

# A nearly-conservative, high-order, forward Lagrange–Galerkin method for the resolution of compressible flows on unstructured triangular meshes

Manuel Colera<sup>a,\*</sup>, Jaime Carpio<sup>a</sup>, Rodolfo Bermejo<sup>b</sup>

<sup>a</sup>*Departamento de Ingeniería Energética, E.T.S. Ingenieros Industriales, Universidad Politécnica de Madrid, C/ José Gutiérrez Abascal 2, 28006 Madrid, Spain*

<sup>b</sup>*Departamento de Matemática Aplicada a la Ingeniería Industrial, E.T.S. Ingenieros Industriales, Universidad Politécnica de Madrid, C/ José Gutiérrez Abascal 2, 28006 Madrid, Spain*

---

## Abstract

In this work, we present a novel Lagrange–Galerkin method for the resolution of compressible and inviscid flows. The scheme considers: (i) high-order continuous space discretizations on unstructured triangular meshes, (ii) high-order implicit–explicit Runge–Kutta schemes for the time discretization, (iii) conservation of mass, momentum and total energy, as long as some integrals in the formulation are computed exactly, and (iv) subgrid-stabilization and discontinuity-capturing operators based on Brenner’s model [*Physica A: Statistical Mechanics and its Applications*, 370 (2006), pp. 190–224] for viscous flows. The method has been tested on several benchmark problems using a fourth-order time-marching formula and up to fifth-order continuous finite elements, yielding the expected results both for smooth and discontinuous solutions.

**Keywords:** finite element method, Lagrangian–Eulerian method, compressible flows, high-order methods, triangular meshes, discontinuity-capturing

---

## 1. Introduction

The development of finite element methods for the computation of compressible flows is an extensive topic of research nowadays due to their many applications in industry and science, e.g., aircraft design [1], turbomachinery [2] and heat transfer [3].

Among the finite element methods found in the literature, we can first distinguish the so-called pure Lagrangian methods, in which the computational mesh is moved forward-in-time with the local fluid velocity. As advantages with respect to fixed-mesh (Eulerian) methods, the convective terms of the equations are naturally stabilized, the CFL condition is less restrictive and the shocks and free surfaces are easily tracked. However, these methods suffer from mesh distortion problems under the presence of large gradients in the velocity field. Arbitrary Lagrangian–Eulerian (ALE) methods [4] alleviate this drawback either by smoothing the mesh and remapping the solution –Lagrangian+remap type– or by directly moving the mesh with a smoother velocity field –monolithic or direct type. As examples of Lagrangian and ALE methods for compressible flows, we mention the staggered grid method of Dobrev et al. [5, 6], the nodal methods of Scovazzi and co-workers [7–12], Guermond et al. [13–15] and Cremonesi and Frangi [16], the ADER-DG method of Boscheri and Dumbser [17], and the staggered grid residual distribution method of Abgrall et al. [18, 19].

Other families of methods which are somehow similar in nature to ALE schemes, but are often considered separately, are the so-called semi-Lagrangian [20] and Lagrange–Galerkin methods [21–23]. In these

---

\*Corresponding author

Email addresses: m.colera@upm.es (Manuel Colera), jaime.carpio@upm.es (Jaime Carpio), rbermejo@etsii.upm.es (Rodolfo Bermejo)

algorithms, the material derivatives are discretized (typically) backward-in-time along the fluid trajectories. Then, the information of some of the previously computed solutions is transferred from a background reference mesh to a backward-in-time displaced mesh either by interpolation –semi-Lagrangian methods– or by more accurate  $L^2$  projections [24] –Lagrange–Galerkin methods. Hence, mesh distortion problems are avoided and, as with pure Lagrangian methods, the convective terms in the equations are naturally stabilized.

Most part of the Lagrange–Galerkin methods available in the literature lack conservation properties and are second-order accurate in time. However, they can be made conservative through the so-called *weak Lagrange–Galerkin* formulation [21], which is of great help to adequately capture and track discontinuities in the solution. This formulation was first introduced by Benqué et al. [25, 26] and employed later by Giraldo [27, 28], Kaazempur-Mofrad et al. [29, 30] and Colera et al. [31, 32], and achieves conservation by enforcing that the weighting functions be constant along the trajectories of the fluid particles. Thus, it is somehow similar to that of some versions of the Eulerian–Lagrangian localized adjoint method (ELLAM) [33, 34] and to that employed in some Lagrangian methods such as [6]. Furthermore, some authors have proposed to integrate the fluid trajectories forward-in-time [32, 35–38] –as usual in ALE methods– to allow for an efficient and simple implementation of higher-order time-marching schemes.

Though Lagrange–Galerkin methods have been successfully employed for the resolution of many interesting problems such as the incompressible Navier–Stokes equations [22, 39–41], the shallow water equations [27, 28, 42] and combustion [43, 44], we are not aware of their application to compressible flows.

It is interesting to note that forward Lagrange–Galerkin schemes and ALE methods of Lagrangian+remap type are very similar. The main difference is that, in the former schemes, the solution is remapped onto a fixed reference mesh via  $L^2$  projections, instead of onto a smoothed Lagrangian mesh via some remapping techniques [45, 46]. Therefore, as also remarked by [38], it seems that pure Lagrangian and semi-Lagrangian methods, which were quite different in nature at their origins, are evolving towards the same point. Due to this similitude, and also for simplicity, hereafter we shall consider forward Lagrange–Galerkin methods as another class of ALE methods.

In this work, we propose a novel forward Lagrange–Galerkin scheme for the resolution of two-dimensional, compressible and inviscid flows, which is an extension of our previous work [32] for scalar hyperbolic conservation laws. To the best of our knowledge, it is the first Lagrangian-type method in the literature which considers, at the same time, (i) high-order, continuous space discretizations on unstructured triangular meshes, (ii) high-order time discretizations, and (iii) conservation of mass, momentum and total energy irrespective of the employed time-marching scheme, as long as some integrals in the formulation are computed exactly.

The motivation for unstructured triangular meshes comes from the fact that, as pointed by [11, 16, 42], these are more amenable for automatic mesh generation and remeshing than quadrilateral ones, and also allow for an easier coupling with state-of-the-art codes for electromagnetism and other physics. On the other hand, high-order elements are able to capture shocks within a smaller number of elements [47] and have better scaling properties in parallel computations [48]. Furthermore, continuous space discretizations lead to a smaller number of degrees of freedom and thus require less memory than the widely employed discontinuous Galerkin approximations [49, 50]. Finally, it is clear that preservation of mass, momentum and total energy and the possibility to employ high-order time discretizations are also desirable properties.

Another interesting feature of the proposed method is the usage of Brenner’s model [51] to regularize the Euler equations instead of the more common Navier–Stokes model, following, e.g. [13, 52, 53]. In contrast to the latter, Brenner’s regularization satisfies entropy principles and positivity of the density and the internal energy [54], and provides better numerical results [52]. Starting from this model, we extend the subgrid stabilization techniques in [55, 56] to remove acoustic instabilities and develop a residual-based operator to detect and capture the discontinuities in the solution.

The remainder of the paper is structured as follows. Section 2 introduces the weak formulation for the continuous problem and explains its numerical discretization. In Section 3, some numerical experiments are performed to address the accuracy and efficiency of the proposed method. Finally, Section 4 discusses some conclusions and possible future works.

## 2. Formulation and discretization

In this work, we focus on the Euler equations for compressible and inviscid flows,

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho v_j)}{\partial x_j} = 0, \quad (1)$$

$$\frac{\partial m_i}{\partial t} + \frac{\partial (m_i v_j)}{\partial x_j} = -\frac{\partial p}{\partial x_i}, \quad (2)$$

$$\frac{\partial \mathcal{E}}{\partial t} + \frac{\partial (\mathcal{E} v_j)}{\partial x_j} = -\frac{\partial (p v_j)}{\partial x_j}, \quad (3)$$

with  $t \in [0, T]$  the time,  $\mathbf{x} = [x_1, x_2] \in \Omega$  the position vector,  $\rho$  the density,  $\mathbf{v} = [v_1, v_2]$  the velocity,  $\mathbf{m} = \rho \mathbf{v}$  the momentum per unit of volume,  $p$  the pressure,  $\mathcal{E} = p/(\gamma - 1) + \rho v_i v_i / 2$  the total energy per unit of volume and  $\gamma$  the adiabatic constant of the gas. The fluid variables are functions of  $(\mathbf{x}, t)$  and Einstein's summation convention for tensor operations is adopted. The system (1)-(3) is subject to the initial conditions

$$\rho(\mathbf{x}, 0) = \rho^0(\mathbf{x}), \quad \mathbf{m}(\mathbf{x}, t) = \mathbf{m}^0(\mathbf{x}), \quad \mathcal{E}(\mathbf{x}, t) = \mathcal{E}^0(\mathbf{x}),$$

and to appropriate boundary conditions.

The conservative, weak Lagrange-Galerkin formulation of Eqs. (1)-(3) is [28, 31]:

$$\frac{d}{dt} \int_{\Omega_f} \rho \psi \, d\Omega = 0, \quad (4)$$

$$\frac{d}{dt} \int_{\Omega_f} m_i \psi \, d\Omega = \int_{\Omega_f} p \frac{\partial \psi}{\partial x_i} \, d\Omega - \oint_{\partial \Omega_f} p \psi n_i \, d\sigma, \quad (5)$$

$$\frac{d}{dt} \int_{\Omega_f} \mathcal{E} \psi \, d\Omega = \int_{\Omega_f} p v_j \frac{\partial \psi}{\partial x_j} \, d\Omega - \oint_{\partial \Omega_f} p v_j \psi n_j \, d\sigma, \quad (6)$$

where  $\Omega_f = \Omega_f(t) \subseteq \Omega$  is an arbitrary domain that moves with the velocity field –i.e., a *fluid domain*–,  $n_i$  is the outward normal vector to  $\partial \Omega_f$  and  $\psi(\mathbf{x}, t)$  is a continuous test function such that  $D\psi/Dt := \partial \psi / \partial t + v_i \partial \psi / \partial x_i = 0$ . Note that Eqs. (4)-(6) can be interpreted from a physical point of view as conservation equations for a *weighted mass*, a *weighted momentum* and a *weighted energy*, and that the classical conservation laws of mass, momentum and energy are a particular case in which  $\psi \equiv 1$ .

In what follows,  $(\cdot)^n$  refers to any variable evaluated at a given time  $t^n$ ,  $\partial_t \equiv \partial / \partial t$ ,  $\partial_i \equiv \partial / \partial x_i$  and

$$(u, v)_D = \int_D uv \, d\Omega, \quad \langle u, v \rangle_{\partial D} = \oint_{\partial D} uv \, d\sigma,$$

denote respectively the scalar product of two generic functions  $u$  and  $v$  over a generic domain  $D$  and over its boundary  $\partial D$ .

### 2.1. Space discretization

The space discretization employed here is very similar to that of our previous work [31]. Thus, for completeness, we provide here a brief description of it and refer the reader to this reference for further details.

Let  $\Omega_h$  be a polygonal domain that approximates the fixed domain  $\Omega$ . We consider an admissible decomposition  $\mathbb{T}_h$  of  $\Omega_h$  into isoparametric triangles, with the nodes distributed according to a Chebyshev grid of the second kind [57] to allow for stable high-order discretizations. The finite element space  $V_h$  associated with  $\mathbb{T}_h$  is the direct sum of a *large-scale space*,  $\bar{V}_h$ , and a *fine-scale space*,  $V'_h$ , –i.e.,  $V_h = \bar{V}_h \oplus V'_h$ –, so that any function  $\psi_h \in V_h$  is expressed as the sum

$$\psi_h = \bar{\psi}_h + \psi'_h, \quad (7)$$

of a *large-scale term*,  $\bar{\psi}_h \in \bar{V}_h$ , and a *fine-scale term*,  $\psi'_h \in V'_h$ . This is done with a view to employ subgrid stabilization techniques [55, 56, 58] as well as the discontinuity-capturing operator developed later.

The large-scale space is the well-known space spanned by the continuous Lagrange polynomials –of degree  $r$ – associated with the mesh nodes, i.e.,  $\bar{V}_h = P_r(\mathbb{T}_h)$ , with  $r$  the order of the space discretization. The fine-scale space is the part of the space spanned by the products of the standard bubble function with the basis functions of  $P_{r-1}(\mathbb{T}_h)$  that is not already contained in  $P_r(\mathbb{T}_h)$ , i.e.,  $V'_h = B_{r-1}(\mathbb{T}_h)$  –see also [58]. More specifically, let  $\widehat{K} := \{\widehat{\mathbf{x}} \in \mathbb{R}^2 : 0 \leq \widehat{x}_1 \leq 1, 0 \leq \widehat{x}_2 \leq 1 - \widehat{x}_1\}$  be the well-known reference triangle,  $\widehat{b} = 27(1 - \widehat{x}_1 - \widehat{x}_2)\widehat{x}_1\widehat{x}_2$  the bubble function,  $\widehat{\mathbf{x}}$  the coordinate vector in  $\widehat{K}$  and  $\widehat{P}_r(\widehat{K})$  the space of polynomials in  $\widehat{K}$  of maximal degree  $r$ . The standard isoparametric transformation from  $\widehat{K}$  to a given element  $K \in \mathbb{T}_h$  is

$$\mathbf{F}_K : \widehat{\mathbf{x}} \in \widehat{K} \longrightarrow \mathbf{x} \in K, \quad \mathbf{x} = \sum_{k=1}^{N_v} \mathbf{x}_{n_k} \widehat{L}_k(\widehat{\mathbf{x}}),$$

where  $N_v$  is the number of nodes of the element,  $n_k$  is the  $k$ th local node of  $K$  and  $\widehat{L}_k(\widehat{\mathbf{x}})$  is the Lagrange polynomial associated with  $n_k$ . Then,

$$\begin{aligned} \bar{V}_h = P_r(\mathbb{T}_h) &:= \{\psi \in C^0(\Omega_h) : \psi|_K = \widehat{p} \circ \mathbf{F}_K^{-1}, \widehat{p} \in \widehat{P}_r(\widehat{K}), \forall K \in \mathbb{T}_h\}, \\ V'_h = B_{r-1}(\mathbb{T}_h) &:= \{\psi \in C^0(\Omega_h) : \psi|_K = (\widehat{b}\widehat{p}) \circ \mathbf{F}_K^{-1}, \widehat{p} \in \widehat{P}_{r-1}(\widehat{K}), \forall K \in \mathbb{T}_h\} - P_r(\mathbb{T}_h), \end{aligned}$$

and  $V_h = P_r^{\text{bubble}}(\mathbb{T}_h) := P_r(\mathbb{T}_h) \oplus B_{r-1}(\mathbb{T}_h)$ . Recall that the functions in  $B_{r-1}(\mathbb{T}_h)$  cancel at the edges of the elements.

As it will be shown later, the reference mesh  $\Omega_h$  is to be displaced in a Lagrangian fashion from a reference time  $t^n$  to a generic time  $t \geq t^n$ . For that purpose, it is convenient to introduce some notation and to recall some important properties of the conservative Lagrange–Galerkin method. Let  $\chi_N(t)$  be the trajectory of the  $N$ th mesh node  $\mathbf{x}_N$ , which evolves according to the system

$$\begin{cases} \frac{d\chi_N(t)}{dt} = \mathbf{v}(\chi_N(t), t), & t \geq t^n, \\ \chi_N(t^n) = \mathbf{x}_N. \end{cases}$$

Assuming for the moment that the trajectories of the mesh nodes are known, then [32] (see also Fig. 1):

- (i) For each element  $K \in \mathbb{T}_h$ , an isoparametric, curvilinear, *convected element*  $\widetilde{K}_h(t)$  is defined from the convected nodes of  $K$ . That is,  $\widetilde{K}_h(t)$  is the image of the isoparametric transformation

$$\widetilde{\mathbf{F}}_K : \widehat{\mathbf{x}} \in \widehat{K}, t \in [t^n, \infty) \longrightarrow \mathbf{x} \in \widetilde{K}_h(t), \quad \mathbf{x} = \sum_{k=1}^{N_v} \chi_{n_k}(t) \widehat{L}_k(\widehat{\mathbf{x}}).$$

At the same time, the union of convected elements defines a *convected mesh*  $\widetilde{\mathbb{T}}_h(t)$  and a *convected domain*  $\widetilde{\Omega}_h(t)$ . We assume that the time step  $t - t^n$  is small enough to avoid distortion of the elements in  $\widetilde{\mathbb{T}}_h(t)$ .

- (ii) The *convected finite element space*  $\widetilde{V}_h(t) = P_r^{\text{bubble}}(\widetilde{\mathbb{T}}_h(t))$  is defined. Note that the latter is a standard polynomial–bubble space, although it is defined in the curvilinear, convected mesh  $\widetilde{\mathbb{T}}_h(t)$ . Also note that  $\widetilde{V}_h^n \equiv V_h$ .
- (iii) The position of a fluid particle initially –i.e., at  $t^n$ – at  $\mathbf{x}$  is approximated by  $\mathbf{X}_h(\mathbf{x}, t) = \widetilde{\mathbf{F}}_K(\mathbf{F}_K^{-1}(\mathbf{x}), t)$ .
- (iv) If  $\psi_h(\mathbf{x}, t)$  is a function such that  $D\psi_h/Dt = 0$ , then it is assumed that  $\psi_h(\mathbf{X}_h(\mathbf{x}, t), t) = \psi_h^n(\mathbf{x})$  or, mapping back to the reference element,

$$\psi_h|_{\widetilde{K}_h(t)}(\widetilde{\mathbf{F}}_K(\widehat{\mathbf{x}}, t), t) = \psi_h^n|_K(\mathbf{F}_K(\widehat{\mathbf{x}})), \quad \forall K \in \mathbb{T}_h.$$

That is,  $\psi_h^n(\mathbf{x})$  and  $\psi_h(\mathbf{x}, t)$  read the same in the reference element. As a consequence, if  $\psi_h^n(\mathbf{x})$  is the  $i$ th shape function of  $V_h$ , then  $\psi_h(\mathbf{x}, t)$  is the  $i$ th shape function of  $\widetilde{V}_h(t)$ . This is the most important property for the numerical discretization of the equations.

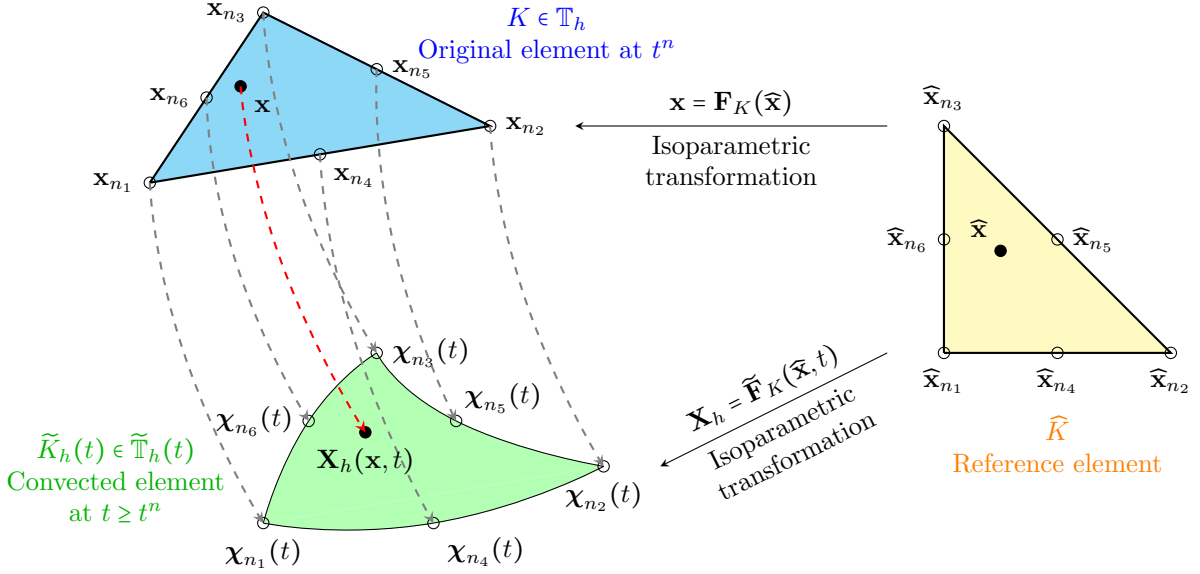


Figure 1: Scheme to illustrate the convection of the reference mesh with the velocity field for the case of quadratic elements. Assuming the forward-in-time trajectories of the mesh nodes (gray) are known, new isoparametric, curvilinear elements (green) are constructed. At the same time, the inner points of the elements are convected so that each  $\mathbf{x} \in K$  and  $\mathbf{X}_h(\mathbf{x}, t) \in \tilde{K}_h(t)$  (red trajectory) share the same natural coordinates  $\hat{\mathbf{x}}$  in the reference element  $\hat{K}$ .

The conservative variables  $\rho$ ,  $\mathbf{m}$  and  $\mathcal{E}$  are to be approximated by functions belonging to the finite element space  $\tilde{V}_h(t)$ . Two numerical approximations are to be considered. The first one, denoted by  $\mathbf{u}_* = [\rho_*, \mathbf{m}_*, \mathcal{E}_*] \in (\tilde{V}_h(t))^4$ , is the so-called *non-stabilized solution*, and is obtained by solving Eqs. (4)-(6) without any kind of stabilization. As it is well known,  $\mathbf{u}_*$  may present spurious oscillations due to the effect of acoustic perturbations [11] or to the presence of discontinuities. The second one, denoted by  $\mathbf{u}_h = [\rho_h, \mathbf{m}_h, \mathcal{E}_h] \in (\tilde{V}_h(t))^4$ , is the so-called *stabilized solution*, and is obtained by solving Eqs. (4)-(6) equipped with some stabilization operators –see Section 2.3. Whenever required (typically at quadrature points), numerical approximations to  $v_i$  and  $p$  are computed through the relations

$$v_{i\Box} = \frac{m_{i\Box}}{\rho_{\Box}}, \quad p_{\Box} = (\gamma - 1) \left( \mathcal{E}_{\Box} - \frac{m_{i\Box} m_{i\Box}}{2\rho_{\Box}} \right),$$

where the square symbol represents either  $h$  or  $*$ .

