A free–energy stable p–adaptive nodal discontinuous Galerkin for the Cahn–Hilliard equation

Gerasimos Ntoukas *,[1], Juan Manzanero[1,2], Gonzalo Rubio[1,2], Eusebio Valero[1,2], and Esteban Ferrer[1,2]

[1]ETSIAE-UPM - School of Aeronautics, Universidad Politécnica de Madrid, Plaza Cardenal Cisneros 3, E-28040 Madrid, Spain
[2]Center for Computational Simulation, Universidad Politécnica de Madrid, Campus de Montegancedo, Boadilla del Monte, 28660 Madrid, Spain

August 2020

Abstract

A novel free–energy stable discontinuous Galerkin method is developed for the Cahn–Hilliard equation with non–conforming elements. This work focuses on dynamic polynomial adaptation (p–refinement) and constitutes an extension of the method developed by Manzanero et al. in Journal of Computational Physics 403:109072, 2020, which makes use of the summation–by–parts simultaneous–approximation term technique along with Gauss–Lobatto points and the Bassi–Rebay 1 (BR1) scheme. The BR1 numerical flux accommodates non–conforming elements, which are connected through the mortar method. The scheme has been analytically proven to retain its free–energy stability when transitioning to non-conforming elements. Furthermore, a methodology to perform the adaptation is introduced based on the knowledge of the location of the interface between phases. The adaptation methodology is tested for its accuracy and effectiveness through a series of steady and unsteady test cases. We test the scheme for freestream preservation and primary quantity conservation on non–conforming curvilinear meshes. We solve a steady one–dimensional interface test case to initially examine the accuracy of the adaptation. Furthermore, we study the formation of a static bubble in two dimensions and verify that the accuracy of the solver is maintained while the degrees of freedom decrease to less than half compared to the uniform solution. Lastly, we examine an unsteady case such as the spinodal decomposition and show that the same results for the free–energy are recovered, with a 35% reduction of the degrees of freedom for the two–dimensional case considered and a 48% reduction for the three–dimensional case.

Keywords: Discontinuous Galerkin, Cahn–Hilliard, Phase–field method, High–Order Methods, P-adaptation, Energy Stable

*Corresponding Author
E-mail address: gerasimos.ntoukas@upm.es (G. Ntoukas)

1 Introduction

A variety of different approaches and methodologies have been developed throughout the years to model and simulate systems of two or more immiscible fluids. These methods can be divided into two large categories. The interface–tracking and the interface–capturing techniques. The former includes the Marker–And–Cell (MAC) and front–tracking methods whereas the latter include the widely known Volume Of Fluid (VOF), level–set and phase field methods. A review of the characteristics and features of the various methods can be found in [1].

In this work, we focus on phase–field methods and especially the Cahn–Hilliard model [2]. Its use for multiphase modeling has growing popularity in the recent years because of the favorable total mass conservation characteristics [3] as well as the bounded behavior of the free–energy [4].

The essence of using a diffuse interface method such as the Cahn–Hilliard model is that the sharp interface between the two immiscible fluids is represented by an interface of a finite width throughout which, the fluid thermodynamic properties smoothly vary from those of one phase to another. Typically, the physical width of the interface is of the order of nanometers [5]. However, this constitutes an unrealistic target as to numerically capture this interface would require a large amount of resources even for the simplest of the cases [6]. The answer to this is partially derived from the sharp interface limit of the Cahn–Hilliard model. Through asymptotic analysis [7], it can be proved that the Cahn–Hilliard model converges to the physical solution even with the use of a substantially larger interface. This characteristic makes a simulation cost–effective while retaining the desired accuracy.

The discretization scheme adopted is the nodal Discontinuous Galerkin Spectral Element Method (DGSEM) based on previous work presented by Manzanero et al. [8]. The DGSEM offers great flexibility as the solution is represented through an arbitrary approximation polynomial order and supports the use of unstructured meshes of curvilinear hexahedral elements to approximate complex geometries. Furthermore, another advantage of the DGSEM exploited in this work is that the approximation order can vary across elements. There is a limited amount of publications concerning the use of discontinuous Galerkin methods in order to solve the Cahn–Hilliard equation [8–14] and there is still a variety of aspects regarding the efficiency, robustness and accuracy that should be addressed.

The DGSEM version of this work uses Gauss–Lobatto (GL) points, which through the Summation–By–Parts Simultaneous–Approximation Term (SBP–SAT) property [15, 16] allow the derivation of free–energy stable schemes. There is a plethora of work focused on the construction of entropy and energy stable schemes for the discontinuous Galerkin method [15, 17–23] and the references therein. A review of entropy stable DGSEM schemes is given in [24]. As for the Cahn–Hilliard equation, a DGSEM with the SBP–SAT property has been developed in [8, 9]. These schemes however, are designed for conforming elements and they should be modified accordingly such that they can retain their stability properties when using non–conforming elements.

There is a growing popularity of entropy stable schemes with special consideration on the extension of their characteristics to non–conforming elements. These ideas can contribute to the creation of a free–energy stable scheme as the one developed in this work. The foundation for the establishment of this family of methods has been laid in [25–29] on linear problems. One of the first works is the one presented by Carpenter et al. [30] which makes a first attempt to create an entropy–stable scheme using the local discontinuous Galerkin method. Again, the scheme uses the SBP–SAT property of the Gauss–Lobatto (GL) points and is proved to be entropy–stable for the compressible Euler equations for unstructured hexahedral meshes. This method has been extended to the Navier–Stokes equations by Parsani et al. [31, 32]. Then in [33] the

classic mortar method is modified to guarantee entropy stability of the discontinuous Galerkin method. Following that work, the authors in [34] extended the scheme from the compressible Euler equations to the compressible Navier–Stokes equations and into the use of curvilinear elements. In addition, [35] is also an extension of the aforementioned work and focuses on an entropy stable discontinuous Galerkin scheme. Another work on non–conforming curvilinear elements for hyperbolic conservation laws is presented in [36]. There is a plethora of publications concerning the design of energy– and entropy– stable operators for linear and nonlinear problems for non–conforming elements and up to our knowledge, an equivalent free–energy stable scheme for non–conforming elements has not been previously established for the Cahn–Hilliard model.

For diffuse interface methods, such as the Cahn–Hilliard, the region of interest is that of the interface. More specifically, within the interface there are steep gradients, which should be adequately resolved. On the contrary, on the bulk of each phase, the phase-field variable is constant and the resolution can be lowered. Thus, the resolution can be coarsened in the regions away from the interface and refined within the vicinity of the interface. The interested reader can find a wide array of publications that address the aforementioned issue in the context of mesh adaptation (h–refinement) [37–45] with respect to the refinement indicator, mesh refinement strategy and dynamic adaptation. These correspond to different frameworks, employing numerical schemes such as finite differences [39, 40], finite volume [41, 42] and finite elements [37, 38, 44, 45]. Also, there are some works on finite elements that identify the effect of the polynomial order on the accuracy of the solution [13, 46], but polynomial adaptation (p–refinement) is not addressed. Up to the authors' knowledge, there is no similar work performed on the effectiveness of local polynomial adaptation for phase-field problems when used in conjunction with the discontinuous Galerkin scheme.

There are various methods that can be used to mark the refinement region for Adaptive Mesh Refinement (AMR), which could potentially be applied to polynomial refinement. Similar studies on dynamic mesh adaptation for diffuse interface models have been performed with the use of gradient based indicators [38,44], Legendre discretisation spectrum extrapolation [47], summation of the tail of the spectrum [48], flux jump of the Laplace operator [37], the location of the interface [39,49], as well as other error indicators [50–52]. Additionally, there is a theoretical study on the a priori determination of the error for the discontinuous Galerkin method when used for the Cahn–Hilliard equation [53]. A possible alternative approach for the polynomial refinement of the Cahn–Hilliard is the truncation error estimation as presented in [54] and applied to the Navier–Stokes equations in [55–57], which could allow anisotropic refinement in each element. In this work, we take advantage of the knowledge of the location of the interface to perform the adaptation.

The aim of this work is to create a scheme that retains the accuracy and stability characteristics when transitioning to the use of the non–uniform polynomial order across the domain. To achieve this, the first step was to modify the BR1 numerical flux to handle such elements and incorporate the mortar method [58]. The modified scheme is then proved analytically to be free–energy stable for the general case of non-conforming element boundaries following the methodology in [8]. The time marching is performed through a first order IMplicit–EXplicit (IMEX) scheme. This choice has been made to alleviate the stiffness of the system due to the higher order derivatives. The free–energy of the system is proved to be bounded for the continuous and the discrete time settings.

We also develop a methodology to perform automatic p–adaptation for the Cahn–Hilliard equation when using the DGSEM. This is based on the location of the interface and it takes advantage of the attributes of the scheme's spatial locality. The DGSEM offers the opportunity to augment

the polynomial order within each element of the mesh and also individually in each direction. Therefore, only the elements that contain part of the interface are refined while the rest are coarsened. Thus, through this simple and efficient process we can achieve a reduction in the degrees of freedom without further processing of the mesh. The accuracy and convergence characteristics are subsequently examined through a series of different tests and direct comparison with the conforming version of the scheme.

The rest of the work is organized as follows. First the model used and its characteristics are specified in Sec. 2. Then the spatial and temporal discretisation methods are introduced in Sec. 3, as well as the modified BR1 numerical fluxes that allow non-uniform polynomial order in Sec. 3.2. The adaptation methodology is described in Sec. 4. In Sec. 5 the foundation of the method is laid through an analytical proof of the free–energy boundedness of the scheme. A verification for the freestream preservation and primary quantity conservation for general 3D p–non–conforming meshes is presented in Section 6. In Sec. 7 three different test cases are presented along with the results from the developed method and a comparison is made with the conforming solver version with respect to the achieved accuracy and degrees of freedom.

2 Cahn–Hilliard equation and continuous energy estimates

This work is based on the Cahn–Hilliard equation, which can be utilized to describe a vast variety of physical phenomena such as the dynamics of phase separation of two and N phase flows, binary alloys, tumor growth, etc. [3]. The application of interest in this particular case is the study of two phase flows. The model consists of a constant mobility parameter and a polynomic double–well chemical free–energy [59, 60]. The Cahn–Hilliard equation is defined as

$$\phi_t = \nabla \cdot (M \nabla w), \quad \text{in } \Omega, \tag{1}$$

The phase field variable ϕ represents the concentration of each phase and satisfies (1), M represents the mobility which is a positive parameter, Ω is the physical domain (with boundary $\partial\Omega$), and w is a scalar field representing the *chemical potential*. The chemical potential is designed to minimize an arbitrary free–energy functional, $\mathcal{F}(\phi, \nabla\phi)$, which depends on the phase field and its gradient,

$$w = \frac{\delta \mathcal{F}}{\delta \phi}. \tag{2}$$

A homogeneous Neumann boundary condition is applied for the chemical potential in order to ensure that mass is conserved [61],

$$\nabla w \cdot \vec{n}\Big|_{\partial\Omega} = 0. \tag{3}$$

The free–energy consists of two terms that impose opposing effects. The chemical free–energy, ψ, which drives phase separation, and the interfacial energy $\frac{1}{2}k|\nabla\phi|^2$, which promotes homogenization by penalizing the existence of gradients. The parameter k is positive and controls the width of the interface. The two energies lead into the existence of an interface,

$$\mathcal{F} = \int_\Omega \left(\psi(\phi) + \frac{1}{2} k |\nabla \phi|^2 \right) \mathrm{d}\vec{x} - \int_{\partial\Omega} g(\phi)\, \mathrm{d}S = F_v(\phi) + F_s(\phi). \tag{4}$$

In (4), $F_v(\phi)$ and $F_s(\phi)$ represent the volumetric and surface free–energies accordingly. The term $g(\phi)$ represents a boundary energy that will also be minimized with appropriate boundary conditions, and k is the interfacial energy coefficient.
The minimization is performed by linearization of the free–energy (4) around an equilibrium solution,

$$\delta\mathcal{F} = \int_\Omega \left(\frac{\mathrm{d}\psi}{\mathrm{d}\phi}\delta\phi + k\nabla\phi \cdot \nabla(\delta\phi) \right) \mathrm{d}\vec{x} - \int_{\partial\Omega} \frac{\mathrm{d}g}{\mathrm{d}\phi}\delta\phi\, \mathrm{d}S, \tag{5}$$

where $\delta\phi$ acts as a small perturbation. Since we will also apply Neumann boundary conditions for ϕ, the perturbation $\delta\phi$ is not restricted to vanish at the boundaries $\partial\Omega$. The integration of the second term of the first integral in (5) by parts,

$$\delta\mathcal{F} = \int_\Omega \left(\frac{\mathrm{d}\psi}{\mathrm{d}\phi} - k\nabla^2\phi \right) \delta\phi \, \mathrm{d}\vec{x} - \int_{\partial\Omega} \left(\frac{\mathrm{d}g}{\mathrm{d}\phi} - k\nabla\phi \cdot \vec{n} \right) \delta\phi \, \mathrm{d}S, \tag{6}$$

yields both the chemical potential definition,

$$w = \frac{\mathrm{d}\psi}{\mathrm{d}\phi} - k\nabla^2\phi, \tag{7}$$

and the appropriate Neumann boundary conditions prescription,

$$k\nabla\phi \cdot \vec{n}\Big|_{\partial\Omega} = \frac{\mathrm{d}g}{\mathrm{d}\phi}. \tag{8}$$

In this work we use a polynomial double–well function for the chemical free–energy [2],

$$\psi(\phi) = (1-\phi)^2(1+\phi)^2, \tag{9}$$

and a linear function for the boundary free–energy,

$$g(\phi) = \beta\phi, \tag{10}$$

since they represent standard choices in the literature, but other choices that are not covered here exist (e.g. logarithmic chemical free–energy [62]).
An exact solution of the one–dimensional steady state Cahn–Hilliard equation (1), with the chemical free–energy (9) and the free–energy (4), in an infinite domain with $\phi(\pm\infty) = \pm 1$ is given by

$$\phi = \tanh\left(\frac{\sqrt{2}x}{\sqrt{k}} \right). \tag{11}$$

To facilitate the stability analysis of the system as well as the discretization, and following the processes highlighted in [8, 10], the Cahn–Hilliard equation (1) is transformed into a system of four first order equations,

$$\phi_t = \nabla \cdot \left(M\vec{f} \right), \tag{12a}$$

$$\vec{f} = \nabla w, \tag{12b}$$

$$w = \frac{\mathrm{d}\psi}{\mathrm{d}\phi} - k\nabla \cdot \vec{q}, \tag{12c}$$

$$\vec{q} = \nabla\phi. \tag{12d}$$

To create the weak form, we have introduced two auxiliary variables $\vec{q} = \nabla\phi$ and $\vec{f} = \nabla w$. We multiply (12a) and (12c) with the arbitrary scalar test functions φ_Φ and φ_W respectively and (12b) and (12d) with the arbitrary vectorial counterparts $\vec{\varphi}_F$ and $\vec{\varphi}_Q$. Then we integrate over the domain Ω and derive four weak forms,

$$\langle \phi_t, \varphi_\Phi \rangle = \left\langle \nabla \cdot \left(M\vec{f} \right), \varphi_\Phi \right\rangle, \tag{13a}$$

$$\left\langle \vec{f}, \vec{\varphi}_F \right\rangle = \langle \nabla w, \vec{\varphi}_F \rangle, \tag{13b}$$

$$\langle w, \varphi_W \rangle = \left\langle \frac{\mathrm{d}\psi}{\mathrm{d}\phi}, \varphi_W \right\rangle - k \langle \nabla \cdot \vec{q}, \varphi_W \rangle, \tag{13c}$$

$$\langle \vec{q}, \vec{\varphi}_Q \rangle = \langle \nabla\phi, \vec{\varphi}_Q \rangle, \tag{13d}$$

where the operator $\langle f, g \rangle$ is the L^2 inner product

$$\langle f, g \rangle = \int_\Omega fg \, \mathrm{d}x. \tag{14}$$

2.1 Continuous free–energy stability

The analytical proof of the free–energy boundedness with a given initial condition is given in [8]. The process makes use of the transformed system (13) and then through some manipulation of the equations the following inequality can be derived,

$$\mathcal{F}(T) = \mathcal{F}(0) - \int_0^T \left\langle M\vec{f}, \vec{f} \right\rangle \mathrm{dt} \leqslant \mathcal{F}(0). \tag{15}$$

As a result, the Cahn–Hilliard equation (1) with chemical potential (7) and Neumann boundary conditions (3) and (8) guarantees that the free–energy $\mathcal{F}$ evaluated at any time instant T, as defined in (4), is bounded.This is the property to be mimicked by the subsequent approximation.

3 The nodal discontinuous Galerkin spectral element method

In this section we briefly introduce the underlying theory and the construction of the non–conforming nodal DGSEM. The interpolating nodes are the GL points as they satisfy the SBP–SAT property (19). This is a crucial aspect to prove the scheme's stability, avoiding the use of exact integration. Furthermore, a non–conforming DGSEM is constructed, where the polynomial order can vary across different elements [63].

The computational domain Ω is tessellated into non–overlapping hexahedral elements, which are then geometrically transformed from a reference element $e = [-1, 1]^3$ by means of a transfinite mapping. This mapping relates the physical ($\vec{x} = (x^1, x^2, x^3) = (x, y, z)$) and the local ($\vec{\xi} = (\xi^1, \xi^2, \xi^3) = (\xi, \eta, \zeta)$) coordinates,

$$\vec{x} = \vec{X}(\vec{\xi}) = \vec{X}(\xi, \eta, \zeta). \tag{16}$$

The solutions and functions are approximated by order N_e polynomials in an element e,

$$\mathcal{I}^{N_e}\left[u(x, y, z, t)\right]_e = U(\xi, \eta, \zeta, t) = \sum_{i,j,k=0}^{N_e} U_{ijk}(t) l_i(\xi) l_j(\eta) l_k(\zeta), \tag{17}$$

where the approximation order N_e can vary from element to element. In (17), l_j are the Lagrange interpolating polynomials whose nodes are a set of Gauss–Lobatto points in the reference element e and $U_{ijk}(t)$ are the (time dependent) nodal values of an arbitrary function u. The notation is as follows: we use lower cases for the exact functions, whereas upper cases represent their polynomial approximation.
We approximate the integrals with quadrature rules that use the same GL nodes as those that represent the solution,

$$\int_{-1}^{1} FG \,\mathrm{d}\xi \approx \int_{E,N_e} FG \,\mathrm{d}\xi = \sum_{m=0}^{N_e} w_m F_m G_m, \tag{18}$$

where w_i are the quadrature weights [64]. This provides an exact integration for $2N_e - 1$ order polynomial (see [65]). The choice of the GL nodes is essential since it makes the quadrature rule (18) to satisfy the discrete SBP-SAT property, that is, a discrete Gauss law, which in one dimension is

$$\int_{E,N_e} \frac{\mathrm{d}U}{\mathrm{d}\xi} V \,\mathrm{d}\xi = U_{N_e} V_{N_e} - U_0 V_0 - \int_{E,N_e} U \frac{\mathrm{d}V}{\mathrm{d}\xi} \,\mathrm{d}\xi. \tag{19}$$

The SBP–SAT property makes it possible to resemble the continuous analysis that proves the boundedness of the free energy $\mathcal{F}$ in (15) discretely.

The transformations from the reference element to the physical space must create a *watertight* mesh (i.e. without gaps across the elements). This constraints the mapping functions at the inter–element faces to be the same,

$$\vec{X}^{e_L}\left(\xi,\eta,\zeta\right)\Big|_{\text{face}} = \vec{X}^{e_R}\left(\xi,\eta,\zeta\right)\Big|_{\text{face}}. \tag{20}$$

From the mapping, we compute a set of covariant and contravariant vector bases,

$$\vec{a}_j = \frac{\partial \vec{X}}{\partial \xi^j}, \quad \vec{a}^j = \nabla \xi^j, \quad j = 1,2,3. \tag{21}$$

The relation between the covariant and (volume weighted) contravariant bases is

$$J\vec{a}^i = \vec{a}_j \times \vec{a}_k, \quad (i,j,k) \text{ cyclic}, \tag{22}$$

where J is the Jacobian of the transformation, $J = \vec{a}_1 \cdot (\vec{a}_2 \times \vec{a}_3)$. The volume weighted contravariant bases satisfy the continuous metric identities,

$$\sum_{i=1}^{3} \frac{\partial J a_n^i}{\partial \xi^i} = 0, \quad n = 1,2,3. \tag{23}$$

In the discrete setting, we must ensure that the approximation of the metrics is free–stream preserving. Having a *watertight* mesh is a necessary condition but not sufficient. For free–stream preservation, the mesh has to satisfy two additional conditions [66],

1. **Condition (F):** the projection of the discrete volume weighted contravariant bases at the faces has to be also continuous $\mathcal{J}\vec{a}^i\big|_{face}^{e_L} = \mathcal{J}\vec{a}^i\big|_{face}^{e_R}$. The symbol $\mathcal{J}$ represents the polynomial approximation of the Jacobian.

