

Entropy–stable discontinuous Galerkin approximation with summation–by–parts property for the incompressible Navier–Stokes/Cahn–Hilliard system

Juan Manzanero · Gonzalo Rubio ·
David A. Kopriva · Esteban Ferrer · Eusebio Valero

Received: date / Accepted: date

Abstract We develop an entropy stable two–phase incompressible Navier–Stokes/Cahn–Hilliard discontinuous Galerkin (DG) flow solver method. The model poses the Cahn–Hilliard equation as the phase field method, a skew–symmetric form of the momentum equation, and an artificial compressibility method to compute the pressure. We design the model so that it satisfies an entropy law, including free– and no–slip wall boundary conditions with non–zero wall contact angle. We then construct a high–order DG approximation of the model that satisfies the SBP–SAT property. With the help of a discrete stability analysis, the scheme has two modes: an entropy conserving approximation with central advective fluxes and the Bassi–Rebay 1 (BR1) method for diffusion, and an entropy stable approximation with an exact Riemann solver for advection and interface stabilization added to the BR1 method. The scheme is applicable to, and the stability proofs hold for, three–dimensional unstructured meshes with curvilinear hexahedral elements. We test the convergence of the schemes on a manufactured solution, and their robustness by solving a flow initialized from random numbers. In the latter, we find that a similar scheme that does not satisfy an entropy inequality had 30% probability to fail, while the entropy stable scheme never does. We also solve the static and rising bubble test problems, and to challenge the solver capabilities we compute a three–dimensional pipe flow in the annular regime.

Keywords Navier–Stokes · Cahn–Hilliard · Computational fluid dynamics · High-Order methods · Discontinuous Galerkin · SBP–SAT.

Juan Manzanero (E-mail: juan.manzanero@upm.es), Gonzalo Rubio, Esteban Ferrer, Eusebio Valero
ETSIAE-UPM - School of Aeronautics, Universidad Politécnica de Madrid. Plaza Cardenal Cisneros 3, E-28040
Madrid, Spain. // Center for Computational Simulation, Universidad Politécnica de Madrid, Campus de Monte-
gancedo, Boadilla del Monte, 28660, Madrid, Spain.

David A. Kopriva

Department of Mathematics, Florida State University and Computational Science Research Center, San Diego State
University.

1 Introduction

The study of multiphase flows is of broad interest, from both scientific and industrial perspectives. In particular, the oil industry investigates two-phase flows of immiscible fluids (e.g. oil and water). Without mixing, the typical flow configuration is the segregation of the two fluids, separated by a thin interfacial region that behaves like a permeable membrane. At the interface of two dissimilar fluids, the forces acting on the molecules of each of these fluids are not the same as within each phase and generate interfacial tension.

Among the various techniques to approach the solution of immiscible two-phase flows, one finds two broad categories: sharp and diffuse interface models. In the former, the interface is considered infinitely thin, and acts as a physical boundary condition where the two fluids are coupled and balanced by interfacial forces [1]. An example of a sharp interface model is the level-set method [2], where the interface is tracked by an additional variable advected by the flow velocity field.

Diffuse interface approaches, however, regularize the problem with the introduction of an interface with non-zero thickness and a smooth transition of thermodynamic variables between both fluids. This time, interfacial forces (which in the sharp interface approach are delta functions applied at the interface) are transformed to body forces whose effect is concentrated at the interface [3].

In this work we study the diffuse interface model of Cahn–Hilliard [4, 5] combined with the incompressible Navier–Stokes equations with variable density and artificial (also called *pseudo*) compressibility [6]. A review of alternate Navier–Stokes/Cahn–Hilliard models can be found in [7].

We discretize the system of partial differential equations that represents the model with a high-order Discontinuous Galerkin Spectral Element Method (DGSEM). The DGSEM allows for arbitrary order of accuracy and can represent complex geometries through the use of unstructured meshes with curvilinear elements. Here, we develop an entropy–stable discretization, which enhances its robustness and enables the industrialization of the method.

This work is the third on a roadmap to obtain an entropy stable multiphase solver. In [8], we developed a free–energy stable discretization of the Cahn–Hilliard equation, while in [9] we derived an entropy–stable approximation of the incompressible Navier–Stokes equations with artificial compressibility. We combine the developments in those previous papers to construct an entropy–stable DG approximation of the incompressible Navier–Stokes/Cahn–Hilliard system.

As in [8, 9], the method is a nodal DG approximation with Gauss–Lobatto points, which satisfies the summation–by–parts simultaneous–approximation–term (SBP–SAT) property [10, 11]. The SBP–SAT property satisfied by the Gauss–Lobatto variant of the DGSEM allows us to mimic the continuous entropy analysis semi–discretely (i.e. discrete in space, continuous in time). The SBP–SAT property originated with finite difference methods [10, 11, 12], and has been exploited in recent years to obtain entropy–stable DG schemes for the linear advection equation [13, 14], Burgers equation [15, 16], shallow water equations [17], Euler and Navier–Stokes equations [18, 16, 9], the magneto–hydrodynamics equations [19], and the Cahn–Hilliard equation [8] among others. Although we develop a DG approximation, the approach taken here can be applied to any discretization that satisfies the SBP–SAT property (e.g. finite difference).

An entropy–stable approximation requires that the continuous system of partial differential equations satisfies an entropy law. When the density is allowed to vary between the two fluids in a multiphase flow,

as opposed to the more traditional Model H [20] with constant density, the design of an entropy-stable scheme requires care. The variable density compromises the formulation used for the momentum equation [21]. Two main approaches to the design of a variable density entropy-stable approximation are commonly adopted: to use a skew-symmetric version of the momentum equation [22], or to augment the momentum equation with an additional diffusive flux that depends on the density difference [23, 24].

The addition of the artificial compressibility model, where a divergence-free velocity is not strictly enforced, completes the formulation used for the body interfacial forces. Without proper choices, terms proportional to $\vec{\nabla} \cdot \vec{u}$, which can not be neglected nor bounded, might appear in the entropy equation. With the proper choices, we confirm that the entropy of the Navier-Stokes/Cahn-Hilliard system obeys the Onsager principle; the total entropy is the sum of the entropy of the incompressible Navier-Stokes equations, plus the free-energy of the Cahn-Hilliard equation [4]. In [9] we derived a mathematical entropy of the incompressible Navier-Stokes equations with artificial compressibility as the sum of the traditional kinetic energy, plus an additional energy term that accounts for compressibility effects. As a result, the total entropy becomes the sum of the kinetic energy, the artificial compressibility energy, and the free-energy.

In this paper we construct a multiphase approximation that is appropriate to solve typical applications in the oil industry. An example is the transport of crude oil in pipes, where the flow of oil and water is solved under high Reynolds number conditions. Since one desires to keep the mesh and degrees of freedom as low as possible, the flow configuration is often under-resolved. Under these circumstances, a robust method that is provably stable is attractive, as it avoids aliasing driven numerical instabilities that might lead to numerical divergence [14].

The rest of this work is organized as follows: in Sec. 2 we describe the incompressible Navier-Stokes/Cahn-Hilliard model, with the continuous entropy analysis in Sec. 2.1. The construction of the discrete DG approximation is described in Sec.3. Then, the semi-discrete stability analysis of the approximation is performed in Sec. 4, for which a summary can be found in Sec. 4.5. Lastly, we provide numerical experiments in Sec. 5 that assess both the accuracy and robustness of the method. We solve a manufactured solution convergence analysis in Sec. 5.1, and an assessment of the robustness solving a random initial condition in Sec. 5.2. Then, we solve classic static and rising bubble test problems in Sec. 5.3 and Sec. 5.4 respectively. Lastly, in Sec. 5.5 we challenge the technique by solving a multiphase pipe in the annular flow regime. Conclusions and discussions can be found in Sec. 6.

2 Governing equations. Continuous entropy analysis

In this section we describe a model that combines the incompressible Navier-Stokes and the Cahn-Hilliard equations, which we will refer to as iNS/CH. In the single phase variable density incompressible Navier-Stokes equations, the density $\rho(\vec{x}, t)$ is an independent variable that satisfies the continuity equation (see [22, 25, 9]),

$$\rho_t + \vec{\nabla} \cdot (\rho \vec{u}) = 0, \quad (1)$$

where $\vec{u}(\vec{x}, t) = (u, v, w)$ is the velocity field. In the two-phase iNS/CH system, however, the density is computed from the concentration $c(\vec{x}, t)$ of the two fluids [3],

$$\rho = \rho(c) = \rho_1 c + \rho_2 (1 - c), \quad (2)$$

where $\rho_{1,2}$ are the densities of fluids 1 and 2, respectively, which are constant in space and time.

In phase field methods, an advection–diffusion equation drives the concentration. Among the model choices, we pick the Cahn–Hilliard equation [4, 5],

$$c_t + \vec{\nabla} \cdot (c\vec{u}) = M_0 \vec{\nabla}^2 \mu. \quad (3)$$

In (3), μ is the chemical potential,

$$\mu = \frac{df_0(c)}{dc} - \frac{3}{2} \sigma \varepsilon \vec{\nabla}^2 c, \quad (4)$$

where $f_0(c)$ is the chemical free–energy,

$$f_0 = \frac{12\sigma}{\varepsilon} c^2(1-c)^2, \quad (5)$$

σ is the coefficient of interface tension between the fluids, ε is the interface width, and M_0 is the mobility, computed in this work with the chemical characteristic time, t_{CH} , as

$$M_0 = \frac{\varepsilon}{\sigma t_{\text{CH}}}. \quad (6)$$

The parameters σ , ε , M_0 , and t_{CH} are positive constants. The Cahn–Hilliard equation (3), with the chemical potential definition (4), has an associated free–energy

$$\mathcal{F}(c, \vec{\nabla} c) = f_0(c) + \frac{3}{4} \sigma \varepsilon |\vec{\nabla} c|^2. \quad (7)$$

The velocity field is computed from the momentum equation. If one considers that the continuity equation (1) holds, the conservative,

$$(\rho\vec{u})_t + \vec{\nabla} \cdot (\rho\vec{u}\vec{u}) = -\vec{\nabla} p_s - \vec{\nabla} \cdot \left(\vec{\nabla} c \otimes \frac{\partial \mathcal{F}}{\partial \vec{\nabla} c} \right) + \vec{\nabla} \cdot \left(\eta \left(\vec{\nabla} \vec{u} + \vec{\nabla} \vec{u}^T \right) \right) + \rho \vec{g}, \quad (8)$$

non–conservative,

$$\rho(\vec{u})_t + \rho\vec{u} \cdot \vec{\nabla} \vec{u} = -\vec{\nabla} p_s - \vec{\nabla} \cdot \left(\vec{\nabla} c \otimes \frac{\partial \mathcal{F}}{\partial \vec{\nabla} c} \right) + \vec{\nabla} \cdot \left(\eta \left(\vec{\nabla} \vec{u} + \vec{\nabla} \vec{u}^T \right) \right) + \rho \vec{g}, \quad (9)$$

and skew–symmetric

$$\frac{1}{2} (\rho\vec{u})_t + \frac{1}{2} \rho(\vec{u})_t + \vec{\nabla} \cdot \left(\frac{1}{2} \rho\vec{u}\vec{u} \right) + \frac{1}{2} \rho\vec{u} \cdot \vec{\nabla} \vec{u} = -\vec{\nabla} p_s - \vec{\nabla} \cdot \left(\vec{\nabla} c \otimes \frac{\partial \mathcal{F}}{\partial \vec{\nabla} c} \right) + \vec{\nabla} \cdot \left(\eta \left(\vec{\nabla} \vec{u} + \vec{\nabla} \vec{u}^T \right) \right) + \rho \vec{g}, \quad (10)$$

forms of the momentum equation are identical in the continuous setting. However, since the density does not satisfy the continuity equation (1), but rather the compatibility condition (2), the three forms (8), (9) and (10) are no longer equivalent. In this situation, the momentum equation is *frozen* as one of the three forms. Following [22, 21], we choose the skew–symmetric version of the momentum (10) since is the only one that satisfies an entropy inequality for any positive density field. (An alternative that produces an entropy–stable scheme for the iNS/CH system is to include a relative flux in the momentum equation that models the diffusion of the components [26].) We use the chain rule in time to perform an additional manipulation on the time derivative of (10) and obtain the momentum form by Guermond et al. [22],

$$\sqrt{\rho}(\sqrt{\rho}\vec{u})_t + \vec{\nabla} \cdot \left(\frac{1}{2} \rho\vec{u}\vec{u} \right) + \frac{1}{2} \rho\vec{u} \cdot \vec{\nabla} \vec{u} = -\vec{\nabla} p_s - \vec{\nabla} \cdot \left(\vec{\nabla} c \otimes \frac{\partial \mathcal{F}}{\partial \vec{\nabla} c} \right) + \vec{\nabla} \cdot \left(\eta \left(\vec{\nabla} \vec{u} + \vec{\nabla} \vec{u}^T \right) \right) + \rho \vec{g}. \quad (11)$$

In (11), $p_s(\vec{x}, t)$ is the static pressure. The second term is the body force approximation of the capillary pressure [3],

$$\vec{p}_c = -\vec{\nabla} \cdot \left(\vec{\nabla} c \otimes \frac{\partial \mathcal{F}}{\partial \vec{\nabla} c} \right) = -\frac{3}{2} \sigma \varepsilon \vec{\nabla} \cdot \left(\vec{\nabla} c \otimes \vec{\nabla} c \right). \quad (12)$$

Finally η is the viscosity, computed from the concentration c and the viscosities of fluids 1 and 2, $\eta_{1,2}$,

$$\eta = \eta(c) = \eta_1 c + \eta_2 (1 - c). \quad (13)$$

As in [3], we write the capillary pressure more conveniently. Applying the derivative product rule,

$$\vec{p}_c = -\frac{3}{2} \sigma \varepsilon \left(\vec{\nabla}^2 c \vec{\nabla} c + \vec{\nabla} \left(\frac{1}{2} |\vec{\nabla} c|^2 \right) \right), \quad (14)$$

and adding and subtracting the term $\vec{\nabla} f_0 = f'_0(c) \vec{\nabla} c$, we can rewrite the expression for the body force as

$$\vec{p}_c = \left(f'_0(c) - \frac{3}{2} \sigma \varepsilon \vec{\nabla}^2 c \right) \vec{\nabla} c - \vec{\nabla} \left(f_0(c) + \frac{3}{4} \sigma \varepsilon |\vec{\nabla} c|^2 \right) = \mu \vec{\nabla} c - \vec{\nabla} \mathcal{F} = -c \vec{\nabla} \mu - \vec{\nabla} (\mathcal{F} - \mu c). \quad (15)$$

In the last form, an additional application of the product derivative rule for the first term has been performed. Eq. (15) represents equivalent expressions of the capillary pressure (12), the last form being the one adopted here. The first term of the last form, $-c \vec{\nabla} \mu$, is a non-conservative term, and the second, $-\vec{\nabla} (\mathcal{F} - \mu c)$, is combined with the static pressure gradient to define an auxiliary pressure

$$p = p_s + \mathcal{F} - \mu c. \quad (16)$$

With all of these manipulations, we get the final expression of the momentum equation,

$$\sqrt{\rho} (\sqrt{\rho} \vec{u})_t + \vec{\nabla} \cdot \left(\frac{1}{2} \rho \vec{u} \vec{u} \right) + \frac{1}{2} \rho \vec{u} \cdot \vec{\nabla} \vec{u} + c \vec{\nabla} \mu = -\vec{\nabla} p + \vec{\nabla} \cdot \left(\eta \left(\vec{\nabla} \vec{u} + \vec{\nabla} \vec{u}^T \right) \right) + \rho \vec{g}. \quad (17)$$

Many authors have explored how to relax the incompressibility constraint $\vec{\nabla} \cdot \vec{u} = 0$ in Navier–Stokes and Navier–Stokes/Cahn–Hilliard systems by using the artificial (or *pseudo*) compressibility method [6, 27, 28, 29]. We use the time derivative of the pressure [27, 25, 9],

$$p_t + \rho_0 c_0^2 \vec{\nabla} \cdot \vec{u} = 0, \quad (18)$$

where c_0 is the artificial sound speed, and $\rho_0 = \max(\rho_1, \rho_2)$.

Remark 1 *Alternatively, one can take the time derivative of the pressure Laplacian [6, 21, 30],*

$$-\vec{\nabla}^2 p_t + \rho_0 c_0^2 \vec{\nabla} \cdot \vec{u} = 0. \quad (19)$$

Since both models share the same velocity divergence term, we can modify the proofs of stability for the approximation of (18) and study the approximation of the second model (19). We do so in Appendix A. The use of the Laplacian of the pressure leads to a more complicated implicit implementation, so for that scheme we include only the stability proof in this paper.

The combination of (3), (17) and (18) is the iNS/CH system studied in this paper. The coupling between the Cahn–Hilliard equation and the incompressible Navier–Stokes equations is *two-way*, through the compatibility conditions (2), (13), and the capillary pressure. We note that even in the continuous setting, the momentum is not conserved because of the choice of the skew–symmetric form (17). Although one can argue about the physical implications, we note that the capillary force is a non–conservative term as well, so momentum conservation is not guaranteed in any of the forms (8), (9), and (17). Alternative incompressible Navier–Stokes/Cahn–Hilliard discretizations using the skew–symmetric form of the momentum equation can be found in [31, 21].

To simplify the notation, we write the iNS/CH system as a general advection–diffusion equation,

$$\begin{aligned} & \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \sqrt{\rho} & 0 & 0 & 0 \\ 0 & 0 & \sqrt{\rho} & 0 & 0 \\ 0 & 0 & 0 & \sqrt{\rho} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} c \\ \sqrt{\rho}u \\ \sqrt{\rho}v \\ \sqrt{\rho}w \\ p \end{pmatrix}_t + \vec{\nabla} \cdot \begin{pmatrix} c\vec{u} \\ \frac{1}{2}\rho\vec{u}u + p\vec{e}_1 \\ \frac{1}{2}\rho\vec{u}v + p\vec{e}_2 \\ \frac{1}{2}\rho\vec{u}w + p\vec{e}_3 \\ 0 \end{pmatrix} \\ & + \begin{pmatrix} \vec{0} \\ \frac{1}{2}\rho\vec{u} \cdot \vec{\nabla}u + c\vec{e}_1 \cdot \vec{\nabla}\mu \\ \frac{1}{2}\rho\vec{u} \cdot \vec{\nabla}v + c\vec{e}_2 \cdot \vec{\nabla}\mu \\ \frac{1}{2}\rho\vec{u} \cdot \vec{\nabla}w + c\vec{e}_3 \cdot \vec{\nabla}\mu \\ \rho_0 c_0^2 (\vec{e}_1 \cdot \vec{\nabla}u + \vec{e}_2 \cdot \vec{\nabla}v + \vec{e}_3 \cdot \vec{\nabla}w) \end{pmatrix} = \vec{\nabla} \cdot \begin{pmatrix} M_0 \vec{\nabla}\mu \\ 2\eta\mathbf{S} \cdot \vec{e}_1 \\ 2\eta\mathbf{S} \cdot \vec{e}_2 \\ 2\eta\mathbf{S} \cdot \vec{e}_3 \\ \vec{0} \end{pmatrix} + \begin{pmatrix} 0 \\ \rho\vec{g} \cdot \vec{e}_1 \\ \rho\vec{g} \cdot \vec{e}_2 \\ \rho\vec{g} \cdot \vec{e}_3 \\ 0 \end{pmatrix}, \end{aligned} \quad (20)$$

where,

$$\mathbf{S} = \text{sym}(\vec{\nabla}\vec{u}) = \frac{1}{2}(\vec{\nabla}\vec{u} + \vec{\nabla}\vec{u}^T), \quad (21)$$

is the strain tensor, and \vec{e}_i are the space unit vectors. Note that we have grouped the velocity divergence in the artificial compressibility equation (18) into the non–conservative terms. Although treating the velocity divergence as conservative or non–conservative is equivalent, the latter makes it easier to show stability.

We adopt the notation in [16] to work with vectors of different nature. We define space vectors (e.g. $\vec{x} = (x, y, z) \in \mathbb{R}^3$) with an arrow on top, and state vectors (e.g. $\mathbf{q} = (c, \sqrt{\rho}\vec{u}, p) \in \mathbb{R}^5$) in bold. Moreover, we define block vectors as the result of stacking three state vectors (e.g. fluxes),

$$\vec{\mathbf{f}}_e = \begin{pmatrix} \mathbf{f}_{e,1} \\ \mathbf{f}_{e,2} \\ \mathbf{f}_{e,3} \end{pmatrix} = \begin{pmatrix} \mathbf{f}_e \\ \mathbf{g}_e \\ \mathbf{h}_e \end{pmatrix}, \quad \vec{\mathbf{f}}_v = \begin{pmatrix} \mathbf{f}_{v,1} \\ \mathbf{f}_{v,2} \\ \mathbf{f}_{v,3} \end{pmatrix} = \begin{pmatrix} \mathbf{f}_v \\ \mathbf{g}_v \\ \mathbf{h}_v \end{pmatrix}, \quad (22)$$

and define the operator \mathcal{Y} that transforms a 5×3 (state–space) matrix into a 15×1 block vector,

$$\vec{\mathbf{f}} = \mathcal{Y} \begin{pmatrix} f_1 & g_1 & h_1 \\ f_2 & g_2 & h_2 \\ f_3 & g_3 & h_3 \\ f_4 & g_4 & h_4 \\ f_5 & g_5 & h_5 \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{g} \\ \mathbf{h} \end{pmatrix}. \quad (23)$$

This notation allows us to define products of state, space, and block vectors,

$$\vec{\mathbf{f}} \cdot \vec{\mathbf{g}} = \sum_{i=1}^3 \mathbf{f}_i^T \mathbf{g}_i, \quad \vec{g} \cdot \vec{\mathbf{f}} = \sum_{i=1}^3 g_i \mathbf{f}_i, \quad \vec{g} \mathbf{f} = \begin{pmatrix} g_1 \mathbf{f} \\ g_2 \mathbf{f} \\ g_3 \mathbf{f} \end{pmatrix}. \quad (24)$$

We can then define the divergence and gradient operators using (24),

$$\vec{\nabla} \cdot \vec{\mathbf{f}} = \sum_{i=1}^3 \frac{\partial \mathbf{f}_i}{\partial x_i}, \quad \vec{\nabla} \mathbf{q} = \begin{pmatrix} \mathbf{q}_x \\ \mathbf{q}_y \\ \mathbf{q}_z \end{pmatrix}. \quad (25)$$

Lastly, we refer to state matrices (i.e. 5×5 matrices) with an underline, e.g. $\underline{\mathbf{B}}$, which can be combined to construct a block matrix,

$$\mathcal{B} = \begin{pmatrix} \underline{\mathbf{B}}_{11} & \underline{\mathbf{B}}_{12} & \underline{\mathbf{B}}_{13} \\ \underline{\mathbf{B}}_{21} & \underline{\mathbf{B}}_{22} & \underline{\mathbf{B}}_{23} \\ \underline{\mathbf{B}}_{31} & \underline{\mathbf{B}}_{32} & \underline{\mathbf{B}}_{33} \end{pmatrix}. \quad (26)$$

Block matrices can be directly multiplied to a block vector to obtain another block vector. For instance, to perform a matrix multiplication in space (e.g. a rotation),

$$\vec{g} = \mathbf{M} \vec{f}, \quad (27)$$

for each of the variables in the state vector, we construct the block matrix version of \mathbf{M} ($\underline{\mathbf{I}}_5$ is the 5×5 identity matrix),

$$\mathcal{M} = \begin{pmatrix} M_{11} \underline{\mathbf{I}}_5 & M_{12} \underline{\mathbf{I}}_5 & M_{13} \underline{\mathbf{I}}_5 \\ M_{21} \underline{\mathbf{I}}_5 & M_{22} \underline{\mathbf{I}}_5 & M_{23} \underline{\mathbf{I}}_5 \\ M_{31} \underline{\mathbf{I}}_5 & M_{32} \underline{\mathbf{I}}_5 & M_{33} \underline{\mathbf{I}}_5 \end{pmatrix}, \quad (28)$$

so that we can compactly write

$$\vec{\mathbf{g}} = \mathcal{M} \vec{\mathbf{f}}. \quad (29)$$

For more details, see [16].

