



Monolithic convex limiting in discontinuous Galerkin discretizations of hyperbolic conservation laws

Item Type	Preprint
Authors	Hajduk, Hennes
Eprint version	Pre-print
Publisher	arXiv
Rights	Archived with thanks to arXiv
Download date	2024-04-18 12:49:29
Link to Item	http://hdl.handle.net/10754/664384

Monolithic convex limiting in discontinuous Galerkin discretizations of hyperbolic conservation laws

Hennes Hajduk

*Institute of Applied Mathematics (LS III), TU Dortmund University,
Vogelpothsweg 87, D-44227 Dortmund, Germany*

Abstract

In this work we present a framework for enforcing discrete maximum principles in discontinuous Galerkin (DG) discretizations. The developed schemes are applicable to scalar conservation laws as well as hyperbolic systems. Our methodology for limiting volume terms is similar to recently proposed methods for continuous Galerkin approximations, while DG flux terms require novel stabilization techniques. Piecewise Bernstein polynomials are employed as shape functions for the DG spaces, thus facilitating the use of very high order spatial approximations. We discuss the design of a new, provably invariant domain preserving DG scheme that is then extended by state-of-the-art subcell flux limiters to obtain a high-order bound preserving approximation. The limiting procedures can be formulated in the semi-discrete setting. Thus convergence to steady state solutions is not inhibited by the algorithm. We present numerical results for a variety of benchmark problems. Conservation laws considered in this study are linear and nonlinear scalar problems, as well as the Euler equations of gas dynamics and the shallow water system.

Keywords: discontinuous Galerkin methods, discrete maximum principles, limiters, Bernstein polynomials, Euler equations of gas dynamics, shallow water equations

1. Introduction

For many partial differential equations (PDE) the analytical solution is known to be member of a convex set, called *invariant domain*, of physically admissible states [10, 18]. Stabilized finite element methods for hyperbolic conservation laws often rely on limiters or other shock-capturing techniques to prevent the numerical approximation from dropping out of the invariant domain. Computational methods which prohibit the occurrence of such nonphysical solutions are therefore called *invariant domain preserving* (IDP) [18, 20, 23]. Many representatives of such schemes (e.g. [7, 8, 18, 34, 43]) use algebraic flux correction (AFC) techniques to combine a (provably IDP) *low order method* with a corresponding high-order *target scheme*.

Email address: hennes.hajduk@math.tu-dortmund.de (Hennes Hajduk)

Low order methods for AFC are obtained by performing mass lumping and adding diffusive matrices to the semi-discrete formulation. The minimal amount of such stabilization for scalar problems is provided by *discrete upwinding* [32, 40]. For a general hyperbolic problem, the appropriate amount of *graph viscosity* is proportional to the maximal wave speed. Guermond and Popov [18] analyzed the so-defined *Rusanov-type dissipation* [2, 21] and proved the IDP property of the resulting low-order method using its representation in terms of the so-called *bar states*. Lohmann et al. [43] improved discrete upwinding by preconditioning the convective matrices in a way that makes this low order scheme better applicable to high order finite element spaces. Based on this upwinding strategy, various high order extensions were considered in [27, 39, 43]. The concept of residual distribution [2, 3] makes it possible to design matrix-free low order methods for scalar equations [27, 28]. These approaches provide an efficient alternative to matrix-based upwinding but their generalization to systems is not straightforward.

Most of the limiters mentioned so far utilize the *Flux-Corrected-Transport* (FCT) methodology [11, 41, 48]. This class of schemes is based on the fully discrete formulation and predictor-corrector algorithms which do not converge to steady-state solutions. Monolithic limiting techniques were recently designed in [28, 34, 42] to overcome this shortcoming of FCT and construct nonlinear problems with well-defined residuals. Building on the analysis of Guermond and Popov [18], Kuzmin [36] proposed a new monolithic limiter for enforcing the IDP property for general hyperbolic problems discretized using (multi-)linear continuous finite elements. A generalization to higher order space discretizations can be found in [39]. In this work, we extend these concepts to discontinuous Galerkin (DG) approximations of scalar equations and hyperbolic systems. Classical DG discretizations make use of total variation bounded (TVB) slope limiters or weighted essentially non-oscillatory (WENO) schemes (e.g. [9, 30, 33, 49]) rather than algebraic limiting techniques. However, the AFC methodology can be advantageous in the DG setting as well [5, 7, 27, 44] since it guarantees the IDP property, in contrast to geometric limiters. A high order IDP-DG scheme for nonlinear hyperbolic conservation laws was recently developed by Pazner [44] who constrained a collocated spectral element approximation using FCT. In their recent work [24], Guermond et al. mentioned the possibility of unified algebraic limiting for fluxes and volume integrals of DG schemes. In the present paper, we handle the DG fluxes separately, which enables us to sparsify the element matrices of the high order discrete gradient operator via the application of the local mass lumping preconditioner proposed by Lohmann et al. [43]. All currently available AFC-DG methods are designed for linear scalar problems and/or equipped with FCT limiters which inhibit convergence to steady state solutions. The methodology that we propose yields the first monolithic limiting framework for high order DG discretizations of hyperbolic systems.

The approach presented in this work employs Bernstein polynomials as basis functions, as has been done previously, e.g., in [3, 4, 5, 29, 43]). It is suitable for tensor-product elements (lines in 1D, quadrilaterals in 2D, hexahedra in 3D) – which we conflate using the term *box elements* – as well as simplices and prisms. Limiting for volume terms works similarly to the corresponding methods in [18, 36, 39], while DG flux terms are handled by a generalization of our approach for the advection equation [27]. The concept of bar states, which was originally

introduced for volume terms of continuous Galerkin discretizations [18, 23, 36], translates naturally to the DG context leading to simple proofs of the IDP property. The monolithic limiting strategy and customized treatment of DG flux terms distinguish our approach from the *discretization-independent* AFC tools suggested in [24]. The presented methods are tested on benchmark configurations for the advection and Burgers problems, as well as for the systems of Euler and shallow water equations.

The structure of this article is as follows: Section 2 describes the standard DG discretization, followed by the derivation of the low order method in Section 3. Limiting aspects are discussed in Section 4. Numerical examples are presented in Section 5 and conclusions are drawn in Section 6. Further properties of the methods are detailed in the Appendix.

2. Standard DG discretization

Let $U(\mathbf{x}, t) \in \mathbb{R}^m$, $m \in \mathbb{N}$ be the vector of conserved unknowns at location $\mathbf{x} \in \mathbb{R}^d$, $d \in \{1, 2, 3\}$ and time $t \geq 0$. We consider the initial value problem

$$\frac{\partial U}{\partial t} + \nabla \cdot \mathbf{F}(U) = 0 \quad \text{in } \Omega \times \mathbb{R}_+, \quad (1a)$$

$$U(\cdot, 0) = U_0 \quad \text{in } \Omega, \quad (1b)$$

where $\Omega \subseteq \mathbb{R}^d$ is a Lipschitz domain, $\mathbf{F}(U) \in \mathbb{R}^{m \times d}$ is the hyperbolic flux matrix, and $U_0(\mathbf{x}) \in \mathbb{R}^m$ denotes some initial datum. A convex set $\mathcal{G} \subseteq \mathbb{R}^m$ such that $U_0(\mathbf{x}) \in \mathcal{G}$ for all $\mathbf{x} \in \Omega$, is called an invariant set of (1) if $U(\mathbf{x}, t) \in \mathcal{G}$ for all $\mathbf{x} \in \Omega$ and $t \geq 0$ [18]. To obtain a well-posed problem, (1) has to be equipped with suitable boundary conditions on $\Gamma = \partial\Omega$. In the scalar case ($m = 1$), boundary values U_{in} are imposed weakly on the inflow boundary $\Gamma_- = \{\mathbf{x} \in \Gamma : \mathbf{F}'(U) \cdot \mathbf{n}(\mathbf{x}) < 0\}$, where \mathbf{n} is the unit outward normal to Γ . The definition of the invariant set $\mathcal{G} = [U^{\min}, U^{\max}]$ must be modified to include U_{in} [18, 36]. For systems of conservation laws ($m > 1$) one has to distinguish between various boundary types, some of which are mentioned in Section 5. Alternatively, one can use periodic domains to avoid the issue of handling boundary conditions altogether.

To discretize (1a), we decompose Ω into elements K^e of a triangulation $\mathcal{T}_h = \{K^1, \dots, K^E\}$, such that $\bar{\Omega} = \cup_{e=1}^E K^e$. Every $K^e \in \mathcal{T}_h$ is assumed to be the image of a certain reference element $\hat{K} \subseteq \mathbb{R}^d$ under a C^1 -mapping \mathbf{F}^e , i.e., $K^e = \mathbf{F}^e(\hat{K})$, and $\mathbf{J}^e(\mathbf{x}) := \nabla \mathbf{F}^e(\mathbf{x}) \in \mathbb{R}^{d \times d}$ denotes the Jacobian of \mathbf{F}^e . For $p \in \mathbb{N}_0$, we define standard polynomial spaces $\mathbb{V}_p(K^e)$ based on the geometry of elements

$$\mathbb{V}_p(K^e) = \begin{cases} \mathbb{P}_p(K^e) & \text{if } K^e \text{ is a simplex,} \\ \mathbb{Q}_p(K^e) & \text{otherwise.} \end{cases}$$

For a given mesh and polynomial degree, the DG finite element spaces read

$$\mathbb{V}_p(\mathcal{T}_h) = \{\varphi_h \in L^2(\Omega) : \varphi_h|_{K^e} \in \mathbb{V}_p(K^e) \forall K^e \in \mathcal{T}_h\}. \quad (2)$$

A set of basis functions for (2) can have support which is local to only one respective element. Hence, for each element, we can choose basis functions $\{\varphi_i^e\}_{i=1}^N$ of $\mathbb{V}_p(K^e)$, where

$N = N(e) = \dim \mathbb{V}_p(K^e)$ is the local number of unknowns. On K^e , we can thus write the solution as

$$U_h^e(\mathbf{x}, t) := U_h(\mathbf{x}, t) |_{K^e} = \sum_{i=1}^N u_i^e(t) \varphi_i^e(\mathbf{x}), \quad u_i^e(t) \in \mathbb{R}^m. \quad (3)$$

Testing (1a) with $V_h \in \mathbb{V}_p(K^e)^d$ and using integration by parts yields

$$\int_{K^e} V_h \cdot \frac{\partial U_h^e}{\partial t} d\mathbf{x} - \int_{K^e} \nabla V_h : \mathbf{F}(U_h^e) d\mathbf{x} + \int_{\partial K^e} V_h \cdot \mathbf{F}(U_h^e) \mathbf{n}^e ds = 0, \quad (4)$$

where, \cdot and $:$ denote the standard vector and matrix scalar products and \mathbf{n}^e is the outward unit normal to K^e . The boundary integral can be split into integrals over separate open faces Γ_k^e , such that $\partial K^e = \cup_{k=1}^{n_f} \overline{\Gamma_k^e}$. In this context, we use the term face to describe the $(d-1)$ -dimensional interfaces with other elements (points in 1D, lines in 2D,...). By $n_f = n_f(e)$ we denote the cardinality of this partition (e.g. $n_f = 2$ in 1D, $n_f = 3$ for triangular elements, $n_f = 4$ for quadrilaterals,...). Due to the discontinuous space approximation (2), U_h^e is not uniquely defined in $\mathbf{x} \in \Gamma_k^e$; its one-sided limits are given by

$$U_h^e(\mathbf{x}, t) = U_h^{e,-}(\mathbf{x}, t) := \lim_{\varepsilon \rightarrow 0_+} U_h^e(\mathbf{x} - \varepsilon \mathbf{n}^e, t),$$

$$\hat{U}_h^e(\mathbf{x}, t) = U_h^{e,+}(\mathbf{x}, t) := \lim_{\varepsilon \rightarrow 0_+} U_h^e(\mathbf{x} + \varepsilon \mathbf{n}^e, t),$$

i.e., values from the interior are denoted by U_h^e and exterior ones by \hat{U}_h^e .

To obtain local conservation, we replace $\mathbf{F}(U_h^e) \mathbf{n}^e$ in (4) by a numerical flux $\mathcal{H}(U_h^e, \hat{U}_h^e; \mathbf{n}^e)$, satisfying the following properties

$$\mathcal{H}(U, U; \mathbf{n}^e) = \mathbf{F}(U) \mathbf{n}^e \quad (\text{consistency}), \quad (5a)$$

$$\mathcal{H}(U, V; \mathbf{n}^e) + \mathcal{H}(V, U; -\mathbf{n}^e) = 0 \quad (\text{conservation}). \quad (5b)$$

In our numerical experiments, we use the local Lax-Friedrichs flux

$$\mathcal{H}(U, V; \mathbf{n}^e) = \frac{1}{2} (\mathbf{F}(U) + \mathbf{F}(V)) \mathbf{n}^e + \frac{\lambda}{2} (U - V), \quad (6)$$

where λ is an estimate on the speed of fastest wave propagating along \mathbf{n}^e

$$\lambda = \lambda(U, V; \mathbf{n}^e) \geq \max_{\omega \in [0,1]} \text{spr}(\mathbf{A}(\omega U + (1-\omega)V; \mathbf{n}^e)) \quad (7)$$

$$\mathbf{A}(U; \mathbf{n}^e) = \frac{\partial}{\partial U} (\mathbf{F}(U) \mathbf{n}^e) \in \mathbb{R}^{m \times m},$$

and $\text{spr}(\cdot)$ denotes the spectral radius of a matrix. For scalar conservation laws having a convex flux \mathbf{F} , the maximum in (7) is attained at one of the two states (either U or V). Practical upper bounds for more general problems such as the Euler and shallow-water equations can be found in [19, 22].

Remark 1. Our methodology is in no way restricted to using the Lax-Friedrichs flux in the target scheme. In fact, we show that any flux satisfying (5) fits into our framework. However, the low order method presented in the next section, does indeed rely on a Lax-Friedrichs flux. Setting one component of V_h equal to φ_i^e and the other components equal to zero, we test (4) with V_h and arrive at the semi-discrete system of equations for the target scheme

$$\int_{K^e} \varphi_i^e \frac{\partial U_h^e}{\partial t} d\mathbf{x} = \int_{K^e} \mathbf{F}(U_h^e) \nabla \varphi_i^e d\mathbf{x} - \sum_{k=1}^{n_f} \int_{\Gamma_k^e} \varphi_i^e \mathcal{H}(U_h^e, \hat{U}_h^e; \mathbf{n}^e) ds, \quad (8)$$

where $i \in \{1, \dots, N\}$ and $e \in \{1, \dots, E\}$.

3. Derivation of the low order method

The standard DG discretization presented in Section 2 is valid for any basis of $\mathbb{V}_p(\mathcal{T}_h)$. However, the approach presented in this section relies on the use of Bernstein (also called Bézier) polynomials b_i^p . On $[0, 1]$ these functions are defined as

$$b_i^p(x) = \binom{p}{i} (1-x)^{p-i} x^i, \quad i \in \{0, \dots, p\}.$$

Definitions for other reference elements and further properties can be found in the literature, e.g., [3, 4, 29, 43]. Bernstein polynomials possess a number of advantages compared to other bases. In particular, they attain values in $[0, 1]$ and form a partition of unity. Therefore, a numerical approximation defined as a linear combination of Bernstein polynomials is bounded in the following way:

$$\min_{j \in \{1, \dots, N\}} u_j^e \leq \sum_{j=1}^N u_j^e \varphi_j^e(\mathbf{x}) \leq \max_{j \in \{1, \dots, N\}} u_j^e.$$

The latter property makes the Bernstein basis a natural choice for constraining numerical solutions to satisfy discrete maximum principles which do not necessarily hold, e.g., for an interpolatory Lagrange basis even if the coefficients of the finite element approximation are bounded as desired. Similarly to Lagrange basis polynomials, their Bernstein counterparts can be associated with certain *nodes* $\mathbf{x}_i^e \in K^e$, which are uniformly distributed within the elements. Due to the DG setting, we need to distinguish between element-local nodes \mathbf{x}_i^e and physical locations $\mathbf{x}_I \in \{\mathbf{x}_i^e \in \Omega : e \in \{1, \dots, E\}, i \in \{1, \dots, N\}\}$ of these points, which are assigned unique global numbers I and may be shared by boundary nodes of adjacent elements.

