

A structure-preserving Lagrangian discontinuous Galerkin method using flux and slope limiting

Joshua Vedral¹, Nathaniel Morgan^{2*}, Dmitri Kuzmin¹, Jacob Moore³

¹*Institute of Applied Mathematics (LS III), TU Dortmund University
Vogelpothsweg 87, D-44227 Dortmund, Germany*

²*Computational Physics Division, Los Alamos National Laboratory
Bikini Atoll Rd SM 30, Los Alamos, NM 87545, USA*

³*ISER Mississippi State University
3909 Halls Ferry Rd, Vicksburg, MS 39180, USA*

Abstract

We introduce a Lagrangian nodal discontinuous Galerkin (DG) cell-centered hydrodynamics method for solving multi-dimensional hyperbolic systems. By incorporating an adaptation of Zalesak's flux-corrected transport algorithm, we combine a first-order positivity-preserving scheme with a higher-order target discretization. This results in a flux-corrected Lagrangian DG scheme that ensures both global positivity preservation and second-order accuracy for the cell averages of specific volume. The correction factors for flux limiting are derived from specific volume and applied to all components of the solution vector. We algebraically evolve the volumes of mesh cells using a discrete version of the geometric conservation law (GCL). The application of a limiter to the GCL fluxes is equivalent to moving the mesh using limited nodal velocities. Additionally, we equip our method with a locally bound-preserving slope limiter to effectively suppress spurious oscillations. Nodal velocity and external forces are computed using a multidirectional approximate Riemann solver to maintain conservation of momentum and total energy in vertex neighborhoods. Employing linear finite elements and a second-order accurate time integrator guarantees GCL consistency. The results for standard test problems demonstrate the stability and superb shock-capturing capabilities of our scheme.

Keywords: Lagrangian schemes, discontinuous Galerkin methods, cell-centered, approximate Riemann solver, flux correction, slope limiter, volume consistency

*Corresponding author

Email addresses: `joshua.vedral@math.tu-dortmund.de` (Joshua Vedral¹), `nmorgan@lanl.gov` (Nathaniel Morgan²), `kuzmin@math.uni-dortmund.de` (Dmitri Kuzmin¹), `jlmoore@iser.msstate.edu` (Jacob Moore³)

1. Introduction

Numerical solutions to conservation laws in gas dynamics and magneto-hydrodynamics can be obtained using finite difference (FD), finite volume (FV), or finite element (FE) methods that can be further subdivided into Eulerian, Lagrangian, and arbitrary Lagrangian-Eulerian (ALE) approaches. Within the Eulerian framework, the mesh remains fixed and the material moves through the mesh, introducing numerical viscosity. Especially in the presence of shocks and contact discontinuities, this behavior has a negative impact on accuracy. In contrast, Lagrangian methods directly associate the material motion with the computational mesh. In these methods, the mesh moves alongside the material, offering a flexible framework for capturing shocks in multidimensional flows.

Within the Lagrangian framework, two distinct concepts have emerged — staggered grid and cell-centered schemes. The first Lagrangian schemes were one-dimensional FD staggered-grid hydrodynamic (SGH) methods that discretized the conservation equations in a spatially staggered manner [51]. In this approach, the momentum is computed at the nodes of the mesh, while thermodynamic variables are located at the cell center. In their seminal work, von Neumann and Richtmyer introduced a FD SGH method [51, 65] for one-dimensional flows and equipped it with artificial viscosity, which was later extended by Kolsky to the two-dimensional case [51]. Many subsequent works in the field, including those by Wilkins [67] and Burton [4], would advance the FV Lagrangian SGH method, which uses staggered control volumes. Across multi-dimensional Lagrangian SGH methods, artificial viscosity is essential to achieving stable and accurate solutions of shock problems. Thus, many research efforts have focused on developing accurate and robust artificial viscosity-based methods, wherein dissipation is either added explicitly [26, 49, 66] or through solving Riemann problems [7, 13, 39, 40, 41, 47, 53].

In contrast to FV SGH methods, the cell-centered hydrodynamics (CCH) approach solves the conservation equations for each cell within a single control volume. The computation of nodal velocity for mesh movement is achieved by solving multidimensional approximate Riemann problems at each node or by least squares fitting the velocities on the cell faces that come from 1D Riemann solvers. The first FV CCH method was proposed by Godunov [21, 22]. In contrast to the SGH framework, approximate Riemann solvers of CCH schemes inherently introduce sufficient numerical dissipation. Thus, there is no need for adding artificial viscosity terms. Research in early to mid 1980's by Addessio et al. [1] yielded a multi-dimensional Lagrangian CCH method based on using 1D Riemann solvers and calculating the nodal mesh velocity by least-squares fitting the 1D face velocities. This least squares fitting approach could give rise to spurious mesh motion, requiring the use of a vorticity correction technique for stable solutions [14]. Loubère et al. [42] introduced a novel cell-centered Lagrangian scheme based on a fully Lagrangian formulation of the gas dynamics equations, i.e., the gradient and divergence operators are expressed in Lagrangian form. Després and Mazeran [8] extended this scheme, resulting in the pivotal GLACE scheme [6]. They assumed the velocity to be continuous at nodes, allowing them to develop the first node-centered approximate Riemann solver. The discrete volume evolution equation thus satisfies the geometric conservation law (GCL). The resulting two-dimensional Lagrangian FV CCH scheme is conservative and entropy consistent. Despite its strengths, the nodal velocity in GLACE schemes is sensitive to cell aspect ratios, leading to numerical instabilities. To address this, a new nodal solver was developed by introducing multiple pressure fluxes at the grid

nodes, giving rise to the EUCCLHYD scheme proposed by Maire et al. [45, 46]. Several robust nodal Riemann solvers can be found, e.g., in [5, 44, 45, 52]. Nodal multidimensional approximate Riemann solvers are the mainstay of Lagrangian CCH methods as they deliver accurate and stable mesh motion.

The blending of finite element methods with Lagrangian schemes dates back to the two-dimensional Lagrangian DG hydrodynamics (DGH) scheme which was proposed by Loubère et al. [42] for linear triangular meshes. While their scheme uses nodal basis functions, nearly all subsequent efforts focused on modal DGH techniques; see, e.g., [24, 32, 34, 35, 54, 63, 64]. Recently, an arbitrary-order nodal Lagrangian DGH scheme was developed by Moore et al. [48] for smooth flows and weak shocks. This 3D nodal Lagrangian DGH scheme was later combined with a classical slope limiter to simulate shock problems with linear hexahedral elements [50]. However, aggressive limiting was required to keep the internal energy positive, degrading the accuracy of simulations. The modal and nodal Lagrangian DGH methods can be viewed as extensions of the Lagrangian FV CCH methods. Ellis [15] proposed a natural extension of Lagrangian SGH approaches by employing biquadratic polynomial basis functions for kinematic variables and bilinear basis functions for thermodynamic variables. Dobrev et al. [9] extended the methodology employed by Ellis to allow for arbitrary order of kinematic and thermodynamic basis functions [12], as well as for axisymmetric problems [10]. Scovazzi et al. [59, 60] proposed a SUPG-stabilized formulation for Lagrangian hydrodynamics providing a globally conservative approximation. Their approach allows for the representation of pressure gradients on element interiors, unlike traditional CCH schemes.

Inspired by the progress on 2D Lagrangian DGH schemes, and seeing literature gaps on robust 3D Lagrangian DGH methods, we propose a new element-centered Lagrangian DGH method for solving the Euler equations of gas dynamics in 2D and 3D Cartesian coordinates. Following the principles of flux-corrected transport (FCT) methods [3, 30, 31, 37, 68], we combine a first-order positivity-preserving scheme with a high-order counterpart. Employing flux limiting guarantees global positivity preservation for the element (also called cell or zone) averages of specific volume, while slope limiting effectively suppresses spurious oscillations. Our proposed scheme is fully conservative, GCL consistent, and second-order accurate.

We begin, in Section 2, by formulating the governing equations of gas dynamics in Lagrangian form. The DG spatial discretization is presented in Section 3. We proceed, in Section 4, with the description of a multidimensional approximate Riemann solver. Our limiting strategy is presented in Section 5. A brief discussion on GCL consistency follows in Section 6. The physical quantities are evolved in time using the time integrator presented in Section 7. Numerical experiments for standard benchmark problems are performed in Section 8. Finally, we give concluding remarks in Section 9.

2. Governing equations

The differential Lagrangian forms of the evolution equations for specific volume, momentum, and specific total energy of an inviscid compressible fluid are given by

$$\begin{aligned}\varrho \frac{d\nu}{dt} &= \nabla \cdot \mathbf{u}, \\ \varrho \frac{d\mathbf{u}}{dt} &= \nabla \cdot \boldsymbol{\sigma}, \\ \varrho \frac{d\tau}{dt} &= \nabla \cdot (\boldsymbol{\sigma} \cdot \mathbf{u}),\end{aligned}\tag{2.1}$$

where ϱ is the density, ν is the specific volume, $\mathbf{u}$ is the velocity, τ is the specific total energy, and $\boldsymbol{\sigma}$ is the stress. In gas dynamics, the stress tensor is isotropic and reads $\boldsymbol{\sigma} = -p\mathbf{I}$, where p denotes the pressure and $\mathbf{I}$ is the identity matrix. The pressure is calculated using an equation of state for the material. In the case of a polytropic ideal gas, the pressure can be calculated using

$$p = (\gamma - 1)\varrho e,\tag{2.2}$$

where γ is the heat capacity ratio and e is the specific internal energy.

We further denote by $\mathbf{x}$ the position of a particle in the Eulerian frame and by $\tilde{\mathbf{x}}$ the position of a particle in the Lagrangian frame. The trajectory of a particle or a mesh node moving with velocity $\mathbf{u}$ can now be defined as the solution of the initial value problem

$$\frac{d\mathbf{x}}{dt} = \mathbf{u}, \quad \mathbf{x}|_{t=0} = \tilde{\mathbf{x}}.\tag{2.3}$$

3. Discontinuous Galerkin discretization

Let $\{\omega_c\}$, $c = 1, \dots, C_h$ be a decomposition of the solution domain $\omega(t)$ into C_h non-overlapping polygonal elements (also called cells). We denote by m_c the constant mass in the element ω_c . The initial element configuration, denoted by ω_c^0 , $c = 1, \dots, C_h$ satisfies $\cup_{c \in C_h} \omega_c^0 = \omega(0) =: \omega^0$. Each cell has N_p nodes (or vertices). Let N_c represent the number of unknowns in a cell ω_c . The boundary of the element ω_c is denoted by $\partial\omega_c$.

