $\boldsymbol{u}^n_h := \sum_{i \in \mathscr{V}} \mathbf{U}^n_i \varphi^{\rm H}_i \in \boldsymbol{P}^{\rm H}(\mathscr{T}_h)$ the high-order approximation of (1.1) at t^n, and by induction, we assume $\mathbf{U}^n_i \in \mathscr{A}$ for all $i \in \mathscr{V}$. We now briefly describe a way to create a high-order update $\boldsymbol{u}^{{\rm H},n+1}_h := \sum_{i \in \mathscr{V}} \mathbf{U}^{{\rm H},n+1}_i \varphi^{\rm H}_i \in \boldsymbol{P}^{\rm H}(\mathscr{T}_h)$ at the time level t^{n+1} using the forward Euler method.

Our first task is to construct a high-order approximation of $\nabla \cdot \mathbb{f}(\boldsymbol{u})$. When using finite elements it is natural to consider the Galerkin approximation thereof; that is, for all $i \in \mathscr{V}$ one computes $\int_D \nabla \cdot (\mathbb{f}(\boldsymbol{u}_h^n))\varphi_i^{\mathrm{H}}\,\mathrm{d}x$ using an appropriate quadrature. The problem with this expression is that although the values $\{\mathbf{U}_i^n\}_{i\in\mathscr{V}}$ are in $\mathscr{A}$ (by the induction assumption), there is no guarantee that $\boldsymbol{u}_h^n(\boldsymbol{x})$ is in $\mathscr{A}$ for all $\boldsymbol{x}$ in D (or at least for all the quadrature points of the quadrature approximating the integral. Recall that it is imperative that $\boldsymbol{u}_h^n(\boldsymbol{x}) \in \mathscr{A}$ for $\mathbb{f}(\boldsymbol{u}_h^n(\boldsymbol{x}))$ to make sense.) Notice that $\boldsymbol{u}_h^n(\boldsymbol{x}) \in \mathscr{A}$ for all $\boldsymbol{x} \in D$ for $\mathbb{P}_1$, $\mathbb{Q}_1$, and Bernstein finite elements, but this is not the case when using Lagrange elements of degree two and higher. We solve this difficulty as in [3, Eq. (3.7)] by using the Lagrange interpolant of the flux. Let $\Pi_h^{\mathrm{H}} : C^0(D;\mathbb{R}) \to P^{\mathrm{H}}(\mathscr{T}_h)$ be the Lagrange interpolation operator. With an obvious abuse of notation, we have $\Pi_h^{\mathrm{H}}(\mathbb{f}(\boldsymbol{u}_h^n)) := \sum_{i\in\mathscr{V}} \mathbb{f}(\boldsymbol{u}_h^n(\boldsymbol{a}_i))\varphi_i^{\mathrm{H}}$. For Lagrange elements we have $\boldsymbol{u}_h^n(\boldsymbol{a}_i) = \mathbf{U}_i^n$, and in this case

$$\Pi_h^{\mathrm{H}}\left(\mathbb{f}\left(\boldsymbol{u}_h^n\right)\right) := \sum_{i\in\mathscr{V}} \mathbb{f}\left(\mathbf{U}_i^n\right)\varphi_i^{\mathrm{H}}. \tag{3.1}$$

For Bernstein elements we have $\boldsymbol{u}_h^n(\boldsymbol{a}_i) = \mathbf{U}_i^n + \mathscr{O}(h^2)$. In this case (3.1) is no longer an identity by a second order approximation. The method has to be modified for higher-order Bernstein polynomials by invoking a change of basis. We omit the details to keep the presentation simple. The simulations reported in the paper are done only for $\mathbb{P}_2$ Bernstein elements. The term $\int_D \nabla \cdot (\mathbb{f}(\boldsymbol{u}))\varphi_i^{\mathrm{H}}\,\mathrm{d}x$ is then approximated by $\int_D \nabla \cdot (\Pi_h^{\mathrm{H}}(\mathbb{f}(\boldsymbol{u}_h^n)))\varphi_i^{\mathrm{H}}\,\mathrm{d}x$. Recalling the definition of $\boldsymbol{c}_{ij}^{\mathrm{H}}$ in (2.4a), the approximation takes the following form: $\sum_{j\in\mathscr{V}} \mathbb{f}(\mathbf{U}_j^n)\boldsymbol{c}_{ij}^{\mathrm{H}}$ (recall that $\mathbb{f}(\mathbf{U}_j^n)$ is a $m \times d$ matrix and $\boldsymbol{c}_{ij}^{\mathrm{H}}$ is $d \times 1$ column vector).

Recalling that $\boldsymbol{u}_h^{\mathrm{H},n+1} = \sum_{i\in\mathscr{V}} \mathbf{U}_i^{\mathrm{H},n+1}\varphi_i^{\mathrm{H}}$ and using the forward Euler technique for the time approximation, we define the high-order approximation of (1.1) by

$$\sum_{j\in\mathscr{V}} m_{ij}\frac{\mathbf{U}_j^{\mathrm{H},n+1} - \mathbf{U}_j^n}{\tau} = -\sum_{j\in\mathscr{I}(i)} \mathbb{f}\left(\mathbf{U}_j^n\right)\boldsymbol{c}_{ij}^{\mathrm{H}} + \sum_{j\in\mathscr{I}^{\mathrm{L}}(i)} d_{ij}^{\mathrm{H},n}\left(\mathbf{U}_j^n - \mathbf{U}_i^n\right). \tag{3.2}$$

The term $d_{ij}^{\mathrm{H},n}$ is some high-order graph viscosity that can be defined in many ways. It could be based on a smoothness indicator like in [26, Eq. (12)], [27, p. 6], [28, Thm. 4.1], [29, §4.3], [18, §6.2], or it could be based on an entropy viscosity commutator like in [17, §6.4]. The exact definition of $d_{ij}^{\mathrm{H},n}$ does not really matter at the moment provided it induces a high-order perturbation of (1.1) and satisfies $d_{ij}^{\mathrm{H},n} = d_{ji}^{\mathrm{H},n} \geq 0$. That is to say, we assume that the term $\sum_{j\in\mathscr{I}^{\mathrm{L}}(i)} d_{ij}^{\mathrm{H},n}(\mathbf{U}_j^n - \mathbf{U}_i^n)$ is of the same order as the consistency error of the approximation of the flux when the solution is smooth. Notice that in (3.2) we insists on the connectivity of $d_{ij}^{\mathrm{H},n}$ to be that of the low-order approximation, i.e., the summation is done over the low-order stencil $\mathscr{I}^{\mathrm{L}}(i)$ instead of the high-order stencil $\mathscr{I}(i)$. The way $d_{ij}^{\mathrm{H},n}$ is computed in the numerical simulations reported at the end of the paper is explained in Section 6.1. The following result clarifies the conservation properties of the high-order method.

Lemma 3.1. *The scheme* (3.2) *has the following conservation property:*

$$\int_D \boldsymbol{u}_h^{\mathrm{H},n+1}\,\mathrm{d}x = \int_D \boldsymbol{u}_h^n\,\mathrm{d}x - \tau\int_D \nabla \cdot \left(\Pi_h^{\mathrm{H}}\left(\mathbb{f}\left(\boldsymbol{u}_h^n\right)\right)\right)\mathrm{d}x. \tag{3.3}$$

Proof. Notice that $\sum_{i,j\in\mathscr{V}} m_{ij}\mathbf{U}_j^{\mathrm{H},n+1} = \int_D \sum_{j\in\mathscr{V}} \mathbf{U}_j^{\mathrm{H},n+1}\varphi_j^{\mathrm{H}}\,\mathrm{d}x = \int_D \boldsymbol{u}_h^{\mathrm{H},n+1}\,\mathrm{d}x$ and $\sum_{i,j\in\mathscr{V}} m_{ij}\mathbf{U}_j^n = \int_D \boldsymbol{u}_h^n\,\mathrm{d}x$. Moreover using $\sum_{i\in\mathscr{V}}\sum_{j\in\mathscr{I}^{\mathrm{L}}(i)} = \sum_{j\in\mathscr{V}}\sum_{i\in\mathscr{I}^{\mathrm{L}}(j)}$ and $d_{ij}^{\mathrm{H},n} = d_{ji}^{\mathrm{H},n}$, we have

$$\sum_{i\in\mathscr{V}}\sum_{j\in\mathscr{I}^{\mathrm{L}}(i)} d_{ij}^{\mathrm{H},n}\left(\mathbf{U}_j^n - \mathbf{U}_i^n\right) = \sum_{i\in\mathscr{V}}\sum_{j\in\mathscr{I}^{\mathrm{L}}(i)} d_{ij}^{\mathrm{H},n}\mathbf{U}_j^n - \sum_{i\in\mathscr{V}}\sum_{j\in\mathscr{I}^{\mathrm{L}}(i)} d_{ji}^{\mathrm{H},n}\mathbf{U}_i^n = 0.$$

Summing (3.2) over $i \in \mathscr{V}$ gives the assertion. □

4. Low-order method

We introduce in this section the low-order method that will be used as reference for limiting the high-order method. The low-order solution is approximated in space using the Lagrange finite element space $P^{\mathrm{L}}(\mathscr{T}_h^{\mathrm{s}})$.

4.1. Motivation

Since the technique described in [3, Eq (3.9)] is a priori independent of the polynomial degree of the approximation space $\boldsymbol{P}^{\mathrm{H}}(\mathscr{T}_h)$, we can in principle define the low-order update in $\boldsymbol{P}^{\mathrm{H}}(\mathscr{T}_h)$ as follows:

$$m_i\frac{\mathbf{U}_i^{\mathrm{H},n+1} - \mathbf{U}_i^n}{\tau} = -\sum_{j\in\mathscr{I}(i)} \mathbb{f}(\mathbf{U}_j)\boldsymbol{c}_{ij}^{\mathrm{H}} + \sum_{j\in\mathscr{I}(i)} d_{ij}^{\mathrm{L},n}\left(\mathbf{U}_j^n - \mathbf{U}_i^n\right), \tag{4.1}$$

where the stencil for the low-order viscosity $d_{ij}^{\mathrm{L},n}$ in (4.1) is $\mathscr{I}(i)$, i.e., the summation is done over $j \in \mathscr{I}(i)$ instead of $j \in \mathscr{I}^{\mathrm{L}}(i)$. A definition of $d_{ij}^{\mathrm{L},n}$ that makes the method invariant-domain preserving is

$$d_{ij}^{\mathrm{L},n} := max\Big(\lambda_{max}\Big(\boldsymbol{n}_{ij}, \mathbf{U}_i^n, \mathbf{U}_j^n\Big)\Big\|\boldsymbol{c}_{ij}^{\mathrm{H}}\Big\|_{\ell^2}, \lambda_{max}\Big(\boldsymbol{n}_{ji}, \mathbf{U}_j^n, \mathbf{U}_i^n\Big)\|\boldsymbol{c}_{ji}{}^{\mathrm{H}}\|_{\ell^2}\Big), \tag{4.2}$$

where $\lambda_{max}(\boldsymbol{n}_{ij}, \mathbf{U}_i^n, \mathbf{U}_j^n)$ is any upper bound on the maximum wave speed in the Riemann problem $\partial_t \boldsymbol{v} + \partial_x(\mathbb{f}(\boldsymbol{v})\boldsymbol{n}_{ij}) = 0$, where $\boldsymbol{n}_{ij} := \mathbf{c}_{ij}^{\mathrm{H}}/\Big\|\boldsymbol{c}_{ij}^{\mathrm{H}}\Big\|$, with left and right data $\mathbf{U}_i^n$ and $\mathbf{U}_j^n$ respectively (see e.g., [3, Eq (3.16)]). Using ideas, now classical, from [30, p. 163], [31], [32, p. 375], and [33, §5], it is established in [3, Th,. 4.1] and [18, Thm. 3.6] that using (4.2) yields a low-order method that is invariant-domain preserving (see Theorem 4.4 for a more precise statement).

As observed in the PhD theses of [1, §3.3.2.1] and [2, §4.2.2], the above scheme is not robust with the polynomial degree. This phenomenon is also reported [25, Tab. 1] and [8, Fig. 1]. To illustrate this observation we show in Fig. 4 the results of three two-dimensional simulations using the low-order method (4.1) with $\mathbb{P}_1$, $\mathbb{P}_2$ and $\mathbb{P}_3$ Lagrange finite elements on nonuniform triangular meshes. The problem solved is the linear transport equation $\partial_t u + \boldsymbol{\beta}\cdot\nabla u = 0$ with the divergence-free velocity $\boldsymbol{\beta}(\boldsymbol{x},t) = cos(\pi t)(-sin(2\pi x_2)sin^2(\pi x_1)\boldsymbol{e}_1 + sin(2\pi x_1)sin^2(\pi x_2)\boldsymbol{e}_2)$. The initial data is $u_0(\boldsymbol{x}) = 2^8(x_1(1-x_1)x_2(1-x_2))^2 sin(2\pi x_1)sin(2\pi x_2)$. The solution is periodic in time with period 1. Fig. 4 shows that graph of the low-order approximation at $t = 1$. The total number of degrees of freedom is approximately the same for the three cases to make the comparison fair (1927 for $\mathbb{P}_1$, 1945 for $\mathbb{P}_2$, and 1888 for $\mathbb{P}_3$). We observe that the quality of the approximation deteriorates as the polynomial degree of the approximation increases. (One can verify though that the asymptotic convergence rates are identical, see Section 6.2.) The reason for this behavior is that the cardinality of the high-order stencil $\mathscr{I}(i)$ used in (4.1) increases with the polynomial degree; actually, $\mathrm{card}(\mathscr{I}(i)) \sim k^d$ where k is the polynomial degree and d is the space dimension.

The purpose of the rest of this section is to introduce a low-order method that relies only on the next neighbors to be invariant-domain preserving; that is, we are going to construct a low-order flux based on the stencil $\mathscr{I}^{\mathrm{L}}(i)$ instead of $\mathscr{I}(i)$. The main difficulty in this exercise is to make sure that the low-order solution somehow carries the same mass as the high-order one.

4.2. Reducing the stencil to next neighbors

We are going to use an idea introduced in [4, Prop. 3.1]. Recall the low-order Lagrange interpolation operator $\Pi_S^{\mathrm{L}} : P_K^{\mathrm{H}} \rightarrow P_S^{\mathrm{L}}$ introduced in (2.11). Let us define $\widehat{\boldsymbol{P}}^{\mathrm{H}} := (\widehat{P}^{\mathrm{H}})^d$. We assume in the rest of the paper that the finite element setting described in Section 2 satisfies the following assumption.