The semi-discrete formulation presented in Section 2 satisfies the IDP property only when piecewise constant basis functions are employed. To obtain a low order method that is IDP in general, we make a number of modifications to (8). A first simplification due to the use of Bernstein polynomials is that the boundary integral reduces to integration over the set of

faces Γ_k^e , $k \in \mathcal{F}_i$ which contain the corresponding node. Reversing integration by parts, we write (8) in the equivalent form

$$\begin{aligned} \int_{K^e} \varphi_i^e \frac{\partial U_h^e}{\partial t} d\mathbf{x} &= - \int_{K^e} \varphi_i^e \nabla \cdot \mathbf{F}(U_h^e) d\mathbf{x} \\ &\quad - \sum_{k \in \mathcal{F}_i} \int_{\Gamma_k^e} \varphi_i^e \left[\mathcal{H}(U_h^e, \hat{U}_h^e; \mathbf{n}^e) - \mathbf{F}(U_h^e) \mathbf{n}^e \right] ds. \end{aligned}$$

Next, we linearize the volume term using the *group finite element formulation*

$$\mathbf{F}_j^e := \mathbf{F}(u_j^e), \quad j \in \{1, \dots, N\}, \quad \mathbf{F}_h^e := \sum_{j=1}^N \mathbf{F}_j^e \varphi_j^e \approx \mathbf{F}(U_h^e) = \mathbf{F} \left(\sum_{j=1}^N u_j^e \varphi_j^e \right).$$

This approximation produces

$$\begin{aligned} \sum_{j=1}^N m_{ij}^e \frac{du_j^e}{dt} &= - \sum_{j=1}^N \mathbf{F}_j^e \mathbf{c}_{ij}^e - \sum_{k \in \mathcal{F}_i} \int_{\Gamma_k^e} \varphi_i^e \left[\mathcal{H}(U_h^e, \hat{U}_h^e; \mathbf{n}^e) - \mathbf{F}(U_h^e) \mathbf{n}^e \right] ds, \\ \text{where } m_{ij}^e &= \int_{K^e} \varphi_i^e \varphi_j^e d\mathbf{x}, \quad \mathbf{c}_{ij}^e = \int_{K^e} \varphi_i^e \nabla \varphi_j^e d\mathbf{x}. \end{aligned}$$

Since Bernstein polynomials form a partition of unity, the matrices $\mathbf{C}^e = (C_1^e, \dots, C_d^e)^T = (\mathbf{c}_{ij}^e)_{i,j=1}^N$ have zero row sums, which implies $\sum_{j=1}^N \mathbf{F}_j^e \mathbf{c}_{ij}^e = \sum_{j=1}^N (\mathbf{F}_j^e - \mathbf{F}_i^e) \mathbf{c}_{ij}^e$. The next step toward constructing an IDP low order method is mass lumping. At this stage, the consistent mass matrix $M_C^e = (m_{ij}^e)_{i,j=1}^N$ is transformed into its lumped counterpart $M_L^e = \text{diag}(m_i^e)_{i=1}^N$ by applying the *preconditioning operator* $P^e = M_L^e (M_C^e)^{-1}$. Following the approach proposed in [39, 43], we apply P^e to the element matrices stored in \mathbf{C}^e as well, which yields

$$m_i^e \frac{du_i^e}{dt} = - \sum_{j \in \tilde{\mathcal{N}}_i} (\mathbf{F}_j^e - \mathbf{F}_i^e) \tilde{\mathbf{c}}_{ij}^e - \sum_{k \in \mathcal{F}_i} \int_{\Gamma_k^e} \varphi_i^e \left[\mathcal{H}(U_h^e, \hat{U}_h^e; \mathbf{n}^e) - \mathbf{F}(U_h^e) \mathbf{n}^e \right] ds, \quad (9)$$

$$\text{where } \tilde{\mathbf{C}}^e = (\tilde{C}_1^e, \dots, \tilde{C}_d^e)^T = (\tilde{\mathbf{c}}_{ij}^e)_{i,j=1}^N, \quad \tilde{C}_k^e = M_L^e (M_C^e)^{-1} C_k^e.$$

One can show that the element matrices $\tilde{\mathbf{C}}_k^e$ also have zero row sums and their entries are non-zero only if nodes \mathbf{x}_i^e and \mathbf{x}_j^e are nearest neighbors within the element [39, 43]. This fact allows us to restrict summation to the smaller index set $\tilde{\mathcal{N}}_i \subset \{1, \dots, N\}$ consisting of only the indices j whose nodes are nearest neighbors to node \mathbf{x}_i^e .

Remark 2. For high order elements, consistent mass matrices M_C^e become extremely ill-conditioned and, therefore, one should not rely on their inversion to compute $\tilde{\mathbf{C}}_k^e$. In the appendix, we present formulas for the entries of $\tilde{\mathbf{C}}_k^e$ on box elements, as well as a MATLAB code for computing $\tilde{\mathbf{C}}_k^e$ on simplices.

Next, we introduce dissipative matrices $D^e = (d_{ij}^e)_{i,j=1}^N$ defined by [18, 39, 40]

$$d_{ij}^e = \begin{cases} \max\{|\tilde{\mathbf{c}}_{ij}^e| \lambda_{ij}^e, |\tilde{\mathbf{c}}_{ji}^e| \lambda_{ji}^e\} & \text{if } i \neq j, \\ -\sum_{k=1}^N d_{ik}^e & \text{otherwise,} \end{cases} \quad (10)$$

where $\lambda_{ij}^e = \lambda(u_i^e, u_j^e; \mathbf{n}_{ij}^e)$, $\mathbf{n}_{ij}^e = \frac{\tilde{\mathbf{c}}_{ij}^e}{|\tilde{\mathbf{c}}_{ij}^e|}$.

To stabilize the volume integral, the term $\sum_{j \in \tilde{\mathcal{N}}_i} d_{ij}^e (u_j^e - u_i^e)$ is added to (9). Note that summation is over the compact stencil $\tilde{\mathcal{N}}_i$. The sparsity of D^e follows from (10) and is the main advantage of applying the preconditioner P^e to the element matrices of the discrete gradient operator. A sparsified diffusion operator implies that $d_{ij}^e (u_j^e - u_i^e) = 0$ if i and j are far away nodes within a high order element. Diffusive fluxes between such nodes are nonphysical and tend to be large. The compact-stencil version produces nonvanishing fluxes only for pairs of nearest neighbor nodes. The necessity of such sparsification for high order spaces has been recognized in [27, 39, 43, 44].

Remark 3. The analysis in the appendix of [39] proves the **all**-nearest-neighbors sparsity pattern for box elements in more than one dimension. In fact, the element matrices C_k^e have more zero entries than this analysis suggests. We prove in the appendix of this paper, that nodes which are closest neighbors in the diagonal direction, produce a zero entry in \tilde{C}_k^e . On the one hand, this remarkable property implies lower computational cost since fewer pairs of nodes need to be considered (e.g., $2d$ neighbors for one interior node of an element instead of $3^d - 1$). On the other hand, the low order method bears an interesting resemblance to the spectral element method in [44]: Dissipation is only added in Cartesian directions, not along diagonals.

Having discussed the stabilization of volume terms, let us now modify the DG face integrals in order to obtain the low order IDP scheme. To this end, we generalize our interface treatment for the advection equation [27, 28]. In the 1D case, face contributions to the global system matrix already have the right signs and do not inhibit the IDP property. Due to positivity of Bernstein polynomials and the use of upwind (Lax-Friedrichs) fluxes, off-diagonal entries corresponding to the 1D face terms (i.e. contributions from other elements) are nonnegative, while diagonal entries are nonpositive. In multiple dimensions, the off-diagonal entries within the diagonal blocks of the matrix need to be “lumped” in a conservative manner. This *flux lumping* is accomplished by replacing interior and exterior contributions U_h^e, \hat{U}_h^e by their nodal counterparts, i.e., solution coefficients u_i^e, \hat{u}_i^e , which yields

$$m_i^e \frac{du_i^e}{dt} = \sum_{j \in \tilde{\mathcal{N}}_i} [d_{ij}^e (u_j^e - u_i^e) - (\mathbf{F}_j^e - \mathbf{F}_i^e) \tilde{\mathbf{c}}_{ij}^e] - \sum_{k \in \mathcal{F}_i} \int_{\Gamma_k^e} \varphi_i^e [\mathcal{H}(u_i^e, \hat{u}_i^e; \mathbf{n}^e) - \mathbf{F}(u_i^e) \mathbf{n}^e] ds.$$

Note that for nodes i located at a junction of two or more interfaces (e.g. a vertex in 2D) \hat{u}_i^e implicitly depends on the index k of the face (in 2D, there are two such neighbors, one for

each face). At this stage, the use of Lax-Friedrichs flux (6) is beneficial since it allows us to write the face term as

$$\begin{aligned}\mathcal{H}(u_i^e, \hat{u}_i^e; \mathbf{n}^e) - \mathbf{F}(u_i^e) \mathbf{n}^e &= \frac{1}{2} [(\mathbf{F}(\hat{u}_i^e) - \mathbf{F}(u_i^e)) \mathbf{n}^e + \lambda_i^e (u_i^e - \hat{u}_i^e)] \\ &= \frac{1}{2} [(\hat{\mathbf{F}}_i^e - \mathbf{F}_i^e) \mathbf{n}^e + \lambda_i^e (u_i^e - \hat{u}_i^e)],\end{aligned}$$

where we introduced the notation $\hat{\mathbf{F}}_i^e := \mathbf{F}(\hat{u}_i^e)$, and the nodal wave speed $\lambda_i^e = \lambda(u_i^e, \hat{u}_i^e; \mathbf{n}^e)$. In summary, the low order scheme reads

$$\begin{aligned}m_i^e \frac{du_i^e}{dt} &= \sum_{j \in \mathcal{N}_i} [d_{ij}^e (u_j^e - u_i^e) - (\mathbf{F}_j^e - \mathbf{F}_i^e) \tilde{\mathbf{c}}_{ij}^e] \\ &\quad + \sum_{k \in \mathcal{F}_i} \frac{1}{2} \int_{\Gamma_k^e} \varphi_i^e ds [(\mathbf{F}_i^e - \hat{\mathbf{F}}_i^e) \mathbf{n}^e + \lambda_i^e (\hat{u}_i^e - u_i^e)].\end{aligned}\quad (11)$$

Equation (11) constitutes the semi-discrete form of our low order method. Its IDP property follows from its equivalent *bar state* form [18]. Similarly to volume integrals that can be expressed in terms of two-node bar states \bar{u}_{ij}^e , DG fluxes between nodes corresponding to the same physical location in two different elements can be written in terms of interfacial bar states $\bar{u}_{i,k}^e, k \in \mathcal{F}_i$. The volume and face bar states of our DG method are defined by

$$\bar{u}_{ij}^e = \frac{u_i^e + u_j^e}{2} - \frac{(\mathbf{F}_j^e - \mathbf{F}_i^e) \tilde{\mathbf{c}}_{ij}^e}{2d_{ij}^e}, \quad \bar{u}_{i,k}^e = \frac{u_i^e + \hat{u}_i^e}{2} - \frac{(\hat{\mathbf{F}}_i^e - \mathbf{F}_i^e) \mathbf{n}^e}{2\lambda_i^e},\quad (12)$$

respectively. Writing (11) in terms of (12) produces

$$m_i^e \frac{du_i^e}{dt} = \sum_{j \in \mathcal{N}_i} 2d_{ij}^e (\bar{u}_{ij}^e - u_i^e) + \sum_{k \in \mathcal{F}_i} 2d_{i,k}^e (\bar{u}_{i,k}^e - u_i^e),\quad (13)$$

$$\text{where } d_{i,k}^e = \frac{\lambda_i^e}{2} \int_{\Gamma_k^e} \varphi_i^e ds$$

and $d_{i,k}^e > 0$ due to the positivity of the Bernstein basis.

Remark 4. By definition, the volumetric bar states \bar{u}_{ij}^e and \bar{u}_{ji}^e coincide if $\tilde{\mathbf{c}}_{ij}^e + \tilde{\mathbf{c}}_{ji}^e = 0$. This skew-symmetry condition holds if (i) no preconditioning is performed for the element matrices \mathbf{C}^e of the discrete gradient operator and (ii) at least one of the two nodes is located in the interior of K^e . The sparse preconditioned element matrices $\tilde{\mathbf{C}}^e$ are not skew-symmetric. Therefore, \bar{u}_{ij}^e and \bar{u}_{ji}^e may differ even if i or j is an internal node. However, an interfacial bar state $\bar{u}_{i,k}^e$ always equals the bar state $\bar{u}_{i',k'}^e$ of the node $\mathbf{x}_{i'}$ occupying the same physical location \mathbf{x}_I in an adjacent element $K^{e'}$. Indeed, we have

$$\bar{u}_{i',k'}^e = \frac{u_{i'}^{e'} + \hat{u}_{i'}^{e'}}{2} - \frac{(\hat{\mathbf{F}}_{i'}^{e'} - \mathbf{F}_{i'}^{e'}) \mathbf{n}^{e'}}{2\lambda_{i'}^{e'}} = \frac{\hat{u}_i^e + u_i^e}{2} - \frac{(\hat{\mathbf{F}}_i^e - \mathbf{F}_i^e) \mathbf{n}^e}{2\lambda_i^e} = \bar{u}_{i,k}^e.\quad (14)$$

In [18], the authors prove that \bar{u}_{ij}^e is in the invariant set \mathcal{G} . Following their proof, we now show that the same property holds for $\bar{u}_{i,k}^e$.

Theorem 1. Assume that the one-dimensional Riemann problem

$$\frac{\partial U}{\partial t} + \frac{\partial}{\partial x} (\mathbf{F}(U) \mathbf{n}^e) = 0, \quad \text{in } \mathbb{R} \times \mathbb{R}_+, \quad U(x, 0) = \begin{cases} u_i^e & \text{if } x < 0, \\ \hat{u}_i^e & \text{if } x > 0 \end{cases} \quad (15)$$

has a unique solution $U(\mathbf{n}^e, u_i^e, \hat{u}_i^e)$. Then the flux bar states $\bar{u}_{i,k}^e$ defined by (12) belong to the invariant set \mathcal{G} .

Proof. Consider the *fake time* $\tau = |\mathbf{n}^e| / (2\lambda_i^e) = 1 / (2\lambda_i^e)$. By definition (7), λ_i^e is a guaranteed upper bound for the maximum wave speed λ_{\max} of (15), i.e., $\tau\lambda_{\max} \leq \frac{1}{2}$. Hence, following the analysis of the 1D Riemann problem in [18], we can rewrite the averaged solution to (15) as

$$\bar{U}(\mathbf{n}^e, u_i^e, \hat{u}_i^e) := \int_{-\frac{1}{2}}^{\frac{1}{2}} U(\mathbf{n}^e, u_i^e, \hat{u}_i^e)(x, \tau) \, dx = \frac{u_i^e + \hat{u}_i^e}{2} - \frac{(\hat{\mathbf{F}}_i^e - \mathbf{F}_i^e) \mathbf{n}^e}{2\lambda_i^e} = \bar{u}_{i,k}^e.$$

Since $\bar{U}(\mathbf{n}^e, u_i^e, \hat{u}_i^e) \in \mathcal{G}$, this representation of $\bar{u}_{i,k}^e$ implies the statement of the theorem. \square

The forward Euler time discretization of (13) preserves the IDP property if the time step Δt is sufficiently small. The updated solution

$$\tilde{u}_i^e = u_i^e - \frac{2\Delta t}{m_i^e} \left(\sum_{j \in \tilde{\mathcal{N}}_i} d_{ij}^e + \sum_{k \in \mathcal{F}_i} d_{i,k}^e \right) u_i^e + \frac{2\Delta t}{m_i^e} \left(\sum_{j \in \tilde{\mathcal{N}}_i} d_{ij}^e \bar{u}_{ij}^e + \sum_{k \in \mathcal{F}_i} d_{i,k}^e \bar{u}_{i,k}^e \right) \quad (16)$$

is a convex combination of u_i^e and the bar states \bar{u}_{ij}^e , $\bar{u}_{i,k}^e$, provided that

$$\frac{2\Delta t}{m_i^e} \left(\sum_{j \in \tilde{\mathcal{N}}_i} d_{ij}^e + \sum_{k \in \mathcal{F}_i} d_{i,k}^e \right) \leq 1. \quad (17)$$

Corollary 1. If (17) holds, then \tilde{u}_i^e is a convex combination of u_i^e and the bar states, which are members of \mathcal{G} . This proves the IDP property of (11).

Remark 5. It is possible to write (13) using a unified notation for volumetric and interfacial bar states. Disguising the different nature of underlying approximations, the resulting representation resembles the bar state forms of the low order continuous finite element methods considered in [18, 39]. Let \mathcal{E}^e denote the DG element stencil, i.e., the integer set containing the numbers of element K^e and its common face neighbors. Furthermore, define

$$\mathcal{N}_i^{e,e'} = \begin{cases} \tilde{\mathcal{N}}_i & \text{if } e = e', \\ \{j \in \{1, \dots, N\} : \mathbf{x}_j^{e'} = \mathbf{x}_i^e\} & \text{otherwise,} \end{cases}$$

and

$$d_{ij}^{e,e'} = \begin{cases} d_{ij}^e & \text{if } e = e', \\ d_{i,k}^e & \Gamma_k^e = K^e \cap K^{e'}, \end{cases} \quad \bar{u}_{ij}^{e,e'} = \begin{cases} \bar{u}_{ij}^e & \text{if } e = e', \\ \bar{u}_{i,k}^e & \Gamma_k^e = K^e \cap K^{e'}. \end{cases}$$

Then (13) becomes

$$m_i^e \frac{du_i^e}{dt} = \sum_{e' \in \mathcal{E}^e} \sum_{j \in \mathcal{N}_i^{e,e'}} 2d_{ij}^{e,e'} (\bar{u}_{ij}^{e,e'} - u_i^e). \quad (18)$$

Since the volume and flux terms are treated differently in practice, we avoid using the compressed single-sum representation (18). However, it is quite remarkable that the bar state form of the low order IDP method analyzed in [18] can be generalized to the DG setting in such a way.

Remark 6. Condition (17) provides a valuable property for time step control. If the solution of (13) is advanced in time using the forward Euler method, the IDP property is guaranteed for time steps Δt satisfying

$$\Delta t \leq \min_{e \in \{1, \dots, E\}} \min_{i \in \{1, \dots, N\}} \frac{m_i^e}{2 \left(\sum_{j \in \tilde{\mathcal{N}}_i} d_{ij}^e + \sum_{k \in \mathcal{F}_i} d_{i,k}^e \right)}. \quad (19)$$

In higher order strong stability preserving (SSP) Runge-Kutta methods [16], each stage corresponds to an update of the form (16). Hence the validity of (19) is sufficient to ensure the IDP property for each stage and for the final result which represents a convex combination of IDP approximations.

4. Monolithic limiting approach

This section describes all mathematical aspects of our limiter starting with the definition of raw antidiffusive fluxes. It then summarizes the subcell limiting approach of [36] and extends it to the interfacial bar states of flux-corrected DG approximations. Next, we describe the general methodology for limiting hyperbolic systems and define admissible bounds for the inequality constraints of each step. For the reader's convenience, we give a brief summary of the presented limiting techniques at the end of this section.