The deformation of each element is described by linking Lagrangian coordinates (the initial configuration) with Eulerian coordinates (the configuration at time t). Using (2.3), the mapping from the initial configuration to the current configuration can be written as

$$\mathbf{x} = \Pi(\tilde{\mathbf{x}}, t),$$

where $\Pi : \omega_c^0 \rightarrow \omega_c$ is the mapping function. Similarly, an isoparametric mapping from a reference element ω_R to the current configuration is

$$\mathbf{x} = \Phi(\boldsymbol{\xi}, t) = \sum_{p=1}^{N_p} b_p(\boldsymbol{\xi}) \mathbf{x}_p(t), \quad \boldsymbol{\xi} \in \omega_R.$$

Here, $\Phi : \omega_R \rightarrow \omega_c$ is the mapping function, ξ are the reference coordinates, b_p are canonical shape functions at the nodes, and $\mathbf{x}_p$ are the nodal coordinates. Both the deformation gradient $\mathbf{F} = \frac{\partial \mathbf{x}}{\partial \xi}$ and the Jacobian matrix $\mathbf{J} = \frac{\partial \mathbf{x}}{\partial \xi}$ are time-dependent.

Unlike existing DGH methods in which specific volume, velocity and specific total energy are represented as modal fields approximated with Taylor polynomials [32, 35, 54], we opt for nodal basis functions to be determined later. In the process of numerical integration, we evaluate the conserved quantities at Gaussian quadrature points.

We write the evolution equations (2.1) in the generic form

$$\varrho \frac{d\mathbb{U}}{dt} = -\nabla \cdot \mathbb{H}(\mathbb{U}), \quad (3.1)$$

where $\mathbb{U} = (\nu, \mathbf{u}, \tau)^\top$ is the vector of conserved quantities and $\mathbb{H} = (-\mathbf{u}, -\boldsymbol{\sigma}, -\boldsymbol{\sigma} \cdot \mathbf{u})$ is the corresponding flux vector.

Proceeding with spatial discretization using DG finite elements, we introduce the vector of conserved unknowns $\mathbb{U}_h = (\nu_h, \mathbf{u}_h, \tau_h)^\top$. On each element ω_c , we seek an approximate solution

$$\mathbb{U}_{h,c}(\mathbf{x}, t) := \mathbb{U}_h(\mathbf{x}, t)|_{\omega_c} = \sum_{j=1}^{N_c} \mathbb{U}_j(t) \psi_j(\mathbf{x}), \quad \mathbf{x} \in \omega_c, \quad t \in [0, T], \quad (3.2)$$

where ψ_j , $j = 1, \dots, N_c$ are linear Lagrange basis functions.

By inserting (3.2) into (3.1), then multiplying by a sufficiently smooth test function Ψ_i and integrating over the element ω_c , we derive the local weak formulation

$$\int_{\omega_c} \Psi_i \left(\varrho \frac{d\mathbb{U}_{h,c}}{dt} + \nabla \cdot \mathbb{H}(\mathbb{U}_{h,c}) \right) dw = \int_{\partial\omega_c} \Psi_i [\mathbb{H}(\mathbb{U}_{h,c}) \cdot \mathbf{n} - \mathbb{H}^*(\mathbf{u}^*, \boldsymbol{\sigma}^*; \mathbf{n})] ds,$$

where $\mathbf{n}$ is the outward unit normal to the surface and $\mathbb{H}^*$ is a numerical flux that depends on an approximate solution of a Riemann problem posed on the element boundary ds . The computation of the Riemann velocity $\mathbf{u}^*$ and stress $\boldsymbol{\sigma}^*$ will be discussed in the next section.

Using the local consistent mass matrix

$$M_c = (m_{ij})_{i,j=1,\dots,N_c}, \quad m_{ij} = \int_{\omega_c} \varrho \Psi_i \Psi_j dw, \quad (3.3)$$

we obtain

$$\sum_{j=1}^{N_c} m_{ij} \frac{d\mathbb{U}_j}{dt} = - \int_{\omega_c} \Psi_i \nabla \cdot \mathbb{H}(\mathbb{U}_{h,c}) dw + \int_{\partial\omega_c} \Psi_i [\mathbb{H}(\mathbb{U}_{h,c}) \cdot \mathbf{n} - \mathbb{H}^*(\mathbf{u}^*, \boldsymbol{\sigma}^*; \mathbf{n})] ds, \quad i = 1, \dots, N_c.$$

Employing integration by parts and the divergence theorem yields

$$\sum_{j=1}^{N_c} m_{ij} \frac{d\mathbb{U}_j}{dt} = \int_{\omega_c} \nabla \Psi_i \cdot \mathbb{H}(\mathbb{U}_{h,c}) dw - \int_{\partial\omega_c} \Psi_i \mathbb{H}^*(\mathbf{u}^*, \boldsymbol{\sigma}^*; \mathbf{n}) ds, \quad i = 1, \dots, N_c. \quad (3.4)$$

To compute the integrals that appear in (3.4) on the reference element ω_R , we express the corresponding evolution equations as

$$\sum_{j=1}^{N_c} m_{ij} \frac{d\mathbb{U}_j}{dt} = \int_{\omega_R} \nabla_{\xi} \Psi_i \cdot \mathbb{H}(\mathbb{U}_{h,c}) \cdot \mathbf{J}^{-1} \det(\mathbf{J}) d\omega_R - \int_{\partial\omega_R} \Psi_i \cdot \mathbb{H}^*(\mathbf{u}^*, \boldsymbol{\sigma}^*; \mathbf{n}) \cdot \mathbf{J}^{-1} \det(\mathbf{J}) d\mathbf{s}_R \quad (3.5)$$

for $i = 1, \dots, N_c$, where $d\omega_R$ and $d\mathbf{s}_R$ denote integration over the reference element and its surface, respectively.

It is straightforward to verify that the nodal Lagrangian DG method (3.4) conserves mass, momentum, and total energy. Summing over the basis functions and using their partition of unity property, we obtain

$$\sum_{i=1}^{N_c} \sum_{j=1}^{N_c} m_{ij} \frac{d\mathbb{U}_j}{dt} = - \int_{\partial\omega_c} \mathbb{H}^*(\mathbf{u}^*, \boldsymbol{\sigma}^*; \mathbf{n}) d\mathbf{s}. \quad (3.6)$$

Summing over all elements and using the definition of the mass matrix in (3.3), we have

$$\sum_{c=1}^{C_h} \int_{\omega_c} \varrho \frac{d\mathbb{U}_{h,c}}{dt} d\omega = 0.$$

Remark 3.1. The element averages of the Lagrangian DG method (3.4) correspond to a first-order FV method (cf. [18, 19]).

4. Riemann problem

Before starting with the description of the Riemann solver, we introduce some notation. The element ω_c is defined by its nodes (or vertices) $p \in \mathcal{P}(c)$, where $\mathcal{P}(c)$ is the set of all nodes of ω_c . We denote by $\mathcal{C}(p) := \{\omega_c : p \in \omega_c\}$ the set of elements sharing the node p . We split ω_c into subcells ω_{cp} (cf. [18, 19, 35, 54]). Here, ω_{cp} is the subcell associated with cell ω_c and node p , formed by joining the element centroid $\mathbf{x}_c$ to the midpoints of faces f attached to node p . Hence, the set of subcells ω_{cp} for $p \in \mathcal{P}(c)$ constitutes a partition of the element ω_c , i.e., $\omega_c = \cup_{p \in \mathcal{P}(c)} \omega_{cp}$. Let p be a node of element ω_c and f be a face connected to node p in cell ω_c . We denote by $\mathcal{F}(c, p)$ the set of faces of element ω_c attached to p . Each face f of element ω_c is divided into subfaces s_f via the partition of ω_c induced by the subcells ω_{cp} for $p \in \mathcal{P}(c)$. We denote by $\mathcal{SF}(c, p)$ the set of subfaces of element ω_c attached to corner l_{cp} , equivalent to the set of faces of subcell ω_{cp} connected to node p . We then define a_{cps_f} as the weighted length of the subface s_f related to node p in element ω_c , and $a_{cps_f} \mathbf{n}_{cps_f}$ as the weighted area normal vector to the subface s_f related to node p in element ω_c . The geometrical entities attached to element ω_c are displayed in Fig. 1.

We employ the nodal multidirectional approximate Riemann solver proposed in [54] to solve the Riemann problem arising in (3.4). This solver, based on work by Burton et al. [5], computes the Riemann force acting on a subface of the element by

$$\mathbf{F}_{cps_f}^* = a_{cps_f} \mathbf{n}_{cps_f} \boldsymbol{\sigma}_{cps_f}^* = a_{cps_f} \mathbf{n}_{cps_f} \boldsymbol{\sigma}_{l_{cp}} + \mu_{cps_f} a_{cps_f} (\mathbf{u}_p^* - \mathbf{u}_{l_{cp}}). \quad (4.1)$$

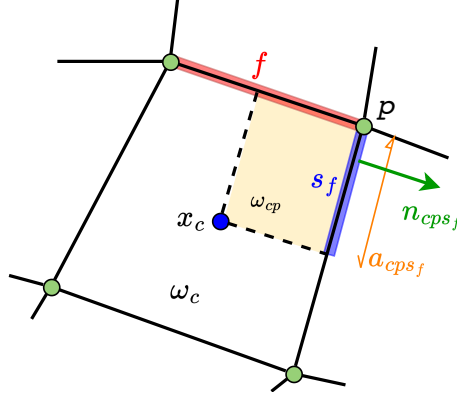


Figure 1: Geometrical entities attached to element ω_c .

Here, $\mu_{cps_f} = \varrho_{cps_f} c^+ |\hat{\mathbf{e}}_{cp} \cdot \mathbf{n}_{cps_f}|$, where $\varrho_{cps_f} c^+$ denotes the acoustic impedance, c^+ is the sound speed of the element, and $\hat{\mathbf{e}}_{cp}$ is a unit vector in the direction of the difference between the node velocity and the element average velocity.

Momentum conservation at node p requires the summation of all forces around the node to be zero, i.e.,

$$\sum_{\omega_c \in \mathcal{C}(p)} \sum_{s_f \in \mathcal{SF}(c,p)} \mathbf{F}_{cps_f}^* = 0. \quad (4.2)$$

Rearranging (4.1) and using (4.2) yields the Riemann velocity

$$\mathbf{u}_p^* = \sum_{\omega_c \in \mathcal{C}(p)} \sum_{s_f \in \mathcal{SF}(c,p)} \frac{\mu_{cps_f} a_{cps_f} \mathbf{u}_{l_{cp}} - a_{cps_f} \mathbf{n}_{cps_f} \sigma_{l_{cp}}}{\mu_{cps_f} a_{cps_f}}. \quad (4.3)$$

The Riemann stress $\sigma_{cps_f}^*$ can now be calculated using (4.3) and the Riemann jump condition.

The approximate Riemann problem in (4.1) introduces dissipation based on discontinuities in the velocity and/or stress fields. Combining (4.2) and (4.3), we observe conservation of total energy. Liu et al. [34] verified that the Lagrangian DG method along with this Riemann solver satisfies the second law of thermodynamics.