## 2.2. Time discretization

Let us assume that we dispose of a numerical solution  $\mathbf{u}_h^n \in (V_h)^4$  to Eqs. (1)-(3) at a given time  $t^n$ , and that we aim to compute the solution at the next time level  $t^{n+1}$ . To do so, we consider a fluid domain such that  $\Omega_f^n \equiv \Omega_h$  and integrate Eqs. (4)-(6) forward-in-time with an implicit–explicit Runge–Kutta method. The implicit (I) and explicit (E) parts of the method are described by the Butcher tableaux

$$\begin{array}{c|ccc|ccc}
 c^1 = 0 & 0 & & & 0 & & & \\
 c^2 & a_I^{2,1} & a_I^{2,2} & & a_E^{2,1} & & & \\
 c^3 & a_I^{3,1} & a_I^{3,2} & a_I^{3,3} & a_E^{3,1} & a_E^{3,2} & & \\
 \vdots & \vdots & & & \vdots & & \ddots & \\
 c^s & a_I^{s,1} & \dots & \dots & a_I^{s,s} & a_E^{s,1} & \dots & \dots & a_E^{s,s-1} \\
 \hline
 & b_I^1 & \dots & \dots & b_I^s & b_E^1 & \dots & \dots & b_E^{s-1} & b_E^s
 \end{array} . \tag{8}$$

Note that we consider methods with the same time splitting for both parts.

As usual, the first stage of the Runge–Kutta method is assumed to be equal to the solution at  $t^n$ , i.e.,  $(\cdot)^{[1]} = (\cdot)^n$  for any variable  $(\cdot)$ . In addition, we consider  $\mathbf{u}_*^{[1]} = \mathbf{u}_h^{[1]}$ . Then, for the stages  $l = 2, \dots, s$ , we set  $t^{[l]} = t^{[1]} + \Delta t c^l$ , with  $\Delta t = t^{n+1} - t^n$ , and perform the steps described next.

*Step 1 (convect mesh nodes).* Each node of the reference mesh  $\mathbb{T}_h$  is displaced with the explicit part of the method according to

$$\chi_N^{[l]} = \chi_N^{[1]} + \Delta t \sum_{k=1}^{l-1} a_E^{lk} \mathbf{v}_h^{[k]} \left( \chi_N^{[k]} \right), \quad N = 1, \dots, N_h. \quad (9)$$

As shown in the previous section, the convected nodes define a convected mesh  $\tilde{\mathbb{T}}_h^{[l]}$ , a convected domain  $\tilde{\Omega}_h^{[l]}$  and a convected finite element space  $\tilde{V}_h^{[l]} = P_r^{\text{bubble}} \left( \tilde{\mathbb{T}}_h^{[l]} \right)$ .

*Step 2 (compute non-stabilized solution).* We employ the explicit part of the method to find a non-stabilized solution  $\mathbf{u}_*^{[l]} \in \left( \tilde{V}_h^{[l]} \right)^4$  such that

$$\left( \rho_*^{[l]}, \psi_h^{[l]} \right)_{\tilde{\Omega}_h^{[l]}} = \left( \rho_h^{[1]}, \psi_h^{[1]} \right)_{\tilde{\Omega}_h^{[1]}}, \quad (10)$$

$$\left( m_{i*}^{[l]}, \psi_h^{[l]} \right)_{\tilde{\Omega}_h^{[l]}} = \left( m_{ih}^{[1]}, \psi_h^{[1]} \right)_{\tilde{\Omega}_h^{[1]}} + \Delta t \sum_{k=1}^{l-1} a_E^{lk} f_{m_i} \left( \rho_h^{[k]}, \mathbf{m}_h^{[k]}, \mathcal{E}_h^{[k]}, \psi_h^{[k]} \right)_{\tilde{\Omega}_h^{[k]}}, \quad (11)$$

$$\left( \mathcal{E}_*^{[l]}, \psi_h^{[l]} \right)_{\tilde{\Omega}_h^{[l]}} = \left( \mathcal{E}_h^{[1]}, \psi_h^{[1]} \right)_{\tilde{\Omega}_h^{[1]}} + \Delta t \sum_{k=1}^{l-1} a_E^{lk} f_{\mathcal{E}} \left( \rho_h^{[k]}, \mathbf{m}_h^{[k]}, \mathcal{E}_h^{[k]}, \psi_h^{[k]} \right)_{\tilde{\Omega}_h^{[k]}}, \quad (12)$$

for all  $\psi_h^{[l]} \in \tilde{V}_h^{[l]}$ , where

$$f_{m_i} \left( \rho_h^{[k]}, \mathbf{m}_h^{[k]}, \mathcal{E}_h^{[k]}, \psi_h^{[k]} \right)_{\tilde{\Omega}_h^{[k]}} := \left( p_h^{[k]}, \frac{\partial \psi_h^{[k]}}{\partial x_i} \right)_{\tilde{\Omega}_h^{[k]}} - \left\langle p_h^{[k]} n_i, \psi_h^{[k]} \right\rangle_{\partial \tilde{\Omega}_h^{[k]}}, \quad (13)$$

$$f_{\mathcal{E}} \left( \rho_h^{[k]}, \mathbf{m}_h^{[k]}, \mathcal{E}_h^{[k]}, \psi_h^{[k]} \right)_{\tilde{\Omega}_h^{[k]}} := \left( p_h^{[k]} v_j^{[k]}, \frac{\partial \psi_h^{[k]}}{\partial x_j} \right)_{\tilde{\Omega}_h^{[k]}} - \left\langle p_h^{[k]} v_j^{[k]} n_j, \psi_h^{[k]} \right\rangle_{\partial \tilde{\Omega}_h^{[k]}}. \quad (14)$$

We recall that, when  $\psi_h^{[l]}$  is the  $i$ th shape function of  $\tilde{V}_h^{[l]}$ , then  $\psi_h^{[k]}$  is the  $i$ th shape function of  $\tilde{V}_h^{[k]}$ . Therefore, when performing the numerical discretization, the terms  $f_{m_i}(\dots)$  and  $f_{\mathcal{E}}(\dots)$  can be readily computed by means of the convected meshes  $\tilde{\Omega}_h^{[k]}$  and lead to vectors that do not depend on the current Runge–Kutta stage  $l$ .

In addition, we are assuming pressure boundary conditions for simplicity, which are imposed weakly through the boundary integral in (13). Normal velocity conditions can be strongly imposed in the momentum equation (11) following the usual procedure –see, e.g. [32]–, although the densities have to be computed first via Eq. (10). Note that, in the case of pressure (resp. normal velocity) boundary conditions, the normal velocities (resp. pressures) have to be computed from the numerical solution in order to evaluate the boundary integral in (14).

Finally, it should be pointed that the left-hand side of the system defined by (10)–(12) is formed by four uncoupled mass matrices, and thus its resolution is straightforward.

*Step 3 (compute stabilized solution).* The non-stabilized solution  $\mathbf{u}_*^{[l]}$  is post-processed to detect spurious oscillations and to compute adequate stabilization operators of artificial-diffusion type –introduced in Section 2.3–, namely

$$\mathcal{S}_\rho \left( \rho_h^{[l]}, \psi_h^{[l]} \right)_{\tilde{\Omega}_h^{[l]}}, \quad \mathcal{S}_{m_i} \left( \rho_h^{[l]}, \mathbf{m}_h^{[l]}, \psi_h^{[l]} \right)_{\tilde{\Omega}_h^{[l]}}, \quad \mathcal{S}_{\mathcal{E}} \left( \rho_h^{[l]}, \mathbf{m}_h^{[l]}, \mathcal{E}_h^{[l]}, \psi_h^{[l]} \right)_{\tilde{\Omega}_h^{[l]}}.$$

We omit the dependence on  $\mathbf{u}_*^{[l]}$  in the definitions above for brevity. As we shall see, the operator  $\mathcal{S}_\rho$  is linear with respect to  $\rho_h^{[l]}$ , whereas  $\mathcal{S}_{m_i}$  and  $\mathcal{S}_\mathcal{E}$  are nonlinear with respect to  $\rho_h^{[l]}$ ,  $\mathbf{m}_h^{[l]}$  and  $\mathcal{E}_h^{[l]}$ . Since these operators are stiff, we now employ the implicit part of the Runge–Kutta method to march in time. More precisely, we first find  $\rho_h^{[l]} \in \tilde{V}_h^{[l]}$  such that

$$\left(\rho_h^{[l]}, \psi_h^{[l]}\right)_{\tilde{\Omega}_h^{[l]}} + \Delta t a_I^l \mathcal{S}_\rho \left(\rho_h^{[l]}, \psi_h^{[l]}\right)_{\tilde{\Omega}_h^{[l]}} = \left(\rho_h^{[1]}, \psi_h^{[1]}\right)_{\tilde{\Omega}_h^{[1]}} - \Delta t \sum_{k=1}^{l-1} a_I^{lk} \mathcal{S}_\rho \left(\rho_h^{[k]}, \psi_h^{[k]}\right)_{\tilde{\Omega}_h^{[k]}}, \quad \forall \psi_h^{[l]} \in \tilde{V}_h^{[l]}. \quad (15)$$

Then, we find  $m_{i_h}^{[l]} \in \tilde{V}_h^{[l]}$  and  $\mathcal{E}_h^{[l]} \in \tilde{V}_h^{[l]}$  such that

$$\left(m_{i_h}^{[l]}, \psi_h^{[l]}\right)_{\tilde{\Omega}_h^{[l]}} = b_{m_i}^{[l]}(\psi_h) + \Delta t a_I^l g_{m_i} \left(\rho_h^{[l]}, \mathbf{m}_h^{[l]}, \mathcal{E}_h^{[l]}, \psi_h^{[l]}\right)_{\tilde{\Omega}_h^{[l]}}, \quad (16)$$

$$\left(\mathcal{E}_h^{[l]}, \psi_h^{[l]}\right)_{\tilde{\Omega}_h^{[l]}} = b_{\mathcal{E}}^{[l]}(\psi_h) + \Delta t a_I^l g_{\mathcal{E}} \left(\rho_h^{[l]}, \mathbf{m}_h^{[l]}, \mathcal{E}_h^{[l]}, \psi_h^{[l]}\right)_{\tilde{\Omega}_h^{[l]}}, \quad (17)$$

for all  $\psi_h^{[l]} \in \tilde{V}_h^{[l]}$ , where

$$b_{m_i}^{[l]}(\psi_h) := \left(m_{i_h}^{[1]}, \psi_h^{[1]}\right)_{\tilde{\Omega}_h^{[1]}} + \Delta t \sum_{k=1}^{l-1} a_I^{lk} g_{m_i} \left(\rho_h^{[k]}, \mathbf{m}_h^{[k]}, \mathcal{E}_h^{[k]}, \psi_h^{[k]}\right)_{\tilde{\Omega}_h^{[k]}},$$

$$b_{\mathcal{E}}^{[l]}(\psi_h) := \left(\mathcal{E}_h^{[1]}, \psi_h^{[1]}\right)_{\tilde{\Omega}_h^{[1]}} + \Delta t \sum_{k=1}^{l-1} a_I^{lk} g_{\mathcal{E}} \left(\rho_h^{[k]}, \mathbf{m}_h^{[k]}, \mathcal{E}_h^{[k]}, \psi_h^{[k]}\right)_{\tilde{\Omega}_h^{[k]}},$$

$$g_{m_i} \left(\rho_h^{[k]}, \mathbf{m}_h^{[k]}, \mathcal{E}_h^{[k]}, \psi_h^{[k]}\right)_{\tilde{\Omega}_h^{[k]}} := f_{m_i} \left(\rho_h^{[k]}, \mathbf{m}_h^{[k]}, \mathcal{E}_h^{[k]}, \psi_h^{[k]}\right)_{\tilde{\Omega}_h^{[k]}} - \mathcal{S}_{m_i} \left(\rho_h^{[k]}, \mathbf{m}_h^{[k]}, \psi_h^{[k]}\right)_{\tilde{\Omega}_h^{[k]}}, \quad (18)$$

$$g_{\mathcal{E}} \left(\rho_h^{[k]}, \mathbf{m}_h^{[k]}, \mathcal{E}_h^{[k]}, \psi_h^{[k]}\right)_{\tilde{\Omega}_h^{[k]}} := f_{\mathcal{E}} \left(\rho_h^{[k]}, \mathbf{m}_h^{[k]}, \mathcal{E}_h^{[k]}, \psi_h^{[k]}\right)_{\tilde{\Omega}_h^{[k]}} - \mathcal{S}_{\mathcal{E}} \left(\rho_h^{[k]}, \mathbf{m}_h^{[k]}, \mathcal{E}_h^{[k]}, \psi_h^{[k]}\right)_{\tilde{\Omega}_h^{[k]}}. \quad (19)$$

When discretized, Eq. (15) leads to a linear system defined by a positive definite matrix and thus can readily be solved. However, Eqs. (16)–(17) lead to a coupled nonlinear system since both the left and the right-hand side terms depend on the unknowns  $m_{i_h}^{[l]}$  and  $\mathcal{E}_h^{[l]}$ . Its resolution will be detailed in Section 2.4.

After Step 3 is finished, we go back to Step 1 if  $l < s$ . Otherwise, all the internal stages  $l = 1, \dots, s$  of the Runge–Kutta method have been solved and the solution  $\tilde{\mathbf{u}}_h^{n+1} \in (\tilde{V}_h^{n+1})^4$  at  $t^{n+1}$  is computed through

$$\tilde{\chi}_N^{n+1} = \chi_N^{[1]} + \Delta t \sum_{k=1}^s b_E^k \mathbf{v}_h^{[k]} \left(\chi_N^{[k]}\right), \quad N = 1, \dots, N_h, \quad (20)$$

$$\left(\tilde{\rho}_h^{n+1}, \tilde{\psi}_h^{n+1}\right)_{\tilde{\Omega}_h^{n+1}} = \left(\rho_h^{[1]}, \psi_h^{[1]}\right)_{\tilde{\Omega}_h^{[1]}} - \Delta t \sum_{k=1}^s b_I^k \mathcal{S}_\rho \left(\rho_h^{[k]}, \psi_h^{[k]}\right)_{\tilde{\Omega}_h^{[k]}}, \quad (21)$$

$$\left(\tilde{m}_{i_h}^{n+1}, \tilde{\psi}_h^{n+1}\right)_{\tilde{\Omega}_h^{n+1}} = \left(m_{i_h}^{[1]}, \psi_h^{[1]}\right)_{\tilde{\Omega}_h^{[1]}} + \Delta t \sum_{k=1}^s b_I^k g_{m_i} \left(\rho_h^{[k]}, \mathbf{m}_h^{[k]}, \mathcal{E}_h^{[k]}, \psi_h^{[k]}\right)_{\tilde{\Omega}_h^{[k]}}, \quad (22)$$

$$\left(\tilde{\mathcal{E}}_h^{n+1}, \tilde{\psi}_h^{n+1}\right)_{\tilde{\Omega}_h^{n+1}} = \left(\mathcal{E}_h^{[1]}, \psi_h^{[1]}\right)_{\tilde{\Omega}_h^{[1]}} + \Delta t \sum_{k=1}^s b_I^k g_{\mathcal{E}} \left(\rho_h^{[k]}, \mathbf{m}_h^{[k]}, \mathcal{E}_h^{[k]}, \psi_h^{[k]}\right)_{\tilde{\Omega}_h^{[k]}}, \quad (23)$$

for all  $\tilde{\psi}_h^{n+1} \in \tilde{V}_h^{n+1}$ , where the tilde symbol is used to emphasize that the solution is defined at the convected mesh  $\tilde{\Omega}_h^{n+1}$ . This solution has to be transferred to the reference mesh  $\mathbb{T}_h$  in order to proceed again as shown above and compute the solution at  $t^{n+2}$ . To do so, we compute the  $L^2$  projection onto the reference mesh, that is, we find  $\mathbf{u}_h^{n+1} \in (V_h)^4$  such that

$$\left(\rho_h^{n+1}, \psi_h\right)_{\Omega_h} = \left(\tilde{\rho}_h^{n+1}, \psi_h\right)_{\Omega_h}, \quad (24)$$

$$\left(m_{i_h}^{n+1}, \psi_h\right)_{\Omega_h} = \left(\tilde{m}_{i_h}^{n+1}, \psi_h\right)_{\Omega_h}, \quad (25)$$

$$\left(\mathcal{E}_h^{n+1}, \psi_h\right)_{\Omega_h} = \left(\tilde{\mathcal{E}}_h^{n+1}, \psi_h\right)_{\Omega_h}, \quad (26)$$

for all  $\psi_h \in V_h$ .

As already discussed in [31, 32] (see also Fig. 2):

- (i) The right-hand side terms in (24)-(26) involve functions defined piecewise in different meshes. Thus, we approach their computation via high-order quadrature rules [59] and search-locate algorithms [31, 60, 61]. Mesh intersection techniques [62–64] and other remap algorithms [45, 46] could be employed as well.
- (ii) When there is flow leaving the reference domain, there is a part of  $\Omega_h$  that lies outside of  $\tilde{\Omega}_h^{n+1}$ . In those cases, we assume that the solution at  $\Omega_h \setminus \tilde{\Omega}_h^{n+1}$  equals to some known (incoming) flow conditions  $\mathbf{u}_{\text{in}}(\mathbf{x}, t^{n+1})$  in order to solve (24)-(26).
- (iii) If  $\tilde{\mathbf{u}}_h^{n+1}$  presents discontinuities,  $\mathbf{u}_h^{n+1}$  may suffer from spurious oscillations. However, these are not large because the sizes of the elements in  $\tilde{\mathbb{T}}_h^{n+1}$  and  $\mathbb{T}_h$  are of the same order; hence, if a discontinuity is well captured with former, it should also be well captured with the latter. In addition, the oscillations that may appear are to be removed by artificial diffusion when integrating from  $t^{n+1}$  to  $t^{n+2}$ . Nevertheless, it is more accurate to take  $\tilde{\mathbf{u}}_h^{n+1}$  as the solution at  $t^{n+1}$  and to consider Eqs. (24)-(26) as the first step for integrating from  $t^{n+1}$  to  $t^{n+2}$  rather than the last step for integrating from  $t^n$  to  $t^{n+1}$ .

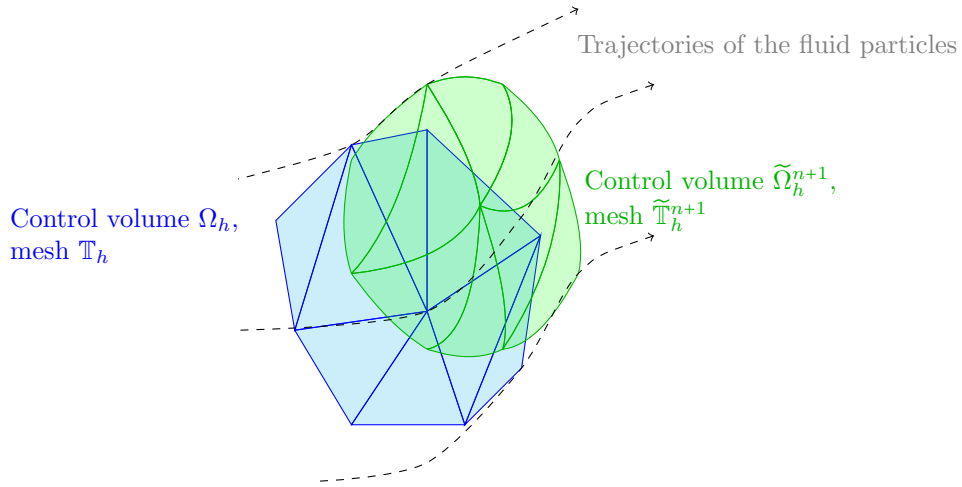


Figure 2: Scheme that illustrates the domains and meshes that appear in the formulation. The elements in  $\mathbb{T}_h$  are convected forward in time with the flow velocity field to form  $\tilde{\mathbb{T}}_h^{n+1}$ . The variables  $\rho_h^{n+1}$ ,  $m_h^{n+1}$ ,  $\mathcal{E}_h^{n+1}$  and  $\psi_h$  in Eqs. (24)-(26) are defined piecewise in  $\mathbb{T}_h$ , whereas  $\tilde{\rho}_h^{n+1}$ ,  $\tilde{m}_h^{n+1}$  and  $\tilde{\mathcal{E}}_h^{n+1}$  are defined piecewise in  $\tilde{\mathbb{T}}_h^{n+1}$ .

In the next sections we address the computation of the stabilization operators and the resolution of the nonlinear system (16)-(17).

### 2.3. Stabilization operators

#### 2.3.1. Brenner's model

The stabilization operators employed in this work are based on the model developed by Brenner [51] for compressible and viscous flows. In contrast to the classical Navier–Stokes model, Brenner's model takes into account for mass diffusion and for the transport of momentum and energy associated with the latter. This is achieved by replacing the velocity  $v_i$  in the convective derivative terms of Navier–Stokes equations by a *mass velocity*

$$(v_m)_i := v_i - \epsilon \frac{\partial_i \rho}{\rho},$$



with  $\epsilon$  the mass-diffusion coefficient. Hence, Brenner's model reads

$$\begin{aligned}\partial_t \rho + \partial_j (\rho v_j) &= \partial_j (\epsilon \partial_j \rho), \\ \partial_t m_i + \partial_j (m_i v_j) &= -\partial_i p + \partial_j \sigma_{ij} + \partial_j \left( \epsilon m_i \frac{\partial_j \rho}{\rho} \right), \\ \partial_t \mathcal{E} + \partial_j (\mathcal{E} v_j) &= -\partial_j (p v_j) + \partial_j (v_i \sigma_{ij}) + \partial_j \left( \frac{\kappa}{c_v} \partial_j e \right) + \partial_j \left( \epsilon \mathcal{E} \frac{\partial_j \rho}{\rho} \right),\end{aligned}$$

with  $e = p/\rho/(\gamma - 1)$  the specific internal energy,  $\kappa$  the conductivity coefficient,  $c_v$  the specific heat for constant volume,

$$\sigma_{ij} = \rho \nu \left[ \partial_j v_i + \partial_i v_j - \frac{2}{3} (\partial_k v_k) \delta_{ij} \right]$$

the viscous stress tensor,  $\delta_{ij}$  the Kronecker delta and  $\nu$  the kinematic viscosity.

Brenner originally developed his model from physical arguments only. However, its mathematical basis is sound as it has also been derived by Guermond and Popov [54] from purely mathematical arguments, such as the satisfaction of entropy inequalities and the positivity of density and energy –conditions that are not met by the classical Navier–Stokes regularization. The latter authors proved that the most robust regularization corresponds to the case  $\kappa/c_v = \rho\epsilon$ . In addition, Feireisl and Vasseur [65] have proved the existence of weak solutions to Brenner's model. Finally, Nazarov and Larcher [52] have demonstrated that this model yields better numerical results than the classical Navier–Stokes regularization, which is also in accordance with our own experience.