2. **Condition (V):** the approximation of the contravariant bases has to satisfy a discrete version of the metric identities (23).

The mapping function is approximated with the interpolation operator (17). However, although the mapping is represented by order N_e polynomials, the genuine order of the mapping is given by the approximation order of the faces. We highlight the construction of a *watertight* mesh that satisfies the two conditions:

- Edges: the approximation order of the edges in the mesh has to be unique, and the face functions that share an edge must reduce to the same curvilinear function at the edge. The order of an edge shared by various faces has to be $N_{\text{edge}} = \min\left(N_{f_1}, N_{f_2}, ...\right)$ for general three-dimensional non–conforming elements, two dimensional, two dimensional extruded and conforming problems, at most.

- Faces: the approximation order of the faces in the mesh has to be unique and it must be $N_f = \min\left(N_{e_L}, N_{e_R}\right)/2$ for general three–dimensional non–conforming elements, and $N_f = \min\left(N_{e_L}, N_{e_R}\right)$ for two–dimensional, two–dimensional extruded, and conforming problems, at most.

- Volume: the contravariant basis have to be computed in a curl form [67]

$$\mathcal{J}a_n^i = -\hat{\vec{x}}^i \cdot \nabla_\xi \times \mathcal{I}^N\left(\mathcal{X}_l \nabla_\xi \mathcal{X}_m\right), \quad i,n = 1,2,3, \quad (n,m,l) \text{ cyclic}. \tag{24}$$

We use the contravariant basis to transform the differential operators from physical to computational space. The divergence of a vector is

$$\nabla \cdot \vec{f} = \frac{1}{J}\nabla_\xi \cdot \left(M^T \vec{f}\right) = \frac{1}{J}\nabla_\xi \cdot \vec{\tilde{f}}, \tag{25}$$

where,

$$\vec{\tilde{f}} = \left[\tilde{f}^1 \;\; \tilde{f}^2 \;\; \tilde{f}^3\right]^T = M^T \vec{f}, \tag{26}$$

is the contravariant flux, and

$$M = [J\vec{a}^1 \;\; J\vec{a}^2 \;\; J\vec{a}^3] = [J\vec{a}^\xi \;\; J\vec{a}^\eta \;\; J\vec{a}^\zeta]. \tag{27}$$

The gradient of a scalar is transformed as [67]

$$\nabla u = \frac{1}{\mathcal{J}} M \nabla_\xi u. \tag{28}$$

Lastly we approximate the three dimensional integrals in an element by GL tensor product quadratures

$$\int_e fg \,\mathrm{d}e \approx \int_{E,N_e} \mathcal{J}FG \,\mathrm{d}E = \langle \mathcal{J}F, G\rangle_{E,N_e} = \sum_{i,j,k=0}^{N_e} w_{ijk}\mathcal{J}_{ijk}F_{ijk}G_{ijk}, \tag{29}$$

which allows us to write the discrete Gauss law as in [68]

$$\left\langle \nabla_\xi U, \tilde{F} \right\rangle_{E,N_e} = \int_{\partial E,N_e} U\tilde{F}\cdot\hat{n}\,\mathrm{d}\hat{S} - \left\langle \nabla_\xi \cdot \tilde{F}, U \right\rangle_{E,N_e}. \tag{30}$$

In (30), $\hat{n}$ is the unit outward normal vector at the reference element faces, $\mathrm{d}\hat{S}$ is the surface local integration variables ($\mathrm{d}\hat{S}^i = \mathrm{d}\xi^j\,\mathrm{d}\xi^k$ for i–oriented faces). To compute the surface integral, the two dimensional quadratures in each of the six faces that define the element are defined

$$\begin{aligned}\int_{\partial E,N_e} U\tilde{F}\cdot\hat{n}\,\mathrm{d}\hat{S} = \int_{f,N_e} U\tilde{F}^\xi \,\mathrm{d}\eta\,\mathrm{d}\zeta\Big|_{\xi=-1}^{\xi=1} + \int_{f,N_e} U\tilde{F}^\eta \,\mathrm{d}\xi\,\mathrm{d}\zeta\Big|_{\eta=-1}^{\eta=1} + \\ \int_{f,N_e} U\tilde{F}^\zeta \,\mathrm{d}\xi\,\mathrm{d}\eta\Big|_{\zeta=-1}^{\zeta=1}.\end{aligned} \tag{31}$$

Moreover, surface integrals can be written in either physical or computational space. The relation between the two spaces is

$$\mathrm{d}S^i = \left|J\vec{a}^i\right| \mathrm{d}\xi^j\,\mathrm{d}\xi^k = \mathcal{J}_f^i\,\mathrm{d}\hat{S}^i, \tag{32}$$

where we defined the face Jacobian $\mathcal{J}_f^i = \left|\mathcal{J}\vec{a}^i\right|$. An equivalent relation can be deducted for the surface flux in the reference element, $\tilde{F}\cdot\hat{n}$, as well as the physical, $\vec{F}\cdot\vec{n}$, through the relation

$$\vec{\tilde{F}}\cdot\hat{n}^i\,\mathrm{d}\hat{S}^i = \left(\mathcal{M}^T\vec{F}\right)\cdot\hat{n}^i\,\mathrm{d}\hat{S}^i = \vec{F}\cdot\left(\mathcal{M}\hat{n}^i\right)\mathrm{d}\hat{S}^i = \vec{F}\cdot\vec{n}\left|\mathcal{J}\vec{a}^i\right|\mathrm{d}\hat{S}^i = \vec{F}\cdot\vec{n}^i\,\mathrm{d}S^i. \tag{33}$$

Therefore, quadratures can be represented both in physical and computational spaces,

$$\int_{\partial E,N_e} \vec{\tilde{F}}\cdot\hat{n}\,\mathrm{d}\hat{S} = \int_{\partial e,N_e} \vec{F}\cdot\vec{n}\,\mathrm{d}S, \tag{34}$$

and the use of one form over the other depends on whether we study an isolated element (computational space) or the whole combination of elements in the mesh (physical space).

3.1 The mortar element method

Following [58], we use two projection operators $\mathcal{P}_{lh}$ and $\mathcal{P}_{hl}$ from the order N_l (low order) space to its $N_h > N_l$ (high order) counterpart and vice versa known as the *mortar method* [47]. Let $F \in \mathbb{P}^{N_l}$ and $G \in \mathbb{P}^{N_h}$ two polynomial functions. Then from [25, 33], the projection operator is designed to satisfy

$$\langle F, \mathcal{P}_{hl} G \rangle_{E,N_l} = \langle \mathcal{P}_{lh} F, G \rangle_{E,N_h} , \tag{35}$$

which is an SBP–preserving operator that satisfies the M–compatibility condition [25, 26, 33, 69]. A consequence of (35) is that the operator to augment the polynomial space differs to the opposite one, known as *restriction*. To augment the polynomial space one simply uses an interpolation operation,

$$(\mathcal{P}_{lh})_{ij} = l_j^{N_l}\left(\xi_i^{N_h}\right), \tag{36}$$

that is, we evaluate the Lagrange interpolating polynomials of the low–order N_l space at the GL nodes of the high–order N_h space. To fulfill the condition (35), the backward projection (restriction) from N_h to N_l must be

$$\mathcal{P}_{hl} = M_l^{-1} \mathcal{P}_{lh}^T M_h, \quad M_e = \text{diag}\left(w_0^{N_e}, w_1^{N_e}, ..., w_{N_e}^{N_e},\right), \tag{37}$$

where $w_j^{N_e}$ are the quadrature weights at the GL quadrature points for an element e with polynomial order N_e. As presented in Section 6 the error remains down to machine rounding error. Constructed in such way, the operators are not invertible, that is $\mathcal{P}_{hl}\mathcal{P}_{lh} \neq \mathcal{I}_h$ and $\mathcal{P}_{lh}\mathcal{P}_{hl} \neq \mathcal{I}_l$. This means that when an arbitrary polynomial $A \in \mathbb{P}^{N_l}$ from the lower space is projected to the mortar, it is not recovered by using the restriction operator,

$$\mathcal{P}_{hl}\left(\mathcal{P}_{lh}\left(A\right)\right) \neq A. \tag{38}$$

An exception to (38) is any constant function, $\mathcal{P}_{hl}\left(\mathcal{P}_{lh}\left(k\right)\right) = k$. Therefore, we can show that although the polynomials in (38) differ, the mortar is still able to keep the integral value,

$$\langle \mathcal{P}_{hl}\left(\mathcal{P}_{lh}\left(A\right)\right), 1\rangle_{E,N_l} = \langle \mathcal{P}_{lh}\left(A\right), \mathcal{P}_{lh}\left(1\right)\rangle_{E,N_h} = \langle A, \mathcal{P}_{hl}\left(\mathcal{P}_{lh}\left(1\right)\right)\rangle_{E,N_l} = \langle A, 1\rangle_{E,N_l} . \tag{39}$$

This property is useful for proving that the scheme is conservative [33].

3.2 Discontinuous Galerkin spectral element approximation of the Cahn–Hilliard equation

We now assemble the discrete version of (13). The first step is to transform (13) into the local coordinate system as described in (25) and (28), and to get the weak form of the system in the reference element E. To do that, we restrict the test functions φ_Φ, φ_W (scalar), $\vec{\varphi}_F$, $\vec{\varphi}_Q$ (vectorial), to the order N_e polynomial space,

$$\langle J\phi_t, \varphi_\Phi\rangle_E = \left\langle \nabla_\xi \cdot \left(M\tilde{f}\right), \varphi_\Phi \right\rangle_E , \tag{40a}$$

$$\left\langle J\vec{f}, \vec{\varphi}_F \right\rangle_E = \langle \mathcal{M}\nabla_\xi w, \vec{\varphi}_F\rangle_E = \left\langle \nabla_\xi w, \mathcal{M}^T\vec{\varphi}_F\right\rangle_E = \langle \nabla_\xi w, \tilde{\varphi}_F\rangle_E , \tag{40b}$$

$$\langle Jw, \varphi_W\rangle_E = \left\langle J\frac{\mathrm{d}\psi}{\mathrm{d}\phi}, \varphi_W \right\rangle_E - k\left\langle \nabla_\xi \cdot \tilde{q}, \varphi_W\right\rangle_E , \tag{40c}$$

$$\langle J\vec{q}, \vec{\varphi}_Q\rangle_E = \langle \mathcal{M}\nabla_\xi \phi, \vec{\varphi}_Q\rangle_E = \left\langle \nabla_\xi \phi, \mathcal{M}^T\vec{\varphi}_Q\right\rangle_E = \langle \nabla_\xi \phi, \tilde{\varphi}_Q\rangle_E . \tag{40d}$$

The following step is to integrate the right hand side terms that contain a ∇_ξ operator by parts, to replace the continuous functions with their polynomial approximations and to replace exact integrals by quadratures. Furthermore, we apply the discrete Gauss law (30) to (41b) and (41d).

$$\langle \mathcal{J}\Phi_t, \varphi_\Phi \rangle_{E,N_e} = \int_{\partial E,N_e} \varphi_\Phi \left(M\tilde{F}\right)^\star \cdot \hat{n}\, \mathrm{d}S_\xi - \left\langle M\tilde{F}, \nabla_\xi \varphi_\Phi \right\rangle_{E,N_e}, \tag{41a}$$

$$\left\langle \mathcal{J}\vec{F}, \vec{\varphi}_F \right\rangle_{E,N_e} = \int_{\partial E,N_e} \left((W^\star - W) \left|\mathcal{J}\vec{a}\right|^i \vec{n}^i \right) \cdot \vec{\varphi}_F \, \mathrm{d}S_\xi + \langle \nabla_\xi W, \tilde{\varphi}_F \rangle_{E,N_e}, \tag{41b}$$

$$\langle \mathcal{J}W, \varphi_W \rangle_{E,N_e} = \left\langle \mathcal{J}\frac{\mathrm{d}\Psi}{\mathrm{d}\Phi}, \varphi_W \right\rangle_{E,N_e} - k \int_{\partial E,N} \varphi_W \tilde{Q}^\star \cdot \hat{n}\, \mathrm{d}S_\xi + k \left\langle \tilde{Q}, \nabla_\xi \varphi_W \right\rangle_{E,N_e}, \tag{41c}$$

$$\left\langle \mathcal{J}\vec{Q}, \vec{\varphi}_Q \right\rangle_{E,N_e} = \int_{\partial E,N} \left((\Phi^\star - \Phi) \left|\mathcal{J}\vec{a}\right|^i \vec{n}^i \right) \cdot \vec{\varphi}_Q \, \mathrm{d}S_\xi + \langle \nabla_\xi \Phi, \tilde{\varphi}_Q \rangle_{E,N_e}, \tag{41d}$$

where index i is used with Einstein convention.
The terms with star superscript in (41) are the *numerical fluxes*, which make the flux uniquely defined at the boundaries. In this work, we use the Bassi–Rebay 1 scheme (BR1) [70], which handles non–conforming interfaces. Without loss of generality, we consider the inter–element face with orders $N_l < N_h$. The mortar method computes the scalar fluxes on the higher order element, which are then transferred to the lower order element through the restriction operator,

$$\begin{aligned}
\left((W^\star - W) \left|\mathcal{J}\vec{a}\right|^i \vec{n}^i \right)_h &= \left(\frac{W_h + \mathcal{P}_{lh}(W_l)}{2} - W_h \right) \left|\mathcal{J}\vec{a}\right|^h \vec{n}^h = \frac{1}{2} \left(\mathcal{P}_{lh}(W_l) - W_h \right) \left|\mathcal{J}\vec{a}\right|^h \vec{n}^h, \\
\left((W^\star - W) \left|\mathcal{J}\vec{a}\right|^i \vec{n}^i \right)_l &= \mathcal{P}_{hl} \left((W^\star - W) \left|\mathcal{J}\vec{a}\right|^i \vec{n}^i \right)_h, \\
\left((\Phi^\star - \Phi) \left|\mathcal{J}\vec{a}\right|^i \vec{n}^i \right)_h &= \left(\frac{\Phi_h + \mathcal{P}_{lh}(\Phi_l)}{2} - \Phi_h \right) \left|\mathcal{J}\vec{a}\right|^h \vec{n}^h = \frac{1}{2} \left(\mathcal{P}_{lh}(\Phi_l) - \Phi_h \right) \left|\mathcal{J}\vec{a}\right|^h \vec{n}^h, \\
\left((\Phi^\star - \Phi) \left|\mathcal{J}\vec{a}\right|^i \vec{n}^i \right)_l &= \mathcal{P}_{hl} \left((\Phi^\star - \Phi) \left|\mathcal{J}\vec{a}\right|^i \vec{n}^i \right)_h.
\end{aligned} \tag{42}$$

Whereas for the vector fluxes, we transform the contravariant fluxes from the lower order to the higher order element and then we take their difference (since the normal vectors are opposite) and add an interface stabilizing term,

$$\begin{aligned}
\vec{\tilde{F}}_h^\star \cdot \hat{n} &= \frac{1}{2} \left|\mathcal{J}\vec{a}\right|^h \vec{n}^h \cdot \left(\vec{F}_h + \mathcal{P}_{lh} \left(\vec{F}_l \right) \right) - \sigma \left|\mathcal{J}\vec{a}\right|^h \left(W_h - \mathcal{P}_{lh}(W_l) \right), \\
\vec{\tilde{F}}_l^\star \cdot \hat{n} &= -\mathcal{P}_{hl} \left(\vec{\tilde{F}}_h^\star \cdot \hat{n} \right), \\
\vec{\tilde{Q}}_h^\star \cdot \hat{n} &= \frac{1}{2} \left|\mathcal{J}\vec{a}\right|^h \vec{n}^h \cdot \left(\vec{Q}_h + \mathcal{P}_{lh} \left(\vec{Q}_l \right) \right) - \sigma \left|\mathcal{J}\vec{a}\right|^h \left(\Phi_h - \mathcal{P}_{lh}(\Phi_l) \right), \\
\vec{\tilde{Q}}_l^\star \cdot \hat{n} &= -\mathcal{P}_{hl} \left(\vec{\tilde{Q}}_h^\star \cdot \hat{n} \right),
\end{aligned} \tag{43}$$

In (43), $\left|\mathcal{J}\vec{a}\right|^h \vec{n}^h$ and $\left|\mathcal{J}\vec{a}\right|^l \vec{n}^l$ are the (scaled) normal vectors at the face, and they satisfy $\left|\mathcal{J}\vec{a}\right|^h \vec{n}^h = -\left|\mathcal{J}\vec{a}\right|^l \vec{n}^l = -\mathcal{P}_{lh}\left(\left|\mathcal{J}\vec{a}\right|^l \vec{n}^l\right)$ (**Condition (F)**).
For Neumann boundary conditions, we use the adjacent element interior value to compute the gradients in (41b) and (41d),

$$W^\star = W\big|_{\partial e}, \quad \Phi^\star = \Phi\big|_{\partial e}, \tag{44}$$

and we directly impose Neumann boundary values for divergence weak forms (41a) and (41c),

$$\left(M\vec{F}\right)^\star \cdot \vec{n} = 0, \quad \vec{Q}^\star \cdot \vec{n} = \frac{\mathrm{d}G\left(\Phi\big|_{\partial e}\right)}{\mathrm{d}\phi} = \beta, \tag{45}$$

where β is kept constant to $\beta = 0$ for all the numerical experiments.
Note that we have presented the physical interface fluxes, rather than the contravariant fluxes, so that we can relate in a straightforward manner the interface values shared by two elements (recall that the same physical flux yields different contravariant flux values, as it depends on each element geometry).

3.3 Temporal Discretization

The Cahn–Hilliard equation incorporates higher order derivatives which make the system stiff. To combat the stiffness, a first order IMplicit–EXplicit (IMEX) scheme has been employed. Throughout this work a constant timestep Δt has been used. Following the theory developed in [8, 71], the scheme takes the following form:

$$\phi^{n+1} - \phi^n = \Delta t \nabla \cdot \left(M \nabla \left(\frac{d\psi(\phi^n)}{d\phi} + S_0 \left(\phi^{n+1} - \phi^n \right) - k \nabla^2 \left(K_0 \phi^{n+1} + (1 - K_0) \phi^n \right) \right) \right). \quad (46)$$

The notation follows that n will be used as a superscript to denote state values at $t_n = n\Delta t$. The chemical free–energy $\psi(\phi)$ is non–linear and will be treated explicitly as denoted in (46). The treatment of the interface energy term $k\nabla^2\phi$ is controlled by the K_0 parameter. For $K_0 = 0$ there is a fully explicit time marching, for $K_0 = 1/2$ we get the Crank–Nicolson scheme, whereas for $K_0 = 1$ it is fully implicit with a backwards Euler scheme. For the numerical tests of Sec. 7, the fully implicit treatment has been utilized. The second term of the right hand side of (46) is an additional numerical stabilization according to [71]. It has been shown in [8] that a value of $S_0 = 1$ the system is stable for $\Phi \in [-1, 1]$ and thus has been chosen for the following numerical tests.

Applying the scheme (46) to the discretized scheme of (41), the fully discrete discontinuous Galerkin approximation is obtained

$$\left\langle \mathcal{J} \frac{\Phi^{n+1} - \Phi^n}{\Delta t}, \varphi_\Phi \right\rangle_{E,N} = \int_{\partial E,N} \varphi_\Phi \left(M \tilde{F} \right)^{\star,\theta} \cdot \hat{n} \, \mathrm{d}S_\xi - \left\langle M \tilde{F}^\theta, \nabla_\xi \varphi_\Phi \right\rangle_{E,N}, \quad (47a)$$

$$\left\langle \mathcal{J} \vec{F}^\theta, \vec{\varphi}_F \right\rangle_{E,N} = \int_{\partial E,N} \left(W^{\star,\theta} - W^\theta \right) \tilde{\varphi}_F \cdot \hat{n} \, \mathrm{d}S_\xi + \left\langle \nabla_\xi W^\theta, \tilde{\varphi}_F \right\rangle_{E,N}, \quad (47b)$$

$$\left\langle \mathcal{J} W^\theta, \varphi_W \right\rangle_{E,N} = \left\langle \left(\frac{d\Psi}{d\Phi} \right)^n + S_0 \left(\Phi^{n+1} - \Phi n \right), \mathcal{J} \phi_w \right\rangle_{E,N} - k \int_{\partial E,N} \varphi_W \tilde{Q}^{\star,n+1} \cdot \hat{n} \, \mathrm{d}S_\xi + k \left\langle \tilde{Q}^{n+1}, \nabla_\xi \varphi_W \right\rangle_{E,N}, \quad (47c)$$

$$\left\langle \mathcal{J} \vec{Q}^{n+1}, \vec{\varphi}_Q \right\rangle_{E,N} = \int_{\partial E,N} \left(\Phi^{\star,n+1} - \Phi^{n+1} \right) \tilde{\varphi}_Q \cdot \hat{n} \, \mathrm{d}S_\xi + \left\langle \nabla_\xi \Phi^{n+1}, \tilde{\varphi}_Q \right\rangle_{E,N}. \quad (47d)$$

The superscript θ has been used for variables (e.g. $\vec{F}^\theta$ or W^θ) that are not directly evaluated at t_n or t_{n+1} with the IMEX strategy, but on a combination of those depending on the different terms involved in (47c).