5. Limiting

The Lagrangian DG discretization (3.4) combined with the Riemann solver (4.1) suffers from oscillations near shocks and steep gradients. To address these issues, we present a novel limiting approach for Lagrangian DGH methods. Using the limiting framework outlined in [27], we incorporate flux correction for element averages and slope correction for directional derivatives. Aligning with the principles of flux-corrected transport (FCT) methods [3, 30, 31, 37, 68], we adapt Zalesak's algorithm [68, 69] to ensure non-negativity of element-averaged specific volumes in our DG discretization. The correction

factors derived from specific volume are then applied to all components of the solution vector. To ensure local bound preservation, we make use of a clip-and-scale limiter [2, 28, 36]. In line with the employed flux limiting approach, the correction factors for slope limiting are also computed from the specific volume and applied to all fields.

5.1. Element averages of a Lagrangian DGH method

As stated in Remark 3.1, our Lagrangian DG method reduces to a first-order FV scheme when considering only the element averages. Let $\bar{\mathbb{U}}_{h,c} = \frac{1}{m_c} \int_{\omega_c} \mathbb{U}_h dw$ denote the cell average of $\mathbb{U}_h$ over ω_c . In view of (3.6), the element averages of a Lagrangian DG approximation satisfy [18, 19]

$$m_c \frac{d\bar{\mathbb{U}}_{h,c}}{dt} + \int_{\partial\omega_c} \mathbb{H}^*(\mathbf{u}^*, \boldsymbol{\sigma}^*; \mathbf{n}) ds = 0. \quad (5.1)$$

Employing a first-order explicit time integrator turns (5.1) into

$$m_c(\bar{\mathbb{U}}_{h,c}^{n+1} - \bar{\mathbb{U}}_{h,c}^n) + \Delta t \int_{\partial\omega_c} \mathbb{H}^*(\mathbf{u}^*, \boldsymbol{\sigma}^*; \mathbf{n}) ds = 0, \quad (5.2)$$

where $\bar{\mathbb{U}}_{h,c}^{n+1}$ and $\bar{\mathbb{U}}_{h,c}^n$ denote the approximation of $\bar{\mathbb{U}}_{h,c}(t)$ at time instants t^{n+1} and t^n , respectively, and Δt is the time step size, i.e., $\Delta t = t^{n+1} - t^n$.

Applying the subcell-based partition of ω_c from [18, 19] and adopting the notation from the previous section, we rewrite the surface integral in (5.2) as

$$\int_{\partial\omega_c} \mathbb{H}^*(\mathbf{u}^*, \boldsymbol{\sigma}^*; \mathbf{n}) ds = \sum_{p \in \mathcal{P}(c)} \sum_{s_f \in \mathcal{SF}(c,p)} a_{cps_f} \mathbb{H}_{cps_f} \cdot \mathbf{n}_{cps_f} = 0, \quad (5.3)$$

where $\mathbb{H}_{cps_f} = (-\mathbf{u}_p, p_{cps_f} \mathbf{I}, p_{cps_f} \mathbf{u}_p)^\top$. Inserting (5.3) into (5.2), the element averages of a Lagrangian DG approximation can be evolved by

$$m_c(\bar{\mathbb{U}}_{h,c}^{n+1} - \bar{\mathbb{U}}_{h,c}^n) + \Delta t \sum_{p \in \mathcal{P}(c)} \sum_{s_f \in \mathcal{SF}(c,p)} a_{cps_f} \mathbb{H}_{cps_f} \cdot \mathbf{n}_{cps_f} = 0. \quad (5.4)$$

5.2. Flux-corrected transport

In FCT methods, a property-preserving low-order scheme is combined with a high-order target discretization, which generally violates physical properties. Following this approach, a low-order version of (5.4) can be derived using

$$\mathbf{u}_p^L = \mathbf{u}_p^n, \quad \mathbf{n}_{cps_f}^L = \mathbf{n}_{cps_f}^n. \quad (5.5)$$

The corresponding low-order fluxes are denoted by $\mathbb{H}_{cps_f}^L$. The so-defined approximation $\bar{\mathbb{U}}_h^L$ is positivity preserving [19] and first-order accurate in time. Alternatively, a second-order accurate but generally not positivity-preserving approximation $\bar{\mathbb{U}}_h^H$ can be obtained using the temporal average

$$\mathbf{u}_p^H = \frac{\mathbf{u}_p^{n+1} + \mathbf{u}_p^n}{2}, \quad \mathbf{n}_{cps_f}^H = \frac{\mathbf{n}_{cps_f}^{n+1} + \mathbf{n}_{cps_f}^n}{2}. \quad (5.6)$$

The corresponding high-order fluxes are denoted by $\mathbb{H}_{cps_f}^H$.

The low- and high-order approximations derived from (5.5) and (5.6), respectively, are linked by

$$m_c \bar{\mathbb{U}}_{h,c}^H = m_c \bar{\mathbb{U}}_{h,c}^L + \Delta t \sum_{p \in \mathcal{P}(c)} \mathbb{H}_{cps_f}^B,$$

where

$$\mathbb{H}_{cps_f}^B = \sum_{s_f \in \mathcal{SF}(c,p)} a_{cps_f} (\mathbb{H}_{cps_f}^L \cdot \mathbf{n}_{cps_f}^L - \mathbb{H}_{cps_f}^H \cdot \mathbf{n}_{cps_f}^H)$$

are numerical fluxes that possess the conservation property

$$\sum_{\omega_c \in \mathcal{C}(p)} \mathbb{H}_{cps_f}^B = 0.$$

Now, we can formulate a flux-corrected Lagrangian DG scheme (cf. [27])

$$m_c \bar{\mathbb{U}}_{h,c}^{n+1} = m_c \bar{\mathbb{U}}_{h,c}^L + \Delta t \sum_{p \in \mathcal{P}(c)} \alpha_p \mathbb{H}_{cps_f}^B. \quad (5.7)$$

The conservation property is preserved by defining

$$\alpha_p = \min_{\omega_c \in \mathcal{C}(p)} \alpha_{cp}$$

using correction factors $\alpha_{cp} \in [0, 1]$ that ensure global positivity preservation for the element averages.

5.3. Flux limiting

It remains to show how the correction factors α_{cp} are calculated. In essence, any element-based limiting technique outlined in [28, Chapter 4] can be adapted and applied. However, instead of constraining element contributions to a node, we constrain nodal contributions to an element. Scalar limiting is applied to the specific volume, and all fluxes in (5.7) are scaled by the resulting correction factor.

Let $\mathbb{H}_{cps_f}^B|_{1/\varrho} := \mathbb{H}_{cps_f}^B(\mathbf{u}^B, \cdot, \cdot)^\top$ be the flux function entries corresponding to the specific volume evolution equation. The nodal-based multidimensional version of Zalesak's limiting strategy [28, 68, 69] is as follows:

(1) Compute the sums of antidiffusive fluxes:

$$P_c^+ = \sum_{p \in \mathcal{P}(c)} \max(0, \mathbb{H}_{cps_f}^B|_{1/\varrho}), \quad P_c^- = \sum_{p \in \mathcal{P}(c)} \min(0, \mathbb{H}_{cps_f}^B|_{1/\varrho}). \quad (5.8)$$

(2) Determine the bounds for limited increments:

$$Q_c^+ = \gamma_c(\mathbb{U}_c^{\max}|_{1/\varrho} - \bar{\mathbb{U}}_{h,c}^L|_{1/\varrho}), \quad Q_c^- = \gamma_c(\mathbb{U}_c^{\min}|_{1/\varrho} - \bar{\mathbb{U}}_{h,c}^L|_{1/\varrho}). \quad (5.9)$$

(3) Compute the bounds for correction factors:

$$R_c^+ = \min \left(1, \frac{Q_c^+}{P_c^+} \right), \quad R_c^- = \min \left(1, \frac{Q_c^-}{P_c^-} \right). \quad (5.10)$$

(4) Select an upper bound:

$$\alpha_{cp} = \begin{cases} R_c^+ & \text{if } \mathbb{H}_{cps_f}^B|_{1/\varrho} \geq 0, \\ R_c^- & \text{if } \mathbb{H}_{cps_f}^B|_{1/\varrho} < 0. \end{cases} \quad (5.11)$$

We set the upper and lower *global* bounds for the specific volume as $\mathbb{U}_c^{\max}|_{1/\varrho} := \mathbb{U}_c^{\max}(1/\varrho, \cdot, \cdot)^\top = \infty$ and $\mathbb{U}_c^{\min}|_{1/\varrho} := \mathbb{U}_c^{\min}(1/\varrho, \cdot, \cdot)^\top = 0$, respectively. Additionally, we set $\gamma_c = m_c/\Delta t$. The correction factors computed from (5.8)–(5.11) ensure a positivity-preserving approximation for the element averages of specific volume.

Remark 5.1. Applying a flux limiter is equivalent to moving the node p using the velocity

$$\mathbf{u}_{p,\text{lim}} = \alpha_p \mathbf{u}_p^H + (1 - \alpha_p) \mathbf{u}_p^L. \quad (5.12)$$

Remark 5.2. Our approach involves scalar limiting of the specific volume. Alternatively, one can employ FCT-type sequential limiting [28, Chapter 4], where different correction factors are computed for each field. For the Euler equations of gas dynamics, after the density limiting step, the gradients of total energy and momentum are adjusted to account for the density changes [11]. An extension of this method to the equations of ideal magnetohydrodynamics can be found in [29].

5.4. Slope limiting

The flux limiter ensures the non-negativity of the element-averaged specific volume $\bar{\mathbb{U}}_{h,c}|_{1/\varrho}$, $c = 1, \dots, C_h$. However, local spurious oscillations, while not violating the positivity of the solution, may still arise. To eliminate these oscillations, we construct a locally bound-preserving conservative approximation $\mathbb{U}_{h,c}^*$ using the clip-and-scale limiting strategy [2, 28, 36].

The element averages $\bar{\mathbb{U}}_{h,c}$ of a Lagrangian DG approximation can be expressed in terms of their nodal values as follows:

$$\bar{\mathbb{U}}_{h,c} = \frac{1}{m_c} \sum_{j=1}^{N_c} m_{c,j} \mathbb{U}_j, \quad m_{c,j} = \int_{\omega_c} \varrho_h \Psi_j \, dw,$$

where $\sum_{j=1}^{N_c} m_{c,j} = m_c$. By defining $\mathbb{H}_{c,j} := m_{c,j}(\mathbb{U}_j - \bar{\mathbb{U}}_{h,c})$, the nodal values $\mathbb{U}_j$ are given by

$$\mathbb{U}_j = \bar{\mathbb{U}}_{h,c} + \frac{\mathbb{H}_{c,j}}{m_{c,j}}, \quad \sum_{j=1}^{N_c} \mathbb{H}_{c,j} = 0.$$

We aim to construct a locally bound-preserving conservative approximation

$$\mathbb{U}_j^* = \bar{\mathbb{U}}_{h,c} + \frac{\mathbb{H}_{c,j}^*}{m_{c,j}}, \quad \sum_{j=1}^{N_c} \mathbb{H}_{c,j}^* = 0 \quad (5.13)$$

using limited fluxes $\mathbb{H}_{c,j}^*$ to be defined below.