For ease of implementation, we rewrite Brenner's model as

$$\partial_t \rho + \partial_j (\rho v_j) = \partial_j (\epsilon \partial_j \rho), \quad (27)$$

$$\partial_t m_i + \partial_j (m_i v_j) = -\partial_i p + \partial_j \sigma_{ij}^*, \quad (28)$$

$$\partial_t \mathcal{E} + \partial_j (\mathcal{E} v_j) = -\partial_j (p v_j) + \partial_j q_j^*, \quad (29)$$

with

$$\sigma_{ij}^* = \sigma_{ij}^* (\epsilon, \nu; \mathbf{v}; \nabla \rho, \nabla \mathbf{m}) := \nu [\partial_i m_j - v_j \partial_i \rho + \partial_j m_i - v_i \partial_j \rho] - \frac{2}{3} \nu (\partial_k m_k - v_k \partial_k \rho) \delta_{ij} + \epsilon v_i \partial_j \rho, \quad (30)$$

$$q_j^* = q_j^* (\epsilon, \nu; \mathbf{v}; \nabla \rho, \nabla \mathbf{m}, \nabla \mathcal{E}) := \epsilon [\partial_j \mathcal{E} - v_i \partial_j m_i] + v_i \sigma_{ij}^* (\epsilon, \nu; \mathbf{v}; \nabla \rho, \nabla \mathbf{m}). \quad (31)$$

### 2.3.2. Discontinuity-capturing operators

The discontinuity capturing (DC) operators aim to stabilize shocks, shear layers and contact discontinuities, and are based on post-processing the non-stabilized solution  $\mathbf{u}_*$  as follows.

For simplicity of notation, we drop for the moment the superscript  $[k]$  denoting the corresponding stage of the Runge–Kutta method. First, we employ the fine-scale terms  $\rho'_h, m'_{ih}, \mathcal{E}'_h$  to locate the discontinuities. To do so, we define the fluctuation operator  $\tilde{\theta}_h := \mathbb{I} - \tilde{L}_h$ , with  $\mathbb{I}$  the identity and  $\tilde{L}_h$  the Lagrange interpolation operator onto the convected mesh  $\tilde{\mathbb{T}}_h$ , and the  $q$ -average

$$\llbracket u \rrbracket_{q,D} := \left[ \int_D u^q d\Omega / \int_D d\Omega \right]^{1/q}.$$

Then, we compute the quantities

$$e_{\rho,K} = \frac{\llbracket \tilde{\theta}_h \rho_* \rrbracket_{2,K}}{\llbracket \tilde{\rho}_* \rrbracket_{2,\tilde{\Omega}_h}}, \quad e_{m_i,K} = \frac{\llbracket \tilde{\theta}_h m_{i*} \rrbracket_{2,K}}{\sqrt{\llbracket \tilde{\rho}_* \rrbracket_{2,\tilde{\Omega}_h} \llbracket \tilde{\mathcal{E}}_* \rrbracket_{2,\tilde{\Omega}_h}}}, \quad e_{\mathcal{E},K} = \frac{\llbracket \tilde{\theta}_h \mathcal{E}_* \rrbracket_{2,K}}{\llbracket \tilde{\mathcal{E}}_* \rrbracket_{2,\tilde{\Omega}_h}}, \quad e_K = \max \{e_{\rho,K}, e_{m_i,K}, e_{\mathcal{E},K}\},$$

for each element  $K \in \tilde{\mathbb{T}}_h$ . Recall that  $\tilde{\theta}_h \psi_h = \tilde{\theta}_h \psi'_h \forall \psi_h \in \tilde{V}_h$  according to the multiscale decomposition (7), and that  $\psi'_h$  can be seen as a measure of the error in the large-scale term [66]. Hence,  $e_K$  is expected to be large at the discontinuities and small at the regions where the solution is smooth.

From  $e_K$ , we define the discontinuity sensor

$$S_K := \min \left\{ (r+1)! \left( \frac{C_{\text{DC}}}{r} \right)^{r+1} e_K, 1 \right\}, \quad (32)$$

and the diffusion coefficient

$$\tau_K^{\text{DC}} := \begin{cases} \left[ \left( \frac{h_K}{r} \right)^2 \|\nabla \cdot \mathbf{v}_*\|_{1,K} + C_c \|c_*\|_{2,K} \frac{h_K}{r} \right] S_K, & \text{if } \|\nabla \cdot \mathbf{v}_*\|_{1,K} < 0, \\ C_c \|c_*\|_{2,K} \frac{h_K}{r} S_K, & \text{if } \|\nabla \cdot \mathbf{v}_*\|_{1,K} \geq 0, \end{cases}$$

where  $h_K = [\int_K d\Omega]^{1/2}$ ,  $c_* = \sqrt{\gamma p_*/\rho_*}$  is the sound speed and  $C_{\text{DC}}$  and  $C_c$  are positive user-defined constants to fine-tune the level of diffusion. Recall that  $r$  is the order of the space discretization. The rationale behind Eq. (32) is shown in Appendix A. Then, we perform a linear reconstruction [67] of the  $\tau_K^{\text{DC}}$ 's –defined element-wise– to obtain a continuous, piecewise-linear diffusion coefficient  $\tau_h$ . More specifically,  $\tau_h^{\text{DC}}$  is defined at each vertex of the mesh as the weighted average of the  $\tau_K$ 's of the corresponding elements, where the weights are the inverses of the areas of those elements. This is done with the purpose of obtaining smoother results [47, 68]. Furthermore, we set the mass-diffusion coefficient and the kinematic viscosity

$$\epsilon_h^{\text{DC}} = C_\epsilon \tau_h^{\text{DC}}, \quad \nu_h^{\text{DC}} = C_\nu \tau_h^{\text{DC}},$$

where  $C_\epsilon$  and  $C_\nu$  are other positive user-defined constants.

Looking back to Brenner's model (27)-(29) and recalling definitions (30)-(31), we finally introduce the DC operators

$$\begin{aligned} \mathcal{S}_\rho^{\text{DC}}(\rho_h, \psi_h)_{\tilde{\Omega}_h} &:= (\epsilon_h^{\text{DC}} \partial_j \rho_h, \partial_j \psi_h)_{\tilde{\Omega}_h}, \\ \mathcal{S}_{m_i}^{\text{DC}}(\rho_h, \mathbf{m}_h, \psi_h)_{\tilde{\Omega}_h} &:= (\sigma_{ij}^* (\epsilon_h^{\text{DC}}, \nu_h^{\text{DC}}; \mathbf{v}_h; \nabla \rho_h, \nabla \mathbf{m}_h), \partial_j \psi_h)_{\tilde{\Omega}_h}, \\ \mathcal{S}_\mathcal{E}^{\text{DC}}(\rho_h, \mathbf{m}_h, \mathcal{E}_h, \psi_h)_{\tilde{\Omega}_h} &:= (q_j^* (\epsilon_h^{\text{DC}}, \nu_h^{\text{DC}}; \mathbf{v}_h; \nabla \rho_h, \nabla \mathbf{m}_h, \nabla \mathcal{E}_h), \partial_j \psi_h)_{\tilde{\Omega}_h}. \end{aligned}$$

### 2.3.3. Subgrid stabilization

Apart from discontinuities, instabilities may arise also from (initially) small perturbations which propagate through acoustic waves, as demonstrated by Scovazzi [11]. These instabilities can also be seen as a consequence of the presence of nonsymmetric terms in the Galerkin formulation (4)-(6) and of the inexact computation of the integrals in (24)-(26) [59]. Unluckily, the DC operators above are not able to detect and remove these acoustic instabilities, and thus it is necessary to provide another stabilization term.

For that purpose, we extend the subgrid stabilization (SS) technique developed by Guermond et al. for incompressible flows [56] to the present case. In particular, we define

$$\tau_K^{\text{SS}} = C_{\text{SS}} \|c_*\|_{2,K} \frac{h_K}{r}$$

for each element  $K \in \tilde{\mathcal{T}}_h$ , perform a linear reconstruction of the  $\tau_K^{\text{SS}}$ 's to obtain  $\tau_h^{\text{SS}}$ , and set

$$\epsilon_h^{\text{SS}} = C_\epsilon \tau_h^{\text{SS}}, \quad \nu_h^{\text{SS}} = C_\nu \tau_h^{\text{SS}}.$$

Then, we add diffusion only in the fine-scale terms, that is,

$$\begin{aligned} \mathcal{S}_\rho^{\text{SS}}(\rho_h, \psi_h)_{\tilde{\Omega}_h} &:= (\epsilon_h^{\text{SS}} \partial_j \tilde{\theta}_h \rho_h, \partial_j \tilde{\theta}_h \psi_h)_{\tilde{\Omega}_h}, \\ \mathcal{S}_{m_i}^{\text{SS}}(\rho_h, \mathbf{m}_h, \psi_h)_{\tilde{\Omega}_h} &:= (\sigma_{ij}^* (\epsilon_h^{\text{SS}}, \nu_h^{\text{SS}}; \mathbf{v}_h; \nabla \tilde{\theta}_h \rho_h, \nabla \tilde{\theta}_h \mathbf{m}_h), \partial_j \tilde{\theta}_h \psi_h)_{\tilde{\Omega}_h}, \\ \mathcal{S}_\mathcal{E}^{\text{SS}}(\rho_h, \mathbf{m}_h, \mathcal{E}_h, \psi_h)_{\tilde{\Omega}_h} &:= (q_j^* (\epsilon_h^{\text{SS}}, \nu_h^{\text{SS}}; \mathbf{v}_h; \nabla \tilde{\theta}_h \rho_h, \nabla \tilde{\theta}_h \mathbf{m}_h, \nabla \tilde{\theta}_h \mathcal{E}_h), \partial_j \tilde{\theta}_h \psi_h)_{\tilde{\Omega}_h}. \end{aligned}$$

Above,  $C_\epsilon$  and  $C_\nu$  are the same as for the DC operators, whereas  $C_{SS}$  is another positive user-defined constant. We have seen in our experiments that the results are not very sensitive with respect to  $C_{SS}$  once stabilization is achieved.

The stabilization operators in Eqs. (15), (18)-(19) are hence

$$\begin{aligned}\mathcal{S}_\rho(\rho_h, \psi_h)_{\tilde{\Omega}_h} &= \mathcal{S}_\rho^{\text{DC}}(\rho_h, \psi_h)_{\tilde{\Omega}_h} + \mathcal{S}_\rho^{\text{SS}}(\rho_h, \psi_h)_{\tilde{\Omega}_h}, \\ \mathcal{S}_{m_i}(\rho_h, \mathbf{m}_h, \psi_h)_{\tilde{\Omega}_h} &= \mathcal{S}_{m_i}^{\text{DC}}(\rho_h, \mathbf{m}_h, \psi_h)_{\tilde{\Omega}_h} + \mathcal{S}_{m_i}^{\text{SS}}(\rho_h, \mathbf{m}_h, \psi_h)_{\tilde{\Omega}_h}, \\ \mathcal{S}_\mathcal{E}(\rho_h, \mathbf{m}_h, \mathcal{E}_h, \psi_h)_{\tilde{\Omega}_h} &= \mathcal{S}_\mathcal{E}^{\text{DC}}(\rho_h, \mathbf{m}_h, \mathcal{E}_h, \psi_h)_{\tilde{\Omega}_h} + \mathcal{S}_\mathcal{E}^{\text{SS}}(\rho_h, \mathbf{m}_h, \mathcal{E}_h, \psi_h)_{\tilde{\Omega}_h}.\end{aligned}$$

#### 2.4. Resolution of nonlinear system (16)-(17)

To solve nonlinear system (16)-(17), we consider first the following fixed-point iteration: given  $m_i^{[l],k}$ ,  $\mathcal{E}_h^{[l],k} \in \tilde{V}_h^{[l]}$ , find  $m_i^{[l],k+1}$ ,  $\mathcal{E}_h^{[l],k+1} \in \tilde{V}_h^{[l]}$  such that

$$\begin{aligned}\left(m_i^{[l],k+1}, \psi_h^{[l]}\right)_{\tilde{\Omega}_h^{[l]}} + \Delta t a_I^l K_{m_i} \left(\mathbf{m}_h^{[l],k+1}, \psi_h^{[l]}\right)_{\tilde{\Omega}_h^{[l]}} &= b_{m_i}^{[l]}(\psi_h) + \Delta t a_I^l g_{m_i} \left(\rho_h^{[l]}, \mathbf{m}_h^{[l],k}, \mathcal{E}_h^{[l],k}, \psi_h^{[l]}\right)_{\tilde{\Omega}_h^{[l]}} + \\ &\quad \Delta t a_I^l K_{m_i} \left(\mathbf{m}_h^{[l],k}, \psi_h^{[l]}\right)_{\tilde{\Omega}_h^{[l]}},\end{aligned}\tag{33}$$

$$\begin{aligned}\left(\mathcal{E}_h^{[l],k+1}, \psi_h^{[l]}\right)_{\tilde{\Omega}_h^{[l]}} + \Delta t a_I^l K_\mathcal{E} \left(\mathcal{E}_h^{[l],k+1}, \psi_h^{[l]}\right)_{\tilde{\Omega}_h^{[l]}} &= b_\mathcal{E}^{[l]}(\psi_h) + \Delta t a_I^l g_\mathcal{E} \left(\rho_h^{[l]}, \mathbf{m}_h^{[l],k+1}, \mathcal{E}_h^{[l],k}, \psi_h^{[l]}\right)_{\tilde{\Omega}_h^{[l]}} + \\ &\quad \Delta t a_I^l K_\mathcal{E} \left(\mathcal{E}_h^{[l],k}, \psi_h^{[l]}\right)_{\tilde{\Omega}_h^{[l]}},\end{aligned}\tag{34}$$

for all  $\psi_h^{[l]} \in \tilde{V}_h^{[l]}$ . The equations above are obtained by adding the terms  $K_{m_1}$ ,  $K_{m_2}$  and  $K_\mathcal{E}$  to the left and right-hand sides of Eqs. (16)-(17). These terms are respectively the contributions of  $\nabla m_1$ ,  $\nabla m_2$  and  $\nabla \mathcal{E}$  to  $\mathcal{S}_{m_1}$ ,  $\mathcal{S}_{m_2}$  and  $\mathcal{S}_\mathcal{E}$ , viz.,

$$\begin{aligned}K_{m_1} \left(\mathbf{m}_h^{[l]}, \psi_h^{[l]}\right)_{\tilde{\Omega}_h^{[l]}} &:= \frac{4}{3} \left( \nu_h^{\text{DC}} \partial_1 m_1^{[l]}, \partial_1 \psi_h^{[l]} \right)_{\tilde{\Omega}_h^{[l]}} + \left( \nu_h^{\text{DC}} \partial_2 m_1^{[l]}, \partial_2 \psi_h^{[l]} \right)_{\tilde{\Omega}_h^{[l]}} + \\ &\quad \frac{4}{3} \left( \nu_h^{\text{SS}} \partial_1 \tilde{\theta}_h^{[l]} m_1^{[l]}, \partial_1 \tilde{\theta}_h^{[l]} \psi_h^{[l]} \right)_{\tilde{\Omega}_h^{[l]}} + \left( \nu_h^{\text{SS}} \partial_2 \tilde{\theta}_h^{[l]} m_1^{[l]}, \partial_2 \tilde{\theta}_h^{[l]} \psi_h^{[l]} \right)_{\tilde{\Omega}_h^{[l]}}, \\ K_{m_2} \left(\mathbf{m}_h^{[l]}, \psi_h^{[l]}\right)_{\tilde{\Omega}_h^{[l]}} &:= \left( \nu_h^{\text{DC}} \partial_1 m_2^{[l]}, \partial_1 \psi_h^{[l]} \right)_{\tilde{\Omega}_h^{[l]}} + \frac{4}{3} \left( \nu_h^{\text{DC}} \partial_2 m_2^{[l]}, \partial_2 \psi_h^{[l]} \right)_{\tilde{\Omega}_h^{[l]}} + \\ &\quad \left( \nu_h^{\text{SS}} \partial_1 \tilde{\theta}_h^{[l]} m_2^{[l]}, \partial_1 \tilde{\theta}_h^{[l]} \psi_h^{[l]} \right)_{\tilde{\Omega}_h^{[l]}} + \frac{4}{3} \left( \nu_h^{\text{SS}} \partial_2 \tilde{\theta}_h^{[l]} m_2^{[l]}, \partial_2 \tilde{\theta}_h^{[l]} \psi_h^{[l]} \right)_{\tilde{\Omega}_h^{[l]}}, \\ K_\mathcal{E} \left(\mathcal{E}_h^{[l]}, \psi_h^{[l]}\right)_{\tilde{\Omega}_h^{[l]}} &:= \left( \epsilon_h^{\text{DC}} \partial_j \mathcal{E}_h^{[l]}, \partial_j \psi_h^{[l]} \right)_{\tilde{\Omega}_h^{[l]}} + \left( \epsilon_h^{\text{SS}} \partial_j \tilde{\theta}_h^{[l]} \mathcal{E}_h^{[l]}, \partial_j \tilde{\theta}_h^{[l]} \psi_h^{[l]} \right)_{\tilde{\Omega}_h^{[l]}},\end{aligned}$$

and their purpose is to approximately cancel the stiff terms in the right-hand side of the equations so as to accelerate the convergence rate of the fixed-point iteration and to allow for a larger<sup>1</sup>  $\Delta t$ . Note that these terms lead to standard stiffness matrices in the numerical implementation, and that the system (33)-(34) is therefore linear and formed by three uncoupled positive definite matrices. Also note that the energy equation is solved with the updated momentum.

Then, we apply Anderson's acceleration method [69, 70] to the fixed point iteration (33)-(34). This method is very simple to implement, converges faster than the classical fixed-point iteration method – superlinearly instead of linearly – and is more robust. The iterations are started by assuming that the

<sup>1</sup>Recall that the fixed-point iteration  $x^{k+1} = g(x^k)$  converges if the eigenvalues of the Jacobian  $J(x) := \partial g(x)/\partial x$  are inside the unit circumference in the complex plane. In addition, the convergence speed is governed by the largest modulus eigenvalue. Thus, a stiff right-hand side results in a very slow convergence. Instead, we can consider the fixed point iteration  $x^{k+1} + G^k x^{k+1} = g(x^k) + G^k x^k$ , with  $G^k$  an approximation to  $-J(x^k)$ , so as to ideally cancel the stiff part in the right-hand side and to obtain a higher convergence rate. Note that the latter iteration is equivalent to the quasi-Newton iteration  $x^{k+1} = x^k - [I + G^k]^{-1} [x^k - g(x^k)]$ .

degrees of freedom of  $\rho_h^{[l],0}$ ,  $m_{i_h}^{[l],0}$  and  $\mathcal{E}_h^{[l],0}$  equal those of  $\rho_h^{[l-1]}$ ,  $m_{i_h}^{[l-1]}$  and  $\mathcal{E}_h^{[l-1]}$ , and are stopped when the relative variation of the  $L^2$  norm of the unknowns is under a prescribed tolerance ( $10^{-8}$  or smaller in the case of this work). We remark that, in our experiments, Anderson's acceleration method has considerably reduced the number of required iterations and increased the stability of the scheme in comparison to the classical fixed-point iteration method.

### 2.5. Summary of the scheme and remarks

The scheme described in previous sections is summarized in Algorithm 1 and will be referred hereafter as the Runge–Kutta forward Lagrange–Galerkin (RK-FLG) scheme.

---

**Algorithm 1:** Step from  $t^n$  to  $t^{n+1}$  of the RK-FLG method.

---

**Data:** convected mesh  $\tilde{\mathbb{T}}_h^n$  and solution  $\tilde{\mathbf{u}}_h^n = [\tilde{\rho}_h^n, \tilde{\mathbf{m}}_h^n, \tilde{\mathcal{E}}_h^n] \in (\tilde{V}_h^n)^4$  at  $t^n$ ; reference mesh  $\mathbb{T}_h$  and space  $V_h$ ; inflow conditions  $\mathbf{u}_{\text{in}}(\mathbf{x}, t)$ ; time step  $\Delta t$ ; Butcher tableaux (8).

- 1 **perform** the  $L^2$  projection of  $\tilde{\rho}_h^n$ ,  $\tilde{m}_{i_h}^n$  and  $\tilde{\mathcal{E}}_h^n$  onto  $V_h$  (Eqs. (24)-(26) with  $n+1$  replaced by  $n$ ) to obtain  $\mathbf{u}_h^n = [\rho_h, \mathbf{m}_h, \mathcal{E}_h] \in (V_h)^4$ ,
- 2 **set**  $t^{[1]} = t^n$ ,  $\mathbf{u}_*^{[1]} = \mathbf{u}_h^n$ ,  $\mathbf{u}_h^{[1]} = \mathbf{u}_h^n$ ,  $\tilde{\mathbb{T}}_h^{[1]} = \mathbb{T}_h$ ,
- 3 **for**  $l = 2, \dots, s$  **do**
- 4     **compute** convected mesh  $\tilde{\mathbb{T}}_h^{[l]}$  from Eq. (9),
- 5     **compute** non-stabilized solution  $\mathbf{u}_*^{[l]}$  by solving (10),
- 6     **compute** discontinuity-capturing and subgrid stabilization operators as shown in Section 2.3,
- 7     **compute** stabilized density  $\rho_h^{[l]}$  by solving (15),
- 8     **solve** fixed-point iteration (33)-(34) with Anderson's acceleration method [69] to obtain  $\mathbf{m}_h^{[l]}$  and  $\mathcal{E}_h^{[l]}$ ,
- 9 **end**
- 10 **compute** convected mesh  $\tilde{\mathbb{T}}_h^{n+1}$  and solution  $\tilde{\mathbf{u}}_h^{n+1}$  from Eqs. (20)-(23).

**Result:**  $\tilde{\mathbb{T}}_h^{n+1}$ ,  $\tilde{\mathbf{u}}_h^{n+1} \in (\tilde{V}_h^{n+1})^4$ .

---

The time step is

$$t^{n+1} - t^n = \min_{K \in \mathbb{T}_h} \left\{ \text{CFL}_c^n \frac{h_K^{\min}/r}{\|\mathbf{c}_h^n\|_{2,K}}, \frac{\text{CFL}_v^n}{\|\nabla \cdot \mathbf{v}_h^n\|_{1,K}}, \frac{\text{CFL}_v^n}{\|\nabla \times \mathbf{v}_h^n\|_{1,K}} \right\},$$

where  $h_K^{\min}$  is the minimum edge size, whereas  $\text{CFL}_c^n$  and  $\text{CFL}_v^n$  are two CFL numbers. Depending on the problem, two strategies are adopted, namely that (S1) the CFL numbers are constant, i.e.,  $\text{CFL}_\square^n = \text{CFL}_\square$ , or that (S2) the CFL numbers increase geometrically at the first time steps, i.e.,  $\text{CFL}_\square^n = \min\{2\text{CFL}_\square^{n-1}, \text{CFL}_\square\}$ , with  $\text{CFL}_\square^0 = \text{CFL}_\square/2^{10}$ . Here, the square symbol denotes either  $c$  or  $v$ , and  $\text{CFL}_c$  and  $\text{CFL}_v$  are user-defined parameters.