The notation introduced above makes it possible to write the iNS/CH system (20) compactly in the form of a general advection–diffusion equation,

$$\underline{\mathbf{m}} \mathbf{q}_t + \vec{\nabla} \cdot \vec{\mathbf{f}}_e(\mathbf{q}) + \sum_{m=1}^5 \vec{\phi}_m(\mathbf{q}) \cdot \vec{\nabla} w_m = \vec{\nabla} \cdot \vec{\mathbf{f}}_v(\mathbf{q}, \vec{\nabla} \mathbf{w}) + \mathbf{s}(\mathbf{q}), \quad (30)$$

with state vector $\mathbf{q} = (c, \sqrt{\rho} \vec{u}, p)$, gradient variables vector $\mathbf{w} = (w_1, w_2, \dots, w_5) = (\mu, \vec{u}, p)$, mass matrix $\underline{\mathbf{m}}$,

$$\underline{\mathbf{m}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \sqrt{\rho} \mathbf{I}_3 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad (31)$$

inviscid fluxes $\vec{\mathbf{f}}_e(\mathbf{q})$,

$$\mathbf{f}_{e,1} = \mathbf{f}_e = \begin{pmatrix} cu \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ 0 \end{pmatrix}, \quad \mathbf{f}_{e,2} = \mathbf{g}_e = \begin{pmatrix} cv \\ \rho v \\ \rho v^2 + p \\ \rho vw \\ 0 \end{pmatrix}, \quad \mathbf{f}_{e,3} = \mathbf{h}_e = \begin{pmatrix} cw \\ \rho w \\ \rho vw \\ \rho w^2 + p \\ 0 \end{pmatrix}, \quad (32)$$

non-conservative term coefficients $\vec{\phi}_m(\mathbf{q})$,

$$\begin{aligned} \vec{\phi}_1 &= \Upsilon \begin{pmatrix} 0 \\ c\vec{e}_1 \\ c\vec{e}_2 \\ c\vec{e}_3 \\ 0 \end{pmatrix}, \quad \vec{\phi}_2 = \Upsilon \begin{pmatrix} 0 \\ \frac{1}{2}\rho\vec{u} \\ 0 \\ 0 \\ \rho_0 c_0^2 \vec{e}_1 \end{pmatrix}, \quad \vec{\phi}_3 = \Upsilon \begin{pmatrix} 0 \\ 0 \\ \frac{1}{2}\rho\vec{u} \\ 0 \\ \rho_0 c_0^2 \vec{e}_2 \end{pmatrix}, \\ \vec{\phi}_4 &= \Upsilon \begin{pmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{2}\rho\vec{u} \\ \rho_0 c_0^2 \vec{e}_3 \end{pmatrix}, \quad \vec{\phi}_5 = \Upsilon \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \end{aligned} \quad (33)$$

viscous fluxes $\vec{\mathbf{f}}_v(\vec{\nabla}\mathbf{w})$,

$$\mathbf{f}_{v,1} = \mathbf{f}_v = \begin{pmatrix} M_0\mu_x \\ 2\eta\mathcal{S}_{11} \\ 2\eta\mathcal{S}_{21} \\ 2\eta\mathcal{S}_{31} \\ 0 \end{pmatrix}, \quad \mathbf{f}_{v,2} = \mathbf{g}_v = \begin{pmatrix} M_0\mu_y \\ 2\eta\mathcal{S}_{12} \\ 2\eta\mathcal{S}_{22} \\ 2\eta\mathcal{S}_{32} \\ 0 \end{pmatrix}, \quad \mathbf{f}_{v,3} = \mathbf{h}_v = \begin{pmatrix} M_0\mu_z \\ 2\eta\mathcal{S}_{13} \\ 2\eta\mathcal{S}_{23} \\ 2\eta\mathcal{S}_{33} \\ 0 \end{pmatrix}, \quad (34)$$

and source term $\mathbf{s}(\mathbf{q}) = (0, \rho\vec{g}, 0)$.

An attractive property of the iNS/CH system from the point of view of computational efficiency is that the same gradient variables \mathbf{w} are used in both the non-conservative terms and the viscous fluxes. Moreover, these gradient variables will be shown to be the entropy variables associated with the mathematical entropy in Section 2.1. Additionally, recall that the velocity divergence from the artificial compressibility equation (18) was grouped into the non-conservative terms since it is beneficial when proving stability.

2.1 Entropy analysis of the iNS/CH system

The entropy analysis rests on the existence of a pair $\mathcal{E}(\mathbf{q})$ (mathematical entropy) and \mathbf{w} (entropy variables) that contract the system of equations (30) into a conservation law [32],

$$\begin{aligned} \mathbf{w}^T \left(\underline{\mathbf{m}}\mathbf{q}_t + \vec{\nabla} \cdot \vec{\mathbf{f}}_e(\mathbf{q}) + \sum_{m=1}^5 \vec{\phi}_m(\mathbf{q}) \cdot \vec{\nabla} w_m - \vec{\nabla} \cdot \vec{\mathbf{f}}_v(\mathbf{q}, \vec{\nabla}\mathbf{w}) \right) \\ = \mathcal{E}_t + \nabla \cdot \vec{f}^{\mathcal{E}} + \nabla\mathbf{w}^T \cdot \vec{\mathbf{f}}_v = 0, \end{aligned} \quad (35)$$

with an entropy flux $\vec{f}^{\mathcal{E}}$, and a viscous dissipative contribution,

$$\vec{\nabla}\mathbf{w}^T \cdot \vec{\mathbf{f}}_v \geq 0. \quad (36)$$

The difference between the usual entropy analysis [32, 11, 11] and this work is that we will incorporate the Cahn–Hilliard free-energy into the mathematical entropy, which makes it depend not only on the

solution, but also its gradient. To obtain the entropy of the iNS/CH system, we add the Cahn–Hilliard free–energy \mathcal{F} to the incompressible NSE entropy defined in [9]. The latter combines the traditional kinetic energy \mathcal{K} with an extra energy term due to the artificial compressibility \mathcal{E}_{AC} . For the iNS/CH system, the total entropy is,

$$\begin{aligned}\mathcal{E} &= \mathcal{F}(c, \vec{\nabla}c) + \mathcal{K}(\sqrt{\rho}\vec{u}) + \mathcal{E}_{AC}(p) \\ &= f_0(c) + \frac{3}{4}\sigma\varepsilon|\vec{\nabla}c|^2 + \frac{1}{2}\rho v_{tot}^2 + \frac{p^2}{2\rho_0 c_0^2} \geq 0,\end{aligned}\quad (37)$$

where $v_{tot}^2 = u^2 + v^2 + w^2$ is the square of the total speed. Eq. (37) assumes positivity on the density ρ , which is ensured by limiting the maximum and minimum density values used in the momentum equation with a simple cutoff,

$$\rho(\hat{c}) = \rho_1 \hat{c} + \rho_2 (1 - \hat{c}), \quad \hat{c} = \min(\max(c, 0), 1). \quad (38)$$

We find a set of entropy variables \mathbf{w} that contracts the time derivative of the state vector, \mathbf{q}_t , into the time derivative of the entropy \mathcal{E}_t , plus an additional divergence term of a time derivative flux $\vec{f}_t^\mathcal{E}$,

$$\mathbf{w}^T \underline{\mathbf{m}} \mathbf{q}_t = \mathcal{E}_t + \vec{\nabla} \cdot \vec{f}_t^\mathcal{E}. \quad (39)$$

To do so, we compute the time derivative of the entropy (37),

$$\begin{aligned}\mathcal{E}_t &= f'_0(c)c_t + \frac{3}{2}\sigma\varepsilon\nabla c \cdot \nabla c_t + \sqrt{\rho}\vec{u} \cdot (\sqrt{\rho}\vec{u})_t + \frac{pp_t}{\rho_0 c_0^2} \\ &= \left(f'_0(c) - \frac{3}{2}\sigma\varepsilon\vec{\nabla}^2 c\right)c_t + \vec{\nabla} \cdot \left(\frac{3}{2}\sigma\varepsilon c_t \vec{\nabla} c\right) + \sqrt{\rho}\vec{u} \cdot (\sqrt{\rho}\vec{u})_t + \frac{pp_t}{\rho_0 c_0^2} \\ &= \mu c_t + \vec{\nabla} \cdot \left(\frac{3}{2}\sigma\varepsilon c_t \vec{\nabla} c\right) + \sqrt{\rho}u(\sqrt{\rho}u)_t + \sqrt{\rho}v(\sqrt{\rho}v)_t + \sqrt{\rho}w(\sqrt{\rho}w)_t + \frac{pp_t}{\rho_0 c_0^2},\end{aligned}\quad (40)$$

and the product of the entropy variables with the time derivative of the state vector,

$$w_1 c_t + \sqrt{\rho}w_2(\sqrt{\rho}u)_t + \sqrt{\rho}w_3(\sqrt{\rho}v)_t + \sqrt{\rho}w_4(\sqrt{\rho}w)_t + w_5 p_t. \quad (41)$$

Thus, we replace (40) and (41) in (39) and rearrange,

$$\begin{aligned}(w_1 - \mu)c_t + \sqrt{\rho}(w_2 - u)(\sqrt{\rho}u)_t + \sqrt{\rho}(w_3 - v)(\sqrt{\rho}v)_t + \sqrt{\rho}(w_4 - w)(\sqrt{\rho}w)_t \\ + \left(w_5 - \frac{p}{\rho_0 c_0^2}\right)p_t = \vec{\nabla} \cdot \left(\vec{f}_t^\mathcal{E} + \frac{3}{2}\sigma\varepsilon c_t \vec{\nabla} c\right),\end{aligned}\quad (42)$$

to extract the entropy variables \mathbf{w} and time entropy flux $\vec{f}_t^\mathcal{E}$

$$\mathbf{w} = \left(\mu, u, v, w, \frac{p}{\rho_0 c_0^2}\right), \quad \vec{f}_t^\mathcal{E} = -\frac{3}{2}\sigma\varepsilon c_t \vec{\nabla} c. \quad (43)$$

Note that the minimum of the entropy is found when all the entropy variables are zero.

The contraction of the inviscid fluxes rests on one property to be satisfied by the inviscid fluxes and the non-conservative terms.

Property 1 *If inviscid fluxes, $\vec{\mathbf{f}}_e$, and non-conservative terms coefficients, $\vec{\phi}_m$, satisfy*

$$\mathbf{e}_m^T \vec{\mathbf{f}}_e = \mathbf{w}^T \vec{\phi}_m, \quad (44)$$

where \mathbf{e}_m is the state unit vector along the m -th state variable, then the entropy variables automatically contract the inviscid fluxes, and the entropy flux is computed as

$$\vec{f}_e^{\mathcal{E}} = \mathbf{w}^T \vec{\mathbf{f}}_e. \quad (45)$$

To show the contraction property (44), we multiply the divergence of the inviscid fluxes by the entropy variables, apply the derivative product rule, and write the second scalar product as the sum of the product of the five state components. Lastly we use (44) and (45) to see that

$$\begin{aligned} \mathbf{w}^T \vec{\nabla} \cdot \vec{\mathbf{f}}_e &= \vec{\nabla} \cdot (\mathbf{w}^T \vec{\mathbf{f}}_e) - \nabla \mathbf{w}^T \cdot \vec{\mathbf{f}}_e = \vec{\nabla} \cdot (\mathbf{w}^T \vec{\mathbf{f}}_e) - \sum_{m=1}^5 \mathbf{e}_m^T \vec{\mathbf{f}}_e \cdot \vec{\nabla} w_m \\ &= \vec{\nabla} \cdot \vec{f}_e^{\mathcal{E}} - \sum_{m=1}^5 \mathbf{w}^T \vec{\phi}_m \cdot \vec{\nabla} w_m. \end{aligned} \quad (46)$$

Therefore,

$$\mathbf{w}^T \left(\vec{\nabla} \cdot \vec{\mathbf{f}}_e(\mathbf{q}) + \sum_{m=1}^5 \vec{\phi}_m(\mathbf{q}) \cdot \vec{\nabla} w_m \right) = \vec{\nabla} \cdot \vec{f}_e^{\mathcal{E}}. \quad (47)$$

The iNS/CH system does satisfy the property (44), for

$$\begin{aligned} \mathbf{e}_1^T \vec{\mathbf{f}}_e &= c\bar{u} = \mathbf{w}^T \vec{\phi}_1, \\ \mathbf{e}_2^T \vec{\mathbf{f}}_e &= \frac{1}{2} \rho u \bar{u} + p \bar{e}_1 = \mathbf{w}^T \vec{\phi}_2, \\ \mathbf{e}_3^T \vec{\mathbf{f}}_e &= \frac{1}{2} \rho v \bar{u} + p \bar{e}_2 = \mathbf{w}^T \vec{\phi}_3, \\ \mathbf{e}_4^T \vec{\mathbf{f}}_e &= \frac{1}{2} \rho w \bar{u} + p \bar{e}_3 = \mathbf{w}^T \vec{\phi}_4, \\ \mathbf{e}_5^T \vec{\mathbf{f}}_e &= 0 = \mathbf{w}^T \vec{\phi}_5. \end{aligned} \quad (48)$$

So, the inviscid entropy flux for the iNS/CH system follows from (45) and is

$$\vec{f}_e^{\mathcal{E}} = \left(\frac{1}{2} \rho v_{tot}^2 + p + \mu c \right) \bar{u}. \quad (49)$$

For completeness, insight into how each of the different terms are combined into an entropy flux is provided in Appendix B.

Lastly, the entropy variables contract viscous fluxes. The entropy flux follows (45),

$$\vec{f}_v^{\mathcal{E}} = \mathbf{w}^T \vec{\mathbf{f}}_v = M_0 \mu \vec{\nabla} \mu + 2\eta S \cdot \bar{u}. \quad (50)$$

Also, (36) holds,

$$\begin{aligned}\vec{\nabla} \mathbf{w}^T \cdot \vec{\mathbf{f}}_v &= M_0 |\vec{\nabla} \mu|^2 + \eta \left(\vec{\nabla} \vec{u} + \vec{\nabla} \vec{u}^T \right) : \vec{\nabla} \vec{u} \\ &= M_0 |\vec{\nabla} \mu|^2 + 2\eta \frac{1}{2} \left(\vec{\nabla} \vec{u} + \vec{\nabla} \vec{u}^T \right) : \frac{1}{2} \left(\vec{\nabla} \vec{u} + \vec{\nabla} \vec{u}^T \right) \\ &= M_0 |\vec{\nabla} \mu|^2 + 2\eta \mathbf{S} : \mathbf{S} \geq 0.\end{aligned}\tag{51}$$

Therefore,

$$\mathbf{w}^T \vec{\nabla} \cdot \vec{\mathbf{f}}_v = \vec{\nabla} \cdot \left(\mathbf{w}^T \vec{\mathbf{f}}_v \right) - \vec{\nabla} \mathbf{w}^T \cdot \vec{\mathbf{f}}_v \leq \vec{\nabla} \cdot \vec{f}_v^{\mathcal{E}}.\tag{52}$$

In (51), we replaced $\vec{\nabla} \vec{u}$ by its symmetric part (21) since it multiplies a symmetric tensor.

With these results, we confirm that the iNS/CH system, (20), with entropy, (37), satisfies the conservation law

$$\mathcal{E}_t + \vec{\nabla} \cdot \vec{f}^{\mathcal{E}} = -\vec{\nabla} \mathbf{w}^T \cdot \vec{\mathbf{f}}_v = -M_0 |\vec{\nabla} \mu|^2 - 2\eta \mathbf{S} : \mathbf{S} \leq 0,\tag{53}$$

with entropy flux $\vec{f}^{\mathcal{E}}$,

$$\vec{f}^{\mathcal{E}} = \vec{f}_t^{\mathcal{E}} + \vec{f}_e^{\mathcal{E}} + \vec{f}_v^{\mathcal{E}} = -\frac{3}{2} \sigma \varepsilon c_t \vec{\nabla} c + \left(\frac{1}{2} \rho v_{tot}^2 + p + \mu c \right) \vec{u} + 2\eta \mathbf{S} \cdot \vec{u} + M_0 \mu \vec{\nabla} \mu.\tag{54}$$

Eq. (53) shows that the entropy (37) is always dissipated in the interior of the domain, and it can only increase due to boundary exchanges. The entropy equation can be written as a global equation by integrating over the whole domain Ω ,

$$\frac{d\bar{\mathcal{E}}}{dt} + \int_{\partial\Omega} \vec{f}^{\mathcal{E}} \cdot \vec{n} dS = - \int_{\Omega} \left(M_0 |\vec{\nabla} \mu|^2 + 2\eta \mathbf{S} : \mathbf{S} \right) d\vec{x} \leq 0\tag{55}$$

where $\bar{\mathcal{E}}$ is the total entropy,

$$\bar{\mathcal{E}} = \int_{\Omega} \mathcal{E} d\vec{x}.\tag{56}$$

The discrete approximation will be constructed so that it mimics (55), which will guarantee that the discrete entropy of the approximation will remain bounded by the boundary and initial data.

2.1.1 Boundary conditions

Boundedness of the total entropy depends on proper specification of boundary conditions. We examine the effect of free- and no-slip boundary conditions. For the Cahn–Hilliard equation, we use a non-homogeneous Neumann condition for the concentration [33], and a homogeneous Neumann condition for the chemical potential,

$$-\frac{3}{2} \sigma \varepsilon \frac{\partial c}{\partial \vec{n}} \Big|_{\partial\Omega} = f'_w(c), \quad \frac{\partial \mu}{\partial \vec{n}} \Big|_{\partial\Omega} = 0,\tag{57}$$

where $f_w(c)$ is the boundary free-energy function that controls the wall contact angle. One choice is to use the function [33]

$$f_w(c) = \frac{1}{2} \sigma \cos \theta_w (2c - 1)(1 + 2c - 2c^2), \quad f'_w(c) = 6\sigma \cos \theta_w c(1 - c),\tag{58}$$

where θ_w is the imposed contact angle with the wall. In most simulations performed in this work we use a 90° angle, which simplifies $f_w(c) = 0$.

For free-slip boundary conditions, we impose zero normal velocity and zero normal stress for momentum, while for no-slip walls, all velocity components are zero ($\vec{u} = 0$), and we do not impose any conditions on the stress tensor. Either way, the boundary entropy flux is,

$$\begin{aligned} \vec{f}^{\mathcal{E}} \cdot \vec{n} &= -\frac{3}{2}\sigma\varepsilon c_t \vec{\nabla}c \cdot \vec{n} + \left(\frac{1}{2}\rho v_{tot}^2 + p + \mu c\right) \vec{u} \cdot \vec{n} + 2\eta \vec{n} \cdot \mathbf{S} \cdot \vec{u} + M_0 \mu \vec{\nabla} \mu \cdot \vec{n}. \\ &= c_t f'_w(c) = \frac{df_w}{dt}, \end{aligned} \quad (59)$$

i.e., both entropy fluxes for the free- and no-slip walls coincide. The entropy balance (55) with wall boundary conditions is therefore

$$\frac{d}{dt} \left(\bar{\mathcal{E}} + \int_{\partial\Omega} f_w(c) dS \right) = - \int_{\Omega} \left(M_0 |\vec{\nabla} \mu|^2 + 2\eta \mathbf{S} : \mathbf{S} \right) d\vec{x} \leq 0, \quad (60)$$

where the volume entropy is augmented with the surface free energy as in [33, 8].

3 Space and time discretization

We now construct the entropy-stable DGSEM approximation. We restrict ourselves to the tensor product DGSEM with Gauss-Lobatto (GL) points, since it satisfies the Summation-By-Parts Simultaneous-Approximation-Term (SBP-SAT) property [11]. The SBP-SAT property allows us to discretely follow the continuous stability steps to construct a discrete entropy law.

3.1 Differential geometry and curvilinear elements

The physical domain Ω is tessellated with non-overlapping hexahedral elements, e , which are geometrically transformed from a reference element $E = [-1, 1]^3$. This transformation is performed using a (polynomial) transfinite mapping \vec{X} that relates physical coordinates ($\vec{x} = (x^1, x^2, x^3) = (x, y, z) = x\hat{x} + y\hat{y} + z\hat{z}$) to local reference coordinates ($\vec{\xi} = (\xi^1, \xi^2, \xi^3) = (\xi, \eta, \zeta) = \xi\hat{\xi} + \eta\hat{\eta} + \zeta\hat{\zeta}$) through

$$\vec{x} = \vec{X}(\vec{\xi}) = \vec{X}(\xi, \eta, \zeta). \quad (61)$$

The space vectors \hat{x}_i and $\hat{\xi}^i$ are unit vectors in the three Cartesian directions of physical and reference coordinates, respectively.

From the transformation (61) one can define three covariant basis vectors,

$$\vec{a}_i = \frac{\partial \vec{X}}{\partial \xi^i}, \quad i = 1, 2, 3, \quad (62)$$

and three contravariant basis vectors,

$$\vec{a}^i = \vec{\nabla} \xi^i = \frac{1}{J} (\vec{a}_j \times \vec{a}_k), \quad (i, j, k) \text{ cyclic}, \quad (63)$$

where

$$J = \vec{a}_1 \cdot (\vec{a}_2 \times \vec{a}_3) \quad (64)$$

is the Jacobian of the mapping \vec{X} . The contravariant coordinate vectors satisfy the metric identities [34],

$$\sum_{i=1}^3 \frac{\partial (Ja_n^i)}{\partial \xi^i} = 0, \quad n = 1, 2, 3, \quad (65)$$

where a_n^i is the n -th Cartesian component of the contravariant vector \vec{a}^i .

We use the volume weighted contravariant basis $J\vec{a}^i$ to transform differential operators from physical ($\vec{\nabla}$) to reference ($\vec{\nabla}_\xi$) space. The divergence of a vector is [16]

$$\vec{\nabla} \cdot \vec{f} = \frac{1}{J} \vec{\nabla}_\xi \cdot (\mathbf{M}^T \vec{f}), \quad (66)$$

where $\mathbf{M} = (J\vec{a}^\xi, J\vec{a}^\eta, J\vec{a}^\zeta)$. We use (28) to write the divergence of an entire block vector compactly. Thus, we define the block matrix \mathcal{M} ,

$$\mathcal{M} = \begin{pmatrix} Ja_1^1 \mathbf{I}_5 & Ja_1^2 \mathbf{I}_5 & Ja_1^3 \mathbf{I}_5 \\ Ja_2^1 \mathbf{I}_5 & Ja_2^2 \mathbf{I}_5 & Ja_2^3 \mathbf{I}_5 \\ Ja_3^1 \mathbf{I}_5 & Ja_3^2 \mathbf{I}_5 & Ja_3^3 \mathbf{I}_5 \end{pmatrix}, \quad (67)$$

which allows us to write (66) for all the state variables simultaneously,

$$\vec{\nabla} \cdot \vec{\mathbf{f}} = \frac{1}{J} \vec{\nabla}_\xi \cdot (\mathcal{M}^T \vec{\mathbf{f}}) = \frac{1}{J} \vec{\nabla}_\xi \cdot \vec{\mathbf{f}}, \quad (68)$$

with $\vec{\mathbf{f}}$ being the block vector of the contravariant fluxes,

$$\vec{\mathbf{f}} = \mathcal{M}^T \vec{\mathbf{f}}, \quad \vec{\mathbf{f}}^i = J\vec{a}^i \cdot \vec{\mathbf{f}}. \quad (69)$$

The gradient of a scalar is [16]

$$\vec{\nabla} w = \frac{1}{J} \mathbf{M} \vec{\nabla}_\xi w, \quad (70)$$

which we can also extend to all entropy variables using (67),

$$\vec{\mathbf{g}} = \vec{\nabla} \mathbf{w} = \frac{1}{J} \mathcal{M} \vec{\nabla}_\xi \mathbf{w}, \quad (71)$$

and to non-conservative terms,

$$\vec{\phi}_m \cdot \vec{\nabla} w_m = \frac{1}{J} \vec{\phi}_m \cdot \vec{\nabla}_\xi w_m. \quad (72)$$

To transform the iNS/CH system, (20), into reference space, we first write it as a first order system. To do so, we define the auxiliary variables $\vec{\mathbf{g}} = \vec{\nabla} \mathbf{w}$ and $\vec{g}_c = \vec{\nabla} c$ so that

$$\begin{aligned} \underline{\mathbf{m}} \mathbf{q}_t + \vec{\nabla} \cdot \vec{\mathbf{f}}_e(\mathbf{q}) + \sum_{m=1}^5 \left(\vec{\phi}_m(\mathbf{q}) \cdot \vec{\nabla} w_m \right) &= \vec{\nabla} \cdot \vec{\mathbf{f}}_v(\vec{\mathbf{g}}) + \mathbf{s}(\mathbf{q}), \\ \vec{\mathbf{g}} &= \vec{\nabla} \mathbf{w}, \\ \mu &= f'_0(c) - \frac{3}{2} \sigma \varepsilon \vec{\nabla} \cdot \vec{g}_c, \\ \vec{g}_c &= \vec{\nabla} c. \end{aligned} \quad (73)$$

Recall that the viscous fluxes of the incompressible NSE depend only on the gradient of the entropy variables, $\vec{\mathbf{g}}$, and not on the state vector \mathbf{q} . Note that in the first equation the non-conservative terms are written in terms of $\vec{\nabla} \mathbf{w}$ and not in terms of $\vec{\mathbf{g}}$. This allows us to construct an entropy-stable scheme.