Again, we denote by $\mathbb{H}_{c,j}|_{1/\varrho} := \mathbb{H}_{c,j}(\mathbf{u}, \cdot, \cdot)^\top$ the flux function entries corresponding to the specific volume evolution equation. Following the flux limiting approach, the correction factors for slope limiting are derived from the specific volume and subsequently applied to all components of the solution vector.

We begin by limiting each component of the element flux individually, i.e.,

$$\tilde{\mathbb{H}}_{c,j}|_{1/\varrho} = \max(\mathbb{H}_{c,j}^{\min}|_{1/\varrho}, \min(\mathbb{H}_{c,j}|_{1/\varrho}, \mathbb{H}_{c,j}^{\max}|_{1/\varrho})),$$

where

$$\mathbb{H}_{c,j}^{\min}|_{1/\varrho} := \gamma_{c,j}(\mathbb{U}_{c,j}^{\min}|_{1/\varrho} - \mathbb{U}_j|_{1/\varrho}), \quad \mathbb{H}_{c,j}^{\max}|_{1/\varrho} := \gamma_{c,j}(\mathbb{U}_{c,j}^{\max}|_{1/\varrho} - \mathbb{U}_j|_{1/\varrho}) \quad (5.14)$$

and $\gamma_{c,j} = m_{c,j}/\Delta t$. Next, we compute the sums of clipped element contributions

$$P_c^+ = \sum_{j=1}^{N_c} \max(0, \tilde{\mathbb{H}}_{c,j}|_{1/\varrho}), \quad P_c^- = \sum_{j=1}^{N_c} \min(0, \tilde{\mathbb{H}}_{c,j}|_{1/\varrho}) \quad (5.15)$$

and subsequently scale positive or negative components as follows:

$$\mathbb{H}_{c,j}^*|_{1/\varrho} = \begin{cases} -\frac{P_c^-}{P_c^+} \tilde{\mathbb{H}}_{c,j}|_{1/\varrho} & \text{if } \tilde{\mathbb{H}}_{c,j}|_{1/\varrho} > 0 \text{ and } P_c^+ + P_c^- > 0, \\ -\frac{P_c^+}{P_c^-} \tilde{\mathbb{H}}_{c,j}|_{1/\varrho} & \text{if } \tilde{\mathbb{H}}_{c,j}|_{1/\varrho} < 0 \text{ and } P_c^+ + P_c^- < 0, \\ \tilde{\mathbb{H}}_{c,j}|_{1/\varrho} & \text{otherwise.} \end{cases} \quad (5.16)$$

We set the upper and lower *local* bounds for the specific volume as

$$\mathbb{U}_{c,j}^{\max}|_{1/\varrho} := \mathbb{U}_{c,j}^{\max}(1/\varrho, \cdot, \cdot)^\top = \max_{i \in \mathcal{N}_j} \mathbb{U}_i|_{1/\varrho}, \quad \mathbb{U}_{c,j}^{\min}|_{1/\varrho} := \mathbb{U}_{c,j}^{\min}(1/\varrho, \cdot, \cdot)^\top = \min_{i \in \mathcal{N}_j} \mathbb{U}_i|_{1/\varrho},$$

respectively. Here, $\mathcal{N}_j$ is the index set of nodes belonging to elements containing node j .

Remark 5.3. The limiting step (5.16) enforces the zero-sum condition in (5.13).

Remark 5.4. The clip-and-scale strategy is first applied to the specific volume, with the resulting correction factors then used across all fields. Alternatively, an extended version of the clip-and-scale limiter designed for systems can be employed. In this approach, after the scalar limiting of the specific volume, a product rule variant of the clip-and-scale limiter is applied to the remaining fields. For further details, we refer the reader to [28, Section 6.3.2].

6. GCL consistency

As explained in [43], the volume v_c^{n+1} of an element can be calculated using its definition as a function of vertex coordinates or evolved using the geometric conservation law (GCL). Solving (3.4) involves a discretized GCL. The two approaches are equivalent for certain types of elements if the mesh velocity is defined appropriately. To maintain volumetric consistency, we evolve the nodes according to (5.12).

In the one-dimensional case, the discretized GCL for an element $\omega_c = [x_i, x_{i+1}]$ reads

$$\tilde{v}_{i+1/2}^{n+1} = \tilde{v}_{i+1/2}^n + \Delta t(u_{i+1} - u_i),$$

whereas the geometric definition of the evolving element volume is

$$v_{i+1/2}(t) = x_{i+1}(t) - x_i(t), \quad t \in [t^n, t^{n+1}].$$

Let the nodal velocities be fixed on $[t^n, t^{n+1}]$. Then we have

$$x_i^{n+1} = x_i^n + \Delta t u_i, \quad x_{i+1}^{n+1} = x_{i+1}^n + \Delta t u_{i+1}$$

and therefore

$$v_{i+1/2}^{n+1} = v_{i+1/2}^n + \Delta t(u_{i+1} - u_i).$$

Thus, the discretized GCL holds for geometric volumes.

To show GCL consistency for triangles and quadrilaterals, we use the fact that [19, Section 1.3.2]

$$\frac{dv_c}{dt} = \sum_{p \in \mathcal{P}(c)} \sum_{s_f \in \mathcal{SF}(c,p)} a_{cps_f} \mathbf{u}_p \cdot \mathbf{n}_{cps_f}. \quad (6.1)$$

Since the coordinates $x_p(t)$ and $y_p(t)$ of node p are linear in t , the time derivative of $v_c(t)$ and hence the right-hand side of (6.1) is also linear in t . Therefore, any second-order accurate time discretization ensures GCL consistency. However, a first-order accurate time integrator may not guarantee it. A more sophisticated proof of GCL consistency for triangles and quadrilaterals can be found in the [Appendix](#).

Remark 6.1. The impact of the time discretization on GCL consistency and stability is further discussed in [16, 56]. Increasing the degree of the finite element mapping results in a higher polynomial degree of $v_c(t)$. Consequently, the order of the time integrator must match the order of the space discretization for a Lagrangian DG method to be GCL consistent.

Remark 6.2. GCL consistency in 3D has been investigated by Georges [19] and Georges et al. [20] who proposed face-splitting techniques applicable to arbitrary polyhedra. Polyhedra, by definition, have planar faces. In contrast, general hexahedral elements defined by trilinear mappings typically have non-planar faces. The assumption of planar faces simplifies the analysis, as it enables consistent and exact computation of the element volume using methods such as tetrahedral decomposition or closed-form geometric formulas. For elements with non-planar faces, the geometry is not equivalent to a collection of tetrahedra, and the geometric volume defined by the finite element mapping may differ from that obtained by a piecewise linear decomposition.

7. Time integration

The Lagrangian DG scheme (3.4) can be written in matrix form as

$$M \frac{d\mathbb{U}_h}{dt} = R(\mathbb{U}_h), \quad (7.1)$$

where M denotes the global consistent mass matrix and $R(\mathbb{U}_h)$ is the residual vector.

The numerical solution is evolved in time using the second-order strong stability preserving (SSP) Runge-Kutta method [23]

$$\begin{aligned} \mathbb{U}_h^{(1)} &= \mathbb{U}_h^n + \Delta t M^{-1} R(\mathbb{U}_h^n), \\ \mathbb{U}_h^{n+1} &= \frac{1}{2} \mathbb{U}_h^n + \frac{1}{2} \mathbb{U}_h^{(1)} + \frac{1}{2} \Delta t M^{-1} R(\mathbb{U}_h^{(1)}). \end{aligned}$$

The nodal positions $\mathbf{x}_p$ are updated using

$$\begin{aligned} \mathbf{x}_p^{(1)} &= \mathbf{x}_p^n + \Delta t \mathbf{u}_{p,\text{lim}}^n, \\ \mathbf{x}_p^{n+1} &= \frac{1}{2} \mathbf{x}_p^n + \frac{1}{2} \mathbf{x}_p^{(1)} + \frac{1}{2} \Delta t \mathbf{u}_{p,\text{lim}}^{(1)}, \end{aligned}$$

where $\mathbf{u}_{p,\text{lim}}$ denotes the limited Riemann velocity at node p , see (5.12).

We refer the reader to [19, 44] for how to evaluate the time step size based on the CFL criterion.

8. Numerical examples

We solve a series of standard test problems for the Euler equations of gas dynamics to demonstrate the accuracy and robustness of our DGH method. Flux and slope limiters are applied at the end of each Runge-Kutta stage, and all test problems use the gamma-law equation of state as defined in (2.2). A novel contribution of this work is the development of a Lagrangian DGH method capable of simulating three-dimensional problems involving both smooth and shock-driven flows. To the best of our knowledge, accurate and robust 3D Lagrangian DGH schemes for shock-dominated problems are lacking in the literature.

For smooth flows, we consider the Taylor-Green vortex problem to verify the second-order accuracy of the fully discrete scheme. For shock-driven flows, we study the Sedov blast wave, the Noh problem, and the triple point problem, which together demonstrate the accuracy and stability of our DGH method in the presence of strong shocks.

In some test cases, we introduce an additional elementwise-constant parameter β_c that is applied after slope limiting to prevent spurious mesh motion and mesh tangling. Specifically, after obtaining the slope-limited solution $\mathbb{U}_j^*$ from (5.13), we further modify it as

$$\mathbb{U}_j^{*,\text{new}} = \beta_c \mathbb{U}_j^* \quad \forall j \in N_c.$$

By choosing $\beta_c = 1$, we recover exactly the slope-limited solution $\mathbb{U}_j^*$ from (5.13). A similar idea was investigated by Liu et al. [34], who adjusted the slope-limiting parameter in a Barth-Jespersen limiter and found that the expected order of convergence is preserved for $\beta_c \in [0.6, 1.0]$. Following their findings, we also present results using $\beta_c = 0.8$ in cases where mesh tangling is observed. We emphasize that the scheme remains IDP regardless of the value of β_c ; the parameter is introduced solely to enhance mesh stability.

8.1. 2D Taylor-Green vortex problem

In our first numerical experiment, we investigate the two-dimensional Taylor-Green vortex problem [12, 55, 62], which can be derived by adapting an analytical solution of the incompressible Navier-Stokes equations to the compressible Euler equations. The computational domain is $\Omega = (0, 1)^2$, with $\gamma = 5/3$. The initial conditions are given by

$$\varrho^0 = 1, \quad \mathbf{u}^0 = [\sin(\pi x) \cos(\pi y), -\cos(\pi x) \sin(\pi y)]^\top, \quad p^0 = \frac{1}{4}[\cos(2\pi x) + \cos(2\pi y)] + 1.$$

To satisfy the total energy evolution equation, an energy source term S_E is added:

$$S_E = \frac{\pi (\varrho^0)^3}{4 \gamma - 1} [\cos(3\pi x) \cos(\pi y) - \cos(\pi x) \cos(3\pi y)].$$

For details on the derivation and construction of this source term, we refer the reader to [63].