The linear systems to be solved at the different steps are all symmetric, positive definite, and involve only one conservative variable at once. However, for each conservative variable there are two coupled blocks, which appear from the discretization of the large and the fine-scale spaces. The block corresponding to the bubble degrees of freedom is eliminated by static condensation, leading to a linear system for the polynomial degrees of freedom only. This system is solved via the conjugate gradients method with a Jacobi preconditioner.

Next we address some properties of the scheme.

**Global conservation.** Theoretically, the present method preserves mass, momentum and total energy irrespective of the order of the space and time discretizations, whenever (C1) there is no fluid entering nor leaving the domain  $\Omega_h$ , (C2) the total force over the domain is null and (C3) no power is introduced through the boundary. In effect, under these conditions,

$$\begin{aligned} f_{m_i}(\rho_h^{[k]}, \mathbf{m}_h^{[k]}, \mathcal{E}_h^{[k]}, 1)_{\tilde{\Omega}_h^{[k]}} &= -\langle p_h^{[k]} n_i, 1 \rangle_{\partial \tilde{\Omega}_h^{[k]}} = 0, \\ f_{\mathcal{E}}(\rho_h^{[k]}, \mathbf{m}_h^{[k]}, \mathcal{E}_h^{[k]}, 1)_{\tilde{\Omega}_h^{[k]}} &= -\langle p_h^{[k]} v_j^{[k]} n_j, 1 \rangle_{\partial \tilde{\Omega}_h^{[k]}} = 0, \end{aligned}$$

for any stage  $k$  of the Runge–Kutta method. Furthermore, it can be easily checked that

$$\mathcal{S}_\rho \left( \rho_h^{[k]}, 1 \right)_{\tilde{\Omega}_h^{[k]}} = 0, \quad \mathcal{S}_{m_i} \left( \rho_h^{[k]}, \mathbf{m}_h^{[k]}, 1 \right)_{\tilde{\Omega}_h^{[k]}} = 0, \quad \mathcal{S}_\mathcal{E} \left( \rho_h^{[k]}, \mathbf{m}_h^{[k]}, \mathcal{E}_h^{[k]}, 1 \right)_{\tilde{\Omega}_h^{[k]}} = 0.$$

Hence, we have from Eqs. (18)-(19), (21)-(23) that

$$\begin{aligned} (\tilde{\rho}_h^{n+1}, 1)_{\tilde{\Omega}_h^{n+1}} &= (\rho_h^{[1]}, 1)_{\tilde{\Omega}_h^{[1]}} = (\rho_h^n, 1)_{\Omega_h}, \\ (\tilde{m}_{i_h}^{n+1}, 1)_{\tilde{\Omega}_h^{n+1}} &= (m_{i_h}^{[1]}, 1)_{\tilde{\Omega}_h^{[1]}} = (m_{i_h}^n, 1)_{\Omega_h}, \\ (\tilde{\mathcal{E}}_h^{n+1}, 1)_{\tilde{\Omega}_h^{n+1}} &= (\mathcal{E}_h^{[1]}, 1)_{\tilde{\Omega}_h^{[1]}} = (\mathcal{E}_h^n, 1)_{\Omega_h}. \end{aligned}$$

On the other hand, since no fluid enters nor leaves the domain,  $\tilde{\Omega}_h(t) \equiv \Omega_h \forall t$ . Thus, the  $L^2$  projections (24)-(26) and the equations above yield

$$\begin{aligned} (\rho_h^{n+1}, 1)_{\Omega_h} &= (\tilde{\rho}_h^{n+1}, 1)_{\Omega_h} = (\tilde{\rho}_h^{n+1}, 1)_{\tilde{\Omega}_h^{n+1}} = (\rho_h^n, 1)_{\Omega_h}, \\ (m_{i_h}^{n+1}, 1)_{\Omega_h} &= (\tilde{m}_{i_h}^{n+1}, 1)_{\Omega_h} = (\tilde{m}_{i_h}^{n+1}, 1)_{\tilde{\Omega}_h^{n+1}} = (m_{i_h}^n, 1)_{\Omega_h}, \\ (\mathcal{E}_h^{n+1}, 1)_{\Omega_h} &= (\tilde{\mathcal{E}}_h^{n+1}, 1)_{\Omega_h} = (\tilde{\mathcal{E}}_h^{n+1}, 1)_{\tilde{\Omega}_h^{n+1}} = (\mathcal{E}_h^n, 1)_{\Omega_h}, \end{aligned}$$

q.e.d. In practice, however, conservation [with machine accuracy](#) is not achieved because:

- (i) The linear systems are not solved exactly, but with an iterative method until a prescribed tolerance. Thus, conservation errors cannot be expected to be smaller than the latter. This inconvenience can be circumvented by employing constrained iterative solvers such as [71, 72].
- (ii) It is not possible to impose conditions (C1)-(C3) at the same time. For example, if we impose  $p = 0$  on the entire boundary, (C2)-(C3) are met, and mass, momentum and total energy are preserved in the moving domain  $\tilde{\Omega}_h(t)$ . However, due to numerical errors, there is some flux across  $\partial\Omega_h$  even if the theoretical solution satisfies  $v_i n_i = 0$ . Therefore,  $\tilde{\Omega}_h^{n+1} \neq \Omega_h$  and there are mass, momentum and energy losses through Eqs. (24)-(26). On the other hand, if we impose  $v_i n_i = 0$  at the boundary, conditions (C1) and (C3) are met and hence we could expect conservation of mass and energy. Nevertheless, the total force over the boundary is not numerically zero, even if that is the case for the theoretical solution, and therefore exact momentum conservation cannot be achieved.
- (iii) Even if  $\tilde{\Omega}_h^{n+1} \equiv \Omega_h$ , the  $L^2$  projections (24)-(26) are not fully conservative when computed via high-order quadrature rules, as is our case. To achieve full conservation, mesh intersection techniques [62–64] or remap algorithms [45, 46] can be employed.

Normally, Lagrangian methods are considered *conservative* when they theoretically preserve mass, momentum and total energy if conditions (C2)-(C3) are imposed; see, e.g. [6, 10]. It is implicitly understood that conservation cannot be guaranteed from a physical point of view under other conditions, and that the resolution of linear systems and other operations may provoke slight conservation errors. In comparison to conservative Lagrangian methods, the present RK-FLG method additionally requires (C1) and suffers from drawback (iii) above, whose associated conservation errors are still small, but more noticeable. Thus, one can expect the conservation errors of this method to be small, but not close to machine accuracy. To avoid confusions, we shall say that the RK-FLG method is *nearly-conservative*, i.e., conservative only with high accuracy.

**Geometric conservation law.** The present scheme also satisfies the so-called discrete geometric conservation law, namely, it preserves uniform solution states, which is of great importance for maintaining nonlinear

stability [73, 74]. In effect, if  $\mathbf{u}_h^{[k]}$  represents a uniform state  $[\rho, \mathbf{m}, \mathcal{E}]$ , then

$$\begin{aligned} f_{m_i}(\rho_h^{[k]}, \mathbf{m}_h^{[k]}, \mathcal{E}_h^{[k]}, \psi_h^{[k]})_{\widetilde{\Omega}_h^{[k]}} &= p \left[ \left( 1, \frac{\partial \psi_h^{[k]}}{\partial x_i} \right)_{\widetilde{\Omega}_h^{[k]}} - \langle n_i, \psi_h^{[k]} \rangle_{\partial \widetilde{\Omega}_h^{[k]}} \right] = 0, \\ f_{\mathcal{E}}(\rho_h^{[k]}, \mathbf{m}_h^{[k]}, \mathcal{E}_h^{[k]}, \psi_h^{[k]})_{\widetilde{\Omega}_h^{[k]}} &= pv_j \left[ \left( 1, \frac{\partial \psi_h^{[k]}}{\partial x_j} \right)_{\widetilde{\Omega}_h^{[k]}} - \langle n_j, \psi_h^{[k]} \rangle_{\partial \widetilde{\Omega}_h^{[k]}} \right] = 0, \end{aligned}$$

and

$$\mathcal{S}_\rho(\rho_h^{[k]}, \psi_h^{[k]})_{\widetilde{\Omega}_h^{[k]}} = 0, \quad \mathcal{S}_{m_i}(\rho_h^{[k]}, \mathbf{m}_h^{[k]}, \psi_h^{[k]})_{\widetilde{\Omega}_h^{[k]}} = 0, \quad \mathcal{S}_{\mathcal{E}}(\rho_h^{[k]}, \mathbf{m}_h^{[k]}, \mathcal{E}_h^{[k]}, \psi_h^{[k]})_{\widetilde{\Omega}_h^{[k]}} = 0,$$

for all  $\psi_h^{[k]} \in \widetilde{V}_h^{[k]}$ . Also, if  $\mathbf{u}_h^{[1]}, \dots, \mathbf{u}_h^{[l-1]}$  represent uniform states, by virtue of (9), the mesh  $\widetilde{\mathbb{T}}_h^{[l]}$  is just a translation of the mesh  $\widetilde{\mathbb{T}}_h^{[1]}$  and therefore  $(1, \psi_h^{[l]})_{\Omega_h^{[l]}} = (1, \psi_h^{[1]})_{\widetilde{\Omega}_h^{[1]}}$ . Looking back at Eqs. (15)-(17) and taking into account the relations above, it is easy to check that  $\mathbf{u}_h^{[l]}$  is also uniform state solution, i.e.,  $\rho_h^{[l]} = \rho$ ,  $m_{i_h}^{[l]} = m_i$ ,  $\mathcal{E}_h^{[l]} = \mathcal{E}$ . Hence, it is deduced that  $\widetilde{\mathbf{u}}_h^{n+1}$  is a uniform state solution if so is  $\mathbf{u}_h^n$ . Since  $\widetilde{\mathbf{u}}_h^{n+1}$  is smooth, the  $L^2$  projection (24)-(26) can be computed exactly, even though  $\widetilde{\mathbb{T}}_h^{n+1} \neq \mathbb{T}_h$ , and thus  $\mathbf{u}_h^{n+1}$  is also a uniform state solution, q.e.d.

*Local conservation.* Since we employ a continuous Galerkin discretization, the scheme is locally conservative, as pointed by Hughes et al. [75]. In addition, it can be written as a residual distribution method [18, 19], which also guarantees local conservation and, under some reasonable assumptions such as stability and consistency of the scheme, convergence of the numerical solution to a weak solution of the Brenner's equations (recall that these are the equations we are actually solving, since we employ artificial viscosity). At the same time, Brenner's regularization ensures positivity of pressure and density and satisfies entropy principles [54], so we can assume that the weak solution of the viscous model converges to a weak dissipative solution of the Euler equations when the diffusion parameters  $\epsilon, \nu \rightarrow 0$ . Since this is automatically satisfied when the mesh size  $h \rightarrow 0$ , we can expect that our method converges to a weak dissipative solution [76] of the Euler equations.

### 3. Numerical experiments

In this section, we test the RK-FLG method against several canonical examples in the literature to analyze its performance. For this purpose, we introduce the  $L^1$  error indicator

$$e_1 := \frac{\|\rho_h - \rho\|_{1, \Omega_h}}{\|\rho\|_{1, \Omega_h}} + \sum_{i=1}^2 \frac{\|m_{ih} - m_i\|_{1, \Omega_h}}{\sqrt{\|\rho\|_{1, \Omega_h} \|\mathcal{E}\|_{1, \Omega_h}}} + \frac{\|\mathcal{E}_h - \mathcal{E}\|_{1, \Omega_h}}{\|\mathcal{E}\|_{1, \Omega_h}},$$

where  $\|u\|_{1, \Omega_h} := \int_{\Omega_h} |u| d\Omega$  denotes the usual  $L^1$  norm. Also, we introduce the conservation errors

$$e_\rho := \left| \frac{[\![\rho_h - \rho_h^0]\!]_{1, \Omega_h}}{[\![\rho_h^0]\!]_{1, \Omega_h}} \right|, \quad e_{m_i} := \left| \frac{[\![m_{ih} - m_{ih}^0]\!]_{1, \Omega_h}}{\sqrt{[\![\rho_h^0]\!]_{1, \Omega_h} [\![\mathcal{E}_h^0]\!]_{1, \Omega_h}}} \right|, \quad e_{\mathcal{E}} := \left| \frac{[\![\mathcal{E}_h - \mathcal{E}_h^0]\!]_{1, \Omega_h}}{[\![\mathcal{E}_h^0]\!]_{1, \Omega_h}} \right|.$$

for those problems in which either mass, momentum or energy are preserved. Here, the variables without superscript are assumed to be evaluated at the final time  $T$ . The mesh size is defined as  $h := \sqrt{4|\Omega_h|/\sqrt{3}N_E}$ , with  $|\Omega_h| = \int_{\Omega_h} d\Omega$  and  $N_E$  the number of elements; that is, if all the elements were equilateral triangles of the same size,  $h$  would represent the edge size of those.

In all tests, we set

$$C_{SS} = 100, \quad C_{DC} = 5, \quad C_c = 0.1, \quad C_\epsilon = 10, \quad C_\nu = 5, \quad CFL_c = 1, \quad CFL_v = 0.30,$$

unless specified otherwise. In addition, the reference meshes were generated with BAMG software [77], space discretizations up to fifth-order were considered and the fourth-order ARK4(3)7L[2]SA1 Runge–Kutta method of Kennedy and Carpenter [78] was employed to march in time. The integrals in the formulation are computed via the quadrature rules detailed in Table 1. We employ one set of rules for Eqs. (24)–(26) and another set for the rest of integrals. The codes were implemented in Julia language [79].

Table 1: Number of nodes, degree of accuracy and reference of the employed quadrature rules.

	$r = 1$	$r = 2$	$r = 3$	$r = 4$	$r = 5$
Integrals in (24)–(26)	25, 10, [80]	88, 20, [81]	175, 30, [82]	175, 30, [82]	175, 30, [82]
Rest of integrals	16, 8 [80]	42, 14, [80]	88, 20, [81]	88, 20, [81]	175, 30, [82]

Discontinuous initial conditions are regularized by using functions of the form  $H_h(x) := 0.5 + 0.5 \tanh(x/(2h/r))$  instead of unit step functions. In addition, the initial conditions of some tests such as the Shu–Osher tube and the shock–vortex interaction consist on the superposition of a shock wave with a given fluid field. However, it is very difficult to provide an initial condition that matches exactly the numerical solution for a shock wave, and hence some small wiggles may appear and spoil the solution –see, e.g., [52, Fig. 1]. As a remedy, we first simulate a (steady or traveling) shock wave and save the solution when those wiggles are away from the shock. Then, we employ the numerical solution at the shock (as well as the given incremental fluid field) to define the initial condition of those tests.

### 3.1. Acoustic wave propagation

As initial benchmark, we consider the simple case of the one-dimensional propagation of an acoustic wave. This is done to prove the effect of the SS and to compute the maximum CFL<sub>c</sub> number at which the method can run without becoming unstable. Following [10, 11], we consider the initial conditions

$$\rho^0 = 1 + \omega, \quad p^0 = 1 + \omega, \quad v_1^0 = \omega, \quad v_2^0 = 0,$$

with

$$\omega = \begin{cases} 0.01 \left[ 1 - \cos \left( \frac{2\pi (x_1 - 0.25)}{0.25} \right) \right], & 0.25 < x_1 < 0.50, \\ 0, & \text{otherwise.} \end{cases}$$

The domain is the rectangle  $[0, 1] \times [0, 0.1]$ ,  $\gamma = 7/5$ , we impose  $v_i n_i = 0$  on the entire boundary, the final time is  $T = 0.3$  and strategy (S1) is employed.

Fig. 3 shows the solution for fifth-order elements and (i) no SS and no DC, (ii) no SS and DC, (iii) SS and no DC, and (iv) SS and DC. As can be seen, in the case (i), the solution turns unstable when  $t = 0.04$ . In the case (ii), the method runs until  $t = T$ , although the solution presents many oscillations and is smeared at the peaks. However, in the cases (iii) and (iv), the solutions are coincident, smooth and compare very well with those of [10, 11, 16]. Hence, the SS is necessary even if the DC operators are active, as already pointed by [11] in the context of variational multiscale methods. Also note that the DC operators do not smear the solution if the SS is applied. On the other hand, Table 2 shows the maximum CFL<sub>c</sub> number at which the method can run without suffering from acoustic instabilities.

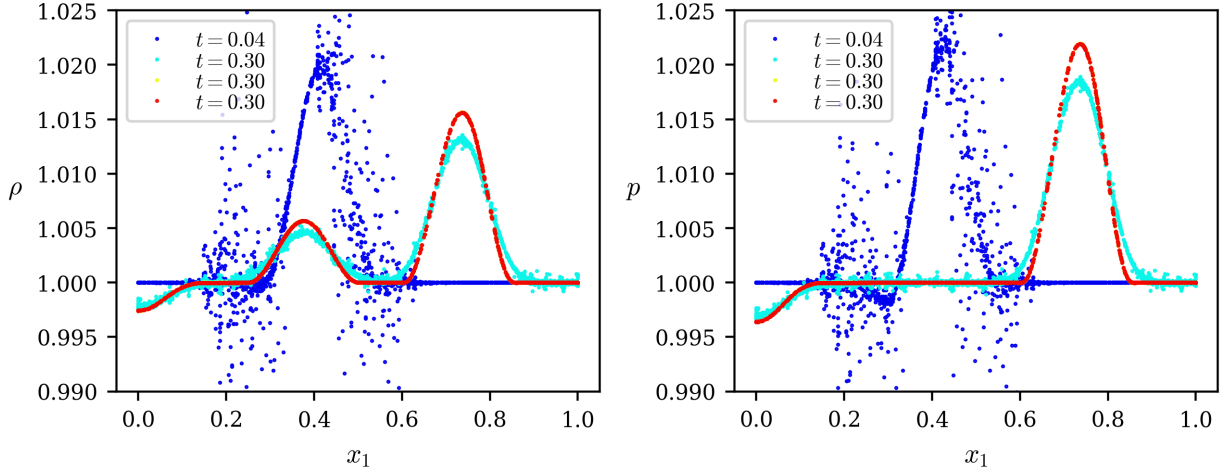


Figure 3: Numerical solution of the acoustic wave test at time  $t$  for  $r = 5$ ,  $h/r \approx 0.0075$  and  $\text{CFL}_c = 1.45$ . Legend: (—) no SS and no DC, (—) no SS and DC, (—) SS and no DC, (—) SS and DC.

Table 2: Maximum  $\text{CFL}_c$  number for acoustic stability.

$r$	1	2	3	4	5
$\text{CFL}_c^{\max}$	3.10	2.20	1.90	1.70	1.45

### 3.2. Sod tube

Next we run the classical Sod tube test to assess the ability of the method to capture planar waves. The initial condition is given by

$$\rho^0, p^0 = \begin{cases} 1, & 1, & \text{if } x_1 \leq 0.5, \\ 0.125, & 0.1, & \text{if } x_1 > 0.5, \end{cases}$$

and  $v_1^0 = v_2^0 = 0$ . The domain is the rectangle  $[0, 1] \times [0, 0.1]$ ,  $\gamma = 7/5$ ,  $v_i n_i = 0$  is imposed at each boundary, strategy (S1) is employed and  $T = 0.2$ . The theoretical solution is obtained from [83]. To demonstrate the robustness of the method, we consider meshes that are somehow irregular and with edges that are not aligned with the  $x_2$  direction, as shown in Fig. 4.

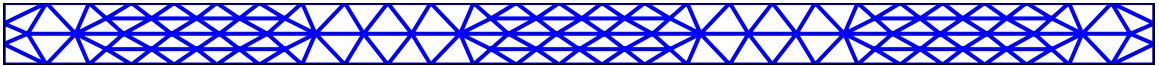
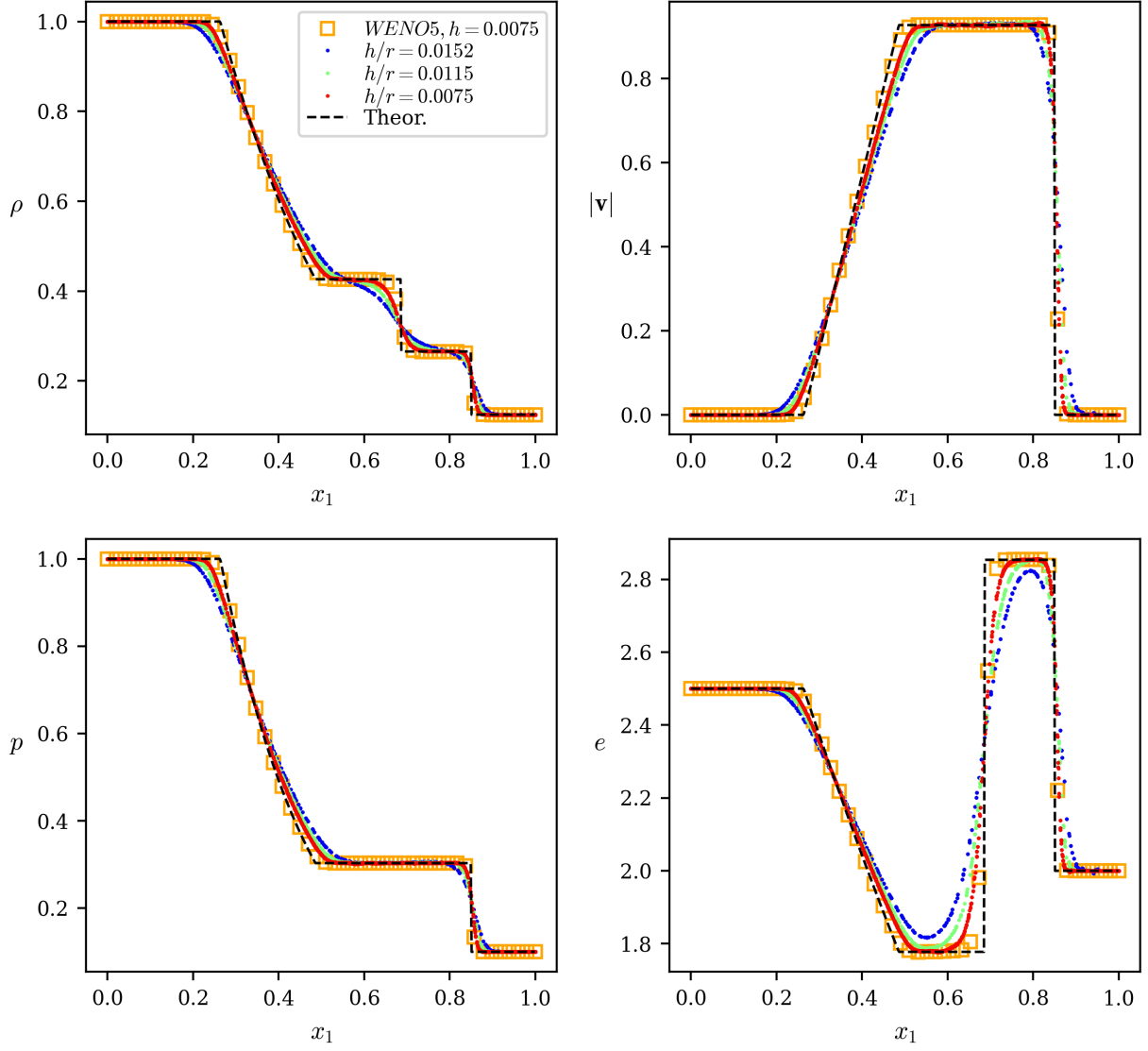


Figure 4: Mesh of size  $h \approx 0.0375$  employed in the Sod test.