Next, we transform the operators to reference space using (68), (71), and (72)

$$J \underline{\mathbf{m}} \mathbf{q}_t + \vec{\nabla}_\xi \cdot \vec{\mathbf{f}}_e(\mathbf{q}) + \sum_{m=1}^5 \left(\vec{\phi}_m(\mathbf{q}) \cdot \vec{\nabla}_\xi w_m \right) = \vec{\nabla}_\xi \cdot \vec{\mathbf{f}}_v(\vec{\mathbf{g}}) + J \mathbf{s}(\mathbf{q}), \quad (74a)$$

$$J \vec{\mathbf{g}} = \mathcal{M} \vec{\nabla}_\xi \mathbf{w}, \quad (74b)$$

$$J \mu = J f'_0(c) - \frac{3}{2} \sigma \varepsilon \vec{\nabla}_\xi \cdot \vec{g}_c, \quad (74c)$$

$$J \vec{g}_c = \mathcal{M} \vec{\nabla}_\xi c, \quad (74d)$$

to obtain the final form of the equations to be approximated.

The DG approximation is obtained from weak forms of the equations (74). We first define the inner product in the reference element, E , for state and block vectors

$$\langle \mathbf{f}, \mathbf{g} \rangle_E = \int_E \mathbf{f}^T \mathbf{g} \, dE, \quad \langle \vec{\mathbf{f}}, \vec{\mathbf{g}} \rangle_E = \int_E \vec{\mathbf{f}} \cdot \vec{\mathbf{g}} \, dE. \quad (75)$$

We construct four weak forms by multiplying (74) by four test functions φ_q , $\vec{\varphi}_g$, φ_μ , and $\vec{\varphi}_c$, then we integrate over the reference element E , and finally we integrate by parts to get

$$\begin{aligned} \langle J \underline{\mathbf{m}} \mathbf{q}_t, \varphi_q \rangle_E + \int_{\partial E} \varphi_q^T \left(\vec{\mathbf{f}}_e + \sum_{m=1}^5 \vec{\phi}_m w_m - \vec{\mathbf{f}}_v \right) \cdot \hat{n} \, dS_\xi - \langle \vec{\mathbf{f}}_e, \vec{\nabla}_\xi \varphi_q \rangle_E \\ - \sum_{m=1}^5 \langle w_m, \vec{\nabla}_\xi \cdot \left(\varphi_q^T \vec{\phi}_m \right) \rangle_E = - \langle \vec{\mathbf{f}}_v, \vec{\nabla}_\xi \varphi_q \rangle_E + \langle J \mathbf{s}, \varphi_q \rangle_E, \\ \langle J \vec{\mathbf{g}}, \vec{\varphi}_g \rangle_E = \int_{\partial E} \mathbf{w}^T (\vec{\varphi}_g \cdot \hat{n}) \, dS_\xi - \langle \mathbf{w}, \vec{\nabla}_\xi \cdot \vec{\varphi}_g \rangle_E, \\ \langle J \mu, \varphi_\mu \rangle_E = \langle J f'_0, \varphi_\mu \rangle_E - \int_{\partial E} \frac{3}{2} \sigma \varepsilon \varphi_\mu \vec{g}_c \cdot \hat{n} \, dS_\xi + \left\langle \frac{3}{2} \sigma \varepsilon \vec{g}_c, \vec{\nabla}_\xi \varphi_\mu \right\rangle_E, \\ \langle J \vec{g}_c, \vec{\varphi}_c \rangle_E = \int_{\partial E} c \vec{\varphi}_c \cdot \hat{n} \, dS_\xi - \langle c, \vec{\nabla}_\xi \cdot \vec{\varphi}_c \rangle_E. \end{aligned} \quad (76)$$

The quantities \hat{n} and dS_ξ are the unit outward pointing normal and surface differential at the faces of E , respectively. The contravariant test functions $\vec{\varphi}_g$ and $\vec{\varphi}_c$ follow the definition (69). Finally, surface integrals extend to all six faces of an element,

$$\int_{\partial E} \vec{f} \cdot \hat{n} \, dS_\xi = \int_{[-1,1]^2} \tilde{f}^1 \, d\eta \, d\zeta \Big|_{\xi=-1}^{\xi=1} + \int_{[-1,1]^2} \tilde{f}^2 \, d\xi \, d\zeta \Big|_{\eta=-1}^{\eta=1} + \int_{[-1,1]^2} \tilde{f}^3 \, d\xi \, d\eta \Big|_{\zeta=-1}^{\zeta=1}. \quad (77)$$

We can write surface integrals in either physical or reference space. The relation between physical and reference surface differentials is given by,

$$dS^i = |J \vec{a}^i| \, d\xi^j \, d\xi^k = \mathcal{J}_f^i \, dS_\xi^i, \quad (78)$$

where we have defined the face Jacobian $\mathcal{J}_f^i = |\mathcal{J}\vec{a}^i|$. We can write the surface flux in either reference element, $\vec{f} \cdot \hat{n}$, or physical, $\vec{f} \cdot \vec{n}$, variables through

$$\vec{f} \cdot \hat{n}^i \, dS_\xi = \left(\mathcal{M}^T \vec{f} \right) \cdot \hat{n}^i \, dS_\xi = \vec{f} \cdot \left(\mathcal{M} \hat{n}^i \right) \, dS_\xi = \vec{f} \cdot \vec{n} \, |J\vec{a}^i| \, dS_\xi = \vec{f} \cdot \vec{n}^i \, dS. \quad (79)$$

Therefore, the surface integrals can be written in both physical and reference spaces,

$$\int_{\partial E} \vec{f} \cdot \hat{n} \, dS_\xi = \int_{\partial e} \vec{f} \cdot \vec{n} \, dS, \quad (80)$$

and we will use one or the other depending on whether we are studying an isolated element (reference space) or the entire mesh (physical space).

3.2 Polynomial approximation and the DGSEM

We now construct the discrete version of (76). The approximation of the state vector inside each element E is an order N polynomial,

$$\mathbf{q} \approx \mathbf{Q} \left(\vec{\xi} \right) = \sum_{i,j,k=0}^N \mathbf{Q}_{ijk}(t) l_i(\xi) l_j(\eta) l_k(\zeta) \in \mathbb{P}^N, \quad (81)$$

where \mathbb{P}^N is the space of polynomials of degree less than or equal to N on $[-1, 1]^3$. The state values $\mathbf{Q}_{ijk}(t) = \mathbf{Q}(\xi_i, \eta_j, \zeta_k, t)$ are the nodal degrees of freedom (time dependent coefficients) at the tensor product of each of the Gauss–Lobatto (GL) points $\{\xi_i\}_{i=0}^N$. Then, Lagrange polynomials $l_i(\xi)$ are

$$l_i(\xi) = \prod_{\substack{j=0 \\ j \neq i}}^N \frac{\xi - \xi_j}{\xi_i - \xi_j}. \quad (82)$$

The geometry and metric terms are also approximated with order N polynomials. Let us denote \mathbb{I}^N as the polynomial interpolation operator [35]. The transfinite mapping is approximated using $\mathcal{X} = \mathbb{I}^N(X)$, but special attention must be paid to its derivatives (i.e. the contravariant basis) since $\mathcal{J}\vec{a}^i \neq \mathbb{I}^N(\vec{a}_j \times \vec{a}_k)$. For the metric identities (65) to hold discretely,

$$\sum_{i=1}^3 \frac{\partial \mathbb{I}^N(Ja_n^i)}{\partial \xi^i} = 0, \quad n = 1, 2, 3, \quad (83)$$

we approximate metric terms using the curl form [34],

$$\mathcal{J}a_n^i = -\hat{x}_i \cdot \vec{\nabla}_\xi \times \left(\mathbb{I}^N \left(\mathcal{X}_i \vec{\nabla}_\xi \mathcal{X}_m \right) \right) \in \mathbb{P}^N, \quad i = 1, 2, 3, \quad n = 1, 2, 3, \quad (n, m, l) \text{ cyclic}. \quad (84)$$

If we compute $\mathcal{J}\vec{a}^i$ using (84), we ensure discrete free–stream preservation, which is crucial to avoid grid induced solution changes [34].

Next, we approximate integrals that arise in the weak formulation using Gauss quadratures. Let $\{w_i\}_{i=0}^N$ be the quadrature weights associated to Gauss–Lobatto nodes $\{\xi_i\}_{i=0}^N$. Then, in one dimension,

$$\int_{-1}^1 f(\xi) d\xi \approx \int_N f(\xi) d\xi = \sum_{m=0}^N w_m f(\xi_m) = \sum_{m=0}^N w_m F_m. \quad (85)$$

For Gauss–Lobatto points, the approximation is exact if $f(\xi) \in \mathbb{P}^{2N-1}$. The extension to three dimensions has three nested quadratures, one for each of the three reference space dimensions. For example, the inner product is

$$\langle f, g \rangle_E \approx \langle f, g \rangle_{E,N} = \sum_{m,n,l=0}^N w_{mnl} F_{mnl} G_{mnl}, \quad w_{mnl} = w_m w_n w_l, \quad (86)$$

with a similar definition for block vectors. The approximation of surface integrals is performed similarly, replacing exact integrals by Gauss quadratures in (77),

$$\begin{aligned} \int_{\partial E} \vec{f} \cdot \hat{n} dS_\xi &\approx \int_{\partial E,N} \vec{f} \cdot \hat{n} dS_\xi = \int_N \tilde{f}^1 d\eta d\zeta \Big|_{\xi=-1}^{\xi=1} + \int_N \tilde{f}^2 d\xi d\zeta \Big|_{\eta=-1}^{\eta=1} + \int_N \tilde{f}^3 d\xi d\eta \Big|_{\zeta=-1}^{\zeta=1} \\ &= \sum_{j,k=0}^N w_{jk} (\tilde{F}_{Njk}^1 - \tilde{F}_{0jk}^1) + \sum_{i,k=0}^N w_{ik} (\tilde{F}_{iNk}^2 - \tilde{F}_{i0k}^2) + \sum_{i,j=0}^N w_{ij} (\tilde{F}_{ijN}^3 - \tilde{F}_{ij0}^3). \end{aligned} \quad (87)$$

Gauss–Lobatto points are used to construct entropy–stable schemes using split–forms [11, 18]. There is no need to perform an interpolation from the volume polynomials in (87) to the boundaries, since boundary points are included, which is known as the Simultaneous–Approximation–Term (SAT) property. The exactness of the numerical quadrature and the SAT property yield the discrete Gauss law [11, 36]: for any polynomials $\vec{\mathbf{F}}$ and \mathbf{V} in \mathbb{P}^N ,

$$\langle \vec{\nabla}_\xi \cdot \vec{\mathbf{F}}, \mathbf{V} \rangle_{E,N} = \int_{\partial E,N} (\vec{\mathbf{F}} \cdot \hat{n}) \mathbf{V} dS_\xi - \langle \vec{\mathbf{F}}, \vec{\nabla}_\xi \mathbf{V} \rangle_{E,N}. \quad (88)$$

We start the discretization with the insertion of the discontinuous polynomial ansatz (81), (86), and (87) into the continuous weak forms (76),

$$\begin{aligned} \langle \mathcal{J} \underline{\mathbf{M}} \mathbf{Q}_t, \varphi_q \rangle_{E,N} &+ \int_{\partial E,N} \varphi_q^T \left(\vec{\mathbf{F}}_e + \sum_{m=1}^5 \vec{\mathbf{F}}_m W_m - \vec{\mathbf{F}}_v \right) \cdot \hat{n} dS_\xi - \langle \vec{\mathbf{F}}_e, \vec{\nabla}_\xi \varphi_q \rangle_{E,N} \\ &- \sum_{m=1}^5 \langle W_m, \vec{\nabla}_\xi \cdot (\varphi_q^T \vec{\mathbf{F}}_m) \rangle_{E,N} = - \langle \vec{\mathbf{F}}_v, \vec{\nabla}_\xi \varphi_q \rangle_{E,N} + \langle \mathcal{J} \mathbf{S}, \varphi_q \rangle_{E,N}, \\ \langle \mathcal{J} \vec{\mathbf{G}}, \vec{\varphi}_g \rangle_{E,N} &= \int_{\partial E,N} \mathbf{W}^T (\vec{\varphi}_g \cdot \hat{n}) dS_\xi - \langle \mathbf{W}, \vec{\nabla}_\xi \cdot \vec{\varphi}_g \rangle_{E,N}, \\ \langle \mathcal{J} \mu, \varphi_\mu \rangle_{E,N} &= \langle \mathcal{J} F'_0, \varphi_\mu \rangle_{E,N} - \int_{\partial E,N} \frac{3}{2} \sigma_\varepsilon \varphi_\mu \vec{G}_c \cdot \hat{n} dS_\xi + \left\langle \frac{3}{2} \sigma_\varepsilon \vec{G}_c, \vec{\nabla}_\xi \varphi_\mu \right\rangle_{E,N}, \\ \langle \mathcal{J} \vec{G}_c, \vec{\varphi}_c \rangle_{E,N} &= \int_{\partial E,N} C \vec{\varphi}_c \cdot \hat{n} dS_\xi - \langle C, \vec{\nabla}_\xi \cdot \vec{\varphi}_c \rangle_{E,N}. \end{aligned} \quad (89)$$

In (89), the test functions are restricted to polynomial spaces, \mathbb{P}^N . Taken in context, μ (which lacks an upper case symbol) refers to the polynomial approximation of the chemical potential.

Euler conservative fluxes and viscous fluxes have inter–element coupling and physical boundary conditions enforced by *numerical fluxes* in the element boundary quadratures in (89),

$$\begin{aligned}\vec{\mathbf{F}}_e &\approx \vec{\mathbf{F}}_e^*(\mathbf{Q}_L, \mathbf{Q}_R), \quad \vec{\mathbf{F}}_v \approx \vec{\mathbf{F}}_v^*(\vec{\mathbf{G}}_L, \vec{\mathbf{G}}_R), \quad \mathbf{W} \approx \mathbf{W}^*(\mathbf{Q}_L, \mathbf{Q}_R), \\ \vec{G}_c &\approx \vec{G}_c^*(\vec{G}_{cL}, \vec{G}_{cR}), \quad C \approx C^*(C_L, C_R).\end{aligned}\tag{90}$$

Whereas for non–conservative terms, we follow [37] and use *diamond fluxes* at the boundaries,

$$\vec{\Phi}_m W_m \approx \left(\vec{\Phi}_m W_m \right)^\diamond (\mathbf{Q}_L, \mathbf{Q}_R).\tag{91}$$

Both numerical and diamond flux functions are detailed below in Sec. 3.3. In (90) and (91), L and R represent the values from the left and right adjacent elements. Although numerical fluxes are single valued at each interface, diamond fluxes are not so constrained and their value can jump from one side to the other.

Inserting the numerical (90) and diamond (91) fluxes into (89) completes the semi–discretization,

$$\begin{aligned}\langle \mathcal{J}\underline{\mathbf{M}}\mathbf{Q}_t, \varphi_q \rangle_{E,N} + \int_{\partial E,N} \varphi_q^T \left(\vec{\mathbf{F}}_e^* + \sum_{m=1}^5 \left(\vec{\Phi}_m W_m \right)^\diamond - \vec{\mathbf{F}}_v^* \right) \cdot \hat{n} \, dS_\xi - \langle \vec{\mathbf{F}}_e, \vec{\nabla}_\xi \varphi_q \rangle_{E,N} \\ - \sum_{m=1}^5 \langle W_m, \vec{\nabla}_\xi \cdot (\varphi_q^T \vec{\Phi}_m) \rangle_{E,N} = - \langle \vec{\mathbf{F}}_v, \vec{\nabla}_\xi \varphi_q \rangle_{E,N} + \langle \mathcal{J}\mathbf{S}, \varphi_q \rangle_{E,N},\end{aligned}\tag{92a}$$

$$\langle \mathcal{J}\vec{\mathbf{G}}, \vec{\varphi}_g \rangle_{E,N} = \int_{\partial E,N} \mathbf{W}^{*,T} (\vec{\varphi}_g \cdot \hat{n}) \, dS_\xi - \langle \mathbf{W}, \vec{\nabla}_\xi \cdot \vec{\varphi}_g \rangle_{E,N},\tag{92b}$$

$$\langle \mathcal{J}\mu, \varphi_\mu \rangle_{E,N} = \langle \mathcal{J}F_0', \varphi_\mu \rangle_{E,N} - \int_{\partial E,N} \frac{3}{2} \sigma \varepsilon \varphi_\mu \vec{G}_c^* \cdot \hat{n} \, dS_\xi + \left\langle \frac{3}{2} \sigma \varepsilon \vec{G}_c, \vec{\nabla}_\xi \varphi_\mu \right\rangle_{E,N},\tag{92c}$$

$$\langle \mathcal{J}\vec{G}_c, \vec{\varphi}_c \rangle_{E,N} = \int_{\partial E,N} C^* \vec{\varphi}_c \cdot \hat{n} \, dS_\xi - \langle C, \vec{\nabla}_\xi \cdot \vec{\varphi}_c \rangle_{E,N}.\tag{92d}$$

Because it enhances the algorithm efficiency and makes it easier to show stability, we apply the discrete Gauss law (88) to the non–conservative terms and inviscid fluxes of (92a), in (92b), and in (92d), and

use (80) to write surface integrals in physical variables,

$$\begin{aligned} \langle \mathcal{J}\underline{\mathbf{M}}\mathbf{Q}_t, \varphi_q \rangle_{E,N} &+ \int_{\partial e,N} \varphi_q^T \left(\vec{\mathbf{F}}_e^* - \vec{\mathbf{F}}_e + \sum_{m=1}^5 \left(\vec{\mathbf{\Phi}}_m W_m \right)^\diamond - \vec{\mathbf{\Phi}}_m W_m \right) \cdot \vec{n} \, dS \\ &+ \langle \varphi_q, \vec{\nabla}_\xi \cdot \vec{\mathbf{F}}_e \rangle_{E,N} + \sum_{m=1}^5 \langle \varphi_q, \vec{\mathbf{\Phi}}_m \cdot \vec{\nabla}_\xi W_m \rangle_{E,N} \\ &= \int_{\partial e,N} \varphi_q^T \vec{\mathbf{F}}_v^* \cdot \vec{n} \, dS - \langle \vec{\mathbf{F}}_v, \vec{\nabla}_\xi \varphi_q \rangle_{E,N} + \langle \mathcal{J}\mathbf{S}, \varphi_q \rangle_{E,N}, \end{aligned} \quad (93a)$$

$$\langle \mathcal{J}\vec{\mathbf{G}}, \vec{\varphi}_g \rangle_{E,N} = \int_{\partial e,N} (\mathbf{W}^{*,T} - \mathbf{W}^T) (\vec{\varphi}_g \cdot \vec{n}) \, dS + \langle \vec{\varphi}_g, \vec{\nabla}_\xi \mathbf{W} \rangle_{E,N}, \quad (93b)$$

$$\langle \mathcal{J}\mu, \varphi_\mu \rangle_{E,N} = \langle \mathcal{J}F'_0, \varphi_\mu \rangle_{E,N} - \int_{\partial e,N} \frac{3}{2} \sigma \varepsilon \varphi_\mu \vec{G}_c^* \cdot \vec{n} \, dS + \left\langle \frac{3}{2} \sigma \varepsilon \vec{G}_c, \vec{\nabla}_\xi \varphi_\mu \right\rangle_{E,N}, \quad (93c)$$

$$\langle \mathcal{J}\vec{G}_c, \vec{\varphi}_c \rangle_{E,N} = \int_{\partial e,N} (C^* - C) \vec{\varphi}_c \cdot \vec{n} \, dS + \langle \vec{\varphi}_c, \vec{\nabla}_\xi C \rangle_{E,N}. \quad (93d)$$

The algorithm efficiency is improved by the use of single calculation of the local gradient of the entropy variables, $\vec{\nabla}_\xi \mathbf{W}$, for both non-conservative terms in (93a) and gradients in (93b). As a result, non-conservative terms are as expensive as source terms (i.e. no additional matrix multiplications are required). Then for the gradients $\vec{\mathbf{G}}$ we augment the pre-computed local gradients $\vec{\nabla}_\xi \mathbf{W}$ with the surface integral in (93b).

The point-wise discretization is obtained by replacing the test function in (93) by the Lagrange polynomials. For completeness, we write the implementation of the discretization in Appendix C.

3.3 Numerical fluxes

The approximation (93) is completed with the specification of numerical fluxes \mathbf{F}_e^* , \mathbf{F}_v^* , \mathbf{W}^* , \vec{G}_c^* , C^* , and diamond fluxes $(\vec{\mathbf{\Phi}}_m W_m)^\diamond$. For the inviscid fluxes we propose two options: an entropy conserving option using using central fluxes, and an entropy-stable approximation using the exact Riemann solver derived in [25]. For the viscous fluxes and concentration gradient, we use the Bassi-Rebay 1 (BR1) scheme [38].

Before we write the numerical fluxes, we first describe the rotational invariance property satisfied by inviscid fluxes and non-conservative terms. The rotational invariance of the flux [39] allows us to write the normal flux $\vec{\mathbf{F}}_e \cdot \vec{n}$ from a rotated version of the inviscid flux x -component \mathbf{F}_e ,

$$\vec{\mathbf{F}}_e \cdot \vec{n} = \underline{\mathbf{T}}^T \mathbf{F}_e (\underline{\mathbf{T}}\mathbf{Q}) = \underline{\mathbf{T}}^T \mathbf{F}_e (\mathbf{Q}_n), \quad \underline{\mathbf{T}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & n_x & n_y & n_z & 0 \\ 0 & t_{1,x} & t_{1,y} & t_{1,z} & 0 \\ 0 & t_{2,x} & t_{2,y} & t_{2,z} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad (94)$$

where $\underline{\mathbf{T}}$ is a rotation matrix that only affects velocities, $\vec{n} = (n_x, n_y, n_z)$ is the normal unit vector to the face, and \vec{t}_1 and \vec{t}_2 are two tangent unit vectors to the face. When the rotation matrix $\underline{\mathbf{T}}$ multiplies

the state vector \mathbf{Q} , we obtain the face normal state vector \mathbf{Q}_n ,

$$\mathbf{Q}_n = \mathbf{T}\mathbf{Q} = (C, \sqrt{\rho}U_n, \sqrt{\rho}V_{t1}, \sqrt{\rho}V_{t2}, P), \quad (95)$$

where $U_n = \vec{U} \cdot \vec{n}$ the normal velocity, and $V_{ti} = \vec{U} \cdot \vec{t}_i$ ($i = 1, 2$) are the two tangent velocities. Note that the reference system rotation does not affect the total speed

$$V_{tot}^2 = U^2 + V^2 + W^2 = U_n^2 + V_{t1}^2 + V_{t2}^2. \quad (96)$$

The non-conservative terms,

$$\sum_{m=1}^5 \vec{\Phi}_m W_m = \mathcal{T} \begin{pmatrix} 0 & 0 & 0 \\ \frac{1}{2}\rho U^2 + \mu C & \frac{1}{2}\rho UV & \frac{1}{2}\rho UW \\ \frac{1}{2}\rho UV & \frac{1}{2}\rho V^2 + \mu C & \frac{1}{2}\rho VW \\ \frac{1}{2}\rho UW & \frac{1}{2}\rho VW & \frac{1}{2}\rho W^2 + \mu C \\ \rho_0 c_0^2 U & \rho_0 c_0^2 V & \rho_0 c_0^2 W \end{pmatrix}, \quad (97)$$

are also rotationally invariant,

$$\sum_{m=1}^5 \left(\vec{\Phi}_m W_m \right) \cdot \vec{n} = \mathbf{T}^T \begin{pmatrix} 0 \\ \frac{1}{2}\rho U_n^2 + \mu C \\ \frac{1}{2}\rho U_n V_{t1} \\ \frac{1}{2}\rho U_n V_{t2} \\ \rho_0 c_0^2 U_n \end{pmatrix} = \mathbf{T}^T \mathbf{F}_{\phi W}(\mathbf{Q}_n), \quad (98)$$

with its equivalent x -component,

$$\mathbf{f}_{\phi w}(\mathbf{q}) = \begin{pmatrix} 0 \\ \frac{1}{2}\rho u^2 + \mu c \\ \frac{1}{2}\rho uv \\ \frac{1}{2}\rho uw \\ \rho_0 c_0^2 u \end{pmatrix}. \quad (99)$$

3.3.1 Inviscid fluxes: entropy conserving central fluxes

The first choice for the inviscid numerical and diamond fluxes is to use central fluxes, which will lead to an entropy conserving approximation. We adapt the approach used for the resistive MHD in [37], so that

$$\vec{\mathbf{F}}_e^* \cdot \vec{n} = \{\{\vec{\mathbf{F}}_e\}\} \cdot \vec{n} = \mathbf{T}^T \begin{pmatrix} \{\{CU_n\}\} \\ \{\{\frac{1}{2}\rho U_n^2 + p\}\} \\ \{\{\frac{1}{2}\rho U_n V_{t1}\}\} \\ \{\{\frac{1}{2}\rho U_n V_{t2}\}\} \\ 0 \end{pmatrix}, \quad (100)$$

and

$$\sum_{m=1}^5 \left(\vec{\Phi}_m W_m \right)^\diamond \cdot \vec{n} = \sum_{m=1}^5 \vec{\Phi}_m \{ \{ W_m \} \} \cdot \vec{n} = \mathbf{T}^T \begin{pmatrix} 0 \\ \frac{1}{2} \rho U_n \{ \{ U_n \} \} + C \{ \{ \mu \} \} \\ \frac{1}{2} \rho U_n \{ \{ V_{t1} \} \} \\ \frac{1}{2} \rho U_n \{ \{ V_{t2} \} \} \\ \rho_0 c_0^2 \{ \{ U_n \} \} \end{pmatrix}. \quad (101)$$

In (100) and (101), $\{ \{ u \} \}$ represents the average operator,

$$\{ \{ u \} \} = \frac{u_L + u_R}{2} \quad (102)$$

and $\vec{\Phi}_m$ (without average) represents the element local state at the face.