4.1. Raw antidiffusive fluxes

In order to adapt the convex limiting strategy originally proposed in [36, 39] to the DG setting, we must first define raw antidiffusive terms f_i^e , $f_{i,k}^e$ which transform the low order method (11) into the compact-stencil form

$$\begin{aligned} m_i^e \frac{du_i^e}{dt} &= \sum_{j \in \tilde{\mathcal{N}}_i} [d_{ij}^e (u_j^e - u_i^e) - (\mathbf{F}_j^e - \mathbf{F}_i^e) \tilde{\mathbf{c}}_{ij}^e] + f_i^e \\ &+ \sum_{k \in \mathcal{F}_i} \left(\frac{1}{2} \int_{\Gamma_k^e} \varphi_i^e ds \left[(\mathbf{F}_i^e - \hat{\mathbf{F}}_i^e) \mathbf{n}^e + \lambda_i^e (\hat{u}_i^e - u_i^e) \right] + f_{i,k}^e \right) \end{aligned} \quad (20)$$

of the target scheme (8). These correction terms are given by

$$\begin{aligned}
f_i^e &= \sum_{j=1}^N (m_i^e \delta_{ij} - m_{ij}^e) \dot{u}_j^e - \sum_{j \in \tilde{\mathcal{N}}_i} [d_{ij}^e (u_j^e - u_i^e) - (\mathbf{F}_j^e - \mathbf{F}_i^e) \tilde{\mathbf{c}}_{ij}^e] \\
&\quad + \int_{K^e} \mathbf{F}(U_h^e) \nabla \varphi_i^e \, d\mathbf{x} - \sum_{k \in \mathcal{F}_i} \int_{\Gamma_k^e} \varphi_i^e \mathbf{F}_i^e \mathbf{n}^e \, ds
\end{aligned} \tag{21}$$

and

$$f_{i,k}^e = \int_{\Gamma_k^e} \varphi_i^e \left[\mathcal{H}(u_i^e, \hat{u}_i^e; \mathbf{n}^e) - \mathcal{H}(U_h^e, \hat{U}_h^e; \mathbf{n}^e) \right] \, ds. \tag{22}$$

The vector $\dot{u}^e = (\dot{u}_j^e)_{j=1}^N$ of nodal time derivatives in element K^e is obtained by solving (8).

Lemma 1. The semi-discrete target scheme (8) is equivalent to (20).

Proof. It is easy to verify that substitution of f_i^e and $f_{i,k}^e$ into (20) reverses the effect of algebraic manipulations performed in Section 3. Moving the time derivative term to the left hand side, we rewrite (20) as

$$\begin{aligned}
\sum_{j=1}^N m_{ij}^e \frac{du_j^e}{dt} &= \int_{K^e} \mathbf{F}(U_h^e) \nabla \varphi_i^e \, d\mathbf{x} - \sum_{k \in \mathcal{F}_i} \int_{\Gamma_k^e} \varphi_i^e \mathbf{F}_i^e \mathbf{n}^e \, ds \\
&\quad + \sum_{k \in \mathcal{F}_i} \left(\frac{1}{2} \int_{\Gamma_k^e} \varphi_i^e \, ds \left[(\mathbf{F}_i^e - \hat{\mathbf{F}}_i^e) \mathbf{n}^e + \lambda_i^e (\hat{u}_i^e - u_i^e) \right] \right) \\
&\quad + \sum_{k \in \mathcal{F}_i} \int_{\Gamma_k^e} \varphi_i^e \left[\mathcal{H}(u_i^e, \hat{u}_i^e; \mathbf{n}^e) - \mathcal{H}(U_h^e, \hat{U}_h^e; \mathbf{n}^e) \right] \, ds \\
&= \int_{K^e} \mathbf{F}(U_h^e) \nabla \varphi_i^e \, d\mathbf{x} - \sum_{k \in \mathcal{F}_i} \int_{\Gamma_k^e} \varphi_i^e \mathcal{H}(U_h^e, \hat{U}_h^e; \mathbf{n}^e) \, ds \\
&\quad + \underbrace{\sum_{k \in \mathcal{F}_i} \int_{\Gamma_k^e} \varphi_i^e \, ds \left(\mathcal{H}(u_i^e, \hat{u}_i^e; \mathbf{n}^e) - \frac{1}{2} \left[(\mathbf{F}_i^e + \hat{\mathbf{F}}_i^e) \mathbf{n}^e + \lambda_i^e (u_i^e - \hat{u}_i^e) \right] \right)}_{=0}.
\end{aligned}$$

This semi-discrete scheme is, indeed, equivalent to (8). \square

Lemma 2. The antidiffusive fluxes f_i^e defined by (21) satisfy $\sum_{i=1}^N f_i^e = 0$ for all $e \in \{1, \dots, E\}$.

Proof. By definition, the contributions of mass and dissipative matrices sum to zero. Using the zero row sum property of \tilde{C}_k^e , the partition of unity property of the Bernstein basis and

integration by parts, we find that

$$\begin{aligned}
\sum_{i=1}^N f_i^e &= \sum_{i=1}^N \left[\sum_{j \in \tilde{\mathcal{N}}_i} (\mathbf{F}_j^e - \mathbf{F}_i^e) \tilde{\mathbf{c}}_{ij}^e - \sum_{k \in \mathcal{F}_i} \int_{\Gamma_k^e} \varphi_i^e \mathbf{F}_i^e \mathbf{n}^e ds \right] \\
&= \sum_{i=1}^N \sum_{j=1}^N \mathbf{F}_j^e (\tilde{\mathbf{c}}_{ij}^e - \mathbf{c}_{ij}^e + \mathbf{c}_{ij}^e) - \sum_{i=1}^N \int_{\partial K^e} \varphi_i^e \mathbf{F}_i^e \mathbf{n}^e ds \\
&= \sum_{i,j=1}^N \left[\mathbf{F}_j^e (\tilde{\mathbf{c}}_{ij}^e - \mathbf{c}_{ij}^e) - \mathbf{F}_j^e \mathbf{c}_{ji}^e + \mathbf{F}_j^e \int_{\partial K^e} \varphi_i^e \varphi_j^e \mathbf{n}^e ds \right] - \sum_{i=1}^N \int_{\partial K^e} \varphi_i^e \mathbf{F}_i^e \mathbf{n}^e ds \\
&= \sum_{j=1}^N \mathbf{F}_j^e \left[\sum_{i=1}^N (\tilde{\mathbf{c}}_{ij}^e - \mathbf{c}_{ij}^e) - \sum_{i=1}^N \mathbf{c}_{ji}^e \right] + \sum_{i,j=1}^N (\mathbf{F}_j^e - \mathbf{F}_i^e) \int_{\partial K^e} \varphi_i^e \varphi_j^e \mathbf{n}^e ds = 0.
\end{aligned}$$

The final step exploits the partition of unity property and the fact that the matrices $\tilde{C}_k^e - C_k^e$ have zero column sums, cf. [39]. \square

Remark 7. The antidiffusive DG fluxes $f_{i,k}^e$ contain the difference between the low and high order numerical fluxes. They constitute a straightforward generalization of the corresponding terms arising in limiters for DG discretizations of advection problems [27, 28]. As was the case there, $f_{i,k}^e$ vanishes in 1D because face integrals correspond to point evaluations and only one Bernstein polynomial is non-zero at the end points of a 1D element.

The next lemma is a direct consequence of (5) and of the fact that local basis functions from different elements coincide along the shared face if the corresponding nodes are associated with the same physical location.

Lemma 3. Let $K^{e'}$ be the neighbor element of K^e along face $\Gamma_k^e = \Gamma_{k'}^{e'}$ and $\mathbf{x}_{i'}^{e'}$ be the node in $K^{e'}$ with the same physical location as node $\mathbf{x}_i^e \in K^e$. Then the antidiffusive DG fluxes satisfy $f_{i,k}^e = -f_{i',k'}^{e'}$.

4.2. Limiting strategy for volumetric fluxes

This section summarizes the methods originally proposed in [36, 39] for limiting antidiffusive element contributions in continuous Galerkin methods. These approaches are also applicable to DG volume terms. Due to Lemma 2, we can split f_i^e into fluxes f_{ij}^e between pairs of nodes as follows:

$$f_i^e = \sum_{j \in \tilde{\mathcal{N}}_i \setminus \{i\}} f_{ij}^e, \quad f_{ij}^e = -f_{ji}^e \quad \forall j \in \tilde{\mathcal{N}}_i \setminus \{i\}. \quad (23)$$

To preserve the compact sparsity pattern of the low order scheme, it is worthwhile to use a flux decomposition such that f_{ij}^e is nonzero if and only if \mathbf{x}_i^e and \mathbf{x}_j^e are closest neighbors. Following the design principle in [39], we use a natural decomposition of the term $\sum_{j \in \tilde{\mathcal{N}}_i} d_{ij}^e (u_i^e - u_j^e)$ into

antidiffusive fluxes $d_{ij}^e(u_i^e - u_j^e)$ which possess the compact stencil property by construction. The remaining components

$$q_i^e = f_i^e + \sum_{j \in \tilde{\mathcal{N}}_i} d_{ij}^e(u_j^e - u_i^e)$$

also sum to zero and are now distributed in the following manner. Let $\tilde{M}_C^e = (\tilde{m}_{ij}^e)_{i,j=1}^N$ and \tilde{M}_L^e be consistent and lumped element matrices of the continuous, piecewise \mathbb{V}_1 shape functions on Bézier subcells of K^e (as defined, e.g., in [27, 39]). On simplices, we define auxiliary vectors $v^e \in \mathbb{R}^N$ as solutions to $(\tilde{M}_L^e - \tilde{M}_C^e)v^e = q^e$ and use them to construct the antidiffusive fluxes

$$f_{ij}^e = \tilde{m}_{ij}^e(v_i^e - v_j^e) + d_{ij}^e(u_i^e - u_j^e) \quad \forall j \in \tilde{\mathcal{N}}_i \setminus \{i\} \quad (24)$$

satisfying (23). Just as in the low order method, fluxes of box elements should be zero for diagonal neighbors. Therefore, we modify the element matrices \tilde{M}_C^e of such Bernstein elements by performing partial mass lumping which sets each off-diagonal entry \tilde{m}_{ij}^e associated with a diagonal neighbor $j \in \tilde{\mathcal{N}}_i \setminus \{i\}$ of node i to zero and adds it to the diagonal element \tilde{m}_{ii}^e . As a consequence of this modification, the corresponding fluxes $\tilde{m}_{ij}^e(v_i^e - v_j^e)$ become equal to zero and definition (24) produces $f_{ij}^e = 0$ whenever $d_{ij}^e = 0$.

Remark 8. The sparse element matrix $\tilde{M}_L^e - \tilde{M}_C^e$ is a Poisson-type operator in the sense that it is symmetric positive semi-definite with zero row sums and negative off-diagonal entries. The solution v^e of the linear system involving this matrix is defined up to a constant which has no influence on the flux f_{ij}^e defined by (24). We fix the value of this arbitrary constant by setting each entry in the last row of $\tilde{M}_L^e - \tilde{M}_C^e$ to one. To avoid the relatively high cost of inverting the resulting non-singular matrix repeatedly, we define it on the reference element and store its inverse.

In the process of limiting, we multiply f_{ij}^e and f_{ji}^e with correction factors $\alpha_{ij}^e = \alpha_{ji}^e \in [0, 1]$, which retains the conservation property established in Lemma 2. The limited fluxes $f_{ij}^{e,*} = \alpha_{ij}^e f_{ij}^e$ transform the bar states $\bar{u}_{ij}^e \in \mathcal{G}$ of the low order scheme (13) into

$$\bar{u}_{ij}^{e,*} = \bar{u}_{ij}^e + \frac{f_{ij}^{e,*}}{2d_{ij}^e} \in \mathcal{G}_i^e \cap \mathcal{G}, \quad (25)$$

where \mathcal{G}_i^e is a set of admissible states defined by the solution U_h at the beginning of the current Runge-Kutta stage or iteration. Let

$$\mathcal{G}_i^e = \{u_i^e \in \mathbb{R}^m : u_i^{e,\min} \cdot e_l \leq u_i^e \cdot e_l \leq u_i^{e,\max} \cdot e_l, \quad l = 1, \dots, m\}$$

be defined using the unit vector e_l of \mathbb{R}^m to pick out the inequality constraints for the l -th conserved variable. We select numerically admissible bounds $u_i^{e,\min}, u_i^{e,\max}$ for all steps of the limiting process in Section 4.5 after completing the general presentation of our method.

For now, let us assume that generic bounds $u_i^{e,\min}, u_i^{e,\max}$ to be imposed on u_i^e are available. Following Kuzmin [36], we define the ‘monolithically’ limited fluxes

$$f_{ij}^{e,*} = \begin{cases} \min \{ f_{ij}^e, 2d_{ij}^e \min \{ u_i^{e,\max} - \bar{u}_{ij}^e, \bar{u}_{ji}^e - u_j^{e,\min} \} \} & \text{if } f_{ij}^e \geq 0, \\ \max \{ f_{ij}^e, 2d_{ij}^e \max \{ u_i^{e,\min} - \bar{u}_{ij}^e, \bar{u}_{ji}^e - u_j^{e,\max} \} \} & \text{if } f_{ij}^e < 0 \end{cases} \quad (26)$$

directly instead of calculating a correction factor α_{ij}^e and applying it to f_{ij}^e . The semi-discrete bar state form utilizing the limited fluxes $f_{ij}^{e,*}$ is obtained by replacing \bar{u}_{ij}^e in (13) with $\bar{u}_{ij}^{e,*}$. This is equivalent to replacing f_i^e in (20) by $f_i^{e,*} = \sum_{j \in \mathcal{N}_i \setminus \{i\}} f_{ij}^{e,*}$. The latter form is better suited for practical implementation purposes. As explained in [36], direct calculation of the products $2d_{ij}^e \bar{u}_{ij}^e$ and $2d_{ij}^e \bar{u}_{ij}^{e,*}$ makes it possible to avoid unnecessary division and multiplication by d_{ij}^e . Hence, the limited fluxes $f_{ij}^{e,*}$ defined by (26) and the contribution of each node pair to the (flux-corrected) bar state form can be calculated without using the formal definition of \bar{u}_{ij}^e and $\bar{u}_{ij}^{e,*}$.

4.3. Limiting strategy for interfacial fluxes

To preserve the conservation property $f_{i,k}^e = -f_{i',k'}^e$ of the antidiffusive DG fluxes defined by (22), we need to use the same (virtual) correction factor for $f_{i,k}^e$ and $f_{i',k'}^e$. This is achieved similarly to the flux limiter for f_{ij}^e . Once again, we assume the availability of generic bounds satisfying

$$u_i^{e,\min} = u_{i'}^{e',\min}, \quad u_i^{e,\max} = u_{i'}^{e',\max}. \quad (27)$$

To enforce these local bounds in a way which guarantees that conservation is not inhibited, the limited interfacial fluxes are defined by

$$f_{i,k}^{e,*} = \begin{cases} \min \{ f_{i,k}^e, 2d_{i,k}^e \min \{ u_i^{e,\max} - \bar{u}_{i,k}^e, \bar{u}_{i,k}^e - u_i^{e,\min} \} \} & \text{if } f_{i,k}^e \geq 0, \\ \max \{ f_{i,k}^e, 2d_{i,k}^e \max \{ u_i^{e,\min} - \bar{u}_{i,k}^e, \bar{u}_{i,k}^e - u_i^{e,\max} \} \} & \text{if } f_{i,k}^e < 0. \end{cases} \quad (28)$$

Note that this formula resembles (26). However, we have simplified (28) by exploiting (14). The semi-discrete scheme including limited DG fluxes is obtained by replacing $f_{i,k}^e$ in (20) with $f_{i,k}^{e,*}$. Again, direct calculation of the products $2d_{i,k}^e \bar{u}_{i,k}^e$ that appear in the definition of $f_{i,k}^{e,*}$ is preferable because denominator-free implementations are less sensitive to rounding errors. The addition of $f_{i,k}^{e,*}$ has the same effect as replacement of $\bar{u}_{i,k}^e$ with

$$\bar{u}_{i,k}^{e,*} = \bar{u}_{i,k}^e + \frac{f_{i,k}^{e,*}}{2d_{i,k}^e} \quad (29)$$

in the bar state form (13) of the semi-discrete scheme. We emphasize again that such representations are useful for analysis and verification purposes but the flux-corrected version of (20) may be employed in practice.

4.4. Sequential limiting for systems of conservation laws

The scalar case $m = 1$ is mostly covered by the explanations in the previous sections. Limiting for systems ($m > 1$), however, is more involved due to the coupling of multiple variables. Some conserved quantities can be expressed as products of a *specific variable* (a ratio of two conserved quantities) and a scalar conserved variable which we call the *main unknown* in what follows. For instance, momentum is the product of velocity and density in gas dynamics, whereas the main unknown of shallow water models is the water height. When it comes to limiting for such systems, it makes sense to impose bounds on the specific variables, rather than conserved products, i.e., limit velocity rather than momentum, in the above examples. This task can be accomplished by evolving the main variable first and constraining the products to satisfy local maximum principles for the specific variables in subsequent steps. Algorithms based on this approach are called *sequential limiters* [12, 26].