As can be seen in Fig. 5, as the mesh is refined, the shock wave, the contact discontinuity and the expansion wave are captured without oscillations and within the same planes. In addition, the method is able to keep constant the velocity and the pressure across the contact discontinuity despite these are not the discretization variables. For brevity, we only show the results for  $r = 5$  as those for lower-order discretizations are almost identical whenever  $h/r$  remains the same. The figure also shows the results for a WENO5 scheme<sup>2</sup> [84]. It can be checked that our method is slightly more dissipative at the expansion wave and at the contact discontinuity, whereas the shock wave is captured similarly.

<sup>2</sup>Note that, in finite volume methods, the distance between nodes is  $h$ , and not  $h/r$  as in finite element methods.



Figure 5: Numerical solution of the Sod test at  $t = 0.2$  for  $r = 5$ .

In addition, Fig. 6 shows the behavior of the  $L^1$  error and of the conservation errors in mass and energy with respect to the mesh discretization. It can be checked that the  $L^1$  error does not depend sensitively with  $r$  and converges as  $\mathcal{O}(h/r)$ , as expected. The conservation errors vary from  $10^{-8}$  to  $10^{-5}$ , and are one order of magnitude smaller for  $r \geq 3$ .

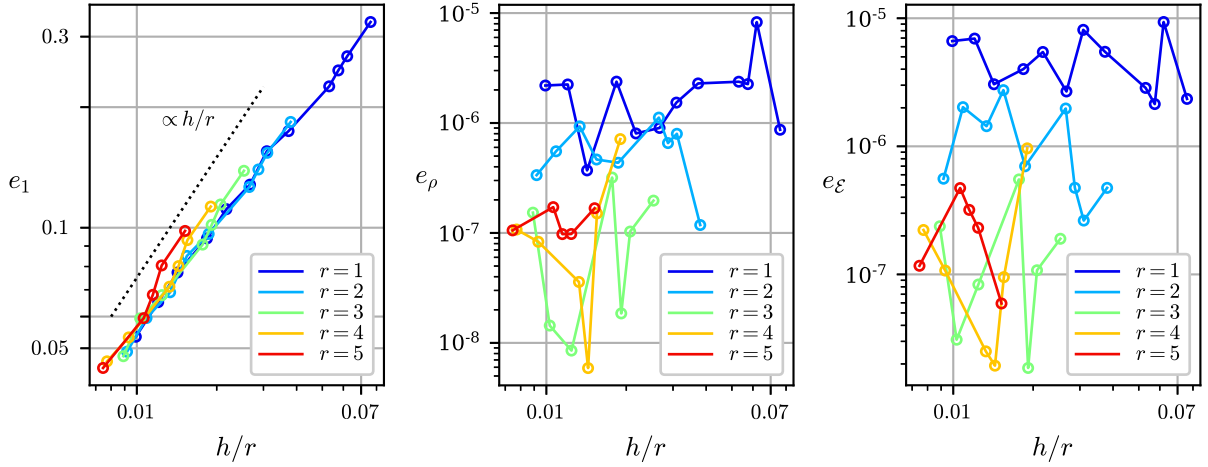


Figure 6: Numerical errors in the Sod test.

### 3.3. Shu–Osher tube

The Shu–Osher tube test [85] is given by the initial conditions

$$\rho^0, v_1^0, v_2^0, p^0 = \begin{cases} 3.857143, & 2.629369, & 0 & 10.33333, & \text{if } x_1 < -4, \\ 1 + \lambda \sin(5x_1), & 0, & 0, & 1, & \text{if } x_1 \geq -4, \end{cases}$$

with  $\lambda = 0.2$ , in the domain  $[-5, 5] \times [-0.5, 0.5]$ . Note that  $\lambda = 0$  corresponds to the case of a Mach 3 shock wave moving to the right. Normal velocity conditions are imposed on the entire boundary,  $\gamma = 7/5$ , the final time is  $T = 1.8$  and strategy (S1) is employed.

The reference solution of this test has a very complex structure in the density, and thus it is usually employed to demonstrate the advantages of using high-order space discretizations. However, as shown in Fig. 7, in our case we obtain similar results for different  $r$  whenever  $h/r$  –or the number of nodes– remains similar. Indeed, linear elements seem to be a little less dissipative. The latter is due to the fact that the discontinuity sensor is defined element-wise; thus, for high-order elements the artificial viscosity acts on a wider region and smears more the post-shock oscillations. In the figure, we also plot the results obtained with a second-order MUSCL code [84] and with a WENO5 finite volume scheme [84] for a similar number of degrees of freedom in the  $x_1$  direction. As can be seen, our method yields better results than the MUSCL method, although it is again more dissipative than the WENO5 scheme. Nevertheless, the structure of the post-shock oscillations is well captured and the results seem to converge to the reference solution when the mesh is refined. Also note that our results for  $r = 1$  look qualitatively more similar to those of WENO5 scheme than to those of MUSCLE method; therefore, we can say that in our case low and high-order elements yield similar results not because of a poor performance of high-order elements, but because of a good performance of low-order ones.

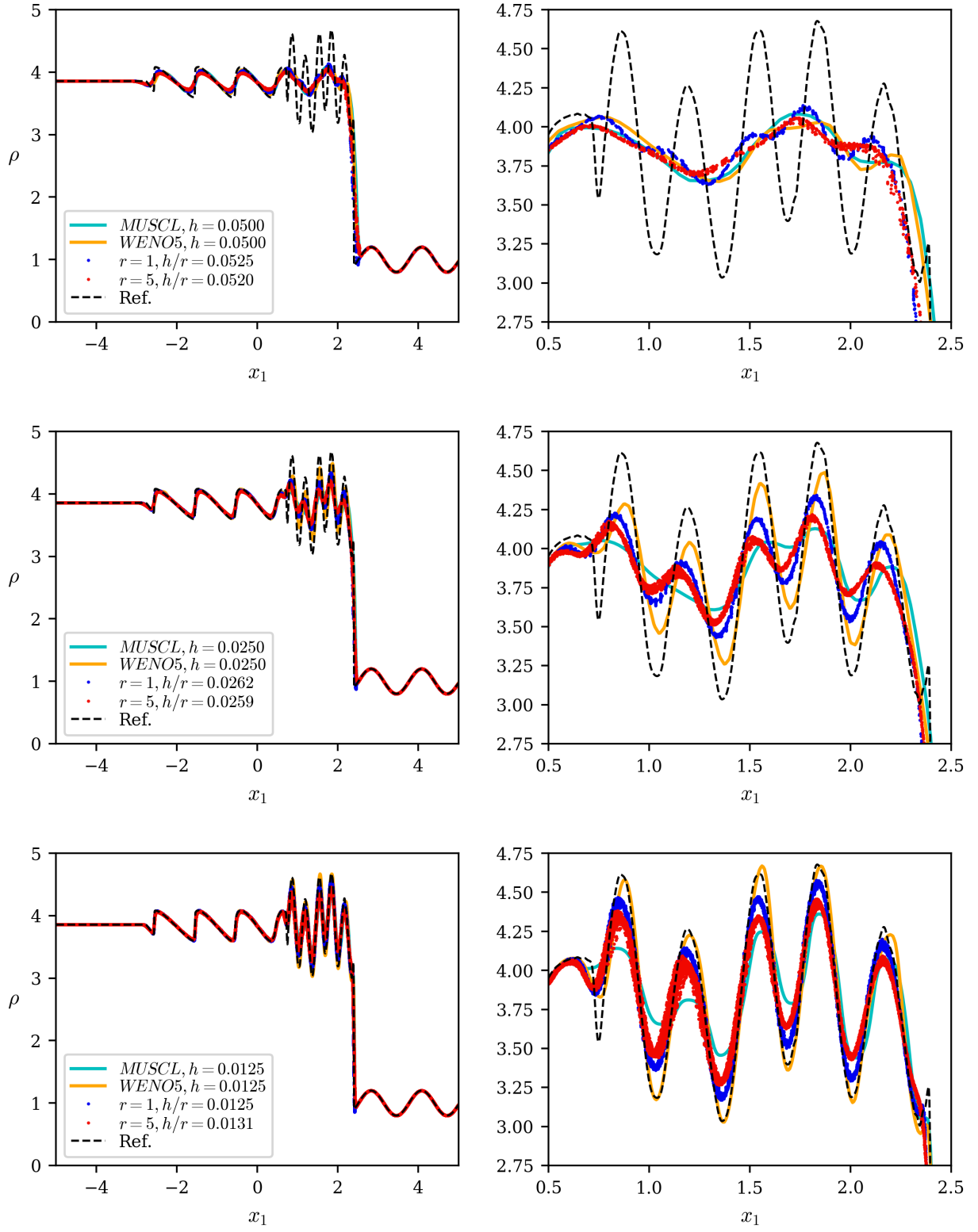


Figure 7: Numerical solution of the Shu–Osher test at  $t = 1.8$  at the full domain (left) and at the region  $0.5 \leq x_1 \leq 2.5$  (right). The reference solution corresponds to the WENO5 scheme for  $h = 0.003125$ .

### 3.4. Two-dimensional Noh implosion

The two-dimensional Noh implosion [86] is a classical benchmark used to check the preservation of symmetry in radial shocks. It is defined by an ideal gas with  $\gamma = 5/3$  and initial conditions

$$\rho^0 = 1, \quad v_i^0 = -\frac{x_i}{|\mathbf{x}|}, \quad p^0 = 0.$$

To avoid numerical problems, the initial pressure is set to  $p^0 = 10^{-9}$ . The theoretical solution is a shock which propagates radially outwards with a velocity  $1/3$ , viz.,

$$\rho, v_i, p = \begin{cases} 16, & 0, & \frac{16}{3}, & \text{if } |\mathbf{x}| \leq t/3, \\ 1 + \frac{t}{|\mathbf{x}|}, & -\frac{x_i}{|\mathbf{x}|}, & 0, & \text{if } |\mathbf{x}| > t/3. \end{cases}$$

We consider two versions of this test. In the first one, the equations are solved in the domain  $[0, 0.4]^2$  with a uniform mesh, with zero pressure conditions on the right and upper boundaries and non-penetration conditions –i.e.,  $v_i n_i = 0$ – on the left and bottom boundaries. In the second one, the equations are solved in the domain  $[-0.4, 0.4]^2$  with a non-uniform mesh defined approximately by  $N \times 2N$  elements in the first quadrant,  $2N \times 2N$  elements in the second quadrant,  $2N \times N$  elements in the third quadrant and  $N \times N$  elements in the fourth quadrant, with  $N$  a given parameter. In this case, zero pressure boundary conditions are imposed on the entire boundary. The final time is  $T = 0.6$  and strategy (S2) is employed. **It should be pointed that some finite volume and finite differences methods, including WENO schemes, fail to solve this problem as they develop instabilities in the cold gas region [87, 88].**

Fig. 8 shows the solution for fifth-order elements. As can be seen, radial symmetry is preserved very well in the case of uniform meshes. In the case of non-uniform meshes, the numerical solution loses symmetry at the center of the domain; e.g., the velocity is no longer zero at this region. However, radial symmetry is still preserved at the shock, as can also be seen in Fig. 9. This is in accordance with other references [6, 13]. For brevity, the results for  $r < 5$  are not shown, although they are nearly identical whenever  $h/r$  remains similar. The large error in the densities at the center of the domain –both for uniform and non-uniform meshes– is a common pathology known as wall-heating problem [86].

On the other hand, Fig. 10 shows an example of non-uniform mesh and the values of the discontinuity sensor at each element. It can be checked that the sensor is active only at the elements closest to the shock. Finally, it is seen in Fig. 11 that the  $L^1$  error indicator behaves as  $\mathcal{O}(h/r)$  and does not depend sensitively on  $r$ . Note that this error is only slightly higher for non-uniform meshes.

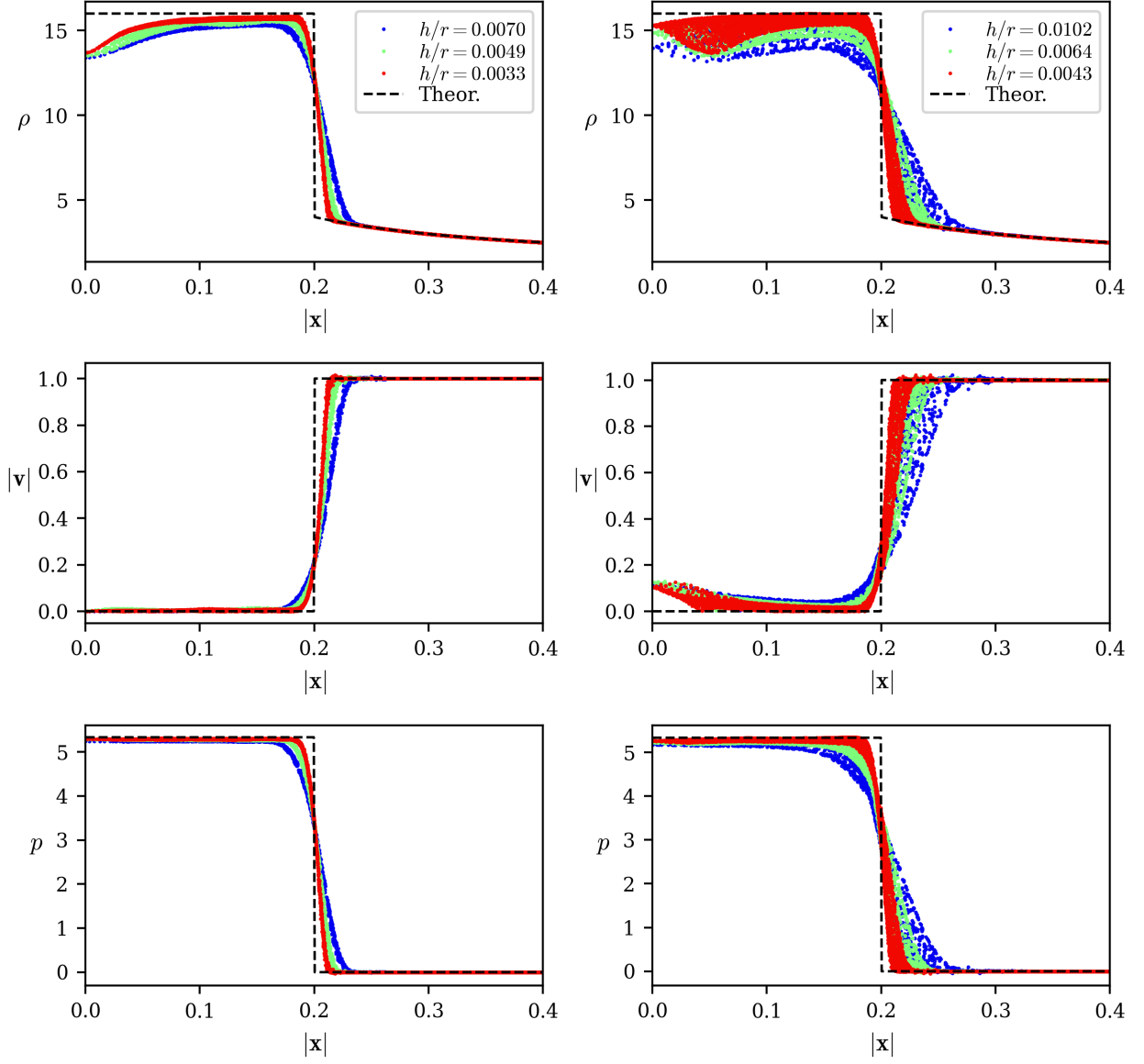


Figure 8: Numerical solution of the Noh test at  $t = 0.6$  in the domain  $[0, 0.4]^2$  (left) and in the domain  $[-0.4, 0.4]^2$  (right) for  $r = 5$ .

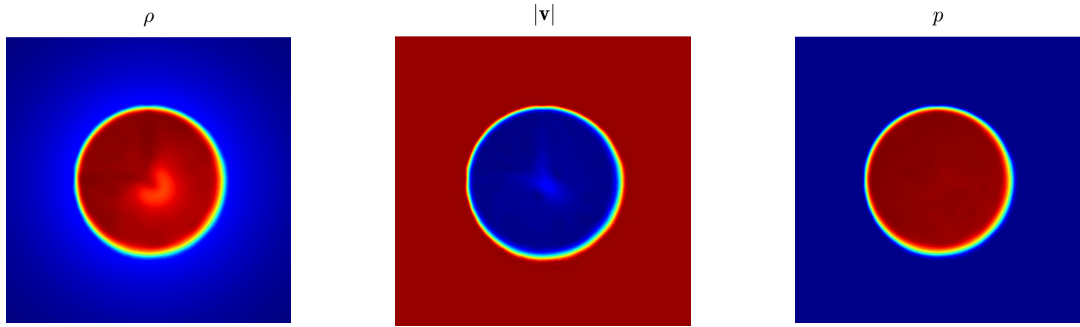


Figure 9: Numerical solution of the Noh test at  $t = 0.6$  in the domain  $[-0.4, 0.4]^2$  for  $r = 5$  and  $h/r \approx 0.0043$ .

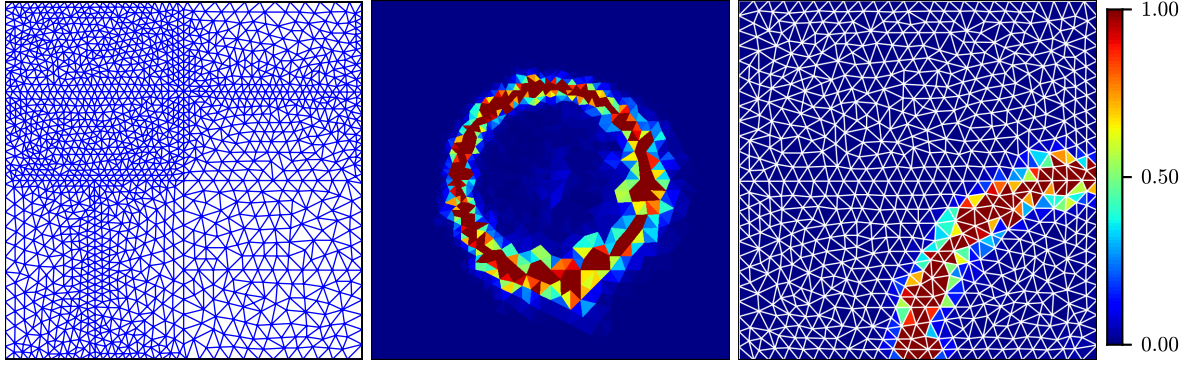


Figure 10: Mesh (left), discontinuity sensor (middle) and detail of the mesh and the discontinuity sensor at the second quadrant (right) in the Noh test for  $r = 5$  and  $h/r \simeq 0.0043$ .

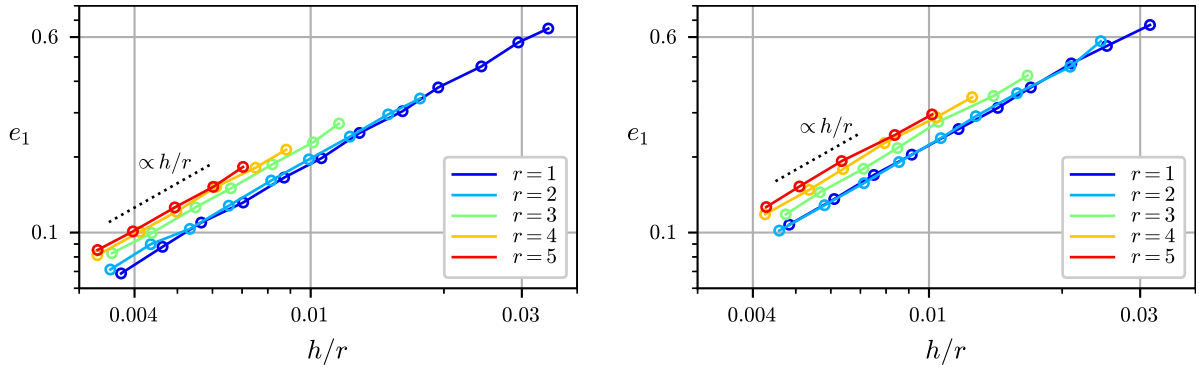


Figure 11:  $L^1$  errors in the Noh test for uniform (left) and non-uniform (right) meshes.

### 3.5. Two-dimensional Sedov explosion

The Sedov explosion is another classical test with radial symmetry. Initially,  $\rho^0 = 1$ ,  $v_i^0 = 0$  and the internal energy is zero everywhere except at the origin, where there is a delta source such that the total energy is  $E_{\text{total}} \simeq 0.98404$ . To avoid numerical problems, this initial energy field is regularized as

$$\mathcal{E}^0 = \begin{cases} \frac{10^{-9}}{\gamma - 1} + \frac{\pi E^{\text{total}}}{(\pi^2 - 4)\delta^2} \left[ 1 + \cos\left(\frac{\pi|\mathbf{x}|}{\delta}\right) \right], & \text{if } |\mathbf{x}| \leq \delta, \\ \frac{10^{-9}}{\gamma - 1}, & \text{if } |\mathbf{x}| > \delta, \end{cases}, \quad \delta = 4\frac{h}{r}.$$

This source of energy is transformed into kinetic energy through a shock wave which propagates radially outward. The theoretical solution can be estimated as shown in [89]. The same two versions as in the Noh test are considered –i.e., the same kind of meshes and the same boundary conditions–, although here the domains are respectively  $[0, 1.2]^2$  and  $[-1.2, 1.2]^2$ . We march in time with strategy (S2) until  $T = 1.0$  and the adiabatic constant is  $\gamma = 7/5$ .

Figs. 12 and 13 show the solutions for fifth-order elements –again, the solutions for  $r < 5$  are nearly identical and thus are not shown for brevity. The numerical solution of the WENO5 scheme [84] for the first version of the test is also plotted. As with Noh test, radial symmetry is preserved very well in the case of uniform meshes, and with reasonable accuracy in the case of non-uniform ones. In both cases, there are some slight oscillations before the shock, although they diminish as the mesh is refined. The maximum value for the density is very sensitive to  $h/r$ , and thus is somehow far from its theoretical value for the considered mesh sizes. Nevertheless, the results for this variable are very similar to those of WENO5 scheme, and it

---

is clear that finer meshes or anisotropic mesh refinement techniques would provide a more accurate result. Also note that in our method the velocity near the origin does not suffer from spurious oscillations. This is a pathology present in many methods, such as some WENO schemes –as shown in Fig. 12 and also in [87]–, which stems from the numerical divisions into a near-zero density –see also the comparison made by [90] for different methods. Finally, it can be checked that the discontinuity sensor is active only at the shock.