We conduct the simulation on uniformly refined hexahedral meshes consisting of a single-element-thick slab, with resolutions of 8×8 , 16×16 , 32×32 , and 64×64 . The velocity magnitude fields and corresponding meshes at times $t = 0.0$, $t = 0.5$ and $t = 0.75$ are shown in Fig. 2. We observe mesh entangling when using $\beta_c = 1.0$ which can be mitigated by reducing the parameter to $\beta_c = 0.8$. Although the solution becomes severely distorted at later times, the simulation remains stable, and we are able to advance the computation beyond $t = 2.0$ without any mesh remapping or code crashes, even when the mesh is heavily deformed.

8.2. 3D Sedov blast wave problem

The Sedov problem [58, 61] involves an outward-traveling blast wave in a gamma-law gas generated by an energy source. We set $\gamma = 7/5$ and test our method for the three-dimensional case. An exact solution, derived from self-similarity arguments, is available; see, e.g., [25]. The initial conditions read

$$\varrho^0 = 1, \quad \mathbf{u}^0 = 0, \quad p^0 = 10^{-6}.$$

The energy source is positioned at the origin, and the pressure in the element containing the origin is calculated as $p_{or} = (\gamma - 1) \varrho_{or} \frac{e_0}{v_{or}}$, where e_0 denotes the total amount of released internal energy and v_{or} is the volume of the element containing the origin. By setting $e_0 = 0.493390$, the solution features a diverging shock wave of infinite strength, with the front located at a radius of $r = 1$ at $t = 1$, and a peak density reaching 6. The computational domain is $\Omega = (0, 1.2)^3$.

We split the computational domain into hexahedral meshes with 8, 16, and 30 elements in each spatial direction. Fig. 3 shows the density profiles and corresponding scatter plots of element-averaged density versus radius. For all mesh resolutions, the shock front is accurately located and the solution maintains cylindrical symmetry. As the mesh is refined, the density peak nearly reaches 6.

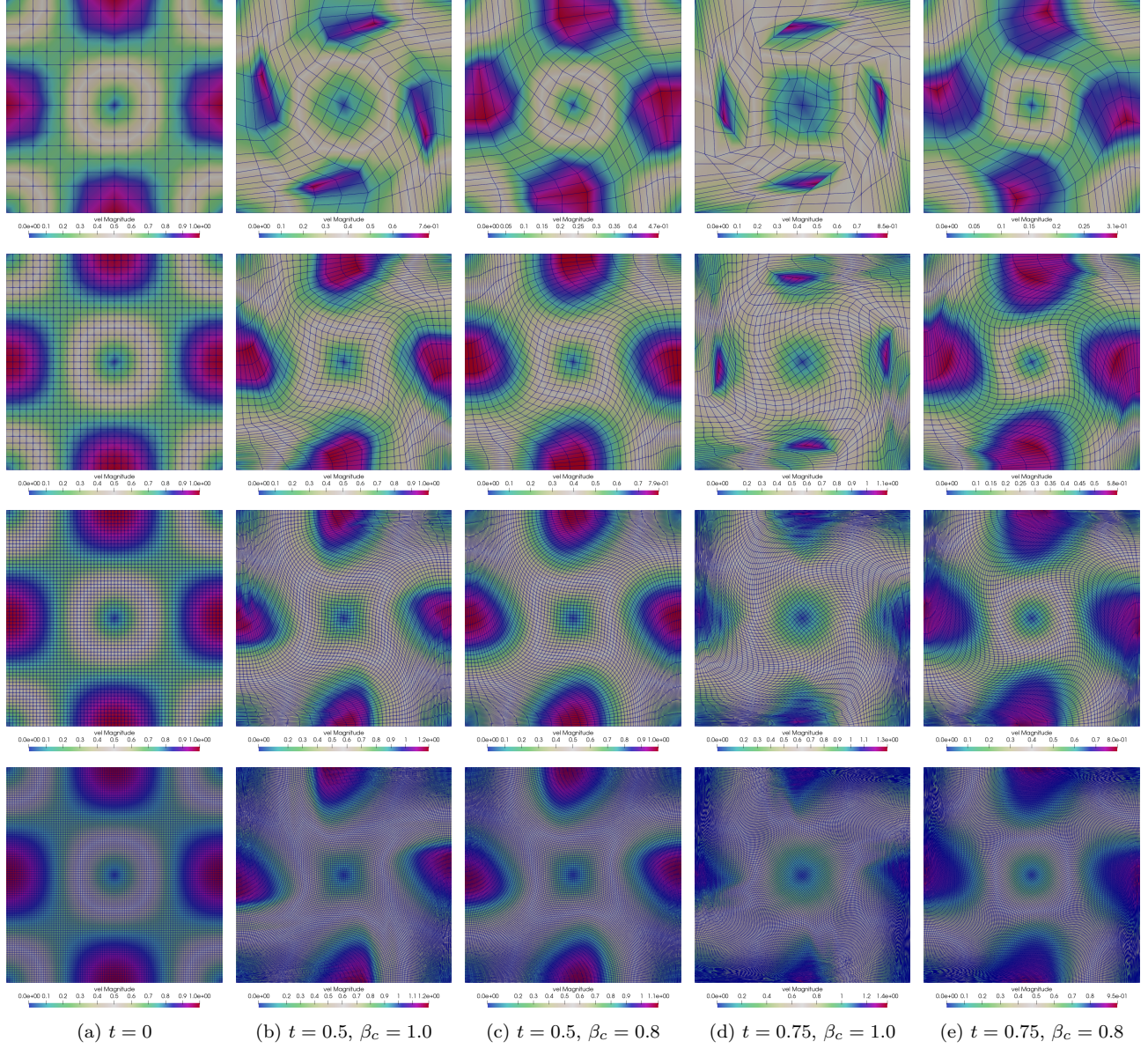


Figure 2: Taylor-Green vortex problem, velocity magnitude fields at different times t for various mesh resolutions. Note that the solver outputs the subcells associated with each element. See Fig. 1 for details.

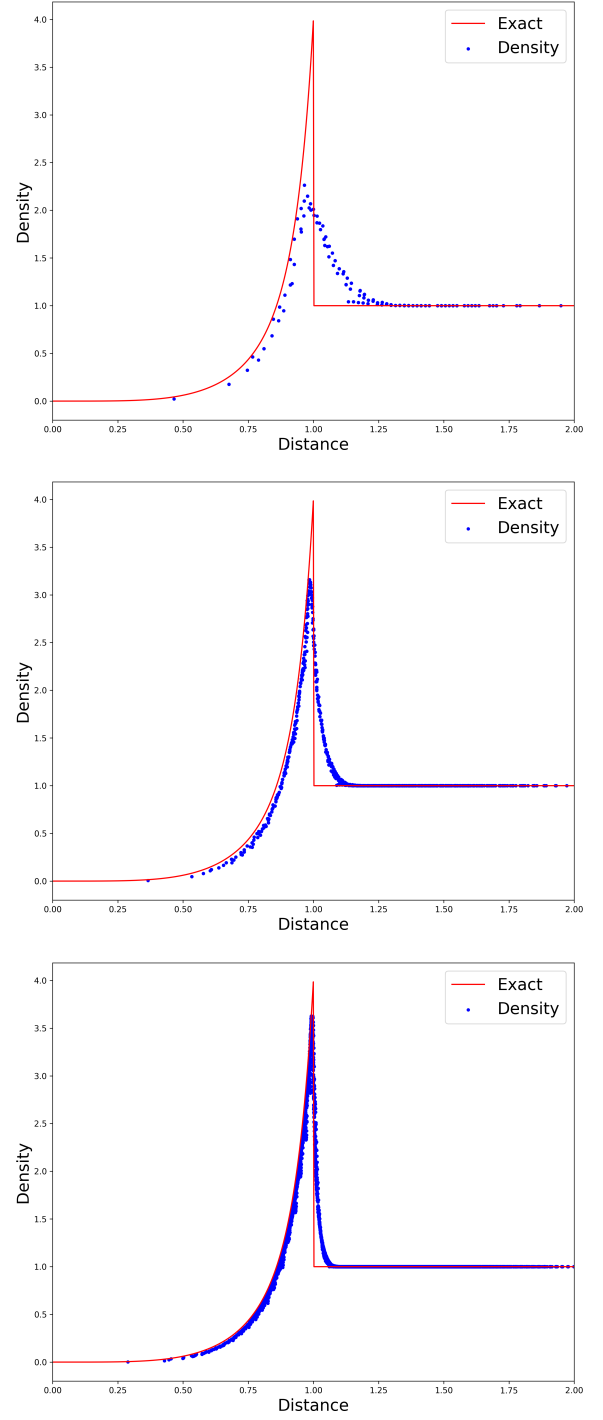
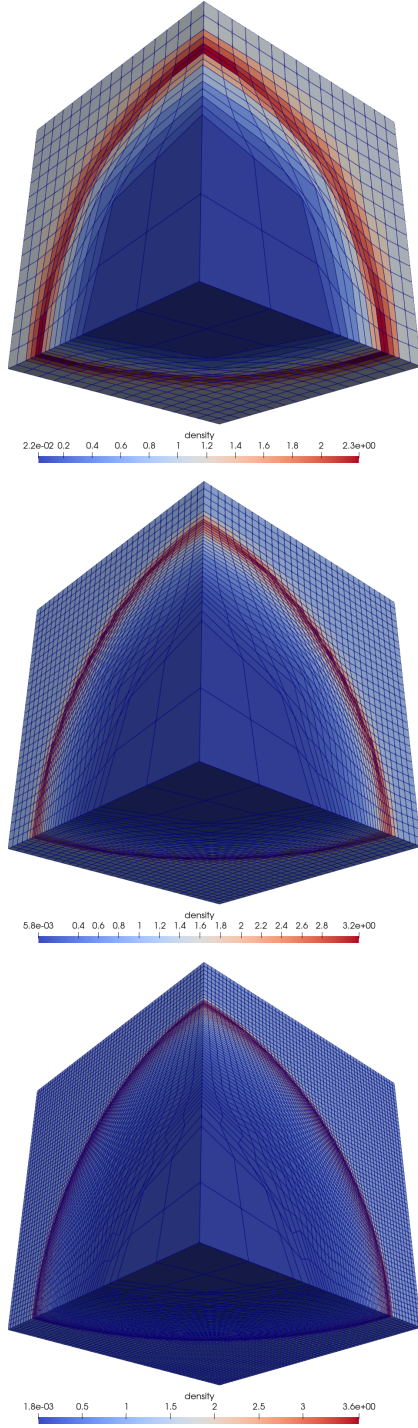


Figure 3: Three-dimensional Sedov blast wave problem, density profiles ϱ at $t = 1.0$ for various mesh resolutions. Note that the solver outputs the subcells associated with each element. See Fig. 1 for details.