A sequential bar state limiter for piecewise (multi-)linear continuous Galerkin discretizations of hyperbolic systems was introduced in [36]. In this work, we extend it to the DG setting and high order approximations. We begin with the volumetric terms and use the generic notation ρ and $(\rho\phi)$ for the conserved quantities of system (1a). The main variable associated with $(\rho\phi)$ is ρ , whereas ϕ is the corresponding specific variable (amount of $\rho\phi$ per unit of ρ). The bar states for such derived quantities can be defined as [36]

$$\bar{\phi}_{ij}^e = \frac{\overline{(\rho\phi)}_{ij}^e + \overline{(\rho\phi)}_{ji}^e}{\bar{\rho}_{ij}^e + \bar{\rho}_{ji}^e} = \bar{\phi}_{ji}^e.$$

The previous section details the steps necessary to produce limited bar states $\bar{\rho}_{ij}^{e,*} = \bar{\rho}_{ij}^e + \frac{f_{\rho,ij}^{e,*}}{2d_{ij}^e}$ for the scalar main variable ρ at the first step of the sequential approach. To limit $(\rho\phi)$ in a manner consistent with the product rule of calculus, we split the antidiffusive fluxes $f_{\rho\phi,ij}^e$ of the unknown $(\rho\phi)$ into a flux that leaves $\bar{\phi}_{ij}^e$ unchanged and a remainder $g_{\rho\phi,ij}^e$ which requires additional limiting. This product rule splitting is defined by

$$\begin{aligned} f_{\rho\phi,ij}^{e,*} &= 2d_{ij}^e \left(\bar{\rho}_{ij}^{e,*} \bar{\phi}_{ij}^e - \overline{(\rho\phi)}_{ij}^e \right) + g_{\rho\phi,ij}^{e,*}, \\ g_{\rho\phi,ij}^e &= f_{\rho\phi,ij}^e - 2d_{ij}^e \left(\bar{\rho}_{ij}^{e,*} \bar{\phi}_{ij}^e - \overline{(\rho\phi)}_{ij}^e \right). \end{aligned} \quad (30)$$

The setting $g_{\rho\phi,ij}^{e,*} = g_{\rho\phi,ij}^e$ in (30) recovers the unlimited version of the scheme, while the setting $g_{\rho\phi,ij}^{e,*} = 0$ yields a low order approximation with the bound preserving bar states $\bar{\rho}_{ij}^{e,*} \bar{\phi}_{ij}^e$. As explained in [36], the bar states $\overline{(\rho\phi)}_{ij}^{e,*}$ of the flux-corrected high order scheme should be constrained to stay in the range $[\bar{\rho}_{ij}^{e,*} \phi_i^{e,\min}, \bar{\rho}_{ij}^{e,*} \phi_i^{e,\max}]$, where $\phi_i^{e,\min}, \phi_i^{e,\max}$ are suitably defined local bounds of numerical admissibility conditions. It follows that the limited fluxes $g_{\rho\phi,ij}^{e,*}$ should satisfy the inequality constraints

$$g_{\rho\phi,ij}^{e,\min} := 2d_{ij}^e \bar{\rho}_{ij}^{e,*} (\phi_i^{e,\min} - \bar{\phi}_{ij}^e) \leq g_{\rho\phi,ij}^{e,*} \leq 2d_{ij}^e \bar{\rho}_{ij}^{e,*} (\phi_i^{e,\max} - \bar{\phi}_{ij}^e) =: g_{\rho\phi,ij}^{e,\max}$$

and the equality constraint $g_{\rho\phi;ij}^{e,*} + g_{\rho\phi;ji}^{e,*} = 0$ which ensures conservation. These requirements can be met, similarly to (26),(28), by setting

$$g_{\rho\phi;ij}^{e,*} = \begin{cases} \min \{g_{\rho\phi;ij}^e, \min \{g_{\rho\phi;ij}^{e,\max}, -g_{\rho\phi;ji}^{e,\min}\}\} & \text{if } g_{\rho\phi;ij}^e \geq 0, \\ \max \{g_{\rho\phi;ij}^e, \max \{g_{\rho\phi;ij}^{e,\min}, -g_{\rho\phi;ji}^{e,\max}\}\} & \text{if } g_{\rho\phi;ij}^e < 0. \end{cases} \quad (31)$$

The fluxes $f_{\rho\phi;ij}^{e,*}$ are obtained by substituting (31) into (30) and the flux-corrected scheme is defined by replacing the specific components of f_i^e in (20) with the sums $f_{\rho\phi;i}^{e,*} = \sum_{j \in \tilde{\mathcal{N}}_i \setminus \{i\}} f_{\rho\phi;ij}^{e,*}$. The sequential limiter for interfacial DG fluxes operates in virtually the same fashion as its volumetric counterpart. After computing the limited bar states $\bar{\rho}_{i,k}^{e,*} = \bar{\rho}_{i,k}^e + \frac{f_{\rho;i,k}^e}{2d_{i,k}}$ for ρ and the bar states

$$\bar{\phi}_{i,k}^e = \bar{\phi}_{i',k'}^{e'} = \frac{(\overline{\rho\phi})_{i,k}^e + (\overline{\rho\phi})_{i',k'}^{e'}}{\bar{\rho}_{i,k}^e + \bar{\rho}_{i',k'}^{e'}} = \frac{(\overline{\rho\phi})_{i,k}^e}{\bar{\rho}_{i,k}^e} \quad (32)$$

for ϕ , the limited antidiffusive fluxes are calculated using the splitting

$$\begin{aligned} f_{\rho\phi;i,k}^{e,*} &= 2d_{i,k}^e \left(\bar{\rho}_{i,k}^{e,*} \bar{\phi}_{i,k}^e - (\overline{\rho\phi})_{i,k}^e \right) + g_{\rho\phi;i,k}^{e,*}, \\ g_{\rho\phi;i,k}^e &= f_{\rho\phi;i,k}^e - 2d_{i,k}^e \left(\bar{\rho}_{i,k}^{e,*} \bar{\phi}_{i,k}^e - (\overline{\rho\phi})_{i,k}^e \right). \end{aligned} \quad (33)$$

Again, the flux-corrected counterpart $\bar{\phi}_{i,k}^{e,*}$ of the interfacial bar state $\bar{\phi}_{i,k}^e$ stays in the range determined by suitable generic bounds $\phi_i^{e,\min}, \phi_i^{e,\max}$ if

$$g_{\rho\phi;i,k}^{e,\min} := 2d_{i,k}^e \bar{\rho}_{i,k}^{e,*} (\phi_i^{e,\min} - \bar{\phi}_{i,k}^e) \leq g_{\rho\phi;i,k}^{e,*} \leq 2d_{i,k}^e \bar{\rho}_{i,k}^{e,*} (\phi_i^{e,\max} - \bar{\phi}_{i,k}^e) =: g_{\rho\phi;i,k}^{e,\max}.$$

We enforce these interfacial flux constraints by setting

$$g_{\rho\phi;i,k}^{e,*} = \begin{cases} \min \{g_{\rho\phi;i,k}^e, \min \{g_{\rho\phi;i,k}^{e,\max}, -g_{\rho\phi;i,k}^{e,\min}\}\} & \text{if } g_{\rho\phi;i,k}^e \geq 0, \\ \max \{g_{\rho\phi;i,k}^e, \max \{g_{\rho\phi;i,k}^{e,\min}, -g_{\rho\phi;i,k}^{e,\max}\}\} & \text{if } g_{\rho\phi;i,k}^e < 0 \end{cases} \quad (34)$$

and replace the components of $f_{i,k}^e$ in (20) by the limited fluxes (33). Similarly to other limiters considered in this work, the sequential limiting of interfacial fluxes supports the possibility of a denominator-free implementation.

4.5. Definition of admissible bounds

So far, we have left unspecified how to choose the bounds for limiting in (26),(28),(31), and (34). Let us begin with the scalar case. The mean value theorem implies that the bar states of a scalar variable u satisfy

$$\min \{u_i^e, u_j^e\} \leq \bar{u}_{ij}^e \leq \max \{u_i^e, u_j^e\}, \quad \min \{u_i^e, u_{i'}^{e'}\} \leq \bar{u}_{i,k}^e \leq \max \{u_i^e, u_{i'}^{e'}\}.$$

The presented limiting strategy guarantees that the flux-corrected bar states $\bar{u}_{ij}^{e,*}$ and $\bar{u}_{i,k}^{e,*}$ stay in a range $[u_i^{e,\min}, u_i^{e,\max}]$ of admissible values which must contain \bar{u}_{ij}^e for all $j \in \tilde{\mathcal{N}}_i \setminus \{i\}$

and $\bar{u}_{i,k}^e$ for all $k \in \mathcal{F}_i$ by definition. To ensure that the same local bounds $u_i^{e,\min} = u_I^{\min}$ and $u_i^{e,\max} = u_I^{\max}$ can be used for the Bernstein coefficients of all nodes \mathbf{x}_i^e having the same physical location \mathbf{x}_I , we define

$$u_I^{\min} := \min_{(e,i): \mathbf{x}_i^e = \mathbf{x}_I} \min_{j \in \tilde{\mathcal{N}}_i} u_j^e, \quad u_I^{\max} := \max_{(e,i): \mathbf{x}_i^e = \mathbf{x}_I} \max_{j \in \tilde{\mathcal{N}}_i} u_j^e. \quad (35)$$

Note that (35) satisfies condition (27). In contrast to the DG approximation U_h , the Bernstein finite element interpolants U_h^{\min} and U_h^{\max} of the bounds defined by (35) are globally continuous. In our experience [27, 28], the use of continuous bounding functions in limiters for DG schemes leads to more reliable algorithms than discontinuous alternatives (as considered, for instance, in [5]). The compact stencil bounds defined (35) are generally more restrictive than the full stencil bounds that we used in [27, 28]. While the latter choice was found to produce good results in the context of limited DG schemes for the linear advection equation, its use for nonlinear problems may give rise to spurious oscillations within high order elements. To avoid such ripples, we construct the nodal bounds (35) by taking maxima/minima over the union of subcell stencils $\tilde{\mathcal{N}}_i$ rather than full element stencils (cf. also [39]).

For a scalar conservation law, the bar states \bar{u}_{ij}^e and $\bar{u}_{i,k}^e$ are bounded by pairs of Bernstein coefficients belonging to $\mathcal{G}_I = [u_I^{\min}, u_I^{\max}]$, where u_I^{\min} and u_I^{\max} are defined by (35). The main variable $u = \rho$ of a hyperbolic system can be limited as a scalar quantity but the mean value theorem is no longer applicable and, therefore, the bar states are not provably bounded by u_I^{\min} and u_I^{\max} . To rectify this, we include them in the definition

$$u_i^{e,\min} = \min \{ u_I^{\min}, \bar{u}_{ij}^e \}, \quad u_i^{e,\max} = \max \{ u_I^{\max}, \bar{u}_{ij}^e \} \quad \text{in (26)} \quad (36a)$$

$$u_i^{e,\min} = \min \{ u_I^{\min}, \bar{u}_{i,k}^e \}, \quad u_i^{e,\max} = \max \{ u_I^{\max}, \bar{u}_{i,k}^e \} \quad \text{in (28)} \quad (36b)$$

of the local bounds for the volumetric and interfacial terms, respectively. These bounds can be seen as a generalization of (35) to the system case.

An appropriate choice of bounds for specific variables ϕ of conserved products ($\rho\phi$) appears to be crucial to successful prevention of spurious ripples in sequential limiters for hyperbolic systems. We have observed violations of maximum principles if these bounds were chosen too wide. For volumetric terms, we obtained the best results using the maxima and minima of all states contributing to ϕ_i^e to define the nodal bounds (cf. [36])

$$\phi_I^{\min} = \min_{(e,i): \mathbf{x}_i^e = \mathbf{x}_I} \min \left\{ \frac{(\rho\phi)_i^e}{\rho_i^e}, \min \left\{ \min_{j \in \tilde{\mathcal{N}}_i} \bar{\phi}_{ij}^e, \min_{k \in \mathcal{F}_i} \bar{\phi}_{i,k}^e \right\} \right\}, \quad (37a)$$

$$\phi_i^{\max} = \max_{(e,i): \mathbf{x}_i^e = \mathbf{x}_I} \max \left\{ \frac{(\rho\phi)_i^e}{\rho_i^e}, \max \left\{ \max_{j \in \tilde{\mathcal{N}}_i} \bar{\phi}_{ij}^e, \max_{k \in \mathcal{F}_i} \bar{\phi}_{i,k}^e \right\} \right\}. \quad (37b)$$

In contrast to (35), we are not using any nodal states other than $\frac{(\rho\phi)_i^e}{\rho_i^e}$ here but all low order bar states associated with the given node are included. The corresponding bounding functions ϕ_h^{\min} and ϕ_h^{\max} are continuous again.

The different nature of volumetric terms and DG fluxes may be taken into account when it comes to constructing the property-preserving bounds $\phi_k^{e,\min}$ and $\phi_k^{e,\max}$ for the interfacial flux limiter (34). Sampling the auxiliary bar states (32) of nodes \mathbf{x}_i^e belonging to the face component Γ_k^e , we define

$$\phi_k^{e,\min} = \min_{\mathbf{x}_i^e \in \Gamma_k^e} \bar{\phi}_{i,k}^e, \quad \phi_k^{e,\max} = \max_{\mathbf{x}_i^e \in \Gamma_k^e} \bar{\phi}_{i,k}^e. \quad (38)$$

Although this definition is inconsistent with (37), it is motivated by the fact that the high-order flux $\mathcal{H}\left(U_h^e, \hat{U}_h^e; \mathbf{n}^e\right)$ across Γ_k^e depends only on the Bernstein coefficients of nodes that are physically located on Γ_k^e . To preserve this domain of dependence in the process of flux correction, we construct the bounds (38) of the limiting formula (34) individually for each face.

4.6. Summary of the limiting procedure

We end the description of our method with a brief summary of the limiting procedure. The semi-discrete bound preserving formulation reads

$$m_i^e \frac{du_i^e}{dt} = \sum_{j \in \tilde{\mathcal{N}}_i} [d_{ij}^e (u_j^e - u_i^e) - (\mathbf{F}_j^e - \mathbf{F}_i^e) \tilde{\mathbf{c}}_{ij}^e] + \sum_{j \in \tilde{\mathcal{N}}_i \setminus \{i\}} f_{ij}^{e,*} + \sum_{k \in \mathcal{F}_i} \left(\frac{1}{2} \int_{\Gamma_k^e} \varphi_i^e ds \left[(\mathbf{F}_i^e - \hat{\mathbf{F}}_i^e) \mathbf{n}^e + \lambda_i^e (\hat{u}_i^e - u_i^e) \right] + f_{i,k}^{e,*} \right). \quad (39)$$

Note that by omitting $f_{ij}^{e,*}$, $f_{i,k}^{e,*}$ in (39) altogether, we reproduce the low order scheme (11). For the scalar case, $f_{ij}^{e,*}$ and $f_{i,k}^{e,*}$ are defined by (26) and (28), respectively. The bounds for both limiters are defined by (35).

For the system case, the fluxes for main unknowns are also limited using (26) and (28) with bounds (36). Products of the main variables and ratios of conserved quantities are constrained using the limited fluxes (30), (33) based on (31), (34) with respective bounds given by (37), (38).

5. Numerical results

Having completed the description of our methodology, we now present results for classical benchmarks of various conservation laws. We begin with tests for scalar linear and nonlinear transport models. Further numerical examples demonstrate the feasibility of applying our method to the Euler and shallow water equations. The implementation of all algorithms presented in this article is based on the open source C++ library MFEM [1, 6].

For piecewise polynomial approximations of degree $p \in \mathbb{N}_0$, we abbreviate the methods under investigation using the suffix $\mathbb{V}_p \in \{\mathbb{P}_p, \mathbb{Q}_p\}$ to indicate whether simplicial or box element meshes are used. In our numerical study, we compare the results for the following spatial discretizations:

- DG- \mathbb{V}_p : Standard DG scheme, as defined by (8),
- LO- \mathbb{V}_p : Low order method, as defined by (11),
- MCL- \mathbb{V}_p : Monolithic convex limiting, as defined by (39).

In this section, we occasionally compare piecewise constant approximations to limited solutions. In such instances, $p = 0$ always refers to DG- \mathbb{V}_0 approximations (rather than LO or MCL) because they do not require limiting. As in [27, 28], we often compare results for approximations of increasingly high order on a sequence of successively coarsened meshes. The number of mesh elements $E = E(p)$ for each simulation is fitted to the number of nodes per element $N = N(p)$ in such a way that the total number of degrees of freedom per variable ($\#\text{DOF} = EN$) remains constant. For instance, the finite element spaces \mathbb{Q}_2 and \mathbb{Q}_5 have the same dimensions if the mesh for \mathbb{Q}_2 corresponds to one level of uniform refinement of the mesh for \mathbb{Q}_5 .

All numerical solutions are evolved or marched to a steady state using explicit SSP Runge-Kutta methods [16]. In all time-dependent benchmarks, we employ the third order explicit time integrator with three stages, whereas forward Euler time stepping is sufficient for problems with steady state solutions.

5.1. Linear transport in 1D

Advective transport of a density field $u : \Omega \times \mathbb{R}_+ \rightarrow \mathbb{R}$ in a 1D domain $\Omega \subseteq \mathbb{R}$ with unit velocity is described by the conservation law

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0 \quad \text{in } \Omega \times \mathbb{R}_+.$$

The one-dimensional unit normal \mathbf{n} has as single component $n_x \in \{-1, 1\}$ and the numerical flux (6) simplifies to the upwind-sided trace

$$\mathcal{H}(u, v, \mathbf{n}) = \frac{1}{2} ((u + v)n_x + |n_x|(u - v)) = \begin{cases} u & \text{if } n_x = 1, \\ v & \text{if } n_x = -1. \end{cases}$$

We test the ability of the proposed methods to capture discontinuous and smooth initial profiles without introducing spurious oscillations. To this end, we equip $\Omega = (0, 1)$ with periodic boundary conditions. Hence the analytical solution at any time $t \in \mathbb{N}$ coincides with the initial condition u_0 . In this setting, the invariant set is $\mathcal{G} = [\min_{\bar{\Omega}} u_0, \max_{\bar{\Omega}} u_0]$. The initial profile

$$u_0(x) = \begin{cases} 1 & \text{if } 0.2 \leq x \leq 0.4, \\ \exp(10) \exp\left(\frac{1}{0.5-x}\right) \exp\left(\frac{1}{x-0.9}\right) & \text{if } 0.5 < x < 0.9, \\ 0 & \text{otherwise} \end{cases} \quad (40)$$

features a discontinuous part and an infinitely differentiable part. We begin by advecting this profile up to the final time $t = 1$.

For these simulations, we use time step $\Delta t = 10^{-3}$, $\#\text{DOF} = 192$ and consider orders $p \in \{0, 2, 5, 11, 23\}$. The results for DG, LO and MCL are displayed in Fig. 1. As expected, the standard DG method produces spurious ripples close to the discontinuities. The LO scheme yields very inaccurate approximations which are even slightly more diffusive than the DG- \mathbb{Q}_0 result employing the same $\#\text{DOF}$. However, it is remarkable that the accuracy of LO solutions does not deteriorate significantly as p is increased while the mesh is coarsened. This property of the method is due to the use of preconditioned gradient matrix \tilde{C}_1^e in definition (10) of the artificial viscosity coefficients.

The accuracy of numerical solutions obtained with the subcell flux limiting approach is also very similar for different orders and, in contrast to the pure DG results, the presented MCL approximations satisfy maximum principles. This behavior indicates optimality of the method for a given number of $\#\text{DOF}$ (cf. [27]). The slight peak clipping effects at the top of the exponential hill can be avoided by using smoothness indicators [28, 43].

