A stabilized finite element method for modeling dispersed multiphase flows using orthogonal subgrid scales

Hauke Gravenkamp[a,*], Ramon Codina[a,b], Javier Principe[a,b]

[a] *International Centre for Numerical Methods in Engineering, 08034 Barcelona, Spain*
[b] *Universitat Politècnica de Catalunya, 08034 Barcelona, Spain*

Abstract

We propose a finite-element formulation for simulating multi-component flows occupying the same domain with spatially varying concentrations. Each constituent is assumed to behave as an incompressible Newtonian fluid, and solutions are sought for the velocities and volume fractions of each phase, as well as the common pressure. Stabilization terms are derived within the framework of the variational multiscale method based on an approximation of the finite-element residual to achieve control of the pressure and volume fractions. We utilize the concept of term-by-term stabilization in conjunction with orthogonal subgrid scales, thus incorporating only those terms of the residual essential to obtain stability and projecting them on a space orthogonal to the finite element space. The resulting system of equations is solved in a monolithic manner, requiring a small number of nonlinear iterations. Several benchmark tests have been performed to confirm the stability and optimal asymptotic convergence rates for linear and higher-order elements using the proposed formulation.

Keywords: multiphase flow; finite elements; variational multiscale method; orthogonal subgrid scales; dispersed flow

1. Introduction

Multiphase flow phenomena are ubiquitous in nature and engineering whenever fluids with different properties can mix or interact. Common examples include sprays (droplets in air) or liquids containing bubbles. Although the terminology is often employed to some extent ambiguously, for the purpose of this discussion, we will refer by multiphase flow to any flow problem that entails multiple components – irrespective of whether or not they are in the same state of matter (i.e., gaseous, liquid, or solid particles). An important distinction allows a categorization into two main classes of multiphase flows: *dispersed flows* form a mixture of several constituents that can occupy the same domain but in generally different concentrations that can depend on space and time. This is in contrast to *separated flows*, which assume different materials to occupy distinct subdomains (of potentially time-dependent topology), divided by well-defined interfaces. Both general flow types can be further classified based on the underlying model assumptions or the resulting

*Corresponding author
Email address: hgravenkamp@cimne.upc.edu (Hauke Gravenkamp)

behavior on the macroscale. Notably, in the case of separated flows, phenomenological distinctions are often made, resulting in more specific terminology, such as slurry flows, stratified flows, plug flows, or wavey flows. For dispersed flows, more specific models are obtained by assuming, for instance, a vanishing or small relative velocity (mixture models) [1] or a small number of solid particles that can be individually tracked [2]. A good overview of the different phenomena and interactions can be found in [3].

In this contribution, we focus on one specific, relatively common model for *dispersed multiphase flows*, describing a mixture of fluids whose components interact through drag terms and share a *common pressure*. Details on this model shall be provided later; in a nutshell, the governing equations for each phase k are of the form

$$\partial_t \alpha_k + \nabla\cdot(\alpha_k \mathbf{u}_k) = 0\,, \tag{1a}$$

$$\rho_k \alpha_k \partial_t \mathbf{u}_k + \rho_k \alpha_k \mathbf{u}_k\cdot\nabla\mathbf{u}_k - 2\mu_k \nabla\cdot(\alpha_k \boldsymbol{\varepsilon}(\mathbf{u}_k)) + \alpha_k \nabla p - \sum_\ell g_{k\ell}(\mathbf{u}_\ell - \mathbf{u}_k) = \alpha_k \mathbf{f}_k\,. \tag{1b}$$

Here and in the following, $\mathbf{u}_k$, $\boldsymbol{\varepsilon}(\mathbf{u}_k)$, and α_k denote the velocity, strain rate, and volume fraction of phase k; p is the pressure, ρ_k and μ_k are the mass density and viscosity, and $\mathbf{f}_k$ includes any body forces. In addition, there are drag terms, to be specified later, characterizing the momentum exchange between phase k and all other phases through $g_{k\ell}$ (which may be functions of $\mathbf{u}_\ell$, $\mathbf{u}_k$, α_ℓ, and α_k). It is usually assumed that $g_{k\ell} = g_{\ell k}$ and $g_{kk} = 0$.

Equations (1) resemble the Navier-Stokes equations describing an incompressible fluid. Apart from the mechanisms that cause the phases to interact, the major difference lies in the presence of the unknown volume fractions α_k, leading to several additional nonlinearities and thereby significantly complicating the solution. This and similar models have been frequently employed in numerical simulations, particularly using the finite volume method [4–6] and, to a much lesser extent, the finite element method [7–9]. When applying the finite element method to this problem, difficulties can be foreseen due to instabilities that are well-known in the context of the single-phase problem (Navier-Stokes equations). One source of instability stems from the mixed form of the problem, requiring inf-sub stable discretizations of velocities, pressure, and volume fractions.[1] In addition, strong oscillations can be expected in convection-dominated flows, which even occur in the simpler problem of the convection-diffusion-reaction equation [12]. Both issues can be vanquished simultaneously by employing the variational multiscale concept with a view to deriving additional stabilization terms to include in the discrete weak form of the problem at hand. Consequently, we design in this paper a stabilized finite element method for the above problem statement by extending results previously obtained for the case of a single fluid.

The underlying concept is based on the framework of the variational multiscale (VMS) method [13, 14] which relies on the idea of splitting the unknown field variables into a finite element component and a so-called subgrid-scale, i.e., those components that cannot be represented by the chosen finite element space. Different models for the subscales exist, leading to different categories of stabilized methods. We will focus on those approaches in which the subgrid scales are

[1] In the case of the single-phase problem, a frequently employed remedy consists in utilizing interpolants of lower polynomial degree for the pressure than for the velocity, yielding an inf-sub stable Galerkin-type method. For the multiphase problem, there is, to the authors' knowledge, no similarly straightforward choice of interpolation functions that would guarantee the stability of velocities, pressure, and volume fractions. Other attempts to stabilize similar multiphase problems involve stabilization by hierarchical decomposition [10], employing the Control Volume Method for the volume fractions [8], or simply adding artificial diffusion [11].

expressed in the same basis as the finite element components. Formally, the exact problem for the subgrid scales involves inverting a differential operator, and the approximation of the same leads to the stabilization parameters on which the formulation depends. In general, these methods have a residual-based structure, and consequently, the stabilization terms involve (an approximation of) the residual of the discretized governing equations. We will utilize in this work a variant of this technique in which the stabilization terms are based on the projection of the residual orthogonal to the finite element space. This approach, first introduced in [15], is commonly termed as *orthogonal subgrid scales*. One of the main advantages of this formulation is that it allows omitting terms in the residual that are not crucial for achieving stability without sacrificing optimal convergence of the method [16]. This concept is also known as *term-by-term stabilization* [17–19]. We will see that this property is extremely beneficial in stabilizing the multiphase problem where the inclusion of the complete residual would require a large number of nonlinear terms, few of which are necessary to guarantee stability.

We may also note that we will use the concept of *dynamic subgrid scales* [20]. In transient problems, the subgrid scales are naturally time-dependent and are obtained as solutions to evolution equations. For the subscales of velocities and volume fractions, these equations involve temporal derivatives of the subscales. Oftentimes, the term involving time derivatives is neglected, in which case the approach is referred to as *quasi-static*. Here, we will solve said evolution equation by a time-stepping scheme, namely the first-order backward difference formula (BDF1). Dynamic subgrid scales have previously been found to yield a more robust method and frequently decrease computational costs by reducing the number of iterations in each time step [19]. In addition, it is known that quasi-static subgrid scales can result in substantial errors if the time step is small [19, 21, 22].

Lastly, let us remark that we will solve the resulting system of equations in a *monolithic* way, which has been suggested in the context of a related stabilized method [7] but is in contrast to the vast majority of works in this field both in the context of finite volume and finite element methods. The motivation for employing monolithic schemes stems from the hope for significantly faster convergence compared to their sequential counterparts. However, we will refrain from detailed comparisons between both paradigms at this stage.

The objective of this work is rather to set the numerical framework and to establish a new approach to solving multiphase problems. We will discuss the relevant features of this formulation in quite some detail and explain various choices we made in the implementation. On the other hand, our numerical examples are, for now, restricted to relatively elementary cases involving two-dimensional geometries and two interacting phases. We will begin with a concise overview of the particular model in Section 2, highlighting the underlying assumptions and the scope of application. The variational form is briefly presented in Section 3 after which we derive the finite-element formulation, including its stabilization in Section 4. In the ensuing, we provide further inside into some details of the formulation and its implementation in Section 5. Lastly, Section 6 encompasses numerical examples, demonstrating optimal convergence of the proposed method with respect to spatial and temporal discretization and showing the applicability to typical flow problems involving moment exchange terms and buoyancy effects.

2. Multiphase model

In this section, we provide a succinct exposition of the utilized model. In particular, Section 2.1 illuminates the principal assumptions that underlie the microscale and the averaging procedure in

this general class of models. A more detailed discussion of this and other models can be found in [23]. In Section 2.2, we discuss an approach we propose to enforcing the constraint that the volume fractions must add up to unity.

2.1. Summary of model assumptions and averaging procedures

We consider dispersed flows consisting of different components, each of which behaving macroscopically like a fluid. While the following discussion generalizes to an arbitrary number of fluids, we have, so far, tested the formulation for systems of two components and will limit the numerical studies to this scenario. Furthermore, we restrict the discussion in this paper to isothermal flows. Consequently, we assume the standard balance equations to hold on the microscale; i.e., the velocities $\mathbf{u}_k$, densities ρ_k, and pressures p_k of each phase k satisfy

$$\partial_t \rho_k + \nabla\cdot(\rho_k \mathbf{u}_k) = 0\,, \tag{2a}$$

$$\partial_t (\rho_k \mathbf{u}_k) + \nabla\cdot(\rho_k \mathbf{u}_k \mathbf{u}_k) = -\nabla p_k + \nabla\cdot\tau_k + \mathbf{f}_k\,, \tag{2b}$$

where τ_i is the deviatoric stress tensor, and $\mathbf{f}_i$ denotes body loads. In topologically exact models, the above equations are coupled through interface conditions between domains occupied by different phases. Such interface conditions depend not only on the shape and local velocity of the interfaces but also on material parameters such as the interface tension coefficient and may potentially be affected by numerous phenomena such as mass transfer, phase transitions, chemical reactions, etc. One of the key aspects of dispersed multiphase models can be seen in the assumption that the typical size of the individual simply-connected domains occupied by the dispersed phase(s) is small compared to the total model domain. Hence, it becomes unfeasible – if not impossible – to track the exact geometry of each individual interface. Instead, averaging procedures are employed to describe the interaction of different phases on the macroscale. Here, we will refrain from discussing in length the different possibilities and their physical interpretation but rather stick to one particularly popular model that has been applied successfully to numerous engineering applications. According to [23], a common choice is to employ Favré averaging (denoted by $\bar{\bullet}$) for the velocity and partial $\langle\bullet\rangle$ or intrinsic $\widetilde{\bullet}$ averaging for the density and pressure, see Appendix A. This leads to the following variant of the balance equations averaged over a control volume

$$\partial_t (\alpha_k \widetilde{\rho}_k) + \nabla\cdot(\alpha_k \widetilde{\rho}_k \overline{\mathbf{u}}_k) = \Gamma_k\,, \tag{3a}$$

$$\partial_t (\alpha_k \widetilde{\rho}_k \overline{\mathbf{u}}_k) + \nabla\cdot(\alpha_k \widetilde{\rho}_k \overline{\mathbf{u}}_k \overline{\mathbf{u}}_k) = -\nabla(\alpha_k \widetilde{p}_k) + \nabla\cdot\langle\tau_k\rangle + \alpha_k \widetilde{\mathbf{f}}_k + \mathbf{M}_k + \nabla\cdot\langle\tau_{\delta k}\rangle\,. \tag{3b}$$

Here, the volume fractions are introduced as

$$\alpha_k = \frac{V_k}{V}\,,$$

where V_k denotes the volume occupied by phase k inside the control volume V. The averaging procedure invokes the so-called pseudo-turbulent stress tensor[2]

$$\langle\tau_{\delta k}\rangle = -\langle\rho_k \delta\mathbf{u}_k \delta\mathbf{u}_k\rangle\,, \tag{4}$$

[2]The pseudo-turbulent stress tensor resembles the turbulent stress in single-phase flows. In multiphase flows, however, this tensor is a consequence of the phase averaging and is generally nonzero even if the flow is laminar.