3.3.2 Inviscid fluxes: entropy-stable exact Riemann solver (ERS)

The second option uses the solution of the exact Riemann problem derived in [25]. The star region solution of the incompressible NSE (i.e. without the Cahn–Hilliard equation) is,

$$U_n^* = \frac{P_L - P_R + \rho_L U_{nL} \lambda_L^+ - \rho_R U_{nR} \lambda_R^-}{\rho_L \lambda_L^+ - \rho_R \lambda_R^-}, \quad P^* = P_L + \rho_L \lambda_L^+ (U_{nL} - U_n^*),$$

$$\rho^* = \begin{cases} \rho_L^* & \text{if } U_n^* \geq 0 \\ \rho_R^* & \text{if } U_n^* < 0 \end{cases}, \quad \rho_L^* = \frac{\rho_L \lambda_L^+}{U_n^* - \lambda_L^+}, \quad \rho_R^* = \frac{\rho_R \lambda_R^-}{U_n^* - \lambda_R^-}, \quad V_{ti}^* = \begin{cases} V_{tiL} & \text{if } U_n^* \geq 0 \\ V_{tiR} & \text{if } U_n^* < 0 \end{cases}, \quad (103)$$

with eigenvalues,

$$\lambda_L^\pm = \frac{U_{nL} \pm a_L}{2}, \quad \lambda_R^\pm = \frac{U_{nR} \pm a_R}{2}, \quad a = \sqrt{U_n^2 + \frac{4\rho_0 c_0^2}{\rho}}. \quad (104)$$

We evaluate the numerical flux with the star region solution with the normal state \mathbf{Q}_n in (94) for the momentum, and perform the average for the Cahn–Hilliard part,

$$\vec{\mathbf{F}}_e^* \cdot \vec{n} = \mathbf{T}^T \begin{pmatrix} \{ \{ C U_n \} \} \\ \frac{1}{2} \rho^* U_n^{*,2} + p^* \\ \frac{1}{2} \rho^* U_n^* V_{t1}^* \\ \frac{1}{2} \rho^* U_n^* V_{t2}^* \\ 0 \end{pmatrix}. \quad (105)$$

For non-conservative terms, we choose the diamond fluxes,

$$\sum_{m=1}^5 \left(\vec{\Phi}_m W_m \right)^\diamond \cdot \vec{n} = \vec{\Phi}_1 \{ \{ W_1 \} \} \cdot \vec{n} + \sum_{m=2}^5 \left(\vec{\Phi}_m^* W_m^* + \vec{\Phi}_m W_m - \vec{\Phi}_m^* W_m \right) \cdot \vec{n}$$

$$= \mathbf{T}^T \begin{pmatrix} 0 \\ \frac{1}{2} \rho^* U_n^{*,2} + \frac{1}{2} \rho U_n^2 - \frac{1}{2} \rho^* U_n^* U_n + C \{ \{ \mu \} \} \\ \frac{1}{2} \rho^* U_n^* V_{t1}^* + \frac{1}{2} \rho U_n V_{t1} - \frac{1}{2} \rho^* U_n^* V_{t1} \\ \frac{1}{2} \rho^* U_n^* V_{t2}^* + \frac{1}{2} \rho U_n V_{t2} - \frac{1}{2} \rho^* U_n^* V_{t2} \\ \rho_0 c_0^2 U_n^* \end{pmatrix} = \mathbf{T}^T \mathbf{F}_{\Phi W}^{\text{ERS}}, \quad (106)$$

where $\vec{\Phi}_m^*$ and W_m^* refer to non-conservative coefficients and entropy variables evaluated with the star region solution (103). This choice is justified by the stability analysis.

3.3.3 Viscous and chemical potential fluxes: Bassi–Rebay 1 (BR1) method

For the viscous fluxes and the chemical potential we use the Bassi–Rebay 1 (BR1) scheme [38], which averages entropy variables and fluxes between the adjacent elements. However, for the Cahn–Hilliard equation, we include the possibility to add interface stabilization. Although it is not a requirement for stability, we have found it enhances accuracy when the flow configuration is under–resolved [8]. The viscous and Cahn–Hilliard fluxes are,

$$\mathbf{W}^* = \{\{\mathbf{W}\}\}, \quad \vec{\mathbf{F}}_v^* = \{\{\vec{\mathbf{F}}_v\}\} + \beta \llbracket \mu \rrbracket \mathbf{e}_1 \vec{n}_L, \quad C^* = \{\{C\}\}, \quad \vec{G}_c^* = \{\{\vec{G}_c\}\} + \beta \llbracket C \rrbracket \vec{n}_L, \quad (107)$$

where β is the penalty parameter, computed in this work following [40, 8],

$$\beta = \kappa_\beta \frac{N(N+1)}{2} |\mathcal{J}_f| \{\{\mathcal{J}^{-1}\}\}, \quad (108)$$

with κ_β a dimensionless free parameter (in this work we only use $\kappa_\beta = 0$ to disable interface stabilization, and $\kappa_\beta = 1$ to enable interface stabilization), we define the inter–element jumps as $\llbracket \bullet \rrbracket = \bullet_R - \bullet_L$, and \vec{n}_L is the outward normal vector to the left element. In (108) N is the polynomial order, $|\mathcal{J}_f|$ is the surface Jacobian of the face, and $\{\{\mathcal{J}^{-1}\}\}$ is the average of the inverse of the Jacobians of left and right elements.

3.4 Boundary conditions

The approximation is completed with the addition of boundary conditions. Here we show how we impose free– and no–slip wall boundary conditions. The boundary conditions are imposed through the numerical and diamond fluxes at the physical boundary faces. We consider the inviscid and viscous fluxes individually.

3.4.1 Inviscid flux

The inviscid numerical flux controls the normal velocity $\vec{u} \cdot \vec{n} = 0$ for both free– and no–slip boundary conditions. Again, as at interior faces, we provide an entropy conserving option with central fluxes, and an entropy stable version with the exact Riemann problem solution. Either way, we apply central fluxes (100)–(101) or the ERS (105)–(106) to the interior state \mathbf{Q}_n^i and a mirrored ghost state \mathbf{Q}_n^e ,

$$\mathbf{Q}_n^i = \begin{pmatrix} C \\ \sqrt{\rho} U_n \\ \sqrt{\rho} V_{t1} \\ \sqrt{\rho} V_{t2} \\ P \end{pmatrix}, \quad \mathbf{Q}_n^e = \begin{pmatrix} C \\ -\sqrt{\rho} U_n \\ \sqrt{\rho} V_{t1} \\ \sqrt{\rho} V_{t2} \\ P \end{pmatrix}. \quad (109)$$

The fluxes differ depending on which numerical flux is used.

1. *The entropy conserving approximation with central fluxes.* The numerical (100) and diamond (101) central fluxes for (109) are,

$$\vec{\mathbf{F}}_e^* \cdot \vec{n} = \mathbf{T}^T \begin{pmatrix} 0 \\ \frac{1}{2}\rho U_n^2 + P \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \sum_{m=1}^5 \left(\vec{\Phi}_m W_m \right)^\diamond \cdot \vec{n} = \mathbf{T}^T \begin{pmatrix} 0 \\ C\mu \\ \frac{1}{2}\rho U_n V_{t1} \\ \frac{1}{2}\rho U_n V_{t2} \\ 0 \end{pmatrix}. \quad (110)$$

2. *The entropy stable approximation with the exact Riemann solver.* The star region values computed from the states (109) are

$$U_n^* = 0, \quad P^* = P + \rho\lambda^+ U_n, \quad \lambda^+ = \frac{U_n + a}{2}. \quad (111)$$

With these star values the numerical (105) and diamond (106) fluxes become

$$\vec{\mathbf{F}}_e^* \cdot \vec{n} = \mathbf{T}^T \begin{pmatrix} 0 \\ P + \rho\lambda^+ U_n \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \sum_{m=1}^5 \left(\vec{\Phi}_m W_m \right)^\diamond \cdot \vec{n} = \mathbf{T}^T \begin{pmatrix} 0 \\ \frac{1}{2}\rho U_n^2 + C\mu \\ \frac{1}{2}\rho U_n V_{t1} \\ \frac{1}{2}\rho U_n V_{t2} \\ 0 \end{pmatrix}. \quad (112)$$

3.4.2 Viscous and Cahn–Hilliard fluxes

For the free–slip wall boundary condition, we impose zero normal stress $\mathbf{S} \cdot \vec{n} = 0$ (Neumann), whereas for the no–slip wall boundary condition, we impose zero velocity $\vec{u} = 0$ (Dirichlet). Furthermore, we implement the discrete version of the homogeneous and non–homogeneous Neumann boundary conditions (57) for the Cahn–Hilliard equation. The concentration, concentration gradient, entropy variables, and viscous fluxes at the boundaries are:

1. *Free–slip wall boundary condition.*

$$C^* = C, \quad \vec{G}_c^* = g'(C), \quad \mathbf{W}^* = \mathbf{W}, \quad \vec{\mathbf{F}}_v^* \cdot \vec{n} = 0. \quad (113)$$

2. *No–slip wall boundary condition.*

$$C^* = C, \quad \vec{G}_c^* = g'(C), \quad \mathbf{W}^* = (\mu, 0, 0, 0, P), \quad \vec{\mathbf{F}}_v^* \cdot \vec{n} = \begin{pmatrix} 0 \\ 2\eta\mathcal{S} \\ 0 \end{pmatrix} \cdot \vec{n}. \quad (114)$$

3.5 Implicit–Explicit (IMEX) time discretization

The fully–discrete scheme is completed with the discretization of the time derivatives \mathbf{Q}_t in (93). The numerical stiffness induced by the fourth order derivatives present in the Cahn–Hilliard equation prevents us from practically using fully explicit time marching. Therefore, we use the IMplicit–EXplicit (IMEX)

Backward Differentiation Formula (BDF) time integrator described in [24]. We revisit the continuous setting (20)

$$\begin{aligned} c_t + \vec{\nabla} \cdot (c\vec{u}) &= M_0 \vec{\nabla}^2 \left(f'_0(c) - \frac{3}{2} \sigma \varepsilon \vec{\nabla}^2 c \right), \\ \sqrt{\rho} (\sqrt{\rho} \vec{u})_t + \vec{\nabla} \cdot \left(\frac{1}{2} \rho \vec{u} \vec{u} \right) + \frac{1}{2} \rho \vec{u} \cdot \vec{\nabla} \vec{u} + c \vec{\nabla} \mu &= -\vec{\nabla} p + 2\eta \mathbf{S} + \rho \vec{g}, \\ p_t + \rho_0 c_0^2 \vec{\nabla} \cdot \vec{u} &= 0, \end{aligned} \quad (115)$$

to describe the approximation in time.

The IMEX version of the BDF integrator for a time derivative dy/dt is

$$\gamma_0 = \begin{cases} 1 & \text{if } J = 1 \\ \frac{3}{2} & \text{if } J = 2 \end{cases}, \quad \hat{y} = \begin{cases} y^n & \text{if } J = 1 \\ 2y^n - \frac{1}{2}y^{n-1} & \text{if } J = 2 \end{cases}, \quad y^{n+1,e} = \begin{cases} y^n & \text{if } J = 1 \\ 2y^n - y^{n-1} & \text{if } J = 2 \end{cases}, \quad (116)$$

where $y^n = y(t_n)$, γ_0 and \hat{y} yield a J -th order approximation of the time derivative,

$$y_t^{n+1} \approx \frac{\gamma_0 y^{n+1} - \hat{y}}{\Delta t}, \quad (117)$$

and $y^{n+1,e}$ is a J -th order explicit approximation of y^{n+1} . The IMEX scheme used in [24] was constructed to solve the fourth order derivative implicitly, and the remaining terms explicitly,

$$\begin{aligned} \frac{\gamma_0 c^{n+1} - \hat{c}}{\Delta t} + \vec{\nabla} \cdot (c\vec{u})^{n+1,e} &= M_0 \vec{\nabla}^2 \left(f'_0(c^{n+1,e}) - \frac{3}{2} \sigma \varepsilon \vec{\nabla}^2 c^{n+1} \right), \\ \sqrt{\rho^{n+1,e}} \frac{\gamma_0 \sqrt{\rho} \vec{u}^{n+1} - \widehat{\sqrt{\rho} \vec{u}}}{\Delta t} + \left(\vec{\nabla} \cdot \left(\frac{1}{2} \rho \vec{u} \vec{u} \right) + \frac{1}{2} \rho \vec{u} \cdot \vec{\nabla} \vec{u} + c \vec{\nabla} \mu \right)^{n+1,e} &= \left(-\vec{\nabla} p + 2\eta \mathbf{S} + \rho \vec{g} \right)^{n+1,e}, \\ \frac{\gamma_0 p^{n+1} - \hat{p}}{\Delta t} + \rho_0 c_0^2 \vec{\nabla} \cdot \vec{u}^{n+1,e} &= 0. \end{aligned} \quad (118)$$

As in the vast majority of BDF implementations, the first step is first order ($J = 1$), and the following steps can be maintained first or use second order ($J = 2$). Note that the Jacobian matrix associated with the fourth order derivative is constant in time. Hence, only one computation of its solution is required. In this work, we perform a single LU decomposition, and afterwards use Gauss substitution to solve the linear system at each time step, but other direct or iterative solvers could be used.

One can use a fully explicit time integrator if the time step restriction allows it (e.g. when the chemical characteristic time is large enough). We have also implemented a low-storage third-order Runge–Kutta method to explicitly solve the system [41].

4 Semi-discrete stability analysis

We prove the stability of the spatial discretization (93), by showing that it mimics a discrete version of the entropy conservation law (55). As in the continuous analysis, we do not study the effect of lower

order source terms ($\mathbf{S} = 0$). The entropy analysis reproduces the continuous analysis steps discretely. Therefore, we use proper choices for the test functions, which contract the system of equations into a single equation, to be shaped into an entropy equation. The latter is produced in four steps, as we study the time derivative terms, inviscid terms, viscous terms, and finally interior and physical boundary terms.

First, we follow [8] and take the time derivative of (93d),

$$\left\langle \mathcal{J}\vec{G}_{c,t}, \vec{\varphi}_c \right\rangle_{E,N} = \int_{\partial e,N} (C_t^* - C_t) \vec{\varphi}_c \cdot \vec{n} \, dS + \left\langle \vec{\varphi}_c, \vec{\nabla}_\xi C_t \right\rangle_{E,N}, \quad (119)$$

and then replace the test functions. Following [16, 9], we insert $\varphi_q = \mathbf{W}$, and $\vec{\varphi}_g = \vec{\mathbf{F}}_v$, and following [8], $\varphi_\mu = C_t$, and $\vec{\varphi}_c = \frac{3}{2}\sigma\varepsilon\vec{G}_c$,

$$\begin{aligned} \langle \mathcal{J}\underline{\mathbf{M}}\mathbf{Q}_t, \mathbf{W} \rangle_{E,N} &+ \int_{\partial e,N} \mathbf{W}^T \left(\vec{\mathbf{F}}_e^* - \vec{\mathbf{F}}_e + \sum_{m=1}^5 \left(\vec{\boldsymbol{\Phi}}_m W_m \right)^\diamond - \vec{\boldsymbol{\Phi}}_m W_m \right) \cdot \vec{n} \, dS \\ &+ \left\langle \mathbf{W}, \vec{\nabla}_\xi \cdot \vec{\mathbf{F}}_e \right\rangle_{E,N} + \sum_{m=1}^5 \left\langle \mathbf{W}, \vec{\boldsymbol{\Phi}}_m \cdot \vec{\nabla}_\xi W_m \right\rangle_{E,N} \\ &= \int_{\partial e,N} \mathbf{W}^T \vec{\mathbf{F}}_v^* \cdot \vec{n} \, dS - \left\langle \vec{\mathbf{F}}_v, \vec{\nabla}_\xi \mathbf{W} \right\rangle_{E,N}, \end{aligned} \quad (120a)$$

$$\left\langle \mathcal{J}\vec{\mathbf{G}}, \vec{\mathbf{F}}_v \right\rangle_{E,N} = \int_{\partial e,N} (\mathbf{W}^{*,T} - \mathbf{W}^T) (\vec{\mathbf{F}}_v \cdot \vec{n}) \, dS + \left\langle \vec{\mathbf{F}}_v, \vec{\nabla}_\xi \mathbf{W} \right\rangle_{E,N}, \quad (120b)$$

$$\langle \mathcal{J}\mu, C_t \rangle_{E,N} = \langle \mathcal{J}F'_0, C_t \rangle_{E,N} - \int_{\partial e,N} \frac{3}{2}\sigma\varepsilon C_t \vec{G}_c^* \cdot \vec{n} \, dS + \left\langle \frac{3}{2}\sigma\varepsilon \vec{G}_c, \vec{\nabla}_\xi C_t \right\rangle_{E,N}, \quad (120c)$$

$$\frac{3}{2}\sigma\varepsilon \left\langle \mathcal{J}\vec{G}_{c,t}, \vec{G}_c \right\rangle_{E,N} = \int_{\partial e,N} \frac{3}{2}\sigma\varepsilon (C_t^* - C_t) \vec{G}_c \cdot \vec{n} \, dS + \left\langle \frac{3}{2}\sigma\varepsilon \vec{G}_c, \vec{\nabla}_\xi C_t \right\rangle_{E,N}. \quad (120d)$$

Next, we replace the last term in (120a) by the last term in (120b), and the last term in (120c) by the last term in (120d),

$$\begin{aligned} \langle \mathcal{J}\underline{\mathbf{M}}\mathbf{Q}_t, \mathbf{W} \rangle_{E,N} &+ \int_{\partial e,N} \mathbf{W}^T \left(\vec{\mathbf{F}}_e^* - \vec{\mathbf{F}}_e + \sum_{m=1}^5 \left(\vec{\boldsymbol{\Phi}}_m W_m \right)^\diamond - \vec{\boldsymbol{\Phi}}_m W_m \right) \cdot \vec{n} \, dS \\ &+ \left\langle \mathbf{W}, \vec{\nabla}_\xi \cdot \vec{\mathbf{F}}_e \right\rangle_{E,N} + \sum_{m=1}^5 \left\langle \mathbf{W}, \vec{\boldsymbol{\Phi}}_m \cdot \vec{\nabla}_\xi W_m \right\rangle_{E,N} \\ &= \int_{\partial e,N} \left(\mathbf{W}^T \vec{\mathbf{F}}_v^* + \mathbf{W}^{*,T} \vec{\mathbf{F}}_v - \mathbf{W}^T \vec{\mathbf{F}}_v \right) \cdot \vec{n} \, dS - \left\langle \mathcal{J}\vec{\mathbf{G}}, \vec{\mathbf{F}}_v \right\rangle_{E,N}, \end{aligned} \quad (121a)$$

$$\begin{aligned} \langle \mathcal{J}\mu, C_t \rangle_{E,N} &= \langle \mathcal{J}F'_0, C_t \rangle_{E,N} + \frac{3}{2}\sigma\varepsilon \left\langle \mathcal{J}\vec{G}_{c,t}, \vec{G}_c \right\rangle_{E,N} \\ &- \int_{\partial e,N} \frac{3}{2}\sigma\varepsilon \left(C_t \vec{G}_c^* + C_t^* \vec{G}_c - C_t \vec{G}_c \right) \cdot \vec{n} \, dS, \end{aligned} \quad (121b)$$

to combine four equations into two. Under the assumption of exactness in time, we can use the chain rule on (121b),

$$\begin{aligned} \langle \mathcal{J}F'_0, C_t \rangle_{E,N} + \frac{3}{2}\sigma\varepsilon \langle \mathcal{J}\vec{G}_{c,t}, \vec{G}_c \rangle_{E,N} &= \left\langle \mathcal{J} \left(F_0 + \frac{3}{4}\sigma\varepsilon |\vec{G}_c|^2 \right)_t, 1 \right\rangle_{E,N} \\ &= \langle \mathcal{J}\mathcal{F}_t, 1 \rangle_{E,N}, \end{aligned} \quad (122)$$

to obtain the time derivative of the discrete approximation of the free–energy (7).

We complete the analysis by examining each term in (121) separately to get the semi–discrete entropy law. The rest of the section computes the terms as follows: the time derivative coefficients in Sec. 4.1, inviscid volume integrals in Sec. 4.2, viscous volume integrals in Sec. 4.3, and lastly boundary integrals in Sec. 4.4. The discrete entropy law is completed in Sec. 4.5.

4.1 Time derivative coefficients

We first study the discrete inner product containing time derivatives found in (121a). We expand the inner product,

$$\langle \mathcal{J}\underline{\mathbf{M}}\mathbf{Q}_t, \mathbf{W} \rangle_{E,N} = \langle \mathcal{J}C_t, \mu \rangle_{E,N} + \left\langle \mathcal{J} \left(\sqrt{\rho}\vec{U} \right)_t, \sqrt{\rho}\vec{U} \right\rangle_{E,N} + \frac{1}{\rho_0 c_0^2} \langle \mathcal{J}P_t, P \rangle_{E,N}, \quad (123)$$

into which we insert the discrete approximation of $\langle \mathcal{J}C_t, \mu \rangle_{E,N}$ found in (121b), and apply the chain rule in time for the kinetic and artificial compressibility terms,

$$\begin{aligned} \langle \mathcal{J}\underline{\mathbf{M}}\mathbf{Q}_t, \mathbf{W} \rangle_{E,N} &= \langle \mathcal{J}\mathcal{F}_t, 1 \rangle_{E,N} - \int_{\partial e,N} \frac{3}{2}\sigma\varepsilon \left(C_t \vec{G}_c^* + C_t^* \vec{G}_c - C_t \vec{G}_c \right) \cdot \vec{n} \, dS \\ &\quad + \left\langle \mathcal{J} \left(\frac{1}{2}\rho V_{tot}^2 \right)_t, 1 \right\rangle_{E,N} + \left\langle \mathcal{J} \left(\frac{P^2}{2\rho_0 c_0^2} \right)_t, 1 \right\rangle_{E,N} \\ &= \langle \mathcal{J}(\mathcal{F} + \mathcal{K} + \mathcal{E}_{AC})_t, 1 \rangle_{E,N} \\ &\quad - \int_{\partial e,N} \frac{3}{2}\sigma\varepsilon \left(C_t \vec{G}_c^* + C_t^* \vec{G}_c - C_t \vec{G}_c \right) \cdot \vec{n} \, dS \\ &= \langle \mathcal{J}\mathcal{E}_t, 1 \rangle_{E,N} - \int_{\partial e,N} \frac{3}{2}\sigma\varepsilon \left(C_t \vec{G}_c^* + C_t^* \vec{G}_c - C_t \vec{G}_c \right) \cdot \vec{n} \, dS. \end{aligned} \quad (124)$$

Eq. (124) is the discrete version of (39). As a result, we get the discrete version of (39) for the entropy,

$$\mathcal{E} = \mathcal{F} + \mathcal{K} + \mathcal{E}_{AC} = F_0 + \frac{3}{4}\sigma\varepsilon |\vec{G}_c|^2 + \frac{1}{2}\rho V_{tot}^2 + \frac{P^2}{2\rho_0 c_0^2}, \quad (125)$$

and for the time entropy flux $\vec{F}_t^{\mathcal{E}}$, which appears as the argument of the surface integral in (124).

4.2 Inviscid volume terms

We now show that the contraction of the inviscid volume terms into a boundary entropy flux holds discretely as in the continuous analysis (47), i.e.,

$$\left\langle \mathbf{W}, \vec{\nabla}_\xi \cdot \vec{\mathbf{F}}_e \right\rangle_{E,N} + \sum_{m=1}^5 \left\langle \mathbf{W}, \vec{\Phi}_m \cdot \vec{\nabla}_\xi W_m \right\rangle_{E,N} = \int_{\partial e,N} \vec{F}^\mathcal{E} \cdot \vec{n} \, dS. \quad (126)$$

Two properties were used in the continuous analysis: the integration by parts (which holds discretely by the discrete Gauss law (88)), and the relationship between inviscid fluxes and non-conservative terms (44), which also holds for contravariant components,

$$\mathbf{e}_m^T \vec{\mathbf{F}}_e = \mathbf{e}_m^T \left(\mathcal{M}^T \vec{\mathbf{F}}_e \right) = \mathcal{M}^T \left(\mathbf{e}_m^T \vec{\mathbf{F}}_e \right) = \mathcal{M}^T \left(\mathbf{W}^T \vec{\Phi}_m \right) = \mathbf{W}^T \left(\mathcal{M}^T \vec{\Phi}_m \right) = \mathbf{W}^T \vec{\Phi}_m. \quad (127)$$

In (127), space and state multiplications commute since the matrix (67) is built from identity matrices state-wise.