Assumption 4.1. (i) For all $K \in \mathscr{T}_h$, there exists a collection of real numbers $\{\beta^S\}_{S\in\mathscr{T}_K^{\mathrm{s}}}$ such that the following holds true for every $\boldsymbol{p} \in \boldsymbol{P}_K^{\mathrm{H}} := \{\widehat{\boldsymbol{p}} \circ \boldsymbol{T}_K^{-1} | \widehat{\boldsymbol{p}} \in \widehat{\boldsymbol{P}}^{\mathrm{H}}\}$:

$$\int_K \nabla\cdot\boldsymbol{p}\,\mathrm{d}x = \sum_{S\in\mathscr{T}_K^{\mathrm{s}}} \beta^S \int_S \nabla\cdot\big(\Pi_S^{\mathrm{L}}(\boldsymbol{p})\big)\,\mathrm{d}x. \tag{4.3}$$

(ii) For all $K \in \mathscr{T}_h$ and all $S \in \mathscr{T}_K^{\mathrm{s}}$, there exists a set of real numbers $\{\alpha_j^S\}_{j\in\mathscr{I}^{\mathrm{L}}(S)}$ such that the following identities holds:

$$\forall j \in \mathscr{I}(K), \quad \sum_{S\in\mathscr{T}_{K,j}^{\mathrm{s}}} \alpha_j^S = 1, \qquad \forall S \in \mathscr{T}_K^{\mathrm{s}}, \quad \sum_{j\in\mathscr{I}^{\mathrm{L}}(S)} \alpha_j^S m_j^K = \beta^S|S|. \tag{4.4}$$

The existence of the coefficients $\{\beta^S\}_{S\in\mathscr{T}_K^{\mathrm{s}}}$ so that (4.3) holds true in dimensions 2 and 3 is established in [4, Prop. 3.1] and [4, Prop. 3.2]. The existence of the coefficients $\{\alpha_j^S\}_{j\in\mathscr{I}^{\mathrm{L}}(S)}$ so that identities (4.4) holds true is established in the Appendices B, C for various finite elements. It is also shown in Lemmas A.1,A.2 that it suffices that (4.3) and (4.4) hold on the reference element for these properties to hold for every mesh cell K in $\mathscr{T}_h$ if the mesh is affine.

We now want to approximate $\int_D \nabla\cdot(\mathbb{f}(\boldsymbol{u}_h))\varphi_i^{\mathrm{H}}\,\mathrm{d}x$ with the restriction that the approximation in question involves only the next neighbors of the Lagrange point $\boldsymbol{a}_i$, i.e., we want to involve only the sub-cells in $\bigcup_{K\in\mathscr{T}(i)}\mathscr{T}_{K,i}^{\mathrm{s}}$. Recalling that Π_h^{L} is the low-order Lagrange interpolation operator introduced in Section 3.1, we now consider $\Pi_h^{\mathrm{L}}(\mathbb{f}(\boldsymbol{u}_h)) := \sum_{i\in\mathscr{V}}\mathbb{f}(\boldsymbol{u}_h)(\boldsymbol{a}_i)\varphi_i^{\mathrm{L}}$. (Notice in passing that $\Pi_h^{\mathrm{L}}(\mathbb{f}(\boldsymbol{u}_h)) = \sum_{i\in\mathscr{V}}\mathbb{f}(\mathsf{U}_i)\varphi_i^{\mathrm{L}}$ because $\mathbb{f}(\boldsymbol{u}_h)(\boldsymbol{a}_i) = \mathbb{f}(\boldsymbol{u}_h(\boldsymbol{a}_i)) = \mathbb{f}(\mathsf{U}_i)$.) Since $\Pi_h^{\mathrm{L}}(\mathbb{f}(\boldsymbol{u}_h))$ is a second-order approximation of $\mathbb{f}(\boldsymbol{u}_h)$, $\nabla\cdot(\Pi_h^{\mathrm{L}}(\mathbb{f}(\boldsymbol{u}_h)))$ is a first-order approximation of $\nabla\cdot(\mathbb{f}(\boldsymbol{u}_h))$. Since for every cell K in $\mathscr{T}(i)$ and every sub-cell S in $\mathscr{T}_{K,i}^{\mathrm{s}}$, the quantity $\frac{1}{|S|}\int_S\nabla\cdot(\mathbb{f}(\boldsymbol{u}_h))\,\mathrm{d}x$ is an acceptable first-order approximation of $\nabla\cdot(\mathbb{f}(\boldsymbol{u}_h))(\boldsymbol{a}_i)$, it is also the case of $\frac{1}{|S|}\int_S\nabla\cdot(\Pi_h^{\mathrm{L}}(\mathbb{f}(\boldsymbol{u}_h)))\,\mathrm{d}x$. Then recalling that $\sum_{S\in\mathscr{T}_{K,i}^{\mathrm{s}}}\alpha_i^S = 1$, we also have

$$\nabla\cdot(\mathbb{f}(\boldsymbol{u}_h))(\boldsymbol{a}_i) = \sum_{S\in\mathscr{T}_{K,i}^{\mathrm{s}}}\alpha_i^S\frac{1}{|S|}\int_S\nabla\cdot\big(\Pi_h^{\mathrm{L}}(\mathbb{f}(\boldsymbol{u}_h))\big)\,\mathrm{d}x + \mathscr{O}(h). \tag{4.5}$$

Since the object of interest is $\int_D\nabla\cdot(\mathbb{f}(\boldsymbol{u}))\varphi_i^{\mathrm{H}}\,\mathrm{d}x$, and up to a quadrature based on the Lagrange points we have $\int_D\nabla\cdot(\mathbb{f}(\boldsymbol{u}))\varphi_i^{\mathrm{H}}\,\mathrm{d}x \approx \Big(\sum_{K\in\mathscr{T}(i)}m_i^K\Big)\nabla\cdot(\mathbb{f}(\boldsymbol{u}))(\boldsymbol{a}_i)$, we finally infer that

$$\int_D \nabla\cdot(\mathbb{f}(\boldsymbol{u}_h))\varphi^{\mathrm{H}}_i \,\mathrm{d}x = \sum_{K\in\mathscr{T}(i)} m^K_i \sum_{S\in\mathscr{T}^{\mathrm{s}}_{K,i}} \alpha^S_i \frac{1}{|S|}\int_S \nabla\cdot\left(\Pi^{\mathrm{L}}_h(\mathbb{f}(\boldsymbol{u}_h))\right)\mathrm{d}x + \mathscr{O}(h). \tag{4.6}$$

Proposition 4.2. *For every $i\in\mathscr{V}$, the approximations (4.5) and (4.6) are exact if $\mathbb{f}$ is linear and $\boldsymbol{u}_h$ is linear over the patch composed of the cells in $\mathscr{T}(i)$.*

Proof. Let $i\in\mathscr{V}$ and $S\in\mathscr{T}^{\mathrm{s}}_{K,i}$. Then $\Pi^{\mathrm{L}}_h(\mathbb{f}(\boldsymbol{u}_h))_{|S} = \sum_{j\in\mathscr{I}^{\mathrm{L}}(S)}\mathbb{f}(\boldsymbol{u}_h(\boldsymbol{a}_j))\varphi^{\mathrm{L}}_{j|S} = \mathbb{f}(\sum_{j\in\mathscr{I}^{\mathrm{L}}(S)}\boldsymbol{u}_h(\boldsymbol{a}_j)\varphi^{\mathrm{L}}_j)_{|S}$ by linearity of $\mathbb{f}$. Since $\boldsymbol{u}_h$ is linear over K, $\boldsymbol{u}_h$ is also linear over $S\subset K$; hence, $\sum_{j\in\mathscr{I}^{\mathrm{L}}(S)}\boldsymbol{u}_h(\boldsymbol{a}_j)\varphi^{\mathrm{H}}_{j|S} = \boldsymbol{u}_{h|S}$. This means that $\Pi^{\mathrm{L}}_h(\mathbb{f}(\boldsymbol{u}_h))_{|S} = \mathbb{f}(\boldsymbol{u}_h)_{|S}$. Then $\nabla\cdot(\Pi^{\mathrm{L}}_h(\mathbb{f}(\boldsymbol{u}_h)))_{|S} = \nabla\cdot(\mathbb{f}(\boldsymbol{u}_h))_{|S}$ and $\nabla\cdot(\Pi^{\mathrm{L}}_h(\mathbb{f}(\boldsymbol{u}_h)))_{|S} = \nabla\cdot(\mathbb{f}(\boldsymbol{u}_h))_{|K}$ because $\mathbb{f}$ is linear and $\boldsymbol{u}_{h|K}$ is linear. Hence,

$$\begin{aligned}\sum_{K\in\mathscr{T}(i)} m^K_i \sum_{S\in\mathscr{T}^{\mathrm{s}}_{K,i}} \alpha^S_i \frac{1}{|S|}\int_S \nabla\cdot\left(\Pi^{\mathrm{L}}_h(\mathbb{f}(\boldsymbol{u}_h))\right)\mathrm{d}x &= \sum_{K\in\mathscr{T}(i)} m^K_i(\nabla\cdot(\mathbb{f}(\boldsymbol{u}_h)))_{|K}\sum_{S\in\mathscr{T}^{\mathrm{s}}_{K,i}}\alpha^S_i \\ &= \sum_{K\in\mathscr{T}(i)}(\nabla\cdot(\mathbb{f}(\boldsymbol{u}_h)))_{|K}\int_K \varphi^{\mathrm{L}}_i\,\mathrm{d}x = \int_D \nabla\cdot(\mathbb{f}(\boldsymbol{u}_h))\varphi^{\mathrm{L}}_i\,\mathrm{d}x\end{aligned}$$

This concludes the proof. □ **Remark 4.3.** (*Second-order Accuracy*) Proposition 4.2 shows that the approximation (4.6) is actually formally second-order accurate in space since it is exact when $\boldsymbol{u}_h$ is piecewise linear (and continuous over D) and $\mathbb{f}$ is linear.

4.3. Low-order update

Recalling that $\{\varphi^{\mathrm{L}}_j\}_{j\in\mathscr{V}}$ are the low-order Lagrange shape functions, we introduce the following quantity $\boldsymbol{c}^{\mathrm{L}}_{ij}$ to simplify the notation:

$$\boldsymbol{c}^{\mathrm{L}}_{ij} := \sum_{K\in\mathscr{T}(i)} m^K_i \sum_{S\in\mathscr{T}^{\mathrm{s}}_{K,i}} \alpha^S_i \frac{1}{|S|}\int_S \nabla\varphi^{\mathrm{L}}_j\,\mathrm{d}x. \tag{4.7}$$

This definition implies that

$$\sum_{K\in\mathscr{T}(i)} m^K_i \sum_{S\in\mathscr{T}^{\mathrm{s}}_{K,i}} \alpha^S_i \frac{1}{|S|}\int_S \nabla\cdot\left(\Pi^{\mathrm{L}}_h(\mathbb{f}(\boldsymbol{u}_h))\right)\mathrm{d}x = \sum_{j\in\mathscr{I}^{\mathrm{L}}(i)}\mathbb{f}\left(\mathsf{U}^n_j\right)\boldsymbol{c}^{\mathrm{L}}_{ij}. \tag{4.8}$$

The low-order update is constructed in the high-order space $\boldsymbol{P}^{\mathrm{H}}_h(\mathscr{T}_h)$, i.e., we set $\boldsymbol{u}^{\mathrm{L},n+1}_h = \sum_{i\in\mathscr{V}}\mathsf{U}^{\mathrm{L},n+1}_i\varphi^{\mathrm{H}}_i$. Using the heuristics (4.6), we compute the low-order coefficients $(\mathsf{U}_i)_{i\in\mathscr{V}}$ as follows:

$$m_i\frac{\mathsf{U}^{\mathrm{L},n+1}_i - \mathsf{U}^n_i}{\tau} = -\sum_{j\in\mathscr{I}^{\mathrm{L}}(i)}\mathbb{f}\left(\mathsf{U}^n_j\right)\boldsymbol{c}^{\mathrm{L}}_{ij} + \sum_{j\in\mathscr{I}^{\mathrm{L}}(i)} d^{\mathrm{L},n}_{ij}\left(\mathsf{U}^n_j - \mathsf{U}^n_i\right). \tag{4.9}$$

This expression has the same form as in [3, Eq (3.9)]. Here $d^{\mathrm{L},n}_{ij}$ is the graph viscosity coefficient; the purpose of the graph viscosity is to make the method invariant-domain preserving. Let $\lambda_{max}(\boldsymbol{n}_{ij},\mathsf{U}^n_i,\mathsf{U}^n_j)$ be any upper bound on the maximum wave speed in the Riemann problem $\partial_t\boldsymbol{v} + \partial_x(\mathbb{f}(\boldsymbol{v})\boldsymbol{n}_{ij}) = 0$ with left and right data U^n_i and U^n_j respectively, where $\boldsymbol{n}_{ij} := \boldsymbol{c}^{\mathrm{L}}_{ij}/\left\|\boldsymbol{c}^{\mathrm{L}}_{ij}\right\|$.

Theorem 4.4. (*Local invariance*) *Let $n\geq 0$ and let $i\in\mathscr{V}$. Assume that*

$$d^{\mathrm{L},n}_{ij} := max\left(\lambda_{max}\left(\boldsymbol{n}_{ij},\mathsf{U}^n_i,\mathsf{U}^n_j\right)\left\|\boldsymbol{c}^{\mathrm{L}}_{ij}\right\|_{\ell^2},\lambda_{max}\left(\boldsymbol{n}_{ji},\mathsf{U}^n_j,\mathsf{U}^n_i\right)\left\|\boldsymbol{c}^{\mathrm{L}}_{ji}\right\|_{\ell^2}\right). \tag{4.10}$$

Let $\mathscr{B}$ be any convex subset of $\mathscr{A}$ that is invariant for (1.1). Assume that τ satisfies the CFL condition $1\geq 2\tau\sum_{j\in\mathscr{I}^{\mathrm{L}}(i)\backslash\{i\}}\frac{d^n_{ij}}{m_i}$ and $\mathsf{U}^n_j\subset\mathscr{B}$ for all $j\in\mathscr{I}^{\mathrm{L}}(i)$. Then $\mathsf{U}^{n+1}_i\in\mathscr{B}$. **Proof.** See [3, Th,. 4.1] and [18, Thm. 3.6]. □

4.4. Conservation

One important property of the approximation (4.6) is that it mimics the identity $\sum_{i\in\mathscr{I}(K)}\int_K\nabla\cdot(\mathbb{f}(\boldsymbol{u}))\varphi^{\mathrm{H}}_i\,\mathrm{d}x = \int_K\nabla\cdot(\mathbb{f}(\boldsymbol{u}))\,\mathrm{d}x$, which we recall is a consequence of the partition of unity. More precisely, we have the following result.

Lemma 4.5. (*Local conservation*) *Let Assumption 4.1 be met. The following holds true for all $\boldsymbol{u}_h\in\boldsymbol{P}^{\mathrm{H}}(\mathscr{T}_h)$:*

$$\sum_{i\in\mathscr{I}(K)} m^K_i \sum_{S\in\mathscr{T}^{\mathrm{s}}_{K,i}} \alpha^S_i \frac{1}{|S|}\int_S \nabla\cdot\left(\Pi^{\mathrm{L}}_h(\mathbb{f}(\boldsymbol{u}_h))\right)\mathrm{d}x = \int_K \nabla\cdot\left(\Pi^{\mathrm{H}}_h\mathbb{f}(\boldsymbol{u}_h)\right)\mathrm{d}x. \tag{4.11}$$

Proof. By rearranging the summations and using (4.3) and (4.4), we obtain

$$\begin{aligned}&\sum_{i\in\mathcal{I}(K)} m_i^K \sum_{S\in\mathcal{T}^{\mathrm{s}}_{K,i}} \alpha_i^S \frac{1}{|S|}\int_S \nabla\cdot\big(\Pi_h^{\mathrm{L}}(\mathbb{f}(\boldsymbol{u}_h))\big)\,\mathrm{d}x \\ &= \sum_{S\in\mathcal{T}^{\mathrm{s}}_{K}} \frac{1}{|S|}\int_S \nabla\cdot\big(\Pi_h^{\mathrm{L}}(\mathbb{f}(\boldsymbol{u}_h))\big)\,\mathrm{d}x \sum_{i\in\mathcal{I}(S)} m_i^K\alpha_i^S \\ = &\sum_{S\in\mathcal{T}^{\mathrm{s}}_{K}} \beta^S \int_S \nabla\cdot\big(\Pi_h^{\mathrm{L}}(\mathbb{f}(\boldsymbol{u}_h))\big)\,\mathrm{d}x = \int_K \nabla\cdot\big(\Pi_h^{\mathrm{H}}(\mathbb{f}(\boldsymbol{u}_h))\big)\,\mathrm{d}x. \quad \square\end{aligned}$$

Lemma 4.5 shows that the low-order flux and the high-order flux produce the same change of mass over each mesh cell in $\mathcal{T}_h$, just like for the residual distribution scheme described in [4] (see comment after Thm. 2.1 therein).