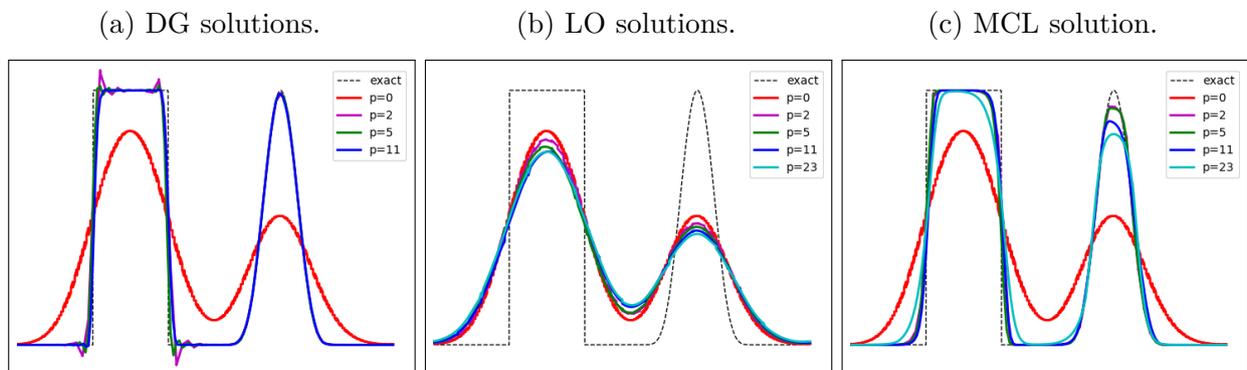


Figure 1: Numerical approximations to the 1D advection equation with initial condition (40) obtained using $p \in \{0, 2, 5, 11, 23\}$.

Next, we impose periodic boundary conditions at the endpoints of the domain $\Omega = (-1, 1)$ and advect the smooth initial profile

$$u_0(x) = \exp(-25x^2) \quad (41)$$

up to the time instant $t = 2$, at which we compute the $L^1(\Omega)$ errors and corresponding estimated orders of convergence (EOC) for DG, LO and MCL approximations using $p \in \{1, \dots, 4\}$. The time step size $\Delta t = 10^{-4}$ is chosen to be small enough for the time discretization errors to be negligible compared to the spatial discretization error on the finest mesh. The results presented in Tables 1 to 3 confirm optimal rates of convergence. DG approximations converge with order $p + 1$, the low order method converges at least with order $\frac{1}{2}$, and limited solutions are at least second order accurate. Higher than second order convergence rates cannot be expected for limiters that preserve the local extrema of the old solution [15, 49]. Thus the local bounds of the inequality constraints for the antidiffusive fluxes need to be relaxed in smooth regions [13, 23, 28, 44, 46]. For that purpose, we plan to equip our algorithm with suitably designed smoothness indicators in the future.

$1/h$	$p = 1$	EOC	$p = 2$	EOC	$p = 3$	EOC	$p = 4$	EOC
24	1.27E-02		3.21E-04		7.38E-06		4.11E-07	
32	6.43E-03	2.36	8.28E-05	4.71	2.13E-06	4.32	9.84E-08	4.97
48	2.26E-03	2.58	1.53E-05	4.16	4.17E-07	4.02	1.27E-08	5.05
64	1.01E-03	2.79	5.74E-06	3.41	1.32E-07	4.01	3.09E-09	4.90
96	3.12E-04	2.91	1.62E-06	3.11	2.61E-08	3.99		
128	1.34E-04	2.92	6.86E-07	3.00				
192	4.17E-05	2.89						

Table 1: The $\|\cdot\|_{L^1(\Omega)}$ errors and corresponding EOC of DG- \mathbb{Q}_p , $p \in \{1, \dots, 4\}$ solutions to the 1D advection equation with initial condition (41).

$1/h$	$p = 1$	EOC	$p = 2$	EOC	$p = 3$	EOC	$p = 4$	EOC
24	9.43E-02		8.11E-02		6.73E-02		6.02E-02	
32	7.93E-02	0.60	6.73E-02	0.65	5.51E-02	0.70	4.89E-02	0.72
48	6.05E-02	0.67	5.05E-02	0.71	4.05E-02	0.76	3.56E-02	0.78
64	4.92E-02	0.72	4.05E-02	0.77	3.21E-02	0.81	2.81E-02	0.83
96	3.58E-02	0.78	2.91E-02	0.82	2.27E-02	0.85		
128	2.82E-02	0.83	2.27E-02	0.86				
192	1.98E-02	0.87						

Table 2: The $\|\cdot\|_{L^1(\Omega)}$ errors and corresponding EOC of LO- \mathbb{Q}_p , $p \in \{1, \dots, 4\}$ solutions to the 1D advection equation with initial condition (41).

$1/h$	$p = 1$	EOC	$p = 2$	EOC	$p = 3$	EOC	$p = 4$	EOC
24	1.04E-02		2.52E-03		1.27E-03		5.51E-04	
32	5.69E-03	2.11	1.36E-03	2.14	6.60E-04	2.26	2.79E-04	2.37
48	2.36E-03	2.17	5.46E-04	2.26	2.59E-04	2.31	1.07E-04	2.38
64	1.27E-03	2.17	2.82E-04	2.30	1.32E-04	2.35	5.53E-05	2.28
96	5.08E-04	2.25	1.08E-04	2.36	4.98E-05	2.40		
128	2.59E-04	2.34	5.58E-05	2.31				
192	1.01E-04	2.33						

Table 3: The $\|\cdot\|_{L^1(\Omega)}$ errors and corresponding EOC of MCL- \mathbb{Q}_p , $p \in \{1, \dots, 4\}$ solutions to the 1D advection equation with initial condition (41).

5.2. Burgers equation

In the next test of this section, we apply the methods under investigation to a nonlinear conservation law for a scalar quantity u evolving in a domain $\Omega \subset \mathbb{R}^d$. Using the constant vector $\mathbf{v} = (1, \dots, 1)^T \in \mathbb{R}^d$ to isotropically extend the flux $f(u) = \frac{u^2}{2}$ into multidimensions (for $d = 2, 3$), we consider the generalized inviscid Burgers equation

$$\frac{\partial u}{\partial t} + \nabla \cdot \left(\frac{\mathbf{v}u^2}{2} \right) = 0 \quad \text{in } \Omega \times \mathbb{R}_+.$$

The flux function $\mathbf{F}(u) = \frac{vu^2}{2}$ is convex and, therefore, the maximal wave speed (7) is given by $\lambda(u, v, \mathbf{n}^e) = \max\{u, v\} \left| \sum_{l=1}^d (\mathbf{n}^e)_l \right|$.

5.2.1. One dimensional benchmark

Once again, we consider the one dimensional case and equip $\Omega = (0, 1)$ with periodic boundary conditions. The initial data [38]

$$u_0(x) = \sin(2\pi x) \quad (42)$$

remains smooth up to the time $t = \frac{1}{2\pi}$ of shock formation at $x = 0.5$. For smaller times, the classical solution can be determined using Newton's method to solve the implicit equation $u(x, t) = \sin(2\pi(x - u(x, t)t))$.

We first run DG, LO and MCL up to $t = 0.1$ using $\Delta t = 4 \cdot 10^{-4}$ and $p \in \{1, \dots, 4\}$. The $\|\cdot\|_{L^1(\Omega)}$ errors and convergence rates for the three methods are presented in Tables 4 to 6. Our results confirm that the optimal convergence behavior, as observed in the previous section, carries over to the nonlinear case. In fact, the low order method becomes first order accurate, which is an improvement compared to Table 2.

$1/h$	$p = 1$	EOC	$p = 2$	EOC	$p = 3$	EOC	$p = 4$	EOC
48	7.45E-04		1.60E-05		7.43E-07		4.96E-08	
64	4.31E-04	1.90	7.23E-06	2.77	2.87E-07	3.30	1.14E-08	5.11
96	1.98E-04	1.92	2.42E-06	2.70	6.69E-08	3.59	1.66E-09	4.76
128	1.13E-04	1.94	1.09E-06	2.77	2.28E-08	3.73	4.59E-10	4.46
192	5.15E-05	1.95	3.47E-07	2.82	4.89E-09	3.80		
256	2.93E-05	1.96	1.53E-07	2.86				
384	1.32E-05	1.97						

Table 4: The $\|\cdot\|_{L^1(\Omega)}$ errors and corresponding EOC of DG- \mathbb{Q}_p , $p \in \{1, \dots, 4\}$ solutions to the 1D Burgers equation with initial condition (42).

$1/h$	$p = 1$	EOC	$p = 2$	EOC	$p = 3$	EOC	$p = 4$	EOC
48	1.62E-02		1.25E-02		9.10E-03		7.69E-03	
64	1.23E-02	0.93	9.48E-03	0.96	6.95E-03	0.94	5.80E-03	0.98
96	8.39E-03	0.95	6.41E-03	0.96	4.68E-03	0.98	3.90E-03	0.98
128	6.37E-03	0.96	4.83E-03	0.98	3.52E-03	0.99	2.94E-03	0.99
192	4.30E-03	0.97	3.25E-03	0.98	2.36E-03	0.99		
256	3.24E-03	0.98	2.45E-03	0.99				
384	2.17E-03	0.99						

Table 5: The $\|\cdot\|_{L^1(\Omega)}$ errors and corresponding EOC of LO- \mathbb{Q}_p , $p \in \{1, \dots, 4\}$ solutions to the 1D Burgers equation with initial condition (42).

We proceed by computing approximations to this problem at final time $t = 0.2$ after the shock has developed. We use $\Delta t = 2.5 \cdot 10^{-3}$, $p \in \{0, 1, 3, 7, 15, 31\}$ and keep $\#\text{DOF} = 96$

$1/h$	$p = 1$	EOC	$p = 2$	EOC	$p = 3$	EOC	$p = 4$	EOC
48	1.29E-03		2.03E-04		9.54E-05		4.87E-05	
64	7.68E-04	1.80	9.98E-05	2.47	4.85E-05	2.36	2.46E-05	2.38
96	3.44E-04	1.98	4.05E-05	2.22	1.95E-05	2.25	9.91E-06	2.24
128	1.94E-04	1.98	2.24E-05	2.06	1.09E-05	2.02	5.07E-06	2.33
192	8.41E-05	2.07	9.23E-06	2.19	4.26E-06	2.32		
256	4.69E-05	2.03	4.74E-06	2.32				
384	2.04E-05	2.05						

Table 6: The $\|\cdot\|_{L^1(\Omega)}$ errors and corresponding EOC of MCL- \mathbb{Q}_p , $p \in \{1, \dots, 4\}$ solutions to the 1D Burgers equation with initial condition (42).

constant for all simulations. The results of LO and MCL approximations are shown in Fig. 2. Before the shock formation time, all approximations are almost indistinguishable, while at time $t = 0.2$ only the highest order approximation is considerably more diffusive than other curves in diagrams depicting the LO and MCL results. This behavior can be attributed to the fact that for $p = 31$ the shock is located inside an element rather than at element boundaries, as for all other approximations. Given the fact that MCL- \mathbb{Q}_{31} uses just three elements, it is remarkable that the result still looks quite reasonable. Around the common boundary of the second and third element from the left, the MCL- \mathbb{Q}_{31} solution exhibits stronger deviations from the other curves but no violations of monotonicity are observed. For this simple example, LO is almost as accurate as MCL with only minor peak clipping effects close to the shock, which is in accordance with the convergence rates presented in Table 5.

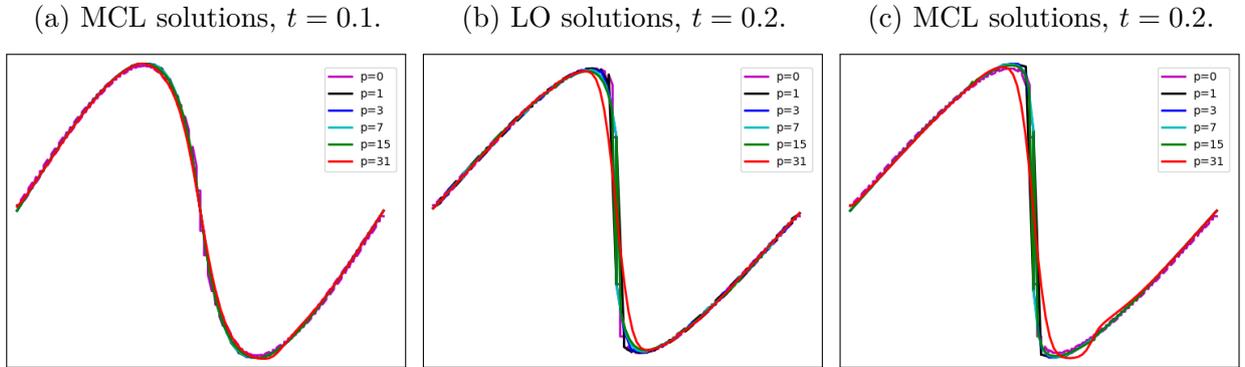


Figure 2: Snapshots of bound preserving $p \in \{0, 1, 3, 7, 15, 31\}$ approximations to the 1D Burgers equation with initial condition (42) at time instants (a) before and (b),(c) after the shock development.

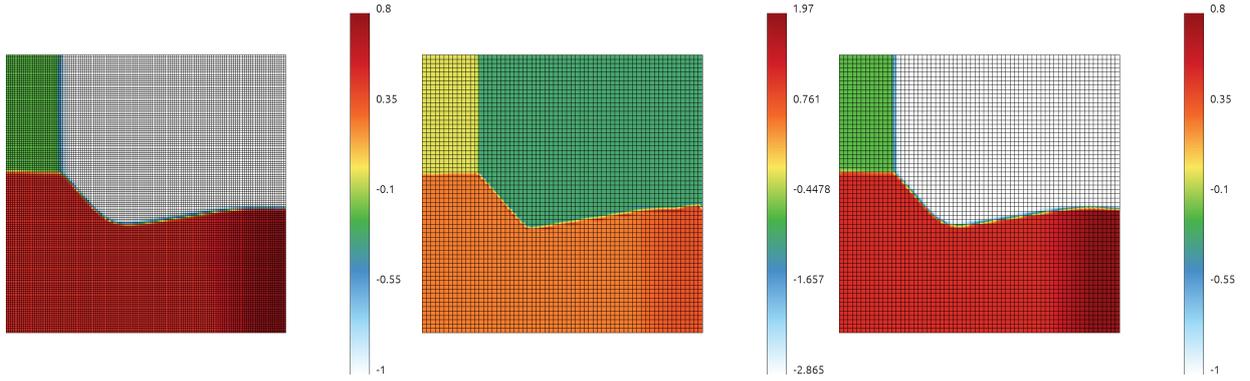
5.2.2. Two dimensional benchmark

For the 2D Burgers test, we use $\Omega = (0, 1)^2$ and the initial condition

$$u_0(x, y) = \begin{cases} -0.2 & \text{if } x < 0.5 \wedge y > 0.5, \\ -1 & \text{if } x > 0.5 \wedge y > 0.5, \\ 0.5 & \text{if } x < 0.5 \wedge y < 0.5, \\ 0.8 & \text{if } x > 0.5 \wedge y < 0.5. \end{cases} \quad (43)$$

The boundary type at a point $(x, y) \in \partial\Omega$ depends on the sign of u which is changing in time. At the time dependent inlet, we prescribe the analytical solution which can be found in [17]. The invariant set of this benchmark is $\mathcal{G} = [-1, 0.8]$. At the final time $t = 0.5$, we compare DG- \mathbb{Q}_p results for $p \in \{0, 1\}$ to MCL- \mathbb{Q}_p approximations using $p \in \{1, 3, 7, 15\}$. The snapshots presented in Fig. 3 were obtained using $\#\text{DOF} = 128^2$ and time step $\Delta t = 10^{-3}$ in all cases.

(a) DG- \mathbb{Q}_0 , $e_1(0.5) = 1.49\text{E-}2$ (b) DG- \mathbb{Q}_1 , $e_1(0.5) = 9.67\text{E-}3$ (c) MCL- \mathbb{Q}_1 , $e_1(0.5) = 1.09\text{E-}2$



(d) MCL- \mathbb{Q}_3 , $e_1(0.5) = 1.28\text{E-}2$ (e) MCL- \mathbb{Q}_7 , $e_1(0.5) = 1.73\text{E-}2$ (f) MCL- \mathbb{Q}_{15} , $e_1(0.5) = 2.88\text{E-}2$

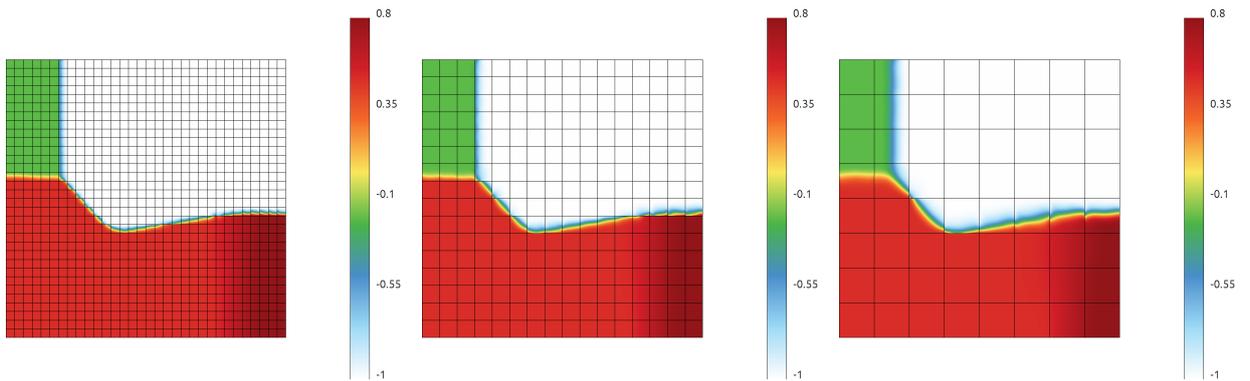


Figure 3: DG and MCL solutions of the 2D Burgers equation with initial condition (43) and corresponding $e_1(t) := \|u(t) - u_h(t)\|_{L^1(\Omega)}$ errors for $p \in \{0, 1, 3, 7, 15\}$ at $t = 0.5$.

As expected, the DG- \mathbb{Q}_0 result is quite diffusive. The DG- \mathbb{Q}_1 solution exhibits severe over- and undershoots close to the shocks, while all MCL approximations are nonoscillatory. The

L^1 error of the MCL- \mathbb{Q}_1 result is just marginally larger than that of the DG- \mathbb{Q}_1 solution. A decline in the accuracy of MCL- \mathbb{Q}_p is observed for higher orders on coarser meshes. For this shock-dominated problem, the MCL- \mathbb{Q}_1 solution is optimal because it is monotonicity preserving and has the smallest error for a fixed #DOF. The MCL- \mathbb{Q}_3 solution is slightly more accurate than the DG- \mathbb{Q}_0 result but the L^1 error of the latter approximation is smaller than that of MCL- \mathbb{Q}_p for $p \in \{7, 15\}$. It is still remarkable how well the MCL- \mathbb{Q}_{15} approximation resolves the analytical solution, even though none of the shocks is aligned with mesh edges.