Eq. (127) together with the discrete Gauss law (88) is sufficient to prove (126). We use the discrete Gauss law for the first term in (126), and explicitly write the scalar product of the resulting integral approximation as the sum of the components,

$$\begin{aligned} \left\langle \mathbf{W}, \vec{\nabla}_\xi \cdot \vec{\mathbf{F}}_e \right\rangle_{E,N} &= \int_{\partial e,N} \mathbf{W}^T \vec{\mathbf{F}}_e \cdot \vec{n} \, dS - \left\langle \vec{\mathbf{F}}_e, \vec{\nabla}_\xi \mathbf{W} \right\rangle_{E,N} \\ &= \int_{\partial e,N} \mathbf{W}^T \vec{\mathbf{F}}_e \cdot \vec{n} \, dS - \sum_{m=1}^5 \left\langle \mathbf{e}_m^T \vec{\mathbf{F}}_e, \vec{\nabla}_\xi W_m \right\rangle_{E,N} \\ &= \int_{\partial e,N} \vec{F}^\mathcal{E} \cdot \vec{n} \, dS - \sum_{m=1}^5 \left\langle \mathbf{W}^T \vec{\Phi}_m, \vec{\nabla}_\xi W_m \right\rangle_{E,N}. \end{aligned} \quad (128)$$

In the last identity, we used (45) and (127). Thus, when we move the non-conservative terms to the left hand side of (128), we prove (126).

4.3 Viscous volume terms

Analogously to (51), viscous volume terms are dissipative if

$$\left\langle \mathcal{J} \vec{\mathbf{G}}, \vec{\mathbf{F}}_v \left(\vec{\mathbf{G}} \right) \right\rangle_{E,N} \geq 0. \quad (129)$$

Thus, we closely follow the continuous steps,

$$\begin{aligned} \left\langle \mathcal{J} \vec{\mathbf{G}}, \vec{\mathbf{F}}_v \right\rangle_{E,N} &= M_0 \left\langle \mathcal{J} \vec{G}_\mu, \vec{G}_\mu \right\rangle_{E,N} + \left\langle \mathcal{J} \vec{G}_{\bar{u}}, 2\eta \mathcal{S} \right\rangle_{E,N} \\ &= M_0 \left\langle \mathcal{J} \vec{G}_\mu, \vec{G}_\mu \right\rangle_{E,N} + \left\langle \mathcal{J} \left(\frac{\vec{G}_{\bar{u}} + \vec{G}_{\bar{u}}^T}{2} \right), 2\eta \mathcal{S} \right\rangle_{E,N} \\ &= \left\langle \mathcal{J} \left(M_0 |\vec{G}_\mu|^2 + 2\eta \mathcal{S} : \mathcal{S} \right), 1 \right\rangle_{E,N} \geq 0, \end{aligned} \quad (130)$$

where $\mathcal{S} = \mathbb{S}(\vec{G}_{\vec{u}})$ is the strain tensor (21) evaluated with the discrete gradients $\vec{G}_{\vec{u}}$. As a result, we find that discrete viscous volume terms are dissipative.

4.4 Boundary terms

Boundary term stability only makes sense when considering all elements in the domain. Thus, we update (121) with (124), (126), and (130), and sum over all elements, creating

$$\frac{d\bar{\mathcal{E}}}{dt} + \text{IBT} + \text{PBT} = - \sum_e \left\langle \mathcal{J} \left(M_0 |\vec{G}_\mu|^2 + 2\eta \mathcal{S} : \mathcal{S} \right), 1 \right\rangle_{E,N}, \quad (131)$$

where $\bar{\mathcal{E}}$ is the total entropy,

$$\bar{\mathcal{E}} = \sum_e \langle \mathcal{J} \mathcal{E}, 1 \rangle_{E,N}, \quad (132)$$

IBT is the contribution from interior faces,

$$\begin{aligned} \text{IBT} &= \text{IBT}_e + \text{IBT}_v + \text{IBT}_{ch} \\ &= - \sum_{\substack{\text{interior} \\ \text{faces}}} \int_{f,N} \left(\left[\mathbf{W}^T \right] \vec{\mathbf{F}}_e^* + \sum_{m=1}^5 \left[\mathbf{W}^T \left(\vec{\Phi}_m W_m \right)^\diamond \right] - \left[\mathbf{W}^T \vec{\Phi}_m W_m \right] \right) \cdot \vec{n}_L dS \\ &\quad + \sum_{\substack{\text{interior} \\ \text{faces}}} \int_{f,N} \left(\left[\mathbf{W}^T \right] \vec{\mathbf{F}}_v^* + \left[\mathbf{W}^T \vec{\mathbf{F}}_v \right] - \mathbf{W}^{*,T} \left[\vec{F}_v \right] \right) \cdot \vec{n}_L dS \\ &\quad + \frac{3}{2} \sigma \varepsilon \sum_{\substack{\text{interior} \\ \text{faces}}} \int_{f,N} \left(\left[C_t \right] \vec{G}_c^* + \left[\vec{G}_c \right] C_t^* - \left[C_t \vec{G}_c \right] \right) \cdot \vec{n}_L dS, \end{aligned} \quad (133)$$

and PBT is the contribution from physical boundary faces,

$$\begin{aligned} \text{PBT} &= \text{PBT}_e + \text{PBT}_v + \text{PBT}_{ch} \\ &= \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} \left(\mathbf{W}^T \vec{\mathbf{F}}_e^* + \sum_{m=1}^5 \mathbf{W}^T \left(\left(\vec{\Phi}_m W_m \right)^\diamond - \vec{\Phi}_m W_m \right) \right) \cdot \vec{n} dS \\ &\quad - \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} \left(\mathbf{W}^T \left(\vec{\mathbf{F}}_v^* - \vec{\mathbf{F}}_v \right) + \mathbf{W}^{*,T} \vec{\mathbf{F}}_v \right) \cdot \vec{n} dS \\ &\quad - \frac{3}{2} \sigma \varepsilon \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} \left(C_t \vec{G}_c^* + C_t^* \vec{G}_c - C_t \vec{G}_c \right) \cdot \vec{n} dS. \end{aligned} \quad (134)$$

In the inviscid fluxes, the entropy flux definition $\vec{F}^\mathcal{E} = \mathbf{W}^T \vec{\mathbf{F}}_e^*$ was taken into account to reduce the number of terms. Moreover, in the interior boundary terms, we used the left element as the reference (hence the \vec{n}_L vector).

We now analyze the contributions of the interior and boundary terms in (131) separately.

4.4.1 Inviscid interior boundary terms: entropy conserving scheme with central fluxes

We address the stability of the inviscid interior boundary terms when using the central fluxes for the conservative numerical (100) and non-conservative diamond fluxes (101). Inserting the central fluxes into the inviscid IBT_e (133) terms,

$$\begin{aligned}
\text{IBT}_e &= - \sum_{\substack{\text{interior} \\ \text{faces}}} \int_{f,N} \left(\llbracket \mathbf{W}^T \rrbracket \vec{\mathbf{F}}_e^* + \sum_{m=1}^5 \left[\llbracket \mathbf{W}^T (\vec{\Phi}_m W_m)^\diamond \rrbracket - \llbracket \mathbf{W}^T \vec{\Phi}_m W_m \rrbracket \right] \right) \cdot \vec{n}_L dS \\
&= - \sum_{\substack{\text{interior} \\ \text{faces}}} \int_{f,N} \left(\llbracket \mathbf{W}^T \rrbracket \{ \{ \vec{\mathbf{F}}_e \} \} + \sum_{m=1}^5 \left[\llbracket \mathbf{W}^T \vec{\Phi}_m \rrbracket \{ \{ W_m \} \} - \llbracket \mathbf{W}^T \vec{\Phi}_m W_m \rrbracket \right] \right) \cdot \vec{n}_L dS \quad (135) \\
&= - \sum_{\substack{\text{interior} \\ \text{faces}}} \int_{f,N} \sum_{m=1}^5 \left(\llbracket W_m \rrbracket \{ \{ \mathbf{e}_m^T \vec{\mathbf{F}}_e \} \} - \llbracket W_m \rrbracket \{ \{ \mathbf{W}^T \vec{\Phi}_m \} \} \right) \cdot \vec{n}_L dS = 0.
\end{aligned}$$

Thus we find that inviscid interior boundaries exactly satisfy $\text{IBT}_e = 0$. In the second equality, we explicitly wrote the first product as the sum of the product of its components, and in the last equality, we used the algebraic relationship satisfied by jump and average operators,

$$\llbracket uv \rrbracket = \{ \{ u \} \} \llbracket v \rrbracket + \llbracket u \rrbracket \{ \{ v \} \} \quad (136)$$

for the second and third products. Additionally, both terms in the last sum of (135) are identical because of the condition (44). We conclude that inviscid interior boundary contribution to the entropy equation is identically zero, i.e. the interface approximation is entropy conserving.

4.4.2 Inviscid interior boundary terms: entropy stable scheme with the exact Riemann solver

The exact Riemann solver was proven to be stable in [9] for the incompressible Navier–Stokes equations. Now we extend the proof to the iNS/CH system, which is done by a consistent choice of the diamond fluxes. The proof rests on the exact Riemann solver star solution satisfying

$$\begin{aligned}
\Delta_e^{\text{ERS}} &= \frac{1}{2} \rho^* \llbracket V_{tot}^2 \rrbracket U_n^* - \rho^* U_n^* (U_n^* \llbracket U_n \rrbracket + V_{t1}^* \llbracket V_{t1} \rrbracket + V_{t2}^* \llbracket V_{t2} \rrbracket) \\
&\quad - P^* \llbracket U_n \rrbracket - \llbracket P \rrbracket U_n^* + \llbracket P U_n \rrbracket \geq 0.
\end{aligned} \quad (137)$$

Eq. (137) implies that the ERS is dissipative for the incompressible NSE. We use the same star region solution for the inviscid numerical fluxes (105) and the choice for the diamond fluxes given in (106). The result is that the argument of the inviscid interior boundary terms is

$$\begin{aligned}
\Delta_e^{\text{IBT}} &= \left(- \llbracket \mathbf{W}^T \rrbracket \vec{\mathbf{F}}_e^* - \sum_{m=1}^5 \left[\llbracket \mathbf{W}^T (\vec{\Phi}_m W_m)^\diamond \rrbracket + \llbracket \mathbf{W}^T \vec{\Phi}_m W_m \rrbracket \right] \right) \cdot \vec{n}_L \\
&= - \llbracket \mathbf{W}^T \rrbracket \underline{\mathbf{T}}^T \mathbf{F}_e^* - \llbracket \mathbf{W}^T \underline{\mathbf{T}}^T \mathbf{F}_{\Phi W}^{\text{ERS}} \rrbracket + \llbracket \mathbf{W}^T \underline{\mathbf{T}}^T \mathbf{F}_{\Phi W} \rrbracket \\
&= - \llbracket \mathbf{W}_n^T \rrbracket \mathbf{F}_e^* - \llbracket \mathbf{W}_n^T (\mathbf{F}_{\Phi W}^{\text{ERS}} - \mathbf{F}_{\Phi W}) \rrbracket.
\end{aligned} \quad (138)$$

We write the first term in the last line of (138) as

$$- \left[\mathbf{W}_n^T \right] \mathbf{F}_e^* = - \llbracket \mu \rrbracket \{ \{ CU_n \} \} - \frac{1}{2} \rho^* U_n^* (U_n^* \llbracket U_n \rrbracket + V_{t1}^* \llbracket V_{t1} \rrbracket + V_{t2}^* \llbracket V_{t2} \rrbracket) - P^* \llbracket U_n \rrbracket, \quad (139)$$

and the second term as

$$\begin{aligned} - \left[\mathbf{W}_n^T (\mathbf{F}_{\Phi W}^{\text{ERS}} - \mathbf{F}_{\Phi W}) \right] &= - \frac{1}{2} \rho^* U_n^{*,2} \llbracket U_n \rrbracket + \frac{1}{2} \rho^* U_n^* \llbracket U_n^2 \rrbracket - \{ \{ \mu \} \} \llbracket CU_n \rrbracket + \llbracket CU_n \mu \rrbracket \\ &\quad - \frac{1}{2} \rho^* U_n^* V_{t1}^* \llbracket V_{t1} \rrbracket + \frac{1}{2} \rho^* U_n^* \llbracket V_{t1}^2 \rrbracket \\ &\quad - \frac{1}{2} \rho^* U_n^* V_{t2}^* \llbracket V_{t2} \rrbracket + \frac{1}{2} \rho^* U_n^* \llbracket V_{t2}^2 \rrbracket \\ &\quad - U_n^* \llbracket P \rrbracket + \llbracket PU_n \rrbracket \\ &= \frac{1}{2} \rho^* U_n^* \llbracket V_{tot}^2 \rrbracket + \llbracket \mu \rrbracket \{ \{ CU_n \} \} \\ &\quad - \frac{1}{2} \rho^* U_n^* (U_n^* \llbracket U_n \rrbracket + V_{t1}^* \llbracket V_{t1} \rrbracket + V_{t2}^* \llbracket V_{t2} \rrbracket) \\ &\quad - U_n^* \llbracket P \rrbracket + \llbracket PU_n \rrbracket, \end{aligned} \quad (140)$$

which when inserted into (138) yields,

$$\begin{aligned} \Delta_e^{\text{IBT}} &= \frac{1}{2} \rho^* U_n^* \llbracket V_{tot} \rrbracket^2 - \rho^* U_n^* (U_n^* \llbracket U_n \rrbracket + V_{t1}^* \llbracket V_{t1} \rrbracket + V_{t2}^* \llbracket V_{t2} \rrbracket) \\ &\quad - P^* \llbracket U_n \rrbracket - U_n^* \llbracket P \rrbracket + \llbracket PU_n \rrbracket = \Delta_e^{\text{ERS}} \geq 0. \end{aligned} \quad (141)$$

The non–negativity of Δ_e^{IBT} confirms that the incompressible NSE exact Riemann solver (105), with the associated diamond flux choice (106) introduced in this work produces an entropy stable interface approximation for the iNS/CH system.

4.4.3 Viscous and Cahn–Hilliard interior terms: Bassi–Rebay 1 method

The use of the standard BR1 method identically cancels the boundary integrals (see [16] for the compressible NSE, [9] for the incompressible NSE, [37] for the MHD equations, and [8] for the Cahn–Hilliard equation). Then, we prove that the extra interface stabilization terms are dissipative. In this work, we obtain the same integral approximation as in [9] for the incompressible NSE, and as in [8] for the Cahn–Hilliard part,

$$\begin{aligned} \text{IBT}_v &= \sum_{\substack{\text{interior} \\ \text{faces}}} \int_{f,N} \left(\left[\mathbf{W}^T \right] \vec{\mathbf{F}}_v^* - \left[\mathbf{W}^T \vec{\mathbf{F}}_v \right] + \mathbf{W}^{*,T} \left[\vec{\mathbf{F}}_v \right] \right) \cdot \vec{n}_L dS \\ &= \sum_{\substack{\text{interior} \\ \text{faces}}} \int_{f,N} \left(\left[\mathbf{W}^T \right] \{ \{ \vec{\mathbf{F}}_v \} \} + \beta \llbracket \mu \rrbracket^2 \vec{n}_L - \left[\mathbf{W}^T \vec{\mathbf{F}}_v \right] + \{ \{ \mathbf{W} \} \}^T \left[\vec{\mathbf{F}}_v \right] \right) \cdot \vec{n}_L dS \\ &= \sum_{\substack{\text{interior} \\ \text{faces}}} \int_{f,N} \beta \llbracket \mu \rrbracket^2 dS \geq 0, \end{aligned} \quad (142)$$

whereas for the Cahn–Hilliard terms,

$$\begin{aligned}
\text{IBT}_{ch} &= \frac{3}{2}\sigma\varepsilon \sum_{\substack{\text{interior} \\ \text{faces}}} \int_{f,N} \left(\llbracket C_t \rrbracket \vec{G}_c^* + \llbracket \vec{G}_c \rrbracket C_t^* - \llbracket C_t \vec{G}_c \rrbracket \right) \cdot \vec{n}_L dS \\
&= \frac{3}{2}\sigma\varepsilon \sum_{\substack{\text{interior} \\ \text{faces}}} \int_{f,N} \left(\llbracket C_t \rrbracket \{\{\vec{G}_c\}\} + \llbracket \vec{G}_c \rrbracket \{\{C_t\}\} + \beta \llbracket C \rrbracket \llbracket C_t \rrbracket \vec{n}_L - \llbracket C_t \vec{G}_c \rrbracket \right) \cdot \vec{n}_L dS \\
&= \frac{3}{4}\sigma\varepsilon \frac{d}{dt} \sum_{\substack{\text{interior} \\ \text{faces}}} \int_{f,N} \beta \llbracket C \rrbracket^2 dS,
\end{aligned} \tag{143}$$

which says that viscous and Cahn–Hilliard interior boundary contributions do not contribute to the entropy balance if $\beta = 0$, since the integrals are identically zero. This means that interface stabilization is not a requirement for stability. When $\beta > 0$, the integrals in IBT_v are positive (i.e. dissipative), and the integral in IBT_{ch} , which is not positive per se because of the time derivative, will be added to the entropy, since the integral argument is positive.

4.4.4 Interior boundary terms: Summary

The contribution to the interior boundary terms from viscous and Cahn–Hilliard terms are identically zero when using the BR1 method without interface stabilization ($\beta = 0$), and dissipative otherwise ($\beta > 0$). As for the inviscid terms, interior boundary terms are conservative with central fluxes ($\text{IBT}_e = 0$), and dissipative with the exact Riemann solver ($\text{IBT}_e \geq 0$).

4.4.5 Physical boundary terms: free- and no-slip walls

We now address the stability of physical wall boundary terms (134). For inviscid fluxes PBT_e , we use the entropy conserving approach with central fluxes (110),

$$\begin{aligned}
\text{PBT}_e &= \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} \left(\mathbf{W}^T \vec{\mathbf{F}}_e^* + \sum_{m=1}^5 \mathbf{W}^T \left(\left(\vec{\Phi}_m W_m \right)^\diamond - \vec{\Phi}_m W_m \right) \right) \cdot \vec{n} dS \\
&= \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} \left(U_n \left(\frac{1}{2} \rho U_n^2 + P \right) + U_n \left(\mu C - \frac{1}{2} \rho U_n - \mu C \right) - P U_n \right) dS = 0,
\end{aligned} \tag{144}$$

or the entropy stable counterpart with the exact Riemann solver (112),

$$\begin{aligned}
\text{PBT}_e &= \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} \left(\mathbf{W}^T \vec{\mathbf{F}}_e^* + \sum_{m=1}^5 \mathbf{W}^T \left(\left(\vec{\Phi}_m W_m \right)^\diamond - \vec{\Phi}_m W_m \right) \right) \cdot \vec{n} dS \\
&= \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} \left(U_n \left(\frac{1}{2} \rho \lambda^+ U_n + P \right) - P U_n \right) dS = \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} \left(\frac{1}{2} \rho \lambda^+ U_n^2 \right) dS \geq 0.
\end{aligned} \tag{145}$$

Either way, the inviscid physical boundary contribution is stable. Furthermore, for the viscous terms, the free–slip wall boundary condition (113),

$$\begin{aligned} \text{PBT}_v &= - \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} \left(\mathbf{W}^T \left(\vec{\mathbf{F}}_v^* - \vec{\mathbf{F}}_v \right) + \mathbf{W}^{*,T} \vec{\mathbf{F}}_v \right) \cdot \vec{n} dS \\ &\quad - \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} \left(\mathbf{W}^T \left(0 - \vec{\mathbf{F}}_v \right) + \mathbf{W}^T \vec{\mathbf{F}}_v \right) \cdot \vec{n} dS = 0, \end{aligned} \quad (146)$$

and for the no–slip wall boundary condition (114),

$$\begin{aligned} \text{PBT}_v &= - \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} \left(\mathbf{W}^T \left(\vec{\mathbf{F}}_v^* - \vec{\mathbf{F}}_v \right) + \mathbf{W}^{*,T} \vec{\mathbf{F}}_v \right) \cdot \vec{n} dS \\ &\quad - \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} \left(-\mu \vec{G}_\mu + \mu \vec{G}_\mu + \vec{U} \mathcal{S} - \vec{U} \mathcal{S} \right) \cdot \vec{n} dS = 0, \end{aligned} \quad (147)$$

are dissipative. Lastly, the Cahn–Hilliard physical boundary condition for both free– and no–slip walls (113)–(114) gives,

$$\begin{aligned} \text{PBT}_{ch} &= - \frac{3}{2} \sigma \varepsilon \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} \left(C_t \vec{G}_c^* + C_t^* \vec{G}_c - C_t \vec{G}_c \right) \cdot \vec{n} dS \\ &= - \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} \left(-C_t f'_w(C) + \frac{3}{2} \sigma \varepsilon \left(C_t \vec{G}_c - C_t \vec{G}_c \right) \right) \cdot \vec{n} dS \\ &= \frac{d}{dt} \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} f_w(C) dS, \end{aligned} \quad (148)$$

which represents the discrete version of the surface free–energy time derivative (60).

4.5 Semi–discrete stability: Summary

We constructed a DG approximation of the iNS/CH system (93) whose analysis mimics the continuous stability analysis of Sec. 2.1 semi–discretely. As a result, we have shown that the discrete entropy satisfies a balance law,

$$\frac{d}{dt} \left(\bar{\mathcal{E}}^\beta + \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} f_w(C) dS \right) \leq - \sum_e \left\langle \mathcal{J} \left(M_0 |\vec{G}_\mu|^2 + 2\eta \mathcal{S} : \mathcal{S} \right), 1 \right\rangle_{E,N}, \quad (149)$$

for both free– and no–slip wall boundary conditions. The entropy has been extended to consider the interface stabilization in the Cahn–Hilliard equation,

$$\bar{\mathcal{E}}^\beta = \sum_e \langle \mathcal{J} \mathcal{E}, 1 \rangle_{E,N} + \frac{3}{4} \sigma \varepsilon \sum_{\substack{\text{interior} \\ \text{faces}}} \int_{f,N} \beta [C]^2 dS \geq 0. \quad (150)$$

Note that if $\beta = 0$ we recover $\bar{\mathcal{E}}^{\beta=0} = \bar{\mathcal{E}} = \sum_e \langle \mathcal{J}\mathcal{E}, 1 \rangle_{E,N}$, and that a continuous solution (i.e. $\llbracket C \rrbracket = 0$) also recovers the original entropy. Otherwise, the extended entropy $\bar{\mathcal{E}}^\beta$ is also always positive, such that its definition remains valid and the scheme is entropy-stable. The justification to include this term in the entropy, is that the free-energy measures the fluid interfaces using the solution gradients $\frac{3}{4}\sigma\varepsilon|\vec{G}_c|^2$. However, if the discrete solution is allowed to be discontinuous at the inter-elements interface, it enables the creation of a fluid interface, which is not reflected in the original free-energy. The addition of this penalization, allows to account for inter-element discontinuities as part of the fluid interface, so that their associated energy is also weighted in the entropy. In practice, we have found that if the flow is under-resolved and $\beta = 0$, the solution minimizes the entropy allowing sharp discontinuities at the inter-elements interface. Since that effect has not been taken into account in the entropy with $\beta = 0$, the flow wrongly enables the discontinuity mechanism to minimize the entropy. This adverse effect (from the point of view of accuracy) is mitigated when $\beta > 0$ [18, 8]. Since now the inter-elements discontinuities belong to the entropy, their size can be kept bounded and controlled. In any case, the use of interface stabilization does not compromise the stability of the scheme.

Finally, the form of (149) depends on the choice of the inviscid numerical and diamond fluxes, and the Cahn–Hilliard interface stabilization: an equality for central fluxes and $\beta = 0$, and an inequality for the exact Riemann solver and $\beta > 0$. For either choice, the discrete entropy plus the surface free energy remains bounded by the initial condition.

5 Numerical experiments

We now explore the capabilities of the new DGSEM with numerical simulations. First, we validate the implementation with a convergence study on a manufactured solution. Second, we explore the robustness of the scheme by integrating in time from a random initial condition, where we compare an entropy-stable scheme with a Gauss point formulation that is not provably stable. Third, we solve a static bubble as a benchmark for steady-state accuracy, and a transient rising bubble. Finally, we present solutions for a three-dimensional pipe flow in the annular regime.