4 Heuristic p–adaptation methodology

In this section we describe a methodology to automatically adapt the polynomial order for solutions of the Cahn–Hilliard equation. This methodology exploits the characteristics of the introduced numerical method which permit the use of different order of accuracy in each element. The indicator identifies the elements that contain at least one point within the interface region. The latter has been defined as the region where the phase field parameter ϕ ranges from $-0.9 < \phi < 0.9$ and is a choice which is usual in diffuse interface literature [72]. If an element is marked for refinement, then it is refined to a user specified polynomial order in all directions. The adaptation process is presented in Algorithm 1.

To enhance the robustness of the method, and since the movement of the interface cannot be predicted when solving the Cahn–Hilliard equation, a buffer region of refined elements is applied to all the neighboring elements of those that contain part of the interface, as presented

in Algorithm 1. On the contrary, the elements that are not marked, are coarsened also to a user specified level. In this work we have used a coarse level of $\mathrm{N_{Coarse}} = 2$. Simulations have also been conducted with a coarse polynomial order of $N_{\mathrm{Coarse}} = 1$ and we noticed that for the meshes used, the $N_{\mathrm{Coarse}} = 1$ is not sufficient to adequately resolve the solution and thus this option has been omitted.

In order to avoid steep changes in the polynomial order across the elements, two different jump restrictions have been tested to bridge the fine and coarse element levels. In both cases, the polynomial order of the elements in fine level remains intact, and modifications take place in the neighboring coarse level elements. The first criterion restricts the polynomial order jump between two subsequent elements not to be greater than unity and from this point on, we will refer to that as the $N-1$ condition. The other criterion is less restrictive and dictates that for two adjacent elements with $N_l < N_h$, the N_l will take a value of $N_l \geq 2N_{\mathrm{h}}/3$. The latter will be referred to as the $2N/3$ criterion for the rest of this text. These conditions have been derived from the work of [63].

One of the most important benefits of this adaptation approach, is that the computational cost is kept to a minimum. The method, calculates the new polynomial order for each element and performs the interpolation or projection from the previous to the following polynomial orders. In this case we use the operator $\mathcal{P}_{PN}$ (35) to perform an L^2 projection from order P to order N as presented in [64],

$$\left(\Phi^N\right) = \mathcal{P}_{PN}\left(\Phi^P\right). \tag{48}$$

For interpolation we use the operator (36) and for restriction the operator (37).
This method is effective since the solutions of the Cahn-Hilliard equation typically involve a single scale, that of the interface. This flexibility also stems from the use of a higher order method such as the discontinuous Galerkin. Due to the higher order approximation, the element size can be substantially larger compared to classic lower order methods and thus the issues for the interface–marking approach detailed in [51] such as the refinement or coarsening of the area around the interface are bypassed. In addition, for phase field simulations, the target is to have a sufficient number of solution points within the interface in order to capture it appropriately. For other discretization methods, this number is defined to be at least 5 points within this region [73], while some researchers prefer a number close to 10 [74,75]. In this work, we use a varying number of points within the interface that ranges from 3 to 30 to assess the convergence characteristics of the scheme.

Therefore, since we have a–priori knowledge of the element size, the interface width and the desired value for points within the interface, it is a very straightforward and low–cost method which does not require a rigorous tuning of parameters such as in various marker methods used for AMR [38, 44, 51]. Thus, as presented in Algorithm 1, the user specifies a value for the maximum and the minimum polynomial order to be used, the iteration interval upon which the adaptation will take place and lastly the criterion for the polynomial order jump.

Algorithm 1 adaptation algorithm used for polynomial adaptation of the Cahn-Hilliard equation

$N_F \leftarrow N_{Fine}$ ▷ Fine polynomial orders specified by the user
$N_C \leftarrow N_{Coarse}$ ▷ Coarse polynomial orders specified by the user
$f_{adpt} \leftarrow$ adaptation Frequency ▷ adaptation frequency specified by the user
Jump $\leftarrow 2N/3$ or $N-1$ ▷ User specifies the desired jump condition
iter $\leftarrow$ Iteration number
procedure ADAPTATION(Element,N_f,N_c,f_{adpt},iter,Jump)
    if $\mathrm{mod}\left(\frac{\mathrm{iter}}{f_{\mathrm{adpt}}}\right) = 0$ or iter $= 1$ **then**
        for $i \leftarrow 1$,number_of_elements **do**
            if any $|\mathrm{Element(i)}.\phi(\mathrm{x},\mathrm{y},\mathrm{z}))| \leq 0.9$ **then**
                Element(i).Polynomial_Order $= N_F$;
            else
                Element(i).Polynomial_Order $= N_C$;
            end if
        end for
        for $i \leftarrow 1$,number_of_elements **do**
            if Element.Polynomial_Order $= N_C$ **then**
                if $|\mathrm{Element.Neighbor}.\phi(\mathrm{x},\mathrm{y},\mathrm{z}))| \leq 0.9$ **then** ▷ Apply N_{Fine} to buffer zone
                    Element.Polynomial_Order $= N_F$;
                end if
            end if
        end for
        ApplyJump(Element) ▷ Application of the Jump condition
        InterpolateToNewPolynomialOrders(Element) ▷ Process of Eq (48)
    end if
end procedure

5 Stability analysis

In this work we focus on the p–non–conforming extension of the stability analysis methodology presented in [8]. The initial steps of the analysis are briefly described and a more detailed description can be found in [8]. This analysis focuses specifically on the treatment of the inter–element boundary terms for elements with different polynomial orders to prove that the scheme is free–energy stable. The derivation of the following steps is given in detail in Appendix A.1.

1. Take the time derivative of (41d).
2. Replace the test functions $\varphi_\Phi = W$, $\vec{\varphi}_F = \vec{F}$, $\varphi_W = \Phi_t$, $\vec{\varphi}_Q = \vec{Q}$.
3. Both (41d) and (41c) share the term $\left\langle \nabla_\xi \Phi_t, \tilde{\vec{Q}} \right\rangle_{E,N_e}$, which is replaced from (41d) in (41c).
4. Replace the term $\left\langle \nabla_\xi W, \tilde{\vec{F}} \right\rangle_{E,N_e}$, from (41b) in (41a).
5. Both (41a) and (41c) have the term$\langle \mathcal{J}\Phi_t, W \rangle_{E,N_e}$, which is replaced from (41c) in (41a).
6. The time derivative of the free–energy is identified from the terms $\left\langle \mathcal{J}\frac{\mathrm{d}\Psi}{\mathrm{d}\Phi}, \Phi_t \right\rangle_{E,N_e} + k \left\langle \mathcal{J}\vec{Q}_t, \vec{Q} \right\rangle_{E,N_e} = \frac{\mathrm{d}}{\mathrm{d}t} \left\langle \mathcal{J}\left(\Psi + \frac{k}{2}\|Q\|^2\right) \right\rangle_{E,N_e} = \mathcal{F}_t^{E,N_e}$.

This results in the following equation for an element e,

$$\mathcal{F}_t^{E,N_e} - \int_{\partial E,N_e} \left(k \left((\Phi_t^\star - \Phi_t)\, \tilde{\vec{Q}} + \Phi_t \tilde{\vec{Q}}^\star \right) + (W^\star - W)\, \tilde{\vec{F}} + W \tilde{\vec{F}}^\star \right) \cdot \hat{n} \mathrm{d}\hat{S} = - \left\langle \mathcal{J} M \vec{F}, \vec{F} \right\rangle_{E,N_e}. \tag{49}$$

Now we sum (49) for all the mesh elements,

$$\sum_e \mathcal{F}_t^{E,N_e} + \mathrm{IBT} + \mathrm{PBT} = - \sum_e \left\langle \mathcal{J} M \vec{F}, \vec{F} \right\rangle_{E,N_e}, \tag{50}$$

where IBT and PBT are the interior and physical boundary terms. As derived in [8], the PBT represent the surface free–energy,

$$\mathrm{PBT} = \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{\partial E,N_e} \Phi_t \frac{\mathrm{d}G}{\mathrm{d}\Phi} \mathrm{d}S = \frac{\mathrm{d}}{\mathrm{d}t} \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{\partial E,N_e} G \mathrm{d}S = \mathcal{F}_{S,t}, \tag{51}$$

thus, the equation for the total free–energy $\bar{\mathcal{F}} = \sum_e \mathcal{F}^{E,N_e} + \sum_{\substack{\text{boundary} \\ \text{faces}}} \int_{\partial E,N_e} G \mathrm{d}S$ is

$$\bar{\mathcal{F}}_t + \mathrm{IBT} = - \sum_e \left\langle \mathcal{J} M \vec{F}, \vec{F} \right\rangle_{E,N_e}. \tag{52}$$

A stable approximation has IBT $\geqslant 0$. It is in the interior boundary terms where the stability of the inter–element coupling through the mortar element method is assessed. Each interior face has the contribution of the two adjacent elements with orders $N_l < N_h$,

$$\begin{aligned}
\mathrm{IBT} = &- \sum_{\substack{\text{interior} \\ \text{faces}}} \int_{f,N_l} \left(k \left(\left((\Phi_t^\star - \Phi_t) \left| \mathcal{J}\vec{a} \right| \vec{n} \right)_l \cdot \vec{Q}_l + \Phi_{l,t} \tilde{\vec{Q}}_l^\star \cdot \hat{n} \right) \right) \mathrm{d}\hat{S} \\
&- \sum_{\substack{\text{interior} \\ \text{faces}}} \int_{f,N_l} \left(\left((W^\star - W) \left| \mathcal{J}\vec{a} \right| \vec{n} \right)_l \cdot \vec{F}_l + W_l \tilde{\vec{F}}_l^\star \cdot \hat{n} \right) \mathrm{d}\hat{S} \\
&- \sum_{\substack{\text{interior} \\ \text{faces}}} \int_{f,N_h} \left(k \left(\left((\Phi_t^\star - \Phi_t) \left| \mathcal{J}\vec{a} \right| \vec{n} \right)_h \cdot \vec{Q}_h + \Phi_{h,t} \tilde{\vec{Q}}_h^\star \cdot \hat{n} \right) \right) \mathrm{d}\hat{S} \\
&- \sum_{\substack{\text{interior} \\ \text{faces}}} \int_{f,N_h} \left(\left((W^\star - W) \left| \mathcal{J}\vec{a} \right| \vec{n} \right)_h \cdot \vec{F}_h + W_h \tilde{\vec{F}}_h^\star \cdot \hat{n} \right) \mathrm{d}\hat{S}.
\end{aligned} \tag{53}$$

For simplicity, we first study the second and fourth terms of (53) (the part that involves W and $\vec{F}$). We replace the inter–element fluxes (42) and (43),

$$\begin{aligned}
\left(\left(W^\star - W\right)\left|\mathcal{J}\vec{a}\right|\vec{n}\right)_h \cdot \vec{F}_h + W_h \vec{\tilde{F}}_h^\star \cdot \hat{n} &= \frac{1}{2}\left(\mathcal{P}_{lh}\left(W_l\right) - W_h\right)\left|\mathcal{J}\vec{a}\right|^h \vec{n}^h \cdot \vec{F}_h \qquad (54a)\\
&\quad + W_h\left(\frac{1}{2}|\mathcal{J}\vec{a}|^h\vec{n}^h \cdot \left(\vec{F}_h + \mathcal{P}_{lh}\left(\vec{F}_l\right)\right) - \sigma\left|\mathcal{J}\vec{a}\right|^h\left(W_h - \mathcal{P}_{lh}\left(W_l\right)\right)\right)\\
&= \frac{1}{2}\left|\mathcal{J}\vec{a}\right|^h\left(\mathcal{P}_{lh}\left(W_l\right)\vec{F}_h + W_h\mathcal{P}_{lh}\left(\vec{F}_l\right)\right)\cdot\vec{n}^h - \sigma\left|\mathcal{J}\vec{a}\right|^h W_h\left(W_h - \mathcal{P}_{lh}\left(W_l\right)\right),\\
\left(\left(W^\star - W\right)\left|\mathcal{J}\vec{a}\right|\vec{n}\right)_l \cdot \vec{F}_l + W_l \vec{\tilde{F}}_l^\star \cdot \hat{n} &= \frac{1}{2}\mathcal{P}_{hl}\left(\left(\mathcal{P}_{lh}\left(W_l\right) - W_h\right)\left|\mathcal{J}\vec{a}\right|^h \vec{n}^h\right)\cdot\vec{F}_l \qquad (54b)\\
&\quad - W_l\mathcal{P}_{hl}\left(\frac{1}{2}|\mathcal{J}\vec{a}|^h\vec{n}^h \cdot \left(\vec{F}_h + \mathcal{P}_{lh}\left(\vec{F}_l\right)\right) - \sigma\left|\mathcal{J}\vec{a}\right|^h\left(W_h - \mathcal{P}_{lh}\left(W_l\right)\right)\right),
\end{aligned}$$

and we apply property (35) to the second term in (53),

$$\begin{aligned}
&\int_{f,N_l}\left(\frac{1}{2}\mathcal{P}_{hl}\left(\left(\mathcal{P}_{lh}\left(W_l\right) - W_h\right)\left|\mathcal{J}\vec{a}\right|^h\vec{n}^h\right)\cdot\vec{F}_l\right.\\
&\left. - W_l\mathcal{P}_{hl}\left(\frac{1}{2}|\mathcal{J}\vec{a}|^h\vec{n}^h\cdot\left(\vec{F}_h + \mathcal{P}_{lh}\left(\vec{F}_l\right)\right) - \sigma\left|\mathcal{J}\vec{a}\right|^h\left(W_h - \mathcal{P}_{lh}\left(W_l\right)\right)\right)\right)\mathrm{d}\hat{S}\\
&= \int_{f,N_h}\left(\frac{1}{2}\left(\left(\mathcal{P}_{lh}\left(W_l\right) - W_h\right)\left|\mathcal{J}\vec{a}\right|^h\vec{n}^h\right)\cdot\mathcal{P}_{lh}\left(\vec{F}_l\right)\right.\\
&\left. - \mathcal{P}_{lh}\left(W_l\right)\left(\frac{1}{2}|\mathcal{J}\vec{a}|^h\vec{n}^h\cdot\left(\vec{F}_h + \mathcal{P}_{lh}\left(\vec{F}_l\right)\right) - \sigma\left|\mathcal{J}\vec{a}\right|^h\left(W_h - \mathcal{P}_{lh}\left(W_l\right)\right)\right)\right)\mathrm{d}\hat{S}\\
&= -\int_{f,N_h}\left(\frac{1}{2}\left|\mathcal{J}\vec{a}\right|^h\left(W_h\mathcal{P}_{lh}\left(\vec{F}_l\right) + \mathcal{P}_{lh}\left(W_l\right)\vec{F}_h\right)\cdot\vec{n}^h - \sigma\left|\mathcal{J}\vec{a}\right|^h\mathcal{P}_{lh}\left(W_l\right)\left(W_h - \mathcal{P}_{lh}\left(W_l\right)\right)\right)\mathrm{d}\hat{S}
\end{aligned} \qquad (55)$$

We add the (negative) contributions from the fourth term in (53) and the transformed lower order integral of (55),

$$\begin{aligned}
&-\int_{f,N_h}\left(\frac{1}{2}\left|\mathcal{J}\vec{a}\right|^h\left(\mathcal{P}_{lh}\left(W_l\right)\vec{F}_h + W_h\mathcal{P}_{lh}\left(\vec{F}_l\right)\right)\cdot\vec{n}^h - \sigma\left|\mathcal{J}\vec{a}\right|^h W_h\left(W_h - \mathcal{P}_{lh}\left(W_l\right)\right)\right)\mathrm{d}\hat{S}\\
&+\int_{f,N_h}\left(\frac{1}{2}\left|\mathcal{J}\vec{a}\right|^h\left(W_h\mathcal{P}_{lh}\left(\vec{F}_l\right) + \mathcal{P}_{lh}\left(W_l\right)\vec{F}_h\right)\cdot\vec{n}^h - \sigma\left|\mathcal{J}\vec{a}\right|^h\mathcal{P}_{lh}\left(W_l\right)\left(W_h - \mathcal{P}_{lh}\left(W_l\right)\right)\right)\mathrm{d}\hat{S}\\
&= \sigma\int_{f,N_h}\left|\mathcal{J}\vec{a}\right|^h\left(W_h\left(W_h - \mathcal{P}_{lh}\left(W_l\right)\right) - \mathcal{P}_{lh}\left(W_l\right)\left(W_h - \mathcal{P}_{lh}\left(W_l\right)\right)\right)\mathrm{d}\hat{S}\\
&= \sigma\int_{f,N_h}\left|\mathcal{J}\vec{a}\right|^h\left(W_h - \mathcal{P}_{lh}\left(W_l\right)\right)^2\mathrm{d}\hat{S} \geqslant 0.
\end{aligned} \qquad (56)$$

We proceed similarly with the first and third integrals of (53), for which we use the same numerical fluxes,

$$\begin{aligned}
&-\int_{f,N_l}\left(k\left(\left(\left(\Phi_t^\star - \Phi_t\right)\left|\mathcal{J}\vec{a}\right|\vec{n}\right)_l\cdot\vec{Q}_l + \Phi_{l,t}\vec{\tilde{Q}}_l^\star\cdot\hat{n}\right)\right)\mathrm{d}\hat{S}\\
&-\int_{f,N_h}\left(k\left(\left(\left(\Phi_t^\star - \Phi_t\right)\left|\mathcal{J}\vec{a}\right|\vec{n}\right)_h\cdot\vec{Q}_h + \Phi_{h,t}\vec{\tilde{Q}}_h^\star\cdot\hat{n}\right)\right)\mathrm{d}\hat{S}\\
&= k\sigma\int_{f,N_h}\left|\mathcal{J}\vec{a}\right|^h\left(\Phi_h\Phi_{h,t} - \Phi_h\mathcal{P}_{lh}\left(\Phi_{l,t}\right) - \Phi_{h,t}\mathcal{P}_{lh}\left(\Phi_l\right) + \mathcal{P}_{lh}\left(\Phi_l\right)P_{lh}\left(\Phi_{l,t}\right)\right)\mathrm{d}\hat{S}\\
&= k\sigma\int_{f,N_h}\left|\mathcal{J}\vec{a}\right|^h\left(\Phi_h - \mathcal{P}_{lh}\left(\Phi_l\right)\right)\left(\Phi_h - \mathcal{P}_{lh}\left(\Phi_l\right)\right)_t\mathrm{d}\hat{S} = \frac{\sigma k}{2}\frac{\mathrm{d}}{\mathrm{d}t}\int_{\partial E,N_h}\left|\mathcal{J}\vec{a}\right|^h\left(\Phi_h - \mathcal{P}_{lh}\left(\Phi_l\right)\right)^2\mathrm{d}\hat{S},
\end{aligned} \qquad (57)$$

which gives,

$$\text{IBT} = \sum_{\substack{\text{interior}\\ \text{faces}}} \sigma \int_{f,N_h} |\mathcal{J}\vec{a}|^h \left(W_h - \mathcal{P}_{lh}\left(W_l\right)\right)^2 \mathrm{d}\hat{S} + \frac{\sigma k}{2}\frac{\mathrm{d}}{\mathrm{d}t}\int_{f,N_h} |\mathcal{J}\vec{a}|^h \left(\Phi_h - \mathcal{P}_{lh}\left(\Phi_l\right)\right)^2 \mathrm{d}\hat{S}, \quad (58)$$

Finally, following [8] we define an augmented free–energy $\bar{\mathcal{F}}^\sigma$ that includes the interface penalization,

$$\bar{\mathcal{F}}^\sigma = \sum_e \mathcal{F}^{E,N_e} + \sum_{\substack{\text{interior}\\ \text{faces}}} \frac{\sigma k}{2}\int_{f,N_h} |\mathcal{J}\vec{a}|^h \left(\Phi_h - \mathcal{P}_{lh}\left(\Phi_l\right)\right)^2 \mathrm{d}\hat{S} + \sum_{\substack{\text{boundary}\\ \text{faces}}} \int_{f,N_e} G\mathrm{d}S, \quad (59)$$

and that satisfies the free–energy equation,

$$\bar{\mathcal{F}}_t^\sigma = -\sum_e \left\langle \mathcal{J}M\vec{F}, \vec{F}\right\rangle_{E,N_e} - \sum_{\substack{\text{interior}\\ \text{faces}}} \sigma \int_{f,N_h} |\mathcal{J}\vec{a}|^h \left(W_h - \mathcal{P}_{lh}\left(W_l\right)\right)^2 \mathrm{d}\hat{S} \leqslant 0. \quad (60)$$

The physical dissipation and the numerical dissipation are both responsible for decreases in the free–energy. The numerical interface penalization is not needed for stability, hence a valid scheme can have $\sigma = 0$, but having $\sigma > 0$ usually enhances the accuracy of the solutions [8, 76].