which involves the following definition of the velocity fluctuation

$$\mathbf{u}_k = \overline{\mathbf{u}}_k + \delta\mathbf{u}_k \,. \tag{5}$$

In addition, the averaged equations include transfer integrals representing the interaction across phase boundaries

$$\Gamma_k = -\frac{1}{V}\int_{A_k} \rho_k \left(\mathbf{u}_k - \mathbf{u}_\mathrm{A}\right)\cdot\mathbf{n}_k \,\mathrm{d}A \,, \tag{6}$$

$$\begin{aligned}\mathbf{M}_k = &-\frac{1}{V}\int_{A_k} \rho_k\mathbf{u}_k \left(\mathbf{u}_k - \mathbf{u}_\mathrm{A}\right)\cdot\mathbf{n}_k \,\mathrm{d}A \,, \\ &+\frac{1}{V}\int_{A_k} \left(-p_k\mathbb{1} + \tau_k\right)\cdot\mathbf{n}_k \,\mathrm{d}A\end{aligned} \tag{7}$$

with the velocity of the interface $\mathbf{u}_\mathrm{A}$ and the normal vector $\mathbf{n}_k$. Such transfer integrals are still defined based on the actual phase boundaries, which are generally unknown. The term Γ_k describes mass transfer between different phases (which we do not consider in the current work), while $\mathbf{M}_k$ is associated with momentum exchange. Approximating these terms as functions of the velocities, volume fractions, and pressures is an essential ingredient of the physical model and involves assumptions on all physical effects capable of influencing the interaction of different phases. In particular, when mass transfer between phases and interface tension can be neglected, the momentum transfer terms are often written in the form

$$\mathbf{M}_k = \widetilde{p}_k\nabla\alpha_k + \mathbf{D}_k$$

with

$$\mathbf{D}_k = \frac{1}{V}\int_A \left(\delta p_k\mathbb{1} - \tau_k\right)\cdot\mathbf{n}_k \,\mathrm{d}A \,. \tag{8}$$

Within the realm of this class of multiphase models, main differences arise in the approximation of the surface integrals, and different expressions for these terms can be obtained for scenarios such as liquid-particle suspensions or bubbly flow [23]. We will move forward assuming that there is no mass transfer between phases and that the momentum exchange term between phases k and ℓ is of the form $g_{k\ell}(\mathbf{u}_\ell - \mathbf{u}_k)$, where $g_{k\ell}$ can be a function of $\alpha_k, \alpha_\ell, \mathbf{u}_k, \mathbf{u}_\ell$. For conciseness, let us summarize the assumptions made in the current model as follows:

- Newtonian fluids
- Incompressible phases: ρ_k constant
- Common pressure between all phases:[3] $p_k = p$
- No mass transfer between phases: $\Gamma_k = 0$
- Neglect pseudo-turbulent stress: $\langle\tau_{\delta k}\rangle = \mathbf{0}$
- Momentum exchange due to drag only: $g_{k\ell}(\mathbf{u}_\ell - \mathbf{u}_k)$
- Neglect surface and interface tension
- Constant viscosities μ_k throughout the domain.

[3] While the common pressure is a typical assumption, particularly when interface tension can be neglected [24], it can, in other scenarios, lead to unphysical effects such as an overly large acceleration of a dispersed phase [10].

To avoid notational overload, we will omit the $\bar{\bullet}$ and $\widetilde{\bullet}$ symbols in what follows and assume the averaging procedures outlined before. Thus, we obtain

$$\partial_t \alpha_k + \nabla\cdot(\alpha_k \mathbf{u}_k) = 0\,, \tag{9a}$$

$$\rho_k \partial_t (\alpha_k \mathbf{u}_k) + \rho_k \nabla\cdot(\alpha_k \mathbf{u}_k \mathbf{u}_k) - 2\mu_k \nabla\cdot(\alpha_k \boldsymbol{\varepsilon}(\mathbf{u}_k)) + \alpha_k \nabla p - \sum_\ell g_{k\ell}(\mathbf{u}_\ell - \mathbf{u}_k) = \alpha_k \mathbf{f}_k\,. \tag{9b}$$

Equation (9b) can be written in a non-conservative form since

$$\begin{aligned}\rho_k \partial_t (\alpha_k \mathbf{u}_k) + \rho_k \nabla\cdot(\alpha_k \mathbf{u}_k \mathbf{u}_k) &= \rho_k \alpha_k [\partial_t \mathbf{u}_k + \mathbf{u}_k\cdot\nabla \mathbf{u}_k] + \rho_k \mathbf{u}_k [\partial_t \alpha_k + \nabla\cdot(\alpha_k \mathbf{u}_k)] \\ &= \rho_k \alpha_k [\partial_t \mathbf{u}_k + \mathbf{u}_k\cdot\nabla \mathbf{u}_k]\,,\end{aligned} \tag{10}$$

where the second equality holds because of Eq. (9a).

2.2. Reformulation of the algebraic constraint

Under the assumptions outlined before, the balance equations (9) for each phase resemble the incompressible Navier-Stokes equations of a single-phase flow. However, they involve the unknown volume fractions α_k, thus introducing additional nonlinearities. In order to close the model and obtain a well-posed problem, we need to make allowance for the algebraic constraint[4]

$$\sum_k \alpha_k = 1\,. \tag{11}$$

The most common approach lies in using the constraint equation to eliminate one of the volume fractions. However, this leads to a highly nonsymmetric system and introduces the arbitrariness of choosing one of the volume fractions to eliminate. Hence, we follow a different path in this work and enforce the constraint equation in a weak sense. Taking the time-derivative of Eq. (11) yields

$$\partial_t \sum_k \alpha_k = 0\,. \tag{12}$$

We introduce a time scale t_0 that shall be specified more explicitly later. We then multiply Eq. (11) by $-t_0^{-1}$ and add the continuity equations for all phases to obtain

$$\sum_k \nabla\cdot(\alpha_k \mathbf{u}_k) - t_0^{-1}\big(\sum_k \alpha_k - 1\big) = 0\,. \tag{13}$$

We will solve this equation together with the continuity and momentum equations. Clearly, Eqs. (9) and (11) imply (13). On the other hand, adding Eqs. (9a) for all phases and subtracting (13), we have

$$\partial_t \sum_k \alpha_k + t_0^{-1}\big(\sum_k \alpha_k - 1\big) = 0 \tag{14}$$

whose general solution is

$$\alpha(t) := \sum_k \alpha_k(t) = 1 + A\mathrm{e}^{-t/t_0} \tag{15}$$

[4]In some models of porous materials, this condition is not assumed in the presence of voids that are not explicitly modeled as a phase.

with a scalar constant A. Assuming that the initial conditions fulfill $\alpha(0) = 1$, we obtain $A = 0$. On the other hand, this condition may not be satisfied exactly, in particular, due to spatial approximation. Say, the initial condition is written as $\alpha(0) = 1 + \varepsilon$, then we have $A = \varepsilon$, and the error will decay within a time scale given by t_0:

$$\alpha(t) = 1 + \varepsilon \mathrm{e}^{-t/t_0} \,. \tag{16}$$

Including (13) as a closing equation fixes the strong form of the problem of interest. Let us summarize the resulting model.

2.3. Initial and boundary value problem

Let Ω be an open, bounded and polyhedral domain of $\mathbb{R}^{\mathrm{s}}$ ($\mathrm{s} = 2$ or 3), $\partial\Omega$ is the domain's boundary, and $[0, T]$ a time interval. The multiphase problem consists in finding the velocities $\mathbf{u}_k$ and volume fractions α_k for each phase k and a pressure p such that

$$\partial_t \alpha_k + \nabla\cdot(\alpha_k \mathbf{u}_k) = 0 \qquad \text{in } \Omega \times (0,T)\,, \tag{17a}$$

$$\rho_k \alpha_k \partial_t \mathbf{u}_k + \rho_k \alpha_k \mathbf{u}_k \cdot \nabla \mathbf{u}_k - 2\mu_k \nabla\cdot(\alpha_k \boldsymbol{\varepsilon}(\mathbf{u}_k)) + \alpha_k \nabla p - \sum_\ell g_{k\ell}(\mathbf{u}_\ell - \mathbf{u}_k) = \alpha_k \mathbf{f}_k \quad \text{in } \Omega \times (0,T)\,, \tag{17b}$$

$$\sum_k \nabla\cdot(\alpha_k \mathbf{u}_k) - t_0^{-1}\big(\sum_k \alpha_k - 1\big) = 0 \qquad \text{in } \Omega \times (0,T)\,. \tag{17c}$$

Denoting the number of phases by n_p, the above constitutes a system of $3n_p + 1$ equations in two dimensions and $4n_p + 1$ equations in three dimensions for the unknown velocity components, volume fractions, and the pressure. The system must be supplied with adequate initial and boundary conditions. Initial conditions will be written in the form $\mathbf{u}_k(\mathbf{x}, 0) = \mathbf{u}_k^0$, $\alpha_k(\mathbf{x}, 0) = \alpha_k^0$ in Ω. Boundary conditions are prescribed for each phase as

$$\mathbf{u}_k = \mathbf{u}_{\mathrm{d},k} \quad \text{on } \partial\Omega_{\mathrm{d},\mathbf{u}_k}\,, \tag{18a}$$

$$\mathbf{n}\cdot\boldsymbol{\sigma}_k = \mathbf{t}_k \quad \text{on } \partial\Omega_{\mathrm{n},\mathbf{u}_k}\,, \tag{18b}$$

$$\alpha_k = \alpha_{\mathrm{d},k} \quad \text{on } \partial\Omega_{\mathrm{d},\alpha_k}\,, \tag{18c}$$

where $\mathbf{n}$ is the outward unit normal vector, $\mathbf{t}_k$ are tractions applied to phase k, and

$$\boldsymbol{\sigma}_k = -\alpha_k p\, \mathbb{1} + 2\mu_k \alpha_k \boldsymbol{\varepsilon}(\mathbf{u}_k)\,. \tag{19}$$

In the above, $\partial\Omega_{\mathrm{d},\mathbf{u}_k}$ and $\partial\Omega_{\mathrm{n},\mathbf{u}_k}$ with $\partial\Omega_{\mathrm{d},\mathbf{u}_k} \cup \partial\Omega_{\mathrm{n},\mathbf{u}_k} = \partial\Omega$, $\partial\Omega_{\mathrm{d},\mathbf{u}_k} \cap \partial\Omega_{\mathrm{n},\mathbf{u}_k} = \emptyset$ denote the parts of the boundary where Dirichlet and Neumann boundary conditions are applied for a given phase, and $\partial\Omega_{\mathrm{d},\alpha_k}$ is the Dirichlet boundary for the volume fraction α_k, which must coincide with the inflow boundary of the corresponding phase. We may also remark that the equations for the different phases are coupled through three distinct mechanisms, namely, the drag forces, the constraint (17c), as well as the assumption of a common pressure.

Finally, for later use and to facilitate the comparison with previous work on related problems, we point out that the above set of partial differential equations is of the generic form

$$\boldsymbol{\mathcal{M}}(\mathbf{U})\,\partial_t \mathbf{U} + \boldsymbol{\mathcal{L}}(\widehat{\mathbf{U}}, \mathbf{U}) = \mathbf{F}\,, \tag{20}$$

where $\mathbf{U} \coloneqq [\mathbf{u}_1, ..., \mathbf{u}_{n_p}, \alpha_1, ..., \alpha_{n_p}, p]$, and the definitions of $\boldsymbol{\mathcal{M}}$, $\boldsymbol{\mathcal{L}}$, $\mathbf{F}$ follow directly from (17).