5.1 Convergence study

We first assess the spatial and temporal accuracy of the implementation of the approximation. We test with the same two-dimensional manufactured solution and configuration used in other Navier–Stokes/Cahn–Hilliard works, specifically [24],

$$\begin{aligned} c_m(x, y; t) &= \frac{1}{2} (1 + \cos(\pi x) \cos(\pi y) \sin(t)), \\ u_m(x, y; t) &= 2 \sin(\pi x) \cos(\pi z) \sin(t), \\ v_m(x, y; t) &= -2 \cos(\pi x) \sin(\pi y) \sin(t), \\ p_m(x, y; t) &= 2 \sin(\pi x) \sin(\pi z) \cos(t), \end{aligned} \tag{151}$$

on the domain $(x, y) \in [-1, 1]^2$. The final time is $t_F = 0.1$, and all physical parameters are presented in Table 1.

Table 1 List of the parameter values used with the manufactured solution (151)

ρ_1	ρ_2 (kg/m ³)	η_1	η_2 (Pa·s)	ε (m)	t_{CH} (s)	c_0^2 (m/s ²) ²	σ (N/m)
1.0	2.0	1.0E-3	1.0E-3	1/√2	1.0E3	1.0E3	6.236E-3

We measure L^2 errors using the discrete inner product,

$$\text{error} = \|\phi - \phi_m\|_{\mathcal{J},N} = \sqrt{\sum_e \langle \mathcal{J}(\Phi - \Phi_m), (\Phi - \Phi_m) \rangle_{E,N}}. \quad (152)$$

Regarding the configuration of the scheme, we use the exact Riemann solver, and we vary the polynomial order N , mesh size (we use an equally spaced Cartesian mesh M^2), and time-step size Δt of the BDF2 scheme ($J = 2$) to evaluate the convergence rates as the resolution is increased either by increasing the polynomial order or decreasing the mesh size.

We first use a Cartesian 4^2 element mesh and vary the polynomial order from $N = 3$ to 10. The L^2 errors (152) are presented in Fig. 1, using $\Delta t = 10^{-4}$ in Fig. 1(a) and $\Delta t = 10^{-5}$ in Fig. 1(b).

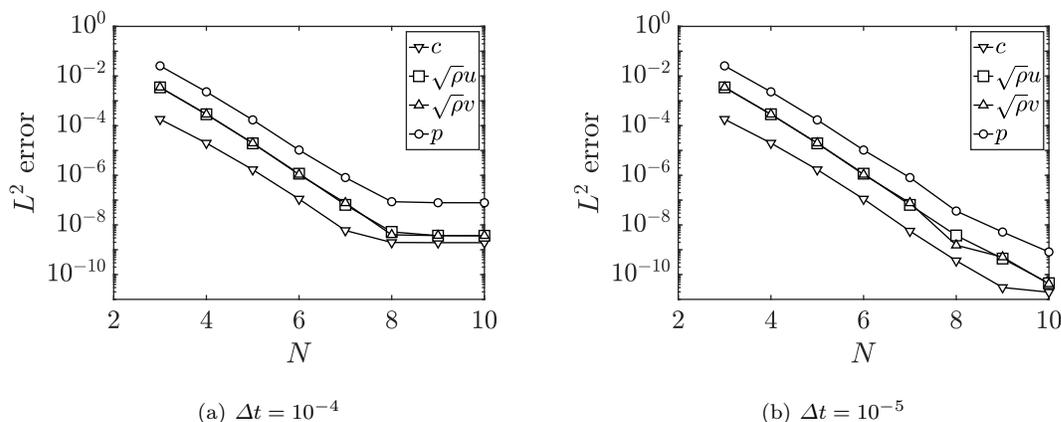


Fig. 1 Exponential convergence of the manufactured solution (151) as the polynomial order increases. We represent the L^2 errors in concentration, x - and y -momentum, and pressure. The polynomial order ranges from 3 to 10, and we integrate in time until $t_F = 0.1$ with two time step sizes: $\Delta t = 10^{-4}$ and 10^{-5} . All physical parameters are given in Table 1

Figs. 1(a) and 1(b) show typical and expected error behavior: an approximately linear semi-log convergence for lower polynomial orders implying exponential convergence of the error (i.e. under-resolved solution in space), and error stagnation for higher orders (under-resolved solution in time). The stagnation error threshold can be controlled with Δt , as shown in Fig. 1(b).

For the next test configuration, we use polynomial orders $N = 2, 3, 4$ and 5 , and meshes with $4^2, 6^2, 8^2, 12^2$ and 16^2 elements for an h -type convergence study. The time-step is $\Delta t = 5 \cdot 10^{-5}$. We solve the manufactured solution problem (151), and summarize the errors in Table 2. An estimate of the scheme's order of convergence is provided there. We note that for the momentum components, the

convergence orders are as expected, as they are close to $N + 1$. For the pressure, the order of convergence is somewhat smaller (closer to N), which was also experienced in [25, 9]. The convergence is also not optimal for the concentration; for even polynomials, the convergence can surpass the theoretical $N + 1$ value, while for odd polynomials, the convergence rate is roughly N in most of the cases. This effect was also noticed in [8], whose nature remains unclear.

Table 2 Manufactured solution (151) convergence analysis: we use 4^3 , 8^3 , and 16^3 meshes, and $N = 2, 3, 4$ and 5 . The final time is $t_F = 0.1$, and we use the IMEX BDF2 scheme with $\Delta t = 5 \cdot 10^{-5}$

	Mesh	c error	order	$\sqrt{\rho u}$ error	order	$\sqrt{\rho v}$ error	order	p error	order
N=2	4^2	1.35E-03	–	3.39E-02	–	3.39E-02	–	2.22E-01	–
	6^2	8.27E-04	1.21	1.24E-02	2.47	1.24E-02	2.47	9.02E-02	2.22
	8^2	4.40E-04	2.19	5.96E-03	2.56	5.95E-03	2.56	4.68E-02	2.28
	12^2	1.30E-04	3.01	2.04E-03	2.65	2.03E-03	2.65	1.78E-02	2.39
	16^2	5.35E-05	3.08	9.23E-04	2.75	9.23E-04	2.75	8.67E-03	2.50
N=3	4^2	1.81E-04	–	3.49E-03	–	3.50E-03	–	2.57E-02	–
	6^2	5.71E-05	2.85	8.28E-04	3.55	8.29E-04	3.55	6.81E-03	3.27
	8^2	2.76E-05	2.53	2.90E-04	3.65	2.91E-04	3.65	2.57E-03	3.39
	12^2	9.53E-06	2.62	6.34E-05	3.75	6.37E-05	3.74	6.17E-04	3.52
	16^2	4.36E-06	2.72	2.10E-05	3.83	2.12E-05	3.83	2.17E-04	3.64
N=4	4^2	1.99E-05	–	2.89E-04	–	2.87E-04	–	2.31E-03	–
	6^2	5.27E-06	3.28	4.40E-05	4.64	4.40E-05	4.62	3.94E-04	4.36
	8^2	7.50E-07	6.78	1.12E-05	4.75	1.12E-05	4.75	1.09E-04	4.47
	12^2	8.46E-08	5.38	1.55E-06	4.89	1.55E-06	4.89	1.70E-05	4.59
	16^2	1.94E-08	5.12	3.61E-07	5.05	3.61E-07	5.05	4.42E-06	4.68
N=5	4^2	1.69E-06	–	1.94E-05	–	1.98E-05	–	1.71E-04	–
	6^2	1.91E-07	5.38	1.96E-06	5.65	1.97E-06	5.70	1.87E-05	5.46
	8^2	4.67E-08	4.89	3.76E-07	5.74	3.77E-07	5.74	3.78E-06	5.57
	12^2	6.43E-09	4.89	3.64E-08	5.76	3.65E-08	5.76	3.79E-07	5.67
	16^2	1.63E-09	4.77	7.08E-09	5.69	7.11E-09	5.69	7.47E-08	5.64

5.2 Assessing robustness: random initial condition

We also assess the robustness of the scheme by introducing a random initial condition. Because the scheme developed here is entropy stable even for severely under-resolved solutions, the numerical implementation should not crash, as long as the time-step is small enough to stay within the time integrator stability region and the positivity condition is satisfied.

As a point of reference, and only in this section, we also consider the DG scheme using Gauss points. The standard DG method with Gauss points does not satisfy the SAT property [35] and the proofs developed for semi-discrete stability do not carry over. As a result, the standard scheme is not provably stable. To make the comparison even more stark, we compare robustness of the Gauss scheme with upwind flux dissipation to the new Gauss-Lobatto method with only central fluxes.

We solve the problem on the domain $\Omega = [0, 1]^3$, and on a three-dimensional 4^3 element Cartesian mesh. We initialize the solution using uniform random numbers in $[0, 1]$ for concentration, and in $[-1, 1]$ for velocities and pressure. We use a high density ratio ($\rho_1/\rho_2 = 1000$), and Reynolds number $Re = 10^6$

so that physical viscosity will have only a weak stabilizing effect. The rest of the parameters are given in Table 3.

Table 3 Physical parameters for the random initial condition test problem

ρ_1	ρ_2 (kg/m ³)	η_1	η_2 (Pa·s)	ε (m)	t_{CH} (s)	c_0^2 (m/s ²) ²	σ (N/m)
1000.0	1.0	10^{-3}	10^{-4}	0.75	10.0	1.0E2	1.0

With Gauss–Lobatto points, the discrete entropy balance (149) shows that the free–energy time derivative \mathcal{E}_t should be always negative as a result of viscous and chemical potential dissipation, for either central fluxes or the ERS. Moreover, the entropy remainder, defined as

$$\mathcal{R}(\mathcal{E}) = \frac{d}{dt} \left(\sum_e \langle \mathcal{J}\mathcal{E}, 1 \rangle_{E,N} + \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} f_w(C) dS \right) + \left\langle \mathcal{J} \left(M_0 |\vec{G}_\mu|^2 + 2\eta \mathcal{S} : \mathcal{S} \right), 1 \right\rangle_{E,N} \leq 0, \quad (153)$$

should be zero at each time step for central fluxes, and negative for the ERS. None of the statements regarding \mathcal{E}_t or $\mathcal{R}(\mathcal{E})$ being negative are guaranteed to hold for Gauss points by theory.

The entropy time derivative evolution is presented in Fig. 2, and the entropy remainder (153) in Fig. 3. Both are presented for three polynomial orders ($N = 2, 3, 4$), and for the three configurations

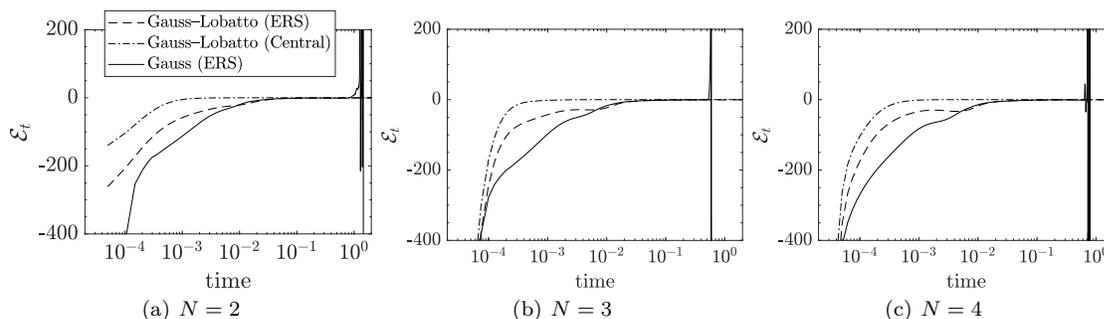


Fig. 2 Random chosen initial condition: evolution of the entropy time derivative \mathcal{E}_t for three polynomial orders $N = 2, 3, 4$. We consider three schemes: Gauss–Lobatto points with the ERS and central fluxes, and Gauss points with the ERS. For all the polynomial orders, using Gauss–Lobatto points is an entropy preserving scheme. The dissipation is a result of physical viscosity and chemical potential for central fluxes, to which the dissipation by the exact Riemann solver can be added. On the contrary, the Gauss approximation entropy time derivative is not always negative, which numerically diverges as a result

described: Gauss–Lobatto points with ERS, Gauss–Lobatto points with central fluxes, and Gauss points with ERS. We find that the solutions follow the same pattern for the different polynomial orders. Gauss–Lobatto points are always stable, and the ERS enhances the stability (i.e. the entropy time derivative is always smaller or equal when compared to central fluxes). This is confirmed by looking at the entropy remainder in Fig. 3, which is machine precision for central fluxes, and negative for the ERS. Whereas for Gauss points, the solution is divergent (i.e. crashes) when $t \geq 1.53$. We see that for this case, Gauss

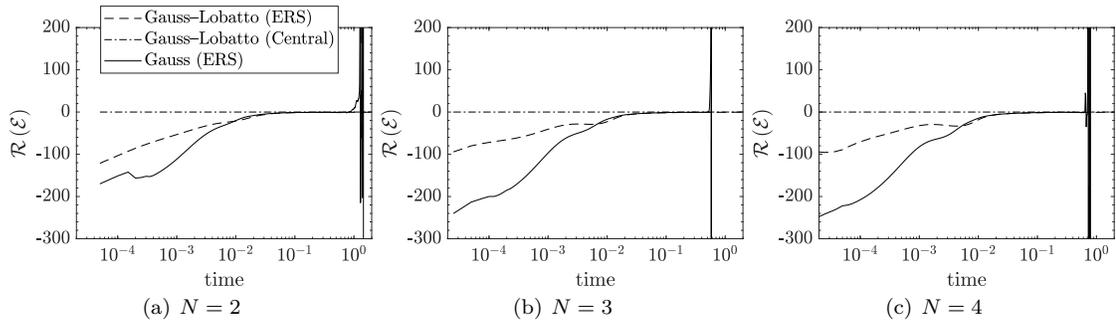


Fig. 3 Random initial condition: evolution of the entropy remainder (153) for three polynomial orders $N = 2, 3, 4$. We consider three schemes: Gauss-Lobatto points with the ERS and central fluxes, and Gauss points with the ERS. The remainder studies the dissipation introduced by the scheme without considering physical viscosity and chemical potential. Thus, the Gauss-Lobatto scheme with the ERS is dissipative (always negative), the Gauss-Lobatto scheme with central fluxes is conserving (always zero), and the scheme with Gauss points is unstable and crashes

points are more dissipative than Gauss-Lobatto points for the first steps, but then the opposite occurs when $t > 10^{-2}$.

With a random initial condition, the Gauss approximation is prone to diverge, but might not. We ran another 100 random initial conditions with different seeds until a final time $t_F = 10.0$, of which 30/100 crashed. For the 70 simulations that did not crash, most of them experienced growth of their entropy at some points in time. We represent in Fig. 4 the number of simulations that crashed before the physical time on the horizontal axis. Most often crashes occur early, i.e. times $t \approx 1.0$, after which the simulation

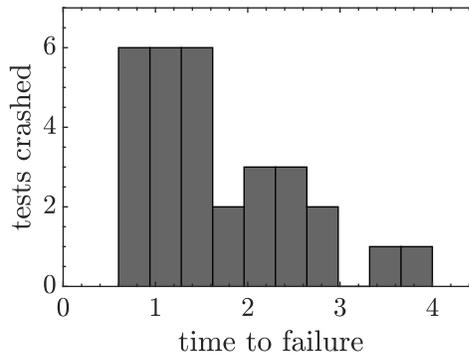


Fig. 4 Random initial condition: we run 100 simulations modifying the random numbers sequence, of which 30 crashed with Gauss points. In this figure, the physical time in which the simulation diverged is represented. Although the other test passed the final time $t_F = 10.0$, their entropy time derivative was found to be positive in several time-steps. None of the 100 simulations crashed with Gauss-Lobatto

is less likely to crash. None of the simulations crashed if they stayed stable for more than 4.0 seconds. In other words, if it can be computed beyond the initial stages, it is more likely to continue.

We repeated the experiment for the entropy stable Gauss–Lobatto variant, to confirm that none of them crashed. There may be advantages to use Gauss points, because of higher accuracy per degree of freedom [42, 43], and indeed the scheme presented herein allows us to use Gauss points in practice. However, as it is not entropy stable, it might crash under certain conditions, perhaps severely under-resolved simulations like high Reynolds turbulent flows, etc.

5.3 Static bubble

In this test problem we solve a steady two–dimensional bubble and validate the pressure jump that results from surface tension. In the domain $\Omega = [0, 1] \times [0, 1]$, the initial condition for the concentration is

$$c(x, y; 0) = \frac{1 - \tanh\left(-\frac{r-0.25}{\zeta}\right)}{2}, \quad r = \sqrt{(x-0.5)^2 + (y-0.5)^2}, \quad \zeta = 2.5 \cdot 10^{-3}, \quad (154)$$

which approximates a circle with radius $R_0 = 0.25$. Since we project the initial condition into our polynomial ansatz, the radius obtained, R_e , subtly differs to R_0 . The analytical pressure jump between outside and inside of the bubble Δp_a is given by the Poisson law,

$$\Delta p_a = p_i - p_e = p(0.5, 0.5) - p(1.0, 0.5) = \frac{\sigma}{R_e}. \quad (155)$$

Note that we used the radius R_e , which is estimated from the final solution contour $c(x, y) = 0.5$, to compute the analytical solution. The rest of the parameters, including the surface tension $\sigma = 1$ N/m, are given in Table 4. We apply periodic boundary conditions at the four physical boundaries, and use

Table 4 Parameters of the Static bubble test problem

Grid	$\rho_{1,2}$ (kg/m ³)	$\eta_{1,2}$ (Pa·s)	ε (m)	t_{CH} (s)	c_0^2 (m/s ²) ²	σ (N/m)	Δt (s)
16 ²	1.0	1.0	0.16	7.0	1.0E3	1.0	10 ⁻⁵
32 ²	1.0	1.0	0.08	7.0	1.0E3	1.0	5 · 10 ⁻⁶
64 ²	1.0	1.0	0.05	7.0	1.0E3	1.0	2.5 · 10 ⁻⁶

the first order IMEX scheme ($J = 1$) until the residuals are kept lower than 10^{-7} .

The results are summarized in Table 5, which are computed with polynomial order $N = 2$ and three meshes 16², 32², and 64². For all meshes, we compute the solution radius R_e , the analytical pressure jump Δp_a , which are compared to the numerical solution interior pressure, p_i , exterior pressure p_e , and pressure jump Δp . Here we refer to the static pressure, and not the auxiliary pressure (16). The results show that the pressure jump converges to that given by the Poisson law (155) as we refine the grid.

In the last column of Table 5 we represent the norm of the velocity, which is of the size of the residuals. This implies that no parasitic currents are produced using this formulation, which are common in alternative Navier–Stokes/Cahn–Hilliard formulations [44]. Usually, the size of the parasitic currents is 10^{-3} for this problem, and since they do not disappear as the grid is refined, they can dominate in low speed simulations.

Table 5 Results of the Static bubble test problem

Grid	R_e (m)	Δp_a	p_i	p_e	Δp (Pa)	$ \Delta p - \Delta p_a /\Delta p_a$	$\ \vec{v}\ $
16^2	0.220	4.546	0.033	-4.571	4.604	$1.27 \cdot 10^{-2}$	$4.0 \cdot 10^{-10}$
32^2	0.240	4.175	0.029	-4.120	4.150	$6.09 \cdot 10^{-3}$	$2.4 \cdot 10^{-8}$
64^2	0.244	4.100	0.173	-4.094	4.111	$2.75 \cdot 10^{-3}$	$3.7 \cdot 10^{-9}$

5.4 Rising bubble

The rising bubble problem has been widely used in the multiphase flow community to assess the space-time accuracy and robustness of the methods [45, 23, 7]. The problem follows the trajectory of a bubble submerged in a heavier fluid as it rises. We consider the domain $(x, y) \in [0, 1] \times [0, 2]$, and a circular bubble with center in $(0.5, 0.5)$ and diameter 0.5. The bubble is approximated with the initial condition,

$$c(x, y; 0) = 1 - \frac{1}{2} \left(\tanh \left(-\frac{2(r - 0.25)}{\varepsilon} \right) + 1 \right), \quad r = \sqrt{(x - 0.5)^2 + (y - 0.5)^2}, \quad (156)$$

and the rest of the variables are initialized to zero. We consider a Cartesian mesh with element size $h = 2^{-4}$, and a polynomial order $N = 10$. The physical parameters are taken from [45] and are summarized in Table 6. We consider two tests: the first with moderate density ratio $\rho_1/\rho_2 = 10$, the second with large

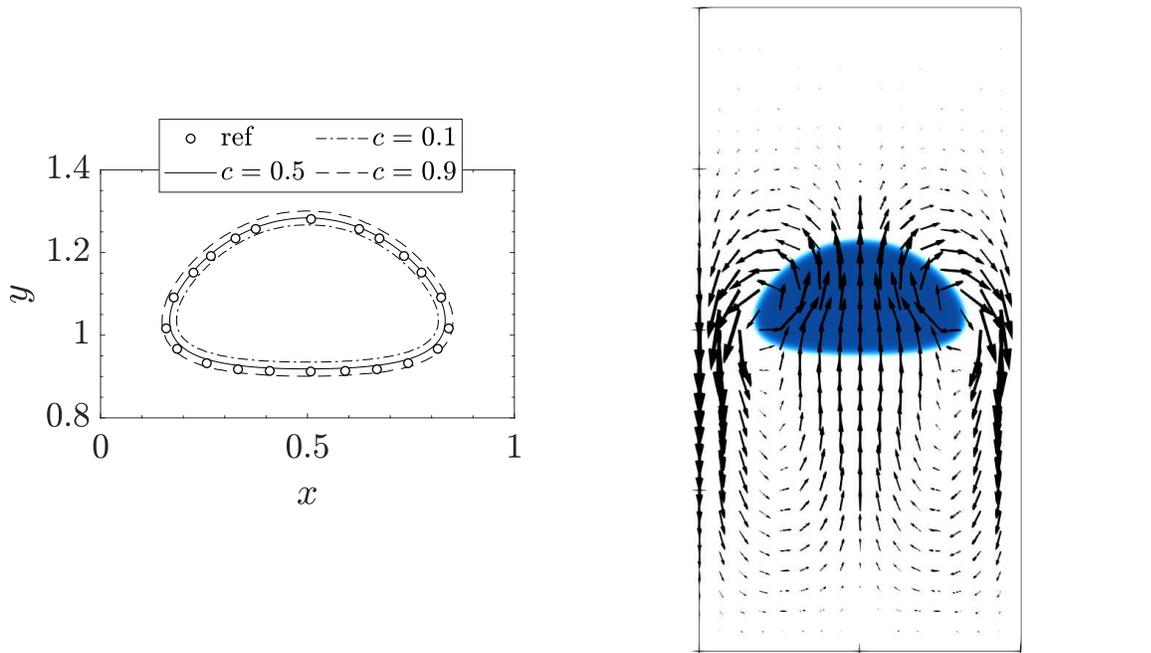
Table 6 Physical parameters of the Rising bubble test problem

Test	ρ_1	ρ_2 (kg/m ³)	η_1	η_2 (Pa·s)	ε (m)	t_{CH} (s)	c_0^2 (m/s ²) ²	σ (N/m)	g (m/s ²)
1	1000.0	100.0	10.0	1.0	0.03	1.0E3	1.0E3	24.5	0.98
2	1000.0	1.0	10.0	0.1	0.04	1.0E4	1.0E3	1.96	0.98

density ratio $\rho_1/\rho_2 = 1000$. The boundary conditions are free-slip walls in $x = 0$ and $x = 1$, and no-slip walls in $y = 0$ and $y = 2$. For the Test 1, the chemical characteristic time is not big enough to use the explicit Runge-Kutta, and we use the second order IMEX BDF method, with time-step $\Delta t = 7.2 \cdot 10^{-6}$. For the Test 2, we use the explicit third order Runge-Kutta, with $\Delta t = 7.2 \cdot 10^{-7}$. Although we use fixed time-stepping, in both cases the CFL number ($\text{CFL} = 16(N + 1)(|u| + a)\Delta t$) is maintained close to 0.2.

Although the model presented in this paper is diffuse interface, we compare it to a sharp interface model. Moreover, in this test case we compare the artificial compressibility method to a pressure-correction method that enforces the incompressible constraint in a transient simulation. Thus, we can assess the validity of the model and its implementation.

The shape of the bubble at the final time $t = 3.0$ is represented in Fig. 5. First, in Fig. 5(a), we represent the final shape of the interface by drawing three contour lines: $c = 0.1$ in dash-dot, $c = 0.5$ as a solid line, and $c = 0.9$ as a dashed line. The sharp interface reference solution is represented with black dots. We find a good agreement in both the position and the shape of the bubble, as the $c = 0.5$ contour follows the shape of the bubble, and the dots are always found inside the $c = 0.1$ and $c = 0.9$ contours. Next, in Fig. 5(b) we represent the concentration contour, with the velocity vectors on top. The solution obtained is also in agreement with other diffuse interface works [7]. In Fig. 6 we represent the evolution of the center of gravity (Fig. 6(a)) and the bubble rise velocity (Fig. 6(b)),



(a) Final shape of the bubble. We represent the interface with three contours: $c = 0.1$, $c = 0.5$, and $c = 1.0$. The dots represent the sharp interface reference [45]

(b) Concentration contour at the final time $t = 3$. The velocity field is also represented, showing the currents that raise the bubble

Fig. 5 Rising bubble Test 1: position of the bubble and flow configuration at the final time $t = 3.0$

$$X_c = \frac{1}{A} \sum_e \langle \mathcal{J}(1-C), X \rangle_{E,N}, \quad V_c = \frac{1}{A} \sum_e \langle \mathcal{J}(1-C), U \rangle_{E,N}, \quad A = \sum_e \langle \mathcal{J}(1-C), 1 \rangle_{E,N}. \quad (157)$$

Both center of gravity position and rise velocity show good agreement with the sharp interface reference. As for the rise velocity in Fig. 6(b), the oscillations are a result of the artificial compressibility pressure waves. One can reduce the amplitude of these oscillations by increasing the artificial sound speed c_0 , but we have only found small differences in the evolution when increasing this parameter above the value used here.