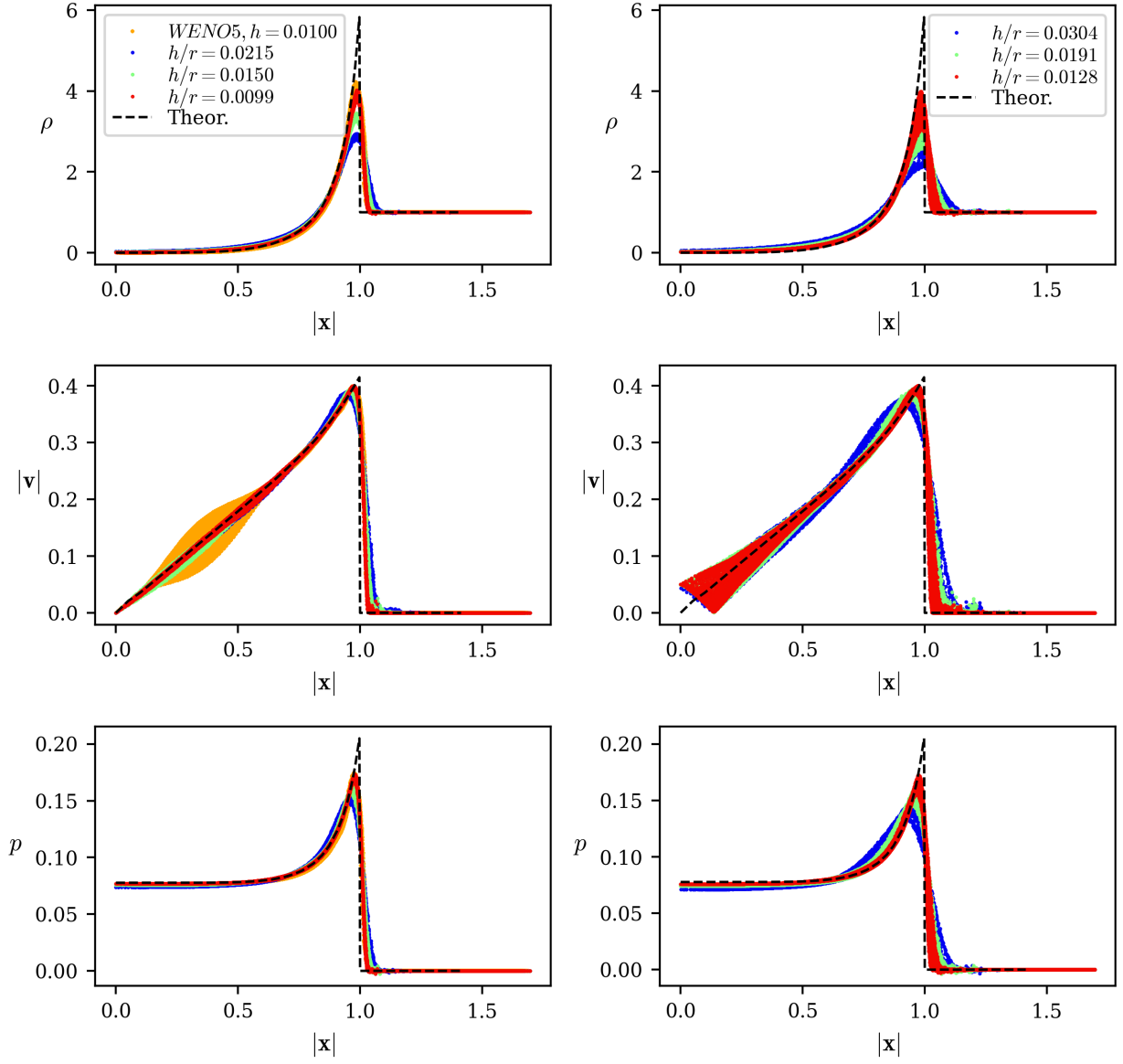


Figure 12: Numerical solution of the Sedov test at  $t = 1$  in the domain  $[0, 1.2]^2$  (left) and in the domain  $[-1.2, 1.2]^2$  (right) for  $r = 5$ .

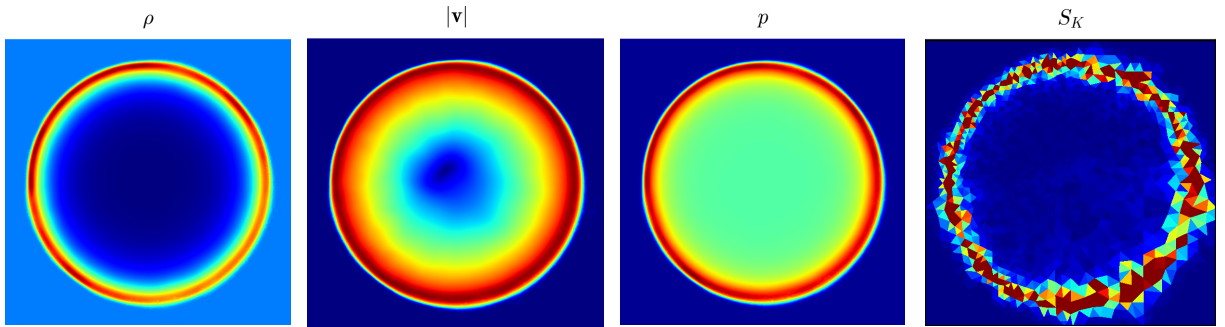


Figure 13: Numerical solution of the Sedov test at  $t = 1$  in the domain  $[-1.2, 1.2]^2$  for  $r = 5$  and  $h/r \approx 0.0128$ .



Furthermore, Fig. 14 shows the errors in the conservation of mass, momentum and energy for the case of non-uniform meshes. As can be seen, the errors are considerably higher for linear elements, whereas for  $r \geq 2$  the results are of the same order of magnitude.

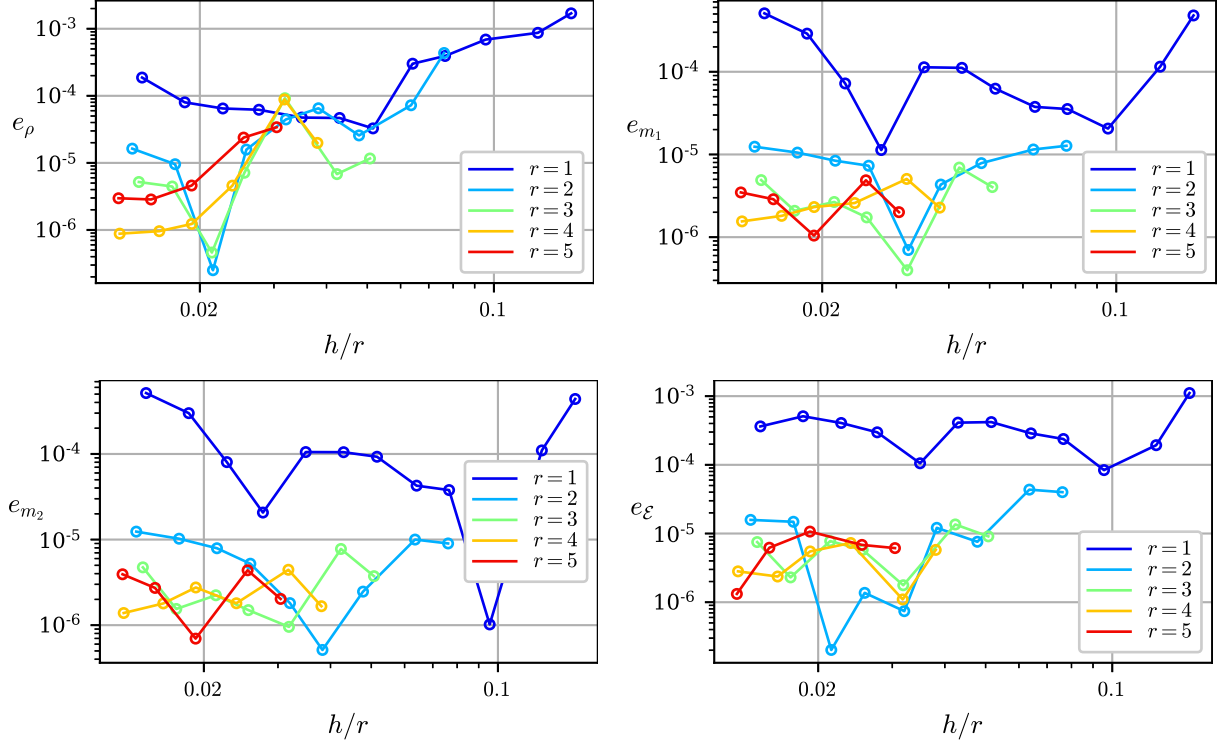


Figure 14: Conservation errors in Sedov test with non-uniform meshes.

### 3.6. Single-material triple-point problem

In this test, the rectangular domain  $[-1, 6] \times [-1.5, 1.5]$  is filled with a gas of constant  $\gamma = 7/5$  at the three different states shown in Fig. 15. The gas is initially at rest in all the domain. Non-penetration conditions are imposed on the entire boundary and time-marching strategy (S1) is employed until  $T = 5$ .

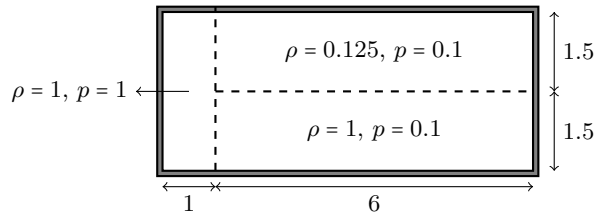
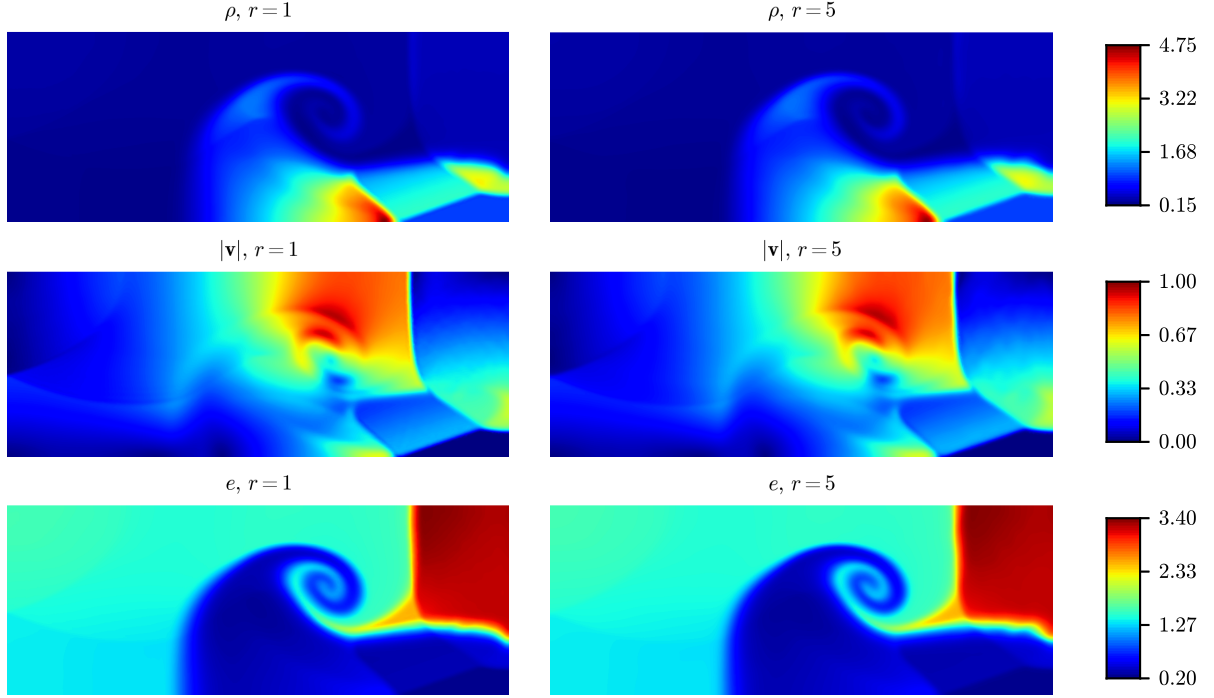


Figure 15: Initial conditions for the single-material triple-point test.

Fig. 16 shows the numerical solution for  $h/r \simeq 0.027$ . The results are nearly identical for first and fifth-order elements. The only differences are seen at the vortex, which is slightly thinner in the case  $r = 5$ , and at the Kelvin–Helmholtz instabilities in the horizontal contact discontinuity, which are slightly more visible for  $r = 5$ . The solution is very good agreement with [45, 91].

Figure 16: Numerical solution of the single-material, triple-point test at  $t = 5$  for  $h/r \simeq 0.027$ .

In addition, Table 3 shows the errors in mass and energy conservation for  $h \simeq 0.027$  and  $r \leq 5$ . As can be seen, they vary from  $\mathcal{O}(10^{-5})$  to  $\mathcal{O}(10^{-8})$  and are slightly smaller for high-order elements.

Table 3: Errors in the conservation of mass and energy for the single-material triple-point test and for  $h/r \simeq 0.027$ .

	$r = 1$	$r = 2$	$r = 3$	$r = 4$	$r = 5$
$e_\rho$	8.824E-05	3.169E-06	1.128E-07	8.785E-08	1.137E-06
$e_\mathcal{E}$	8.207E-05	5.247E-06	1.192E-08	2.628E-07	4.987E-07

### 3.7. Rayleigh–Taylor instability

In this test, a gas of density  $\rho^0 = 2$  rests on top of a lighter gas of density  $\rho^0 = 1$  in static equilibrium under the presence of a gravitational field. The domain is the rectangle  $[0, 0.125] \times [-0.5, 0.5]$ , the interface between the fluids is placed at  $x_2 = 0$ ,  $\gamma = 5/3$  for both gases, the gravitational acceleration is  $g = 1$  and the initial pressure is hydrostatic, with  $p^0(x_2 = 0.5) = 1$ . The equilibrium is broken by a perturbation in the velocity field of the form

$$v_1^0 = 0, \quad v_2^0 = 0.03 [\exp(-2\pi x_2^2) - \exp(\pi/2)] \cos(8\pi x_1).$$

Non-penetration boundary conditions are applied on the entire boundary and strategy (S1) is employed to march in time until  $T = 1.95$ .

Figure 17 shows the results for several values of  $h/r$  and  $r$ . As can be seen, the (physically admissible) Kelvin–Helmholtz instabilities are more visible as the mesh is refined, as expected [92]. The value of  $r$  does not affect sensitively to the results, although it could be said that higher  $r$  leads to slightly more vortical solutions. The results compare well to those of [93].

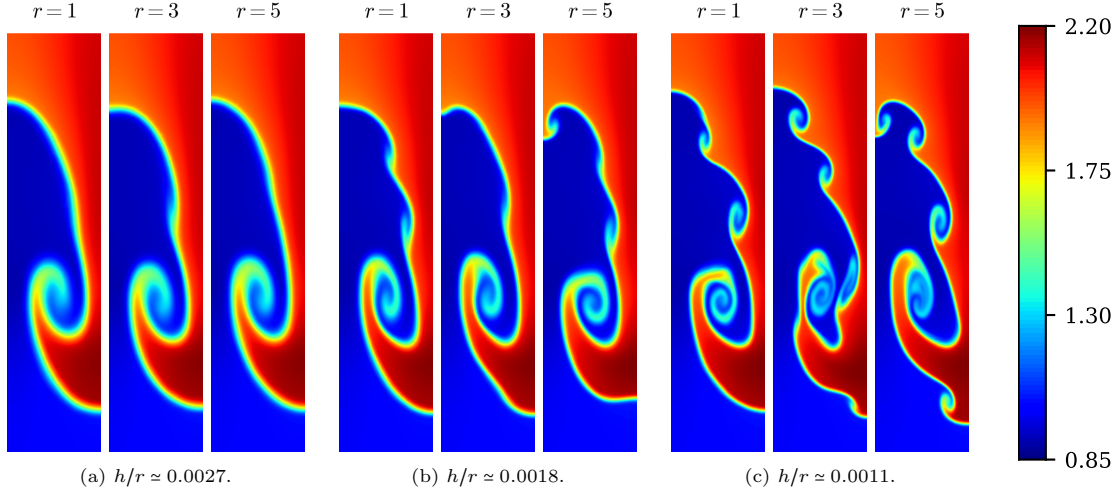


Figure 17: Density field in the Rayleigh–Taylor instability at  $t = 1.95$ . For visualization purposes, we plot the solution only at the subdomain  $[0, 0.125] \times [-0.25, 0.25]$ .

On the other hand, Table 4 shows the mass conservation errors for the considered space discretizations. In this case, these are significantly larger for first-order elements, whereas for  $r \geq 2$  they are of the order of  $10^{-7}$  and  $10^{-8}$ .

Table 4: Mass conservation errors,  $e_\rho$ , in the Rayleigh–Taylor instability.

	$r = 1$	$r = 2$	$r = 3$	$r = 4$	$r = 5$
$h/r \simeq 0.0027$	2.760E-05	9.091E-08	9.874E-08	3.160E-07	1.307E-07
$h/r \simeq 0.0018$	2.278E-05	3.368E-07	7.500E-08	2.293E-07	1.028E-08
$h/r \simeq 0.0011$	2.337E-05	2.171E-07	9.300E-08	1.289E-07	6.026E-08

### 3.8. Shock–vortex interaction

This test considers the interaction between a steady planar shock wave and a rotating vortex [94] in the domain  $[0, 2] \times [0, 1]$ . In particular, the shock is placed at  $x_1 = 0.5$  and is defined by the upstream conditions  $p_\infty = 1$ ,  $\rho_\infty = 1$  and  $\mathbf{v}_\infty = [1.5c_\infty, 0]$ . On the other hand, the vortex is initially centered at  $(0.25, 0.5)$  and is of radius  $b = 0.175$ . The velocity field inside the latter is given by  $\mathbf{v} = \mathbf{v}_\infty + v_\theta \mathbf{e}_\theta$ , where

$$v_\theta = \begin{cases} v_m \frac{r}{a}, & \text{if } r \leq a, \\ v_m \frac{a}{a^2 - b^2} \left( r - \frac{b^2}{r} \right), & \text{if } a < r \leq b, \end{cases}$$

$r$  is the distance to the vortex center,  $\mathbf{e}_\theta$  is the unit vector along the counter-clockwise azimuthal direction (around the vortex center),  $a = 0.075$  is a constant, and  $v_m = 0.9c_\infty$  is the maximum tangential velocity of the vortex. The internal energy and the density inside the vortex are obtained from

$$e = e_\infty - \frac{1}{\gamma} \int_r^b \frac{v_\theta(r')^2}{r'} dr', \quad \rho = \rho_\infty \left( \frac{e}{e_\infty} \right)^{\frac{1}{\gamma-1}}.$$

The adiabatic constant is  $\gamma = 7/5$  and the initial conditions downstream the shock are determined by the well-known Rankine–Hugoniot relations. Normal velocity boundary conditions are imposed on the entire boundary and time-marching strategy (S1) is employed until the final time  $T = 0.7$ .

Fig. 18 shows the solution for  $r = 5$  and  $h/r \simeq 0.0021$ . As can be seen, the vortex moves downstream and crosses the shock. Then, the shock is slightly distorted and the vortex is compressed into an elliptical

524 shape, eventually breaking into two separated vortices. In addition, a complex structure of acoustic waves  
 525 develops after the shock front. The results compare very well with those of [68, 94–96].

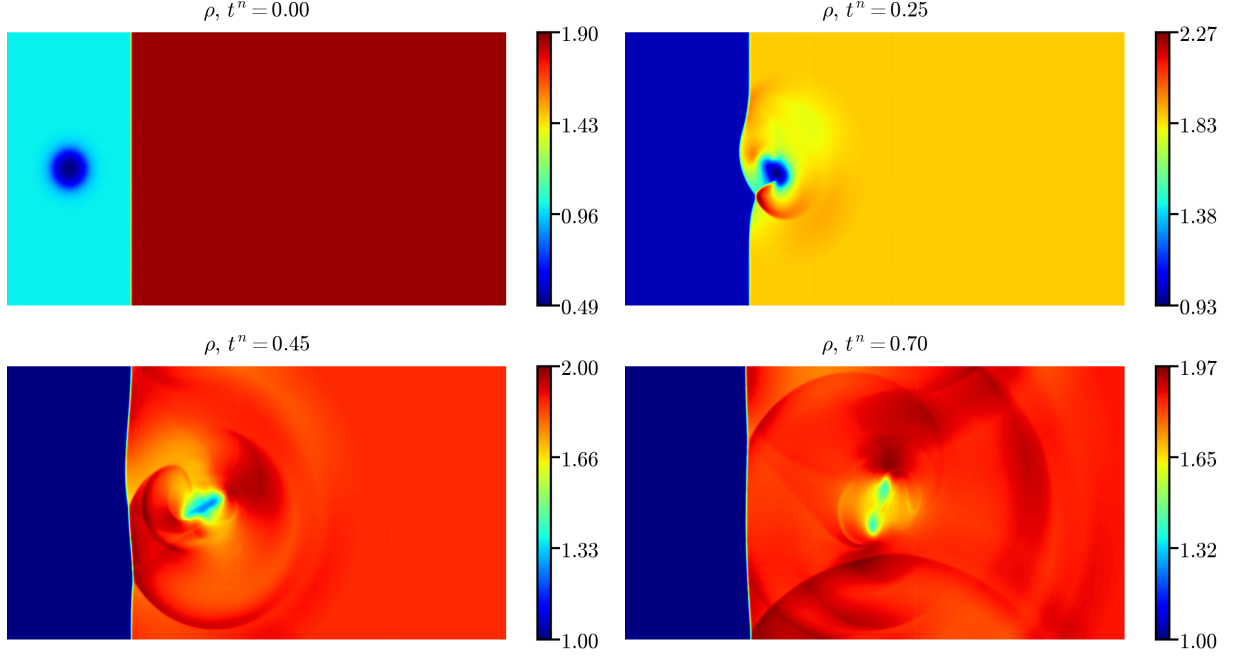


Figure 18: Numerical solution of the shock–vortex interaction problem for  $r = 5$  and  $h/r \approx 0.0021$ .