5.3. Euler equations of gas dynamics

An important example of (1a) are the Euler equations of gas dynamics. The solution vector and flux matrix of this hyperbolic system are given by

$$U = \begin{bmatrix} \varrho \\ \varrho \mathbf{v} \\ \varrho E \end{bmatrix} \in \mathbb{R}^{d+2}, \quad \mathbf{F}(U) = \begin{bmatrix} \varrho \mathbf{v} \\ \varrho \mathbf{v} \otimes \mathbf{v} + p \mathbf{I} \\ (\varrho E + p) \mathbf{v} \end{bmatrix} \in \mathbb{R}^{(d+2) \times d},$$

where ϱ , \mathbf{v} , E denote density, velocity and specific total energy, respectively, $\mathbf{I} \in \mathbb{R}^{d \times d}$ is the identity matrix and p is the pressure satisfying the equation of state for an ideal polytropic gas

$$p = (\gamma - 1) \left(\varrho E - \frac{\varrho |\mathbf{v}|^2}{2} \right) = (\gamma - 1) \varrho e.$$

Here ϱe is the internal energy and $\gamma > 1$ denotes the heat capacity ratio, which is set to 1.4 in all numerical examples considered in this work.

An upper bound for the fastest wave speed has been derived in [19]. An invariant set of the Euler system is given by [23]

$$\mathcal{G} = \{ (\varrho, \varrho \mathbf{v}, \varrho E)^T \in \mathbb{R}^{d+2} : \varrho > 0, e > 0, s \geq s_{\min} \},$$

where $s = \log \left(e^{\frac{1}{\gamma-1}} \varrho^{-1} \right)$ is the specific entropy which should be bounded below by an arbitrary constant $s_{\min} > 0$.

5.3.1. Sod Shock tube

The classical shock tube example by Sod [45] serves as our first test for solving the Euler equations. The 1D domain $\Omega = (0, 1)$ is equipped with wall boundaries at $x \in \{0, 1\}$ and the initial condition is specified as

$$U_0(x) = \begin{cases} (1, 0, 2.5)^T & \text{if } x < 0.5, \\ (0.125, 0, 0.25)^T & \text{if } x > 0.5. \end{cases}$$

The discontinuity of the initial condition at $x = 0.5$ gives rise to a shock, rarefaction wave, and contact discontinuity, which a numerical approximation should resolve reasonably well. We evolve the LO and MCL solutions up to the final time $t = 0.231$ using $\Delta t = 4 \cdot 10^{-4}$, #DOF = 256, and $p \in \{1, 3, 7, 15, 31\}$. The density and pressure profiles obtained with these

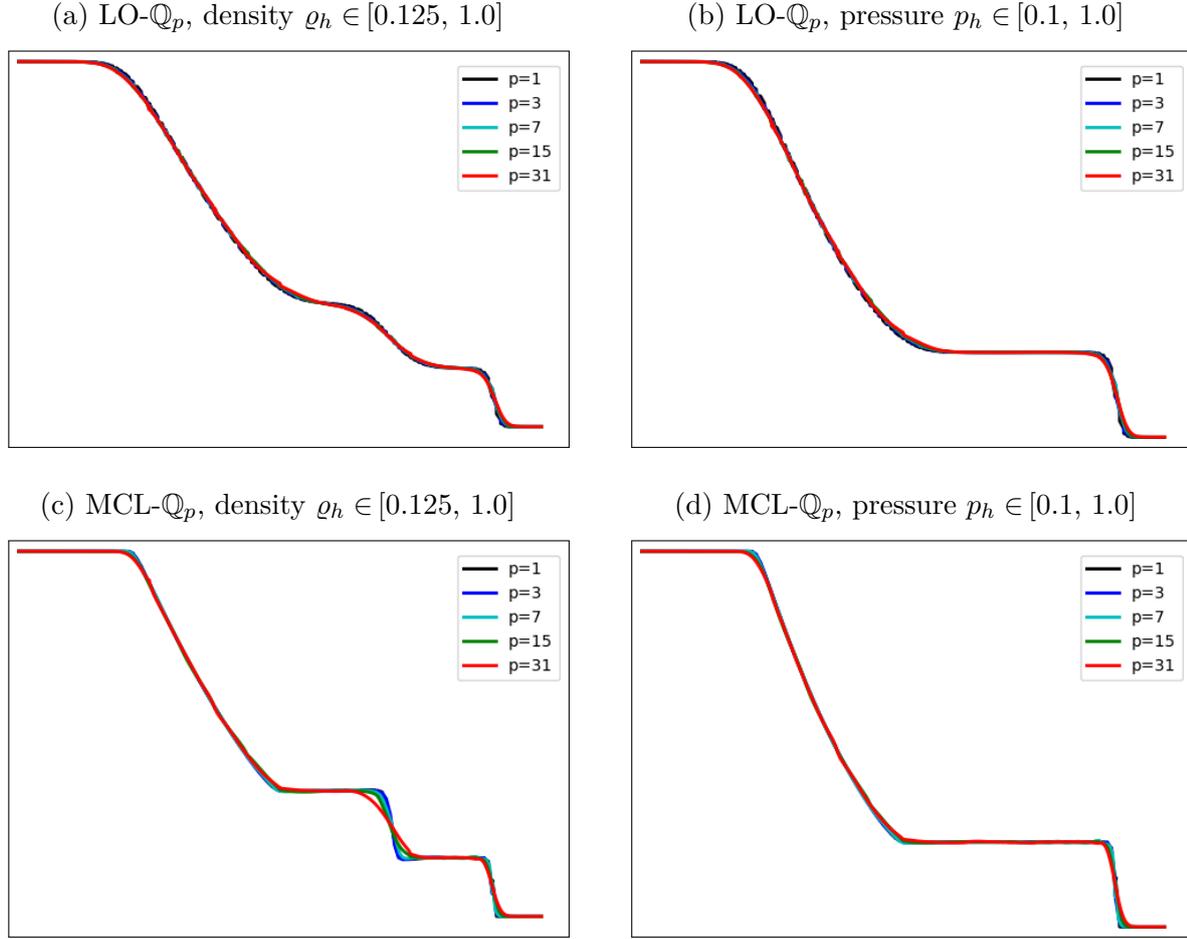


Figure 4: Sod shock tube benchmark for the Euler equations. LO and MCL approximations at time $t = 0.231$ obtained using $p \in \{1, 3, 7, 15, 31\}$.

parameter settings are shown in Fig. 4. They exhibit good qualitative agreement with each other and are all nonoscillatory. Again, this is a Riemann problem with a discontinuous exact solution. Therefore, it is quite remarkable that the solutions obtained with large p on coarse meshes are almost as well resolved as the smaller- p -same#DOF approximations. Note that the MCL- \mathbb{Q}_1 solution is obtained on a mesh with 128 elements, while MCL- \mathbb{Q}_{31} uses only 8 elements. The monolithic limiter employed in this study represents the first high order extension of the sequential limiting procedure proposed in [36] to the system case. So far, only [44] presents algebraically limited high order DG methods for systems, and the limiting strategy adopted therein is not monolithic.

A preliminary conclusion, which has been shown to be true for MCL in the scalar case, is as follows. For shock-dominated problems and fixed #DOF, the most accurate MCL results are obtained with low polynomial degrees on fine meshes. This is not surprising since the potential benefit of using higher order approximations lies in the possibility of achieving improved convergence rates under certain smoothness assumptions. If these assumptions

do not hold, it makes more sense to refine the mesh than to increase the order. For smooth solutions, the opposite may be the case. However, to circumvent the second order accuracy barrier for local extremum diminishing approximations, the MCL scheme must be equipped with a smoothness indicator.

5.3.2. Double Mach reflection

A more difficult benchmark for solving the Euler equations is double Mach reflection [47]. The boundary of the computational domain $\Omega = (0, 4) \times (0, 1)$ consists of a reflecting wall $\Gamma_w = \{(x, 0)^T \in \partial\Omega : 1/6 \leq x \leq 4\}$, a supersonic outlet $\Gamma_o = \{(4, y)^T : 0 < y \leq 1\}$, and a supersonic inlet at $\partial\Omega \setminus (\Gamma_w \cup \Gamma_o)$. Post- and pre-shock states defined by

$$U_L = \begin{bmatrix} 8 \\ 66 \cos(30^\circ) \\ -66 \sin(30^\circ) \\ 563.5 \end{bmatrix}, \quad U_R = \begin{bmatrix} 1.4 \\ 0 \\ 0 \\ 2.5 \end{bmatrix},$$

are used as initial and inflow boundary conditions in the following manner

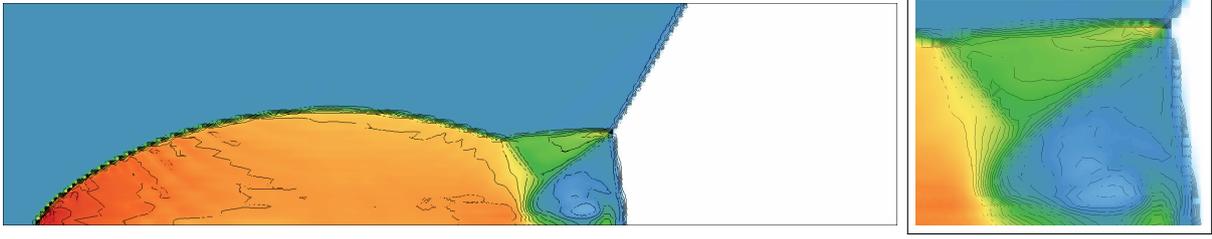
$$U_0(x, y) = \begin{cases} U_L & \text{if } x < \frac{1}{6} + \frac{y}{\sqrt{3}}, \\ U_R & \text{otherwise,} \end{cases} \quad U_{\text{in}}(x, y, t) = \begin{cases} U_L & \text{if } x < \frac{1}{6} + \frac{y+20t}{\sqrt{3}}, \\ U_R & \text{otherwise.} \end{cases}$$

The angle between the initial shock and wall boundary is 60° . The shock-wall interaction results in a propagating Mach 10 shock. To assess the performance of numerical schemes for this benchmark, one tries to resolve the triple point visible in the right part of Ω at time $t = 0.2$ as accurately as possible.

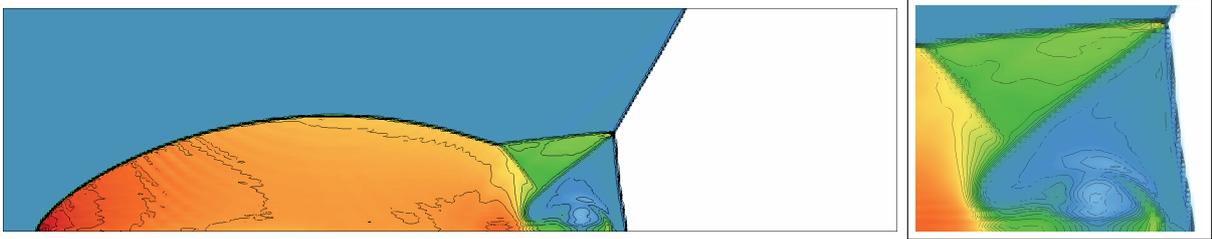
We have already established that the most accurate approximations for fixed #DOF are obtained with piecewise linear polynomials. To study the impact of #DOF on the accuracy of our scheme, we therefore compare the MCL- \mathbb{Q}_1 solutions for this benchmark on a series of refined uniform meshes. On the coarsest level, we use #DOF = $4 \cdot 96^2$, $\Delta t = 5 \cdot 10^{-5}$. The mesh and time step sizes are halved at each refinement level.

The snapshots in Fig. 5 confirm the conjecture that the MCL limiter is a powerful stabilization technique not only on coarse meshes but also for higher resolution. For instance, only the finest mesh simulation can capture the small scale eddies in the zoomed region. On coarser meshes, even the shocks are smeared significantly. The application of MCL to the DG- \mathbb{Q}_1 target discretization seems to introduce enough numerical dissipation in this example. Therefore, we were able to run these simulations without the need to deal with negative pressures and perform entropy fixes. Admittedly, solving this problem with higher order methods does produce negative pressures. As explained in [36], positivity preservation can be readily enforced via additional limiting. This feature makes it possible to solve more challenging problems, so we plan to utilize it in the future. We expect that running MCL in a safe positivity-preserving mode will enable us to achieve better resolution of the small scale eddies in Fig. 5d using higher order spaces on coarser meshes.

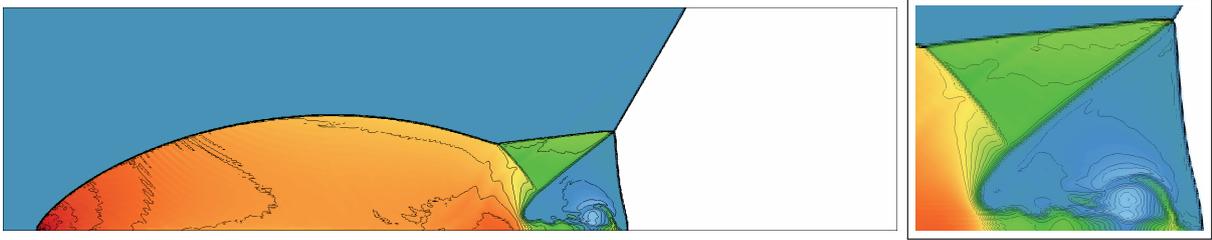
(a) Post-shock density distribution on a mesh of $4 \cdot 48^2$ elements obtained with $\Delta t = 5E-5$.



(b) Post-shock density distribution on a mesh of $4 \cdot 96^2$ elements obtained with $\Delta t = 2.5E-5$.



(c) Post-shock density distribution on a mesh of $4 \cdot 192^2$ elements obtained with $\Delta t = 1.25E-5$.



(d) Post-shock density distribution on a mesh of $4 \cdot 384^2$ elements obtained with $\Delta t = 6.25E-6$.

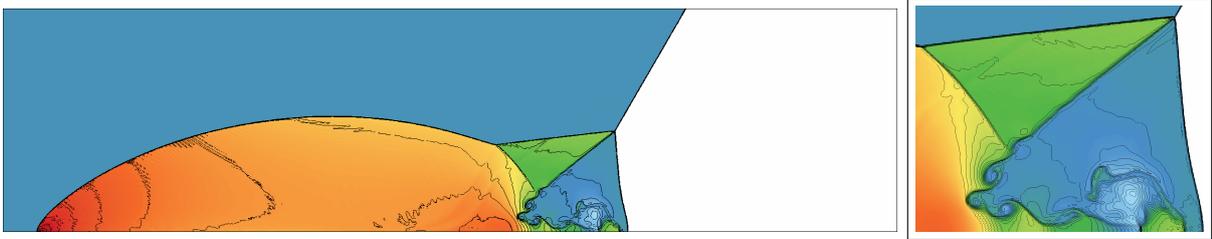


Figure 5: Influence of the mesh size and time step on the accuracy of the MCL- Q_1 scheme for the Euler equations, as illustrated by numerical solutions of the double Mach reflection problem at time $t = 0.2$ with zooms of the triple point region.

5.4. Shallow water equations

Finally, we consider the system of shallow water equations

$$u = \begin{bmatrix} H \\ H\mathbf{v} \end{bmatrix} \in \mathbb{R}^{d+1}, \quad \mathbf{F}(u) = \begin{bmatrix} H\mathbf{v} \\ H\mathbf{v} \otimes \mathbf{v} + \frac{g}{2}H^2\mathbf{I} \end{bmatrix} \in \mathbb{R}^{(d+1) \times d},$$

where H denotes the total water height, \mathbf{v} is the depth-averaged velocity, and g is the acceleration due to gravity. Source terms such as bathymetry gradient and friction terms are omitted in order to have a system of the form (1a). An invariant set of the initial value problem for the shallow water equations is

$$\mathcal{G} = \{(H, H\mathbf{v})^T \in \mathbb{R}^{d+1} : H > 0\},$$

and an upper bound for the fastest wave speed can be found in [22].

5.4.1. Radial dam break problem

This Riemann problem for the shallow water equations in 2D resembles the shock tube example in Section 5.3.1. The whole boundary of the domain $\Omega = (-1, 1)^2$ is treated as an outlet (although reflecting wall boundary conditions could also be prescribed). We choose the initial condition

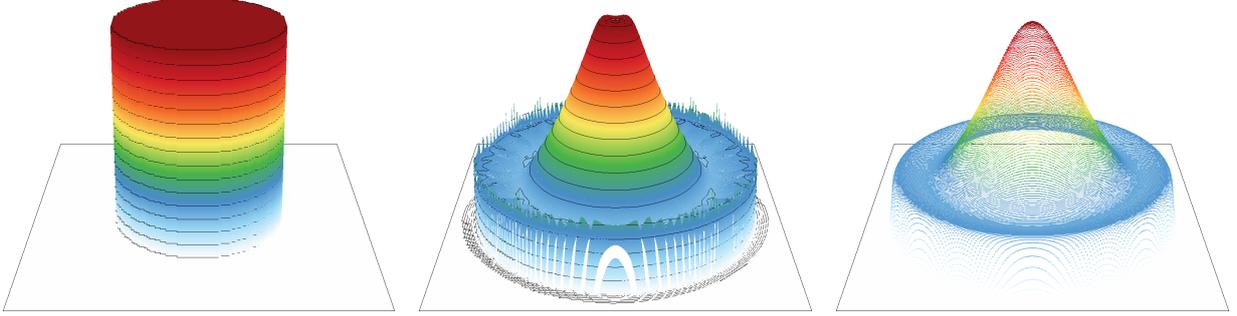
$$U_0(\mathbf{x}) = (H_0, (Hu)_0, (Hv)_0)^T = \begin{cases} (1, 0, 0)^T & \text{if } |x| \leq 0.5, \\ (0.1, 0, 0)^T & \text{if } |x| > 0.5, \end{cases} \quad (44)$$

set $g = 9.81$ and solve the radially symmetric Riemann problem up to the time $t = 0.06$. The radial discontinuity of the initial data produces a rarefaction wave which travels towards the domain center, and a radially symmetric shock which propagates in the outward direction. In contrast to the system of Euler equations, no contact discontinuities appear in the solution for the shallow water equations.