Next, we solve the more challenging rising bubble Test 2 (see Table 6) with a higher density ($\rho_1/\rho_2 = 1000$) and viscosity ($\mu_1/\mu_2 = 100$) ratios. The evolution of the bubble and the flow configuration are represented in Fig. 7 at each 0.5 second. Both the shape and position of the bubble are in agreement with the sharp interface method [45], and other diffuse interface Cahn–Hilliard solvers [7]. Contrary to the rising bubble Test 1, the bubble now leaves behind an elongated skirt, which influences the velocity field. For completeness, we represent the center of gravity position and rise velocity as a function of time in Fig. 8, compared to the sharp interface solution provided in [45]. Compared to the sharp interface reference, there are visible differences. However, it has been noted in [45, 7] that the solution with

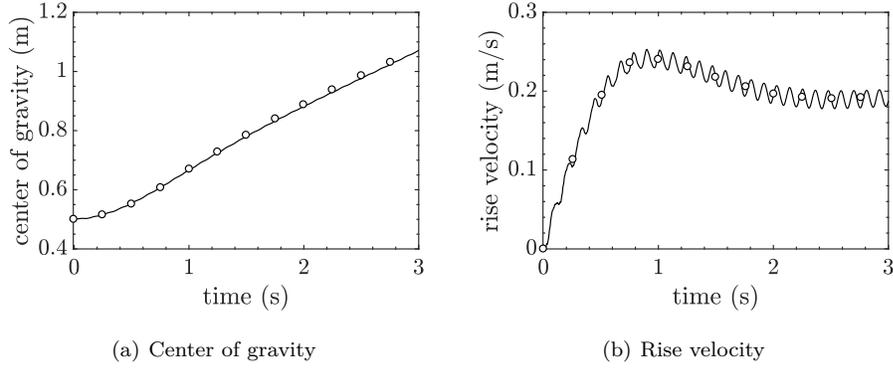


Fig. 6 Rising bubble Test 1: evolution of the bubble center of gravity and rise velocity

different sets of equations might differ for this more challenging test case. Nonetheless, the evolution of the center of gravity and the rise velocity presented here agree with other diffuse interface solvers [7].

5.5 Three-dimensional annular flow simulation

The last numerical experiment we present is a three-dimensional flow in a straight pipe ($L = 10$) with circular section ($D = 1$). Depending on the flow rates of fluid 1 and 2, the flow regime can be of different nature: stratified, slug, dispersed bubble, and annular. The last is the one we compute here, where one of the fluids behaves as a coating around the pipe surface. More details of the flow can be found in [46, 47].

We construct a z -oriented mesh of the pipe with 8200 elements, and use a polynomial order $N = 3$ (for both solutions and physical boundary representation). Note that this is the first time in this work that we use an unstructured curvilinear mesh. The cross section of the mesh is represented in Fig. 10(c). The physical parameters are taken from the literature [46] and given in Table 7.

Table 7 Physical parameters of the annular flow

ρ_1	ρ_2 (kg/m ³)	η_1	η_2 (Pa·s)	ε (m)	t_{CH} (s)	c_0^2 (m/s ²) ²	σ (N/m)	g (m/s ²)	θ_w (deg)
1.0	5.0	$5 \cdot 10^{-3}$	10^{-2}	0.0424	900.0	1.0E3	$2.5 \cdot 10^{-4}$	1.0	45°

For the inflow and outflow boundary conditions, we construct auxiliary ghost states,

$$\mathbf{Q}^{in} = \begin{pmatrix} c_{in} \\ \sqrt{\rho(c_{in})} \bar{u}_{in} \\ p \end{pmatrix}, \quad \mathbf{Q}^{out} = \begin{pmatrix} c \\ \sqrt{\rho} \bar{u} \\ 0 \end{pmatrix}, \quad (158)$$

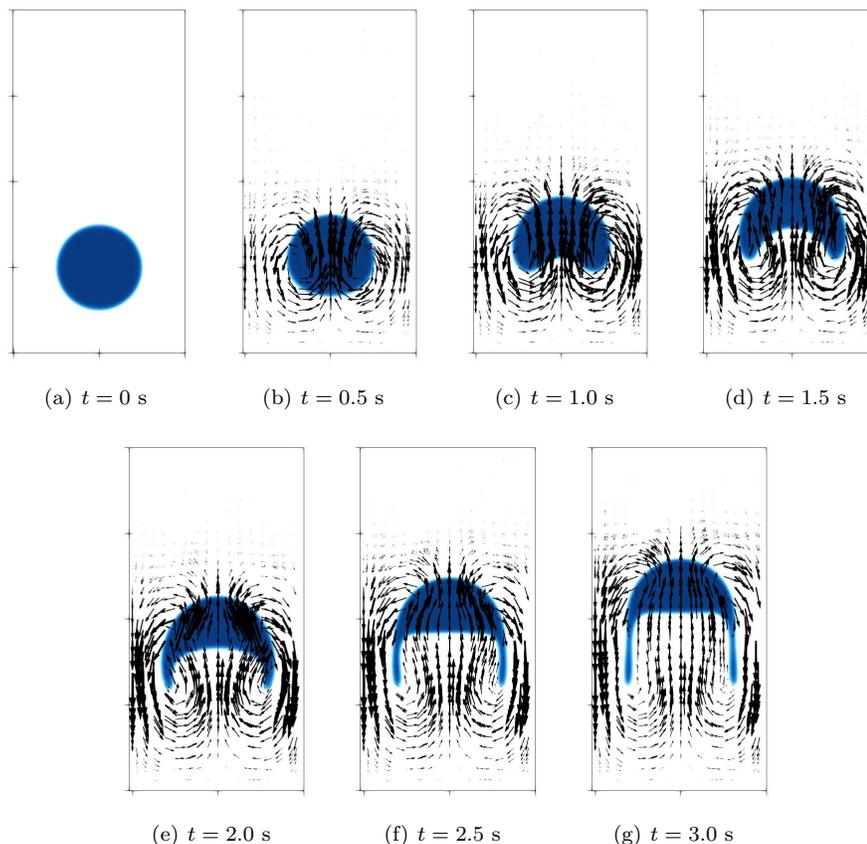


Fig. 7 Rising bubble Test 2: different snapshots showing the position of the bubble and the flow configuration

with

$$\begin{aligned}
 c_{in} &= \frac{1}{2} + \frac{1}{2} \tanh \left(\frac{x - x_0}{\varepsilon} + \frac{1}{10} \sin(10z) + \frac{1}{10} \sin(20y) \right), \\
 u_{in} &= 0, \\
 v_{in} &= 0, \\
 w_{in} &= (V_1 c_{in} + V_2 (1 - c_{in})) \left(1 - 4 \frac{x^2 + y^2}{D^2} \right), \\
 p_{in} &= 0.
 \end{aligned} \tag{159}$$

Furthermore, we use \mathbf{Q}^{in} as the initial condition. The velocities V_1 and V_2 , and interface position x_0 are set so that the superficial velocities, defined as,

$$V_{s1} = \frac{1}{\pi R^2} \int_{\text{inlet}} V_1 c_{in} \left(1 - 4 \frac{x^2 + y^2}{D^2} \right) dS, \quad V_{s2} = \frac{1}{\pi R^2} \int_{\text{inlet}} V_2 (1 - c_{in}) \left(1 - 4 \frac{x^2 + y^2}{D^2} \right) dS, \tag{160}$$

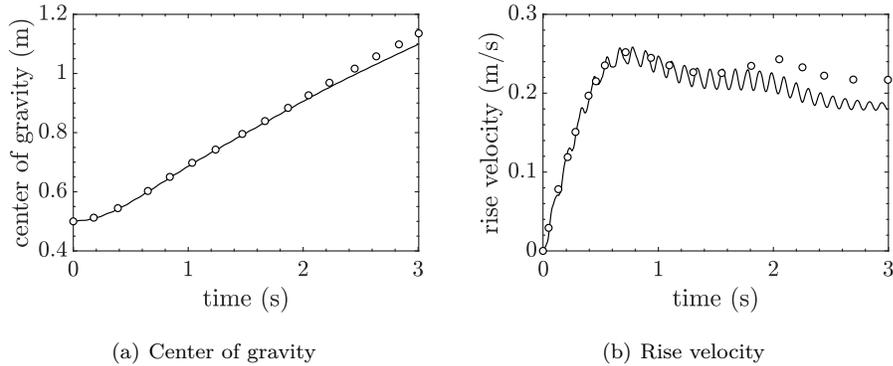


Fig. 8 Rising bubble Test 2: evolution of the bubble center of gravity and rise velocity

are those found in [46]. Precisely, we chose the velocities $V_{s1} = 4.9$ m/s and $V_{s2} = 0.06$ m/s, which produce the annular flow regime. If we place the interface between both fluids at $x_0 = -0.262$ m (which yields a slip velocity $V_1 - V_2 = 10$ m/s), the inlet velocities are $V_1 = 11.06$ m/s and $V_2 = 1.06$ m/s. At the walls, we also impose a contact angle $\theta_w = 45^\circ$. With this configuration, we can use the third order Runge–Kutta scheme as time integrator with $\Delta t = 10^{-5}$. In this simulation, we take into account the effect of gravity, which points in the negative x -direction.

The annular flow configuration is reached through an unstable mode, which makes the transient problem under–resolved until this mode is non–linearly damped and the final annular flow configuration is found. For that reason, we use a non–zero penalty parameter, β , on the Cahn–Hilliard equation. Precisely, we use the definition (108) with $\kappa_\beta = 1$. Otherwise, we have found the scheme to be stable, but the solution obtained is mostly noise (i.e. it is not accurate). The introduction of the penalty parameter enhances the accuracy when the flow is under–resolved [48].

We represent snapshots of the initial stages of the flow in Fig. 9. The fluid 2 is represented ($c \leq 0.5$), and colored by the vertical velocity (along the x -axis, u). We see that in the initial steps there is an unstable mode that grows until it changes the flow configuration. The vertical velocity contour shows that when fluid 2 rises above the interface it is dragged and accelerated by the quicker fluid 1, and the contrary occurs when it is confined below the interface because of the wall. This produces a wavy pattern at $t = 0.3$ s, which eventually wraps the fluid to the wall and produces the annular flow. In the early stages, the flow is highly under–resolved.

We represent a final snapshot for $t = 4.0$ s in Fig. 10, showing the contour $c \leq 0.5$ (i.e. the region occupied by fluid 2), colored by the longitudinal velocity (i.e. along the z -axis, w). Although the interface is flat at the inlet, the flow is unstable and breaks down into the annular flow configuration. The heavy fluid (fluid 2) is confined to the wall by the light fluid, which is introduced 82 times faster at the inlet. In Fig. 10(c) we represent the front view of the pipe, with the mesh detailed on top of it. We see that the flow gets smoother after the numerical and physical dissipation take over the mode breakdown after the under–resolved stages in Fig. 10.

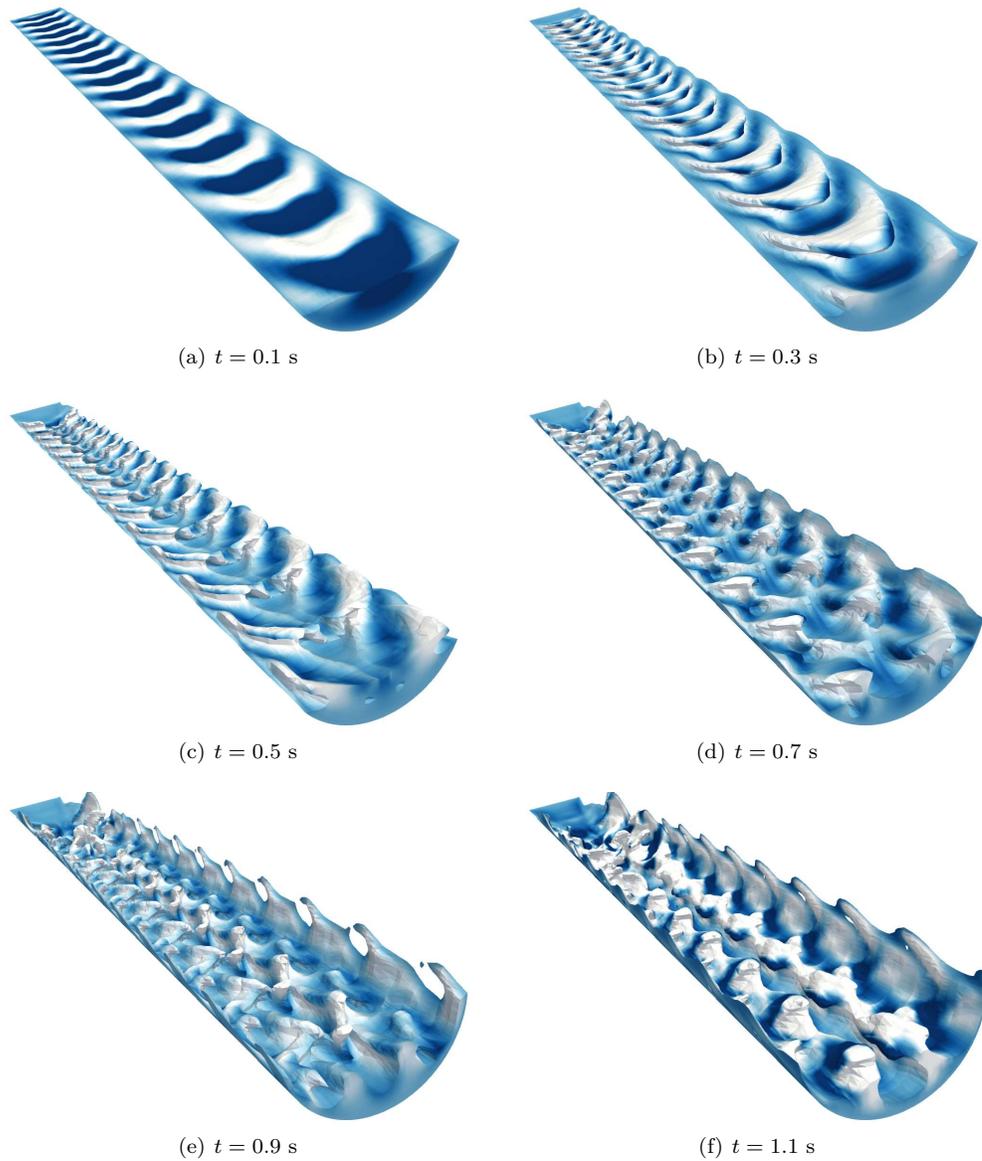


Fig. 9 Evolution of the physical mode that configures the annular flow. We have represented fluid 2 (i.e. $c \leq 0.5$), colored with vertical velocity, u . We see that during early times, there is an unstable physical mode that triggers the annular flow. When fluid 2 rises above the interface, it is dragged and accelerated by the quicker fluid 1. When it is confined below the interface, it is slowed down by the wall. This evolves non-linearly to the final state

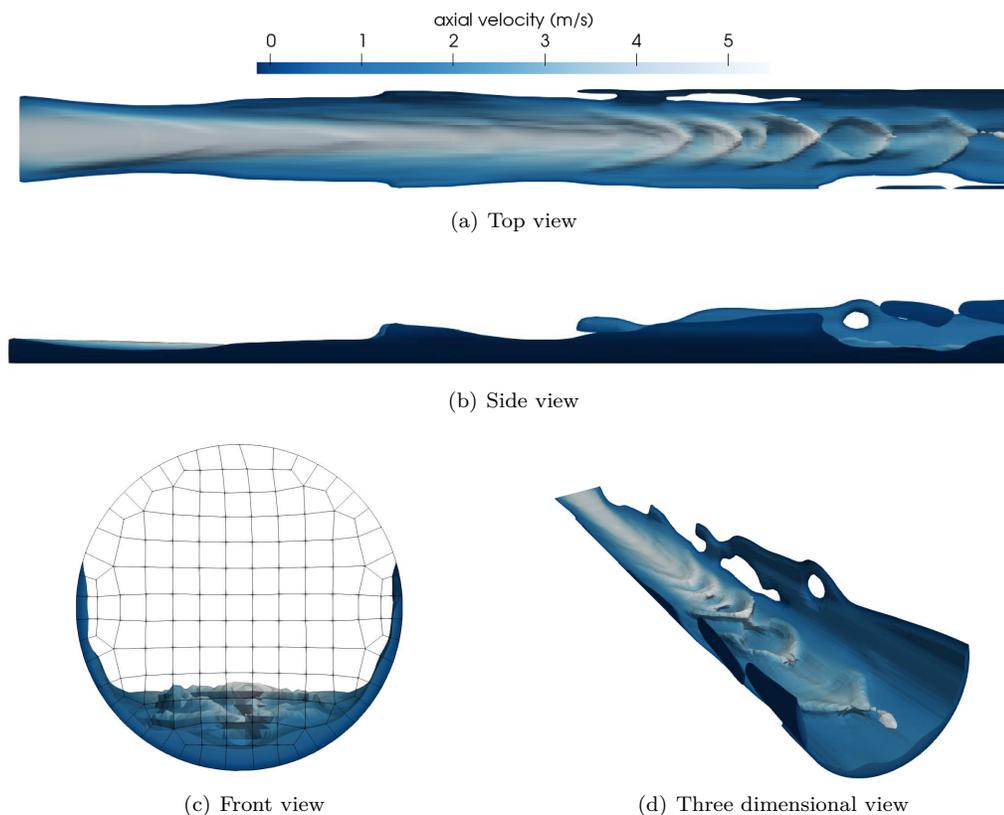


Fig. 10 Annular flow simulation. We present four view angles to the simulation at $t = 4.0$ s. In the front view, the mesh cross section is also represented

6 Summary and conclusions

We have derived a two-phase flow model that combines the Cahn–Hilliard equation (3), a skew-symmetric version of the momentum equation (17), and an artificial compressibility method (18) to get the pressure. Among the many available options, the versions were chosen to satisfy an entropy inequality, (60). The entropy inequality bounds the mathematical entropy inside a finite domain Ω limited with wall boundary conditions by the entropy of the initial condition. As time marches, the mathematical entropy decreases as a result of the physical and the chemical potential dissipation.

We then constructed a DG approximation that satisfies the SBP–SAT property, which allowed us to mimic the continuous entropy analysis discretely. As is usual in a DG approximation, we had choices for the fluxes at inter-element and physical boundaries. We studied two options:

1. Entropy conserving, using central fluxes for the advective fluxes and the BR1 scheme for the diffusion. This choice gives a bound that discretely mimics the continuous entropy bound (60). This scheme is not of practical use, since some amount of numerical dissipation is required to provide accurate

solutions when non-linear terms are present [18, 43, 49, 50]. However, it serves as a baseline model to verify the stability proofs and to obtain a dissipation-free scheme.

2. Entropy stable, with an exact Riemann solver for the advective fluxes and the BR1 scheme for diffusive, with additional interface dissipation for the Cahn–Hilliard equation. This scheme transforms the entropy balance to an entropy inequality in the discrete version of (60), as a result of the numerical dissipation at the inter-element faces. This scheme uses the exact Riemann solver presented in [25] for the incompressible Navier–Stokes equations, with an appropriate choice of the diamond fluxes that arise from the non-conservative terms, and a modification of the discrete entropy to account for solution discontinuities in the concentration as interfacial energy in the numerical solution.

Both the scheme and the stability proofs hold for three-dimensional unstructured meshes with hexahedral curvilinear elements.

We selected two choices to march the scheme in time: an explicit third-order Runge–Kutta method, and an implicit–explicit BDF with first or second order of accuracy. The former is used when the mobility is small enough to not severely restrict the time-step size, and the latter otherwise.

We test the scheme, addressing its accuracy with a manufactured solution convergence analysis, and its robustness by initializing the flow with random initial conditions. We showed that the scheme converges spectrally as expected, and that the scheme is robust in the sense that none of one hundred simulations crashed from random initial conditions, a high density ratio ($\rho_2/\rho_1 = 1000$), and a high Reynolds number ($\text{Re} = 10^6$). We compared the entropy-stable scheme with the more accurate Gauss counterpart (which is not provably entropy stable) to find that the latter crashes in 30% of the simulations. We also solved commonly used static and rising bubble test problems to assess the steady state and transient accuracy of the solver. Finally we challenged the method by solving a three-dimensional pipe flow in the annular regime.

Acknowledgements The authors would like to thank Dr. Gustaaf Jacobs of the San Diego State University for his hospitality. This work was supported by a grant from the Simons Foundation (#426393, David Kopriva). This work has been partially supported by Ministerio de Economía y Competitividad under the research grant (EUI2017-88294, Gonzalo Rubio). This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 785549 (FireExtinction: H2020-CS2-CFP06-2017-01). The authors acknowledge the computer resources and technical assistance provided by the Centro de Supercomputación y Visualización de Madrid (CeSViMa).

A Stability analysis of the alternative artificial compressibility model

We address the stability of the second artificial compressibility model (19). Since changes with respect to the original model (18) only affect to time-derivative terms, the analysis performed for spatial terms hold in this model, and we need only include the new time-derivative terms.