3. Variational form

For conciseness, we denote by $\mathbf{u} \coloneqq [\mathbf{u}_1, \ldots, \mathbf{u}_{n_p}]$ and $\boldsymbol{\alpha} \coloneqq [\alpha_1, \ldots, \alpha_{n_p}]$ the velocities and volume fractions of all phases. In addition, $\mathbf{U} \coloneqq [\mathbf{u}, \boldsymbol{\alpha}, p] \in \mathcal{W}$ denotes all unknown functions, taken from the space $\mathcal{W} \coloneqq \mathcal{V}_1 \times \ldots \times \mathcal{V}_{n_p} \times \mathcal{A}_1 \times \ldots \times \mathcal{A}_{n_p} \times \mathcal{Q}$. Here, in turn, $\mathcal{V}_k$, $\mathcal{A}_k$, and $\mathcal{Q}$ are appropriate spaces of velocities, volume fraction, and pressure with Dirichlet boundary conditions incorporated in the usual manner. To obtain a weak form of the governing equations, let us define test functions β_k, $\mathbf{v}_k$, q associated with the unknown functions α_k, $\mathbf{u}_k$, p and taken from the space denoted as $\mathcal{W}_0$. Here, $\mathcal{W}_0$ contains functions that vanish on the corresponding Dirichlet boundary, which, for the velocities and volume fractions, can be different for each phase. Multiplying Eqs. (17) by test functions, integrating over the domain, and integrating by parts the terms involving second-order spatial derivatives, we state the weak form of the multiphase problem as follows: find $\mathbf{U} \in \mathcal{W}$ such that the initial conditions are satisfied and

$$(\beta_k, \partial_t \alpha_k) + (\beta_k, \nabla\cdot(\alpha_k \mathbf{u}_k)) = 0\,, \tag{21a}$$

$$\begin{aligned}(\mathbf{v}_k, \rho_k \alpha_k \partial_t \mathbf{u}_k) + (\mathbf{v}_k, \rho_k \alpha_k \mathbf{u}_k\cdot\nabla\mathbf{u}_k) + (2\boldsymbol{\varepsilon}(\mathbf{v}_k), \alpha_k \mu_k \boldsymbol{\varepsilon}(\mathbf{u}_k)) \\ - (\nabla\cdot(\alpha_k \mathbf{v}_k), p) - \sum_\ell (\mathbf{v}_k, g_{k\ell}(\mathbf{u}_\ell - \mathbf{u}_k)) = (\mathbf{v}_k, \alpha_k \mathbf{f}_k) + (\mathbf{v}_k, \boldsymbol{\sigma}_k\cdot\mathbf{n})_{\partial\Omega_{\mathrm{n},\mathbf{u}_k}}\,,\end{aligned} \tag{21b}$$

$$\sum_k (q, \nabla\cdot(\alpha_k \mathbf{u}_k)) - \sum_k t_0^{-1}(q, \alpha_k) + t_0^{-1}(q, 1) = 0 \tag{21c}$$

for all $\mathbf{V} \coloneqq [\mathbf{v}, \boldsymbol{\beta}, q] \in \mathcal{W}_0$. Here, we use the symbol $(\bullet, \bullet)$ to indicate integration over the computational domain, irrespective of whether or not this expression represents the L^2-inner product in Ω, as long as the integrals are bounded. For each $\widehat{\mathbf{U}} \in \mathcal{W}$, the variational statement can now be written compactly in terms of a bilinear form $B(\widehat{\mathbf{U}}; \bullet, \bullet)$ defined on $\mathcal{W} \times \mathcal{W}$ as

$$G(\widehat{\boldsymbol{\alpha}}; \mathbf{U}, \mathbf{V}) + B(\widehat{\mathbf{U}}; \mathbf{U}, \mathbf{V}) = L(\mathbf{V}) \tag{22}$$

with

$$\begin{aligned}B(\widehat{\mathbf{U}}; \mathbf{U}, \mathbf{V}) \coloneqq \sum_k \big[(\mathbf{v}_k, \rho_k \widehat{\alpha}_k \widehat{\mathbf{u}}_k\cdot\nabla\mathbf{u}_k) + (2\boldsymbol{\varepsilon}(\mathbf{v}_k), \widehat{\alpha}_k \mu_k \boldsymbol{\varepsilon}(\mathbf{u}_k)) + p_0(\beta_k, \nabla\cdot(\widehat{\alpha}_k \mathbf{u}_k)) - (\nabla\cdot(\widehat{\alpha}_k \mathbf{v}_k), p) \\ + (q, \nabla\cdot(\widehat{\alpha}_k \mathbf{u}_k)) - t_0^{-1}(q, \alpha_k) - (\mathbf{v}_k, \alpha_k \mathbf{f}_k)\big] - \sum_k \sum_\ell (\mathbf{v}_k, g_{k\ell}(\mathbf{u}_\ell - \mathbf{u}_k))\,,\end{aligned} \tag{23a}$$

$$G(\widehat{\boldsymbol{\alpha}}; \mathbf{U}, \mathbf{V}) \coloneqq \sum_k \big[(\mathbf{v}_k, \rho_k \widehat{\alpha}_k \partial_t \mathbf{u}_k) + p_0(\beta_k, \partial_t \alpha_k)\big]\,, \tag{23b}$$

$$L(\mathbf{V}) \coloneqq -t_0^{-1}(q, 1) + \sum_k (\mathbf{v}_k, \mathbf{t}_k)_{\partial\Omega_{\mathrm{n},\mathbf{u}_k}}\,. \tag{23c}$$

Here, we introduced a scaling factor p_0 with the units of pressure, solely to render the different terms dimensionally consistent.

4. Finite-element formulation

4.1. Galerkin discretization

We consider a polygonal/polyhedral (in 2D/3D) finite element partition of the computational domain Ω and assume the partition to be quasi-uniform with an element diameter denoted as h. The

finite element spaces for velocities, volume fractions, and pressure are constructed in the usual manner and denoted as $\mathcal{V}_k^h \subset \mathcal{V}_k$, $\mathcal{A}_k^h \subset \mathcal{A}_k$, $\mathcal{Q}^h \subset \mathcal{Q}$, respectively. Functions in these spaces are indicated by the superscript h accordingly. In particular, we will focus our interest on polynomial interpolation of equal order. Employing the same finite element spaces for the trial and test functions, we obtain the Galerkin discretization as

$$G(\boldsymbol{\alpha}^h;\mathbf{U}^h,\mathbf{V}^h)+B(\widehat{\mathbf{U}}^h;\mathbf{U}^h,\mathbf{V}^h)=L(\mathbf{V}^h)\,. \tag{24}$$

Equation (24) is prone to instabilities that are well-understood in the case of a single incompressible fluid, see, e.g. [15]. Apart from oscillations arising in convection-dominated problems, instabilities occur if the inf-sup-condition, which poses restrictions on the compatibility of the interpolation spaces, is violated. To obtain stable solutions to the multiphase problem, we will require additional stabilization terms to achieve control not only of the pressure p^h but also of the volume fractions α_k^h.

4.2. Stabilized weak form

Following the general concept of the variational multiscale (VMS) method, we split the unknown functions into the component that belongs to the finite element space and a remainder, referred to as *subscale*, which cannot be resolved by the finite element discretization. The contributions to the unknowns and their respective spaces are denoted by $\bullet^h$ and $\widetilde{\bullet}$, i.e.,[5]

$$\mathbf{u}_k=\mathbf{u}_k^h+\widetilde{\mathbf{u}}_k,\quad \alpha_k=\alpha_k^h+\widetilde{\alpha}_k,\quad p=p^h+\widetilde{p},\quad \mathbf{U}=\mathbf{U}^h+\widetilde{\mathbf{U}},\quad \mathcal{W}=\mathcal{W}^h\oplus\widetilde{\mathcal{W}},\quad \mathcal{W}_0=\mathcal{W}_0^h\oplus\widetilde{\mathcal{W}}_0\,. \tag{25}$$

Accordingly, the variational form (22) can be re-written as

$$G(\boldsymbol{\alpha};\mathbf{U},\mathbf{V}^h+\widetilde{\mathbf{V}})+B(\mathbf{U};\mathbf{U}^h+\widetilde{\mathbf{U}},\mathbf{V}^h+\widetilde{\mathbf{V}})=L(\mathbf{V}^h+\widetilde{\mathbf{V}}) \tag{26}$$

or, by separating equations with respect to the test functions

$$G(\boldsymbol{\alpha};\mathbf{U},\mathbf{V}^h)+B(\mathbf{U};\mathbf{U}^h,\mathbf{V}^h)+B(\mathbf{U};\widetilde{\mathbf{U}},\mathbf{V}^h)=L(\mathbf{V}^h) \tag{27a}$$

$$G(\boldsymbol{\alpha};\mathbf{U},\widetilde{\mathbf{V}})+B(\mathbf{U};\mathbf{U}^h,\widetilde{\mathbf{V}})+B(\mathbf{U};\widetilde{\mathbf{U}},\widetilde{\mathbf{V}})=L(\widetilde{\mathbf{V}})\,. \tag{27b}$$

As usual (see, e.g., [25]), the fourth term in Eq. (27a) is integrated by parts while neglecting the contribution of the subscales on the element boundaries, leading to

$$G(\boldsymbol{\alpha};\mathbf{U},\mathbf{V}^h)+B(\mathbf{U};\mathbf{U}^h,\mathbf{V}^h)+\sum_K\left(\widetilde{\mathbf{U}},\boldsymbol{\mathcal{L}}^*(\mathbf{U},\mathbf{V}^h)\right)_K\approx L(\mathbf{V}^h)\,. \tag{28}$$

Here, $\boldsymbol{\mathcal{L}}^*$ denotes the adjoint of the differential operator $\boldsymbol{\mathcal{L}}$, and $\sum_K(\bullet,\bullet)_K$ indicates separate integration over each element and consecutive summation over the contributions of all elements in the mesh. The distinct variants of variational multiscale methods differ in how the subscale equation (27b) is approximated. In residual-based (algebraic or orthogonal) methods, the subscales are described as

$$\boldsymbol{\mathcal{M}}(\mathbf{U})\,\partial_t\widetilde{\mathbf{U}}+\boldsymbol{\tau}^{-1}\widetilde{\mathbf{U}}=\widetilde{P}\left[\mathbf{F}-\boldsymbol{\mathcal{M}}(\mathbf{U})\,\partial_t\mathbf{U}^h-\boldsymbol{\mathcal{L}}(\mathbf{U}^h,\mathbf{U}^h)\right], \tag{29}$$

[5] For consistency with previous publications, we use the symbol $\widetilde{\bullet}$ to denote subscales from here on. This is not to be confused with the intrinsic average mentioned in Section 2.

where τ is a matrix of stabilization parameters to be defined later; $\widetilde{P}$ denotes the projection onto the subscale space [19], here applied to the residual of the finite-element approximation. In principle, this approximation of the subscales can now be inserted into Eq. (28) to obtain a formulation solvable using standard finite-element routines. However, once again, different versions of this approach exist. Here, we opt to employ the following key concepts that have previously been described and analyzed for other problems:

- *Orthogonal subgrid scales:* We will use $\widetilde{P} = P^{\perp}$, i.e., the projection orthogonal to the finite-element space [15, 20, 26].
- *Term-by-term stabilization:* In conjunction with orthogonal subgrid scales, we have the option to neglect terms in Eqs. (29), (28) that are not essential for achieving stability. Specifically, we will only consider products of terms in $\boldsymbol{\mathcal{L}}$ by the corresponding term in $\boldsymbol{\mathcal{L}}^*$, namely those involving contributions of the form $\mathbf{u}_k^h \cdot \nabla \mathbf{v}_k^h \cdot P^{\perp}(\alpha_k^h \mathbf{u}_k^h \cdot \nabla \mathbf{u}_k^h)$, $\alpha^h \nabla q^h \cdot P^{\perp}(\alpha_k^h \nabla p^h)$, and $\mathbf{u}_k^h \cdot \nabla \beta_k^h \cdot P^{\perp}(\mathbf{u}_k^h \nabla \alpha_k^h)$. The resulting formulation has been successfully employed for single-phase flows and other problems and is known to be stable and convergent at an optimal rate [19, 27].
- *Dynamic subscales:* In many applications of variational multiscale methods, the first term in Eq. (29) is neglected, leading to an approximation often referred to as quasi-static subgrid scales. In contrast, we will include this term, which requires temporal discretization of Eq. (29), see [21, 28].
- *Linear subscales*: The stabilization terms (to be derived in the ensuing) are of the form $B_S(\mathbf{U}; \bullet, \bullet)$. A common approximation consists in taking $B_S(\mathbf{U}; \bullet, \bullet) \approx B_S(\mathbf{U}^h; \bullet, \bullet)$, which corresponds to using the finite element component in the computation of the nonlinear terms while neglecting the contribution of the subscales, see, e.g., [22].