5.1 Stability analysis of the non–conforming fully discrete system using the IMEX integrator

To begin the stability analysis for the fully discrete system the following steps are carried out in accordance to [8]. The analytical relations can be found therein, as we will mainly focus on the effect of having p–non–conforming elements which are connected through the mortar method as described in Section 3.1. The initial steps are presented in Appendix A.2. Briefly the steps are:

1. We evaluate (47d) at t_{n+1} and t_n and subtract them. Then divide by Δt and set $\vec{\phi}_Q = \vec{Q}^{n+1}$.
2. Then we set $\phi_w = \left(\Phi^{n+1} - \Phi^n\right)/\Delta t$ in (47c) and subtract it from the modified (47d) from Step 1.
3. The following step is to combine (47a) and (47b). First we set $\vec{\phi}_F = M\vec{F}^\theta$ in (47b) and $\phi_\Phi = W^\theta$ in (47a).
4. We then subtract the equations derived in Steps 2 and 3 and multiply by Δt .
5. The following step is to express the chemical free potential $\psi\left(\phi\right)$ through a Taylor expansion using the definition from (9).
6. Through some manipulation, which is described in detail in [8], we end up to an expression for the discrete volumetric free–energy $\mathcal{F}_v^{n,E,N}$ within each element, which is defined as $\mathcal{F}_v^{n,E,N} = \left\langle \mathcal{J}\left(\Psi^n + \frac{1}{2}k\vec{Q}_n \cdot \vec{Q}_n\right), 1\right\rangle$.
7. The last step is to sum over all the elements within the domain Ω.

Following the aforementioned steps we derive (61) for the discrete volumetric free energy:

$$\mathcal{F}_v^{n+1,N} - \mathcal{F}_v^{n,N} = -\Delta t \sum_e \left\langle \mathcal{J}\vec{F}^\theta, M\vec{F}^\theta\right\rangle_{E,N} + \text{IBT} + \text{PBT} + \sum_e \text{diss}_{\text{IMEX}}^{E,N}, \quad (61)$$

where IBT represents the inter-element coupling terms of non-conforming elements, with approximation orders N_l and N_h respectively, as defined in (62). The terms PBT arise from the exterior boundaries (66) and the application of boundary conditions. The last term is the numerical dissipation of the IMEX scheme, which is negative, and there is a detailed proof in [8]

about its contribution in the decrease of the free energy. As the non–conformity of the domain does not affect or alter that term, it will not be presented analytically. Lastly, the first term of the right hand side of (61) will always be negative and represents the physical dissipation arising from the interior of each element.
Following the aforementioned steps, the interior boundary terms can be expressed as in (62). This is the equivalent version of (53) from the continuous stability analysis:

$$\begin{aligned}
\mathrm{IBT} = &- \sum_{\substack{\text{interior}\\ \text{faces}}} \int_{f,N_l} k \left(\left(\left(\Delta\Phi^{\star} - \Delta\Phi \right) \left| \mathcal{J}\vec{a} \right| \vec{n} \right)_l \cdot \vec{Q}_l^{n+1} + \Delta\Phi_l \vec{\tilde{Q}}_l^{\star,n+1} \cdot \hat{n} \right) \mathrm{d}\hat{S} \\
&- \sum_{\substack{\text{interior}\\ \text{faces}}} \int_{f,N_l} \left(\left(\left(W^{\theta,\star} - W^{\theta} \right) \left| \mathcal{J}\vec{a} \right| \vec{n} \right)_l \cdot \vec{F}_l^{\theta} + W_l^{\theta} \vec{\tilde{F}}_l^{\star,\theta} \cdot \hat{n} \right) \mathrm{d}\hat{S} \\
&- \sum_{\substack{\text{interior}\\ \text{faces}}} \int_{f,N_h} k \left(\left(\left(\Delta\Phi^{\star} - \Delta\Phi \right) \left| \mathcal{J}\vec{a} \right| \vec{n} \right)_h \cdot \vec{Q}_h^{n+1} + \Delta\Phi_h \vec{\tilde{Q}}_h^{\star,n+1} \cdot \hat{n} \right) \mathrm{d}\hat{S} \\
&- \sum_{\substack{\text{interior}\\ \text{faces}}} \int_{f,N_h} \left(\left(\left(W^{\theta,\star} - W^{\theta} \right) \left| \mathcal{J}\vec{a} \right| \vec{n} \right)_h \cdot \vec{F}_h^{\theta} + W_h^{\theta} \vec{\tilde{F}}_h^{\star,\theta} \cdot \hat{n} \right) \mathrm{d}\hat{S}.
\end{aligned} \tag{62}$$

where $\Delta\Phi = \Phi^{\mathrm{n+1}} - \Phi^{\mathrm{n}}$. The second and fourth terms from (62), and following the same procedure as in (56), are equal to:

$$\begin{aligned}
&- \sum_{\substack{\text{interior}\\ \text{faces}}} \int_{f,N_l} \left(\left(\left(W^{\star,\theta} - W^{\theta} \right) \left| \mathcal{J}\vec{a} \right| \vec{n} \right)_l \cdot \vec{F}_l^{\theta} + W_l^{\theta} \vec{\tilde{F}}_l^{\star,\theta} \cdot \hat{n} \right) \mathrm{d}\hat{S} \\
&- \sum_{\substack{\text{interior}\\ \text{faces}}} \int_{f,N_h} \left(\left(\left(W^{\star,\theta} - W^{\theta} \right) \left| \mathcal{J}\vec{a} \right| \vec{n} \right)_h \cdot \vec{F}_h^{\theta} + W_h^{\theta} \vec{\tilde{F}}_h^{\star,\theta} \cdot \hat{n} \right) \mathrm{d}\hat{S} \\
= &- \sum_{\substack{\text{interior}\\ \text{faces}}} \sigma \int_{f,N_h} \left| \mathcal{J}\vec{a} \right|^h \left(W_h^{\theta} - \mathcal{P}_{lh} \left(W_l^{\theta} \right) \right)^2 \mathrm{d}\hat{S}.
\end{aligned} \tag{63}$$

The first and third terms of (62), following the process of (57), are equal to:

$$\begin{aligned}
&- \sum_{\substack{\text{interior}\\ \text{faces}}} \int_{f,N_l} k \left(\left(\left(\Delta\Phi^{\star} - \Delta\Phi \right) \left| \mathcal{J}\vec{a} \right| \vec{n} \right)_l \cdot \vec{Q}_l^{n+1} + \Delta\Phi_l \vec{\tilde{Q}}_l^{\star,n+1} \cdot \hat{n} \right) \mathrm{d}\hat{S} \\
&- \sum_{\substack{\text{interior}\\ \text{faces}}} \int_{f,N_h} k \left(\left(\left(\Delta\Phi^{\star} - \Delta\Phi \right) \left| \mathcal{J}\vec{a} \right| \vec{n} \right)_h \cdot \vec{Q}_h^{n+1} + \Delta\Phi_h \vec{\tilde{Q}}_h^{\star,n+1} \cdot \hat{n} \right) \mathrm{d}\hat{S} \\
= &- \sum_{\substack{\text{interior}\\ \text{faces}}} k\sigma \int_{f,N_h} \left| \mathcal{J}\vec{a} \right|^h \left(\Phi_h^{n+1} - \mathcal{P}_{lh} \left(\Phi_l^{n+1} \right) \right) \left(\Delta\Phi_h - \mathcal{P}_{lh} \left(\Delta\Phi_l \right) \right) \mathrm{d}\hat{S}.
\end{aligned} \tag{64}$$

We manipulate (64) by adding and subtracting Φ^n and we arrange it to get

$$\begin{aligned}
= &- \sum_{\substack{\text{interior}\\ \text{faces}}} k\sigma \int_{f,N_h} \left| \mathcal{J}\vec{a} \right|^h \left(\frac{\Phi_h^{n+1} - \Phi_h^n}{2} + \frac{\Phi_h^{n+1} + \Phi_h^n}{2} - \frac{\mathcal{P}_{lh} \left(\Phi_l^{n+1} \right) - \mathcal{P}_{lh} \left(\Phi_l^n \right)}{2} \right. \\
&\left. - \frac{\mathcal{P}_{lh} \left(\Phi_l^{n+1} \right) + \mathcal{P}_{lh} \left(\Phi_l^n \right)}{2} \right) \left(\Delta\Phi_h - \mathcal{P}_{lh} \left(\Delta\Phi_l \right) \right) \mathrm{d}\hat{S} \\
= &- \sum_{\substack{\text{interior}\\ \text{faces}}} \frac{k\sigma}{2} \int_{f,N_h} \left| \mathcal{J}\vec{a} \right|^h \left(\left(\Phi_h^{n+1} - \mathcal{P}_{lh} \left(\Phi_l^{n+1} \right) \right)^2 - \left(\Phi_h^n - \mathcal{P}_{lh} \left(\Phi_l^n \right) \right)^2 + \left(\Delta\Phi_h - \mathcal{P}_{lh} \left(\Delta\Phi_l \right) \right)^2 \right) \mathrm{d}\hat{S}.
\end{aligned} \tag{65}$$

The boundary terms remain unchanged because we use same approximation in the interior as the exterior for every boundary:

$$\text{PBT} = \sum_{\substack{\text{boundary}\\ \text{faces}}} \int_{\partial e,N} \left(\Phi^{n+1} - \Phi^n\right) \beta \mathrm{d}S = \sum_{\substack{\text{boundary}\\ \text{faces}}} \int_{\partial e,N} \left(G\left(\Phi^{n+1}\right) - G\left(\Phi^n\right)\right) \mathrm{d}S. \tag{66}$$

Gathering the results of (63), (65) and (66) and substituting in (61), we derive (67) for the volumetric free–energy. We define an augmented free–energy $\mathcal{F}^{n,N,\sigma}$

$$\begin{aligned}
\mathcal{F}_v^{n+1,N,\sigma} - \mathcal{F}_v^{n,N,\sigma} - & \sum_{\substack{\text{boundary}\\ \text{faces}}} \int_{\partial e \cap \partial\Omega,N} \left(G\left(\Phi^{n+1}\right) - G\left(\Phi^n\right)\right) \mathrm{d}S \\
& + \sum_{\substack{\text{interior}\\ \text{faces}}} \frac{k\sigma}{2} \int_{f,N_h} |\mathcal{J}\vec{a}|^h \left(\left(\Phi_h^{n+1} - \mathcal{P}_{lh}\left(\Phi_l^{n+1}\right)\right)^2 - \left(\Phi_h^n - \mathcal{P}_{lh}\left(\Phi_l^n\right)\right)^2\right) \mathrm{d}\hat{S} \\
= \mathcal{F}^{n+1,N} - \mathcal{F}^{n,N} = & -\Delta t \sum_e \left\langle \mathcal{J}\vec{F}^\theta, M\vec{F}^\theta \right\rangle_{E,N} + \sum_e \mathrm{diss}_{\text{IMEX}}^{E,N} \\
& - \sum_{\substack{\text{interior}\\ \text{faces}}} \sigma \int_{f,N_h} |\mathcal{J}\vec{a}|^h \left(W_h^\theta - \mathcal{P}_{lh}\left(W_l^\theta\right)\right)^2 \mathrm{d}\hat{S} \\
& - \sum_{\substack{\text{interior}\\ \text{faces}}} \frac{k\sigma}{2} \int_{f,N_h} |\mathcal{J}\vec{a}|^h \left(\Delta\Phi_h - \mathcal{P}_{lh}\left(\Delta\Phi_l\right)\right)^2 \mathrm{d}\hat{S}.
\end{aligned} \tag{67}$$

Equation (67) confirms that the free–energy remains bounded in the discrete time setting, as the four terms on the right hand side of (67) are dissipative.

6 Numerical Verification

The goal of the following tests is to numerically validate the code and showcase the primary conservation characteristics, and that the scheme is freestream preserving for general 3D p–non–conforming meshes. For that purpose we make use of the mesh presented in [77]. The mesh is a 4×4 cube defined in $\Omega = [0,1]^3$ with all boundaries being treated with periodic boundary conditions. Initially we construct a cartesian mesh and apply a transformation to the space variables $\vec{\chi} = (\chi_1, \chi_2, \chi_3)^T$ to create the curved mesh in physical space $\vec{x}$. The transformation function used throughout the following tests is

$$x_l = \chi_l + 0.1 \sin(\pi\chi_1) \sin(\pi\chi_2) \sin(\pi\chi_3) \quad \text{for } x_l = 1, 2, 3. \tag{68}$$

The mesh has been generated using the HOPR package [78] with a geometrical order of approximation of $N_{geo} = 2$.

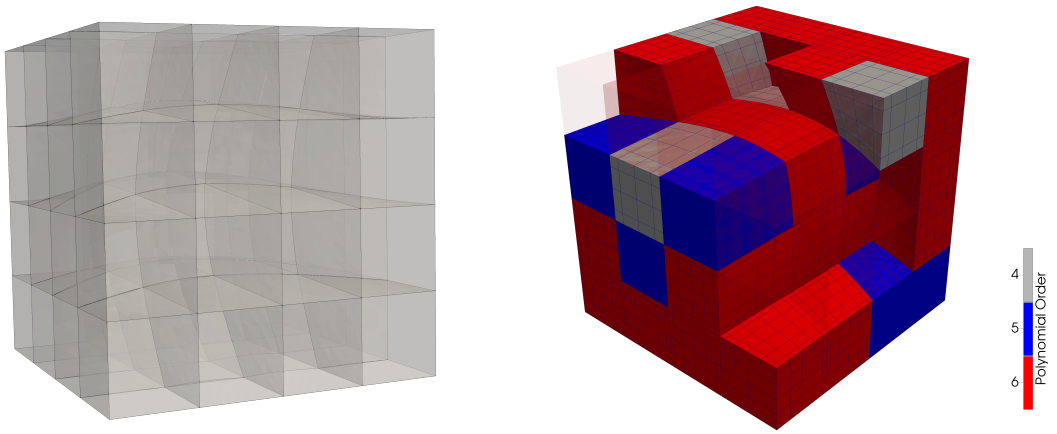


Figure 1: mesh used for the freestream preservation and primary conservation and an example of a random polynomial order distribution across the domain.

The first test is conducted to prove that the scheme is freestream preserving. We set a uniform initial condition, in this case $\phi(t=0) = 1$, and measure the error produced as the solution is marched in time. To measure the error we use the $\mathbb{L}^2$ and the $\mathbb{L}^\infty$ norms which are computed as

$$\|\phi\|_{\mathbb{L}^2} = \sqrt{\sum_e \langle \mathcal{J}\Phi, \Phi \rangle_{E,N}}, \quad \|\phi\|_{\mathbb{L}^\infty} = max\left(\sqrt{\langle \mathcal{J}\Phi, \Phi \rangle_{E,N}}\right). \tag{69}$$

We conduct a series of tests with a random distribution of the polynomial order across the domain from the interval $\mathrm{P} \in [4,6]$, as in the example presented in Figure 1. Since $N_{geo} = 2$, the geometric conditions described for general 3D p–non–conforming meshes are automatically satisfied. The time marching is performed with an explicit low storage 3^{rd} order Runge–Kutta (RK3) scheme [79]. The parameters of the simulation are presented in Table 1.

M	k	σ	Δt	K_0	S_0
1.0	10^{-4}	0.0	10^{-7}	1.0	1.0

Table 1: parameter values used for the freestream preservation and primary quantity conservation tests.

The results obtained for the freestream preservation test are summarized in Table 2. We present the $\mathbb{L}^2$ norm of the initial rate of change of the concentration parameter as well as the $\mathbb{L}^2$ and $\mathbb{L}^\infty$ norms of the error at a time instant of $t = 1$.

$\|\phi_t(t=0)\|_{\mathbb{L}^2}$	$\|\phi(t=1)-\phi(t=0)\|_{\mathbb{L}^2}$	$\|\phi(t=1)-\phi(t=0)\|_{\mathbb{L}^\infty}$
$\phi\, 1.471 \cdot 10^{-11}$	$1.806 \cdot 10^{-14}$	$5.883 \cdot 10^{-16}$

Table 2: results for the freestream preservation test.

As presented in Table 2, the results indicate that the scheme is freestream preserving with the error being close to machine rounding error.
The second test is conducted to prove that the scheme is primary quantity conserving. In this case we measure the error in total mass as

$$\text{Total mass error} = \int_{\Omega} \phi(t) - \phi(0) \mathrm{d}\vec{x}. \tag{70}$$

For this test we use the same 4×4 mesh with a geometrical order of approximation $N_{geo} = 2$. In this case, we use a random distribution for the polynomial order across the domain from the interval P $\in [4, 5]$. The time marching also in this case is performed using the RK3 scheme. The parameters of this simulation are presented in Table 1. The initial condition is a random distribution for the concentration parameter ϕ in the interval $\phi \in [-1, 1]$. The solution is computed until a final time of t=1.
The results are presented in Figure 2. As presented, the error remains close to machine rounding error throughout the simulation and thus the scheme shows primary quantities conservation.

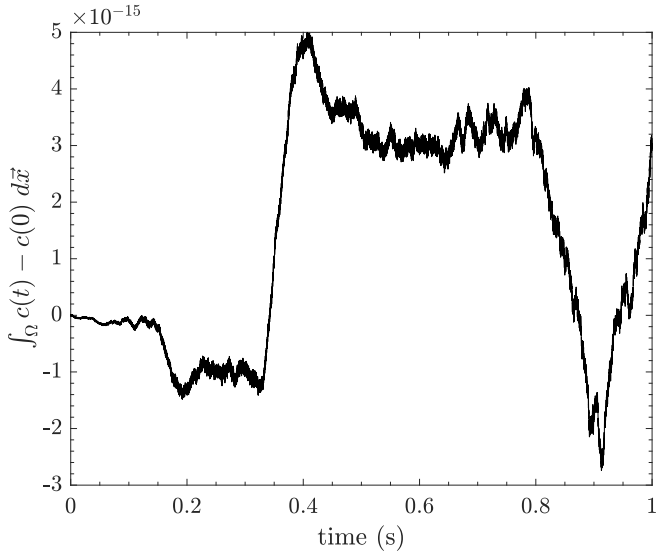


Figure 2: results for the primary quantity conservation test with a random initial condition.

7 Numerical results

In this section, we consider three different numerical experiments: a one–dimensional interface test case, the formation of a circular static bubble and that of a spinodal decomposition in two and three dimensions. The one–dimensional test case is used to initially evaluate the accuracy of the adaptation compared to the conforming version of the scheme as well as to quantify the error introduced when adapting along the direction of the interface. The static bubble formation is an unsteady case used to assess the accuracy of the scheme in two–dimensions and showcase the ability to achieve similar level of accuracy through a significant reduction in the degrees of freedom. Lastly, the spinodal decomposition is a classic problem of the Cahn–Hilliard equation. We examine a two– and a three–dimensional test case of a spinodal decomposition. We show that the same results for the free–energy and final state can be recovered with the use of adaptation and present the course of the reduction of the degrees of freedom as the solution evolves. These test cases have been chosen to illustrate the capabilities, robustness and accuracy of the scheme.

7.1 One–dimensional interface

The first test case to be examined is a one–dimensional interface between two immiscible fluids. It is a case that proves the effectiveness of this methodology and can be used as a preliminary study to assess the characteristics of the scheme. Moreover, the impact of the polynomial order jump condition on the accuracy among neighboring elements is assessed. The initial condition is a sharp jump of the phase–field variable ϕ between the two phases is defined as

$$x \in [0, 40],\ t \in [0, 15], \qquad \phi(x, 0) = \begin{cases} 1 & \text{for} \quad x \geq 20 \\ -1 & \text{for} \quad x < 20 \end{cases} . \tag{71}$$

The Neumann boundary condition (8) is enforced at both ends of the domain. The model parameters used in this case are presented in Table 3. Recall that the parameters K_0 and S_0 are related to the IMEX time marching scheme. The backward difference Euler has been chosen ($K_0 = 1$) as well as the value for $S_0 = 1$ that guarantees non–linear stability [8].