A.1 Continuous entropy analysis

The entropy analysis performed in Sec. 2.1 can be extended to the second artificial compressibility method (19). If we maintain the same entropy variables, all of the steps performed for space operators hold, and only the temporal

terms contraction needs to be recomputed. Doing so,

$$\begin{aligned} \mu c_t + \sqrt{\rho} \vec{u} \cdot (\sqrt{\rho} \vec{u})_t - \frac{p \vec{\nabla}^2 p_t}{\rho_0 c_0^2} &= \mathcal{F}_t + \mathcal{K}_t + \frac{\vec{\nabla} p \cdot \vec{\nabla} p_t}{\rho_0 c_0^2} + \vec{\nabla} \cdot \left(\vec{f}_t^{\mathcal{E}} - \frac{p \vec{\nabla} p_t}{\rho_0 c_0^2} \right) \\ &= \mathcal{F}_t + \mathcal{K}_t + \frac{|\vec{\nabla} p|_t^2}{2 \rho_0 c_0^2} + \vec{\nabla} \cdot \left(\vec{f}_t^{\mathcal{E}} - \frac{p \vec{\nabla} p_t}{\rho_0 c_0^2} \right) = \mathcal{E}_{2,t} + \vec{\nabla} \cdot \vec{f}_t^{\mathcal{E}_2}, \end{aligned} \quad (161)$$

with

$$\mathcal{E}_2 = \mathcal{F}_t + \mathcal{K}_t + \frac{|\vec{\nabla} p|_t^2}{2 \rho_0 c_0^2}, \quad \vec{f}_t^{\mathcal{E}_2} = \frac{3}{2} \sigma \varepsilon c_t \vec{\nabla} c - \frac{p \vec{\nabla} p_t}{\rho_0 c_0^2}. \quad (162)$$

The second artificial compressibility model is complemented with homogeneous Neumann boundary conditions for the pressure time derivative [6],

$$\vec{\nabla} p_t \cdot \vec{n} \Big|_{\partial \Omega} = 0, \quad (163)$$

which makes the pressure term in $\vec{f}_t^{\mathcal{E}_2}$ vanish at the boundaries. Hence, the entropy equation with physical boundary conditions for the second artificial compressibility method is identical to the original model (60)

$$\frac{d}{dt} \left(\bar{\mathcal{E}}_2 + \int_{\partial \Omega} f_w(c) dS \right) = - \int_{\Omega} (M_0 |\vec{\nabla} \mu|^2 + 2\eta \mathbf{S} : \mathbf{S}) d\vec{x} \leq 0, \quad \bar{\mathcal{E}}_2 = \int_{\Omega} \mathcal{E}_2 d\vec{x}. \quad (164)$$

A.2 Discretization

The discretization of the original artificial compressibility method (93) translates to the second method, where only the pressure time derivative terms need to be updated. We write the first order term using $g_p = \vec{\nabla} p$, transform the operators to local coordinates, integrate by parts, and replace the interface fluxes by a numerical flux and integrals by quadratures to get

$$\langle J \vec{\nabla} \cdot \vec{G}_{p,t}, \varphi_q \rangle_E = \langle \vec{\nabla}_\xi \cdot \vec{G}_{p,t}, \varphi_q \rangle_E \approx \int_{\partial e, N} \varphi_q \vec{G}_{p,t}^* \cdot \vec{n} dS - \langle \vec{G}_{p,t}, \vec{\nabla}_\xi \varphi_q \rangle_{E, N}. \quad (165)$$

For the interface fluxes, we use the BR1 method in interior faces,

$$P^* = \{P\}, \quad \vec{G}_{p,t}^* = \{\{\vec{G}_{p,t}\}\}, \quad (166)$$

and a homogeneous Neumann boundary condition in physical boundaries,

$$P^* = P, \quad \vec{G}_{p,t}^* \cdot \vec{n} = 0. \quad (167)$$

A.3 Semi-discrete stability

As in the continuous analysis, we maintain the results from the original model, but update the time derivative terms (123),

$$\langle \mathcal{J} C_t, \mu \rangle_{E, N} + \langle \mathcal{J} (\sqrt{\rho} \vec{U})_t, \sqrt{\rho} \vec{U} \rangle_{E, N} - \frac{1}{\rho_0 c_0^2} \int_{\partial e, N} P \vec{G}_{p,t}^* \cdot \vec{n} dS + \frac{1}{\rho_0 c_0^2} \langle \vec{G}_{p,t}, \vec{\nabla}_\xi P \rangle_{E, N}. \quad (168)$$

From the gradient definition (93b), with test function $\varphi_g = \vec{G}_{p,t}$

$$\langle \mathcal{J} \vec{G}_p, \vec{G}_{p,t} \rangle_{E, N} = \int_{\partial e, N} (P^* - P) (\vec{G}_{p,t} \cdot \vec{n}) dS + \langle \vec{G}_{p,t}, \vec{\nabla}_\xi P \rangle_{E, N}, \quad (169)$$

we get the time derivative terms,

$$\begin{aligned}
& \langle \mathcal{J}\mathcal{F}_t, 1 \rangle_{E,N} - \int_{\partial e,N} \frac{3}{2} \sigma \varepsilon (C_t \vec{G}_c^* + C_t^* \vec{G}_c - C_t \vec{G}_c) \cdot \vec{n} \, dS + \left\langle \mathcal{J} \left(\frac{1}{2} \rho V_{tot}^2 \right)_t, 1 \right\rangle_{E,N} \\
& + \left\langle \mathcal{J} \left(\frac{1}{\rho_0 c_0^2} \vec{G}_p \right), \vec{G}_{p,t} \right\rangle_{E,N} - \frac{1}{\rho_0 c_0^2} \int_{\partial e,N} (P \vec{G}_{p,t}^* + P^* \vec{G}_{p,t} - P \vec{G}_{p,t}) \cdot \vec{n} \, dS \\
= & \langle \mathcal{J}\mathcal{F}_t, 1 \rangle_{E,N} - \int_{\partial e,N} \frac{3}{2} \sigma \varepsilon (C_t \vec{G}_c^* + C_t^* \vec{G}_c - C_t \vec{G}_c) \cdot \vec{n} \, dS + \langle \mathcal{J}\mathcal{K}_t, 1 \rangle_{E,N} \\
& + \left\langle \mathcal{J} \left(\frac{1}{2\rho_0 c_0^2} |\vec{G}_p|^2 \right), 1 \right\rangle_{E,N} - \frac{1}{\rho_0 c_0^2} \int_{\partial e,N} (P \vec{G}_{p,t}^* + P^* \vec{G}_{p,t} - P \vec{G}_{p,t}) \cdot \vec{n} \, dS \\
= & \langle \mathcal{J}\mathcal{E}_{2,t}, 1 \rangle_{E,N} - \int_{\partial e,N} \frac{3}{2} \sigma \varepsilon (C_t \vec{G}_c^* + C_t^* \vec{G}_c - C_t \vec{G}_c) \cdot \vec{n} \, dS \\
& - \frac{1}{\rho_0 c_0^2} \int_{\partial e,N} (P \vec{G}_{p,t}^* + P^* \vec{G}_{p,t} - P \vec{G}_{p,t}) \cdot \vec{n} \, dS.
\end{aligned} \tag{170}$$

We have now obtained the time derivative of the alternative discrete entropy \mathcal{E}_2 (162). The first boundary integral belongs to the Cahn–Hilliard equation and was shown not to contribute at interior faces (Sec. 4.4.3) if $\beta = 0$, and to modify the entropy if $\beta > 0$. The surface free–energy is obtained in physical boundary faces (Sec. 4.4.5). The last integral was added by the discretization of the pressure time derivative Laplacian. The contribution from interior edges with the BR1 scheme (166) is

$$\begin{aligned}
\text{IBT}_{\nabla^2 p_t} &= \sum_{\substack{\text{interior} \\ \text{faces}}} \int_{f,N} \left(\llbracket P \rrbracket \vec{G}_{p,t}^* + \llbracket \vec{G}_{p,t} \rrbracket P^* - \llbracket P \vec{G}_{p,t} \rrbracket \right) \cdot \vec{n}_L \, dS \\
&= \sum_{\substack{\text{interior} \\ \text{faces}}} \int_{f,N} \left(\llbracket P \rrbracket \{ \vec{G}_{p,t} \} + \llbracket \vec{G}_{p,t} \rrbracket \{ P \} - \llbracket P \vec{G}_{p,t} \rrbracket \right) \cdot \vec{n}_L \, dS \\
&= 0.
\end{aligned} \tag{171}$$

For physical boundaries, we apply the homogeneous Neumann boundary condition (167), so that

$$\begin{aligned}
\text{PBT}_{\nabla^2 p_t} &= \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} (P \vec{G}_{p,t}^* + P \vec{G}_{p,t} - P \vec{G}_{p,t}) \cdot \vec{n} \, dS \\
&= \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} (0 + P \vec{G}_{p,t} \vec{n} - P \vec{G}_{p,t}) \cdot \vec{n} \, dS \\
&= 0.
\end{aligned} \tag{172}$$

As a result, the alternative entropy (162) satisfies a continuous bound (164), and discrete entropy law,

$$\frac{d}{dt} \left(\bar{\mathcal{E}}_2^\beta + \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{f,N} f_w(C) \, dS \right) \leq - \sum_e \langle \mathcal{J} (M_0 |\vec{G}_\mu|^2 + 2\eta \mathcal{S} : \mathcal{S}), 1 \rangle_{E,N} \leq 0, \tag{173}$$

where, similarly to (149), the inequality depends on whether the ERS and or central fluxes are used, and the choice of the interface penalty parameter β .

B Entropy contraction of the inviscid fluxes

We show the contraction of the inviscid fluxes (47) for the iNS/CH system. To do so, we replace the entropy variables, fluxes, and non-conservative terms,

$$\begin{aligned} \mathbf{w}^T \left(\vec{\nabla} \cdot \vec{\mathbf{f}}_e(\mathbf{q}) + \sum_{m=1}^5 \vec{\phi}_m(\mathbf{q}) \cdot \vec{\nabla} w_m \right) &= \mu \vec{\nabla} \cdot (c\vec{u}) + \frac{1}{2} u \vec{\nabla} \cdot (\rho u \vec{u}) + \frac{1}{2} v \vec{\nabla} \cdot (\rho v \vec{u}) + \frac{1}{2} w \vec{\nabla} \cdot (\rho w \vec{u}) + \vec{u} \cdot \vec{\nabla} p \\ &\quad + \frac{1}{2} \rho u \vec{u} \cdot \vec{\nabla} u + \frac{1}{2} \rho v \vec{u} \cdot \vec{\nabla} v + \frac{1}{2} \rho w \vec{u} \cdot \vec{\nabla} w + c \vec{u} \cdot \vec{\nabla} \mu + p \vec{\nabla} \cdot \vec{u} \\ &= \vec{\nabla} \cdot \left(\left(\frac{1}{2} \rho v_{tot}^2 + p + c\mu \right) \vec{u} \right) = \vec{\nabla} \cdot \vec{f}_e^{\mathcal{E}}. \end{aligned} \quad (174)$$

Thus, we see that the term $\vec{\nabla} \cdot (c\vec{u})$ in the Cahn–Hilliard equation cancels the capillary pressure $c\vec{\nabla}\mu$ from the momentum non-conservative terms when multiplied by their respective entropy variables. Similarly, it reveals that momentum terms from the conservative and non-conservative parts cancel each other, and that the pressure term in the momentum equation cancels the velocity divergence term by way of the artificial compressibility equation.

C Point-wise discretization

In this section we list the steps to compute the solution time derivative with the DG approximation, (93). To get the point-wise values, we replace the test function by the Lagrange polynomials $l_i(\xi)l_j(\eta)l_k(\zeta)$. Details on the extraction of point-wise values can be found in [35].

1. Compute the entropy variables, \mathbf{W} . The first entropy variable is the chemical potential, μ , which is computed from (93d) and (93c),

$$\begin{aligned} \mathcal{J}_{ijk} \vec{G}_{c,ijk} &= (C^* - C)_{ijk} \left(\frac{\delta_{im}}{w_i} \mathcal{J} \vec{a}_{ijk}^1 + \frac{\delta_{jm}}{w_j} \mathcal{J} \vec{a}_{ijk}^2 + \frac{\delta_{km}}{w_k} \mathcal{J} \vec{a}_{ijk}^3 \right) \Big|_{m=0}^{m=N} \\ &\quad + \mathcal{J} \vec{a}_{ijk}^1 \sum_{m=0}^N D_{im} C_{mjk} + \mathcal{J} \vec{a}_{ijk}^2 \sum_{m=0}^N D_{jm} C_{imk} + \mathcal{J} \vec{a}_{ijk}^3 \sum_{m=0}^N D_{km} C_{ijm} \\ \mathcal{J}_{ijk} \mu_{ijk} &= \mathcal{J}_{ijk} F'_0(C_{ijk}) - \frac{3}{2} \sigma \varepsilon \left(\frac{\delta_{im}}{w_i} \tilde{G}_{c,ijk}^{1,*} + \frac{\delta_{jm}}{w_j} \tilde{G}_{c,ijk}^{2,*} + \frac{\delta_{km}}{w_k} \tilde{G}_{c,ijk}^{3,*} \right) \Big|_{m=0}^{m=N} \\ &\quad + \frac{3}{2} \sigma \varepsilon \sum_{m=0}^N \left(\frac{w_m}{w_i} D_{mi} \tilde{G}_{c,mjk}^1 + \frac{w_m}{w_j} D_{mj} \tilde{G}_{c,imk}^2 + \frac{w_m}{w_k} D_{mk} \tilde{G}_{c,ijm}^3 \right), \end{aligned} \quad (175)$$

where $D_{ij} = l'_j(\xi_i)$ is the derivative matrix, and recall the definition of contravariant components (69). We then get the entropy variables from the current values of the state vector, \mathbf{Q} , and the chemical potential, μ computed by (175),

$$\mathbf{W}_{ijk} = \left(\mu, \frac{Q_2}{\sqrt{\rho}}, \frac{Q_3}{\sqrt{\rho}}, \frac{Q_4}{\sqrt{\rho}}, \frac{Q_5}{\rho_0 c_0^2} \right)_{ijk}, \quad \rho = \rho_1 \hat{C} + \rho_2 (1 - \hat{C}), \quad \hat{C} = \min(\max(C, 0), 1). \quad (176)$$

2. Compute the gradient of the entropy variables $\vec{\mathbf{G}}$ from (93b),

$$\begin{aligned} \mathcal{J}_{ijk} \vec{\mathbf{G}}_{ijk} &= (\mathbf{W}^{*,T} - \mathbf{W}^T)_{ijk} \left(\frac{\delta_{im}}{w_i} \mathcal{J} \vec{a}_{ijk}^1 + \frac{\delta_{jm}}{w_j} \mathcal{J} \vec{a}_{ijk}^2 + \frac{\delta_{km}}{w_k} \mathcal{J} \vec{a}_{ijk}^3 \right) \Big|_{m=0}^{m=N} \\ &\quad + \mathcal{J} \vec{a}_{ijk}^1 \sum_{m=0}^N D_{im} \mathbf{W}_{mjk} + \mathcal{J} \vec{a}_{ijk}^2 \sum_{m=0}^N D_{jm} \mathbf{W}_{imk} + \mathcal{J} \vec{a}_{ijk}^3 \sum_{m=0}^N D_{km} \mathbf{W}_{ijm}. \end{aligned} \quad (177)$$

3. Compute the state vector time derivative,

$$\begin{aligned}
& \mathcal{J}_{ijk} \underline{\mathbf{M}}_{ijk} \frac{d\mathbf{Q}_{ijk}}{dt} + \left(\overset{\leftrightarrow}{\mathbf{F}}_e^* - \overset{\leftrightarrow}{\mathbf{F}}_e + \sum_{m=1}^5 (\overset{\leftrightarrow}{\Phi}_m W_m)^\diamond - \overset{\leftrightarrow}{\Phi}_m W_m \right)_{ijk} \cdot \left(\frac{\delta_{im}}{w_i} \mathcal{J} \bar{a}_{ijk}^1 + \frac{\delta_{jm}}{w_j} \mathcal{J} \bar{a}_{ijk}^2 + \frac{\delta_{km}}{w_k} \mathcal{J} \bar{a}_{ijk}^3 \right) \Big|_{m=0}^{m=N} \\
& + \sum_{m=0}^N (D_{im} \tilde{\mathbf{F}}_{e,mjk}^1 + D_{jm} \tilde{\mathbf{F}}_{e,imk}^2 + D_{km} \tilde{\mathbf{F}}_{e,ijm}^3) \\
& + \sum_{p=1}^5 \sum_{m=1}^N (\tilde{\Phi}_{m,ijk}^1 D_{ip} W_{m,pjk} + \tilde{\Phi}_{m,ijk}^2 D_{jp} W_{m,ipk} + \tilde{\Phi}_{m,ijk}^3 D_{kp} W_{m,ijp}) \\
& = \left(\frac{\delta_{im}}{w_i} \tilde{\mathbf{F}}_{v,ijk}^{1,*} + \frac{\delta_{jm}}{w_j} \tilde{\mathbf{F}}_{v,ijk}^{2,*} + \frac{\delta_{km}}{w_k} \tilde{\mathbf{F}}_{v,ijk}^{3,*} \right) \Big|_{m=0}^{m=N} \\
& + \sum_{m=0}^N \left(\frac{w_m}{w_i} D_{mi} \tilde{\mathbf{F}}_{v,mjk}^1 + \frac{w_m}{w_j} D_{mj} \tilde{\mathbf{F}}_{v,imk}^2 + \frac{w_m}{w_k} D_{mk} \tilde{\mathbf{F}}_{v,ijm}^3 \right) + \mathcal{J}_{ijk} \mathbf{S}_{ijk}.
\end{aligned} \tag{178}$$

References

1. M. Sussman, K. M. Smith, M. Y. Hussaini, M. Ohta, R. Zhi-Wei, A sharp interface method for incompressible two-phase flows, *Journal of computational physics* 221 (2) (2007) 469–505.
2. E. Olsson, G. Kreiss, A conservative level set method for two phase flow, *Journal of computational physics* 210 (1) (2005) 225–246.
3. J. Lowengrub, L. Truskinovsky, Quasi–incompressible Cahn–Hilliard fluids and topological transitions, *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* 454 (1978) (1998) 2617–2654.
4. J. W. Cahn, J. E. Hilliard, Free energy of a nonuniform system. I. Interfacial free energy, *The Journal of chemical physics* 28 (2) (1958) 258–267.
5. J. W. Cahn, J. E. Hilliard, Free energy of a nonuniform system. III. Nucleation in a two-component incompressible fluid, *The Journal of chemical physics* 31 (3) (1959) 688–699.
6. J. Shen, On a new pseudocompressibility method for the incompressible Navier–Stokes equations, *Applied numerical mathematics* 21 (1) (1996) 71–90.
7. B. S. Hosseini, S. Turek, M. Möller, C. Palmes, Isogeometric analysis of the Navier–Stokes–Cahn–Hilliard equations with application to incompressible two-phase flows, *Journal of Computational Physics* 348 (2017) 171–194.
8. J. Manzanero, G. Rubio, D. A. Kopriva, E. Ferrer, E. Valero, A free-energy stable nodal discontinuous Galerkin approximation with summation-by-parts property for the Cahn–Hilliard equation, *arXiv preprint arXiv:1902.08089*.
9. J. Manzanero, G. Rubio, D. A. Kopriva, E. Ferrer, E. Valero, Entropy-stable discontinuous Galerkin approximation with summation-by-parts property for the incompressible Navier–Stokes equations with variable density and artificial compressibility, *arXiv preprint arXiv:1907.05976*.
10. T.C. Fisher and M.H. Carpenter, High-order entropy stable finite difference schemes for nonlinear conservation laws: Finite domains, *Journal of Computational Physics* 252 (2013) 518–557.
11. M. H. Carpenter, T. C. Fisher, E. J. Nielsen, S. H. Frankel, Entropy stable spectral collocation schemes for the Navier–Stokes equations: Discontinuous interfaces, *SIAM Journal on Scientific Computing* 36 (5) (2014) B835–B867.
12. J. Chan, D. C. Del Rey Fernández, M. H. Carpenter, Efficient entropy stable Gauss collocation methods, *SIAM Journal on Scientific Computing* 41 (5) (2019) A2938–A2966.
13. D.A. Kopriva and G.J. Gassner, An energy stable discontinuous Galerkin spectral element discretization for variable coefficient advection problems, *SIAM Journal on Scientific Computing* 36 (4) (2014) A2076–A2099.

14. J. Manzanero, G. Rubio, E. Ferrer, E. Valero and D.A. Kopriva, Insights on aliasing driven instabilities for advection equations with application to Gauss-Lobatto discontinuous Galerkin methods, *Journal of Scientific Computing*.
15. G.J. Gassner, A skew-symmetric discontinuous Galerkin spectral element discretization and its relation to SBP-SAT finite difference methods, *SIAM Journal on Scientific Computing* 35 (3) (2013) 1233–1256.
16. G.J. Gassner, A. Winters, Andrew, F. Hindenlang, D.A. Kopriva, The BR1 scheme is stable for the compressible Navier-Stokes equations, *Journal of Scientific Computing*.
17. G. J. Gassner, A. R. Winters, D. A. Kopriva, A well balanced and entropy conservative discontinuous Galerkin spectral element method for the shallow water equations, *Applied Mathematics and Computation* 272 (2016) 291–308.
18. G.J. Gassner, A.R. Winters and D.A. Kopriva, Split form nodal discontinuous Galerkin schemes with Summation-By-Parts property for the compressible Euler equations, *Journal of Computational Physics*, in Press.
19. A.R. Winters and G.J. Gassner, Affordable, entropy conserving and entropy stable flux functions for the ideal MHD equations, *Journal of Computational Physics* 304 (2016) 72 – 108.
20. P. C. Hohenberg, B. I. Halperin, Theory of dynamic critical phenomena, *Reviews of Modern Physics* 49 (3) (1977) 435.
21. J. Shen, X. Yang, Energy stable schemes for Cahn–Hilliard phase-field model of two-phase incompressible flows, *Chinese Annals of Mathematics, Series B* 31 (5) (2010) 743–758.
22. J.-L. Guermond, L. Quartapelle, A projection fem for variable density incompressible flows, *Journal of Computational Physics* 165 (1) (2000) 167–188.
23. H. Ding, P. D. Spelt, C. Shu, Diffuse interface model for incompressible two-phase flows with large density ratios, *Journal of Computational Physics* 226 (2) (2007) 2078–2095.
24. S. Dong, Multiphase flows of N immiscible incompressible fluids: A reduction-consistent and thermodynamically consistent formulation and associated algorithm, *Journal of Computational Physics* 361 (2018) 1–49.
25. F. Bassi, F. Massa, L. Botti, A. Colombo, Artificial compressibility Godunov fluxes for variable density incompressible flows, *Computers & Fluids* 169 (2018) 186–200.
26. H. Abels, H. Garcke, G. Grün, Thermodynamically consistent, frame indifferent diffuse interface models for incompressible two-phase flows with different densities, *Mathematical Models and Methods in Applied Sciences* 22 (03) (2012) 1150013.
27. J. Shen, Pseudo-compressibility methods for the unsteady incompressible Navier-Stokes equations, in: *Proceedings of the 1994 Beijing symposium on nonlinear evolution equations and infinite dynamical systems*, 1997, pp. 68–78.
28. X. Feng, J. Kou, S. Sun, A novel energy stable numerical scheme for Navier-Stokes-Cahn-Hilliard two-phase flow model with variable densities and viscosities, in: *International Conference on Computational Science*, Springer, 2018, pp. 113–128.
29. G. Zhu, H. Chen, J. Yao, S. Sun, Efficient energy-stable schemes for the hydrodynamics coupled phase-field model, *Applied Mathematical Modelling* 70 (2019) 82–108.
30. J. Shen, X. Yang, Decoupled, energy stable schemes for phase-field models of two-phase incompressible flows, *SIAM Journal on Numerical Analysis* 53 (1) (2015) 279–296.
31. F. Boyer, C. Lapuerta, S. Minjeaud, B. Piar, M. Quintard, Cahn–Hilliard/Navier–Stokes model for the simulation of three-phase flows, *Transport in Porous Media* 82 (3) (2010) 463–483.
32. E. Tadmor, Entropy stability theory for difference approximations of nonlinear conservation laws and related time-dependent problems, *Acta Numerica* 12 (2003) 451–512.
33. S. Dong, On imposing dynamic contact-angle boundary conditions for wall-bounded liquid–gas flows, *Computer Methods in Applied Mechanics and Engineering* 247 (2012) 179–200.
34. D. A. Kopriva, Metric identities and the discontinuous spectral element method on curvilinear meshes, *Journal of Scientific Computing* 26 (3) (2006) 301.
35. D.A. Kopriva, *Implementing spectral methods for partial differential equations*, Springer Netherlands, 2009.
36. D. A. Kopriva, A polynomial spectral calculus for analysis of DG spectral element methods, in: *Spectral and High Order Methods for Partial Differential Equations ICOSAHOM 2016*, Springer, 2017, pp. 21–40.
37. M. Bohm, A. R. Winters, G. J. Gassner, D. Derigs, F. Hindenlang, J. Saur, An entropy stable nodal discontinuous Galerkin method for the resistive MHD equations. part I: Theory and numerical verification, *Journal of Computational Physics*.

38. F. Bassi and S. Rebay, A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations, *Journal of Computational Physics* 131 (2) (1997) 267 – 279.
39. E. Toro, *Riemann solvers and numerical methods for fluid dynamics*, Springer, 2009.
40. J. Manzanero, A.M. Rueda-Ramírez, G. Rubio and E. Ferrer, The Bassi Rebay 1 scheme is a special case of the Symmetric Interior Penalty formulation for discontinuous Galerkin discretisations with Gauss-Lobatto points, *Journal of Computational Physics* 363 (2018) 1 – 10.
41. J.H. Williamson, Low-storage Runge-Kutta schemes, *Journal of Computational Physics*.
42. G.J. Gassner and D.A. Kopriva, A comparison of the dispersion and dissipation errors of Gauss and Gauss-Lobatto discontinuous Galerkin spectral element methods, *SIAM Journal on Scientific Computing* 33 (5) (2011) 2560–2579.
43. J. Manzanero, G. Rubio, E. Ferrer and E. Valero, Dispersion-dissipation analysis for advection problems with nonconstant coefficients: Applications to discontinuous Galerkin formulations, *SIAM Journal on Scientific Computing* 40 (2) (2018) A747–A768.
44. T. Lee, Effects of incompressibility on the elimination of parasitic currents in the lattice Boltzmann equation method for binary fluids, *Computers & Mathematics with Applications* 58 (5) (2009) 987–994.
45. S.-R. Hysing, S. Turek, D. Kuzmin, N. Parolini, E. Burman, S. Ganesan, L. Tobiska, Quantitative benchmark computations of two-dimensional bubble dynamics, *International Journal for Numerical Methods in Fluids* 60 (11) (2009) 1259–1288.
46. Y. Taitel, A. Dukler, A model for predicting flow regime transitions in horizontal and near horizontal gas-liquid flow, *AIChE journal* 22 (1) (1976) 47–55.
47. J. Manzanero, C. Redondo, E. Ferrer, E. Valero, A high-order discontinuous Galerkin solver for multiphase flows, In *Lecture notes in computational science and engineering ICOSAHOM 2018*. (Accepted for publication).
48. E. Ferrer, An interior penalty stabilised incompressible Discontinuous Galerkin - Fourier solver for implicit Large Eddy Simulations, *Journal of Computational Physics* 348 (2017) 754–775.
49. David Flad and Gregor Gassner, On the use of kinetic energy preserving DG-schemes for large eddy simulation, *Journal of Computational Physics* 350 (Supplement C) (2017) 782 – 795.
50. J. Manzanero, E. Ferrer, G. Rubio, E. Valero, On the role of numerical dissipation in stabilising under-resolved turbulent simulations using discontinuous Galerkin methods, *arXiv preprint arXiv:1805.10519*.