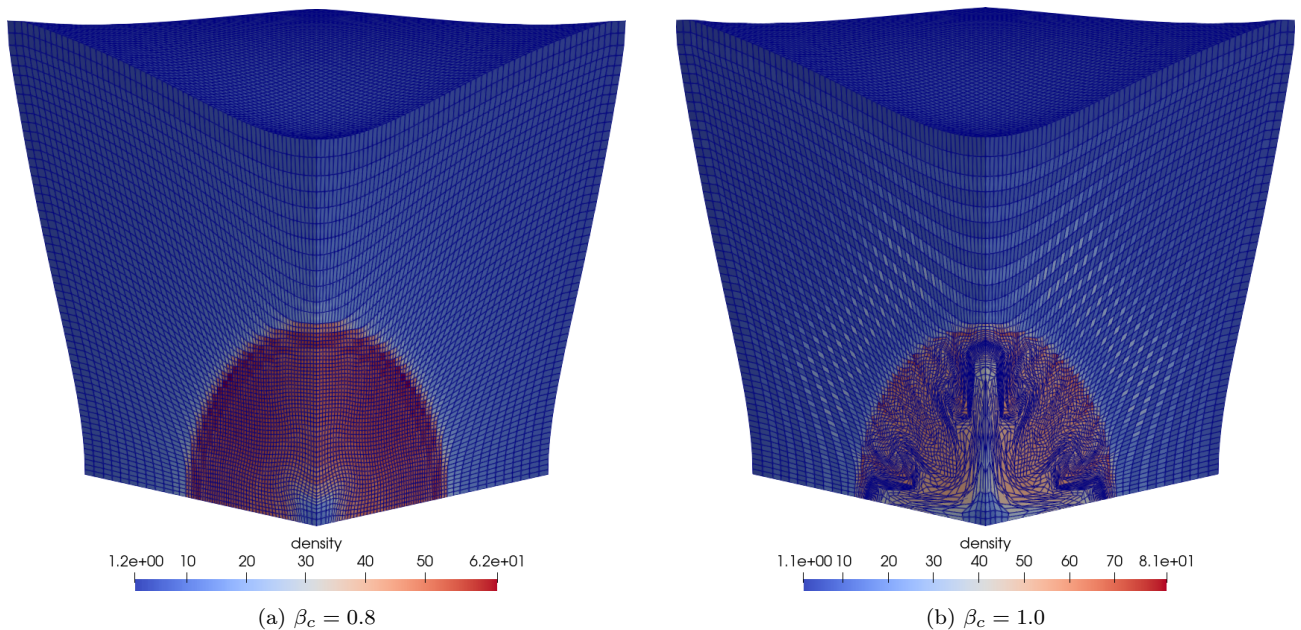


Figure 4: Noh problem, density profiles ρ obtained at $t = 0.6$ on a $30 \times 30 \times 30$ mesh. Note that the solver outputs the subcells associated with each element. See Fig. 1 for details.

8.3. 3D Noh problem

The Noh problem [57] is a standard test case used to validate the accuracy and symmetry preservation of Lagrangian methods. In this scenario, a cold gas with $\gamma = 5/3$ is initially given an inward radial velocity, generating a diverging cylindrical shock wave that propagates at a speed of $s = 1/3$. The density plateau behind the shock wave reaches a value of 16. The initial flow field is given by

$$\rho^0 = 1, \quad \mathbf{u}^0 = (-x/\sqrt{x^2 + y^2}, -y/\sqrt{x^2 + y^2})^\top, \quad p^0 = 10^{-6}.$$

The exact solution is an infinite-strength symmetric shock; see, e.g., [11, 33]. The initial computational domain is $\Omega^0 = (0, 1) \times (0, 1)$, with wall boundary conditions applied along the lines $x = 0$ and $y = 0$. We run the simulation until $t = 0.6$. As noted by Liska and Wendroff [33], many high-resolution schemes tend to produce highly oscillatory results or fail altogether.

We simulate the Noh problem on a uniform hexahedral mesh of size $30 \times 30 \times 30$, making this a challenging test case since the mesh is not aligned with the flow. The density profiles obtained with our DGH method at time $t = 0.6$ are shown in Fig. 4. Mesh entangling occurs for $\beta_c = 1.0$, but can be significantly reduced by using $\beta_c = 0.8$. Notably, the DGH method does not suffer from IDP violations even for highly distorted meshes. The shock is located on a circle of radius approximately 0.2, consistent with results from the literature [63, 64]. For $\beta_c = 0.8$, the shock wave front is sharply resolved and closely matches the one-dimensional cylindrical solution, while the method successfully preserves the radial symmetry of the flow.

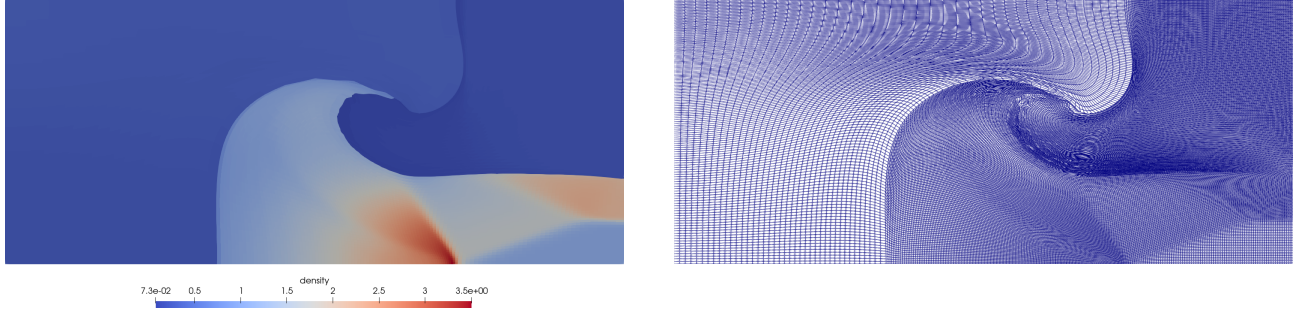


Figure 5: Triple point problem, density profile ϱ and mesh at time $t = 4.0$ using $E_h = 140 \times 80$ elements. Note that the solver outputs the subcells associated with each element. See fig.1 for details.

8.4. 2D Triple point problem

In our final numerical experiment, we consider the triple point problem [17], a three-state, two-material 2D Riemann problem known for generating vorticity. This problem is commonly used to test the robustness of Lagrangian methods in handling complex phenomena, such as significant vorticity, large shear, and interacting shocks [38]. The initial setup consists of three regions: a high-pressure region on the left and two low-pressure regions on the right. The high-pressure region drives a shock wave through the adjacent low-pressure regions, leading to the formation of a vortex at the triple point. The initial configuration on the computational domain $\Omega = (0, 7) \times (0, 3)$ is given by

$$(\varrho^0, \mathbf{u}^0, p^0)^\top = \begin{cases} (1, 0, 1)^\top & \text{if } y \leq 1, \\ (0.1, 0, 0.1)^\top & \text{if } x \geq 1, y \geq 1.5, \\ (1, 0, 0.1)^\top & \text{if } x \geq 1, y < 1.5. \end{cases}$$

We set $\gamma = 1.5$ in the initial left and upper regions, and $\gamma = 1.4$ in the lower region.

We run the simulation using a single-element-thick slab with $E_h = 140 \times 80$ uniform elements and $\beta_c = 0.8$. The density profile and computational mesh at $t = 4.0$ are presented in Fig. 5. The solution demonstrates the ability of our DGH method to accurately capture sharp interfaces between the interacting shock waves. The triple point region is accurately resolved, clearly displaying the physical vorticity.

9. Conclusions

We discussed a new way to limit Lagrangian DG discretizations of the Euler equations. The proposed scheme is second-order accurate while remaining fully conservative and GCL consistent. Following FCT design principles, we combine a property-preserving low-order scheme with a high-order target discretization. By utilizing a nodal version of Zalesak's limiting strategy, we achieve optimal convergence rates and effectively suppress spurious oscillations globally. To address local oscillations, we employ the clip-and-scale slope limiter, which yields a locally bound-preserving conservative approximation. Our analysis highlights the importance of using second-order time integrators to maintain

GCL consistency. Additionally, it is worthwhile to explore FCT-type limiters for systems, where correction factors are computed individually for each field, as opposed to the scalar limiting approach we used. Extending our approach to higher-order finite elements and curvilinear elements offers another potential avenue for further investigation.

Acknowledgments

Nathaniel Morgan and Jacob Moore gratefully acknowledge the funding from the Laboratory Directed Research and Development (LDRD) program at Los Alamos National Laboratory (LANL) under project number 20230028DR. The Los Alamos unlimited release number is LA-UR-25-30128. LANL is operated by Triad National Security, LLC for the U.S. Department of Energy's National Nuclear Security Administration under contract number 89233218CNA000001.

Appendix. GCL consistency for triangles and quadrilaterals.

We consider a triangle with vertices $\mathbf{x}_p = (x_p, y_p)$, $p \in \mathcal{P}(c)$, numbered counterclockwise. Each node moves with a constant velocity $\mathbf{u}_p = (u_p^x, u_p^y)^\top$ over the time interval $[t^n, t^{n+1}]$. The nodal coordinates are defined as functions of time:

$$x_p(t) = x_p^n + u_p^x(t - t^n), \quad y_p(t) = y_p^n + u_p^y(t - t^n), \quad t \in [t^n, t^{n+1}]. \quad (.1)$$

Inserting $t = t^{n+1}$ into (.1) yields

$$x_p^{n+1} = x_p^n + \Delta t u_p^x, \quad y_p^{n+1} = y_p^n + \Delta t u_p^y.$$

The algebraic volume $\tilde{v}_c$ at time t^{n+1} can be computed using the discretized GCL

$$\tilde{v}_c^{n+1} = \tilde{v}_c^n + \Delta t \sum_{p \in \mathcal{P}(c)} \sum_{s_f \in \mathcal{SF}(c,p)} a_{cps_f} \mathbf{u}_p \cdot \mathbf{n}_{cps_f} = \tilde{v}_c^n + \sum_{p \in \mathcal{P}(c)} \sum_{s_f \in \mathcal{SF}(c,p)} a_{cps_f} (\mathbf{x}_p^{n+1} - \mathbf{x}_p^n) \cdot \mathbf{n}_{cps_f}. \quad (.2)$$