M	k	σ	Δt	K_0	S_0
1.5	2.25	0.0	10^{-3}	1.0	1.0

Table 3: parameter values used for the one–dimensional interface test case.

	Number of elements (element size/interface width)		
Domain Dimensions	Mesh 1	Mesh 2	Mesh 3
[0,40]	10 (1.0)	20 (0.5)	40 (0.25)

Table 4: mesh specification for the test case of the steady state one–dimensional interface and relative element size compared to the interface width as approximated by (11).

As the simulation evolves, the system converges to a steady state. We chose to compare the convergence of the scheme based on a very fine solution of $N = 10$ to assess the accuracy of the adaptation process. The initial condition of the problem along with the final solution are presented in Figure 3. The adaptation interval in this test case does not correlate with the accuracy of the results, as this is a steady case and the adaptation performed on the first timestep is sufficient.

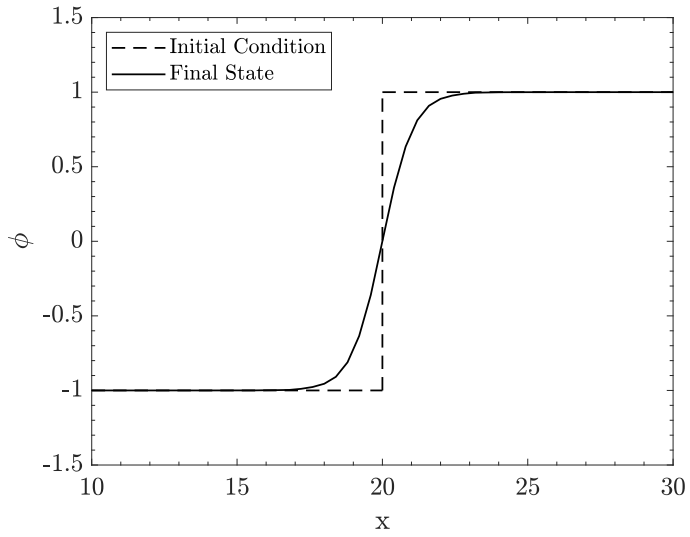


Figure 3: representation of the initial condition for the one–dimensional interface test and the final solution ($t = 15$) on Mesh 1 with $N = 6$.

We first establish the convergence rates and characteristics when uniformly refining the polynomial order of the solution, as presented in Figure 4. More specifically, the convergence test has been performed for three different meshes. The coarsest case corresponds to a cell size which is equal to the interface width, as approximated from the steady state solution (11). The subsequent two meshes have cell sizes of half and a quarter of the initial cell size respectively. From the results presented in Figure 4, we confirm that the polynomial refinement leads to exponential convergence rates. The convergence rate has an even–odd behavior which is also present in the manufactured solution case in [8]. Through this test case, the effect of the polynomial refinement has been showcased as well as the dependency on the cell size of the mesh used.

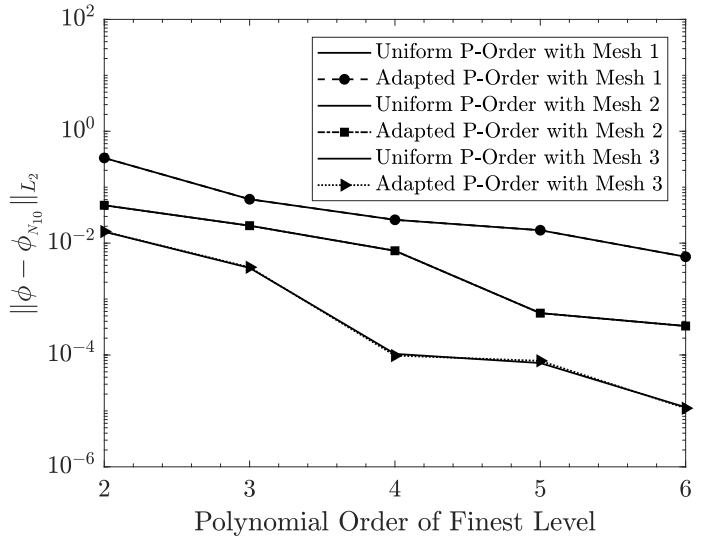


Figure 4: comparison of the L^2 norm of the error for the one–dimensional steady state interface test case at $t = 15$ between the $N = 10$ uniform solution and the adapted solution with the two different jump criteria. The results presented are for varying fine level polynomial order on the three different meshes specified in Table 4.

Having established for this case that the polynomial refinement leads to exponential convergence,

we validate that these characteristics are preserved when performing polynomial local refinement and coarsening. In all cases, the initial setup has been a uniform fine level polynomial order across the domain. This polynomial order is retained in the region of the interface, whereas there is coarsening in the bulk of the phases. The lowest polynomial order chosen for the bulk of the phases is $N_{\text{Coarse}} = 2$ as described in Sec. 4.
This case also incorporates an initial test for the jump of the polynomial order across different elements. Since it is a one–dimensional case it does not unveil the effect and the error that might occur due to the use of the mortar method. However, it has been used as a mean of identifying errors that occur while changing the polynomial order in the direction of the interface. Two cases for the jump of the polynomial order between two subsequent elements have been tested, the $N - 1$ and the $2N/3$ criteria as introduced in Sec. 4.

The results of the adaptation process are presented in Figure 4. The results with the use of adaptation match those of the uniform polynomial order for all meshes and polynomial orders tested. Furthermore, the reduction achieved in degrees of freedom throughout the simulation of the one–dimensional interface is presented in Table 5.

	Mesh 1			
	$N_{\text{Fine}} = 3$	$N_{\text{Fine}} = 4$	$N_{\text{Fine}} = 5$	$N_{\text{Fine}} = 6$
Initial DoFs	40	50	60	70
Final DoFs	34	38	42	46
Reduction %	15	25.5	30	34.2
	Mesh 2			
	$N_{\text{Fine}} = 3$	$N_{\text{Fine}} = 4$	$N_{\text{Fine}} = 5$	$N_{\text{Fine}} = 6$
Initial DoFs	80	100	120	140
Final DoFs	60	64	70	80
Reduction %	25	36	41.6	42.8
	Mesh 3			
	$N_{\text{Fine}} = 3$	$N_{\text{Fine}} = 4$	$N_{\text{Fine}} = 5$	$N_{\text{Fine}} = 6$
Initial DoFs	160	200	240	280
Final DoFs	104	110	118	130
Reduction %	35	45	50.8	53.5

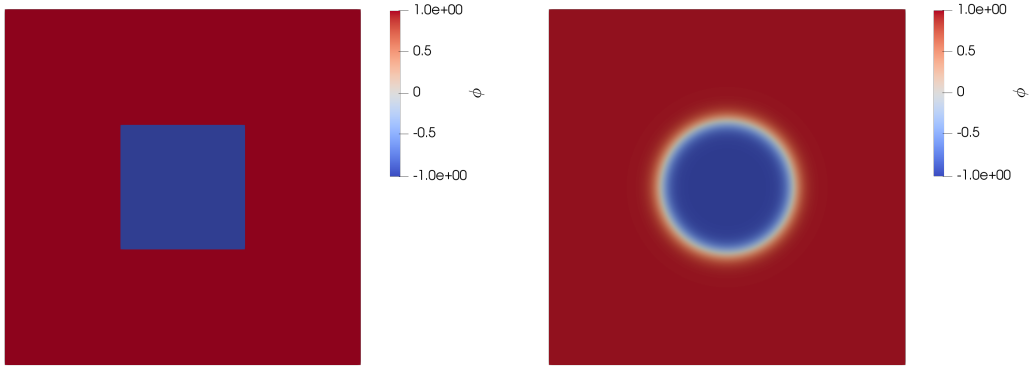
Table 5: final number of degrees of freedom for the one–dimensional steady interface test case upon the adaptation process with the $2N/3$ criterion. Results presented for the three different meshes of Table 4.

7.2 Circular static bubble

The second case to be studied is that of the formation of a static bubble. The initial condition is a square at the center of the domain with the phase-field parameter taking the value of $\phi = -1$ and the rest of the domain the value of $\phi = 1$,

$$\phi(x, y) = \begin{cases} 1 & \text{if } |x - 20| \leq 7 \text{ and } |y - 20| \leq 7 \\ -1 & \text{otherwise} \end{cases} . \tag{72}$$

The initial condition is presented in Figure 5(a). Subsequently, the solution evolves to a state of minimum energy, which is a circular bubble as presented in Figure 5(b).



(a) $t = 0$ (b) $t = 100$

Figure 5: visualization of the (a) initial condition ($t = 0$) and (b) final condition ($t = 100$) of (72) for the two–dimensional circular static bubble test case.

The parameters of the simulation are presented in Table 6. The final solution time is $t = 100$. Two different Cartesian meshes have been considered. The first has an element size which is half of the interface width and the second being a quarter of the interface width. Also in this case, the element size was based on the knowledge of the parameter in the Cahn–Hilliard equation that controls the interface width. The specification of the mesh is presented in Table 7. All boundaries are treated with periodic boundary conditions. A suitable adaptation interval for this test case and the meshes specified in Table 7 has been identified to be every 500 iterations. It has to be mentioned that this value is dependent upon the element size, the dynamics of the problem as well as the timestep chosen for each particular simulation.

M	k	σ	Δt	K_0	S_0
1.5	2.25	0.0	10^{-3}	1	1

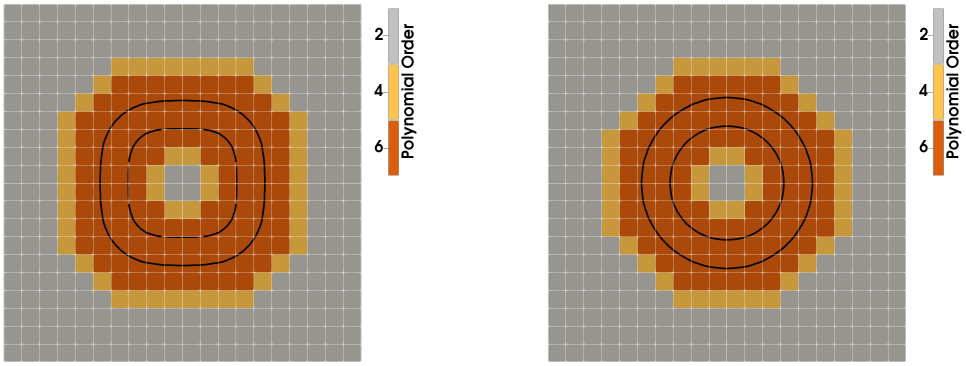
Table 6: parameter values used for the static 2D circular bubble test case.

	Number of elements (element size/interface width)	
Domain Dimensions	Mesh 1	Mesh 2
$[0, 40]^2$	20×20 (0.5)	40×40 (0.25)

Table 7: mesh specification for the test case of the static two–dimensional circular bubble and relative element size compared to the interface width as approximated by (11).

The results using the adaptation algorithm on this problem compared to the uniform solution are presented in Figures 7 and 9. The data points for the adapted results in Figure 7 correspond to Mesh 1 and a fine level polynomial order of $N_{\text{Fine}} \in \{2, 3, 4, 5, 6, 8\}$ whereas $N_{\text{Coarse}} = 2$. For this particular mesh, the results amongst the cases with the different polynomial order jump as well as the uniform solution are matching. The adaptation does not incur any additional error. This is due to the element size being relatively large and in conjunction with the buffer of refined elements the coarse elements are situated far away from the interface region and thus the results match those of the uniform solution. What distinguishes the two solutions that make use of the adaptation, is that the same accuracy can be achieved with fewer degrees of freedom for the case of the $2N/3$ jump condition. Figures 6(a), 6(b) illustrate the field of the polynomial order for the

case of Mesh 1 for two different time instants along with the $2N/3$ criterion. Given a sufficient adaptation rate the polynomial order jump does not affect the solution and does not cause any error for this specific mesh and simulation settings.



(a) t=2.5 (b) t=100

Figure 6: visualization of the polynomial order distribution for the instant of (a) $t = 2.5$ and (b) $t = 100$ for the static circular bubble test case on Mesh 1 with the $2N/3$ jump condition. The solid black lines denote the position of the interface $-0.9 \leq \phi \leq 0.9$.

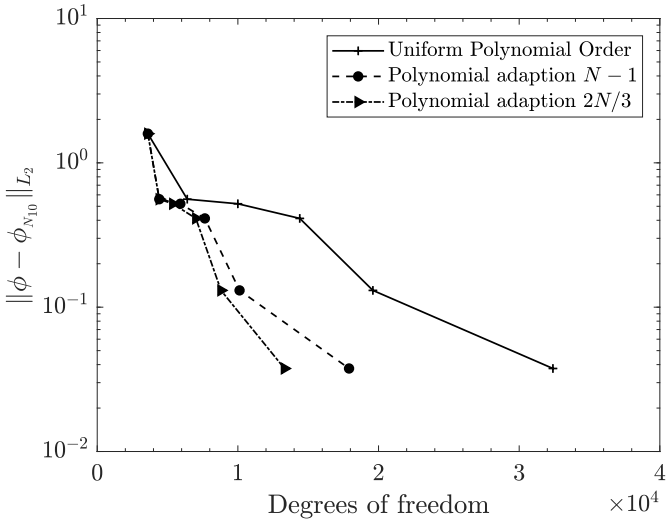
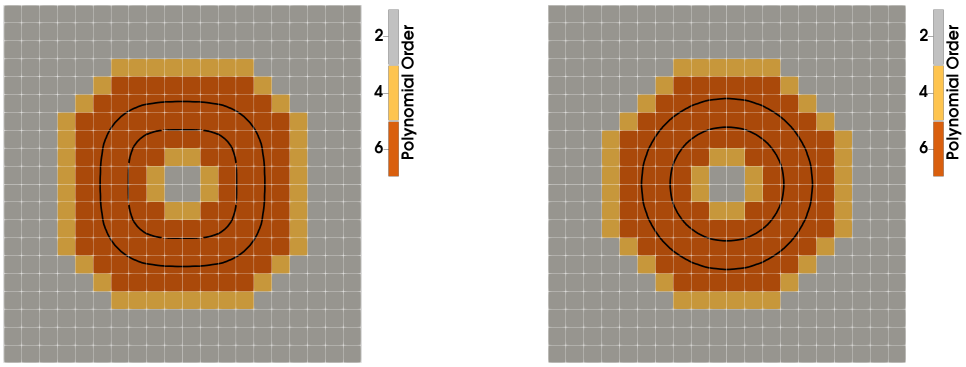


Figure 7: convergence of the L^2 error norm for the case of Mesh 1 of the static bubble test case ($t = 100$). Comparison of the achieved accuracy for a given number of degrees of freedom for a uniform polynomial order and adapted with two different jump conditions.

In Figure 9 the results for Mesh 2 and polynomial orders of $N_{\text{Fine}} \in [2, 6]$ are presented. The fine polynomial order is retained in the interface region and the rest of the domain is coarsened to $N_{\text{Coarse}} = 2$ according to the phase–field parameter value in each element. Furthermore, in Figure 9 the convergence characteristics for the two different jump conditions are presented. The adaptation process achieves significant decrease in the degrees of freedom, with the more aggressive approach of $2N/3$ showcasing the largest benefit. The field of the polynomial order for the latter is presented in Figures 8(a) and 8(b). However, in this case the adaptation incurs

some error in the solution. For the case of $N_{\text{Fine}} = 6$, the offset in the error compared to the uniform solution of both the $N-1$ and $2N/3$ criteria is of the order of 10^{-5}.



(a) $t = 2.5$ (b) $t = 100$

Figure 8: visualization of the polynomial order distribution for the instant of (a) $t = 2.5$ and (b) $t = 100$ for the static circular bubble test case on Mesh 2 with the $2N/3$ jump condition. The solid black lines denote the position of the interface $-0.9 \leq \phi \leq 0.9$.

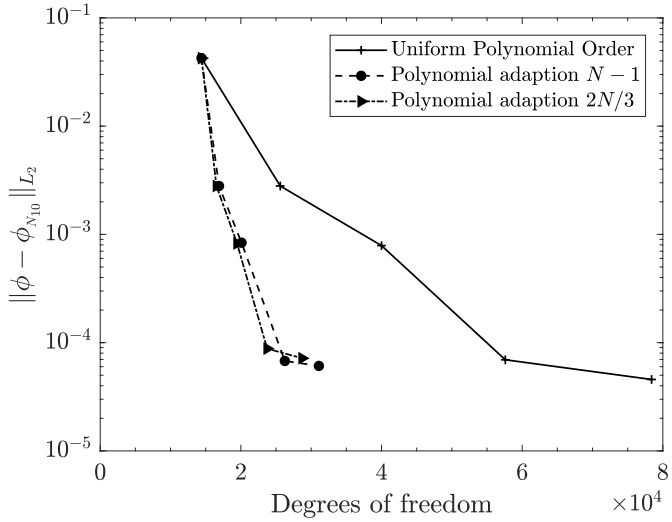
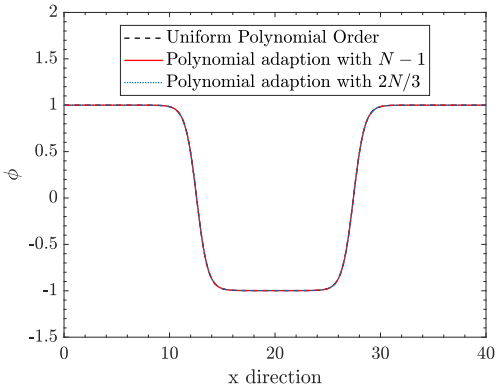


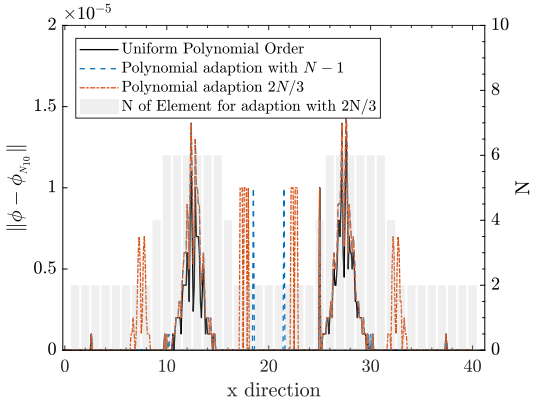
Figure 9: convergence of the L^2 error for the case of Mesh 2 of the static bubble test case ($t = 100$). Comparison of the achieved accuracy for a given number of degrees of freedom for a uniform polynomial order and adapted with two different jump conditions.

To understand the underlying difference between the two adapted solutions as well as the error generated during the polynomial adaptation process, we show a cross section along the middle line of the bubble. Figure 10(a) presents the profiles of the uniform and adapted solutions with $N_{\text{Fine}} = 6$ and Figure 10(b) presents the error of the aforementioned solutions compared to the fine uniform solution with polynomial order $N = 10$. The results in Figure 10(b) show that in the region of the interfaces, $11.5 \leq x \leq 14.5$ and $26.5 \leq x \leq 29.5$, the error between the adapted and uniform solutions is similar. However, some error arise in the bulk region of the phases near the interface. Thus, the interface region is captured adequately with the use of

the adaptation technique and the accuracy is maintained with a significantly smaller number of degrees of freedom.



(a) Cross section of the final solution along the mid–line for Mesh 2 of the static bubble test case comparing the solution profile for a $N = 6$ uniform solution and the adapted solution with $N_{Fine} = 6$ with the two different jump criteria.



(b) Error along the mid–line of the domain compared to a fine uniform solution of $N = 10$ for the uniform solution with $N = 6$, the adapted solution with $N_{Fine} = 6$ and the $N - 1$ criterion and an adapted solution with $N_{Fine} = 6$ and the $2N/3$ criterion

Figure 10: solution along the mid–line of the circular bubble test case (a) and error along the mid–line compared to the fine reference solution with $N = 10$ (b) for ($t = 100$).

7.3 Spinodal Decomposition

The spinodal decomposition is a classic problem of the Cahn–Hilliard equation [3]. The main idea of this test is to introduce noise of different frequencies to an initial state of $\phi = 0$ and allow it to evolve to the state of minimum energy. This is accompanied by a separation of the phases and the creation of bulk regions for each phase. We test the qualities and effectiveness of the adaptation scheme on a two– and a three–dimensional spinodal decomposition.