Applying the term-by-term stabilization to the multiphase problem, we propose the following approximation of the velocity subscales, obtained by considering only the essential terms in the momentum equation for phase k:

$$\rho_k \alpha_k^h \, \partial_t \widetilde{\mathbf{u}}_k + \rho_k \alpha_k^h (\tau_k^u)^{-1} \widetilde{\mathbf{u}}_k = -P^{\perp}\big(\rho_k \alpha_k^h \mathbf{u}_k^h \cdot \nabla \mathbf{u}_k^h\big) - P^{\perp}\big(\alpha_k^h \nabla p^h\big) \tag{30}$$

with a stabilization parameter τ_k^u yet to be defined.[6] Similarly, we propose subscales for the volume fractions based on the continuity equation for phase k as

$$\partial_t \widetilde{\alpha}_k + (\tau_k^{\alpha})^{-1} \, \widetilde{\alpha}_k = -P^{\perp}\big(\mathbf{u}_k^h \cdot \nabla \alpha_k^h\big) \, . \tag{31}$$

A common assumption made, e.g., in the single-phase case [19] states that the velocity subscales can be split into two independent components, such that

$$\rho_k \alpha_k^h \, \partial_t \widetilde{\mathbf{u}}_{1k} + \rho_k \alpha_k^h (\tau_k^u)^{-1} \, \widetilde{\mathbf{u}}_{1k} = -P^{\perp}\big(\rho_k \alpha_k^h \mathbf{u}_k^h \cdot \nabla \mathbf{u}_k^h\big) \, , \tag{32a}$$

$$\rho_k \alpha_k^h \, \partial_t \widetilde{\mathbf{u}}_{2k} + \rho_k \alpha_k^h (\tau_k^u)^{-1} \, \widetilde{\mathbf{u}}_{2k} = -P^{\perp}\big(\alpha_k^h \nabla p^h\big) \, . \tag{32b}$$

[6] The factor of $\rho_k \alpha_k^h$ in the second term is introduced for convenience such that all stabilization parameters have the unit of time.

The time-derivatives of the subscales are approximated by a first-order backward difference formula (*BDF1*), e.g.,

$$\rho_k \alpha_k^h \frac{\widetilde{\mathbf{u}}_{1k}^{j+1} - \widetilde{\mathbf{u}}_{1k}^{j}}{\delta t} + \rho_k \alpha_k^h (\tau_k^u)^{-1} \widetilde{\mathbf{u}}_{1k}^{j+1} = -P^{\perp}\big(\rho_k \alpha_k^{h,j+1} \mathbf{u}_k^{h,j+1} \cdot \nabla \mathbf{u}_k^{h,j+1}\big) \tag{33}$$

and analogously for $\widetilde{\mathbf{u}}_{2k}$ and $\widetilde{\alpha}_k$, where δt is the time step (assumed constant for simplicity), and j indicates the time-step counter. As explained in [28], due to the fact that the stabilization parameters are of the order of δt, this approximation suffices to obtain optimal convergence in second-order schemes. Hence, at a given time step, the subscales are given as

$$\widetilde{\mathbf{u}}_{1k}^{j+1} = \overline{\tau}_k^u \left(\delta t^{-1} \widetilde{\mathbf{u}}_{1k}^{j} - (\alpha_k^{h,j+1})^{-1} P^{\perp}\big(\alpha_k^{h,j+1} \mathbf{u}_k^{h,j+1} \cdot \nabla \mathbf{u}_k^{h,j+1}\big) \right), \tag{34a}$$

$$\widetilde{\mathbf{u}}_{2k}^{j+1} = \overline{\tau}_k^u \left(\delta t^{-1} \widetilde{\mathbf{u}}_{2k}^{j} - (\rho_k \alpha_k^{h,j+1})^{-1} P^{\perp}\big(\alpha_k^{h,j+1} \nabla p^{h,j+1}\big) \right), \tag{34b}$$

$$\widetilde{\alpha}_k^{j+1} = \overline{\tau}_k^{\alpha} \left(\delta t^{-1} \widetilde{\alpha}_k^{j} - P^{\perp}\left(\mathbf{u}_k^{h,j+1} \nabla \alpha^{h,j+1}\right) \right) \tag{34c}$$

with the abbreviations

$$\overline{\tau}_k^u = \left(\frac{1}{\delta t} + \frac{1}{\tau_k^u} \right)^{-1}, \qquad \overline{\tau}_k^{\alpha} = \left(\frac{1}{\delta t} + \frac{1}{\tau_k^{\alpha}} \right)^{-1}.$$

These expressions for the subscales at a given time step $j+1$ can now be substituted into Eq. (28). Considering only the terms relevant to achieve stability, as mentioned before, leads to the following forms to stabilize the momentum equation

$$\begin{aligned} B_S^M(\widehat{\mathbf{U}}^h; \mathbf{U}^h, \mathbf{V}^h)\Big|^{j+1} = &- \sum_K \left(\rho_k \widehat{\mathbf{u}}_k^{h,j+1} \cdot \nabla \mathbf{v}_k^h, \overline{\tau}_k^u \left(\delta t^{-1} \widehat{\alpha}_k^{h,j+1} \widetilde{\mathbf{u}}_{1k}^{j} - P^{\perp}\big(\widehat{\alpha}_k^{h,j+1} \widehat{\mathbf{u}}_k^{h,j+1} \cdot \nabla \mathbf{u}_k^{h,j+1}\big) \right) \right)_K \\ &- \sum_K \left(\nabla q^h, \overline{\tau}_k^u \left(\delta t^{-1} \widehat{\alpha}_k^{h,j+1} \widetilde{\mathbf{u}}_{2k}^{j} - \rho_k^{-1} P^{\perp}\big(\widehat{\alpha}_k^{h,j+1} \nabla p^{h,j+1}\big) \right) \right)_K \end{aligned}$$

and the continuity equation

$$B_S^C(\widehat{\mathbf{U}}^h; \mathbf{U}^h, \mathbf{V}^h)\Big|^{j+1} = -p_0 \sum_K \left(\widehat{\mathbf{u}}_k^{h,j+1} \cdot \nabla \beta_k^h, \overline{\tau}_k^{\alpha} \left(\delta t^{-1} \widetilde{\alpha}_k^{j} - P^{\perp}\big(\widehat{\mathbf{u}}_k^{h,j+1} \cdot \nabla \alpha_k^{h,j+1}\big) \right) \right)_K .$$

Furthermore, we propose adding

$$B_S^S(\mathbf{U}^h, \mathbf{V}^h)\Big|^{j+1} = \sum_K \left(\nabla \beta_k^h, |\mathbf{u}_{k,c}|\, h\, P^{\perp}\big(\nabla \alpha_k^h\big) \right)_K$$

with a characteristic velocity $\mathbf{u}_{k,c}$, which we typically choose as the maximum velocity at the inflow boundary for each phase. Apart from the projection $P^{\perp}(\bullet)$, this contribution represents an artificial diffusion of magnitude $|\mathbf{u}_{k,c}|\, h$. This term is added with the aim to improve stability in regions where one of the velocities is very small and, consequently, the stabilization of the corresponding volume fraction by B_S^C nearly vanishes.

Combining all terms outlined above, the proposed stabilized finite element method consists in finding $\mathbf{U}^h \in \mathcal{W}^h$ such that

$$G(\boldsymbol{\alpha}^h; \mathbf{U}^h, \mathbf{V}^h) + B(\widehat{\mathbf{U}}^h; \mathbf{U}^h, \mathbf{V}^h) + B_S^M(\widehat{\mathbf{U}}^h; \mathbf{U}^h, \mathbf{V}^h) + B_S^C(\widehat{\mathbf{U}}^h; \mathbf{U}^h, \mathbf{V}^h) + B_S^S(\mathbf{U}^h, \mathbf{V}^h) = L(\mathbf{V}^h) \tag{35}$$

for all $\mathbf{V}^h \in \mathcal{W}_0^h$.

4.3. Stabilization parameters

The parameters τ_k^α and τ_k^u need to be chosen such that the residuals of the momentum and continuity equations vanish as the element size h approaches zero and the optimal convergence rate is attained in the asymptotic regime. A common approach to derive expressions for these parameters is based on a Fourier analysis of the residuals, see, e.g., [20, 29]. In these previous works, it has been discussed in detail that the stabilization parameter for a PDE involving 'diffusive,' 'convective,' and 'reactive' terms, say $-\mu\,\Delta\mathbf{u} + \mathbf{a}\cdot\nabla\mathbf{u} + r\,\mathbf{u}$, can be chosen as $(c_1\,\mu\,p_e^4/h^2 + c_2\,|\mathbf{a}|\,p_e/h + r)^{-1}$. Here, p_e denotes the polynomial degree of the interpolation, and c_1, c_2 are algorithmic constants [30]. Applying these results to the continuity and momentum equations of the multiphase problem directly yields

$$\tau_k^\alpha = \left(\frac{c_1 \gamma_k\, p_e^4}{h^2} + \frac{c_2 |\mathbf{u}_k| p_e}{h} + |\nabla\cdot\mathbf{u}_k| \right)^{-1}, \tag{36a}$$

$$\tau_k^u = \left(\frac{c_1\,\mu_k\, p_e^4}{\rho_k h^2} + \frac{c_2 \mu_k |\nabla\alpha_k| p_e}{\rho_k \alpha_k h} + \frac{c_2 |\mathbf{u}_k| p_e}{h} + \sum_\ell \frac{|g_{k\ell}|}{\rho_k \alpha_k} \right)^{-1}. \tag{36b}$$

In the definition of τ_k^α, we introduced scalar parameters γ_k such that τ_k^α is well-defined in regions where one of the velocities approaches zero. For all numerical studies presented in this paper, we chose $\gamma_k = 10^{-8}|\mathbf{u}_{k,c}|h$, where the factor of $|\mathbf{u}_{k,c}|h$ ensures dimensional consistency. The other algorithmic constants were chosen as $c_1 = 4$, $c_2 = 2$. Note that the first two terms in Eq. (36b) both result from the term involving $\nabla\cdot(\alpha\boldsymbol{\varepsilon}) = \nabla\alpha\cdot\boldsymbol{\varepsilon} + \alpha\nabla\cdot\boldsymbol{\varepsilon}$. Similarly, the contribution of the form $\alpha\nabla\cdot\mathbf{u}$ in the continuity equation is treated as a reactive term in the stabilization of the volume fractions. As mentioned in footnote 6, a factor of $\rho_k\alpha_k$ is included in the definition of τ_k^u for convenience.

5. Implementation aspects

5.1. Linearization

A straightforward linearization of system (21) (or its discrete counterpart) is obtained by means of a *fixed-point iteration* (also referred to here as Picard's method), i.e., approximating the nonlinear terms by substituting preliminary solutions computed in a previous iteration. As several terms are nonlinear in both the velocity and volume fraction, this procedure is not unique. Nevertheless, a rather obvious choice may be the following:

$$\begin{aligned} G^{\mathrm{P}}(\widehat{\boldsymbol{\alpha}};\mathbf{U},\mathbf{V})\big|^{i+1} + B^{\mathrm{P}}(\widehat{\mathbf{U}};\mathbf{U},\mathbf{V})\Big|^{i+1} \coloneqq \sum_k \big[&(\mathbf{v}_k, \rho_k\widehat{\alpha}_k^i \partial_t \mathbf{u}_k^{i+1}) + (\beta_k, p_0\,\partial_t\alpha_k^{i+1}) + (\mathbf{v}_k, \rho_k\widehat{\alpha}_k^i\widehat{\mathbf{u}}_k^i\cdot\nabla\mathbf{u}_k^{i+1}) \\ &+ (2\boldsymbol{\varepsilon}(\mathbf{v}_k), \widehat{\alpha}_k^i\mu_k\boldsymbol{\varepsilon}(\mathbf{u}_k^{i+1})) + (\beta_k, p_0\,\widehat{\alpha}_k^i\nabla\cdot\mathbf{u}_k^{i+1}) + (\beta_k, p_0\,\widehat{\mathbf{u}}_k^i\cdot\nabla\alpha_k^{i+1}) - (\nabla\cdot(\widehat{\alpha}_k^i\mathbf{v}_k), p^{i+1}) + (q, \widehat{\alpha}_k^i\nabla\cdot\mathbf{u}_k^{i+1}) \\ &+ (q, \widehat{\mathbf{u}}_k^{i+1}\cdot\nabla\alpha_k^i) - t_0^{-1}(q, \alpha_k^{i+1}) - (\mathbf{v}_k, \alpha_k^{i+1}\mathbf{f}_k)\big] - \sum_k\sum_\ell (\mathbf{v}_k, g_{k\ell}^i(\mathbf{u}_\ell^{i+1} - \mathbf{u}_k^{i+1}))\,. \end{aligned} \tag{37a}$$

Here, superscripts $i+1$ and i refer to variables in the current and previous iteration, respectively (omitting temporal discretization for conciseness), and the superscript P indicates Picard iteration. As the phase-interaction term depends on the model, it may be a function of the volume

fractions and velocities, here evaluated based on the values at iteration i. Hence, we can think of this term as $g_{k\ell}^i := g_{k\ell}(\mathbf{u}^i, \alpha^i)$. We may also note that we treat the term involving body loads implicitly, i.e., solving for the unknowns α_k^{i+1} in $(\mathbf{v}_k, \alpha_k^{i+1}\mathbf{f}_k)$.