Contrary to the method proposed in [3], conservation in (4.9) does not arise from skew-symmetry properties. The graph viscosity still has the skew-symmetry property: $d_{ij}^{\mathrm{L},n}(\mathbf{U}_j^n - \mathbf{U}_i^n) = -d_{ji}^{\mathrm{L},n}(\mathbf{U}_i^n - \mathbf{U}_j^n)$ because $d_{ij}^{\mathrm{L},n} = d_{ji}^{\mathrm{L},n}$, but this is no longer the case for the flux terms $(\mathbb{f}(\mathbf{U}_j^n) + \mathbb{f}(\mathbf{U}_i^n))\boldsymbol{c}_{ij}^{\mathrm{L}}$ because $\boldsymbol{c}_{ij}^{\mathrm{L}} \neq \boldsymbol{c}_{ji}^{\mathrm{L}}$. The key reason for the loss of skew-symmetry is that the flux in (4.9) is approximated by using the low-order Lagrange basis functions, $\{\varphi_j^{\mathrm{L}}\}_{j\in\mathcal{V}}$, (i.e., $\Pi_h^{\mathrm{L}}(\mathbb{f}(\boldsymbol{u}_h)) = \sum_{i\in\mathcal{V}}\mathbb{f}(\mathbf{U}_i)\varphi_i^{\mathrm{L}}$).

Theorem 4.6. (*Conservation*) *Let Assumption 4.1 be met. The following conservation property holds for the low-order scheme (4.9):*

$$\int_D \boldsymbol{u}_h^{\mathrm{L},n+1}\,\mathrm{d}x = \int_D \boldsymbol{u}_h^n\,\mathrm{d}x - \tau\int_D \nabla\cdot\big(\Pi_h^{\mathrm{H}}\big(\mathbb{f}(\boldsymbol{u}_h^n)\big)\big)\,\mathrm{d}x. \tag{4.12}$$

Proof. Summing (4.9) over $i\in\mathcal{V}$ and using that $d_{ij}^{\mathrm{L},n} = d_{ji}^{\mathrm{L},n}$, we obtain

$$\sum_{i\in\mathcal{V}}\frac{m_i}{\tau}\mathbf{U}_i^{\mathrm{L},n+1} = \sum_{i\in\mathcal{V}}\frac{m_i}{\tau}\mathbf{U}_i^n - \sum_{K\in\mathcal{T}_h} m_i^K \sum_{S\in\mathcal{T}^{\mathrm{s}}_{K,i}} \alpha_i^S\frac{1}{|S|}\int_S\nabla\cdot\big(\Pi_h^{\mathrm{L}}(\mathbb{f}(\boldsymbol{u}_h))\big)\,\mathrm{d}x \quad + \sum_{i\in\mathcal{V}}\sum_{j\in\mathcal{V}\setminus\{i\}} d_{ij}^{\mathrm{L},n}\mathbf{U}_j^n - \sum_{i\in\mathcal{V}}\sum_{j\in\mathcal{V}\setminus\{i\}} d_{ji}^{\mathrm{L},n}\mathbf{U}_i^n.$$

Then invoking (4.11) from Lemma 4.5 (since Assumption 4.1 holds), we obtain

$$\sum_{i\in\mathcal{V}}\frac{m_i}{\tau}\mathbf{U}_i^{\mathrm{L},n+1} = \sum_{i\in\mathcal{V}}\frac{m_i}{\tau}\mathbf{U}_i^{\mathrm{L},n} - \sum_{K\in\mathcal{T}_h}\int_K\nabla\cdot\big(\Pi_h^{\mathrm{H}}(\mathbb{f}(\boldsymbol{u}_h))\big)\,\mathrm{d}x.$$

After observing that $\sum_{i\in\mathcal{V}} m_i\mathbf{U}_i^{\mathrm{L},n+1} = \sum_{i\in\mathcal{V}}\int_D\varphi_i^{\mathrm{H}}\mathbf{U}_i^{\mathrm{L},n+1}\,\mathrm{d}x = \int_D\boldsymbol{u}_h^{n+1}\,\mathrm{d}x$ and $\sum_{i\in\mathcal{V}} m_i\mathbf{U}_i^n = \int_D\boldsymbol{u}_h^n\,\mathrm{d}x$, the assertion follows readily. □ **Remark 4.7.** (*Local Conservation*) A local form of conservation making (4.12) more precise can be established by using the residual distribution technique discussed in [14,34].

5. Limiting

We describe in this section a limiting technique that guarantees that, once limited, the high-order update is invariant-domain preserving.

5.1. Maintaining conservation

Let $\mathcal{B}\subset\mathcal{A}$ be any convex set in the phase space. We recall that a function $\Psi:\mathcal{B}\to\mathbb{R}$ is said to be quasiconcave if for all finite sets $\{\theta_j\}_{j\in\mathcal{J}}$, $\{\mathbf{U}_j\}_{j\in\mathcal{J}}$ with $\theta_j\in[0,1]$, $\sum_{j\in\mathcal{J}}\theta_j = 1$ and $\mathbf{U}_j\in\mathcal{B}$ for all $j\in\mathcal{J}$, the following holds: $\Psi(\sum_{j\in\mathcal{J}}\theta_j\mathbf{U}_j)\geq min_{j\in\mathcal{J}}\Psi(\mathbf{U}_j)$. Let $i\in\mathcal{V}$ and let $\Psi_i:\mathcal{B}\to\mathbb{R}$ be a quasiconcave continuous function that is such that $\Psi_i(\mathbf{U}_i^{\mathrm{L},n+1})\geq 0$. Ways to construct quasiconcave functions satisfying this property are explained in [17, § 4] and [18, § 7.2] (including bound relaxation). Our goal is to correct the high-order update $\mathbf{U}_i^{\mathrm{H},n+1}$ by applying a limiting technique so that once limited the update $\mathbf{U}_i^{n+1}$ satisfies $\Psi_i(\mathbf{U}_i^{n+1})\geq 0$ as well. We explain in this section how this can be done while maintaining conservation.

In the spirit of the flux transport corrected literature, we proceed as in [20, §II] (see also [21,22], and [17,18]) by subtracting the low-order update (4.9) from the high-order update (3.2). We obtain

$$m_i\mathbf{U}_i^{\mathrm{H},n+1} = m_i\mathbf{U}_i^{\mathrm{L},n+1} - \sum_{j\in\mathcal{I}(i)}\tau\mathbb{f}\Big(\mathbf{U}_j^n\Big)\boldsymbol{c}_{ij}^{\mathrm{H}} + \sum_{j\in\mathcal{I}^{\mathrm{L}}(i)}\tau\mathbb{f}\Big(\mathbf{U}_j^n\Big)\boldsymbol{c}_{ij}^{\mathrm{L}} \quad + \sum_{j\in\mathcal{I}^{\mathrm{L}}(i)}\mathbf{A}_{ij}^n + \sum_{j\in\mathcal{I}(i)}\mathbf{C}_{ij}^n, \tag{5.1a}$$

$$\mathbf{A}_{ij}^n := \tau\big(d_{ij}^{\mathrm{H},n} - d_{ij}^{\mathrm{L},n}\big)\Big(\mathbf{U}_j^n - \mathbf{U}_i^n\Big), \tag{5.1b}$$

$$\mathbf{C}_{ij}^n := \big(m_i\delta_{ij} - m_{ij}\big)\Big(\mathbf{U}_j^{\mathrm{H},n+1} - \mathbf{U}_i^{\mathrm{H},n+1} - \Big(\mathbf{U}_j^n - \mathbf{U}_i^n\Big)\Big). \tag{5.1c}$$

We observe that $\mathbf{A}_{ij}^n = -\mathbf{A}_{ji}^n$ and $\mathbf{C}_{ij}^n = -\mathbf{C}_{ji}^n$ which are important property to maintain mass conservation. From now on we extend the definition of $\mathbf{A}_{ij}^n$ to all $j\in\mathcal{I}(i)$ by setting $\mathbf{A}_{ij}^n := \mathbf{0}$ if $j\not\in\mathcal{I}^{\mathrm{L}}(i)$, and we define

$$\mathbf{D}^n_{ij} := \mathbf{A}^n_{ij} + \mathbf{C}^n_{ij}, \qquad \forall j \in \mathscr{I}(i). \tag{5.2}$$

The key difference between the present setting and that in [17,18] is that the high-order flux $\sum_{j\in\mathscr{I}(i)} \mathbb{f}(\mathbf{U}^n_j)\boldsymbol{c}^{\mathrm{H}}_{ij}$ and the low-order flux $\sum_{j\in\mathscr{I}^{\mathrm{L}}(i)} \mathbb{f}(\mathbf{U}^n_j)\boldsymbol{c}^{\mathrm{L}}_{ij}$ do not cancel each other (unless $P^{\mathrm{H}}(\mathscr{T}_h) = P^{\mathrm{L}}(\mathscr{T}^{\mathrm{s}}_h)$). Actually, we have

$$-\tau \sum_{j\in\mathscr{I}(i)} \mathbb{f}\left(\mathbf{U}^n_j\right)\boldsymbol{c}^{\mathrm{H}}_{ij} + \tau \sum_{j\in\mathscr{I}^{\mathrm{L}}(i)} \mathbb{f}\left(\mathbf{U}^n_j\right)\boldsymbol{c}^{\mathrm{L}}_{ij} = \sum_{K\in\mathscr{T}(i)} \mathbf{B}^n_{i,K}. \tag{5.3}$$

with

$$\mathbf{B}^n_{i,K} := -\tau \int_K \nabla\cdot\left(\Pi^{\mathrm{H}}_h\left(\mathbb{f}\left(\boldsymbol{u}^n_h\right)\right)\right)\varphi^{\mathrm{H}}_i \,\mathrm{d}x + \tau m^K_i \sum_{T\in\mathscr{T}^{\mathrm{s}}_{K,i}} \frac{\alpha^S_i}{|S|}\int_S \nabla\cdot\left(\Pi^{\mathrm{L}}_h\left(\mathbb{f}\left(\boldsymbol{u}^n_h\right)\right)\right)\mathrm{d}x. \tag{5.4}$$

The following result tells us how limiting should be done to keep the method cell-wise conservative.

Lemma 5.1. *Let $K \in \mathscr{T}_h$ and let $\ell_K \in [0,1]$. Let $i \in \mathscr{V}$ and $j \in \mathscr{I}(i)$, and let $\ell_{ij} \in [0,1]$ with the assumption that $\ell_{ij} = \ell_{ji}$. Let $\mathbf{U}^{n+1}_i$ be defined by*

$$m_i\mathbf{U}^{n+1}_i = m_i\mathbf{U}^{\mathrm{L},n+1}_i + \sum_{j\in\mathscr{I}(i)\setminus\{i\}} \ell_{ij}\mathbf{D}^n_{ij} + \sum_{K\in\mathscr{T}(i)} \ell_K\mathbf{B}^n_{i,K}. \tag{5.5}$$

Then $\boldsymbol{u}^{n+1}_h$ and $\boldsymbol{u}^{\mathrm{L},n+1}_h$ carry the same mass, i.e., $\int_D \boldsymbol{u}^{n+1}_h \,\mathrm{d}x = \int_D \boldsymbol{u}^{\mathrm{L},n+1}_h \,\mathrm{d}x.$ **Proof.** Summing (5.5) over $i \in \mathscr{V}$, we obtain

$$\int_D \boldsymbol{u}^{n+1}_h \,\mathrm{d}x = \int_D \boldsymbol{u}^{\mathrm{L},n+1}_h \,\mathrm{d}x + \sum_{i\in\mathscr{V}}\sum_{j\in\mathscr{I}(i)} \ell_{ij}\mathbf{D}^n_{ij} + \sum_{i\in\mathscr{V}}\sum_{K\in\mathscr{T}(i)} \ell_K\mathbf{B}^n_{i,K}.$$

Since $\mathbf{D}^n_{ij} = -\mathbf{D}^n_{ji}$ and $\ell_{ij} = \ell_{ji}$, we have $\sum_{i\in\mathscr{V}}\sum_{j\in\mathscr{I}(i)}\ell_{ij}\mathbf{D}^n_{ij} = \mathbf{0}$. For the second term on the right-hand side $\boldsymbol{R} := \sum_{i\in\mathscr{V}}\sum_{K\in\mathscr{T}(i)}\ell_K\mathbf{B}^n_{i,K}$ we have

$$-\tau^{-1}\boldsymbol{R} = \sum_{K\in\mathscr{T}_h}\ell_K \sum_{i\in\mathscr{I}(K)} -\tau^{-1}\mathbf{B}^n_{i,K}$$

$$= \sum_{K\in\mathscr{T}_h}\ell_K\left[\int_K \nabla\cdot\left(\Pi^{\mathrm{H}}_h\left(\mathbb{f}\left(\boldsymbol{u}^n_h\right)\right)\right)\mathrm{d}x - \sum_{i\in\mathscr{I}(K)} m^K_i \sum_{S\in\mathscr{T}^{\mathrm{s}}_{K,i}} \frac{\alpha^S_i}{|S|}\int_S \nabla\cdot\left(\Pi^{\mathrm{L}}_h\left(\mathbb{f}\left(\boldsymbol{u}^n_h\right)\right)\right)\mathrm{d}x\right].$$

Then using (4.11) we obtain $\boldsymbol{R} = \mathbf{0}$. The conclusion follows readily. □ We now show how the degree of freedom limiters ℓ_{ij} and cell limiters ℓ_K can be estimated so that the update defined in (5.5) satisfies $\Psi_i(\mathbf{U}^{n+1}) \geq 0$. We consider two cases. We assume that Ψ is affine in the first one, then we address the general situation in the second case.