7.3.1 Two–dimensional spinodal decomposition

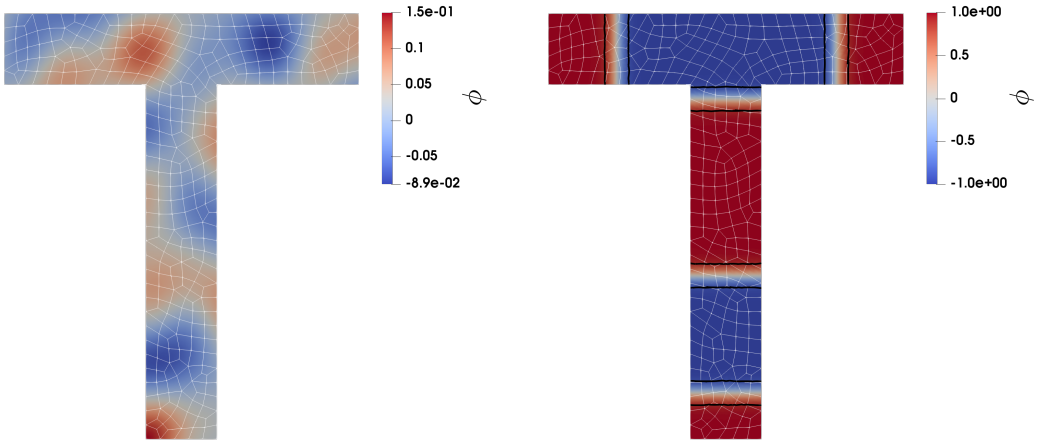
The initial condition,

$$\begin{aligned}\phi_0(x,y) =& 0.05\big(\cos(0.105x)\cos(0.11y) + (\cos(0.13x)\cos(0.087y))^2 \\ &+ \cos(0.025x - 0.15y)\cos(0.07x - 0.02y)\big),\end{aligned} \tag{73}$$

and the problem specification stem from the benchmark cases described in [80].
The visualization of the initial state of the simulation is presented in Figure 11(a). The parameters are specified in Table 8. The adaptation frequency has been set to occur every 500 iterations. This problem has also been studied with the present numerical schemes in [8], and the sensitivity to the parameters of the time marching IMEX scheme has been identified. Also, the effect of having a structured or unstructured mesh has been quantified, and therefore the reader is invited to see the results therein. Here, we focus on the effect of the adaptation using a more general unstructured mesh. The quantity of interest in the spinodal decomposition is the evolution of the free–energy. We also monitor the variation of the degrees of freedom as the simulation evolves.

M	k	σ	Δt	K_0	S_0
1.0	10	1.0	10^{-1}	1.0	1.0

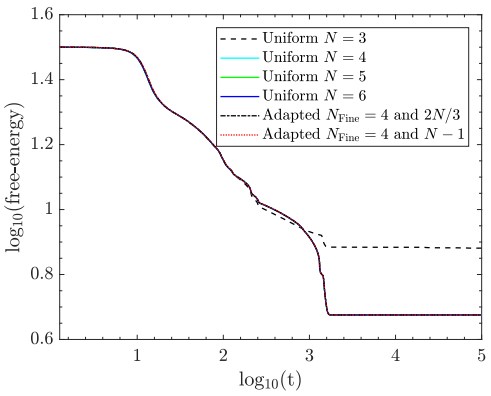
Table 8: parameter values used for the spinodal decomposition test case.



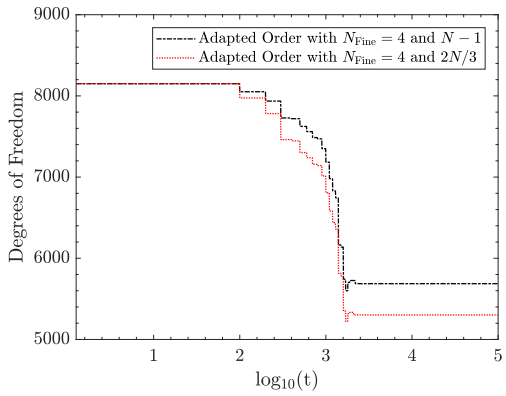
(a) Initial state (t=0)

(b) Final state (t=10000)

Figure 11: visualization of the (a) initial state and (b) final state of the spinodal decomposition problem. The continuous black lines denote the locations of the interface where the phase-field parameter takes the values $|\phi| = 0.9$.



(a) free–energy evolution with varying polynomial order. For all polynomial orders above $N = 4$ the solution converges to the same final state. The adapted solution with $N_{Fine} = 4$ matches the uniform one for both jump criteria.



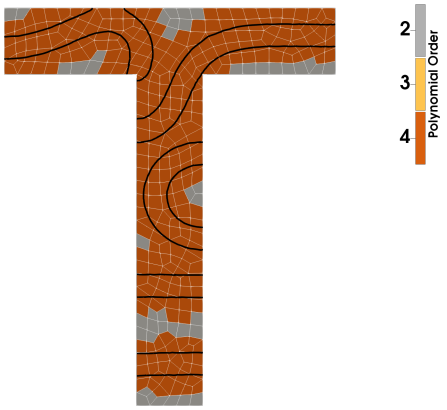
(b) number of degrees of freedom versus the physical time for the two different jump criteria.

Figure 12: results of the free–energy and degrees of freedom for the two–dimensional spinodal decomposition test case in the T–section.

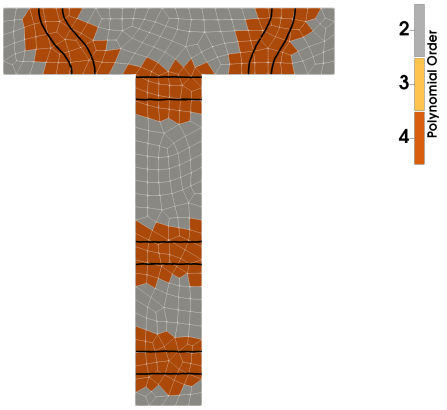
The results presented in Figure 12(a) indicate that for that specific mesh the quantity of interest, the free–energy, converges when the polynomial order used is $N \geq 4$. The final state is presented in Figure 11(b). Thus a polynomial order of $N = 4$ has been used as the basis for the comparison among the uniform and the adapted solution. The coarse level polynomial order used in this test case is $N_{\text{Coarse}} = 2$. As presented in Figure 12(a), the adapted solution manages to achieve the same solution and the same final state. The case with the $N - 1$ criterion manages to achieve a 30% reduction in the degrees of freedom whereas the $2N/3$ achieves a 35% reduction. A significant proportion of the simulation is carried out using fewer degrees of freedom.

The diagrams in Figure 13 present the field of the polynomial order as well as the position of the interfaces for different time instants of this spinodal decomposition problem. In Figure 13(a),

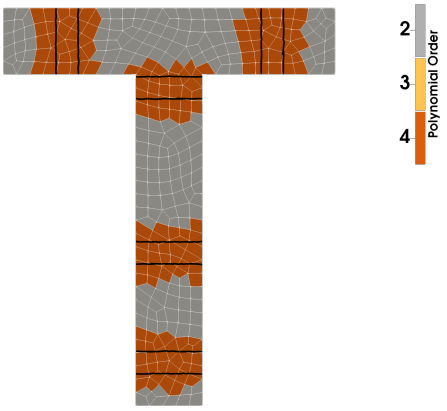
which corresponds to $t = 800$, there are few elements which have been coarsened because the phases still undergo through the process of separation. As time evolves, the two phases are further separated as presented in Figure 13(b) and there is a significant decrease in the degrees of freedom. The final state is presented in Figure 13(c), which reveals that a large proportion of the elements has a coarse level polynomial order.



(a) $t = 800$



(b) $t = 1600$



(c) $t = 10^4$

Figure 13: visualization of the polynomial order distribution on the solution for (a) $t = 800$, (b) $t = 1600$ and (c) $t = 10^4$ for the spinodal decomposition test case with the $2N/3$ criterion. The continuous black lines denote the locations of the interface where $|\phi| = 0.9$.

7.3.2 Three–dimensional spinodal decomposition

We test a similar spinodal decomposition problem on three dimensions. The problem definition stems from [8] and we use the cylindrical mesh, with 920 elements and element size of 0.1 ($L = 1$, $D = 1$). This is a 2D extruded mesh whose external faces are curvilinear and the geometrical order of approximation is $N_{\text{geo}} = 2$ which conforms with the freestream preservation restrictions. The initial condition as defined in [8],

$$\begin{aligned}\phi_0(x, y) =& 0.015\cos(0.5x - 10z)\cos(0.7x + 10zy + 1) + 0.02\cos\left(20y^2 + 15x^2\right)\sin(8x + 2y) \\ &+ 0.02\cos\left(10\sqrt{y^2 + z^2}\right)\cos(5xy)\cos(20x + 10z) + 0.01\cos(3x)\cos(3z)\cos(4y)\,.\end{aligned} \tag{74}$$

In this case, we solve the Cahn–Hilliard equation using the chemical–free energy (9) and we also use an interface width parameter $k = 0.0025$ instead of the value $k = 0.01$ used in [8]. The latter has been chosen in order to simulate a system with a smaller interface and showcase the scheme's capabilities. The parameters of the simulation are presented in Table 9. We use a fine polynomial order of $N_{\text{Fine}} = 4$, a coarse polynomial order of $N_{\text{Coarse}} = 2$ and the $2N/3$ jump criterion. The adaptation frequency has been defined to occur at an interval of 1000 iterations.

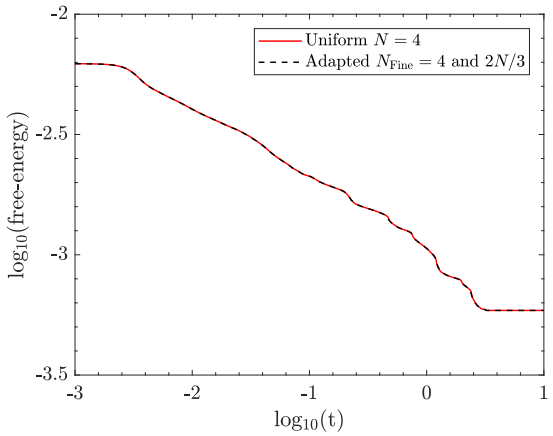


Figure 14: evolution of the free–energy as the three–dimensional spinodal decomposition evolves. The adapted solution with $N_{\text{Fine}} = 4$ matches the uniform solution with $N = 4$.

M	k	σ	Δt	K_0	S_0
1.0	0.0025	1.0	10^{-4}	1	1

Table 9: parameter values used for the three–dimensional spinodal decomposition test case.

The results presented in Figure 14 show that the results for the free–energy from the adapted solution match exactly those recovered when using a uniform polynomial order across the domain. As the solution evolves and the phases are separated, the adaptation scheme reduces the degrees of freedom as presented in Figure 15. For this problem specification, the reduction achieved is 48%. In Figure 16 the polynomial order distribution for different time instants is presented. The final state, as presented in Figure 16(d), is a flat interface separating the two phases.

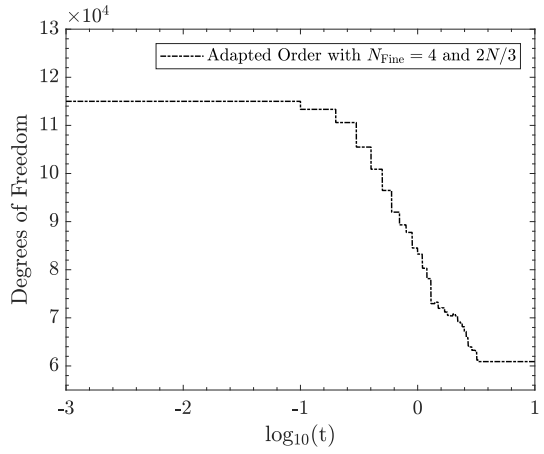
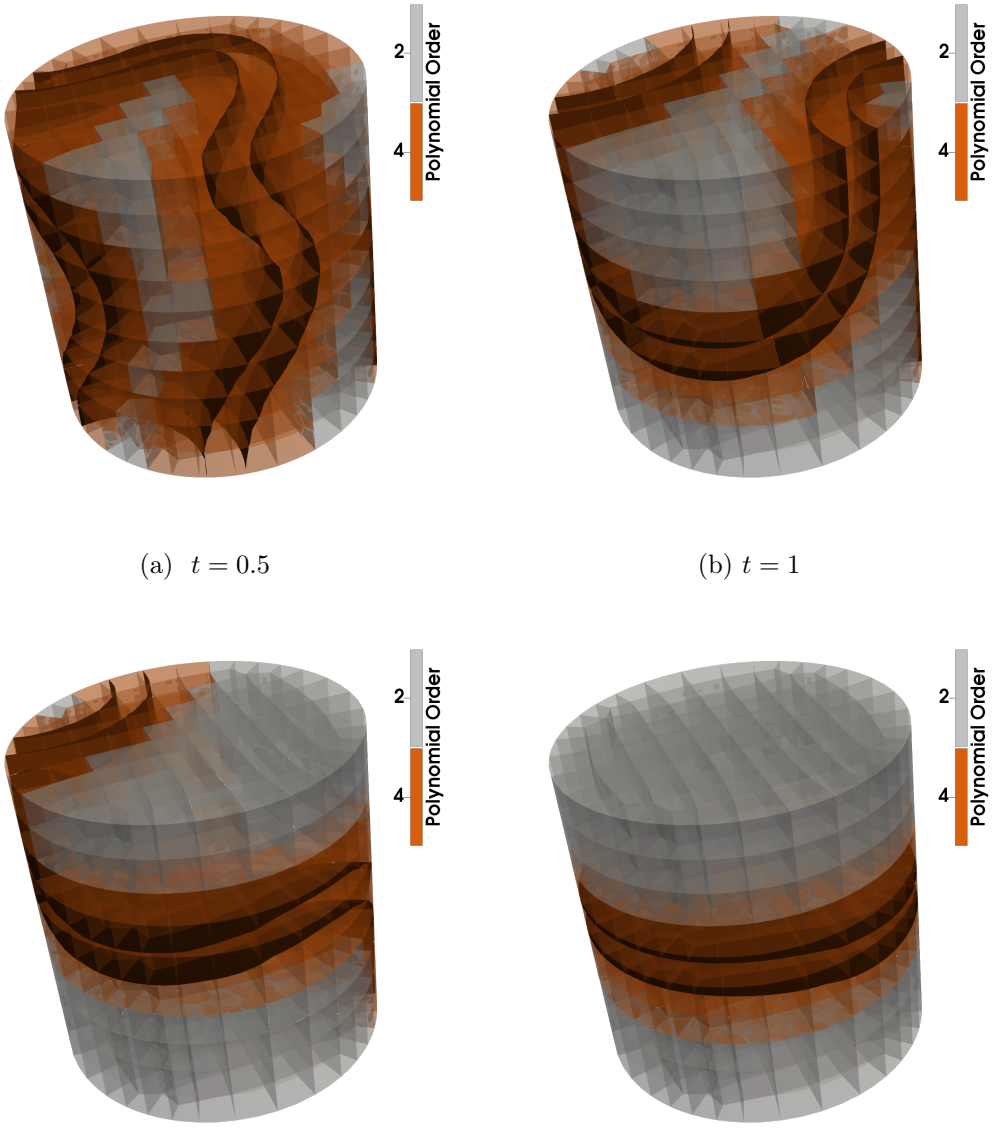


Figure 15: number of degrees of freedom versus the physical time as the three–dimensional spinodal decomposition simulation evolves.



(a) $t = 0.5$ (b) $t = 1$

(c) $t = 1.5$ (d) $t = 10$

Figure 16: visualization of the polynomial order distribution on the solution for (a) $t = 0.5$, (b) $t = 1$, (c) $t = 1.5$ and (d) $t = 10$ for the three–dimensional spinodal decomposition test case with the $2N/3$ criterion. The continuous black lines denote the locations of the interface where $|\phi| = 0.9$.

8 Conclusions

In this work, we have established the theoretical framework for a free–energy stable discontinuous Galerkin scheme for the Cahn–Hilliard equation that targets p–non–conforming meshes. The analysis carried out has proven that for the continuous and the fully–discrete time setting, the stability characteristics of the conforming scheme of [8] can be maintained when transitioning to elements with non–uniform polynomial order. To do so, we use the BR1 numerical flux [70] and the made use of the standard mortar method [47].
In addition, we have designed a methodology for performing local polynomial refinement for the Cahn–Hilliard equation. This specific algorithm has the advantage of being simple and low–cost, based on the advantages that the higher order Discontinuous Galerkin Spectral Element Methods (DGSEM) offer as well as the Cahn–Hilliard model. We have also implemented and tested two different criteria to ensure that the scheme maintains the robustness and accuracy of the uniform solver. Lastly, the effectiveness of the method has been tested and verified through various two– and three–dimensional test cases, such as the spinodal decomposition, which have showcased that there is a significant reduction of the degrees of freedom required to attain the same accuracy as in the conforming version of the scheme.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

Gerasimos Ntoukas and Esteban Ferrer would like to thank the European Union's Horizon 2020 Research and Innovation Program under the Marie Skłodowska-Curie grant agreement No 813605 for the ASIMIA ITN-EID project. Gonzalo Rubio and Eusebio Valero acknowledge the funding received by the project SIMOPAIR (Project No. REF: RTI2018-097075-B-I00) from the Ministry of Innovation of Spain. The authors gratefully acknowledge the Universidad Politécnica de Madrid (www.upm.es) for providing computing resources on Magerit Supercomputer.

A Stability Analysis

A.1 Semi–discrete stability analysis

In this section we follow the steps from [8] used to derive the free–energy bound (15) to derive an equivalent discrete bound for a modification of (13d) assuming exact time integration. Following the continuous analysis, we first take the (exact) time derivative of (13d),

$$\left\langle \mathcal{J}\vec{Q}_t, \vec{\varphi}_Q \right\rangle_{E,N} = \int_{\partial E,N} \Phi_t^\star \tilde{\boldsymbol{\varphi}}_Q \cdot \hat{n}\, \mathrm{d}S_\xi - \langle \Phi_t, \nabla_\xi \cdot \tilde{\boldsymbol{\varphi}}_Q \rangle_{E,N}\,, \tag{75}$$

where we omitted the test function time derivative terms since (13d) holds for any polynomial test function ($\boldsymbol{\varphi}_Q$ or $\boldsymbol{\varphi}_{Q,t}$),

$$\left\langle \mathcal{J}\vec{Q}, \vec{\varphi}_{Q,t} \right\rangle_{E,N} = \int_{\partial E,N} \Phi^\star \tilde{\boldsymbol{\varphi}}_{Q,t} \cdot \hat{n}\, \mathrm{d}S_\xi - \langle \Phi, \nabla_\xi \cdot \tilde{\boldsymbol{\varphi}}_{Q,t} \rangle_{E,N}\,. \tag{76}$$

Next, we use the summation–by–parts property (30) in (75),

$$\left\langle \mathcal{J}\vec{Q}_t, \vec{\varphi}_Q \right\rangle_{E,N} = \int_{\partial E,N} (\Phi_t^\star - \Phi_t)\, \tilde{\boldsymbol{\varphi}}_Q \cdot \tilde{n}\, \mathrm{d}S_\xi + \langle \nabla_\xi \Phi_t, \tilde{\boldsymbol{\varphi}}_Q \rangle_{E,N}\,, \tag{77}$$

and replace $\vec{\varphi}_Q = \vec{Q}$ (and $\tilde{\varphi}_Q = \mathcal{M}^T \vec{\varphi}_Q$ as shown in (40)), to get the time derivative,

$$\left\langle \mathcal{J}\vec{Q}_t, \vec{Q} \right\rangle_{E,N} = \frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{J}||\vec{Q}||^2_{E,N} = \int_{\partial E,N} (\Phi_t^\star - \Phi_t)\, \tilde{\vec{Q}} \cdot \tilde{n}\, \mathrm{d}S_\xi + \left\langle \nabla_\xi \Phi_t, \tilde{\vec{Q}} \right\rangle_{E,N}. \tag{78}$$

We then set $\varphi_W = \Phi_t$ in (41c),

$$\langle \mathcal{J}W, \Phi_t \rangle_{E,N} = \left\langle \mathcal{J}\frac{\mathrm{d}\Psi}{\mathrm{d}\Phi}, \Phi_t \right\rangle_{E,N} - k \int_{\partial E,N} \Phi_t \tilde{\vec{Q}}^\star \cdot \hat{n}\, \mathrm{d}S_\xi + k \left\langle \tilde{\vec{Q}}, \nabla_\xi \Phi_t \right\rangle_{E,N}. \tag{79}$$

Since the time derivative is exact, we can use the chain rule for the chemical free–energy potential derivative,

$$\left\langle \mathcal{J}\frac{\mathrm{d}\Psi}{\mathrm{d}\Phi}, \Phi_t \right\rangle_{E,N} = \frac{\mathrm{d}}{\mathrm{d}t} \langle \mathcal{J}\Psi, 1 \rangle_{E,N}\,. \tag{80}$$