We will also make use of the Newton-Raphson approach to obtain an alternative linearization that is expected to converge faster, provided that sufficiently accurate starting values are given. To distinguish between the different nonlinearities involved in the weak form, let us rewrite Eq. (23) as

$$G(\widehat{\boldsymbol{\alpha}};\mathbf{U},\mathbf{V}) + B(\widehat{\mathbf{U}};\mathbf{U},\mathbf{V}) := B_1(\mathbf{U},\mathbf{V}) + B_2(\widehat{\boldsymbol{\alpha}};\mathbf{U},\mathbf{V}) + B_3(\widehat{\boldsymbol{\alpha}},\widehat{\mathbf{u}};\mathbf{U},\mathbf{V}) + B_g(\widehat{\boldsymbol{\alpha}},\widehat{\mathbf{u}};\mathbf{U},\mathbf{V}) \tag{38}$$

with

$$\begin{aligned}
B_1(\mathbf{U},\mathbf{V}) &= \sum_k \left[(\beta_k, \partial_t \alpha_k) - t_0^{-1}(q,\alpha_k) - (\mathbf{v}_k, \alpha_k \mathbf{f}_k)\right] \\
B_2(\widehat{\boldsymbol{\alpha}};\mathbf{U},\mathbf{V}) &= \sum_k \left[(\mathbf{v}_k, \rho_k \widehat{\alpha}_k \partial_t \mathbf{u}_k) + (2\boldsymbol{\varepsilon}(\mathbf{v}_k), \widehat{\alpha}_k \mu_k \boldsymbol{\varepsilon}(\mathbf{u}_k)) + (\beta_k, \nabla\cdot(\widehat{\alpha}_k \mathbf{u}_k)) - (\nabla\cdot(\widehat{\alpha}_k \mathbf{v}_k), p) + (q, \nabla\cdot(\widehat{\alpha}_k \mathbf{u}_k))\right] \\
B_3(\widehat{\boldsymbol{\alpha}},\widehat{\mathbf{u}};\mathbf{U},\mathbf{V}) &= \sum_k (\mathbf{v}_k, \rho_k \widehat{\alpha}_k \widehat{\mathbf{u}}_k \cdot \nabla \mathbf{u}_k) \\
B_g(\widehat{\boldsymbol{\alpha}},\widehat{\mathbf{u}};\mathbf{U},\mathbf{V}) &= -\sum_k \sum_\ell (\mathbf{v}_k, \widehat{g}_{k\ell}(\mathbf{u}_\ell - \mathbf{u}_k)),
\end{aligned}$$

with $\widehat{g}_{k\ell}$ denoting the momentum exchange, which, depending on the employed model, can be a function of $\widehat{\alpha}_k$, $\widehat{\alpha}_\ell$, $\widehat{\mathbf{u}}_k$, and $\widehat{\mathbf{u}}_\ell$. The Newton-Raphson ('NR') linearization of the terms involving simple polynomial nonlinearities is then obtained as

$$\begin{aligned}
B_1^{\mathrm{NR}} &:= B_1(\mathbf{U}^{i+1},\mathbf{V}), \\
B_2^{\mathrm{NR}} &:= B_2(\widehat{\boldsymbol{\alpha}}^i;\mathbf{U}^{i+1},\mathbf{V}) + B_2(\widehat{\boldsymbol{\alpha}}^{i+1};\mathbf{U}^i,\mathbf{V}) - B_2(\widehat{\boldsymbol{\alpha}}^i;\mathbf{U}^i,\mathbf{V}), \\
B_3^{\mathrm{NR}} &:= B_3(\widehat{\boldsymbol{\alpha}}^i,\widehat{\mathbf{u}}^i;\mathbf{U}^{i+1},\mathbf{V}) + B_3(\widehat{\boldsymbol{\alpha}}^{i+1},\widehat{\mathbf{u}}^i;\mathbf{U}^i,\mathbf{V}) + B_3(\widehat{\boldsymbol{\alpha}}^i,\widehat{\mathbf{u}}^{i+1};\mathbf{U}^i,\mathbf{V}) - 2B_3(\widehat{\boldsymbol{\alpha}}^i,\widehat{\mathbf{u}}^i;\mathbf{U}^i,\mathbf{V}).
\end{aligned}$$

The linearization of the interaction term B_g depends, once again, on the selected model. Assuming a drag term of the form

$$g_{k\ell} = C_g \alpha_k \alpha_\ell |\mathbf{u}_\ell - \mathbf{u}_k| \tag{41}$$

with some constant C_g, we obtain

$$B_g^{\mathrm{NR}} = -\sum_k \sum_\ell \left[2(\mathbf{v}_k, C_g \alpha_k^i \alpha_\ell^i w^i \mathbf{w}^{i+1}) + (\mathbf{v}_k, C_g \alpha_k^{i+1} \alpha_\ell^i w^i \mathbf{w}^i) + (\mathbf{v}_k, C_g \alpha_k^i \alpha_\ell^{i+1} w^i \mathbf{w}^i) - 3(\mathbf{v}_k, C_g \alpha_k^i \alpha_\ell^i w^i \mathbf{w}^i)\right] \tag{42}$$

with the abbreviations $\mathbf{w} = \mathbf{u}_\ell - \mathbf{u}_k$, $w = |\mathbf{w}|$. The linearization of the stabilization terms follows the same procedure. An overview of all terms to be included in the linearized weak form is presented in Appendix B. However, it should be noted that the nonlinearities present in the stabilization parameters $\boldsymbol{\tau}$ as well as the projection of the subscales are always linearized by a simple Picard iteration, even when employing the Newton-Raphson scheme for the Galerkin terms.

5.2. *Time stepping*

To integrate the discretized weak form in time, we may use standard approaches well-known from the solution of the single-phase incompressible Navier-Stokes equations and many other problems. Note, however, that the mass matrix associated with the momentum equation is nonlinear, as it depends on the volume fractions. The numerical examples presented in this paper make use of

a fully implicit 2nd-order backward difference scheme (BDF2) for the Galerkin terms, while the subscales are always integrated by a BDF1 for efficiency.

5.3. *Drag coefficient*

Many different models exist for the moment exchange between two phases. We studied three typical scenarios: Initial convergence studies are performed using a constant drag coefficient, i.e., $g_{k\ell} = g_{\ell k} = \text{const}$. A particularly popular representation of the drag term is based on the Schiller-Naumann model of a dispersed phase (phase 1) in a carrier fluid (phase 2), see, e.g., [31]. The dispersed phase is assumed to be in the form of bubbles of solid particles with uniform diameter d_1, leading to the approximate expression

$$g_{12} = \frac{\alpha_1 \, \alpha_2 \, \rho_1 \, C_D \, \mathrm{Re}}{24 t_1} \tag{43}$$

with

$$C_D = \begin{cases} 24 \left(1 + 0.15 \, \mathrm{Re}^{0.687}\right) / \mathrm{Re} & \mathrm{Re} \leq 1000 \\ 0.44 & \mathrm{Re} > 1000 \end{cases}, \qquad \mathrm{Re} = \frac{\rho_2 \, |\mathbf{u}_2 - \mathbf{u}_1| \, d_1}{\mu_2} \qquad t_1 = \frac{\rho_1 \, d_1^2}{18 \mu_2} . \tag{44}$$

Furthermore, to reproduce results obtained by Hiltunen [7], we employ the following drag coefficient used there and in many other works as a simpler approximation of a model similar to that by Schiller and Naumann:

$$g_{12} = \frac{3 C_D}{4 d_1} \alpha_1 \, |\mathbf{u}_2 - \mathbf{u}_1| \, . \tag{45}$$

5.4. *Enforcing the algebraic constraint*

Physically, the values of the volume fractions must take values $0 \leq \alpha_k \leq 1$. This condition is not explicitly enforced by the variational form of the problem and can be violated point-wise due to spatial and temporal approximation. Furthermore, numerical issues can manifest if one of the volume fractions approaches zero in parts of the domain since the corresponding fluid velocity is undefined where the volume fraction vanishes. To overcome both problems, we introduce a threshold α_{th} such that $\alpha_{\text{th}} \leq \alpha_k \leq 1 - \alpha_{\text{th}}$, see, e.g., [32] for similar strategies. This condition is enforced at the Gauss points during each iteration of the nonlinear solver such that the coefficient matrices are integrated based on the corrected values. In addition, we apply this correction to the nodal values after each time step.

5.5. *Choosing the time scale t_0*

In Section 2.2, we introduced an algorithmic constant t_0 acting as a penalization in enforcing the algebraic constraint $\sum \alpha_k = 1$. As has been explained there, this constant represents a time scale during which deviations from this condition decay. Hence, we propose choosing this value based on a characteristic time t_c, which we define as

$$t_c = \min(t_k) \tag{46}$$

with

$$t_k = \left(\frac{\mu_k}{\rho_k L_c^2} + \frac{|\mathbf{u}_{k,c}|}{L_c} \right)^{-1} \tag{47}$$

with a characteristic length L_c and the characteristic velocities $\mathbf{u}_{c,k}$ introduced before. With these definitions, our numerical studies suggest that robust results can be obtained by choosing

$$t_0 = \frac{t_c}{1000}. \tag{48}$$

5.6. Implicit/explicit treatment of the projections

The stabilization terms require computing projections $P^{\perp}(\bullet)$ of the subscales onto a space orthogonal to the finite element space. These are computable as $P^{\perp}(\bullet) = I(\bullet) - P^h(\bullet)$, where $P^h(\bullet)$ denotes the projection onto the finite element space. Take, as an example, the subscales of the volume fractions, Eq. (34c), consider quasi-static subscales and omit the phase index to simplify notation. Hence, we need to compute

$$\widetilde{\alpha} = -\tau^{\alpha} P^{\perp}\big(\mathbf{u}^h \cdot \nabla \alpha^h\big) = -\tau^{\alpha}\left(\mathbf{u}^h \cdot \nabla \alpha^h - P^h\big(\mathbf{u}^h \cdot \nabla \alpha^h\big)\right) =: -\tau^{\alpha}\left(\mathbf{u}^h \cdot \nabla \alpha^h - \xi\right)$$

with ξ being the projection of this term on the finite element space. The nodal values of the projection are obtained by solving

$$(\eta^h, \mathbf{u}^h \cdot \nabla \alpha^h) = (\eta^h, \xi^h)$$

with adequate test functions η^h. Within a transient nonlinear solution, we have essentially three options to incorporate this term. Denoting by j and i the indices of the time step and nonlinear iteration, respectively, the first option consists in computing the projections based on the previous iteration within the same time step, i.e.,

$$(\eta^h, \mathbf{u}^{h,i,j+1} \cdot \nabla \alpha^{h,i,j+1}) \approx (\eta^h, \xi^{h,i+1,j+1}) \qquad \text{semi-implicit}$$

This approach, which we may refer to as semi-implicit, is usually *not* recommended, as it can require many iterations for the subscales to converge within each time step. Instead, at least in transient problems, it is sufficient and more effective to use a fully explicit approach, i.e., computing the projections based on the results of the previous time step:

$$(\eta^h, \mathbf{u}^{h,j} \cdot \nabla \alpha^{h,j}) \approx (\eta^h, \xi^{h,j+1}) \qquad \text{explicit}$$

In this approach, the iteration counter i is omitted, as the projections are computed based on the converged solutions of the previous time step. Note that, if the utilized finite elements allow 'mass lumping', i.e., the Gram matrix can be diagonalized, then solving the above system does not infer significant computational costs. Hence, the computation of the stabilization terms is negligible compared to the Galerkin terms. This approach introduces an error of order $\mathcal{O}(\delta t)$, but, as explained earlier, this suffices for schemes that are globally of second order.