5.2. Affine functionals

Let us consider the simple case when Ψ_i is affine; that is, let us assume that there exist $\mathbf{J}_i \in \mathbb{R}^m$ and $\mathrm{b}_i \in \mathbb{R}$, s.t. $\Psi_i(\mathbf{V}) = \mathbf{J}_i \cdot \mathbf{V} + \mathrm{b}_i$. In this situation, one can apply the limiting technique inspired by [20] (see Eq. (10)–(13) therein and [19]). Let us give the details. Let $K \in \mathscr{T}(i)$ and

$$\mathscr{I}^-(i) := \left\{j \in \mathscr{I}(i) \middle| \mathbf{J}_i\cdot\mathbf{D}^n_{ij} < 0\right\}, \qquad \mathscr{T}^{-1}(i) := \left\{K \in \mathscr{T}(K) \middle| \mathbf{J}_i\cdot\mathbf{B}^n_{i,K} < 0\right\}. \tag{5.6a}$$

$$\mathbf{P}^-_i := \frac{1}{m_i}\left[\sum_{j\in\mathscr{I}^-(i)} \mathbf{D}^n_{ij} + \sum_{K\in\mathscr{T}^-(i)} \mathbf{B}^n_{i,K}\right], \qquad \ell^i := \frac{\min\left(\Psi_i\left(\mathbf{U}^{\mathrm{L},n+1}_i\right), -\mathbf{P}^-_i\right)}{-\mathbf{P}^-_i}, \tag{5.6b}$$

$$\ell^i_j := \begin{cases} \ell^i & \text{if}\, \mathbf{J}_i\cdot\mathbf{D}^n_{ij} < 0 \\ 1 & \text{otherwise,} \end{cases} \qquad \ell^i_K := \begin{cases} \ell^i & \text{if}\, \mathbf{J}_i\cdot\mathbf{B}^n_{i,K} < 0 \\ 1 & \text{otherwise,} \end{cases} \tag{5.6c}$$

$$\ell_{ij} := \min\left(\overset{i}{\underset{j}{\ell}}, \ell^j_i\right), \qquad \ell_K := \min_{i\in\mathscr{I}(K)} \ell^i_K. \tag{5.6d}$$

with the convention that $\ell^i = 1$ if $\mathscr{I}^-(i) = \emptyset$ and $\mathscr{T}^-(i) = \emptyset$.

Lemma 5.2. (*Affine limiting*) *Let $\mathbf{U}^{n+1}$ be defined in (5.5). Assume that (5.6) holds, then $\Psi_i(\mathbf{U}^{n+1}) \geq 0$.*

Proof. Using (5.5) and (5.6), and since Ψ_i is affine, we infer that

$$
\begin{aligned}
\Psi_i\big(\mathbf{U}^{n+1}\big) &= \Psi_i\big(\mathbf{U}^{\mathrm{L},n+1}\big) + \frac{1}{m_i}\sum_{j\in\mathcal{I}(i)} \ell_{ij}\mathbf{J}_i\cdot\mathbf{D}^n_{ij} + \frac{1}{m_i}\sum_{K\in\mathcal{T}(i)} \ell_K\mathbf{J}_i\cdot\mathbf{B}^n_{i,K} \\
&\geq \Psi_i\big(\mathbf{U}^{\mathrm{L},n+1}\big) + \frac{1}{m_i}\sum_{j\in\mathcal{I}^-(i)} \ell^i_j\mathbf{J}_i\cdot\mathbf{D}^n_{ij} + \frac{1}{m_i}\sum_{K\in\mathcal{T}^-(i)} \ell^i_K\mathbf{J}_i\cdot\mathbf{B}^n_{i,K} \\
&= \Psi_i\big(\mathbf{U}^{\mathrm{L},n+1}\big) + \ell^i\mathbf{P}^-_i \geq 0. \quad \square
\end{aligned}
$$

Remark 5.3. (*Stability with Respect to Roundoff Errors*) We have observed that the expression $\ell^i := (-\mathbf{P}^-_i)^{-1} min(\Psi_i(\mathbf{U}^{\mathrm{L},n+1}_i), -\mathbf{P}^-_i)$ is more stable with respect to roundoff errors than the identity $\ell^i := min(\Psi_i(\mathbf{U}^{\mathrm{L},n+1}_i)(-\mathbf{P}^-_i)^{-1}, 1)$ often found in the "flux corrected transport" literature.

5.3. *Non-affine functionals: Convex limiting*

Now we do not make the assumption that Ψ_i is affine. One cannot use Zalezak's technique consisting of grouping terms in positive and negative contributions. One possible way of dealing with this situation consists of adopting the convex limiting method introduced in [17, § 4.2] and [18, § 7], which is essentially a divide and conquer strategy. Let us give the details.

Lemma 5.4. *Let* $\mathbf{U}^{n+1}$ *be defined in* (5.5)*. Let* $\theta\in(0,1)$ *and let* $(\lambda_j)_{j\in\mathcal{I}(i)}$ *and* $(\mu_K)_{K\in\mathcal{T}(i)}$ *be real numbers in the open interval* $(0,1)$ *such that* $\sum_{j\in\mathcal{I}(i)\setminus\{i\}}\lambda_j = 1$ *and* $\sum_{K\in\mathcal{T}(i)}\mu_K = 1$*. Then,*

$$
\begin{gathered}
\Psi_i\big(\mathbf{U}^{n+1}\big) \geq \\
min\left(\min_{j\in\mathcal{I}(i)\setminus\{i\}}\Psi_i\left(\mathbf{U}^{\mathrm{L},n+1}_i + \frac{\ell_{ij}}{\theta\lambda_j m_i}\overset{n}{\mathbf{D}}_{ij}\right), \min_{K\in\mathcal{T}(i)}\Psi_i\left(\mathbf{U}^{\mathrm{L},n+1}_i + \frac{\ell_K}{(1-\theta)\mu_K m_i}\overset{n}{\mathbf{B}}_{i,K}\right)\right).
\end{gathered}
\tag{5.7}
$$

Proof. We have

$$
\begin{aligned}
\mathbf{U}^{n+1} &:= \mathbf{U}^{\mathrm{L},n+1}_i + \frac{1}{m_i}\sum_{j\in\mathcal{I}(i)\setminus\{i\}}\ell_{ij}\mathbf{D}^n_{ij} + \frac{1}{m_i}\sum_{K\in\mathcal{T}(i)}\ell_K\mathbf{B}^n_{i,K} \\
&= \theta\left(\sum_{j\in\mathcal{I}(i)\setminus\{i\}}\lambda_j\mathbf{U}^{\mathrm{L},n+1}_i + \frac{\ell_{ij}}{\theta m_i}\mathbf{D}^n_{ij}\right) + (1-\theta)\left(\sum_{K\in\mathcal{T}(i)}\mu_K\mathbf{U}^{\mathrm{L},n+1}_i + \frac{\ell_K}{(1-\theta)m_i}\mathbf{B}^n_{i,K}\right) \\
&= \sum_{j\in\mathcal{I}(i)\setminus\{i\}}\theta\lambda_j\left(\mathbf{U}^{\mathrm{L},n+1}_i + \frac{\ell_{ij}}{\theta\lambda_j m_i}\mathbf{D}^n_{ij}\right) + \sum_{K\in\mathcal{T}(i)}(1-\theta)\mu_K\left(\mathbf{U}^{\mathrm{L},n+1}_i + \frac{\ell_K}{(1-\theta)\mu_K m_i}\mathbf{B}^n_{i,K}\right)
\end{aligned}
$$

Then the assertion is a consequence of Ψ_i being quasiconcave. □ **Lemma 5.5.** *Let* $i\in\mathcal{V}$ *and* $\mathbf{P}\in\mathbb{R}^m$*. Assume that* $\mathbf{U}^{\mathrm{L},n+1}_i + \mathbf{P}$ *is in the domain of* Ψ_i*. Let* $\ell(i,\mathbf{P})\in[0,1]$ *be such that*

$$
\ell(i,\mathbf{P}) = \begin{cases} 1 & \text{if} \Psi_i\big(\mathbf{U}^{\mathrm{L},n+1}_i + \mathbf{P}\big) \geq 0, \\ max\{\ell\in[0,1] \,|\, \Psi_i\big(\mathbf{U}^{\mathrm{L},n+1}_i + \ell\mathbf{P}\big) \geq 0\} & \text{otherwise.} \end{cases}
\tag{5.8}
$$

Then $\Psi_i(\mathbf{U}^{\mathrm{L},n+1}_i + \ell\mathbf{P}) \geq 0$ *for every* $\ell\in[0,\ell(i,\mathbf{P})]$. **Proof.** Let us verify that the definition of $\ell(i,\mathbf{P})$ makes sense. We first observe that the set $\{\ell\in[0,1]\big|\Psi_i(\mathbf{U}^{\mathrm{L},n+1}_i + \ell\mathbf{P}) \geq 0\}$ is not empty because $\Psi_i(\mathbf{U}^{\mathrm{L},n+1}_i) \geq 0$. Then this non-empty bounded set must have a maximal element because Ψ_i is continuous. This established that $\ell(i,\mathbf{P})$ is well defined if $\Psi_i(\mathbf{U}^{\mathrm{L},n+1}_i + \mathbf{P}) < 0$. The definition of $\ell(i,\mathbf{P})$ is unambiguous if $\Psi_i(\mathbf{U}^{\mathrm{L},n+1}_i + \mathbf{P}) \geq 0$.

Let $L_0(\Psi_i) := \{\mathbf{U}\in\mathcal{B}|\Psi_i(\mathbf{U}) \geq 0\}$. Notice that the set $L_0(\Psi_i)$ is not empty because $\mathbf{U}^{\mathrm{L},n+1}_i \in L_0(\Psi_i)$. It is also is convex because Ψ_i is quasiconcave. Then for all $\ell\in[0,\ell^i_K]$ we have $\Psi_i(\mathbf{U}^{\mathrm{L},n+1}_i + \ell\mathbf{P}) \geq 0$ because $\mathbf{U}^{\mathrm{L},n+1}_i \in L_0(\Psi_i)$, $\mathbf{U}^{\mathrm{L},n+1}_i + \ell^i_K\mathbf{P}\in L_0(\Psi_i)$ and $L_0(\Psi_i)$ is convex. □

Remark 5.6. (*Line Search*) Computing the maximal element in the set $\{\ell\in[0,1]\big|\Psi_i(\mathbf{U}^{\mathrm{L},n+1}_i + \ell\mathbf{P}) \geq 0\}$ can be done by a line search when $\Psi_i(\mathbf{U}^{\mathrm{L},n+1}_i + \mathbf{P}) < 0$. The line search problem has a unique solution when Ψ_i is strictly quasiconcave since in this case the function $[0,1]\ni\ell\to\Psi_i(\mathbf{U}^{\mathrm{L},n+1}_i + \ell\mathbf{P})$ is strictly monotone decreasing and therefore the equation $\Psi_i(\mathbf{U}^{\mathrm{L},n+1}_i + \ell\mathbf{P}) = 0$ has a unique solution in (0, 1).

Theorem 5.7. *Let* θ*,* $(\lambda_j)_{j\in\mathcal{I}(i)}$*, and* $(\mu_K)_{K\in\mathcal{T}(i)}$ *be real number in* $(0,1)$ *as in* Lemma 5.4*. (i.e.,* $\sum_{j\in\mathcal{I}(i)\setminus\{i\}}\lambda_j = 1$ *and* $\sum_{K\in\mathcal{T}(i)}\mu_K = 1$*). Let*

$$
\ell_{ij} := min\left(\ell\left(i, \frac{1}{\theta\lambda_j m_i}\overset{n}{\mathbf{D}}_{ij}\right), \ell\left(j, \frac{1}{\theta\lambda_i m_i}\mathbf{D}^n_{ji}\right)\right), \quad \overset{i}{\ell}_K := \ell\left(i, \frac{1}{(1-\theta)\mu_K m_i}\mathbf{B}^n_{i,K}\right).
\tag{5.9}
$$

Then $\Psi_i(\mathbf{U}^{n+1}) \geq 0$. **Proof.** Apply Lemma 5.4 and Lemma 5.5. □

Remark 5.8. (*Parameter θ*) The purpose of the coefficients $\theta \in (0,1)$ is to transform the identity $\mathbf{U}^{n+1} := \mathbf{U}_i^{\mathrm{L},n+1} + \frac{1}{m_i}\sum_{j\in\mathscr{I}(i)\setminus\{i\}}\ell_{ij}\mathbf{D}_{ij}^n + \frac{1}{m_i}\sum_{K\in\mathscr{T}(i)}\ell_K\mathbf{B}_{i,K}^n$ into the following convex combination $\mathbf{U}^{n+1} := \theta\Big(\mathbf{U}_i^{\mathrm{L},n+1} + \frac{1}{\theta m_i}\sum_{j\in\mathscr{I}(i)\setminus\{i\}}\ell_{ij}\mathbf{D}_{ij}^n\Big) + (1-\theta)\Big(\mathbf{U}_i^{\mathrm{L},n+1} + \frac{1}{(1-\theta)m_i}\sum_{K\in\mathscr{T}(i)}\ell_K\mathbf{B}_{i,K}^n\Big)$. The free parameter θ can be used to balance the terms $\frac{1}{m_i}\sum_{j\in\mathscr{I}(i)\setminus\{i\}}\ell_{ij}\mathbf{D}_{ij}^n$ and $\frac{1}{m_i}\sum_{K\in\mathscr{T}(i)}\ell_K\mathbf{B}_{i,K}^n$. Assuming that the terms $\mathbf{D}_{ij}^n$ and $\mathbf{B}_{i,K}^n$ have all the same magnitude, then a good choice consists of setting θ so that $\frac{\mathrm{card}(\mathscr{I}(i))}{\theta} = \frac{\mathrm{card}(\mathscr{T}(i))}{1-\theta}$; that is, $\theta = \frac{\mathrm{card}(\mathscr{I}(i))}{\mathrm{card}(\mathscr{I}(i))+\mathrm{card}(\mathscr{T}(i))}$. But it is actually interesting to bias θ a little bit according to the magnitude of the coefficients $\mathbf{D}_{ij}^n$, $\mathbf{B}_{i,K}^n$. More precisely, let us denote by h_i the local meshsize at the node $\boldsymbol{a}_i$, and let us set $(\Delta\mathbf{U})_i = max_{j\in\mathscr{I}(i)}\Big\|\mathbf{U}_j^n - \mathbf{U}_i^n\Big\|_*$, where $\|\mathbf{V}\|_*$ is a norm in $\mathbb{R}^m$ that is dimensionally consistent (i.e., combines the various components of $\mathbf{V}$ in a way that is dimensionally coherent). Then one expects $\mathbf{D}_{ij}^n$ to scale like $\tau m_i h_i^{-1} f'_{i,max} \times (\Delta\mathbf{U})_i$ where $f'_{i,max}$ is an upper bound on the norm induced by $\|\cdot\|_*$ of the Jacobian of $\mathbb{f}$, whereas, owing to Proposition 4.2, one expects $\mathbf{B}_{i,K}^n$ to scale like $\tau m_i h_i^{-1} f''_{i,max} \times (\Delta\mathbf{U})_i^2$, where $f''_{i,max}$ is an upper bound on the induced norm of the Hessian of $\mathbb{f}$. Hence, it is reasonable to expect $\mathbf{B}_{i,K}^n$ to be far smaller than $\mathbf{D}_{ij}^n$ in regions where the solution is smooth. One can then use $\theta = \frac{\sum_{K\in\mathscr{T}(i)}\|\mathbf{D}_{ij}^n\|_*}{\sum_{j\in\mathscr{I}(i)}\|\mathbf{D}_{ij}^n\|_* + \sum_{K\in\mathscr{T}(i)}\|\mathbf{B}_{i,K}^n\|_*}$ or variations of this idea. Another way to proceed consists of replacing $\theta\lambda_j$ in (5.7) by $\frac{\|\mathbf{D}_{ij}^n\|_*}{\sum_{j\in\mathscr{I}(i)\setminus\{i\}}\|\mathbf{D}_{ij}^n\|_* + \sum_{K\in\mathscr{T}(i)}\|\mathbf{B}_{i,K}^n\|_*}$ and $(1-\theta)\mu_K$ by $\frac{\|\mathbf{B}_{i,K}^n\|_*}{\sum_{j\in\mathscr{I}(i)\setminus\{i\}}\|\mathbf{D}_{ij}^n\|_* + \sum_{K\in\mathscr{T}(i)}\|\mathbf{B}_{i,K}^n\|_*}$.