### 526 3.9. Taylor–Green vortex

527 As final benchmark, we solve the Taylor–Green vortex problem to check if the method performs well  
 528 when the solution is smooth. This test is defined by the initial conditions

$$\rho^0 = 1, \quad v_1^0 = \sin(\pi x_1) \cos(\pi x_2), \quad v_2^0 = -\cos(\pi x_1) \sin(\pi x_2), \quad p^0 = 1 + 0.25 [\cos(2\pi x_1) + \cos(2\pi x_2)],$$

529 and a constant-in-time heat source per unit of volume

$$Q = \frac{3\pi}{8} [\cos(3\pi x_1) \cos(\pi x_2) - \cos(\pi x_1) \cos(3\pi x_2)].$$

530 The domain is the square  $[0, 1]^2$ , non-penetration boundary conditions are prescribed on the entire boundary  
 531 and  $\gamma = 5/3$ . The test is run up to  $T = 0.5$  with strategy (S1). The exact solution is  $\rho(\mathbf{x}, t) = \rho^0(\mathbf{x})$ ,  
 532  $m_i(\mathbf{x}, t) = m_i^0(\mathbf{x})$  and  $\mathcal{E}(\mathbf{x}, t) = \mathcal{E}^0(\mathbf{x})$ ; see Fig. 19.

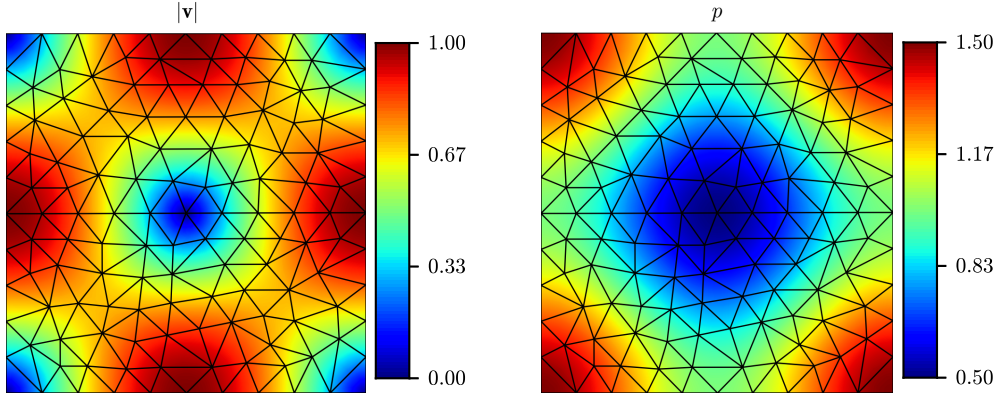


Figure 19: Numerical solution of the Taylor–Green test at  $t = 0.5$  for  $r = 5$  and  $h/r \simeq 0.020$ .

As shown in Fig. 20, the  $L^1$  errors are smaller and converge much faster with  $h/r$  for higher-order elements, as expected. It is also seen that these errors are slightly larger when the DC operators are applied, although the convergence rate –shown in Table 5– is still very good. On the other hand, the conservation errors in mass and energy are nearly identical with and without DC, whereas the momentum conservation errors are slightly larger with DC. In contrast with previous tests, here fifth-order elements clearly yield smaller conservation errors.

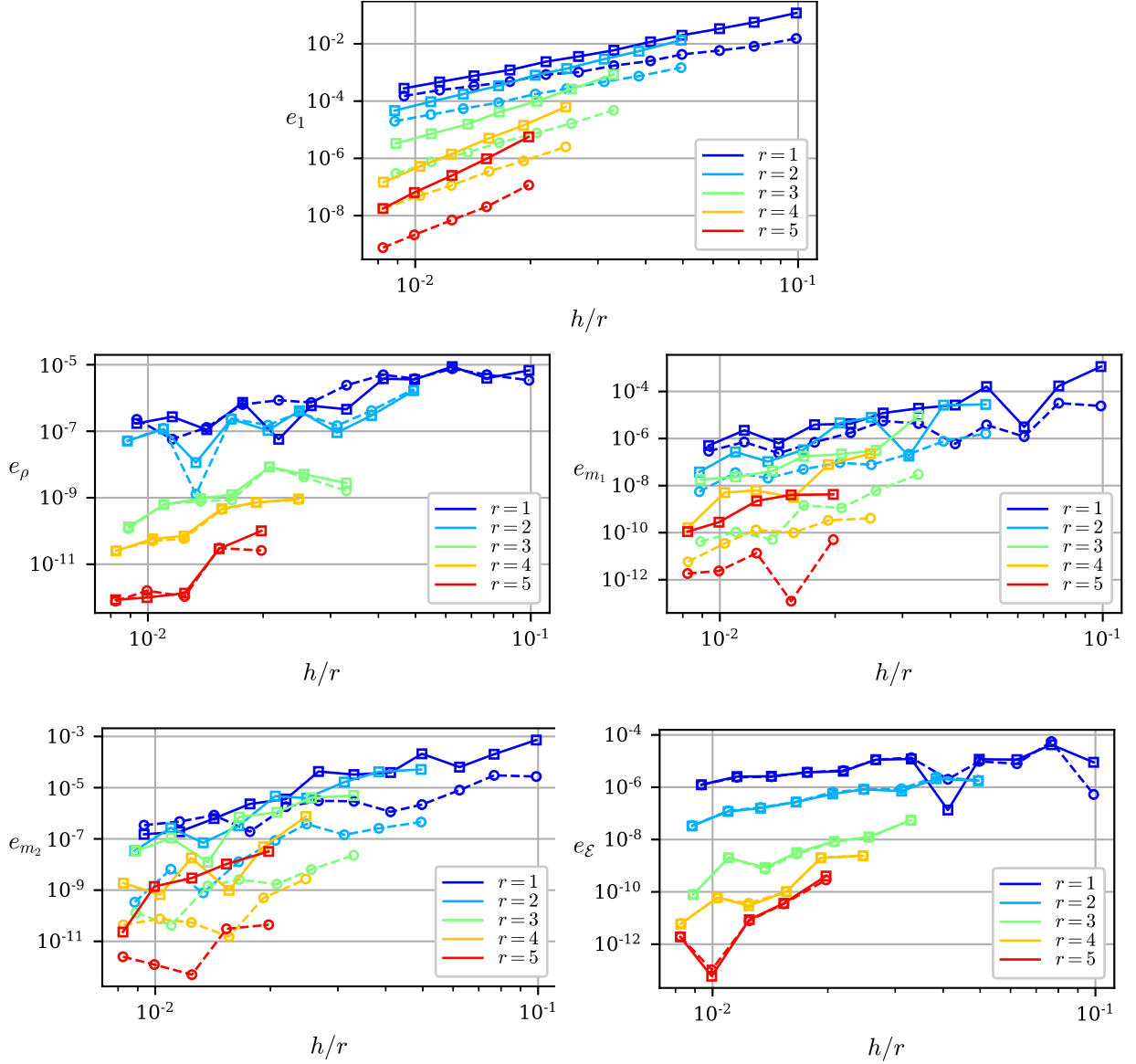


Figure 20: Numerical errors in Taylor–Green test without DC (dashed lines) and with DC (solid lines).

Table 5: Mean convergence rate of the  $e_1$  error with respect to  $h/r$  in the Taylor–Green test.

	$r = 1$	$r = 2$	$r = 3$	$r = 4$	$r = 5$
Without DC	1.94	2.51	3.81	4.54	5.65
With DC	2.57	3.28	4.26	5.47	6.51

### 3.10. Profiling times

Next we address the computer times required to solve some of the experiments above in a 64-core AMD Epyc 7601 CPU 2.20 GHz machine with 260GB of RAM (one experiment per core). For instance, Table 6 shows the total CPU time per simulated instant for the case of the Noh test with non-uniform meshes, as well as the time devoted per simulated instant to the resolution of the linear systems and to the computation of

the  $f$  and  $g$  terms defined in Eqs. (13)-(14) and (18)-(19), which result to be by far the two most expensive tasks. Note that each simulated instant requires the resolution of seven internal stages for the considered ARK4(3)7L[2]SA1 Runge–Kutta scheme [78]. As can be seen, for the same number of nodes, the resolution of the linear systems is less expensive for low-order elements, as the matrices are more sparse, whereas the computation of the  $f$  and  $g$  terms is less expensive for high-order elements, as the total number of quadrature nodes in the mesh (proportional to the number of elements and to number of quadrature nodes per element detailed in Table 1) is smaller. As a consequence, the total time does not result to be very sensitive to  $r$  whenever the number of nodes remains the same. Also note that the algorithm complexity is approximately 1.5 irrespective of  $r$ , and that the number of iterations in the Anderson method increases with  $r$ , but remains constant with  $h/r$ .

Table 6: Profiling times (in seconds) per simulated instant for the RK-FLG scheme in the Noh test.

$r$	$h/r$	Elements	Nodes	Mean nb. of Anderson iterations	Total	Rate w.r.t. nb. of nodes	Resolution of linear systems	Computation of $f$ and $g$ terms
1	0.0117	10782	5524	36.06	10.30	-	0.69	7.75
1	0.0075	25716	13063	35.25	28.09	1.59	1.80	21.79
1	0.0048	62650	31644	34.49	76.08	1.59	5.31	58.31
2	0.0106	3200	6545	40.07	10.42	-	1.43	6.72
2	0.0071	7216	14649	42.18	27.08	1.64	3.45	18.40
2	0.0046	17468	35273	44.04	66.40	1.46	9.43	45.51
3	0.0105	1450	6670	48.69	13.46	-	1.87	8.36
3	0.0071	3200	14617	50.53	31.24	1.51	5.02	19.61
3	0.0047	7216	32797	50.09	74.57	1.56	12.04	47.72
4	0.0104	826	6753	56.26	10.66	-	2.76	4.90
4	0.0064	2256	18289	58.09	36.15	1.68	9.10	18.40
4	0.0043	5030	40601	57.51	79.18	1.45	20.32	40.46
5	0.0102	570	7276	64.02	14.92	-	4.24	6.63
5	0.0064	1450	18366	64.10	46.74	1.68	12.94	23.68
5	0.0043	3200	40361	62.97	99.57	1.43	28.89	49.33

In addition, Fig. 21 shows the relation between the  $L^1$  error and the total CPU time. It is seen that, for discontinuous solutions, the performance is approximately the same for low and high-order elements. This is due to the fact that similar values of  $h/r$  lead both to similar CPU times and to similar  $L^1$  errors, as discussed previously. However, for smooth solutions, similar values of  $h/r$  lead to smaller  $L^1$  errors for high-order elements and to similar CPU times. Thus, high-order elements perform much better in this case.

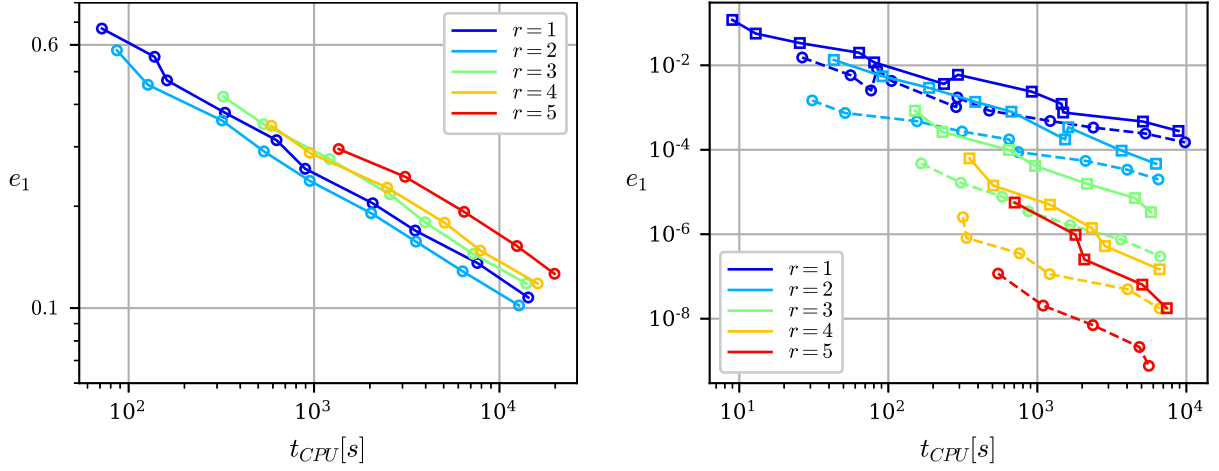


Figure 21:  $L^1$  errors versus the total CPU time (in seconds) for the Noh test with non-uniform mesh (left) and for the Taylor–Green test (right) with DC (solid lines) and without DC (dashed lines).

#### 4. Conclusions

In this work, we present a novel Lagrange–Galerkin method for the resolution of compressible and inviscid flows. As main contributions, the method considers high-order continuous space discretizations on unstructured triangular meshes and is (nearly) conservative irrespective of the order of the time-marching algorithm. Acoustic instabilities are killed via subgrid stabilization techniques, whereas discontinuities are first detected from the values of the fine-scale terms of a previously computed non-stabilized solution, and are then captured via artificial diffusion. Both the subgrid stabilization and the discontinuity capturing operators are based on Brenner’s model [51] for compressible and viscous flows.

The scheme has been tested on several benchmark problems using up to fifth-order elements and a fourth-order Runge–Kutta method. As expected, numerical experiments show that, for the same number of nodes, high-order elements provide much smaller  $L^1$  errors than low-order elements when the solution is smooth, and similar  $L^1$  errors when the solution is discontinuous. In addition, high-order elements present smaller conservation errors and require approximately the same CPU time.

The method marches in time with an implicit–explicit Runge–Kutta scheme. Thus, a non-linear system of equations has to be solved at each stage, which is accomplished by means of a simple, fixed-point iteration accelerated via Anderson’s method [69]. Although the number of required iterations is not very high, avoiding the resolution of this nonlinear system could clearly improve the efficiency of the method. For that purpose, Rosenbrock–Wanner methods [97–99] or LIRK-W methods [100], which require only the resolution of an implicit linear system at each stage, could be implemented. Other works which could be addressed in the future are the implementation of anisotropic mesh refinement techniques [44, 101–105] and of alternative remapping techniques [45, 46], the enforcement of exact conservation via constrained iterative solvers [71, 72] and the extension of the method to three-dimensional and/or large-scale computations.

#### Acknowledgements

This research has been partially funded by project PGC-2018-097565-BI00 from the “Ministerio de Ciencia, Innovación y Universidades” of Spain and the European Regional Development Fund, and by the FPU16/05509 grant to M. Colera from the “Ministerio de Ciencia, Innovación y Universidades” of Spain. The authors acknowledge Professors J.L. Prieto, P. Galán del Sastre, L. Saavedra and L. Sanz for many kind and helpful discussions, as well as two anonymous referees for their insightful comments.



## Appendix A. Derivation of formula (32)

Let  $u(\mathbf{x}, t)$  be a continuous variable. We define at each element  $K$  a characteristic length  $l_K$  such that

$$\max_{i=0,\dots,r+1} \left| \frac{\partial^{r+1} u}{\partial x_1^{r+1-i} \partial x_2^i} \right|_K = \frac{\llbracket u \rrbracket_{2,\Omega}}{l_K^{r+1}}.$$

That is,  $l_K$  represents a length scale associated with the local variations in  $u$ . If  $u_h$  is the finite element approximation to  $u$ , from approximation theory and the fact that  $\llbracket v \rrbracket_{2,K} \leq \|v\|_{\infty,K}$ , we have that the error satisfies

$$e_{u,K} := \frac{\llbracket u - u_h \rrbracket_{2,K}}{\llbracket u \rrbracket_{2,\Omega}} \sim \frac{\llbracket D^{r+1} u \rrbracket_{2,K} h_K^{r+1}}{\llbracket u \rrbracket_{2,\Omega} (r+1)!} \sim \frac{h_K^{r+1}}{\llbracket u \rrbracket_{2,\Omega} (r+1)!} \max_{i=0,\dots,r+1} \left| \frac{\partial^{r+1} u}{\partial x_1^{r+1-i} \partial x_2^i} \right|_K \sim \frac{1}{(r+1)!} \left( \frac{h_K}{l_K} \right)^{r+1}.$$

We assume that the solution presents oscillations when  $l_K$  is small compared with the distance between nodes  $h_K/r$ . Hence, we define the threshold length  $l_K^* = C_{DC} h_K/r$  and the corresponding threshold error

$$e^* := \frac{1}{(r+1)!} \left( \frac{r}{C_{DC}} \right)^{r+1},$$

and set  $S_K = 1$  whenever  $l_K \leq l_K^*$  (or  $e_{u,K} \geq e^*$ ), i.e.,

$$S_K = \min \left\{ \frac{e_{u,K}}{e^*}, 1 \right\} = \min \left\{ (r+1)! \left( \frac{C_{DC}}{r} \right)^{r+1} e_{u,K}, 1 \right\}.$$

Eq. (32) is the straightforward generalization of the latter formula to the case of multiple variables.

## References

- [1] J. D. Anderson Jr., Fundamentals of Aerodynamics, sixth Edition, McGraw-Hill Education, 2016.  
URL <https://www.mheducation.com/highered/product/fundamentals-aerodynamics-anderson/M9781259129919.html>
- [2] M. P. Boyce, Gas Turbine Engineering Handbook, Elsevier, 2012. doi:10.1016/C2009-0-64242-2.
- [3] J. H. Lienhard IV, J. H. Lienhard V, A Heat Transfer Textbook, 5th Edition, Dover Publications, Mineola, N.Y., 2019.  
URL <http://ahtt.mit.edu>
- [4] J. Donea, A. Huerta, J.-P. Ponthot, A. Rodríguez-Ferrán, Arbitrary Lagrangian–Eulerian methods, in: E. Stein, R. de Borst, T. J. R. Hughes (Eds.), Encyclopedia of Computational Mechanics, John Wiley, Chichester, West Sussex, 2004.
- [5] V. A. Dobrev, T. E. Ellis, T. V. Kolev, R. N. Rieben, Curvilinear finite elements for Lagrangian hydrodynamics, International Journal for Numerical Methods in Fluids 65 (11-12) (2011) 1295–1310. doi:10.1002/flid.2366.
- [6] V. A. Dobrev, T. V. Kolev, R. N. Rieben, High-Order Curvilinear Finite Element Methods for Lagrangian Hydrodynamics, SIAM Journal on Scientific Computing 34 (5) (2012) B606–B641. doi:10.1137/120864672.
- [7] G. Scovazzi, M. Christon, T. Hughes, J. Shadid, Stabilized shock hydrodynamics: I. A Lagrangian method, Computer Methods in Applied Mechanics and Engineering 196 (4-6) (2007) 923–966. doi:10.1016/j.cma.2006.08.008.
- [8] G. Scovazzi, Stabilized shock hydrodynamics: II. Design and physical interpretation of the SUPG operator for Lagrangian computations, Computer Methods in Applied Mechanics and Engineering 196 (4) (2007) 967–978. doi:10.1016/j.cma.2006.08.009.
- [9] G. Scovazzi, E. Love, M. Shashkov, Multi-scale Lagrangian shock hydrodynamics on Q1/P0 finite elements: Theoretical framework and two-dimensional computations, Computer Methods in Applied Mechanics and Engineering 197 (9-12) (2008) 1056–1079. doi:10.1016/j.cma.2007.10.002.
- [10] G. Scovazzi, J. Shadid, E. Love, W. Rider, A conservative nodal variational multiscale method for Lagrangian shock hydrodynamics, Computer Methods in Applied Mechanics and Engineering 199 (49-52) (2010) 3059–3100. doi:10.1016/j.cma.2010.03.027.
- [11] G. Scovazzi, Lagrangian shock hydrodynamics on tetrahedral meshes: A stable and accurate variational multiscale approach, Journal of Computational Physics 231 (24) (2012) 8029–8069. doi:10.1016/j.jcp.2012.06.033.
- [12] X. Zeng, G. Scovazzi, A variational multiscale finite element method for monolithic ALE computations of shock hydrodynamics using nodal elements, Journal of Computational Physics 315 (2016) 577–608. doi:10.1016/j.jcp.2016.03.052.
- [13] J.-L. Guermond, B. Popov, V. Tomov, Entropy–viscosity method for the single material Euler equations in Lagrangian frame, Computer Methods in Applied Mechanics and Engineering 300 (2016) 402–426. doi:10.1016/j.cma.2015.11.009.