The third option consists in computing the projections implicitly, i.e., treating them as unknowns in each iteration

$$(\eta^h, \mathbf{u}^{h,i+1,j+1} \cdot \nabla \alpha^{h,i+1,j+1}) \approx (\eta^h, \xi^{h,i+1,j+1}) \qquad \text{implicit}$$

In this variant, the projections must be obtained by solving a coupled system of equations for the

unknowns $\mathbf{U}$ and $\boldsymbol{\xi}$ of the form

$$\begin{bmatrix} \mathbf{K} & -\mathbf{P}_\tau \\ \mathbf{P} & -\mathbf{M} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \boldsymbol{\xi} \end{bmatrix} = \begin{bmatrix} \mathbf{F} \\ \mathbf{0} \end{bmatrix} \tag{49}$$

where $\mathbf{K}$ denotes the stiffness matrix (inlcuding the Galerkin terms and stability without the projection), and the second equation corresponds to the computation of the projections as outlined above. A detailed discussion of this approach can be found in [17] for the case of the Oseen problem. In our example of the subscales $\widetilde{\alpha}$, the matrix $\mathbf{P}_\tau$ represents the term stemming from $(\mathbf{u}^h \cdot \nabla \beta^h, -\tau^\alpha \xi^h)$. The additional degrees of freedom can be eliminated by static condensation, yielding

$$\left(\mathbf{K} - \mathbf{P}_\tau \mathbf{M}^{-1} \mathbf{P}\right) \mathbf{u} = \mathbf{F}\,. \tag{50}$$

Once again, if the Gram matrix $\mathbf{M}$ is diagonal, the cost of computing the additional term $\mathbf{P}_\tau \mathbf{M}^{-1} \mathbf{P}\mathbf{u}$ is small compared to the solution of the final system of equations. Compared to the explicit version, the computational costs for each iteration are significantly larger due to the increased size of the matrix stencil in Eq. (50). Nevertheless, the implicit variant is our method of choice for stationary problems where no initial guess from a previous time step exists.

6. Numerical examples

6.1. Convergence studies

6.1.1. Constant volume fractions

We begin by numerically assessing the convergence of the error with respect to the time step size, mesh size, and element order. For this purpose, we devise an example of a manufactured solution similar to one that has previously been used for verifying solutions to the single-phase problem [21, 33]. We consider both a variant with a stationary solution as well as a transient case in which the time-dependent solution is obtained by means of time stepping (BDF2). Specifically, we choose the velocities, volume fractions, and pressure of a two-phase flow problem as follows

$$\mathbf{u}_1 = 100\, f_t(t)\, \left(f(x) f'(y),\, -f'(x) f(y)\right) + \mathbf{u}_0,\;\; \mathbf{u}_2 = 2\,\mathbf{u}_1,\;\; p = 100 x^2,\;\; \alpha_1 = 0.3,\;\; \alpha_2 = 0.7$$

with

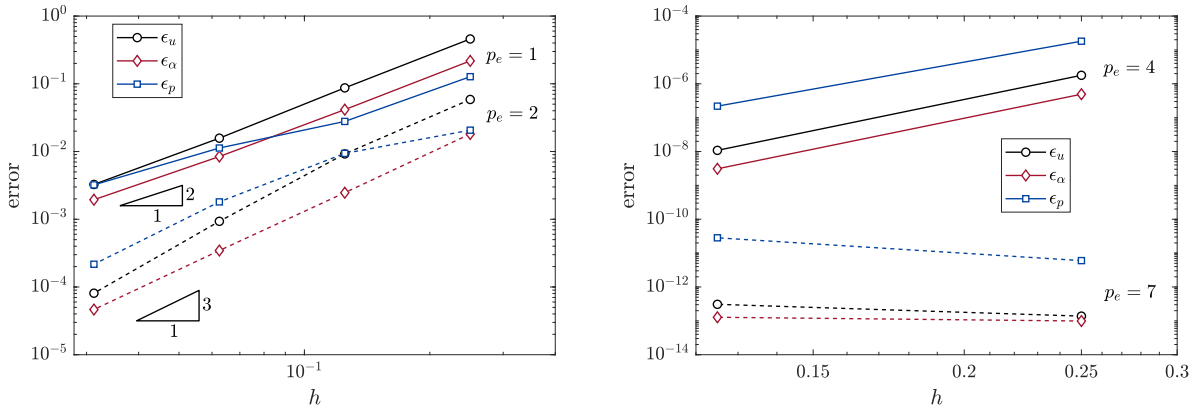
$$f(x) = x^2 (1-x)^2, \quad f_t(t) = \begin{cases} 1 \\ \cos(\pi t) \mathrm{e}^{-t} \end{cases}, \quad \mathbf{u}_0 = \begin{cases} (1,1) & \text{stationary case} \\ (0,0) & \text{transient case} \end{cases}$$

and compute the body loads accordingly such that the selected solution satisfies the strong form of the problem. Note that the velocity fields are divergence-free, and, since the volume fractions are constant, we also have $\nabla(\alpha_k \mathbf{u}_k) = 0$. The computational domain is the unit square, i.e.,

$$\Omega = \left\{ (x,y) \in \mathbb{R}^2 \;\middle|\; 0 \le x \le 1,\; 0 \le y \le 1 \right\}.$$

The material constants are $\mu_1 = 0.1$, $\mu_2 = 0.2$, $\rho_1 = \rho_2 = 1$, and the momentum exchange coefficient is chosen as a constant value of $g_{12} = g_{21} = 0.5$. The exact solution of the velocities is prescribed on the entire boundary, and the pressure is fixed at the point $(1,1)$.

We first address the stationary case, i.e., compute the time-independent solution with $f_t = 1$. The domain is discretized into a regular mesh of quadrilaterals, beginning with 4×4 elements and consistently refining by dividing each element into four. The volume fractions are prescribed at



(a) $p_e = 1, 2$ (b) $p_e = 4, 7$

Figure 1: Convergence of the error in velocities, volume fractions, and pressure using elements of degree $p_e = 1$, $p_e = 2$ (a) and $p_e = 4$, $p_e = 7$ (b).

the inflow boundaries (left and bottom). Figure 1a shows the L^2-norm of the error in velocities, volume fractions, and pressure using elements with a polynomial degree of $p_e = 1$ and $p_e = 2$. These errors are denoted as ϵ_u, ϵ_α, and ϵ_p, respectively. The theoretical asymptotic convergence rates are obtained approximately. Specifically, evaluating the convergence rate based on the first two points of each graph yields the numerical values of 2.3 (velocities), 2.1 (volume fractions), 1.8 (pressure) for the case $p_e = 1$ and 3.5, 2.9, 3.1 for $p_e = 2$. In addition, we present in Fig. 1b the errors computed when employing elements of orders 4 and 7. These polynomial degrees have been selected to demonstrate a peculiarity of the proposed approach: Note that the exact solution is given by polynomials up to a degree of four. Hence, a consistent method would be expected to yield exact results when using elements of the same order. As mentioned before, the proposed method is not consistent but converges at an optimal rate. Consequently, when using fourth-order elements, the error in all variables is very small, yet nonzero. On the other hand, exact solutions are expected if the approximation of the subscales vanishes, i.e., if the arguments of the projection operators in Eqs. (34) are in the finite-element space. In our current example, this happens when $(\alpha_k \mathbf{u}_k \cdot \nabla \mathbf{u}_k)$ is in the finite-element space. As this term contains polynomials up to degree seven, elements of order $p_e = 7$ are expected to result in near-zero errors, which is confirmed by the results in Fig. 1b.

To test the transient case, we utilized a mesh of 4×4 elements of order seven, such that the error due to spatial discretization can be neglected. We then define initial conditions according to the exact solution and compute results using a BDF2-scheme (and BDF1 for the subscales) with different values of the time step. The L^2-norm of the errors in the final time step $T = 1.25$ are depicted in Fig. 2. Optimal convergence is obtained with numerical values of the rates computed between the first two points of 1.97 (velocities), 2.17 (volume fractions), and 2.35 (pressure).

In addition, we studied the effect of treating the projections (semi)-implicitly or explicitly as explained in Section 5.6. To this end, we compare the total number of iterations required in the solution of the transient problem. Figure 3 exemplarily depicts the residual in velocities for a mesh of 8×8 bilinear elements when using the different approaches to computing the projections. Results for other meshes and element orders are qualitatively similar. It is apparent that a 'semi-implicit'

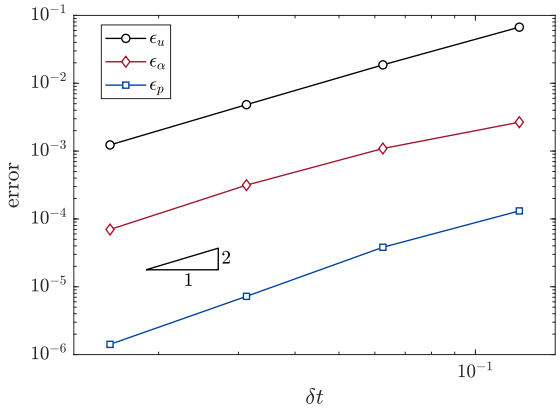
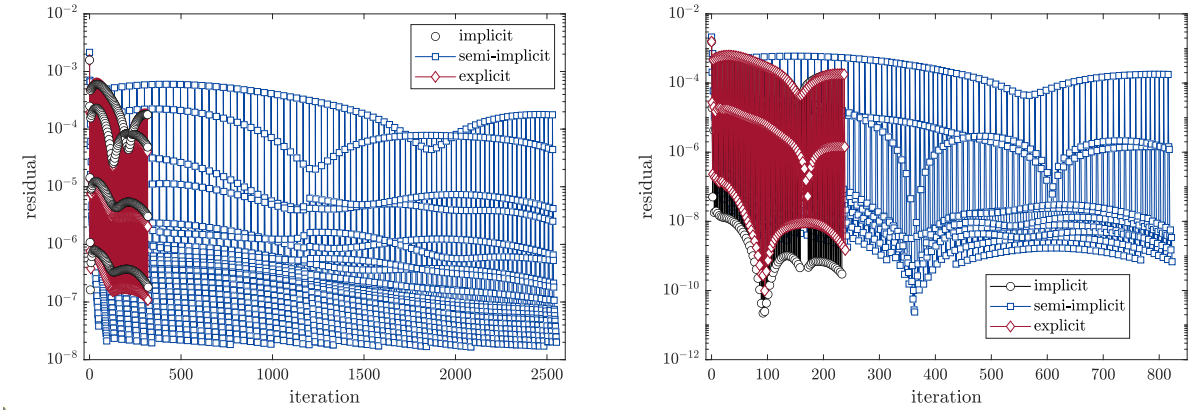


Figure 2: L^2-norm of the error in the final time step for varying time-step size δt.



(a) Picard

(b) Newton-Raphson

Figure 3: Residual vs. number of iterations when using an implicit, semi-implicit, or explicit approach to computing the projections of the subscales. Results are presented for Picard iteration (a) and Newton-Raphson's method.

approach (computing the projections based on the previous nonlinear iteration) requires significantly more iterations to reduce the residual below the chosen threshold (here 10^{-6}) compared to the other two approaches. The effect is more pronounced when employing Picard's iteration (Fig. 1a) compared to Newton-Raphson's method. On the other hand, the differences between the fully implicit and explicit computations in terms of the number of iterations are minimal. As the implicit version is computationally more expensive, the explicit approach is preferred in transient simulations. Obviously, in stationary problems, we can only choose between the implicit and semi-explicit version. In that case, it may be preferable to employ the implicit method due to the significantly better convergence. This will be addressed in the following example.