We approximate the solution to this problem on uniform quadrilateral meshes using #DOF = 256², $p \in \{0, 1, 3, 7\}$ and set $\Delta t = 10^{-4}$ in all tests. The discrete initial heights $H_h(\cdot, 0)$ are obtained by evaluating (44) in the Bernstein nodes and using the pointwise values as Bernstein coefficients. While this approach is not a projection in the mathematical sense (i.e., $P^2 = P$ is violated for $p > 1$) and provides only second order accuracy, it is sufficient for this example and has the advantage that it does not introduce violations of maximum principles, as opposed to consistent L^2 projections.

All of the snapshots presented in Fig. 6 reproduce the propagation of rarefaction and shock waves correctly. However, for a marginally different setup, the DG-Q₁ approximation blows up because the oscillations visible along the shock front can cause the algorithm to produce negative water heights and terminate. Such violations of the IDP property are also visible at the peak of the rarefaction wave but these are clearly not as severe. The DG-Q₀ solution yields a rather poor approximation again, while the MCL results combine the advantages of bound preserving and sufficiently accurate approximations. The ripples observed in the high order approximations Figs. 6e and 6f are not overshoots but rather remnants of discontinuities on meshes coarser than the one employed in Fig. 6d. Still, due to the lack of smoothness, the most accurate results for a fixed #DOF are obtained with MCL-Q₁.

(a) DG- \mathbb{Q}_1 , $H_h(\cdot, 0) \in [0.10, 1.00]$ (b) DG- \mathbb{Q}_1 , $H_h \in [7.98\text{E-}3, 1.02]$ (c) DG- \mathbb{Q}_0 , $H_h \in [0.10, 0.98]$



(d) MCL- \mathbb{Q}_1 , $H_h \in [0.10, 1.00]$ (e) MCL- \mathbb{Q}_3 , $H_h \in [0.10, 1.00]$ (f) MCL- \mathbb{Q}_7 , $H_h \in [0.10, 1.00]$

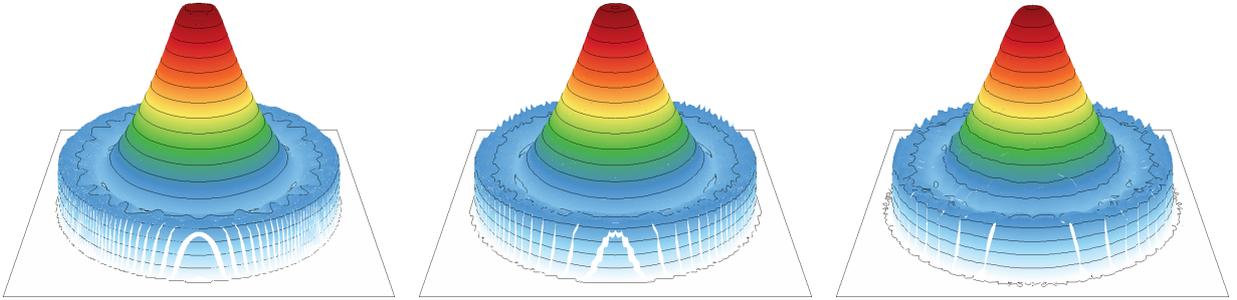


Figure 6: Radial dam break problem for the 2D shallow water equations. The diagrams show (a) the discrete initial condition $H_h(\cdot, 0)$ for the water height and (b)–(f) numerical approximations $H_h(\cdot, 0.06)$ to the water height at $t = 0.06$ obtained with DG and MCL schemes using $p \in \{0, 1, 3, 7\}$.

5.4.2. Narrowing channel

Finally, we study another shock dominated shallow water problem which was proposed in [50]. The domain

$$\Omega = \left\{ (x, y) \in (-10, 80) \times (0, 40) : \tan\left(\frac{\pi}{36}\right)x < y < 40 - \tan\left(\frac{\pi}{36}\right)x \right\}$$

is a 40 meters wide and 90 meters long channel. The lateral wall boundaries are horizontal for the first 10 meters of the channel, after which they are symmetrically constricted with an angle of 5° . At the left boundary, the inflow values $U_{in} = (1, 1, 0)^T$ are prescribed. The right boundary is an outlet, and the remainder of $\partial\Omega$ is a reflecting wall. The gravitational constant is set to $g = 0.16$, which makes the flow regime supercritical with Froude number 2.5. For the initial condition $U_0 = (1, 1, 0)^T$, shocks start to develop at the boundary constrictions. They then travel towards the domain center, interact with each other, and are reflected again at the opposite boundary, after which the solution approaches a symmetric steady state. Since temporal accuracy is irrelevant for such problems, we march numerical solutions to the steady state using the forward Euler method. Computations are terminated when the stopping criterion

$$r^k := |U^k - U^{k+1}| < 10^{-12} \quad (45)$$

is satisfied for solution vectors U^k, U^{k+1} containing the degrees of freedom for all unknowns at pseudo-time steps k and $k + 1$, respectively.

Remark 9. We define r^k in this way to facilitate comparison with limiters for which the stationary nonlinear problem has no well-defined residual. For monolithic limiters, residual-based stopping criteria are preferable to (45) because stagnation of U^k can lead to cancellation effects. These may result in a termination of the simulation run before the norm of the residual becomes smaller than a prescribed tolerance. If the residual is not decreasing monotonically, one should make sure that it stays as small as desired for a certain number of consecutive steps.

We solve the narrowing channel problem using pseudo-time step $\Delta t = 0.025$ on an unstructured triangular mesh consisting of $E = 12,620$ elements. In Fig. 7, we display the distribution of the water height H_h at steady state along with the evolution of the convergence indicators r^k for DG- \mathbb{P}_1 , LO- \mathbb{P}_1 , and MCL- \mathbb{P}_1 . As in other numerical examples, the unlimited solution violates maximum principles, while the low order method smears the shocks almost beyond recognition in this test. The MCL solution is nonoscillatory and exhibits crisp resolution of shocks, as well as good symmetry preservation properties even on this unstructured mesh which is not aligned with the shock angles. As expected, DG- \mathbb{P}_1 and LO- \mathbb{P}_1 converge to steady state. Remarkably, so does MCL- \mathbb{P}_1 , which makes it the first DG limiter of its kind. The fastest convergence is achieved with the unlimited DG scheme. However, the number of pseudo-time steps for the LO and MCL calculations is still reasonable.

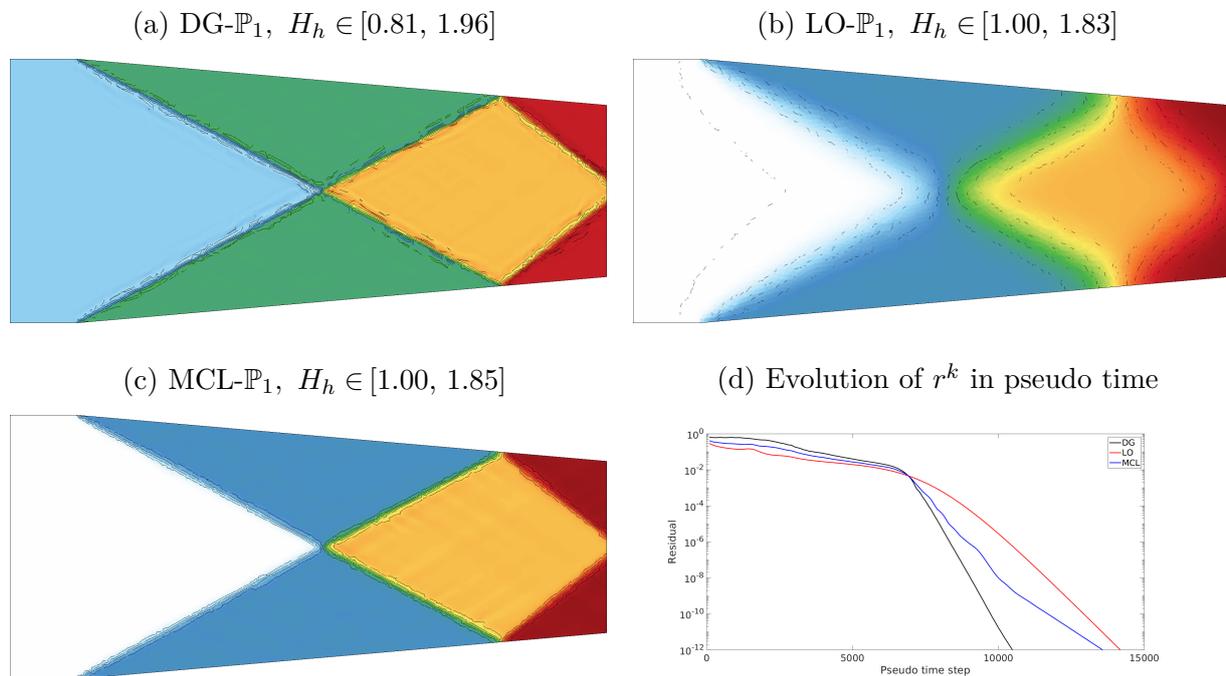


Figure 7: Constricted channel flow for the shallow water equations, water height H_h at steady state obtained with (a) DG- \mathbb{P}_1 , (b) LO- \mathbb{P}_1 , (c) MCL- \mathbb{P}_1 . Diagram (d) plots the convergence indicators r^k vs. pseudo time using a logarithmic scale.

To further motivate the use of monolithic limiters, we solved the narrowing channel problem with another code for the shallow water equations, a simulation tool based on the open source MATLAB/GNU Octave package FESTUNG [14]. Its DG solver is equipped with a vertex-based slope limiter [33] which can be applied (i) just to the water height, (ii) to the water height and momentum separately or (iii) to the water height and momentum sequentially (as in [12, 26]). In contrast to MCL, none of these limiting procedures was even close to satisfying the stopping criterion (45). Only the low order method (corresponding to DG- \mathbb{P}_0) as well as the unlimited solution (corresponding to DG- \mathbb{P}_1) did converge. This outcome is not surprising since the underlying slope limiters are used as postprocessing tools that are algebraically equivalent to localized FCT algorithms [12, 43]. For such methods, convergence to steady-state solutions can not be expected since the residuals depend on the time step size of the pseudo-time stepping method.

We admit that we have also observed stagnation of the convergence indicators r^k for MCL- \mathbb{P}_p with $p > 1$. This disappointing result motivates further theoretical and numerical studies of monolithic limiters for high order DG methods. The ability of the MCL- \mathbb{P}_1 scheme to produce a fully converged steady state solution on an unstructured mesh indicates that steady state convergence can also be achieved with high order extensions if the local bounds are properly chosen and forward Euler pseudo-time stepping is replaced with more advanced iterative solvers for the stationary nonlinear problem.

6. Conclusions

The presented research indicates that monolithic convex limiting is a promising algebraic approach to designing new high order property-preserving DG schemes for hyperbolic conservation laws and systems thereof. We have shown that some components of existing limiting techniques for continuous finite elements require careful generalizations in the DG setting. The accuracy and robustness of our method were demonstrated in numerical studies for scalar equations and systems alike. Thus, we believe that it is suitable for a large class of hyperbolic problems. To the author's best knowledge, the proposed approach constitutes the first monolithic limiting tool for very high order DG methods. The fact that we were able to obtain a fully converged steady state solution of the shallow water equations with the MCL- \mathbb{P}_1 version on an unstructured mesh is a very encouraging result as well.

Recent efforts aimed at the development of limiters for high order finite element approximations [5, 27, 43, 44] are strongly motivated by the fact that they are ideally suited for high-performance computing. The presented limiting methodology can be readily combined with DG discretizations of any order, as illustrated by numerical examples using polynomial degrees of up to $p = 31$. For shock-dominated benchmarks considered in this work, piecewise linear approximations were found to outperform higher order methods using the same number of unknowns. However, local p -refinement may be more beneficial than local h -refinement in smooth regions. Since hp adaptivity is relatively easy to implement for DG methods, we envisage that the proposed methodology may be used to enforce preservation of local bounds in this context. Employing a smoothness indicator that restricts the use of limiting to 'troubled' cells (cf. [13, 23, 28, 44, 46]) would reduce the computational cost and further

improve the capacity of our method to achieve optimal accuracy while preserving important qualitative properties of the exact solution.

The proposed schemes performed reasonably well for all benchmarks considered in the presented study. However, when it comes to solving nonlinear conservation laws with nonunique solutions, additional difficulties can arise even if all local maximum principles are satisfied. Nonphysical behavior of limited high-order solutions is frequently observed, e.g., for the KPP problem [31] or the Woodward-Colella blast wave benchmark [47]. In most cases, it can be cured by performing additional limiting to enforce entropy conditions (cf. [35, 37, 38]) and/or positivity preservation (cf. [12, 26, 36]).

Acknowledgments

The author would like to thank Prof. Dmitri Kuzmin (TU Dortmund University) for many insightful suggestions and supporting this effort in general. Special thanks go also to Dr. Manuel Quezada de Luna (King Abdullah University of Science and Technology) for many fruitful discussions.

Appendix A. Box element dissipation along Cartesian cross stencil

Lohmann et al. [43] proved that the application of their preconditioner to the 1D element matrix of the discretized advection operator reduces the stencil of the high order Bernstein finite element approximation to the set of nearest neighbors in the case of constant velocity. The sparsity of $\tilde{\mathbf{C}}^e$ in 1D follows from this result. Kuzmin and Quezada de Luna [39] generalized it to multiple dimensions and simplicial elements. In the case of box elements, their analysis can be further refined by showing that the sparse matrices $\tilde{\mathbf{C}}_k^e$ have compact Cartesian cross stencils, i.e., that their entries associated with pairs of diagonal nearest neighbors are zero. This remarkable property was already mentioned in Remark 3. In this appendix, we prove it under the assumption that the element Jacobian \mathbf{J}^e is constant or is approximated by a constant matrix. If this assumption is not satisfied, then neither $\tilde{\mathbf{C}}_k^e$ nor D^e are sparse but the magnitude of their entries corresponding to pairs of far away neighbors is small compared to pairs of nearest neighbors.

Without loss of generality, we assume lexicographical ordering of degrees of freedom within the element. Let us first consider the reference element \hat{K} and the associated basis functions $\hat{\varphi}_i$ which are products of 1D Bernstein polynomials \hat{b}_{i_l} of degree p (this superscript is omitted for readability)

$$\hat{\varphi}_i(\hat{\mathbf{x}}) = \prod_{l=1}^d \hat{b}_{i_l}(\hat{x}_l), \quad \frac{\partial \varphi_i(\hat{\mathbf{x}})}{\partial \hat{x}_k} = \hat{b}'_{i_k}(\hat{x}_k) \prod_{\substack{l=1 \\ l \neq k}}^d \hat{b}_{i_l}(\hat{x}_l).$$

The reference element matrices $\hat{\mathbf{C}} = (\hat{C}_1, \dots, \hat{C}_d)$ of the (consistent) discrete gradient

operator are given by

$$\left(\hat{C}_k\right)_{ij} := \int_{\hat{K}} \hat{\varphi}_i \frac{\partial \hat{\varphi}_j}{\partial \hat{x}_k} d\hat{\mathbf{x}} = \int_0^1 \hat{b}_{i_k}(\hat{x}_k) \hat{b}'_{j_k}(\hat{x}_k) d\hat{x}_k \prod_{\substack{l=1 \\ l \neq k}}^d \int_0^1 \hat{b}_{i_l}(\hat{x}_l) \hat{b}_{j_l}(\hat{x}_l) d\hat{x}_l.$$

Let $\mathbf{I} \in \mathbb{R}^{(p+1) \times (p+1)}$ be the identity matrix. Then M_C^e, \hat{C}_k can be computed from Kronecker products, denoted by \otimes , of their 1D counterparts $\hat{M}_C^{1D}, \hat{C}^{1D} \in \mathbb{R}^{(p+1) \times (p+1)}$ as follows:

$$\begin{aligned} \left(\hat{M}_C^{1D}\right)_{ij} &= \int_0^1 \hat{b}_i(\hat{x}) \hat{b}_j(\hat{x}) d\hat{x}, & \left(\hat{C}^{1D}\right)_{ij} &= \int_0^1 \hat{b}_i(\hat{x}) \hat{b}'_j(\hat{x}) d\hat{x}, \\ M_C^e &= |K^e| \underbrace{\hat{M}_C^{1D} \otimes \cdots \otimes \hat{M}_C^{1D}}_{d \text{ times}}, & M_L^e &= \frac{|K^e|}{(p+1)^d} \underbrace{\mathbf{I} \otimes \cdots \otimes \mathbf{I}}_{d \text{ times}}, \\ \hat{C}_k &= \underbrace{\hat{M}_C^{1D} \otimes \cdots \otimes \hat{M}_C^{1D}}_{(d-k) \text{ times}} \otimes \hat{C}^{1D} \otimes \underbrace{\hat{M}_C^{1D} \otimes \cdots \otimes \hat{M}_C^{1D}}_{(k-1) \text{ times}}, & k &\in \{1, \dots, d\}. \end{aligned}$$

Next, we utilize the associativity of \otimes , as well as the identities

$$(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}, \quad (A \otimes B)(C \otimes D) = (AC) \otimes (BD)$$

for matrices A, B, C, D of appropriate size. Invoking integral transformation rules and applying the preconditioner $P^e = M_L^e (M_C^e)^{-1}$ to $\mathbf{C}^e = |K^e| (\mathbf{J}^e)^{-T} \hat{\mathbf{C}} = \text{adj}(\mathbf{J}^e)^T \hat{\mathbf{C}}$, where $\text{adj}(\cdot)$ denotes the adjugate of a matrix, we observe that

$$\begin{aligned} \tilde{C}_k^e &= \frac{|K^e|}{(p+1)^d} \left(|K^e| \hat{M}_C^{1D} \otimes \cdots \otimes \hat{M}_C^{1D} \right)^{-1} \left(\sum_{l=1}^d \text{adj}(\mathbf{J}^e)_{lk} \hat{C}_l \right) \\ &= \sum_{l=1}^d \frac{\text{adj}(\mathbf{J}^e)_{lk}}{(p+1)^d} \left(\hat{M}_C^{1D} \otimes \cdots \otimes \hat{M}_C^{1D} \right)^{-1} \left(\hat{M}_C^{1D} \otimes \cdots \otimes \hat{C}^{1D} \otimes \cdots \otimes \hat{M}_C^{1D} \right) \\ &= \sum_{l=1}^d \frac{\text{adj}(\mathbf{J}^e)_{lk}}{(p+1)^d} \left(\mathbf{I} \otimes \cdots \otimes \mathbf{I} \otimes \left(\left(\hat{M}_C^{1D} \right)^{-1} \hat{C}^{1D} \right) \otimes \mathbf{I} \otimes \cdots \otimes \mathbf{I} \right). \end{aligned} \quad (\text{A.1})$$

The fact that \tilde{C}_k^e has zero entries for all pairs of diagonal neighbors follows from the observation that (A.1) admits a decomposition into matrices corresponding to each spatial dimension. Due to (10), the dissipative matrices D^e have the same sparsity pattern as \tilde{C}_k^e . \square

Example 1. For $p = 1$ we have

$$\hat{M}_C^{1D} = \frac{1}{6} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}, \quad 2\hat{C}^{1D} = \left(\hat{M}_C^{1D} \right)^{-1} \hat{C}^{1D} = \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix}$$

and, therefore, for a (quadrilateral) box element in 2D, it follows that

$$\tilde{C}_1^e = \frac{\text{adj}(\mathbf{J}^e)_{11}}{4} \begin{bmatrix} -1 & 1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & -1 & 1 \end{bmatrix} + \frac{\text{adj}(\mathbf{J}^e)_{21}}{4} \begin{bmatrix} -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}.$$

The matrix \tilde{C}_2^e is obtained similarly. Note that the entries corresponding to pairs of diagonal neighbors (1, 4), (2, 3), (3, 2), (4, 1) are zero.