6. Numerical illustrations

In this section we briefly illustrate the performance of the method described in this paper.

6.1. Technical details

Two independent codes have been written to verify reproducibility. The first one, `Code 1`, does not use any particular software and is written in Fortran 2003. It is based on Lagrange elements on simplices and is dimension-independent. The second code, `Code 2`, uses the open-source finite element library FEniCS and is written in C++ and python, see e.g., [35]. The implementation in FEniCS is independent of the space dimension and the polynomial degree of the approximation. The numerical quadratures in both codes are chosen to be exact for the mass matrix. All the computations are done on simplices. The computations with $\mathbb{P}_1$ and $\mathbb{P}_3$ finite elements are done with the Lagrange bases. The computations with $\mathbb{P}_2$ finite elements are done with the Bernstein basis as the lumped mass matrix for continuous $\mathbb{P}_2$ Lagrange elements is singular. We only report tests for scalar conservation equations. Tests for the compressible Euler equations using the present method combined the time stepping techniques developed in [24] will be reported in the second part of this work.

The time stepping is done with the fourth-order five stages strong stability preserving explicit Runge–Kutta method with five stages (see [36, p. 522]). The time step is defined by the expression

$$\tau = s \times \mathrm{CFL} \times \min_{i\in\mathscr{I}} \frac{m_i}{\sum\limits_{j\in\mathscr{I}^{\mathrm{L}}(i)\setminus\{i\}} d_{ij}^{\mathrm{L},n}}, \tag{6.1}$$

with $d_{ij}^{\mathrm{L},n}$ defined in (4.2) and s is the number of stages of the Runge–Kutta method; here, we use $s = 5$. The high-order viscosity for scalar conservation equations is computed by estimating the entropy commutator $\int_D(\eta'(v)\nabla\cdot\mathbb{f}(v) - \mathbb{f}'(v)\nabla\eta(v))\varphi_i^{\mathrm{H}}\,\mathrm{d}x = 0$ with $\eta(v) = exp(v)$. This is done as follows:

$$X_i\big(\boldsymbol{u}_h^n\big) := m_i^{-1}\int_D \Pi_h^{\mathrm{H}}\big(\eta'\big(\boldsymbol{u}_h^n\big)\big)\nabla\cdot\big(\Pi_h^{\mathrm{H}}\big(\mathbb{f}\big(\boldsymbol{u}_h^n\big)\big)\big)\varphi_i^{\mathrm{H}}\,\mathrm{d}x, \tag{6.2a}$$

$$Y_i\big(\boldsymbol{u}_h^n\big) := m_i^{-1}\int_D \Pi_h^{\mathrm{H}}\big(\mathbb{f}'\big(\boldsymbol{u}^h\big)\big)\cdot\nabla\Pi_h^{\mathrm{H}}\big(\eta\big(\boldsymbol{u}_h^n\big)\big)\varphi_i^{\mathrm{H}}\,\mathrm{d}x, \tag{6.2b}$$

$$\alpha_i^n := \frac{\big|X_i\big(\boldsymbol{u}_h^n\big) - Y_i\big(\boldsymbol{u}_h^n\big)\big|}{|X_i(\boldsymbol{u}_h^n)| + |Y_i(\boldsymbol{u}_h^n)| + \epsilon \max\limits_{i\in\mathscr{V}}(|X_i(\boldsymbol{u}_h^n)| + |Y_i(\boldsymbol{u}_h^n)|)}, \qquad \epsilon = 10^{-2} \tag{6.2c}$$

$$d_{ij}^{\mathrm{H},n} := d_{ij}^{\mathrm{L},n} max\Big(\psi\Big(\underset{i}{\overset{n}{\alpha}}\Big), \psi\Big(\alpha_j^n\Big)\Big), \tag{6.2d}$$

By definition, the normalized residual α_i^n takes values in $[0,1]$. The numerator in the definition of α_i^n behaves like the truncation error of the Galerkin method. This quantity approximately behaves like $\mathscr{O}(h^k)$ where k is the polynomial degree of the approximation and h is the mesh size. Finally we set

$$d_{ij}^{\mathrm{H},n} := d_{ij}^{\mathrm{L},n} max\left(\psi\left(\frac{\alpha_i^n}{\alpha_0}\right),\quad \psi\left(\frac{\alpha_j^n}{\alpha_0}\right)\right), \tag{6.3}$$

where $\alpha_0 \in (0,1]$, the activation function ψ is defined as follows:

$$\psi(t) := 1 + (t-1)^2\left(6t^2 + 3t + 1\right)\left(t - 1 - (t-1)_+\right), \tag{6.4}$$

where $(x)_+ = (x + |x|)/2$. Notice that $\psi(0) = 0$, $\psi\left(\frac{1}{2}\right) = \frac{1}{2}$ and $\psi(t) = 1$ for all $t \in [\alpha_0, 1]$, and $\psi(t) \sim 10t^3 + \mathscr{O}(t^4)$. As a result, one recovers $d_{ij}^{\mathrm{H},n} = d_{ij}^{\mathrm{L},n}$ if the entropy commutator is larger than α_0, and $d_{ij}^{\mathrm{H},n} = \mathscr{O}(h^{3k}) \times d_{ij}^{\mathrm{L},n}$ otherwise. The numerical tests reported in this paper are done with $\alpha_0 = 0.4$. An activation function with the same purpose is used in [37, Eq. (8)]. Up to re-scaling, the activation function therein behaves like $\frac{1}{2}\left(1 + sin\left(\frac{\pi}{2}(t-1)\right)\right)$.

The limiting is done at each grid point at each time stage of the Runge–Kutta method as explained in Section 5.2 and the process is iterated four times. The local bounds are relaxed using the process explained in [17, §4.7] and we use the minmod version therein to average the estimate of the local second variation (see (4.11) and (4.13) in [17]). We now give the details for completeness and reproducibility. First we estimate the second variation of u_h^n at every grid point $i \in \mathscr{V}$ by setting

$$\Delta^2 \mathsf{U}_i^n := \frac{\sum_{j\in\mathscr{I}(i)\setminus\{i\}} \beta_{ij}\left(\mathsf{U}_j^n - \mathsf{U}_i^n\right)}{\sum_{j\in\mathscr{I}(i)\setminus\{i\}} \beta_{ij}}, \tag{6.5}$$

where the coefficients $\beta_{ij} := \int_D \nabla\varphi_i^{\mathrm{H}} \cdot \nabla\varphi_j^{\mathrm{H}}\,\mathrm{d}x$ are the entries of the high-order stiffness matrix associated with the weak form of the Laplace operator. This definition of β_{ij} guarantees that $\Delta^2\mathsf{U}_i^n = 0$ if u_h^n is locally linear on the support of φ_i. It is our experience that using the high-order stiffness matrix gives a better estimation of the local second variation than using the low-order one. Then we set

$$\widetilde{\Delta^2}\mathsf{U}_i^n := \operatorname{minmod}\left\{\Delta^2\mathsf{U}_j^n \,\middle|\, j \in \mathscr{I}^{\mathrm{L}}(i)\right\}, \tag{6.6}$$

where minmod of a finite set of real numbers is zero if there are two numbers of different sign in this set and is equal to the number whose absolute value is the smallest otherwise. Finally, using again the low-order stencil, we define $\mathsf{U}_i^{min,n} := min_{i\in\mathscr{I}^{\mathrm{L}}(i)}\mathsf{U}_i^n$ and $\mathsf{U}_i^{max,n} := max_{i\in\mathscr{I}^{\mathrm{L}}(i)}\mathsf{U}_i^n$. Notice that we use the low-order stencil to define $\widetilde{\Delta^2}\mathsf{U}_i^n$, $\mathsf{U}_i^{min,n}$ and $\mathsf{U}_i^{max,n}$. The rationale for using the low-order stencil in theses definitions is based on the observation that the time step restriction induced by the CFL condition implies that the domain of influence of the degree of freedom i is the support of the low-order shape function φ_i^{L}. To be certain that the global bounds are not violated, the relaxed bounds are not allowed to exceed the minimum and the maximum values of the initial data. More precisely denoting $\mathsf{U}^\flat = \operatorname{ess\,inf} u_0$ and $\mathsf{U}^\sharp = \operatorname{ess\,sup} u_0$, we set

$$\mathsf{U}_i^{min,n} := max\left(\overset{min,n}{\underset{i}{\mathsf{U}}} - \left|\widetilde{\Delta^2}\mathsf{U}_i^n\right|, \mathsf{U}^\flat\right) \tag{6.7a}$$

$$\mathsf{U}_i^{max,n} := min\left(\overset{max,n}{\underset{i}{\mathsf{U}}} + \left|\widetilde{\Delta^2}\mathsf{U}_i^n\right|, \mathsf{U}^\sharp\right). \tag{6.7b}$$

Strictly enforcing the global bounds is a stringent constraint. We impose it to demonstrate that the proposed method is genuinely high-order accurate (as this may not always be the case for other high-order methods proposed in the literature). This relaxation process is applied to all the simulations reported in the paper without exception to demonstrate the robustness of the method. The (affine) functionals corresponding to the above bounds are $\Psi_i^{min}(\mathsf{V}) := \mathsf{V} - \mathsf{U}_i^{min,n}$ and $\Psi_i^{max}(\mathsf{V}) := \mathsf{U}_i^{max,n} - \mathsf{V}$.

For all the tests reported below, the $L^1 -$ norm of the difference $u_h - u$ is measured by using high-order quadratures. We use the exact solution at the quadrature points instead of the Lagrange interpolant of the solution to avoid extraneous super-convergence effects that may bias the tests and lead us to draw over-optimistic conclusions. The error are all relative, that is we report $\|u_h - u\|_{L^1(D)}/\|u\|_{L^1(D)}$.

Remark 6.1. (*Stencil*) One can also use the high-order stencil to define $\mathsf{U}_i^{min,n} := min_{i\in\mathscr{I}(i)}\mathsf{U}_i^n$ and $\mathsf{U}_i^{max,n} := max_{i\in\mathscr{I}(i)}\mathsf{U}_i^n$. This increases the range of CFL numbers for which optimal convergence rate is observed in the L^∞-norm for smooth solutions, but this also slightly deteriorates the L^1-norm convergence rate on problems developing shocks. We have observed that using the high-order stencil in the definition of the second variation $\widetilde{\Delta^2}\mathsf{U}_i^n$ significantly deteriorates the L^1-norm convergence rate on problems developing shocks.

6.2. Swirling flow

We start by demonstrating that progresses have indeed been made on the low-order method since the observations made in [1, §3.3.2.1] and [2, §4.2.2]. We consider the same swirling model problem as in Section 4.1. We solve this problem with continuous $\mathbb{P}_2$

Table 1

Problem (6.8), $\mathbb{P}_1$, structured meshes (left), unstructured meshes (right), $T = 0.3$, CFL $= 0.1$.

Present $\mathbb{P}_2$			Standard $\mathbb{P}_2$		Present $\mathbb{P}_3$			Standard $\mathbb{P}_3$	
I	L^1	Rate	L^1	Rate	I	L^1	Rate	L^1	Rate
1681	6.13E-01	–	9.19E-01	–	2116	5.59E-01	–	9.74E-01	–
3721	4.96E-01	0.53	8.48E-01	0.20	3721	4.72E-01	0.60	9.66E-01	0.03
14641	3.11E-01	0.68	6.66E-01	0.35	14641	2.89E-01	0.72	8.97E-01	0.11
58081	1.77E-01	0.81	4.56E-01	0.55	32761	2.08E-01	0.82	8.15E-01	0.24
130321	1.24E-01	0.88	3.46E-01	0.69	130321	1.13E-01	0.89	6.22E-01	0.39
231361	9.56E-02	0.91	2.78E-01	0.76	292681	7.73E-02	0.93	4.98E-01	0.55
361201	7.77E-02	0.93	2.33E-01	0.80	519841	5.89E-02	0.95	4.14E-01	0.64

Table 2
Convergence tests: $\mathbb{P}_1$, $\mathbb{P}_2$, $\mathbb{P}_3$, structured meshes, Problem (6.8), $T = 0.3$, CFL $= 0.1$.

`Code 1`

I	$\mathbb{P}_1, L^1$	Rate	I	$\mathbb{P}_2, L^1$	Rate	I	$\mathbb{P}_3, L^1$	Rate
8281	5.18E−03	–	8281	3.66E−03	–	3721	1.89E−02	–
14641	2.91E−03	2.02	14641	2.13E−03	1.90	8281	4.56E−03	3.56
32761	1.28E−03	2.05	32761	9.32E−04	2.05	18496	9.92E−04	3.79
58081	7.16E−04	2.02	58081	5.22E−04	2.02	32761	3.12E−04	4.05
90601	4.59E−04	2.01	90601	3.34E−04	2.02	73441	7.38E−05	3.57
160801	2.58E−04	2.01	160801	1.90E−04	1.96	130321	2.00E−05	4.55
251001	1.65E−04	2.00	251001	1.24E−04	1.92	292681	2.60E−06	5.04

`Code 2`

I	$\mathbb{P}_1, L^1$	Rate	I	$\mathbb{P}_3, L^1$	Rate
8281	5.39E−03	–	8281	4.19E−03	–
18496	2.36E−03	2.06	18496	9.81E−04	3.61
41209	1.03E−03	2.08	40804	2.13E−04	3.86
92416	4.53E−04	2.02	90601	4.58E−05	3.85
207025	2.02E−04	2.01	203401	5.11E−06	5.43
465124	8.97E−05	2.01	456976	5.27E−07	5.61

Table 3
Convergence tests: $\mathbb{P}_1$, $\mathbb{P}_2$, $\mathbb{P}_3$, unstructured meshes, problem (6.8), $T = 0.3$, CFL $= 0.1$.

`Code 1`

I	$\mathbb{P}_1, L^1$	Rate	I	$\mathbb{P}_2, L^1$	Rate	I	$\mathbb{P}_3, L^1$	Rate
335	2.12E−01	–	349	1.62E−01	–	760	3.22E−01	–
1249	3.77E−02	2.63	1273	4.00E−02	2.16	2815	3.13E−02	3.56
4852	7.84E−03	2.32	4865	7.10E−03	2.58	10849	2.68E−03	3.64
11742	3.23E−03	2.01	11749	2.67E−03	2.22	42892	1.54E−04	4.16
19167	1.96E−03	2.02	19149	1.55E−03	2.23	104470	1.99E−05	4.59
46568	8.03E−04	2.02	46565	6.20E−04	2.05	170959	4.89E−06	5.69
185518	2.02E−04	1.99	185469	1.64E−04	1.92	416704	2.09E−06	1.91

`Code 2`

I	$\mathbb{P}_1, L^1$	Rate	I	$\mathbb{P}_3, L^1$	Rate
3414	1.55E−02	–	2638	1.01E−01	–
5754	9.96E−03	1.69	6313	2.00E−02	3.72
13295	4.46E−03	1.92	13084	3.47E−03	4.80
16474	3.37E−03	2.60	30106	8.95E−04	3.25
28860	2.45E−03	1.15	64330	1.34E−04	5.01
37412	1.82E−03	2.28	146890	4.00E−05	2.92
65923	1.09E−03	1.81	334648	4.72E−06	5.19

and $\mathbb{P}_3$ Lagrange finite elements with final time $t = 1$. For each polynomial degree we first solve the problem with the low-order method presented in the paper (see (4.9)), then we solve it again with the low-order method using the graph viscosity based on the high-order stencil (see (4.1)). This test is done on various meshes with increasing resolution. The relative L^1-norm of the error is computed at the end of each simulation. The results are reported in Table 1. We clearly observe that the present method has almost no pre-asymptotic range (green columns for the $\mathbb{P}_2$ approximation and the $\mathbb{P}_3$ approximation), whereas the standard method has a long pre-asymptotic range (orange columns). For instance, on the finest mesh composed of 519841 degrees of freedom, the error with the new low-order $\mathbb{P}_3$ approximation is 5.89×10^{-2} whereas the error with the standard method is 4.14×10^{-1}. There is almost one order of magnitude difference between the two approximations. This series of tests confirms the claims made in the paper regarding the low-order viscosity.