6.1.2. Linear volume fractions

To ensure convergence in the case of non-constant volume fractions, we consider another academic, yet slightly more complex, example by assuming both volume fractions to vary linearly in Ω. In order to derive a valid analytical solution, the volume fractions must add up to unity point-wise in the domain. Furthermore, the continuity equations for each phase should be satisfied, which is somewhat more involved compared to the trivial case of constant volume fractions discussed in the previous example. We choose the following stationary solution:

$$\mathbf{u}_1(x,y) = \left(x+1, -\frac{1.2x+0.8}{0.6x+0.2}y+4.5\right), \quad \mathbf{u}_2(x,y) = \left(x^2+1, \frac{1.8x^2-1.6x+0.6}{0.8-0.6x}y+1\right), \tag{51a}$$

$$\alpha_1(x) = 0.2+0.6x, \quad \alpha_2(x) = 0.8-0.6x, \quad p(x,y) = x+y+1\,. \tag{51b}$$

The velocity fields are plotted in Fig. 4a. The material parameters are

$$\rho_1 = 2, \quad \rho_2 = 1/3, \quad \mu_1 = 1, \quad \mu_2 = 1\,,$$

and the phase interaction function is chosen as

$$g_{12} = \alpha_1|\mathbf{u}_2 - \mathbf{u}_1|\,.$$

The body loads $\mathbf{f}_1(x,y)$ and $\mathbf{f}_2(x,y)$ that result in the desired solution are visualized in Fig. 4b (we refrain from displaying the lengthy analytical expressions describing the body loads). Initial guesses are chosen as

$$\mathbf{u}_1^0(x,y) = \mathbf{u}_2^0(x,y) = (1,1), \quad \alpha_1^0(x,y) = \alpha_2^0(x,y) = 0.5 \quad p^0(x,y) = 1 \qquad \text{in } \Omega.$$

Boundary conditions for velocities are prescribed at the whole boundary and for the volume fractions at the inflow boundaries (left and bottom). For the numerical tests, we assume the same square geometry and regular meshes as in the previous example. The L^2-norm of the error relative to the exact solution is depicted for the different meshes in Fig. 5 using elements of order 1 and 2. Finally, in Fig. 6, we compare the convergence of the residual when employing the Newton-Raphson method or Picard iteration, respectively. The presented results are obtained using the mesh of 16×16 elements; however, the results of the other considered discretizations are qualitatively similar. We may point out again that the stabilization parameters and projection terms are always linearized by a simple fixed-point iteration – even when we employ the Newton-Raphson scheme for linearizing the other terms in the weak form. This will eventually lead to suboptimal convergence of the Newton-Raphson scheme in the asymptotic regime. For both methods,

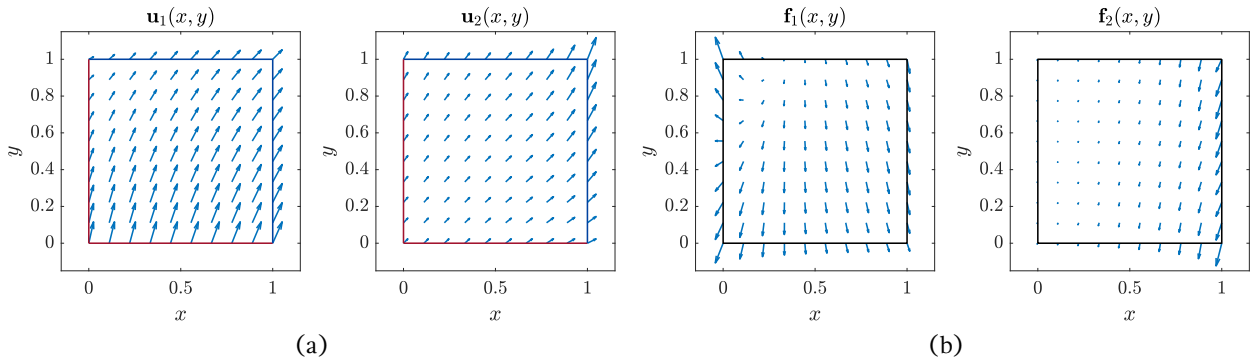
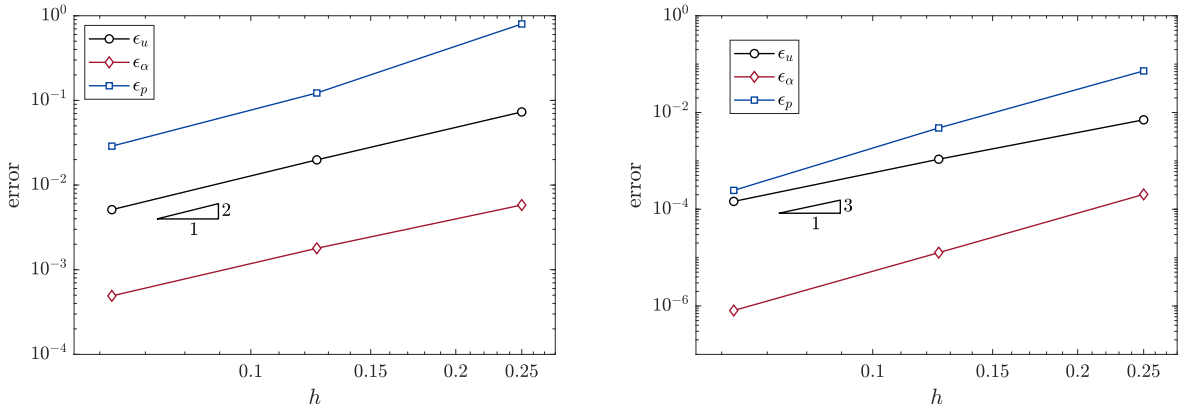


Figure 4: Analytical velocity fields (a) and body loads (b) in the example analyzed in Section. 6.1.2.



(a) $p_e = 1$ (b) $p_e = 2$

Figure 5: Convergence of the error in the L^2-norm under h-refinement, employing linear and quadratic finite elements.

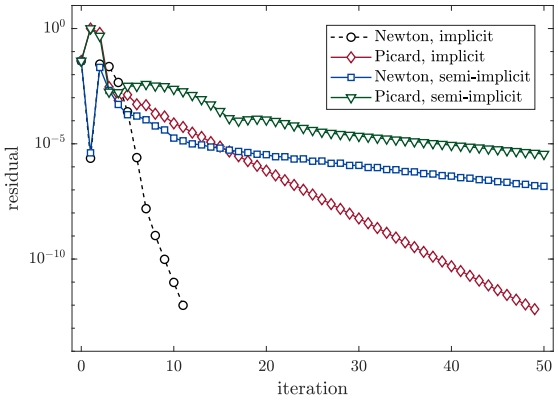


Figure 6: Convergence of the residual when employing Newton-Raphson or Picard iteration with an either implicit or explicit treatment of the projection terms.

we include results obtained when treating the projection terms implicitly and semi-implicitly, as discussed in Section 5.6. The Newton-Raphson method with implicit treatment of the projection requires only nine iterations to yield a residual below 10^{-10} and leads to significantly faster convergence in the asymptotic regime compared to the Picard iteration or the Newton-Raphson scheme with a semi-implicit treatment of the projection. From these results, it seems recommendable to employ the Newton-Raphon scheme with an implicit treatment of the projection for stationary problems.

6.2. *Venturi tube flow*

This example is inspired by the problem described in Section 5.1 of [7], representing a stationary two-phase flow through the tapered tube depicted in Fig. 7, sometimes referred to as *Venturi tube*. The length of the structure is $L = 5$; the thickness of the tube is taken as 1 and reduces to 0.4 at $x = L/2$. The following velocity profile of both phases is prescribed at the inflow boundary (left)

$$\mathbf{u}_1(0,y) = \mathbf{u}_2(0,y) = (4(1-y)y, 0)\,, \tag{52a}$$

and the volume fraction of the dispersed phase equals 1% at the inlet, i.e.,

$$\alpha_1(0,y) = 0.01, \quad \alpha_2(0,y) = 0.99\,. \tag{53}$$

At the outflow boundary, vanishing vertical velocity components and normal stress rates are prescribed, i.e.,

$$\mathbf{u}_{1y} = \mathbf{u}_{2y} = \mathbf{n}\cdot\mathbf{n}\cdot\boldsymbol{\sigma}_1 = \mathbf{n}\cdot\mathbf{n}\cdot\boldsymbol{\sigma}_2 = 0\,, \tag{54}$$

where $\mathbf{n}$ denotes the unit outward normal vector at the outflow. No-slip conditions are assumed for both velocities at the upper and lower walls. The initial conditions are chosen to be constant throughout the domain:

$$\mathbf{u}_1^0(x,y) = \mathbf{u}_2^0(x,y) = (0,0), \quad \alpha_1^0(x,y) = 0.01, \quad \alpha_2^0(x,y) = 0.99, \quad p^0(x,y) = 0 \qquad \text{in } \Omega.$$

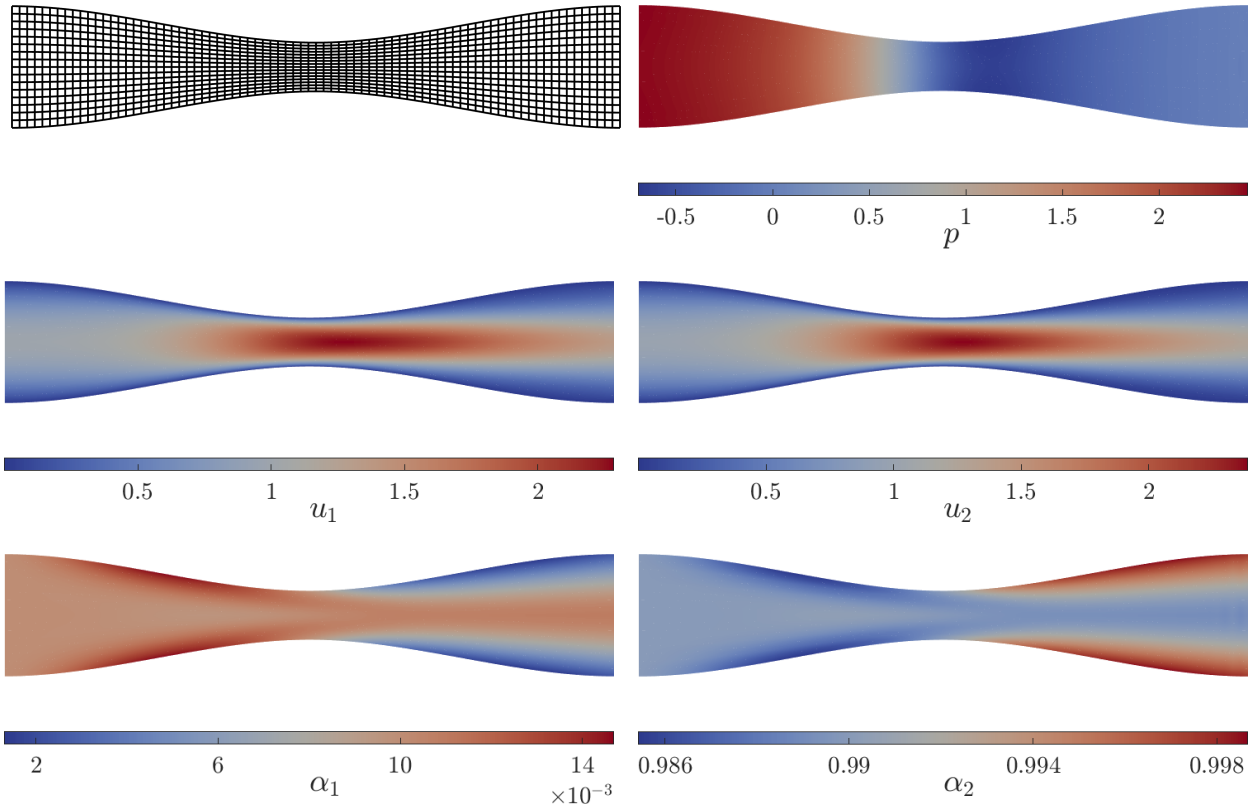


Figure 7: Mesh and numerical solution of the velocities, volume fractions as well as the pressure for the example of the Venturi tube flow.

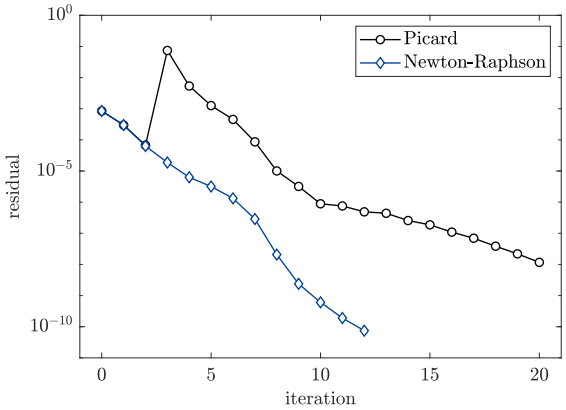


Figure 8: Convergence of the residual in velocity when employing the Newton-Raphson scheme or Picard iteration.