- [14] J.-L. Guermond, B. Popov, L. Saavedra, Y. Yang, Invariant Domains Preserving Arbitrary Lagrangian Eulerian Approximation of Hyperbolic Systems with Continuous Finite Elements, *SIAM Journal on Scientific Computing* 39 (2) (2017) A385–A414. doi:10.1137/16M1063034.
- [15] J.-L. Guermond, B. Popov, L. Saavedra, Second-order invariant domain preserving ALE approximation of hyperbolic systems, *Journal of Computational Physics* 401 (2020) 108927. doi:10.1016/j.jcp.2019.108927.
- [16] M. Cremonesi, A. Frangi, A Lagrangian finite element method for 3D compressible flow applications, *Computer Methods in Applied Mechanics and Engineering* 311 (2016) 374–392. doi:10.1016/j.cma.2016.08.005.
- [17] W. Boscheri, M. Dumbser, Arbitrary-Lagrangian–Eulerian Discontinuous Galerkin schemes with a posteriori subcell finite volume limiting on moving unstructured meshes, *Journal of Computational Physics* 346 (2017) 449–479. doi:10.1016/j.jcp.2017.06.022.
- [18] R. Abgrall, S. Tokareva, Staggered Grid Residual Distribution Scheme for Lagrangian Hydrodynamics, *SIAM Journal on Scientific Computing* 39 (5) (2017) A2317–A2344. doi:10.1137/16M1078781.
- [19] R. Abgrall, K. Lipnikov, N. Morgan, S. Tokareva, Multidimensional Staggered Grid Residual Distribution Scheme for Lagrangian Hydrodynamics, *SIAM Journal on Scientific Computing* 42 (1) (2020) A343–A370. doi:10.1137/18M1223939.
- [20] M. Lentine, J. T. Grétarsson, R. Fedkiw, An unconditionally stable fully conservative semi-Lagrangian method, *Journal of Computational Physics* 230 (8) (2011) 2857–2879. doi:10.1016/j.jcp.2010.12.036.
- [21] J. Donea, L. Quartapelle, An introduction to finite element methods for transient advection problems, *Computer Methods in Applied Mechanics and Engineering* 95 (2) (1992) 169–203. doi:10.1016/0045-7825(92)90139-B.
- [22] R. Bermejo, L. Saavedra, Lagrange–Galerkin methods for the incompressible Navier–Stokes equations: A review, *Communications in Applied and Industrial Mathematics* 7 (3) (2016) 26–55. doi:10.1515/caim-2016-0021.
- [23] K. Futai, N. Kolbe, H. Notsu, T. Suzuki, A Mass-Preserving Two-Step Lagrange–Galerkin Scheme for Convection-Diffusion Problems, *Journal of Scientific Computing* 92 (2) (2022) 37. doi:10.1007/s10915-022-01885-w.
- [24] A. Oliveira, A. M. Baptista, A comparison of integration and interpolation Eulerian–Lagrangian methods, *International Journal for Numerical Methods in Fluids* 21 (3) (1995) 183–204. doi:10.1002/flid.1650210302.
- [25] J. P. Benqué, J. Ronat, Quelques difficultés des modèles numériques en hydraulique, in: *Proceedings of the Fifth International Symposium on Computing Methods in Applied Sciences and Engineering*, INRIA, Versailles, France, 1981.
- [26] J. P. Benqué, B. Ibler, A. Keramsi, G. Labadie, A finite element method for Navier–Stokes equations coupled with a temperature equation, in: *Proceedings of the Fourth International Symposium on Finite Elements in Flow Problems*, North-Holland, Amsterdam, 1982.
- [27] F. X. Giraldo, The Lagrange–Galerkin method for the two-dimensional shallow water equations on adaptive grids, *International Journal for Numerical Methods in Fluids* 33 (6) (2000) 789–832. doi:10.1002/1097-0363(20000730)33:6<789::AID-FLD29>3.0.CO;2-1.
- [28] F. X. Giraldo, Strong and weak Lagrange–Galerkin spectral element methods for the shallow water equations, *Computers & Mathematics with Applications* 45 (1-3) (2003) 97–121. doi:10.1016/S0898-1221(03)80010-X.
- [29] M. Kaazempur-Mofrad, C. Ethier, An efficient characteristic Galerkin scheme for the advection equation in 3-D, *Computer Methods in Applied Mechanics and Engineering* 191 (46) (2002) 5345–5363. doi:10.1016/S0045-7825(02)00461-9.
- [30] M. Kaazempur-Mofrad, P. Mineev, C. Ethier, A characteristic/finite element algorithm for time-dependent 3-D advection-dominated transport using unstructured grids, *Computer Methods in Applied Mechanics and Engineering* 192 (11-12) (2003) 1281–1298. doi:10.1016/S0045-7825(02)00627-8.
- [31] M. Colera, J. Carpio, R. Bermejo, A nearly-conservative high-order Lagrange–Galerkin method for the resolution of scalar convection-dominated equations in non-divergence-free velocity fields, *Computer Methods in Applied Mechanics and Engineering* 372 (2020) 113366. doi:10.1016/j.cma.2020.113366.
- [32] M. Colera, J. Carpio, R. Bermejo, A nearly-conservative, high-order, forward Lagrange–Galerkin method for the resolution of scalar hyperbolic conservation laws, *Computer Methods in Applied Mechanics and Engineering* 376 (2021) 113654. doi:10.1016/j.cma.2020.113654.
- [33] H. Wang, H. K. Dahle., R. E. Ewing, M. S. Espedal, R. C. Sharpley, S. Man, An ELLAM Scheme for Advection-Diffusion Equations in Two Dimensions, *SIAM Journal on Scientific Computing* 20 (6) (1999) 2160–2194. doi:10.1137/S1064827596309396.
- [34] M. Al-Lawatia, A Higher-Order Eulerian–Lagrangian Localized Adjoint Method for Two-Dimensional Unsteady Advection-Diffusion Problems, *Journal of Computational Mathematics* 30 (3) (2012) 324–336. doi:10.4208/jcm.1110-m3465.
- [35] R. J. Purser, L. M. Leslie, An Efficient Semi-Lagrangian Scheme Using Third-Order Semi-Implicit Time Integration and Forward Trajectories, *Monthly Weather Review* 122 (4) (1994) 745–756. doi:10.1175/1520-0493(1994)122<0745:AESLSU>2.0.CO;2.
- [36] H.-M. Henry Juang, S.-Y. Hong, Forward Semi-Lagrangian Advection with Mass Conservation and Positive Definiteness for Falling Hydrometeors, *Monthly Weather Review* 138 (5) (2010) 1778–1791. doi:10.1175/2009MWR3109.1.
- [37] N. Crouseilles, T. Respaud, E. Sonnendrücker, A forward semi-Lagrangian method for the numerical solution of the Vlasov equation, *Computer Physics Communications* 180 (10) (2009) 1730–1745. doi:10.1016/j.cpc.2009.04.024.
- [38] D. Sirajuddin, W. N. Hitchon, A truly forward semi-Lagrangian WENO scheme for the Vlasov–Poisson system, *Journal of Computational Physics* 392 (2019) 619–665. doi:10.1016/j.jcp.2019.04.054.
- [39] R. Bermejo, P. Galán del Sastre, L. Saavedra, A Second Order in Time Modified Lagrange–Galerkin Finite Element Method for the Incompressible Navier–Stokes Equations, *SIAM Journal on Numerical Analysis* 50 (6) (2012) 3084–3109. doi:10.1137/11085548X.
- [40] R. Bermejo, L. Saavedra, Modified Lagrange–Galerkin Methods to Integrate Time Dependent Incompressible Navier–Stokes Equations, *SIAM Journal on Scientific Computing* 37 (6) (2015) B779–B803. doi:10.1137/140973967.

- [41] R. Bermejo, L. Saavedra, A second order in time local projection stabilized Lagrange–Galerkin method for Navier–Stokes equations at high Reynolds numbers, *Computers & Mathematics with Applications* 72 (4) (2016) 820–845. doi:10.1016/j.camwa.2016.05.012.
- [42] F. X. Giraldo, T. Warburton, A nodal triangle-based spectral element method for the shallow water equations on the sphere, *Journal of Computational Physics* 207 (1) (2005) 129–150. doi:10.1016/j.jcp.2005.01.004.
- [43] J. Carpio, J. L. Prieto, M. Vera, A local anisotropic adaptive algorithm for the solution of low-Mach transient combustion problems, *Journal of Computational Physics* 306 (2016) 19–42. doi:10.1016/j.jcp.2015.11.011.
- [44] J. Carpio, J. L. Prieto, P. Galán del Sastre, An anisotropic adaptive, Lagrange–Galerkin numerical method for spray combustion, *Journal of Computational Physics* 381 (2019) 246–274. doi:10.1016/j.jcp.2018.12.022.
- [45] X. Zeng, G. Scovazzi, A frame-invariant vector limiter for flux corrected nodal remap in arbitrary Lagrangian–Eulerian flow computations, *Journal of Computational Physics* 270 (2014) 753–783. doi:10.1016/j.jcp.2014.03.054.
- [46] R. W. Anderson, V. A. Dobrev, T. V. Kolev, R. N. Rieben, Monotonicity in high-order curvilinear finite element arbitrary Lagrangian–Eulerian remap, *International Journal for Numerical Methods in Fluids* 77 (5) (2015) 249–273. doi:10.1002/flid.3965.
- [47] G. E. Barter, D. L. Darmofal, Shock capturing with PDE-based artificial viscosity for DGFEM: Part I. Formulation, *Journal of Computational Physics* 229 (5) (2010) 1810–1827. doi:10.1016/j.jcp.2009.11.010.
- [48] J. F. Kelly, F. X. Giraldo, Continuous and discontinuous Galerkin methods for a scalable three-dimensional nonhydrostatic atmospheric model: Limited-area mode, *Journal of Computational Physics* 231 (24) (2012) 7988–8008. doi:10.1016/j.jcp.2012.04.042.
- [49] J. Freund, The space-continuous–discontinuous Galerkin method, *Computer Methods in Applied Mechanics and Engineering* 190 (26) (2001) 3461–3473. doi:10.1016/S0045-7825(00)00279-6.
- [50] A. Cangiani, J. Chapman, E. Georgoulis, M. Jensen, On the Stability of Continuous–Discontinuous Galerkin Methods for Advection–Diffusion–Reaction Problems, *Journal of Scientific Computing* 57 (2) (2013) 313–330. doi:10.1007/s10915-013-9707-y.
- [51] H. Brenner, Fluid mechanics revisited, *Physica A: Statistical Mechanics and its Applications* 370 (2) (2006) 190–224. doi:10.1016/j.physa.2006.03.066.
- [52] M. Nazarov, A. Larcher, Numerical investigation of a viscous regularization of the Euler equations by entropy viscosity, *Computer Methods in Applied Mechanics and Engineering* 317 (2017) 128–152. doi:10.1016/j.cma.2016.12.010.
- [53] E. Feireisl, M. Lukáčová-Medvid’ová, H. Mizerová, A finite volume scheme for the Euler system inspired by the two velocities approach, *Numerische Mathematik* 144 (1) (2020) 89–132. doi:10.1007/s00211-019-01078-y.
- [54] J.-L. Guermond, B. Popov, Viscous Regularization of the Euler Equations and Entropy Principles, *SIAM Journal on Applied Mathematics* 74 (2) (2014) 284–305. doi:10.1137/120903312.
- [55] J.-L. Guermond, Stabilization of Galerkin approximations of transport equations by subgrid modeling, *ESAIM: Mathematical Modelling and Numerical Analysis* 33 (6) (1999) 1293–1316. doi:10.1051/m2an:1999145.
- [56] J.-L. Guermond, A. Marra, L. Quartapelle, Subgrid stabilized projection method for 2D unsteady flows at high Reynolds numbers, *Computer Methods in Applied Mechanics and Engineering* 195 (44–47) (2006) 5857–5876. doi:10.1016/j.cma.2005.08.016.
- [57] M. G. Blyth, H. Luo, C. Pozrikidis, A comparison of interpolation grids over the triangle or the tetrahedron, *Journal of Engineering Mathematics* 56 (3) (2007) 263–272. doi:10.1007/s10665-006-9063-0.
- [58] H.-G. Roos, M. Stynes, L. Tobiska, *Robust Numerical Methods for Singularly Perturbed Differential Equations*, Vol. 24 of Springer Series in Computational Mathematics, Springer, 2008. doi:10.1007/978-3-540-34467-4.
- [59] K. W. Morton, A. Priestley, E. Suli, Stability of the Lagrange–Galerkin method with non-exact integration, *ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique* 22 (4) (1988) 625–653.  
URL <https://eudml.org/doc/193544>
- [60] R. Löhner, J. Ambrosiano, A vectorized particle tracer for unstructured grids, *Journal of Computational Physics* 91 (1) (1990) 22–31. doi:10.1016/0021-9991(90)90002-I.
- [61] A. Allievi, R. Bermejo, A Generalized Particle Search–Locate Algorithm for Arbitrary Grids, *Journal of Computational Physics* 132 (2) (1997) 157–166. doi:10.1006/jcph.1996.5604.
- [62] P. Farrell, J. Maddison, Conservative interpolation between volume meshes by local Galerkin projection, *Computer Methods in Applied Mechanics and Engineering* 200 (1–4) (2011) 89–100. doi:10.1016/j.cma.2010.07.015.
- [63] F. Alauzet, A parallel matrix-free conservative solution interpolation on unstructured tetrahedral meshes, *Computer Methods in Applied Mechanics and Engineering* 299 (2016) 116–142. doi:10.1016/j.cma.2015.10.012.
- [64] M. Tabata, S. Uchiumi, A genuinely stable Lagrange–Galerkin scheme for convection–diffusion problems, *Japan Journal of Industrial and Applied Mathematics* 33 (1) (2016) 121–143. doi:10.1007/s13160-015-0196-2.
- [65] E. Feireisl, A. Vasseur, New Perspectives in Fluid Dynamics: Mathematical Analysis of a Model Proposed by Howard Brenner, in: A. V. Fursikov, G. P. Galdi, V. V. Pukhnachev (Eds.), *New Directions in Mathematical Fluid Mechanics: The Alexander V. Kazhikhov Memorial Volume*, Advances in Mathematical Fluid Mechanics, Birkhäuser, Basel, 2010, pp. 153–179. doi:10.1007/978-3-0346-0152-8\_9.
- [66] T. J. Hughes, G. R. Feijóo, L. Mazzei, J.-B. Quincy, The variational multiscale method—a paradigm for computational mechanics, *Computer Methods in Applied Mechanics and Engineering* 166 (1–2) (1998) 3–24. doi:10.1016/S0045-7825(98)00079-6.
- [67] F. Schieweck, P. Skrzypacz, A Local Projection Stabilization Method with Shock Capturing and Diagonal Mass Matrix for Solving Non-stationary Transport Dominated Problems, *Computational Methods in Applied Mathematics* 12 (2) (2012) 221–240. doi:10.2478/cmam-2012-0019.

- [68] P. Fernandez, C. Nguyen, J. Peraire, A physics-based shock capturing method for unsteady laminar and turbulent flows, in: 2018 AIAA Aerospace Sciences Meeting, American Institute of Aeronautics and Astronautics, Kissimmee, Florida, 2018. doi:10.2514/6.2018-0062.
- [69] H. F. Walker, P. Ni, Anderson Acceleration for Fixed-Point Iterations, *SIAM Journal on Numerical Analysis* 49 (4) (2011) 1715–1735. doi:10.1137/10078356X.
- [70] H. F. Walker, Anderson Acceleration: Algorithms and Implementations, Tech. rep., Worcester Polytechnic University (2011).  
URL [https://users.wpi.edu/~walker/Papers/anderson\\_accn\\_algsimps.pdf](https://users.wpi.edu/~walker/Papers/anderson_accn_algsimps.pdf)
- [71] M. Shariff, A constrained conjugate gradient method and the solution of linear equations, *Computers & Mathematics with Applications* 30 (11) (1995) 25–37. doi:10.1016/0898-1221(95)00161-Q.
- [72] N. I. M. Gould, M. E. Hribar, J. Nocedal, On the Solution of Equality Constrained Quadratic Programming Problems Arising in Optimization, *SIAM Journal on Scientific Computing* 23 (4) (2001) 1376–1395. doi:10.1137/S1064827598345667.
- [73] H. Guillard, C. Farhat, On the significance of the geometric conservation law for flow computations on moving meshes, *Computer Methods in Applied Mechanics and Engineering* 190 (11) (2000) 1467–1482. doi:10.1016/S0045-7825(00)00173-0.
- [74] C. Farhat, P. Geuzaine, C. Grandmont, The Discrete Geometric Conservation Law and the Nonlinear Stability of ALE Schemes for the Solution of Flow Problems on Moving Grids, *Journal of Computational Physics* 174 (2) (2001) 669–694. doi:10.1006/jcph.2001.6932.
- [75] T. J. Hughes, G. Engel, L. Mazzei, M. G. Larson, The Continuous Galerkin Method Is Locally Conservative, *Journal of Computational Physics* 163 (2) (2000) 467–488. doi:10.1006/jcph.2000.6577.
- [76] E. Feireisl, M. Lukáčová-Medvidová, H. Mizerová, B. She, *Numerical Analysis of Compressible Fluid Flows*, 1st Edition, Springer Cham, 2021.  
URL <https://doi.org/10.1007/978-3-030-73788-7>
- [77] F. Hecht, BAMG: Bidimensional Anisotropic Mesh Generator (2006).  
URL [https://www.researchgate.net/profile/Frederic\\_Hecht/publication/228965458\\_bamg\\_Bidimensional\\_anisotropic\\_mesh\\_generator/links/54d0ac020cf298d65667cca9/bamg-Bidimensional-anisotropic-mesh-generator.pdf](https://www.researchgate.net/profile/Frederic_Hecht/publication/228965458_bamg_Bidimensional_anisotropic_mesh_generator/links/54d0ac020cf298d65667cca9/bamg-Bidimensional-anisotropic-mesh-generator.pdf)
- [78] C. A. Kennedy, M. H. Carpenter, Higher-order additive Runge–Kutta schemes for ordinary differential equations, *Applied Numerical Mathematics* 136 (2019) 183–205. doi:10.1016/j.apnum.2018.10.007.
- [79] J. Bezanson, A. Edelman, S. Karpinski, V. B. Shah, Julia: A Fresh Approach to Numerical Computing, *SIAM Review* 59 (1) (2017) 65–98. doi:10.1137/141000671.
- [80] D. A. Dunavant, High degree efficient symmetrical Gaussian quadrature rules for the triangle, *International Journal for Numerical Methods in Engineering* 21 (6) (1985) 1129–1148. doi:10.1002/nme.1620210612.
- [81] L. Zhang, T. Cui, H. Liu, A Set of Symmetric Quadrature Rules on Triangles and Tetrahedra, *Journal of Computational Mathematics* 1 (2009) 89–96.  
URL [https://global-sci.org/intro/article\\_detail/jcm/8561.html](https://global-sci.org/intro/article_detail/jcm/8561.html)
- [82] S. Wandzura, H. Xiao, Symmetric quadrature rules on a triangle, *Computers & Mathematics with Applications* 45 (12) (2003) 1829–1840. doi:10.1016/S0898-1221(03)90004-6.
- [83] I. Danaila, P. Joly, S. M. Kaber, M. Postel, Gas Dynamics: The Riemann Problem and Discontinuous Solutions: Application to the Shock Tube Problem, in: *An Introduction to Scientific Computing: Twelve Computational Projects Solved with MATLAB*, Springer, New York, NY, 2007, pp. 213–233. doi:10.1007/978-0-387-49159-2\_10.
- [84] M. A. Díaz, *Approximate Riemann Solvers* (2018).  
URL <https://github.com/wme7/ApproximateRiemannSolvers>
- [85] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, II, *Journal of Computational Physics* 83 (1) (1989) 32–78. doi:10.1016/0021-9991(89)90222-2.
- [86] W. Noh, Errors for calculations of strong shocks using an artificial viscosity and an artificial heat flux, *Journal of Computational Physics* 72 (1) (1987) 78–120. doi:10.1016/0021-9991(87)90074-X.
- [87] R. Ramani, J. Reisner, S. Shkoller, A space-time smooth artificial viscosity method with wavelet noise indicator and shock collision scheme, Part 2: The 2-D case, *Journal of Computational Physics* 387 (2019) 45–80. doi:10.1016/j.jcp.2019.02.048.
- [88] R. Liska, B. Wendroff, Comparison of Several Difference Schemes on 1D and 2D Test Problems for the Euler Equations, *SIAM Journal on Scientific Computing* 25 (3) (2003) 995–1017. doi:10.1137/S1064827502402120.
- [89] J. R. Kamm, Evaluation of the Sedov-von Neumann-Taylor Blast Wave Solution, Tech. rep., Los Alamos National Laboratory (2000).  
URL <https://www.bibsonomy.org/bibtex/2e8af6cf40ac2dbe50af880debe489da>
- [90] E. J. Tasker, R. Brunino, N. L. Mitchell, D. Michielsen, S. Hopton, F. R. Pearce, G. L. Bryan, T. Theuns, A test suite for quantitative comparison of hydrodynamic codes in astrophysics, *Monthly Notices of the Royal Astronomical Society* 390 (3) (2008) 1267–1281. doi:10.1111/j.1365-2966.2008.13836.x.
- [91] R. Samulyak, X. Wang, H.-C. Chen, Lagrangian particle method for compressible fluid dynamics, *Journal of Computational Physics* 362 (2018) 1–19. doi:10.1016/j.jcp.2018.02.004.
- [92] J. Shi, Y.-T. Zhang, C.-W. Shu, Resolution of high order WENO schemes for complicated flow structures, *Journal of Computational Physics* 186 (2) (2003) 690–696. doi:10.1016/S0021-9991(03)00094-9.
- [93] N. Fleischmann, S. Adami, N. A. Adams, Numerical symmetry-preserving techniques for low-dissipation shock-capturing schemes, *Computers & Fluids* 189 (2019) 94–107. doi:10.1016/j.compfluid.2019.04.004.

- 
- [94] A. Rault, G. Chiavassa, R. Donat, Shock-Vortex Interactions at High Mach Numbers, *Journal of Scientific Computing* 19 (1) (2003) 347–371. doi:10.1023/A:1025316311633.
  - [95] M. Dumbser, O. Zanotti, R. Loubère, S. Diot, A posteriori subcell limiting of the discontinuous Galerkin finite element method for hyperbolic conservation laws, *Journal of Computational Physics* 278 (2014) 47–75. doi:10.1016/j.jcp.2014.08.009.
  - [96] N. Tonicello, G. Lodato, L. Vervisch, Entropy preserving low dissipative shock capturing with wave-characteristic based sensor for high-order methods, *Computers & Fluids* 197 (2020) 104357. doi:10.1016/j.compfluid.2019.104357.
  - [97] A. Rahunathan, D. Stanescu, High-order W-methods, *Journal of Computational and Applied Mathematics* 233 (8) (2010) 1798–1811. doi:10.1016/j.cam.2009.09.017.
  - [98] J. Rang, Improved traditional Rosenbrock–Wanner methods for stiff ODEs and DAEs, *Journal of Computational and Applied Mathematics* 286 (2015) 128–144. doi:10.1016/j.cam.2015.03.010.
  - [99] D. S. Blom, P. Birken, H. Bijl, F. Kessels, A. Meister, A. H. van Zuijlen, A comparison of Rosenbrock and ESDIRK methods combined with iterative solvers for unsteady compressible flows, *Advances in Computational Mathematics* 42 (6) (2016) 1401–1426. doi:10.1007/s10444-016-9468-x.
  - [100] P. Tranquilli, A. Sandu, H. Zhang, LIRK-W: Linearly-implicit Runge-Kutta methods with approximate matrix factorization, arXiv:1611.07013 [math] (Nov. 2016). arXiv:1611.07013.  
URL <http://arxiv.org/abs/1611.07013>
  - [101] J. Carpio, J. L. Prieto, R. Bermejo, Anisotropic “Goal-Oriented” Mesh Adaptivity for Elliptic Problems, *SIAM Journal on Scientific Computing* 35 (2) (2013) A861–A885. doi:10.1137/120874606.
  - [102] J. Carpio, J. Prieto, An anisotropic, fully adaptive algorithm for the solution of convection-dominated equations with semi-Lagrangian schemes, *Computer Methods in Applied Mechanics and Engineering* 273 (2014) 77–99. doi:10.1016/j.cma.2014.01.025.
  - [103] V. Dolejší, Anisotropic hp-adaptive method based on interpolation error estimates in the  $L_q$ -norm, *Applied Numerical Mathematics* 82 (2014) 80–114. doi:10.1016/j.apnum.2014.03.003.
  - [104] V. Dolejší, G. May, F. Roskovec, P. Solin, Anisotropic hp-mesh optimization technique based on the continuous mesh and error models, *Computers & Mathematics with Applications* 74 (1) (2017) 45–63. doi:10.1016/j.camwa.2016.12.015.
  - [105] V. Dolejší, G. May, A. Rangarajan, A continuous hp-mesh model for adaptive discontinuous Galerkin schemes, *Applied Numerical Mathematics* 124 (2018) 1–21. doi:10.1016/j.apnum.2017.09.015.