We subtract (78) multiplied by k from (79), and use the result in (80) to obtain,

$$\begin{aligned} \langle \mathcal{J}W, \Phi_t \rangle_{E,N} = \frac{\mathrm{d}}{\mathrm{d}t} \langle \mathcal{J}\Psi, 1 \rangle_{E,N} + k\frac{\mathrm{d}}{\mathrm{d}t} \left\langle \mathcal{J}\vec{Q}, \vec{Q} \right\rangle_{E,N} \\ - k \int_{\partial E,N} \left(\Phi_t \tilde{\vec{Q}}^\star \cdot \hat{n} + \Phi_t^\star \tilde{\vec{Q}} \cdot \hat{n} - \Phi_t \tilde{\vec{Q}} \cdot \hat{n} \right) \mathrm{d}S_\xi. \end{aligned} \tag{81}$$

Next, we use the summation–by–parts property (30) again in (41b),

$$\left\langle \mathcal{J}\vec{F}, \vec{\varphi}_F \right\rangle_{E,N} = \int_{\partial E,N} (W^\star - W)\, \tilde{\boldsymbol{\varphi}}_F \cdot \hat{n}\, \mathrm{d}S_\xi + \langle \nabla_\xi W, \tilde{\boldsymbol{\varphi}}_F \rangle_{E,N}\,, \tag{82}$$

and we set $\vec{\varphi}_F = M\vec{F}$, so that

$$\left\langle \mathcal{J}\vec{F}, M\vec{F} \right\rangle_{E,N} = \int_{\partial E,N} (W^\star - W)\, M\tilde{\vec{F}} \cdot \hat{n}\, \mathrm{d}S_\xi + \left\langle \nabla_\xi W, M\tilde{\vec{F}} \right\rangle_{E,N}. \tag{83}$$

Lastly, we set $\varphi_\Phi = W$ in (41a),

$$\langle \mathcal{J}\Phi_t, W \rangle_{E,N} = \int_{\partial E,N} W \left(M\tilde{\vec{F}} \right)^\star \cdot \hat{n}\, \mathrm{d}S_\xi - \left\langle M\tilde{\vec{F}}, \nabla_\xi W \right\rangle_{E,N}, \tag{84}$$

and we sum (83) and (84) to find that

$$\langle \mathcal{J}\Phi_t, W\rangle_{E,N} = \int_{\partial E,N} \left(W\left(M\vec{\tilde{F}}\right)^\star \cdot \hat{n} + W^\star\left(M\vec{\tilde{F}}\right)\cdot\hat{n} - W\left(M\vec{\tilde{F}}\right)\cdot\hat{n}\right) \mathrm{d}S_\xi - \left\langle \mathcal{J}\vec{F}, M\vec{F}\right\rangle_{E,N}. \tag{85}$$

Now, (81) and (85) both contain the term $\langle \mathcal{J}W, \Phi_t\rangle_{E,N}$ on the left hand side, so we can equate both right hand sides to obtain

$$\begin{aligned} &\int_{\partial E,N} \left(W\left(M\vec{\tilde{F}}\right)^\star \cdot \hat{n} + W^\star\left(M\vec{\tilde{F}}\right)\cdot\hat{n} - W\left(M\vec{\tilde{F}}\right)\cdot\hat{n}\right) \mathrm{d}S_\xi - \left\langle \mathcal{J}\vec{F}, M\vec{F}\right\rangle_{E,N} \\ &= \frac{\mathrm{d}}{\mathrm{d}t}\langle \mathcal{J}\Psi, 1\rangle_{E,N} + k\frac{\mathrm{d}}{\mathrm{d}t}\left\langle \mathcal{J}\vec{Q}, \vec{Q}\right\rangle_{E,N} - k\int_{\partial E,N}\left(\Phi_t\vec{\tilde{Q}}^\star\cdot\hat{n} + \Phi_t^\star\vec{\tilde{Q}}\cdot\hat{n} - \Phi_t\vec{\tilde{Q}}\cdot\hat{n}\right)\mathrm{d}S_\xi. \end{aligned} \tag{86}$$

We rearrange (86) to move time derivatives to the left hand side of the equation. Additionally, we identify the volumetric discrete free–energy of the element,

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\langle \mathcal{J}\Psi, 1\rangle_{E,N} + \frac{k}{2}\mathcal{J}\left|\left|\vec{Q}\right|\right|^2_{E,N}\right) = \frac{\mathrm{d}}{\mathrm{d}t}\int_{E,N}\mathcal{J}\left(\Psi + \frac{k}{2}\vec{Q}\cdot\vec{Q}\right)dE = \mathcal{F}_t^{E,N_e}, \tag{87}$$

and thus, we simplify (86) to

$$\mathcal{F}_t^{E,N_e} = -\left\langle \mathcal{J}\vec{F}, M\vec{F}\right\rangle_{E,N} + k\mathrm{BT}_{E,N}(\Phi_t, \vec{Q}) + \mathrm{BT}_{E,N}(W, M\vec{F}). \tag{88}$$

In (88), we find that the volumetric free–energy is dissipated in the element interior by the chemical potential flux (similarly to the continuous counterpart (15)), and exchanged with other elements through the boundary terms $\mathrm{BT}_{E,N}$.
To obtain an energy estimate similar to that in (15), we sum all element contributions, getting

$$\sum_e \mathcal{F}_t^{E,N_e} = \bar{\mathcal{F}}_t = \left\langle \mathcal{J}\vec{F}, M\vec{F}\right\rangle_{E,N} + \sum_e\left(k\mathrm{BT}_{E,N}(\Phi_t, \vec{Q}) + \mathrm{BT}_{E,N}(W, M\vec{F})\right). \tag{89}$$

We then split the boundary quadratures $\sum_e \mathrm{BT}_{E,N} = \mathrm{IBT}_N + \mathrm{PBT}_N$ into the combination of interior (IBT_N) and physical boundary (PBT_N) sums,

$$\sum_e \mathcal{F}_t^{E,N_e} + \mathrm{IBT} + \mathrm{PBT} = -\sum_e \left\langle \mathcal{J}M\vec{F}, \vec{F}\right\rangle_{E,N_e}. \tag{90}$$

A.2 Discrete Stability analysis

These steps are derived from the analysis presented in [8]. The discretized PDE in time is given by the following expression,

$$\phi^{n+1} - \phi^n = \Delta t \nabla \cdot \left(M \nabla \left(\frac{\mathrm{d}\psi(\phi^n)}{\mathrm{d}\phi} + S_0 \Delta\phi - k\nabla^2 \left(K_0 \phi^{n+1} + (1 - K_0)\phi^n \right) \right) \right), \tag{91}$$

The parameter K_0 controls the time marching method. So for $K_0 = 0$ it is explicit Euler, $K_0 = 1/2$ it is the Crank–Nicolson scheme, and $K_0 = 1$ is the implicit Euler. In this work, we only make use of the implicit Euler time marching scheme and thus for the rest of this analysis we assume that $K_0 = 1$

We introduce the time discretization (91) into (41) to obtain the fully–discrete discontinuous Galerkin approximation,

$$\left\langle \mathcal{J} \frac{\Phi^{n+1} - \Phi^n}{\Delta t}, \varphi_\Phi \right\rangle_{E,N} = \int_{\partial E,N} \varphi_\Phi \left(M \vec{\tilde{F}} \right)^{\star,\theta} \cdot \hat{n} \, \mathrm{d}S_\xi - \left\langle M \vec{\tilde{F}}^\theta, \nabla_\xi \varphi_\Phi \right\rangle_{E,N}, \tag{92a}$$

$$\left\langle \mathcal{J} \vec{F}^\theta, \vec{\varphi}_F \right\rangle_{E,N} = \int_{\partial E,N} \left(W^{\star,\theta} - W^\theta \right) \tilde{\boldsymbol{\varphi}}_F \cdot \hat{n} \, \mathrm{d}S_\xi + \left\langle \nabla_\xi W^\theta, \tilde{\boldsymbol{\varphi}}_F \right\rangle_{E,N}, \tag{92b}$$

$$\begin{aligned} \left\langle \mathcal{J} W^\theta, \varphi_W \right\rangle_{E,N} = & \left\langle \left(\frac{\mathrm{d}\Psi}{\mathrm{d}\Phi} \right)^n + S_0 \left(\Phi^{n+1} - \Phi^n \right), \mathcal{J} \varphi_W \right\rangle_{E,N} \\ & - k \int_{\partial E,N} \varphi_W \vec{\tilde{Q}}^{\star,n+1} \cdot \hat{n} \, \mathrm{d}S_\xi + k \left\langle \vec{\tilde{Q}}^{n+1}, \nabla_\xi \varphi_W \right\rangle_{E,N}, \end{aligned} \tag{92c}$$

$$\left\langle \mathcal{J} \vec{Q}^{n+1}, \vec{\varphi}_Q \right\rangle_{E,N} = \int_{\partial E,N} \left(\Phi^{\star,n+1} - \Phi^{n+1} \right) \tilde{\boldsymbol{\varphi}}_Q \cdot \hat{n} \, \mathrm{d}S_\xi + \left\langle \nabla_\xi \Phi^{n+1}, \tilde{\boldsymbol{\varphi}}_Q \right\rangle_{E,N}, \tag{92d}$$

where we use the superscript θ for variables (e.g. $\vec{F}^\theta$ or W^θ) that are not directly evaluated at t_n or t_{n+1} with the IMEX strategy, but on a combination of those depending on the different terms involved in (92c). Moreover, following the semi–discrete analysis, we have already applied the summation–by–parts property (30) in (92b) and (92d).

To analyze the stability of the system (92), we start by combining (92c) and (92d). We perform the first manipulations on (92d), which we set for both t_{n+1} and t_n,

$$\left\langle \mathcal{J} \vec{Q}^{n+1}, \vec{\varphi}_Q \right\rangle_{E,N} = \int_{\partial E,N} \left(\Phi^{\star,n+1} - \Phi^{n+1} \right) \tilde{\boldsymbol{\varphi}}_Q \cdot \hat{n} \, \mathrm{d}S_\xi + \left\langle \nabla_\xi \Phi^{n+1}, \tilde{\boldsymbol{\varphi}}_Q \right\rangle_{E,N}, \tag{93a}$$

$$\left\langle \mathcal{J} \vec{Q}^{n}, \vec{\varphi}_Q \right\rangle_{E,N} = \int_{\partial E,N} \left(\Phi^{\star,n} - \Phi^{n} \right) \tilde{\boldsymbol{\varphi}}_Q \cdot \hat{n} \, \mathrm{d}S_\xi + \left\langle \nabla_\xi \Phi^{n}, \tilde{\boldsymbol{\varphi}}_Q \right\rangle_{E,N}. \tag{93b}$$

Then we subtract (93b) from (93a), divide the result by Δt (note that we have defined $\Delta\Phi = \Phi^{n+1} - \Phi^n$ and $\Delta\vec{Q} = \vec{Q}^{n+1} - \vec{Q}^n$),

$$\left\langle \mathcal{J} \frac{\Delta \vec{Q}}{\Delta t}, \vec{\varphi}_Q \right\rangle_{E,N} = \int_{\partial E,N} \left(\frac{\Delta\Phi^\star}{\Delta t} - \frac{\Delta\Phi}{\Delta t} \right) \tilde{\boldsymbol{\varphi}}_Q \cdot \hat{n} \, \mathrm{d}S_\xi + \left\langle \frac{\nabla_\xi (\Delta\Phi)}{\Delta t}, \tilde{\boldsymbol{\varphi}}_Q \right\rangle_{E,N}, \tag{94}$$

and we set $\vec{\varphi}_Q = \vec{Q}^{n+1}$ in (94) to obtain

$$\left\langle \mathcal{J} \frac{\Delta \vec{Q}}{\Delta t}, \vec{Q}^{n+1} \right\rangle_{E,N} = \int_{\partial E,N} \left(\frac{\Delta\Phi^\star}{\Delta t} - \frac{\Delta\Phi}{\Delta t} \right) \vec{\tilde{Q}}^{n+1} \cdot \hat{n} \, \mathrm{d}S_\xi + \left\langle \frac{\nabla_\xi (\Delta\Phi)}{\Delta t}, \vec{\tilde{Q}}^{n+1} \right\rangle_{E,N}. \tag{95}$$

Next, we set $\varphi_W = \Delta\Phi/\Delta t = \left(\Phi^{n+1} - \Phi^n \right)/\Delta t$ in (92c),

$$\begin{aligned} \left\langle \mathcal{J} W^\theta, \frac{\Delta\Phi}{\Delta t} \right\rangle_{E,N} = & \left\langle \left(\frac{\mathrm{d}\Psi}{\mathrm{d}\Phi} \right)^n + S_0 \Delta\Phi, \mathcal{J} \frac{\Delta\Phi}{\Delta t} \right\rangle_{E,N} \\ & - k \int_{\partial E,N} \frac{\Delta\Phi}{\Delta t} \vec{\tilde{Q}}^{\star,n+1} \cdot \hat{n} \, \mathrm{d}S_\xi + k \left\langle \vec{\tilde{Q}}^{n+1}, \frac{\nabla_\xi (\Delta\Phi)}{\Delta t} \right\rangle_{E,N}, \end{aligned} \tag{96}$$

and replace the last inner product in (96) by that in (95),

$$\begin{aligned}\left\langle \mathcal{J}W^{\theta}, \frac{\Delta\Phi}{\Delta t}\right\rangle_{E,N} = &\left\langle \left(\frac{\mathrm{d}\Psi}{\mathrm{d}\Phi}\right)^{n} + S_0\Delta\Phi, \mathcal{J}\frac{\Delta\Phi}{\Delta t}\right\rangle_{E,N} + k\left\langle \mathcal{J}\frac{\Delta\vec{Q}}{\Delta t}, \vec{Q}^{n+1}\right\rangle_{E,N} \\ &- k\int_{\partial E,N}\left(\frac{\Delta\Phi}{\Delta t}\vec{\tilde{Q}}^{\star,n+1}\cdot\hat{n} + \left(\frac{\Delta\Phi^{\star}-\Delta\Phi}{\Delta t}\right)\vec{\tilde{Q}}^{n+1}\cdot\hat{n}\right)\mathrm{d}S_{\xi}.\end{aligned} \tag{97}$$

In a more compact form, we denote the boundary terms as $\mathrm{BT}_{E,N}$,

$$\begin{aligned}\left\langle \mathcal{J}W^{\theta}, \frac{\Delta\Phi}{\Delta t}\right\rangle_{E,N} = &\left\langle \left(\frac{\mathrm{d}\Psi}{\mathrm{d}\Phi}\right)^{n} + S_0\Delta\Phi, \mathcal{J}\frac{\Delta\Phi}{\Delta t}\right\rangle_{E,N} + k\left\langle \mathcal{J}\frac{\Delta\vec{Q}}{\Delta t}, \vec{Q}^{n+1}\right\rangle_{E,N} \\ &- k\mathrm{BT}_{E,N}\left(\frac{\Delta\Phi}{\Delta t}, \vec{Q}^{n+1}\right).\end{aligned} \tag{98}$$

Next, we combine (92a) and (92b). To do so, we set $\vec{\varphi}_{\vec{F}} = \mathcal{I}^{N}\left[M\vec{F}^{\theta}\right]$ in (92b) (note we drop the $\mathcal{I}^{N}$ operator since the quadrature only requires nodal values, that is $\left\langle \mathcal{I}^{N}\left(M\vec{F}^{\theta}\right), \vartheta\right\rangle_{E,N} = \left\langle M\vec{F}^{\theta}, \vartheta\right\rangle_{E,N}$),

$$\left\langle \mathcal{J}\vec{F}^{\theta}, M\vec{F}^{\theta}\right\rangle_{E,N} = \int_{\partial E,N}\left(W^{\star,\theta} - W^{\theta}\right)M\vec{\tilde{F}}^{\theta}\cdot\hat{n}\,\mathrm{d}S_{\xi} + \left\langle \nabla_{\xi}W^{\theta}, M\vec{\tilde{F}}^{\theta}\right\rangle_{E,N}, \tag{99}$$

and we set $\varphi_{\Phi} = W^{\theta}$ in (92a),

$$\left\langle \mathcal{J}\frac{\Delta\Phi}{\Delta t}, W^{\theta}\right\rangle_{E,N} = \int_{\partial E,N} W^{\theta}\left(M\vec{\tilde{F}}\right)^{\star,\theta}\cdot\hat{n}\,\mathrm{d}S_{\xi} - \left\langle M\vec{\tilde{F}}^{\theta}, \nabla_{\xi}W^{\theta}\right\rangle_{E,N}. \tag{100}$$

We then sum (99) and (100),

$$\left\langle \mathcal{J}\frac{\Delta\Phi}{\Delta t}, W^{\theta}\right\rangle_{E,N} = -\left\langle \mathcal{J}\vec{F}^{\theta}, M\vec{F}^{\theta}\right\rangle_{E,N} + \mathrm{BT}_{E,N}\left(W^{\theta}, M\vec{F}^{\theta}\right). \tag{101}$$

The final step is to combine (98) and (101). Since they share their left hand sides, we equate both right hand sides and multiply them by the time step Δt,

$$\begin{aligned}&\left\langle \left(\frac{\mathrm{d}\Psi}{\mathrm{d}\Phi}\right)^{n} + S_0\Delta\Phi, \mathcal{J}\Delta\Phi\right\rangle_{E,N} + k\left\langle \mathcal{J}\Delta\vec{Q}, \vec{Q}^{n+1}\right\rangle_{E,N} \\ &= -\Delta t\left\langle \mathcal{J}\vec{F}^{\theta}, M\vec{F}^{\theta}\right\rangle_{E,N} + k\mathrm{BT}_{E,N}\left(\Delta\Phi, \vec{Q}^{n+1}\right) + \Delta t\mathrm{BT}_{E,N}\left(W^{\theta}, M\vec{F}^{\theta}\right).\end{aligned} \tag{102}$$

We then perform manipulations on the left hand side to get the free–energy $\mathcal{F}$. First, we perform the Taylor expansion of $\Psi(\Phi)$ centered on Φ^{n},

$$\Psi^{n+1} = \Psi^{n} + \left(\frac{\mathrm{d}\Psi}{\mathrm{d}\Phi}\right)^{n}\Delta\Phi + \frac{1}{2}\left(\frac{\mathrm{d}^2\Psi}{\mathrm{d}\Phi^2}\right)^{n}\Delta\Phi^2 + \frac{1}{6}\left(\frac{\mathrm{d}^3\Psi}{\mathrm{d}\Phi^3}\right)^{n}\Delta\Phi^3 + \frac{1}{24}\left(\frac{\mathrm{d}^4\Psi}{\mathrm{d}\Phi^4}\right)^{n}\Delta\Phi^4 + ... \tag{103}$$

so, the expression for the chemical energy becomes

$$\Psi^{n+1} = \Psi^{n} + \left(\frac{\mathrm{d}\Psi}{\mathrm{d}\Phi}\right)^{n}\Delta\Phi - \frac{1}{2}\left(1 - 3\left(\Phi^{n}\right)^2\right)\Delta\Phi^2 + \Phi^{n}\Delta\Phi^3 + \frac{1}{4}\Delta\Phi^4. \tag{104}$$

We use (104) to write the first volume quadrature in (102) as

$$\left\langle \left(\frac{\mathrm{d}\Psi}{\mathrm{d}\Phi}\right)^{n} + S_0\Delta\Phi, \mathcal{J}\Delta\Phi\right\rangle_{E,N} = \left\langle \mathcal{J}\Psi^{n+1}, 1\right\rangle_{E,N} - \left\langle \mathcal{J}\Psi^{n}, 1\right\rangle_{E,N} + \left\langle \mathcal{J}\Pi, 1\right\rangle_{E,N}, \tag{105}$$

where $\Pi\left(\Phi^{n+1},\Phi^{n}\right)$ is the polynomial function

$$\Pi = S_0\Delta\Phi^2 + \frac{1}{2}\left(1-3\left(\Phi^n\right)^2\right)\Delta\Phi^2 - \Phi^n\Delta\Phi^3 - \frac{1}{4}\Delta\Phi^4 = \Delta\Phi^2\Pi^\star, \tag{106}$$

with

$$\Pi^\star = S_0 + \frac{1}{2}\left(1-3\left(\Phi^n\right)^2\right) - \Phi^n\left(\Phi^{n+1}-\Phi^n\right) - \frac{1}{4}\left(\Phi^{n+1}-\Phi^n\right)^2. \tag{107}$$