6.3. 2D smooth problem

We now consider a two-dimensional linear transport problem $\partial_t u + \nabla \cdot (\boldsymbol{\beta} u) = 0$ with $\boldsymbol{\beta} = (1, 0)^T$ and initial condition

Table 4
Convergence tests: Limited technique vs. Galerkin for $\mathbb{P}_2$ polynomials on structured and unstructured meshes, problem (6.8), `Code 1`, $T = 0.3$, CFL $=$ 0.1.

Structured meshes					Unstructured meshes				
I	Lim., L^1	Rate	Gal., L^1	Rate	I	Lim., L^1	Rate	Gal., L^1	Rate
8281	3.66E−03	–	5.84E−03	–	349	1.62E−01	–	2.74E−01	–
14641	2.13E−03	1.90	3.18E−03	2.13	1273	4.00E−02	2.16	6.99E−02	2.11
32761	9.32E−04	2.05	1.37E−03	2.09	4865	7.10E−03	2.58	1.43E−02	2.37
58081	5.22E−04	2.02	7.60E−04	2.05	11749	2.67E−03	2.22	5.84E−03	2.03
90601	3.34E−04	2.02	4.82E−04	2.05	19149	1.55E−03	2.23	3.31E−03	2.32
160801	1.90E−04	1.96	2.69E−04	2.03	46565	6.20E−04	2.05	1.28E−03	2.13
251001	1.24E−04	1.92	1.72E−04	2.02	185469	1.64E−04	1.92	3.14E−04	2.04

Table 5
Problem (6.9), $\mathbb{P}_2$ and $\mathbb{P}_3$, unstructured meshes, $T = 1$, CFL $= 0.3$. Left: results from `Code 1`. Right: results from `Code 2`.

`Code 1`			`Code 1`			`Code 2`		
I	$\mathbb{P}_2$, L^1	Rate	I	$\mathbb{P}_3$, L^1	Rate	I	$\mathbb{P}_3$, L^1	rate
3743	3.26E−01	–	3880	4.63E−01	–	6211	3.91E−01	–
14528	2.02E−01	0.71	15058	2.72E−01	0.78	10981	3.09E−01	0.83
56510	1.17E−01	0.80	57712	1.55E−01	0.84	24571	2.21E−01	0.83
99690	1.05E−01	0.37	99859	1.22E−01	0.86	55081	1.54E−01	0.89
224699	8.43E−02	0.55	223924	8.82E−02	0.81	121807	1.11E−01	0.82
394000	7.18E−02	0.58	394594	7.22E−02	0.70	270901	8.13E−02	0.79

$$u_0(\boldsymbol{x}) = \begin{cases} e^{\frac{r(\boldsymbol{x})^2+2r_0^2}{r(\boldsymbol{x})^2-r_0^2}} & \text{if} r(\boldsymbol{x}) < r_0, \\ 0 & \text{otherwise,} \end{cases} \tag{6.8}$$

where $r(\boldsymbol{x}) := \|\boldsymbol{x} - \boldsymbol{x}_0\|_{\ell^2}, \boldsymbol{x}_0 = (\frac{7}{20},\frac{1}{2})$, and $r_0 = 0.2$. We solve the problem in the unit square $D = \{\boldsymbol{x} := (x_1, x_2) \in \mathbb{R}^2 \,|\, 0 \leq x_1 \leq 1 \text{and} 0 \leq x_2 \leq 1\}$ up to $T = 0.3$. The simulations are done on several uniform and nonuniform grids with CFL=0.1.

We do two series of computations with $\mathbb{P}_1$, $\mathbb{P}_2$ and $\mathbb{P}_3$ continuous finite elements on meshes with various meshsizes. The first series in done on structured meshes and the other one is done on unstructured Delaunay meshes. The convergence results for structured meshes are shown in Table 2. The results for unstructured meshes are shown in Table 3. We observe second-order accuracy for $\mathbb{P}_1$ elements and fourth-order accuracy for $\mathbb{P}_3$ elements, both with structured and unstructured meshes. We observe sub-optimal second-order for $\mathbb{P}_2$ elements. This behavior is well-known and is independent of the limiting technique described in the paper. It is well established in the literature that (unless super-convergence occurs) one recovers optimal rate for $\mathbb{P}_2$ elements only by adding linear stabilization (e.g., Galerkin Least-Squares, edge stabilization, subgrid viscosity, etc..). To illustrate this point we compare in Table 4 the results obtained with the present method with those obtained with the unlimited Galerkin method with $\mathbb{P}_2$ elements. We observe that the Galerkin method is only second-order accurate. We also notice also that the limited $\mathbb{P}_2$ approximation is slightly more precise than the Galerkin method whether the meshes are structured or not.

Remark 6.2. (*Relaxation*) In tests not reported here for brevity, we have observed that the convergence rate for the $\mathbb{P}_3$ approximation deteriorates to second-order in the L^1-norm and, irrespective of the polynomial degree the approximation, reduces to first-order in the L^∞-norm if one does not relax the local bounds as described in (6.5)–(6.6)–(6.7). This loss of accuracy in the neighborhood of extrema due to limiting is a phenomenon that is well documented in the literature (see, e.g., [38, §3.3], [9, p. 2753]). A typical way to address this issue in the finite volume literature consists of relaxing the slope reconstructions; see, e.g., [39, Eq. (5.7)], [40]. Similar techniques can be used with discontinuous Galerkin approximations as in [9,41]. The relaxation technique (6.5)–(6.6)–(6.7) used in the paper has been introduced in [17, §4.7] (see (4.11) and (4.13) therein).

6.4. Three body rotation

In this example, we consider the linear transport equation with the flux $\boldsymbol{\beta} = 2\pi(-x_2, x_1)^T$ and the following initial condition

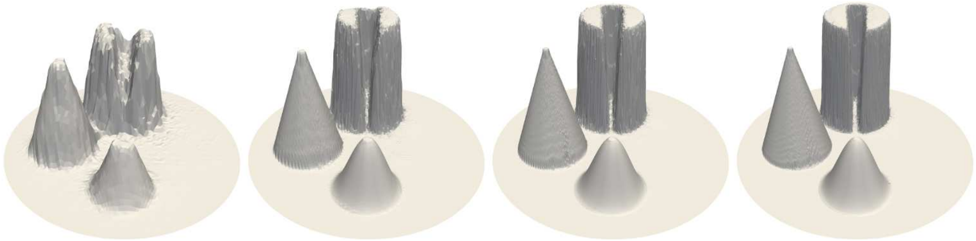
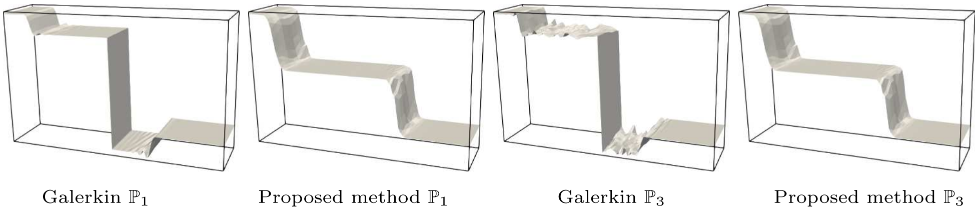


Fig. 5. Three body rotation. From the left to right: approximation with 6211, 55081, 121807 AND 258673 $\mathbb{P}_3$ nodes with CFL=0.3.



Galerkin $\mathbb{P}_1$ Proposed method $\mathbb{P}_1$ Galerkin $\mathbb{P}_3$ Proposed method $\mathbb{P}_3$

Fig. 6. Non-strictly convex flux (6.10); Solution at $t = 0.75$ with CFL=0.5; 847 grid points.

$$u_0(\boldsymbol{x}) = \begin{cases} 1 & \text{if} r_1(\boldsymbol{x}) \leq r_0 \text{and } (|x_1| \geq \frac{1}{20} \text{or} x_2 \geq \frac{7}{10}), \\ 1 - \frac{r_2(\boldsymbol{x})}{r_0} & \text{if} r_2(\boldsymbol{x}) \leq r_0, \\ \frac{1}{4}\left(1 + cos\left(\frac{r_3(\boldsymbol{x})}{r_0}\pi\right)\right) & \text{if} r_3(\boldsymbol{x}) \leq r_0, \\ 0 & \text{otherwise,} \end{cases} \tag{6.9}$$

where $r_0 := 0.3$, $r_1(\boldsymbol{x}) := \|\boldsymbol{x} - \boldsymbol{x}_1\|_{\ell^2}$, $r_2(\boldsymbol{x}) := \|\boldsymbol{x} - \boldsymbol{x}_2\|_{\ell^2}$, $r_3(\boldsymbol{x}) := \|\boldsymbol{x} - \boldsymbol{x}_3\|_{\ell^2}$, with $\boldsymbol{x}_1 := \left(0, \frac{1}{2}\right)$, $\boldsymbol{x}_2 := \left(-\frac{1}{2}, 0\right)$, $\boldsymbol{x}_3 := \left(0, -\frac{1}{2}\right)$.

The computational domain $D = \{\boldsymbol{x} \in \mathbb{R}^2 \big| \|\boldsymbol{x}\|_{\ell^2} \leq 1\}$ is triangulated using non-nested meshes. The simulations are done with $\mathbb{P}_2$ and $\mathbb{P}_3$ continuous finite elements on nonuniform meshes up to $T = 1$ with CFL=0.3. The convergence results are shown in Table 5. We observe that the method converges with a rate that is similar to what is reported in [29, Tab. 3] with $\mathbb{P}_1$ finite elements. This is normal as the solution is not smooth.

We show in Fig. 5 the graph of the approximate solution obtained on four different meshes using `Code 2`. These results are comparable to those in [29, Fig. 2].

6.5. Non-strictly convex flux

In this section we illustrate the performance of the method with a flux that is not strictly convex. We consider the two-dimensional scalar conservation equation $\partial_t u + \nabla \cdot \mathbb{f}(u) = 0$ using the flux $\mathbb{f}(v) = (2 - v, 0)^\mathsf{T}$ if $v \leq 2$ and $\mathbb{f}(v) = (2v - 4, 0)^\mathsf{T}$ otherwise. The initial data is $u_0(x) = 1$ if $x \leq 0$ and $u_0(x) = 3$ otherwise. This flux is convex and Lipschitz, but it is not strictly convex: the velocity is piecewise constant and discontinuous. The entropy solution is

$$u(x,t) = \begin{cases} 1 & \text{if} x \leq -t, \\ 2 & \text{if} -t < x \leq 2t, \\ 3 & \text{if} 2t < x. \end{cases} \tag{6.10}$$

The solution is composed of two contact waves (i.e., the characteristics do not cross) separated by an expansion wave. One contact wave moves to the left at speed -1, the other moves to the right at speed 2. This test is meant to eliminate high-order methods that are not dissipative enough. The lack of dissipation makes these methods to converge to weak solutions that violate entropy inequalities.

We solve the problem over the domain $D = (-2, 2) \times (0, 1)$ using uniform meshes. The solution is computed with the final time $t = 0.75$ using CFL=0.5. We show in Fig. 6 the graph of the solution obtained with $\mathbb{P}_1$ polynomials on a mesh composed of $120\times$ (2×6) elements and with $\mathbb{P}_3$ polynomials on a mesh composed of $40 \times (2 \times 2)$ elements. In both cases the number of grid points is 847. The solutions in the leftmost panel and in the next to last panel have been obtained with the Galerkin method using $\mathbb{P}_1$ and $\mathbb{P}_3$ finite elements, respectively. (We use exactly the same codes as the one used in the accuracy tests reported in Sections 6.3 and 6.4.) We observe that they both converge to a wrong weak solution with a stationary shock at $x = 0$. The reason for this behavior is that the viscosity coefficients d_{ij}^{H} are set to zero for the Galerkin approximation. The solutions in the second panel and in the last panel have

Table 6
Burgers; unstructured meshes; $\mathbb{P}_2$ and $\mathbb{P}_3$ finite elements; $T = 0.5$, CFL $= 0.3$. Left: results from `Code 1`. Right: results from `Code 2`.

`Code 1`			`Code 1`			`Code 2`		
I	$\mathbb{P}_2$, L^1	Rate	I	$\mathbb{P}_3$, L^1	Rate	I	$\mathbb{P}_3$, L^1	Rate
4561	5.41E−02	–	4315	4.67E−02	–	4702	5.59E−02	–
7549	4.10E−02	1.10	16864	2.58E−02	0.87	10315	4.17E−02	0.75
29849	2.27E−02	0.86	66919	1.35E−02	0.93	23338	2.79E−02	0.99
118785	1.33E−02	0.77	119560	1.09E−02	0.75	50962	1.84E−02	1.06
210317	9.37E−03	1.23	266785	6.92E−03	1.13	118477	1.24E−02	0.93
474137	6.77E−03	0.80	472573	4.96E−03	1.17	258673	8.82E−03	0.88
843717	4.98E−03	1.06	1065847	3.54E−03	0.83	469786	6.64E−03	0.95

been obtained with the proposed method using $\mathbb{P}_1$ and $\mathbb{P}_3$ finite elements, respectively. We observe that they both converge to the correct solution. Notice also that the graph of the solution in the expansion wave is free of oscillation (which would not be the case if we had only enforced global bounds as it sometimes done in the literature). We have verified that the method converges with the expected rate both with $\mathbb{P}_1$ and $\mathbb{P}_3$ finite elements (tables not shown here for brevity).

6.6. *Burgers*

We solve now a two-dimensional version of Burgers' equation introduced in [29, §6.1]. The initial data is chosen so that there are sonic points over a one-dimensional manifold across the graph of the solution. This test is meant to weed out methods that are accurate but not dissipative enough to be able to select the entropy solution in expansion waves with sonic points. Here we recall that we use the same codes as the one used in the accuracy tests reported in Section 6.3 and Section 6.4; nothing is changed in the codes. We consider the domain $D = (-0.25, 1.75)^2$ and the following two-dimensional version of Burgers' equation:

$$\partial_t u + \nabla\cdot(\boldsymbol{f}(u)) = 0, \qquad \boldsymbol{f}(u) = \frac{1}{2}\left(u^2, u^2\right)^{\top}, \qquad u(\boldsymbol{x},0) = u_0(\boldsymbol{x})\text{a.e.}\boldsymbol{x} \in D, \tag{6.11a}$$

$$\text{where} \quad u_0(\boldsymbol{x}) = \begin{cases} 1 & \text{if}\left|x_1 - \dfrac{1}{2}\right| \le 1\text{and}\left|x_2 - \dfrac{1}{2}\right| \le 1 \\ -a & \text{otherwise.} \end{cases} \tag{6.11b}$$

The exact solution is fully described in [29, §6.1]. The tests are done with $a = 0.75$. The errors are computed at $T = 0.5$. The results of the convergence tests for $\mathbb{P}_2$ and $\mathbb{P}_3$ continuous finite elements are reported in Table 6. We observe that the convergence rate in the L^1-norm is close to 1, which is optimal.