The change in algebraic volume follows as

$$\begin{aligned} \tilde{v}_c^{n+1} - \tilde{v}_c^n &= \sum_{p \in \mathcal{P}(c)} \sum_{s_f \in \mathcal{SF}(c,p)} a_{cps_f} (\mathbf{x}_p^{n+1} - \mathbf{x}_p^n) \cdot \mathbf{n}_{cps_f} \\ &= \frac{1}{2} [(x_1^{n+1} - x_1^n)(y_1^n - y_3^n) + (y_1^{n+1} - y_1^n)(x_3^n - x_1^n) + (x_1^{n+1} - x_1^n)(y_2^n - y_1^n) + (y_1^{n+1} - y_1^n)(x_1^n - x_2^n) \\ &\quad + (x_2^{n+1} - x_2^n)(y_2^n - y_1^n) + (y_2^{n+1} - y_2^n)(x_1^n - x_2^n) + (x_2^{n+1} - x_2^n)(y_3^n - y_2^n) + (y_2^{n+1} - y_2^n)(x_2^n - x_3^n) \\ &\quad + (x_3^{n+1} - x_3^n)(y_3^n - y_2^n) + (y_3^{n+1} - y_3^n)(x_2^n - x_3^n) + (x_3^{n+1} - x_3^n)(y_1^n - y_3^n) + (y_3^{n+1} - y_3^n)(x_3^n - x_1^n)] \\ &= \frac{1}{2} [(x_1^{n+1} - x_1^n)(y_2^n - y_3^n) + (y_1^{n+1} - y_1^n)(x_3^n - x_2^n) + (x_2^{n+1} - x_2^n)(y_3^n - y_1^n) + (y_2^{n+1} - y_2^n)(x_1^n - x_3^n) \\ &\quad + (x_3^{n+1} - x_3^n)(y_1^n - y_2^n) + (y_3^{n+1} - y_3^n)(x_2^n - x_1^n)]. \end{aligned}$$

The geometric volume v_c of the triangle is

$$v_c = \frac{1}{2} \det \begin{pmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{pmatrix} = \frac{1}{2} (x_1 y_2 - y_1 x_2 + y_1 x_3 - x_1 y_3 + x_2 y_3 - x_3 y_2).$$

The change in geometric volume can be written as [43]

$$v_c^{n+1} - v_c^n = \sum_{p=1}^3 u_p^x \int_{t^n}^{t^{n+1}} \frac{\partial v_c}{\partial x_p} dt + \sum_{p=1}^3 u_p^y \int_{t^n}^{t^{n+1}} \frac{\partial v_c}{\partial y_p} dt.$$

We have

$$\begin{aligned} \frac{\partial v_c}{\partial x_1} &= \frac{1}{2} (y_2 - y_3), & \frac{\partial v_c}{\partial x_2} &= \frac{1}{2} (y_3 - y_1), & \frac{\partial v_c}{\partial x_3} &= \frac{1}{2} (y_1 - y_2), \\ \frac{\partial v_c}{\partial y_1} &= \frac{1}{2} (x_3 - x_2), & \frac{\partial v_c}{\partial y_2} &= \frac{1}{2} (x_1 - x_3), & \frac{\partial v_c}{\partial y_3} &= \frac{1}{2} (x_2 - x_1). \end{aligned}$$

Thus, we obtain

$$\begin{aligned} v_c^{n+1} - v_c^n &= \frac{x_1^{n+1} - x_1^n}{\Delta t} \int_{t^n}^{t^{n+1}} \frac{1}{2} (y_2 - y_3) dt + \frac{x_2^{n+1} - x_2^n}{\Delta t} \int_{t^n}^{t^{n+1}} \frac{1}{2} (y_3 - y_1) dt \\ &+ \frac{x_3^{n+1} - x_3^n}{\Delta t} \int_{t^n}^{t^{n+1}} \frac{1}{2} (y_1 - y_2) dt + \frac{y_1^{n+1} - y_1^n}{\Delta t} \int_{t^n}^{t^{n+1}} \frac{1}{2} (x_3 - x_2) dt \\ &+ \frac{y_2^{n+1} - y_2^n}{\Delta t} \int_{t^n}^{t^{n+1}} \frac{1}{2} (x_1 - x_3) dt + \frac{y_3^{n+1} - y_3^n}{\Delta t} \int_{t^n}^{t^{n+1}} \frac{1}{2} (x_2 - x_1) dt. \end{aligned}$$

By comparing the coefficients of $x_p^{n+1} - x_p^n$ and $y_p^{n+1} - y_p^n$ in the expressions for the algebraic and geometric volume changes, we need

$$\int_{t^n}^{t^{n+1}} \frac{1}{2} (x_p - x_q) dt = \frac{1}{2} \Delta t (x_p - x_q), \quad \int_{t^n}^{t^{n+1}} \frac{1}{2} (y_p - y_q) dt = \frac{1}{2} \Delta t (y_p - y_q)$$

to achieve GCL consistency. However, we have

$$\begin{aligned} \int_{t^n}^{t^{n+1}} \frac{1}{2} (x_p - x_q) dt &= \frac{1}{2} \Delta t (x_p - x_q) + \frac{1}{4} (\Delta t)^2 (u_p^x - u_q^x) \\ &= \frac{1}{2} \Delta t (x_p - x_q) + \frac{1}{4} \Delta t (x_p^{n+1} - x_p^n - x_q^{n+1} + x_q^n), \\ \int_{t^n}^{t^{n+1}} \frac{1}{2} (y_p - y_q) dt &= \frac{1}{2} \Delta t (y_p - y_q) + \frac{1}{4} (\Delta t)^2 (u_p^y - u_q^y) \\ &= \frac{1}{2} \Delta t (y_p - y_q) + \frac{1}{4} \Delta t (y_p^{n+1} - y_p^n - y_q^{n+1} + y_q^n). \end{aligned} \tag{.3}$$

GCL consistency is achieved only if the (sum of the) additional terms in (.3) vanish. Specifically, for first-order time integration, GCL consistency is ensured only for simple translations of the triangle. The additional terms in (.3) represent the consistency error of the forward Euler method. Therefore, this analysis confirms that second-order time accuracy is necessary to achieve GCL consistency.

Remark. The GCL analysis for triangles can be directly extended to quadrilaterals with straight edges. For such quadrilaterals, the geometric volume is given by

$$v_c = \frac{1}{2} \det \begin{pmatrix} 1 & 0 & x_1 & y_1 \\ 1 & 1 & x_2 & y_2 \\ 1 & 0 & x_3 & y_3 \\ 1 & 1 & x_4 & y_4 \end{pmatrix} = \frac{1}{2} (x_1 y_2 - x_1 y_4 - x_2 y_1 + x_2 y_3 - x_3 y_2 + x_3 y_4 + x_4 y_1 - x_4 y_3).$$

Comparing this geometric volume v_c with the algebraic volume $\tilde{v}_c$ computed by (.2) demonstrates once again the need for using a second-order accurate time integrator to maintain GCL consistency.

References

- [1] Frank L. Addessio, John R. Baumgardner, John K. Dukowicz, Norman L. Johnson, Bryan A. Kashiwa, Rick M. Rauenzahn, and Charles Zemach. CAVEAT: A computer code for fluid dynamics problems with large distortion and internal slip. Technical Report LA-10613-MS, Los Alamos National Laboratory, 1986.
- [2] Robert W. Anderson, Veselin A. Dobrev, Tzanio V. Kolev, Dmitri Kuzmin, Manuel Quezada de Luna, Robert N. Rieben, and Vladimir Z. Tomov. High-order local maximum principle preserving (MPP) discontinuous Galerkin finite element method for the transport equation. *J. Comput. Phys.*, 334:102–124, 2017.
- [3] Jay P. Boris and David L. Book. Flux-corrected transport. I. SHASTA, a fluid transport algorithm that works. *J. Comput. Phys.*, 11(1):38–69, 1973.
- [4] Donald E. Burton. Multidimensional discretization of conservation laws for unstructured polyhedral grids. Technical Report UCRL-JC-118306, Lawrence Livermore National Laboratory, 1994.
- [5] Donald E. Burton, Theodore C. Carney, Nathaniel R. Morgan, Shiv K. Sambasivan, and Mikhail J. Shashkov. A cell-centered Lagrangian Godunov-like method for solid dynamics. *Comput. Fluids*, 83:33–47, 2013.
- [6] Gilles Carré, Stéphane Del Pino, Bruno Després, and Emmanuel Labourasse. A cell-centered Lagrangian hydrodynamics scheme on general unstructured meshes in arbitrary dimension. *J. Comput. Phys.*, 228(14):5160–5183, 2009.
- [7] Randy B. Christensen. Godunov methods on a staggered mesh-an improved artificial viscosity. *Preprint UCRL-JC-105269 (Lawrence Livermore Nat. Lab., Livermore, 1990)*, 1990.

- [8] Bruno Després and Constant Mazeran. Lagrangian gas dynamics in two dimensions and Lagrangian systems. *Archive for Rational Mechanics and Analysis*, 178:327–372, 2005.
- [9] Veselin A. Dobrev, Truman E. Ellis, Tzanio V. Kolev, and Robert N. Rieben. Curvilinear finite elements for Lagrangian hydrodynamics. *Int. J. Numer. Meth. Fluids*, 65(11-12):1295–1310, 2011.
- [10] Veselin A. Dobrev, Truman E. Ellis, Tzanio V. Kolev, and Robert N. Rieben. High-order curvilinear finite elements for axisymmetric Lagrangian hydrodynamics. *Comput. Fluids*, 83:58–69, 2013.
- [11] Veselin A. Dobrev, Tzanio V. Kolev, Dmitri Kuzmin, Robert N. Rieben, and Vladimir Z. Tomov. Sequential limiting in continuous and discontinuous Galerkin methods for the Euler equations. *J. Comput. Phys.*, 356:372–390, 2018.
- [12] Veselin A. Dobrev, Tzanio V. Kolev, and Robert N. Rieben. High-order curvilinear finite element methods for Lagrangian hydrodynamics. *SIAM J. Sci. Comput.*, 34(5):B606–B641, 2012.
- [13] John K. Dukowicz. A general, non-iterative Riemann solver for Godunov’s method. *J. Comput. Phys.*, 61(1):119–137, 1985.
- [14] John K. Dukowicz and Bertrand J. A. Meltz. Vorticity errors in multidimensional Lagrangian codes. *J. Comput. Phys.*, 99(1):115–134, 1992.
- [15] Truman E. Ellis. High order finite elements for Lagrangian computational fluid dynamics. 2010.
- [16] Charbel Farhat, Philippe Geuzaine, and Céline Grandmont. The discrete geometric conservation law and the nonlinear stability of ALE schemes for the solution of flow problems on moving grids. *J. Comput. Phys.*, 174(2):669–694, 2001.
- [17] Stéphane Galera, Pierre-Henri Maire, and Jérôme Breil. A two-dimensional unstructured cell-centered multi-material ALE scheme using VOF interface reconstruction. *J. Comput. Phys.*, 229(16):5755–5787, 2010.
- [18] Gérard Gallice, Agnes Chan, Raphaël Loubère, and Pierre-Henri Maire. Entropy stable and positivity preserving Godunov-type schemes for multidimensional hyperbolic systems on unstructured grid. *J. Comput. Phys.*, 468:111493, 2022.
- [19] Gabriel Georges. *Development of a 3D cell-centered Lagrangian scheme for the numerical modeling of the gas dynamics and hyper-elasticity systems*. PhD thesis, Univ. Bordeaux, 2016.
- [20] Gabriel Georges, Jérôme Breil, and Pierre-Henri Maire. A 3D GCL compatible cell-centered Lagrangian scheme for solving gas dynamics equations. *J. Comput. Phys.*, 305:921–941, 2016.
- [21] Sergei K. Godunov. Reminiscences about difference schemes. *J. Comput. Phys.*, 153(1):6–25, 1999.