For the time integration to be stable, it has been shown in [8] that Π^* should remain positive. It has been proven that a value of $S_0 = 1$ suffice for $\Phi \in [-1,1]$ so that the system is stable. In all the numerical experiments conducted in this work, a value of $S_0 = 1$ has been chosen. A more detailed analysis as well as numerical experiments for different values of Φ is presented in [8].
For the interface energy in (102), we complete the square,

$$\begin{aligned}\left\langle \mathcal{J}\Delta\vec{Q},\vec{Q}^{n+1}\right\rangle_{E,N} = \quad & \frac{1}{2}\left\langle \mathcal{J}\vec{Q}^{n+1},\vec{Q}^{n+1}\right\rangle_{E,N} - \frac{1}{2}\left\langle \mathcal{J}\vec{Q}^{n},\vec{Q}^{n}\right\rangle_{E,N} \\ & + \frac{1}{2}\left\langle \mathcal{J}\Delta\vec{Q},\Delta\vec{Q},\right\rangle_{E,N},\end{aligned} \tag{108}$$

and we place (105) and (108) in (102),

$$\begin{aligned}&\left\langle \mathcal{J}\left(\Psi^{n+1}+\frac{1}{2}k\vec{Q}^{n+1}\cdot\vec{Q}^{n+1}\right),1\right\rangle_{E,N} - \left\langle \mathcal{J}\left(\Psi^{n}+\frac{1}{2}k\vec{Q}^{n}\cdot\vec{Q}^{n}\right),1\right\rangle_{E,N} \\ &= -\Delta t\left\langle \mathcal{J}\vec{F}^{\theta},M\vec{F}^{\theta}\right\rangle_{E,N} + k\mathrm{BT}_{E,N}\left(\Delta\Phi,\vec{Q}^{n+1}\right) \\ &+ \Delta t\mathrm{BT}_{E,N}\left(W^{\theta},M\vec{F}^{\theta}\right) - \left\langle \mathcal{J}\Pi,1\right\rangle_{E,N} - \frac{1}{2}k\left\langle \mathcal{J}\Delta\vec{Q},\Delta\vec{Q}\right\rangle_{E,N}.\end{aligned} \tag{109}$$

We define the discrete volumetric free–energy in an element as,

$$\mathcal{F}_v^{n,E,N} = \left\langle \mathcal{J}\left(\Psi^{n}+\frac{1}{2}k\vec{Q}^{n}\cdot\vec{Q}^{n}\right),1\right\rangle_{E,N}, \tag{110}$$

which simplifies (109) to

$$\begin{aligned}\mathcal{F}_v^{n+1,E,N} - \mathcal{F}_v^{n,E,N} = & -\Delta t\left\langle \mathcal{J}\vec{F}^{\theta},M\vec{F}^{\theta}\right\rangle_{E,N} + k\mathrm{BT}_{E,N}\left(\Delta\Phi,\vec{Q}^{n+1}\right) \\ & + \Delta t\mathrm{BT}_{E,N}\left(W^{\theta},M\vec{F}^{\theta}\right) - \left\langle \mathcal{J}\Pi,1\right\rangle_{E,N} \\ & - \frac{1}{2}k\left\langle \mathcal{J}\Delta\vec{Q},\Delta\vec{Q}\right\rangle_{E,N}.\end{aligned} \tag{111}$$

Eq. (111) shows that free–energy changes are due to physical dissipation in the element interior by the term $\Delta t\left\langle \mathcal{J}\vec{F}^{\theta},M\vec{F}^{\theta}\right\rangle_{E,N} \geqslant 0$ (the discrete counterpart of that obtained for the continuous analysis (15)), numerical dissipation as a result of the IMEX scheme,

$$\mathrm{diss}_{\mathrm{IMEX}}^{E,N} = -\left\langle \mathcal{J}\Pi,1\right\rangle_{E,N} - \frac{1}{2}k\left\langle \mathcal{J}\Delta\vec{Q},\Delta\vec{Q}\right\rangle_{E,N} \leqslant 0, \tag{112}$$

and boundary exchanges through all $\mathrm{BT}_{E,N}$ terms.
The effect of boundary exchanges can only be studied from the perspective of all elements in the domain. So we sum (111) for all mesh elements,

$$\mathcal{F}_v^{n+1,N} - \mathcal{F}_v^{n,N} = -\Delta t\sum_e\left\langle \mathcal{J}\vec{F}^{\theta},M\vec{F}^{\theta}\right\rangle_{E,N} + \mathrm{IBT}_N + \mathrm{PBT}_N + \sum_e \mathrm{diss}_{\mathrm{IMEX}}^{E,N}, \tag{113}$$

where $\mathcal{F}_v^{n,N} = \sum_e \mathcal{F}_v^{n,E,N}$ is the sum of all element volumetric free–energies.

References

[1] Mirjalili S, Jain S, Dodd M. Interface-capturing methods for two-phase flows: An overview and recent developments. Center for Turbulence Research - Annual research brief. 2017 12:117–135.

[2] Cahn JW, Hilliard JE. Free energy of a nonuniform system. I. Interfacial free energy. The Journal of chemical physics. 1958;28(2):258–267.

[3] Kim J, Lee S, Choi Y, Lee SM, Jeong D. Basic principles and practical applications of the Cahn–Hilliard equation. Mathematical Problems in Engineering. 2016;2016.

[4] Shen J, Yang X. A phase-field model and its numerical approximation for two-phase incompressible flows with different densities and viscosities. SIAM Journal on Scientific Computing. 2010;32(3):1159–1179.

[5] Yue P, Zhou C, Feng JJ. Sharp-interface limit of the Cahn-Hilliard model for moving contact lines. Journal of Fluid Mechanics. 2010;645:279.

[6] Khatavkar V, Anderson P, Meijer H. On scaling of diffuse–interface models. Chemical engineering science. 2006;61(8):2364–2378.

[7] Caginalp G, Chen X. Convergence of the phase field model to its sharp interface limits. European Journal of Applied Mathematics. 1998;9(4):417–445.

[8] Manzanero J, Rubio G, Kopriva DA, Ferrer E, Valero E. A free–energy stable nodal discontinuous Galerkin approximation with summation–by–parts property for the Cahn–Hilliard equation. Journal of Computational Physics. 2020;403:109072.

[9] Manzanero J, Rubio G, Kopriva DA, Ferrer E, Valero E. Entropy–stable discontinuous Galerkin approximation with summation–by–parts property for the incompressible Navier–Stokes/Cahn–Hilliard system. Journal of Computational Physics. 2020:109363.

[10] Xia Y, Xu Y, Shu CW. Local discontinuous Galerkin methods for the Cahn–Hilliard type equations. Journal of Computational Physics. 2007;227(1):472–491.

[11] Wells GN, Kuhl E, Garikipati K. A discontinuous Galerkin method for the Cahn–Hilliard equation. Journal of Computational Physics. 2006;218(2):860–877.

[12] Frank F, Liu C, Alpak FO, Riviere B. A finite volume/discontinuous Galerkin method for the advective Cahn–Hilliard equation with degenerate mobility on porous domains stemming from micro-CT imaging. Computational Geosciences. 2018;22(2):543–563.

[13] Kay D, Styles V, Süli E. Discontinuous Galerkin finite element approximation of the Cahn–Hilliard equation with convection. SIAM Journal on Numerical Analysis. 2009;47(4):2660–2685.

[14] Aristotelous AC, Karakashian OA, Wise SM. Adaptive, second-order in time, primitive-variable discontinuous Galerkin schemes for a Cahn–Hilliard equation with a mass source. IMA Journal of Numerical Analysis. 2015;35(3):1167–1198.

[15] Fisher TC, Carpenter MH. High-order entropy stable finite difference schemes for nonlinear conservation laws: Finite domains. Journal of Computational Physics. 2013;252:518–557.

[16] Carpenter MH, Fisher TC, Nielsen EJ, Frankel SH. Entropy stable spectral collocation schemes for the Navier–Stokes equations: Discontinuous interfaces. SIAM Journal on Scientific Computing. 2014;36(5):B835–B867.

[17] Kopriva DA, Gassner GJ. An energy stable discontinuous Galerkin spectral element discretization for variable coefficient advection problems. SIAM Journal on Scientific Computing. 2014;36(4):A2076–A2099.

[18] Gassner GJ, Winters AR, Kopriva DA. Split form nodal discontinuous Galerkin schemes with summation-by-parts property for the compressible Euler equations. Journal of Computational Physics. 2016;327:39–66.

[19] Gassner GJ, Winters AR, Hindenlang FJ, Kopriva DA. The BR1 scheme is stable for the compressible Navier–Stokes equations. Journal of Scientific Computing. 2018;77(1):154–200.

[20] Manzanero J, Rubio G, Kopriva DA, Ferrer E, Valero E. Entropy–stable discontinuous Galerkin approximation with summation–by–parts property for the incompressible Navier–Stokes/Cahn–Hilliard system. Journal of Computational Physics. 2020:109363.

[21] Chen T, Shu CW. Entropy stable high order discontinuous Galerkin methods with suitable quadrature rules for hyperbolic conservation laws. Journal of Computational Physics. 2017;345:427–461.

[22] Winters AR, Moura RC, Mengaldo G, Gassner GJ, Walch S, Peiro J, et al. A comparative study on polynomial dealiasing and split form discontinuous Galerkin schemes for underresolved turbulence computations. Journal of Computational Physics. 2018;372:1–21.

[23] Gassner GJ. A skew-symmetric discontinuous Galerkin spectral element discretization and its relation to SBP-SAT finite difference methods. SIAM Journal on Scientific Computing. 2013;35(3):A1233–A1253.

[24] Chen T, Shu CW. Review of entropy stable discontinuous Galerkin methods for systems of conservation laws on unstructured simplex meshes; 2020.

[25] Mattsson K, Carpenter MH. Stable and accurate interpolation operators for high-order multiblock finite difference methods. SIAM Journal on Scientific Computing. 2010;32(4):2298–2320.

[26] Kozdon JE, Wilcox LC. Stable coupling of nonconforming, high-order finite difference methods. SIAM Journal on Scientific Computing. 2016;38(2):A923–A952.

[27] Nissen A, Kormann K, Grandin M, Virta K. Stable difference methods for block-oriented adaptive grids. Journal of Scientific Computing. 2015;65(2):486–511.

[28] Almquist M, Wang S, Werpers J. Order-preserving interpolation for summation-by-parts operators at nonconforming grid interfaces. SIAM Journal on Scientific Computing. 2019;41(2):A1201–A1227.

[29] Kozdon JE, Wilcox LC. An energy stable approach for discretizing hyperbolic equations with nonconforming discontinuous Galerkin methods. Journal of Scientific Computing. 2018;76(3):1742–1784.

[30] Carpenter MH, Parsani M, Nielsen EJ, Fisher TC. Towards an entropy stable spectral element framework for computational fluid dynamics. In: 54th AIAA Aerospace Sciences Meeting; 2016. p. 1058.

[31] Parsani M, Carpenter MH, Fisher TC, Nielsen EJ. Entropy stable staggered grid discontinuous spectral collocation methods of any order for the compressible Navier–Stokes equations. SIAM Journal on Scientific Computing. 2016;38(5):A3129–A3162.

[32] Parsani M, Carpenter MH, Nielsen EJ. Entropy stable discontinuous interfaces coupling for the three-dimensional compressible Navier-Stokes equations. J Comput Phys. 2015;290(1):132–138.

[33] Friedrich L, Winters AR, Fernández DCDR, Gassner GJ, Parsani M, Carpenter MH. An entropy stable h/p non-conforming discontinuous Galerkin method with the summation-by-parts property. Journal of Scientific Computing. 2018;77(2):689–725.

[34] Fernández DCDR, Carpenter MH, Dalcin L, Fredrich L, Winters AR, Gassner GJ, et al. Entropy-stable p-nonconforming discretizations with the summation-by-parts property for the compressible Navier–Stokes equations. Computers & Fluids. 2020:104631.

[35] Chan J, Bencomo M, Fernández DC. Mortar-based entropy-stable discontinuous Galerkin methods on non-conforming quadrilateral and hexahedral meshes. arXiv preprint arXiv:200503237. 2020.

[36] Shadpey S, Zingg DW. Entropy-Stable Multidimensional Summation-by-Parts Discretizations on hp-Adaptive Curvilinear Grids for Hyperbolic Conservation Laws. Journal of Scientific Computing. 2020;82(3):1–46.

[37] Stogner RH, Carey GF, Murray BT. Approximation of Cahn–Hilliard diffuse interface models using parallel adaptive mesh refinement and coarsening with C1 elements. International journal for numerical methods in engineering. 2008;76(5):636–661.

[38] Yue P, Zhou C, Feng JJ, Ollivier-Gooch CF, Hu HH. Phase-field simulations of interfacial dynamics in viscoelastic fluids using finite elements with adaptive meshing. Journal of Computational Physics. 2006;219(1):47–67.

[39] Ceniceros HD, Roma AM. A nonstiff, adaptive mesh refinement-based method for the Cahn–Hilliard equation. Journal of Computational Physics. 2007;225(2):1849–1862.

[40] Chen Y, Shen J. Efficient, adaptive energy stable schemes for the incompressible Cahn–Hilliard Navier–Stokes phase-field models. Journal of Computational Physics. 2016;308:40–56.

[41] Wackers J, Deng G, Guilmineau E, Leroyer A, Queutey P, Visonneau M. Combined refinement criteria for anisotropic grid refinement in free-surface flow simulation. Computers & Fluids. 2014;92:209–222.

[42] Ginzburg I, Wittum G. Two-phase flows on interface refined grids modeled with VOF, staggered finite volumes, and spline interpolants. Journal of Computational Physics. 2001;166(2):302–335.

[43] Agresar G, Linderman J, Tryggvason G, Powell K. An adaptive, Cartesian, front-tracking method for the motion, deformation and adhesion of circulating cells. Journal of Computational Physics. 1998;143(2):346–380.

[44] Barosan I, Anderson P, Meijer H. Application of mortar elements to diffuse-interface methods. Computers & fluids. 2006;35(10):1384–1399.

[45] Compere G, Marchandise E, Remacle JF. Transient adaptivity applied to two-phase incompressible flows. Journal of computational physics. 2008;227(3):1923–1942.

[46] Goudenège L, Martin D, Vial G. High order finite element calculations for the Cahn-Hilliard equation. Journal of Scientific Computing. 2012;52(2):294–321.

[47] Mavriplis C. A posteriori error estimators for adaptive spectral element techniques. In: Proceedings of the Eighth GAMM-Conference on Numerical Methods in Fluid Mechanics. Springer; 1990. p. 333–342.

[48] Henderson R. Unstructured spectral element methods: Parallel algorithms and simulations(Ph. D. Thesis). 1994.

[49] Theodorakakos A, Bergeles G. Simulation of sharp gas–liquid interface using VOF method and adaptive grid local refinement around the interface. International Journal for Numerical Methods in Fluids. 2004;45(4):421–439.

[50] Baňas L, Nürnberg R. Adaptive finite element methods for Cahn–Hilliard equations. Journal of computational and applied mathematics. 2008;218(1):2–11.

[51] Hintermüller M, Hinze M, Kahle C. An adaptive finite element Moreau–Yosida-based solver for a coupled Cahn–Hilliard/Navier–Stokes system. Journal of Computational Physics. 2013;235:810–827.

[52] Hintermüller M, Hinze M, Tber MH. An adaptive finite-element Moreau–Yosida-based solver for a non-smooth Cahn–Hilliard problem. Optimization Methods and Software. 2011;26(4-5):777–811.

[53] Liu C, Frank F, Rivière BM. Numerical error analysis for nonsymmetric interior penalty discontinuous Galerkin method of Cahn–Hilliard equation. Numerical Methods for Partial Differential Equations. 2019;35(4):1509–1537.

[54] Rubio G, Fraysse F, Kopriva DA, Valero E. Quasi-A Priori Truncation Error Estimation in the DGSEM. Journal of Scientific Computing. 2015;64:425–455.

[55] Rueda-Ramírez AM, Rubio G, Ferrer E, Valero E. Truncation Error Estimation in the p-Anisotropic Discontinuous Galerkin Spectral Element Method. Journal of Scientific Computing. 2019;78(1):433–466.

[56] Kompenhans M, Rubio G, Ferrer E, Valero E. Adaptation strategies for high order discontinuous Galerkin methods based on Tau-estimation. Journal of Computational Physics. 2016;306:216 – 236.

[57] Kompenhans M, Rubio G, Ferrer E, Valero E. Comparisons of p-adaptation strategies based on truncation- and discretisation-errors for high order discontinuous Galerkin methods. Computers & Fluids. 2016;139:36 – 46.

[58] Kopriva DA, Woodruff SL, Hussaini MY. Computation of electromagnetic scattering with a non-conforming discontinuous spectral element method. International journal for numerical methods in engineering. 2002;53(1):105–122.

[59] Dong S. An outflow boundary condition and algorithm for incompressible two-phase flows with phase field approach. Journal of Computational Physics. 2014;266:47–73.

[60] Yang Z, Lin L, Dong S. A family of second-order energy-stable schemes for Cahn–Hilliard type equations. Journal of Computational Physics. 2019;383:24–54.

[61] Gal CG, Wu H. Asymptotic behavior of a Cahn-Hilliard equation with Wentzell boundary conditions and mass conservation. Discrete & Continuous Dynamical Systems-A. 2008;22(4):1041.

[62] Copetti M, Elliott CM. Numerical analysis of the Cahn-Hilliard equation with a logarithmic free energy. Numerische Mathematik. 1992;63(1):39–65.

[63] Rueda-Ramírez AM, Manzanero J, Ferrer E, Rubio G, Valero E. A p-multigrid strategy with anisotropic p-adaptation based on truncation errors for high-order discontinuous Galerkin methods. Journal of Computational Physics. 2019;378:209–233.

[64] Kopriva DA. Implementing spectral methods for partial differential equations: Algorithms for scientists and engineers. Springer Science & Business Media; 2009.

[65] Kay D, Styles V, Süli E. Discontinuous Galerkin finite element approximation of the Cahn–Hilliard equation with convection. SIAM Journal on Numerical Analysis. 2009;47(4):2660–2685.

[66] Kopriva DA, Hindenlang FJ, Bolemann T, Gassner GJ. Free-Stream Preservation for Curved Geometrically Non-conforming Discontinuous Galerkin Spectral Elements. Journal of Scientific Computing. 2019;79(3):1389–1408.

[67] Kopriva DA. Metric identities and the discontinuous spectral element method on curvilinear meshes. Journal of Scientific Computing. 2006;26(3):301.

[68] Kopriva DA. 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. p. 21–40.

[69] Friedrich L, Fernández DCDR, Winters AR, Gassner GJ, Zingg DW, Hicken J. Conservative and stable degree preserving SBP operators for non-conforming meshes. Journal of Scientific Computing. 2018;75(2):657–686.

[70] Bassi F, Rebay S. A High-Order Accurate Discontinuous Finite Element Method for the Numerical Solution of the Compressible Navier–Stokes Equations. Journal of Computational Physics. 1997;131(2):267–279.

[71] Dong S. Multiphase flows of N immiscible incompressible fluids: a reduction-consistent and thermodynamically-consistent formulation and associated algorithm. Journal of Computational Physics. 2018;361:1–49.

[72] Jacqmin D. Calculation of two-phase Navier–Stokes flows using phase-field modeling. Journal of Computational Physics. 1999;155(1):96–127.

[73] Boyanova P, Neytcheva M. Efficient numerical solution of discrete multi-component Cahn–Hilliard systems. Computers & Mathematics with Applications. 2014;67(1):106–121.

[74] Yue P, Feng JJ, Liu C, Shen J. A diffuse-interface method for simulating two-phase flows of complex fluids. Journal of Fluid Mechanics. 2004;515:293.

[75] Teigen KE, Song P, Lowengrub J, Voigt A. A diffuse-interface method for two-phase flows with soluble surfactants. Journal of computational physics. 2011;230(2):375–393.

[76] Manzanero J, Rueda-Ramírez AM, Rubio G, Ferrer E. 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. 2018;363:1–10.

[77] Bohm M, Winters AR, Gassner GJ, Derigs D, Hindenlang F, Saur J. An entropy stable nodal discontinuous Galerkin method for the resistive MHD equations. Part I: Theory and numerical verification. Journal of Computational Physics. 2018:108076.

[78] Hindenlang F, Bolemann T, Munz CD. Mesh Curving Techniques for High Order Discontinuous Galerkin Simulations. In: IDIHOM: Industrialization of High-Order Methods-A Top-Down Approach. Springer; 2015. p. 133–152.

[79] Williamson J. Low-storage runge-kutta schemes. Journal of Computational Physics. 1980;35(1):48–56.

[80] Jokisaari AM, Voorhees PW, Guyer JE, Warren J, Heinonen O. Benchmark problems for numerical implementations of phase field models. Computational Materials Science. 2017;126:139–151.