Remark 6.3. (*Convergence Rate*) The convergence rate for Burgers' equation is better than that observed for the linear transport equation with discontinuous data because the solution of Burgers' equation maps $L^\infty(\mathbb{R})$ to the space of functions with bounded variations over $\mathbb{R}$ (say BV($\mathbb{R}$)), see [42, Thm. 11.2.2]; hence, compactness is available. Notice that one can invoke [42, Thm. 11.2.2] here because the flux in (6.11a) is isotropic. We refer the reader to [43, Thm. 2] for further clarifications on this topic.

Remark 6.4. (*Relaxation*) Recall that relaxation of local bounds is only necessary to reach optimal accuracy for problems with smooth solutions in the L^∞-norm. We have observed that the convergence rate in the L^1-norm for Burgers' equation remains close to 1 if the bounds are not relaxed. Tests not reported here for brevity show that optimal convergence is lost on Burgers' equation if one uses the averaged version of the relaxation. A key conclusion of this paper is that using the minmod version of the relaxation produces optimal convergence rates in the L^1-norm for all the cases (smooth linear transport problem, non-smooth linear transport problem, Burgers' equation). In other words, the minmod relaxation method described in (6.5)–(6.6)–(6.7) is robust.

6.7. *Non-convex flux*

We finish by illustrating the performance of the method on a two-dimensional scalar conservation equation with a non-convex flux originally proposed in [44]:

$$\partial_t u + \nabla\cdot\boldsymbol{f}(u) = 0, \quad u(\boldsymbol{x},0) = u_0(\boldsymbol{x}) = \begin{cases} \dfrac{14\pi}{4} & \text{if}\sqrt{x^2+y^2} \le 1 \\ \dfrac{\pi}{4} & \text{otherwise,} \end{cases} \tag{6.12}$$

with $\boldsymbol{f}(u) := (sinu, cosu)^{\top}$. High-order methods that are too greedy usually fail to converge to the entropy solution as they have the tendency to produce shocks where one should have expansions; see e.g., [44]. The computational domain is $D = [-2{,}2] \times [-2.5{,}1.5]$. The final time is $t = 1$. The simulation are done with Lagrange $\mathbb{P}_1$ and Lagrange $\mathbb{P}_3$ finite elements and with $\mathbb{P}_2$ Bernstein finite

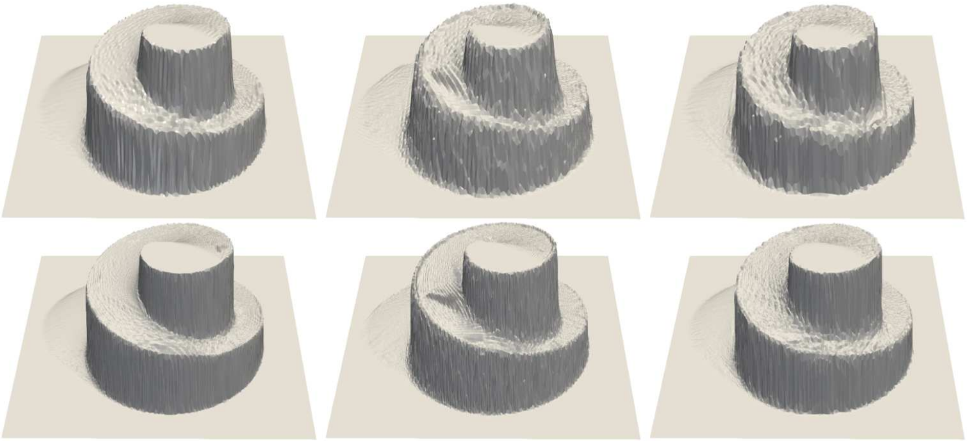


Fig. 7. Problem (6.12) at $t = 1$. From left to right: Lagrange $\mathbb{P}_1$, Bernstein $\mathbb{P}_2$, Lagrange $\mathbb{P}_3$. Top row: approximately 16870 grid points. Bottom row: approximately 67000 grid points.

elements. Two computations are done on unstructured meshes for each polynomial degree. For the $\mathbb{P}_1$ approximation the first mesh is composed of 33240 triangles and 16861 degrees of freedom and the second mesh is composed of 134066 triangles and 67516 degrees of freedom. For the $\mathbb{P}_2$ approximation the first mesh is composed of 8312 triangles and 16865 degrees of freedom and the second mesh is composed of 33240 triangles and 66961 degrees of freedom. For the $\mathbb{P}_3$ approximation the first mesh is composed of 3698 triangles and 16882 degrees of freedom and the second mesh is composed of 14782 triangles and 67000 degrees of freedom.

The results are shown in Fig. 7. The method behaves correctly for each polynomial degree; the spiraling composite wave has the expected shape. These results are identical to what is reported in the literature (see e.g., [44, Fig. 5.11] or [14, Fig. 34]).

7. Conclusions

We have proposed in this paper answers to questions raised in the PhD theses of [1, §3.3.2.1] and [2, §4.2.2], where it was observed that the low-order invariant-domain preserving method proposed in [3] was not robust with respect to the polynomial degree. Building on ideas developed in [4] (see Propositions 3.1 and 3.2 therein), we have proposed a variation of the invariant-domain preserving method from [3] that behaves better as the polynomial degree increases. In particular, the low-order method is based on the closest neighbor stencil. The method has been implemented and tested with continuous $\mathbb{P}_2$ and $\mathbb{P}_3$ finite elements. The numerical tests demonstrate that the method behaves as advertised in the theory.

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.

Data availability

No data was used for the research described in the article.

Appendix A. Sufficient criterion for Assumption 4.1

We give here a sufficient criterion for the condition (i) of Assumption 4.1 to hold.

Lemma A.1. *Let $\widehat{S}_l \in \mathscr{T}^{\rm s}_{\hat{K}}$. Let$\Pi^{\rm L}_{\hat{S}_l} : \widehat{\rm P}^{\rm H} \to \widehat{\rm P}^{\rm L}$ be the Lagrange interpolation operator defined by $\Pi^{\rm L}_{\hat{S}_l}(\widehat{p}) = \sum_{n^{\rm s}\in\hat{\mathscr{N}}^{\rm s}} \widehat{p}(\widehat{\boldsymbol{a}}_{\hat{j}^{\rm s}(n^{\rm s},l)})\widehat{\theta}^{\rm L}_{n^{\rm s}}$. Assume that there exists a collection of numbers $\{\beta^{\hat{S}}\}_{\hat{S}\in\mathscr{T}^{\rm s}_{\hat{K}}}$ such that the following identity holds true for every $\widehat{p} \in \widehat{\rm P}^{\rm H}$ and every $k \in \{1:d\}$:*

$$\int_{\hat{K}} \partial_{\hat{x}_k}\widehat{p}(\widehat{\boldsymbol{x}})\,{\rm d}\widehat{x} = \sum_{l\in\hat{\mathscr{L}}} \beta^{\hat{S}_l} \int_{\hat{S}_l} \partial_{\hat{x}_k}\Pi^{\rm L}_{\hat{S}_l}(\widehat{p})\,{\rm d}\widehat{x}. \tag{A.1}$$

Assume also that the mapping $\boldsymbol{T}_K : \widehat{K} \to K$ is affine for all $K \in \mathscr{T}_h$. Then the condition (i) in Assumption 4.1 is met with$\beta^S := \beta^{\hat{S}_l}$ where the index l is such that $S = \boldsymbol{T}_K(\widehat{S}_l)$.   **Proof.** Let $K \in \mathscr{T}_h$. Let $S_l \in \mathscr{T}^{\rm s}_K$ be a subs-cell of K with $l \in \widehat{\mathscr{L}}$. Recall that $S_l = \boldsymbol{T}_K(\widehat{S}_l)$. Let $\boldsymbol{T}_{K|\hat{S}_l} : \widehat{S}_l \to S_l$ be the restriction of $\boldsymbol{T}_K$ to $\widehat{S}_l$. Let $\mathbb{J}_K$ and $\mathbb{J}_{S_l}$ be the Jacobian matrices of $\boldsymbol{T}_K$ and $\boldsymbol{T}_{K|\hat{S}_l}$, respectively. Since the mapping $\boldsymbol{T}_K$ is affine, we have

Table 7

$\widehat{j}$, $\widehat{\beta}$ and $\widehat{\alpha}$ arrays for $\mathbb{P}_2$ and $\mathbb{P}_3$ polynomials in one dimension (uniform lattice). The arrays $\widehat{\hat{\alpha}}$ are only given for the Lagrange bases.

	$\mathbb{P}_2$		$\mathbb{P}_3$		
$l \in \widehat{\mathcal{L}}$	1	2	1	2	3
$\widehat{\mathrm{j}}(1,l)$	1	2	1	3	2
$\widehat{\mathrm{j}}(2,l)$	2	3	3	4	4
$\widehat{\beta}(l)$	1	1	1	1	1

	$\mathbb{P}_2$		$\mathbb{P}_3$		
$l \in \widehat{\mathcal{L}}$	1	2	1	2	3
$\widehat{\alpha}(1,l)$	1	$\frac{1}{2}$	1	$\frac{4}{9}$	1
$\widehat{\alpha}(2,l)$	$\frac{1}{2}$	1	$\frac{5}{9}$	$\frac{4}{9}$	$\frac{5}{9}$

Table 8

$\widehat{j}$ for $\mathbb{P}_2$ and $\mathbb{P}_3$ polynomials in two dimensions.

	$\mathbb{P}_2$				$\mathbb{P}_3$								
$l \in \widehat{\mathscr{L}}$	1	2	3	4	1	2	3	4	5	6	7	8	9
$\widehat{\mathrm{j}}(1,l)$	1	4	2	3	1	6	8	4	2	6	5	4	3
$\widehat{\mathrm{j}}(2,l)$	5	5	4	4	6	8	9	9	4	7	7	5	5
$\widehat{\mathrm{j}}(3,l)$	6	6	6	5	8	10	10	10	9	10	10	10	7
$\widehat{\beta}(l)$	$\frac{2}{3}$	$\frac{2}{3}$	$\frac{2}{3}$	2	$\frac{3}{4}$	$\frac{3}{2}$	$\frac{3}{4}$	$\frac{3}{2}$	$\frac{3}{4}$	$\frac{3}{4}$	$\frac{3}{2}$	$\frac{3}{4}$	$\frac{3}{4}$

$\mathbb{J}_{K|S_l} = \mathbb{J}_{S_l}$ and the determinants $det(\mathbb{J}_{K|S_l}) = det(\mathbb{J}_K)$ are constant. Let $\boldsymbol{p} \in \boldsymbol{P}_K^{\mathrm{H}}$ with $\boldsymbol{p} = \sum_{i\in\mathscr{I}(K)}\boldsymbol{p}(\boldsymbol{a}_i)\varphi_i^{\mathrm{H}}$. By definition of $\widehat{\boldsymbol{P}}^{\mathrm{H}}$, there is $\widehat{\boldsymbol{p}} \in \widehat{\boldsymbol{P}}$ such that $\widehat{\boldsymbol{p}} = \boldsymbol{p} \circ \boldsymbol{T}_K$. Recall that the chain rule gives $(\nabla\cdot\boldsymbol{p})(\boldsymbol{T}_K(\widehat{\boldsymbol{x}}))det(\mathbb{J}_K) = \nabla\cdot(det(\mathbb{J}_K)\mathbb{J}_K^{-1}\widehat{\boldsymbol{p}})(\widehat{\boldsymbol{x}})$. Then owing to (A.1) and $\nabla\cdot(\mathbb{J}_K^{-1}\widehat{\boldsymbol{p}})(\widehat{\boldsymbol{x}}) = \mathbb{J}_K^{-1} : (\nabla\widehat{\boldsymbol{p}})(\widehat{\boldsymbol{x}})$ (since the mapping $\boldsymbol{T}_K$ is affine), we infer that

$$\int_K \nabla\cdot\boldsymbol{p}(\boldsymbol{x})\,\mathrm{d}x = \int_{\hat{K}} (\nabla\cdot\boldsymbol{p})(\boldsymbol{T}_K(\widehat{\boldsymbol{x}}))det(\mathbb{J}_K)\,\mathrm{d}\widehat{x} = \int_{\hat{K}} \nabla\cdot\left(det(\mathbb{J}_K)\mathbb{J}_K^{-1}\widehat{\boldsymbol{p}}\right)(\widehat{\boldsymbol{x}})\,\mathrm{d}\widehat{x} \quad = det(\mathbb{J}_K)\mathbb{J}_K^{-1} : \int_{\hat{K}} (\nabla\widehat{\boldsymbol{p}})(\widehat{\boldsymbol{x}})\,\mathrm{d}\widehat{x} = det(\mathbb{J}_K)\mathbb{J}_K^{-1}$$

$$: \sum_{l\in\hat{\mathscr{L}}} \beta^{\hat{S}_l} \int_{\hat{S}_l} \left(\nabla\Pi_{\hat{S}_l}^{\mathrm{L}}\widehat{\boldsymbol{p}}\right)(\widehat{\boldsymbol{x}})\,\mathrm{d}\widehat{x} \quad = \sum_{l\in\hat{\mathscr{L}}} \beta^{\hat{S}_l} \int_{\hat{S}_l} \nabla\cdot\left(det\left(\mathbb{J}_{S_l}\right)\mathbb{J}_{S_l}^{-1}\Pi_{\hat{S}_l}^{\mathrm{L}}\widehat{\boldsymbol{p}}\right)(\widehat{\boldsymbol{x}})\,\mathrm{d}\widehat{x} = \sum_{l\in\hat{\mathscr{L}}} \beta^{\hat{S}_l} \int_{S_l} \nabla\cdot\left(\Pi_{\hat{S}_l}^{\mathrm{L}}\widehat{\boldsymbol{p}} \circ \boldsymbol{T}_K^{-1}\right)(\boldsymbol{x})\,\mathrm{d}x.$$

Recalling that $\boldsymbol{p} := \sum_{i\in\mathscr{I}(K)}\boldsymbol{p}(\boldsymbol{a}_i)\varphi_i^{\mathrm{H}}$ and the definitions (2.9) and (2.11), we have

$$\left(\Pi_{\hat{S}_l}^{\mathrm{L}}\widehat{\boldsymbol{p}} \circ \boldsymbol{T}_K^{-1}\right)_{|S_l} = \sum_{n^{\mathrm{s}}\in\hat{\mathscr{N}}^{\mathrm{s}}} \widehat{\boldsymbol{p}}\left(\widehat{\boldsymbol{a}}_{\hat{j}^{\mathrm{s}}(n^{\mathrm{s}},l)}\right)\widehat{\theta}_{n^{\mathrm{s}}}^{\mathrm{L}} \circ \boldsymbol{T}_{K|S_l}^{-1} = \sum_{n^{\mathrm{s}}\in\hat{\mathscr{N}}^{\mathrm{s}}} \boldsymbol{p}\left(\boldsymbol{a}_{\mathrm{j}^{\mathrm{L}}(n^{\mathrm{s}},S_l)}\right)\varphi_{\mathrm{j}^{\mathrm{L}}(n^{\mathrm{s}},S_l)}{}^{\mathrm{L}}$$
$$=: \Pi_{S_l}^{\mathrm{L}}(\boldsymbol{p}).$$

The assertion readily follows. □ The immediate consequence of Lemma A.1 is that, provided the mesh is affine, one just need to verify that (i) holds on the reference element to ascertain that the condition (i) in Assumption 4.1 is met. This type of computation is done for finite elements of arbitrary polynomial degree in two and three dimensions in [4].