Appendix B. Calculation of preconditioned gradient matrices

Formula (A.1) not only illustrates the proposition that dissipation is only added in Cartesian directions but also presents a practical way of computing \tilde{C}_k^e similarly to the 1D case. For box elements, \tilde{C}_k^e should be calculated via (A.1) because computation/application of the preconditioner P^e requires inversion of \hat{M}_C^{1D} , which may give rise to significant errors in the case of high order Bernstein polynomials. This issue arises due to the ill-conditioning of Bernstein mass matrices, which can destroy the sparsity of \tilde{C}_k^e for orders as small as $p = 4$. Fortunately, we do not need to invert \hat{M}_C^{1D} since we can apply a formula derived by Lohmann et al. [43] to our setting thus:

$$\left(\left(\hat{M}_C^{1D} \right)^{-1} \hat{C}^{1D} \right)_{ij} = \begin{cases} p + 2 - j & \text{if } i = j - 1, \\ 2(j - 1) - p & \text{if } i = j, \\ -j & \text{if } i = j + 1, \\ 0 & \text{otherwise.} \end{cases}$$

Matters become more involved if simplices are considered because a decomposition like (A.1) is impossible. Using degree elevation formulas for Bernstein polynomials [29], Kuzmin and Quezada de Luna [39] derived the identity

$$\nabla b_\alpha^p = \sum_{k,l=1}^{d+1} \max\{0, \min\{p, (\alpha_l - \delta_{kl} + 1)\}\} \nabla a_k b_{\alpha - e_k + e_l}^p = \sum_{|\beta|=p} \mu_\beta b_\beta^p, \quad (\text{B.1})$$

where a_k are the barycentric coordinates. From (B.1), one can see that the coefficients μ_β are nonzero only for nearest neighbors (for simplices one has also to consider diagonal neighbors). For a given space dimension d and polynomial degree p , the second identity in (B.1) can be used to compute μ_β , which was shown in [39] to coincide with the entries of $(M_L^e)^{-1} \tilde{C}^e$ corresponding to pairs of nodes associated with multi-indices α, β . Below we present MATLAB routines that compute \tilde{C}_k on the reference simplex for any p . These codes are available on GitHub [25] as well. Algorithm 1 relies on Algorithms 2 and 3, which assume lexicographical numbering of nodes and only cover the case of triangular elements ($d = 2$). Algorithm 1 remains valid for tetrahedra. However, one has to implement analogous mappings.

Algorithm 1: Calculation of preconditioned gradient matrices on simplicial elements.

```
function CTilde = computePrecMatSimplex(dim, p)
N = nchoosek(p+dim,p); % number of local degrees of freedom
CTilde = zeros(N,N,dim); % initialization
i2Alpha = mapI2Alpha(p); % array mapping local indices to multiindices
volRefSimplex = 1 / factorial(dim); % |\hat{K}|
M_L = volRefSimplex / N; % diagonal entries of lumped mass matrix \hat{M}_L
```

```

gradBary = [-ones(1,dim); eye(dim)]; % gradients of barycentric coordinates
for m = 1:dim % compute \tilde{\hat{C}}_m
    for j = 1:N % compute (\tilde{\hat{C}}_m)_{:,j}
        alpha = i2Alpha(j,:); % multiindex for j-th local index
        for k = 1:dim+1
            for l = 1:dim+1
                beta = alpha;
                beta(k) = beta(k) - 1;
                beta(l) = beta(l) + 1;
                if (beta(l) > p || beta(k) < 0)
                    continue;
                end
                i = getIfromAlpha(p,beta); % local index corresponding to beta
                CTilde(i,j,m) = CTilde(i,j,m) + gradBary(k,m) ... % cf.
                    * (alpha(l) - (l==k) + 1); % (B.1)
            end
        end
    end
end
CTilde = M_L * CTilde; % multiplication with lumped mass matrix
end

```

Algorithm 2: Mapping of local numbers to multiindices on triangular elements.

```

function i2alpha = mapI2Alpha(p)
dim = 2; % spacial dimension
N = nchoosek(p+dim,p); % number of local degrees of freedom
i2alpha = zeros(N,dim+1); % initialization
a = [p 0 0]; % first multiindex
ctr = 1; % counter variable
for j = 0:p % step through DOF in y-direction
    for i = 0:p-j % step through DOF in x-direction
        i2alpha(ctr,:) = a; % fill map entry
        ctr = ctr+1; % update counter
        a(1:2) = a(1:2) + [-1 1]; % go to next multiindex in x-direction
    end
    a(1) = p-1-j; % go to next multiindex in y-direction
    a(2) = 0; % go to first DOF in x-direction
    a(3) = a(3)+1; % go to next DOF in y-direction
end
end
end

```

Algorithm 3: Mapping of multiindices to local numbers on triangular elements.

```

function i = getIfromAlpha(p,alpha)
i = (p+1)*alpha(3) - alpha(3)*(alpha(3)-1)/2 + alpha(2);
i = i+1; % MATLAB counts indices starting at 1.
end

```

References

- [1] MFEM: Modular finite element methods library. <https://mfem.org>.
- [2] R. Abgrall. Essentially non-oscillatory residual distribution schemes for hyperbolic problems. *J. Comput. Phys.*, 214:773–808, 2006. doi: 10.1016/j.jcp.2005.10.034.
- [3] R. Abgrall and J. Treflík. An example of high order residual distribution scheme using non-lagrange elements. *J. Sci. Comput.*, 45:3–25, 2010. doi: 10.1007/s10915-010-9405-y.
- [4] M. Ainsworth, G. Andriamaro, and O. Davydov. Bernstein–Bézier finite elements of arbitrary order and optimal assembly procedures. *SIAM J. Sci. Comput.*, 33:3087–3109, 2011. doi: 10.1137/11082539X.

- [5] R. Anderson, V. Dobrev, T. Kolev, D. Kuzmin, M. Quezada de Luna, R. Rieben, and V. Tomov. High-order local maximum principle preserving (MPP) discontinuous Galerkin finite element method for the transport equation. *J. Comput. Phys.*, 334:102–124, 2017. doi: 10.1016/j.jcp.2016.12.031.
- [6] R. Anderson, J. Andrej, A. Barker, J. Bramwell, J.-S. Camier, J. Cerveny, V. Dobrev, Y. Doudoit, A. Fisher, T. Kolev, et al. MFEM: a modular finite element methods library. *arXiv preprint arXiv:1911.09220*, 2019.
- [7] S. Badia, J. Bonilla, and A. Hierro. Differentiable monotonicity-preserving schemes for discontinuous Galerkin methods on arbitrary meshes. *Comput. Method. Appl. M.*, 320:582–605, 2017. doi: 10.1016/j.cma.2017.03.032.
- [8] G. R. Barrenechea, V. John, and P. Knobloch. Analysis of algebraic flux correction schemes. *SIAM J. Numer. Anal.*, 54:2427–2451, 2016. doi: 10.1137/15M1018216.
- [9] T. Barth and D. Jespersen. The design and application of upwind schemes on unstructured meshes. In *Proc. AIAA 27th Aerospace Sciences Meeting, Reno*, page 0366, 1989. doi: 10.2514/6.1989-366.
- [10] C. Berthon. Stability of the MUSCL schemes for the Euler equations. *Commun. Math. Sci.*, 3:133–157, 2005. URL projecteuclid.org/euclid.cms/1118778272.
- [11] J. P. Boris and D. L. Book. Flux-corrected transport. I. SHASTA, a fluid transport algorithm that works. *J. Comput. Phys.*, 11:38–69, 1973. doi: 10.1016/0021-9991(73)90147-2.
- [12] V. Dobrev, T. Kolev, D. Kuzmin, R. Rieben, and V. Tomov. Sequential limiting in continuous and discontinuous Galerkin methods for the Euler equations. *J. Comput. Phys.*, 356:372 – 390, 2018. doi: 10.1016/j.jcp.2017.12.012.
- [13] M. Dumbser, O. Zanotti, R. Loubère, and S. Diot. A posteriori subcell limiting of the discontinuous Galerkin finite element method for hyperbolic conservation laws. *J. Comput. Phys.*, 278:47–75, 2014. doi: 10.1016/j.jcp.2014.08.009.
- [14] F. Frank, B. Reuter, V. Aizinger, H. Hajduk, and A. Rupp. FESTUNG: The Finite Element Simulation Toolbox for UNstructured Grids, 2020. URL <https://github.com/FESTUNG>. Version 1.0.
- [15] S. K. Godunov. A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics. *Mat. Sb.*, 89:271–306, 1959. URL <http://mi.mathnet.ru/eng/msb4873>.
- [16] S. Gottlieb, C.-W. Shu, and E. Tadmor. Strong Stability-Preserving High-Order Time Discretization Methods. *SIAM Rev.*, 43:89–112, 2001. doi: 10.1137/S003614450036757X.
- [17] J.-L. Guermond and M. Nazarov. A maximum-principle preserving C0 finite element method for scalar conservation equations. *Comput. Method. Appl. M.*, 272:198–213, 2014. doi: 10.1016/j.cma.2013.12.015.
- [18] J.-L. Guermond and B. Popov. Invariant domains and first-order continuous finite element approximation for hyperbolic systems. *SIAM J. Numer. Anal.*, 54:2466–2489, 2016. doi: 10.1137/16M1074291.
- [19] J.-L. Guermond and B. Popov. Fast estimation from above of the maximum wave speed in the Riemann problem for the Euler equations. *J. Comput. Phys.*, 321:908–926, 2016. doi: 10.1016/j.jcp.2016.05.054.
- [20] J.-L. Guermond and B. Popov. Invariant domains and second-order continuous finite element approximation for scalar conservation equations. *SIAM J. Numer. Anal.*, 55:3120–3146, 2017. doi: 10.1137/16M1106560.
- [21] J.-L. Guermond, M. Nazarov, B. Popov, and Y. Yang. A second-order maximum principle preserving Lagrange finite element technique for nonlinear scalar conservation equations. *SIAM J. Numer. Anal.*, 52:2163–2182, 2014. doi: 10.1137/130950240.
- [22] J.-L. Guermond, M. Q. de Luna, B. Popov, C. E. Kees, and M. W. Farthing. Well-balanced second-order finite element approximation of the shallow water equations with friction. *SIAM J. Sci. Comput.*, 40:A3873–A3901, 2018. doi: 10.1137/17M1156162.
- [23] J.-L. Guermond, M. Nazarov, B. Popov, and I. Tomas. Second-order invariant domain preserving approximation of the Euler equations using convex limiting. *SIAM J. Sci. Comput.*, 40:A3211–A3239, 2018. doi: 10.1137/17M1149961.
- [24] J.-L. Guermond, B. Popov, and I. Tomas. Invariant domain preserving discretization-independent schemes and convex limiting for hyperbolic systems. *Comput. Method. Appl. M.*, 347:143–175, 2019. doi: 10.1016/j.cma.2018.11.036.
- [25] H. Hajduk. <https://github.com/HennesHajduk/PrecMatSimplex>.

- [26] H. Hajduk, D. Kuzmin, and V. Aizinger. New directional vector limiters for discontinuous Galerkin methods. *J. Comput. Phys.*, 384:308–325, 2019. doi: 10.1016/j.envsoft.2018.01.003.
- [27] H. Hajduk, D. Kuzmin, T. Kolev, and R. Abgrall. Matrix-free subcell residual distribution for Bernstein finite element discretizations of linear advection equations. *Comput. Method. Appl. M.*, 359:112658, 2020. doi: 10.1016/j.cma.2019.112658.
- [28] H. Hajduk, D. Kuzmin, T. Kolev, V. Tomov, I. Tomas, and J. N. Shadid. Matrix-free subcell residual distribution for Bernstein finite elements: Monolithic limiting. *Comput. Fluids*, 200:104451, 2020. doi: 10.1016/j.compfluid.2020.104451.
- [29] R. C. Kirby. Fast simplicial finite element algorithms using Bernstein polynomials. *Numer. Math.*, 117: 631–652, 2011. doi: 10.1007/s00211-010-0327-2.
- [30] L. Krivodonova. Limiters for high-order discontinuous Galerkin methods. *J. Comput. Phys.*, 226: 879–896, 2007. doi: 10.1016/j.jcp.2007.05.011.
- [31] A. Kurganov, G. Petrova, and B. Popov. Adaptive semidiscrete central-upwind schemes for nonconvex hyperbolic conservation laws. *SIAM J. Sci. Comput.*, 29:2381–2401, 2007. doi: 10.1137/040614189.
- [32] D. Kuzmin. Positive finite element schemes based on the flux-corrected transport procedure. In *Computational Fluid and Solid Mechanics*, pages 887–888. Citeseer, 2001.
- [33] D. Kuzmin. A vertex-based hierarchical slope limiter for p-adaptive discontinuous Galerkin methods. *J. Comput. Appl. Math.*, 233:3077–3085, 2010. doi: 10.1016/j.cam.2009.05.028.
- [34] D. Kuzmin. Algebraic flux correction I. Scalar conservation laws. In *Flux-Corrected Transport: Principles, Algorithms, and Applications*, pages 145–192. Springer, 2012. doi: 10.1007/978-94-007-4038-9_6.
- [35] D. Kuzmin. Entropy stabilization and property-preserving limiters for discontinuous Galerkin discretizations of nonlinear hyperbolic equations. *arXiv preprint arXiv:2004.03521*, 2020.
- [36] D. Kuzmin. Monolithic convex limiting for continuous finite element discretizations of hyperbolic conservation laws. *Comput. Method. Appl. M.*, 361:112804, 2020. doi: 10.1016/j.cma.2019.112804.
- [37] D. Kuzmin and M. Q. de Luna. Algebraic entropy fixes and convex limiting for continuous finite element discretizations of scalar hyperbolic conservation laws. *arXiv preprint arXiv:2003.12007*, 2020.
- [38] D. Kuzmin and M. Q. de Luna. Entropy conservation property and entropy stabilization of high-order continuous galerkin approximations to scalar conservation laws. *arXiv preprint arXiv:2005.08788*, 2020.
- [39] D. Kuzmin and M. Quezada de Luna. Subcell flux limiting for high-order Bernstein finite element discretizations of scalar hyperbolic conservation laws. *J. Comput. Phys.*, 411:109411, 2020. doi: 10.1016/j.jcp.2020.109411.
- [40] D. Kuzmin and S. Turek. Flux correction tools for finite elements. *J. Comput. Phys.*, 175:525–558, 2002. doi: 10.1006/jcph.2001.6955.
- [41] D. Kuzmin, R. Löhner, and S. Turek. *Flux-corrected transport: principles, algorithms, and applications*. Springer, 2012.
- [42] C. Lohmann. *Physics-Compatible Finite Element Methods for Scalar and Tensorial Advection Problems*. Springer Spektrum, 2019. doi: 10.1007/978-3-658-27737-6.
- [43] C. Lohmann, D. Kuzmin, J. N. Shadid, and S. Mabuza. Flux-corrected transport algorithms for continuous Galerkin methods based on high order Bernstein finite elements. *J. Comput. Phys.*, 344: 151–186, 2017. doi: 10.1016/j.jcp.2017.04.059.
- [44] W. Pazner. Sparse invariant domain preserving discontinuous Galerkin methods with subcell convex limiting. *arXiv preprint arXiv:2004.08503*, 2020.
- [45] G. A. Sod. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *J. Comput. Phys.*, 27:1–31, 1978. doi: 10.1016/0021-9991(78)90023-2.
- [46] F. Vilar. A posteriori correction of high-order discontinuous Galerkin scheme through subcell finite volume formulation and flux reconstruction. *J. Comput. Phys.*, 387:245–279, 2019. doi: j.jcp.2018.10.050.
- [47] P. Woodward and P. Colella. The numerical simulation of two-dimensional fluid flow with strong shocks. *J. Comput. Phys.*, 54:115–173, 1984. doi: 10.1016/0021-9991(84)90142-6.
- [48] S. T. Zalesak. Fully multidimensional flux-corrected transport algorithms for fluids. *J. Comput. Phys.*, 31:335–362, 1979. doi: 10.1016/0021-9991(79)90051-2.
- [49] X. Zhang and C.-W. Shu. On maximum-principle-satisfying high order schemes for scalar conservation

- laws. *J. Comput. Phys.*, 229:3091–3120, 2010. doi: 10.1016/j.jcp.2009.12.030.
- [50] O. C. Zienkiewicz and P. Ortiz. A split-characteristic based finite element model for the shallow water equations. *Int. J. Numer. Meth. Fl.*, 20:1061–1080, 1995. doi: 10.1002/flid.1650200823.