- [22] Sergei K. Godunov and Valeri Platonov. Résolution numérique des problèmes multidimensionnels de la dynamique des gaz. 1979.
- [23] Sigal Gottlieb, Chi-Wang Shu, and Eitan Tadmor. Strong stability-preserving high-order time discretization methods. *SIAM Rev.*, 43(1):89–112, 2001.
- [24] Zupeng Jia and Shudao Zhang. A new high-order discontinuous Galerkin spectral finite element method for Lagrangian gas dynamics in two-dimensions. *J. Comput. Phys.*, 230(7):2496–2522, 2011.
- [25] James R. Kamm and Frank X. Timmes. On efficient generation of numerically robust Sedov solutions. Technical Report LA-10613-MS, Los Alamos National Laboratory, 2007.
- [26] Valentin F. Kuropatenko. On difference methods for the equations of hydrodynamics. *Difference methods for solutions of problems of mathematical physics*, 1:287–303, 1967.
- [27] Dmitri Kuzmin. A new perspective on flux and slope limiting in discontinuous Galerkin methods for hyperbolic conservation laws. *Computer Methods Appl. Meth. Engrg.*, 373:113569, 2021.
- [28] Dmitri Kuzmin and Hennes Hajduk. *Property-Preserving Numerical Schemes for Conservation Laws*. World Scientific, 2023.
- [29] Dmitri Kuzmin and Nikita Klyushnev. Limiting and divergence cleaning for continuous finite element discretizations of the MHD equations. *J. Comput. Phys.*, 407:109230, 2020.
- [30] Dmitri Kuzmin, Rainald Löhner, and Stefan Turek. *Flux-Corrected Transport*. Springer, 2 edition, 2012.
- [31] Dmitri Kuzmin and Stefan Turek. Flux correction tools for finite elements. *J. Comput. Phys.*, 175(2):525–558, 2002.
- [32] Evan J. Lieberman, Nathaniel R. Morgan, Darby J. Luscher, and Donald E. Burton. A higher-order Lagrangian discontinuous Galerkin hydrodynamic method for elastic–plastic flows. *Comput. Math. Appl.*, 78(2):318–334, 2019.
- [33] Richard Liska and Burton Wendroff. Comparison of several difference schemes on 1D and 2D test problems for the Euler equations. *SIAM J. Sci. Comput.*, 25(3):995–1017, 2003.
- [34] Xiaodong Liu, Nathaniel R. Morgan, and Donald E. Burton. A Lagrangian discontinuous Galerkin hydrodynamic method. *Comput. Fluids*, 163:68–85, 2018.
- [35] Xiaodong Liu, Nathaniel R. Morgan, and Donald E. Burton. A high-order Lagrangian discontinuous Galerkin hydrodynamic method for quadratic cells using a subcell mesh stabilization scheme. *J. Comput. Phys.*, 386:110–157, 2019.

- [36] Christoph Lohmann, Dmitri Kuzmin, John N. Shadid, and Sibusiso Mabuza. Flux-corrected transport algorithms for continuous Galerkin methods based on high order Bernstein finite elements. *J. Comput. Phys.*, 344:151–186, 2017.
- [37] Rainald Löhner, Kenneth Morgan, Jaime Peraire, and Mehdi Vahdati. Finite element flux-corrected transport (FEM-FCT) for the Euler and Navier–Stokes equations. *Int. J. Numer. Meth. Fluids*, 7(10):1093–1109, 1987.
- [38] Raphaël Loubère. Validation test case suite for compressible hydrodynamics computation. *Unpublished notes, Los Alamos National Laboratory*, 39, 2005.
- [39] Raphaël Loubère, Pierre-Henri Maire, and Pavel Váchal. Formulation of a staggered two-dimensional Lagrangian scheme by means of cell-centered approximate Riemann solver. In *Numerical Mathematics and Advanced Applications 2009: Proceedings of ENUMATH 2009, the 8th European Conference on Numerical Mathematics and Advanced Applications, Uppsala, July 2009*, pages 617–625. Springer, 2010.
- [40] Raphaël Loubère, Pierre-Henri Maire, and Pavel Váchal. A second-order compatible staggered Lagrangian hydrodynamics scheme using a cell-centered multidimensional approximate Riemann solver. *Procedia Computer Science*, 1(1):1931–1939, 2010.
- [41] Raphaël Loubère, Pierre-Henri Maire, and Pavel Váchal. 3D staggered Lagrangian hydrodynamics scheme with cell-centered Riemann solver-based artificial viscosity. *Int. J. Numer. Meth. Fluids*, 72(1):22–42, 2013.
- [42] Raphaël Loubère, Jean Ovardia, and Rémi Abgrall. A Lagrangian discontinuous Galerkin-type method on unstructured meshes to solve hydrodynamics problems. *Int. J. Numer. Meth. Fluids*, 44(6):645–663, 2004.
- [43] Raphaël Loubère, Mikhail J. Shashkov, and Burton Wendroff. Volume consistency in a staggered grid Lagrangian hydrodynamics scheme. *J. Comput. Phys.*, 227(8):3731–3737, 2008.
- [44] Pierre-Henri Maire. A high-order cell-centered Lagrangian scheme for two-dimensional compressible fluid flows on unstructured meshes. *J. Comput. Phys.*, 228(7):2391–2425, 2009.
- [45] Pierre-Henri Maire, Rémi Abgrall, Jérôme Breil, and Jean Ovardia. A cell-centered Lagrangian scheme for two-dimensional compressible flow problems. *SIAM J. Sci. Comput.*, 29(4):1781–1824, 2007.
- [46] Pierre-Henri Maire and Jérôme Breil. A second-order cell-centered Lagrangian scheme for two-dimensional compressible flow problems. *Int. J. Numer. Meth. Fluids*, 56(8):1417–1423, 2008.
- [47] Pierre-Henri Maire, Raphaël Loubère, and Pavel Váchal. Staggered Lagrangian discretization based on cell-centered Riemann solver and associated hydrodynamics scheme. *Commun. Comput. Phys.*, 10(4):940–978, 2011.

- [48] Jacob L. Moore. *Elements: A unified framework for supporting low and high order numerical methods for multi-physics material dynamics simulations*. PhD thesis, Mississippi State University, 2021.
- [49] Nathaniel R. Morgan. A dissipation model for staggered grid Lagrangian hydrodynamics. *Comput. Fluids*, 83:48–57, 2013.
- [50] Nathaniel R. Morgan. On a 3D Lagrangian nodal discontinuous Galerkin method. *MultiMat*, 2022.
- [51] Nathaniel R. Morgan and Billy J. Archer. On the origins of Lagrangian hydrodynamic methods. *Nuclear Technology*, 207(sup1):S147–S175, 2021.
- [52] Nathaniel R. Morgan, Mark A. Kenamond, Donald E. Burton, Theodore C. Carney, and Daniel J. Ingraham. An approach for treating contact surfaces in Lagrangian cell-centered hydrodynamics. *J. Comput. Phys.*, 250:527–554, 2013.
- [53] Nathaniel R. Morgan, Konstantin N. Lipnikov, Donald E. Burton, and Mark A. Kenamond. A Lagrangian staggered grid Godunov-like approach for hydrodynamics. *J. Comput. Phys.*, 259:568–597, 2014.
- [54] Nathaniel R. Morgan, Xiaodong Liu, and Donald E. Burton. Reducing spurious mesh motion in Lagrangian finite volume and discontinuous Galerkin hydrodynamic methods. *J. Comput. Phys.*, 372:35–61, 2018.
- [55] Nathaniel R. Morgan, Jacob I. Waltz, Donald E. Burton, Marc R. Charest, Thomas R. Canfield, and John G. Wohlbiel. A Godunov-like point-centered essentially Lagrangian hydrodynamic approach. *J. Comput. Phys.*, 281:614–652, 2015.
- [56] Fabio Nobile and Luca Formaggia. A stability analysis for the arbitrary Lagrangian Eulerian formulation with finite elements. *East-West Journal of Numerical Mathematics*, 7(2):105–132, 1999.
- [57] William F. Noh. Errors for calculations of strong shocks using an artificial viscosity and an artificial heat flux. *J. Comput. Phys.*, 72(1):78–120, 1987.
- [58] Clark Pederson, Bart Brown, and Nathaniel R. Morgan. The Sedov blast wave as a radial piston verification test. *J. Verif. Valid. Uncert.*, 1(3):031001, 2016.
- [59] Guglielmo Scovazzi, Mark A. Christon, Thomas J.R. Hughes, and John N. Shadid. Stabilized shock hydrodynamics: I. A Lagrangian method. *Computer Methods Appl. Meth. Engrg.*, 196(4-6):923–966, 2007.
- [60] Guglielmo Scovazzi, John N. Shadid, Edward Love, and William J. Rider. A conservative nodal variational multiscale method for Lagrangian shock hydrodynamics. *Computer Methods Appl. Meth. Engrg.*, 199(49-52):3059–3100, 2010.

- [61] Leonid I. Sedov. *Similarity and Dimensional Methods in Mechanics*. Academic Press, 1959.
- [62] Geoffrey I. Taylor and Albert E. Green. Mechanism of the production of small eddies from large ones. *Proceedings of the Royal Society of London. Series A-Mathematical and Physical Sciences*, 158(895):499–521, 1937.
- [63] François Vilar. Cell-centered discontinuous Galerkin discretization for two-dimensional Lagrangian hydrodynamics. *Comput. Fluids*, 64:64–73, 2012.
- [64] François Vilar, Pierre-Henri Maire, and Rémi Abgrall. A discontinuous Galerkin discretization for solving the two-dimensional gas dynamics equations written under total Lagrangian formulation on general unstructured grids. *J. Comput. Phys.*, 276:188–234, 2014.
- [65] John von Neumann and Robert D. Richtmyer. A method for the numerical calculation of hydrodynamic shocks. *J. Appl. Phys.*, 21(3):232–237, 1950.
- [66] Mark L. Wilkins. Use of artificial viscosity in multidimensional fluid dynamic calculations. *J. Comput. Phys.*, 36(3):281–303, 1980.
- [67] Mark L. Wilkins et al. *Calculation of elastic-plastic flow*. University of California, Ernest L. Lawrence Radiation Laboratory, 1963.
- [68] Steven T. Zalesak. Fully multidimensional flux-corrected transport algorithms for fluids. *J. Comput. Phys.*, 31(3):335–362, 1979.
- [69] Steven T. Zalesak. The design of flux-corrected transport (FCT) algorithms for structured grids. In Dmitri Kuzmin, Rainald Löhner, and Stefan Turek, editors, *Flux-Corrected Transport*, pages 29–78. Springer, 1 edition, 2005.