Lemma A.2. *Assume that there exist two mappings $\widehat{\alpha} : \widehat{\mathscr{N}} \times \widehat{\mathscr{L}} \to \mathbb{R}$ and $\widehat{\beta} : \widehat{\mathscr{L}} \to \mathbb{R}$ such that the following holds for all $n \in \widehat{\mathscr{N}}$ and all $l \in \widehat{\mathscr{L}}$:*

$$\sum_{\{(n^{\mathrm{s}},l)|n=\hat{j}(n^{\mathrm{s}},l)\}} \widehat{\alpha}(n,l) = 1, \qquad \sum_{n^{\mathrm{s}}\in\hat{\mathscr{N}}^{\mathrm{s}}} \widehat{\alpha}(\widehat{j}(n^{\mathrm{s}},l),l)\widehat{m}_{\hat{j}(n^{\mathrm{s}},l)}^{\hat{K}} = \widehat{\beta}(l)|\widehat{S}_l|. \tag{A.2}$$

Assume that the mapping $\boldsymbol{T}_K$ is affine for all $K \in \mathscr{T}_h$. For all $K \in \mathscr{T}_h$ and all $S_l \in \mathscr{T}_K^{\mathrm{s}}$ with$\boldsymbol{T}_K(\widehat{S}_l) = S_l$, we set$\beta^{S_l} = \widehat{\beta}(l)$. Similarly we set $\alpha_{\hat{\mathrm{j}}(n^{\mathrm{s}},l)}^{S_l} := \widehat{\alpha}(n,l)$. Then the conditions (ii) in Assumption 4.1 are met. **Proof.** This is a simple consequence of the identity $\frac{|\widehat{K}|}{|K|} = \frac{|\widehat{S_l}|}{|S_l|}$ since $\boldsymbol{T}_K$ is affine. □

Examples for the coefficients $\widehat{\alpha}$ and $\widehat{\beta}$ are given in the sections below.

Appendix B. $\widehat{\mathrm{j}}$, $\widehat{\alpha}$ And $\widehat{\beta}$ for $\mathbb{P}_2$ and $\mathbb{P}_3$ in one dimension

The arrays $\widehat{\mathrm{j}}$, $\widehat{\alpha}$ and $\widehat{\beta}$ for $\mathbb{P}_2$ and $\mathbb{P}_3$ polynomials in one dimension are shown in Table 7. The arrays $\widehat{\alpha}$ are only given for the Lagrange bases. The nodes are enumerated using the increasing vertex index enumeration convention (see e.g., Definition 10.6 and §21.3.2 in [45]). Recall that the enumeration of the sub-cells is done with the index set $\mathscr{L}$.

Table 9

$\widehat{\alpha}$ for the $\mathbb{P}_2$ and $\mathbb{P}_3$ Lagrange bases in two dimensions.

	$\mathbb{P}_2$				$\mathbb{P}_3$								
$l \in \widehat{\mathscr{L}}$	1	2	3	4	1	2	3	4	5	6	7	8	9
$\widehat{\alpha}(1,l)$	1	$\frac{1}{2}$	1	1	1	y	x	y	1	x	y	x	1
$\widehat{\alpha}(2,l)$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{3}$	y	x	y	$\frac{1}{3}$	x	y	x	$\frac{1}{3}$
$\widehat{\alpha}(3,l)$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{3}$	z	$\frac{1}{3}-z$	z	$\frac{1}{3}$	$\frac{1}{3}-z$	z	$\frac{1}{3}-z$	$\frac{1}{3}$

Table 10

$\widehat{\alpha}$ for the $\mathbb{P}_2$ and $\mathbb{P}_3$ Bernstein bases in two dimensions.

	$\mathbb{P}_2$				$\mathbb{P}_3$								
$l \in \widehat{\mathscr{L}}$	1	2	3	4	1	2	3	4	5	6	7	8	9
$\widehat{\alpha}(1,l)$	1	1	1	1	1	y	x	y	1	x	y	x	1
$\widehat{\alpha}(2,l)$	0	1	0	0	$-\frac{1}{12}$	y	x	y	$-\frac{1}{12}$	x	y	x	$-\frac{1}{12}$
$\widehat{\alpha}(3,l)$	0	1	0	0	$-\frac{1}{12}$	z	$\frac{1}{3}-z$	z	$-\frac{1}{12}$	$\frac{1}{3}-z$	z	$\frac{1}{3}-z$	$-\frac{1}{12}$

Appendix C. $\widehat{\mathrm{j}}$, $\widehat{\alpha}$ And $\widehat{\beta}$ for $\mathbb{P}_2$ and $\mathbb{P}_3$ in two dimensions

The arrays $\widehat{\mathrm{j}}$ and $\widehat{\beta}$ for $\mathbb{P}_2$ and $\mathbb{P}_3$ polynomials in two dimensions are shown in Table 8 . The nodes are enumerated using the increasing vertex index enumeration convention.

The arrays $\widehat{\alpha}$ for the $\mathbb{P}_2$ and $\mathbb{P}_3$ Lagrange bases in two dimensions are shown in Table 9. For $\mathbb{P}_3$ polynomials the array $\widehat{\alpha}$ is defined up to a free parameter y, and we use the definitions $x := \frac{2}{3} - y$ and $z := \frac{1}{3}\left(\frac{10}{9} - y\right)$ to save space in the table. In the computation reported in the paper we take $y = \frac{7}{36}$.

The arrays $\widehat{\alpha}$ for the $\mathbb{P}_2$ and $\mathbb{P}_3$ Bernstein bases in two dimensions are shown in Table 10. For $\mathbb{P}_3$ polynomials the array $\widehat{\alpha}$ is defined up to a free parameter y and we use the definitions $x := \frac{13}{12} - y$ and $z := \frac{5}{3} - 2y$ to save space in the table. In the computation reported in the paper we take $y = \frac{2}{3}$.

References

[1] F.S.A. Alrashed, Parallel Multiphase Navier-Stokes Solver (Ph.D. thesis –Texas A & M University), ProQuest LLC, Ann Arbor, MI, 2015. URL https://oaktrust.library.tamu.edu/bitstream/handle/1969.1/155339/ALRASHED-DISSERTATION-2015.pdf.

[2] M. Quezada De Luna, High-Order Maximum Principle Preserving (MPP) Techniques for Solving Conservation Laws with Applications on Multiphase Flow (Ph.D. thesis –Texas A & M University), ProQuest LLC, Ann Arbor, MI, 2016. URL https://oaktrust.library.tamu.edu/bitstream/handle/1969.1/158007/QUEZADADELUNA-DISSERTATION-2016.pdf.

[3] J.-L. Guermond, B. Popov, Invariant domains and first-order continuous finite element approximation for hyperbolic systems, SIAM J. Numer. Anal. 54 (4) (2016) 2466–2489.

[4] R. Abgrall, Q. Viville, H. Beaugendre, C. Dobrzynski, Construction of a p-adaptive continuous residual distribution scheme, J. Sci. Comput. 72 (3) (2017) 1232–1268.

[5] R. Sanders, A third-order accurate variation nonexpansive difference scheme for single nonlinear conservation laws, Math. Comp. 51 (184) (1988) 535–558.

[6] X.-D. Liu, E. Tadmor, Third order nonoscillatory central scheme for hyperbolic conservation laws, Numer. Math. 79 (3) (1998) 397–425.

[7] A. Kurganov, G. Petrova, A third-order semi-discrete genuinely multidimensional central scheme for hyperbolic conservation laws and related problems, Numer. Math. 88 (4) (2001) 683–729.

[8] W. Pazner, Sparse invariant domain preserving discontinuous Galerkin methods with subcell convex limiting, Comput. Methods Appl. Mech. Engrg. 382 (28) (2021), 113876.

[9] X. Zhang, C.-W. Shu, Positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations with source terms, J. Comput. Phys. 230 (4) (2011) 1238–1248.

[10] X. Zhang, Y. Xia, C.-W. Shu, Maximum-principle-satisfying and positivity-preserving high order discontinuous Galerkin schemes for conservation laws on triangular meshes, J. Sci. Comput. 50 (1) (2012) 29–62.

[11] R. Abgrall, Residual distribution schemes: current status and future trends, Comput. Fluids 35 (7) (2006) 641–669.

[12] M. Ricchiuto, R. Abgrall, Explicit Runge-Kutta residual distribution schemes for time dependent problems: second order case, J. Comput. Phys. 229 (16) (2010) 5653–5691.

[13] R. Abgrall, P. Bacigaluppi, S. Tokareva, High-order residual distribution scheme for the time-dependent Euler equations of fluid dynamics, Comput. Math. Appl. 78 (2) (2019) 274–297.

[14] F. Vilar, R. Abgrall, A posteriori local subcell correction of high-order discontinuous galerkin scheme for conservation laws on two-dimensional unstructured grids, 2022.

[15] J.-L. Guermond, M. Nazarov, A maximum-principle preserving C^0 finite element method for scalar conservation equations, Comput. Methods Appl. Mech. Engrg. 272 (2014) 198–213.

[16] J.-L. Guermond, M. Nazarov, B. Popov, Y. Yang, A second-order maximum principle preserving Lagrange finite element technique for nonlinear scalar conservation equations, SIAM J. Numer. Anal. 52 (4) (2014) 2163–2182.

[17] J.-L. Guermond, M. Nazarov, B. Popov, I. Tomas, Second-order invariant domain preserving approximation of the Euler equations using convex limiting, SIAM J. Sci. Comput. 40 (5) (2018) A3211–A3239.

[18] J.-L. Guermond, B. Popov, I. Tomas, Invariant domain preserving discretization-independent schemes and convex limiting for hyperbolic systems, Comput. Methods Appl. Mech. Engrg. 347 (2019) 143–175.

[19] J.P. Boris, D.L. Book, Flux-corrected transport. I. SHASTA, a fluid transport algorithm that works, J. Comput. Phys. 135 (2) (1997) 170–186, J. Comput. Phys. 11 (1) (1973) 38–69.

[20] S.T. Zalesak, Fully multidimensional flux-corrected transport algorithms for fluids, J. Comput. Phys. 31 (3) (1979) 335–362.

[21] D. Kuzmin, S. Turek, Flux correction tools for finite elements, J. Comput. Phys. 175 (2) (2002) 525–558.

[22] D. Kuzmin, R. Löhner, S. Turek, Flux-corrected transport: Principles, algorithms, and applications, in: Scientific Computation, Springer, 2012.

[23] R. Anderson, V. Dobrev, T. Kolev, D. Kuzmin, M. Quezada de Luna, R. Rieben, V. Tomov, High-order local maximum principle preserving (MPP) discontinuous Galerkin finite element method for the transport equation, J. Comput. Phys. 334 (2017) 102–124.

[24] A. Ern, J.-L. Guermond, Invariant-domain-preserving high-order time stepping: I Explicit Runge-Kutta schemes, SIAM J. Sci. Comput. 44 (5) (2022) A3366–A3392.

[25] D. Kuzmin, M. Quezada de Luna, Subcell flux limiting for high-order Bernstein finite element discretizations of scalar hyperbolic conservation laws, J. Comput. Phys. 411 (19) (2020), 109411.

[26] A. Jameson, W. Schmidt, E. Turkel, Numerical solution of the Euler equations by finite volume methods using Runge-Kutta time-stepping schemes, in: 14th AIAA Fluid and Plasma Dynamics Conference, 1981, AIAA Paper 1981–1259.

[27] A. Jameson, Origins and further development of the Jameson-Schmidt-Turkel scheme, AIAA J. 55 (5) (2017).

[28] E. Burman, On nonlinear artificial viscosity, discrete maximum principle and hyperbolic conservation laws, BIT 47 (4) (2007) 715–733.

[29] J.-L. Guermond, B. Popov, Invariant domains and second-order continuous finite element approximation for scalar conservation equations, SIAM J. Numer. Anal. 55 (6) (2017) 3120–3146.

[30] P.D. Lax, Weak solutions of nonlinear hyperbolic equations and their numerical computation, Comm. Pure Appl. Math. 7 (1954) 159–193.

[31] A. Harten, P.D. Lax, B. van Leer, On upstream differencing and Godunov-type schemes for hyperbolic conservation laws, SIAM Rev. 25 (1) (1983) 35–61.

[32] E. Tadmor, Numerical viscosity and the entropy condition for conservative difference schemes, Math. Comp. 43 (168) (1984) 369–381.

[33] B. Perthame, C.-W. Shu, On positivity preserving finite volume schemes for Euler equations, Numer. Math. 73 (1) (1996) 119–130.

[34] R. Abgrall, Some remarks about conservation for residual distribution schemes, Comput. Methods Appl. Math. ISSN: 1609-4840 18 (3) (2018) 327–351.

[35] A. Logg, K.-A. Mardal, G.N. Wells, et al., Automated Solution of Differential Equations By the Finite Element Method, Springer, 2012.

[36] J.F.B.M. Kraaijevanger, Contractivity of Runge-Kutta methods, BIT 31 (3) (1991) 482–528.

[37] P.-O. Persson, J. Peraire, Sub-cell shock capturing for discontinuous galerkin methods, in: 44th AIAA Aerospace Sciences Meeting and Exhibit, Number AIAA Paper No. 2015-2006-112 in Aerospace Sciences Meetings, 2006.

[38] B. Khobalatte, B. Perthame, Maximum principle on the entropy and second-order kinetic schemes, Math. Comp. 62 (205) (1994) 119–131.

[39] A. Harten, On the symmetric form of systems of conservation laws with entropy, J. Comput. Phys. 49 (1) (1983) 151–164.

[40] A. Harten, S. Osher, Uniformly high-order accurate nonoscillatory schemes. I, SIAM J. Numer. Anal. 24 (2) (1987) 279–309.

[41] X. Zhang, C.-W. Shu, On maximum-principle-satisfying high order schemes for scalar conservation laws, J. Comput. Phys. 229 (9) (2010) 3091–3120.

[42] C.M. Dafermos, in: Hyperbolic Conservation Laws in Continuum Physics, third ed.Grundlehren der mathematischen Wissenschaften[Fundamental Principles of Mathematical Sciences], vol. 325, Springer-Verlag, Berlin, 2010.

[43] D. Hoff, The sharp form of Oleĭnik's entropy condition in several space variables, Trans. Amer. Math. Soc. 276 (2) (1983) 707–714.

[44] A. Kurganov, G. Petrova, B. Popov, Adaptive semidiscrete central-upwind schemes for nonconvex hyperbolic conservation laws, SIAM J. Sci. Comput. 29 (6) (2007) 2381–2401.

[45] A. Ern, J.-L. Guermond, in: Finite Elements I—Approximation and Interpolation, Texts in Applied Mathematics, vol. 72, Springer, Cham, 2021.