The phase interaction term is also adopted from [7] as

$$g = \frac{3}{4}\frac{C_d}{d}\alpha_1|\mathbf{u}_2 - \mathbf{u}_1|\,, \tag{55}$$

with the drag coefficient $C_d = 0.44$ and the particle diameter $d = 0.005$. Computations have been performed with different Reynolds numbers and density ratios as in [7] to confirm consistency with the previously published results. Here, we present exemplary results using the material parameters

$$\rho_1 = 2, \quad \rho_2 = 1, \quad \mu_1 = \mu_2 = 0.01\,,$$

which corresponds to Reynolds numbers of the two phases of $\mathrm{Re}_1 = 200$, $\mathrm{Re}_2 = 100$. Figure 7 shows the mesh as well as the results of the velocity components, volume fractions, and pressure. While the flow velocities of both phases are rather similar, the moment exchange interaction results in an increased concentration of the dispersed phase near the boundary in the narrowing section of the tube and in the center near the outlet. The results are in accordance with the contour plots presented in [7]. Again, the solution to this stationary problem was obtained after a few iterations of the nonlinear solver. Figure 8 depicts the decay in the scaled residual when employing the Newton-Raphson scheme in comparison with the Picard iteration. In both cases, the projection of the stabilization terms has been treated implicitly. In this example, Picard's iteration led to a sharp increase in the residual after a few iterations before decreasing continuously. On the other hand, the difference between both methods in the asymptotic regime is less pronounced compared to the previous test cases.

6.3. Two-phase flow over a cylinder

In this section, we study a variation of a typical benchmark example, namely, the flow over a cylinder modified to include a mixture of two fluids. We choose the computational domain (the same as, e.g., in [19]) to be $\Omega = [0,16] \times [-4,4] \setminus D$, where D is a unit circle centered at the point $(0,4)$. At the inflow, $x = 0$, we prescribe a uniform mixture of both phases entering the domain at

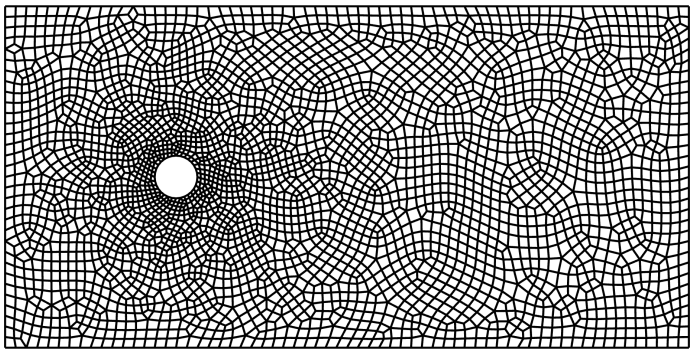


Figure 9: Mesh used for the simulation of two-phase flow over a cylinder.

the same velocity. The boundary conditions are summarized as

$$\begin{aligned} &\text{inflow:} \quad \mathbf{u}_1(x{=}0,y,t) = \mathbf{u}_2(x{=}0,y,t) = (1,0), \quad \alpha_1(x{=}0,y,t) = \alpha_2(x{=}0,y,t) = 0.5\,, \\ &\text{walls:} \quad \mathbf{n}\cdot\mathbf{u}_1(x,y{=}\pm 4,t) = \mathbf{n}\cdot\mathbf{u}_2(x,y{=}\pm 4,t) = 0\,. \end{aligned}$$

All velocity components are unconstrained at the outflow. Uniform initial conditions are chosen as

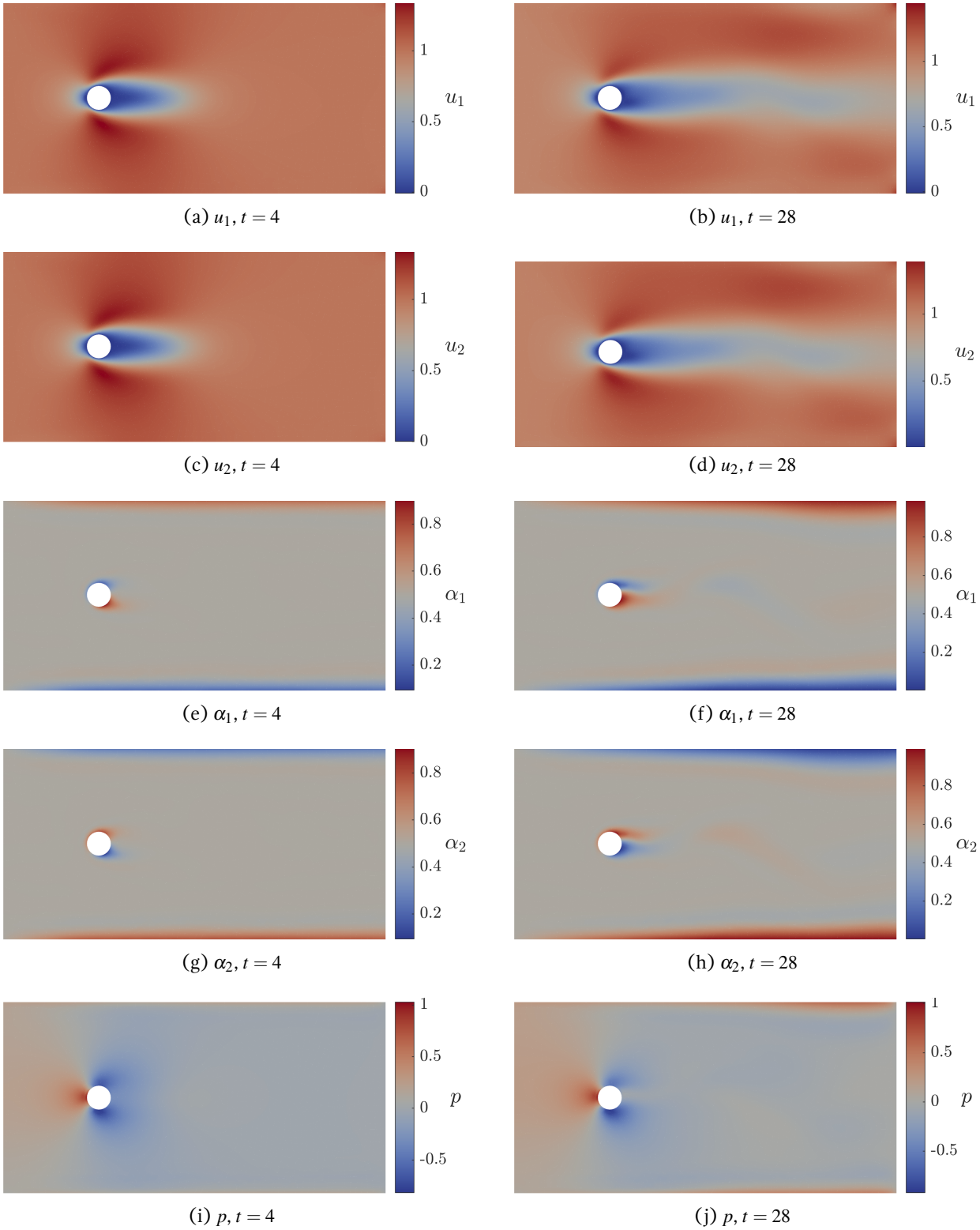
$$\mathbf{u}_1^0 = (1,0), \quad \mathbf{u}_2^0 = (0.9,0), \quad \alpha_1^0 = \alpha_2^0 = 0.5, \quad p^0 = 0 \qquad \text{in } \Omega.$$

The different initial conditions of the two phases were chosen such that the moment exchange term is nonzero in the initial configuration which, in some cases, was found to facilitate the convergence of the nonlinear terms in the first time step. The material parameters of the two constituents are chosen as

$$\mu_1 = 0.01, \quad \mu_2 = 0.1, \quad \rho_1 = 1, \quad \rho_2 = 1.3\,.$$

In this example, buoyancy is considered with a gravitational acceleration of $g = 9.81$, leading to a partial separation of the two fluids as the mixture propagates through the computational domain. The drag term is chosen according to Eq. (45) with $C_D = 0.44$ and $d = 0.001$. The time scale is obtained as 0.0079, and the characteristic velocities in the shock-capturing term are $|\mathbf{u}_{k,c}| = 1$.

The used mesh, consisting of 3284 bilinear elements, is depicted in Fig. 9. Solutions are computed up to $T = 28$ with a time step of $\delta t = 0.05$. Exemplary results of velocities, volume fractions, and pressure at two time instances $t_1 = 4$, $t_2 = 28$ are presented in Fig. 10. The velocity fields roughly resemble those of a single-phase flow. In this example, the velocities of both phases are very similar as a consequence of the relatively large drag term that penalizes velocity differences. The phase separation is well visible in the contour plots of the volume fractions, showing that the lighter/heavier fluid accumulates at the upper/lower wall near the outflow. Separation and subsequent remixing also take place around the cylinder.



(a) $u_1, t = 4$ (b) $u_1, t = 28$

(c) $u_2, t = 4$ (d) $u_2, t = 28$

(e) $\alpha_1, t = 4$ (f) $\alpha_1, t = 28$

(g) $\alpha_2, t = 4$ (h) $\alpha_2, t = 28$

(i) $p, t = 4$ (j) $p, t = 28$

Figure 10: Numerical results of the velocities, volume fractions, and pressure for the example of two-phase flow over a cylinder.

7. Conclusion

The proposed formulation has been confirmed to yield stable solutions to the dispersed multiphase problem and attains optimal convergence rates with respect to spatial and temporal discretization. Thanks to the concept of the so-called term-by-term stabilization – made possible by orthogonal subgrid scales – the stabilization is inexpensive and relatively straightforward to implement compared to alternatives that include the complete residual. Numerical studies have provided evidence of the broad applicability of this method to problems of practical relevance, underscoring its potential to drive further advances in the field of multiphase flow simulations.

Appendix A. Averaging

Different types of volume averages are commonly employed in the derivation of multiphase equations; the most important ones include [7]:

$$\langle q_i \rangle = \frac{1}{V} \int_{V_i} q_i \, \mathrm{d}V \qquad \text{partial average} \tag{A.1a}$$

$$\widetilde{q}_i = \frac{1}{V_i} \int_{V_i} q_i \, \mathrm{d}V = \frac{1}{\alpha_i} \langle q_i \rangle \qquad \text{intrinsic average} \tag{A.1b}$$

$$\bar{q}_i = \frac{\int_{V_i} \rho_i q_i \, \mathrm{d}V}{\int_{V_i} \rho_i \, \mathrm{d}V} = \frac{\langle \rho_i q_i \rangle}{\alpha_i \widetilde{\rho}_i} \qquad \text{Favré/mass-weighted average} \tag{A.1c}$$

Appendix B. Linearization

To ease implementation, we present in Tab. B.1 an overview of all terms to be included in the linearized weak form when employing the fixed-point (Picard) iteration or the Newton-Raphson method. Note that, when using the Newton-Raphson method, the terms included in the column *Picard terms* remain the same, except for a modification in the momentum-exchange term.

Table B.1: Overview of linearization terms.

	Picard terms	additional Newton terms		
Galerkin terms	$(\beta_k, \alpha_k^i \nabla\cdot \mathbf{u}_k^{i+1})$	$(\beta_k, \alpha_k^{i+1} \nabla\cdot \mathbf{u}_k^i)$	$-(\beta_k, \alpha_k^i \nabla\cdot \mathbf{u}_k^i)$	
	$(\beta_k, \mathbf{u}_k^i \cdot\nabla \alpha_k^{i+1})$	$(\beta_k, \mathbf{u}_k^{i+1} \cdot\nabla \alpha_k^i)$	$-(\beta_k, \mathbf{u}_k^i \cdot\nabla \alpha_k^i)$	
	$(\beta_k, \partial_t \alpha_k^{i+1})$			
	$(\mathbf{v}_k, \rho_k \alpha_k^i \mathbf{u}_k^i \cdot\nabla \mathbf{u}_k^{i+1})$	$(\mathbf{v}_k, \rho_k \alpha_k^i \mathbf{u}_k^{i+1} \cdot\nabla \mathbf{u}_k^i)$	$(\mathbf{v}_k, \rho_k \alpha_k^{i+1} \mathbf{u}_k^i \cdot\nabla \mathbf{u}_k^i)$	$-2\,(\mathbf{v}_k, \rho_k \alpha_k^i \mathbf{u}_k^i \cdot\nabla \mathbf{u}_k^i)$
	$(2\boldsymbol{\varepsilon}(\mathbf{v}_k), \alpha_k^i \mu_k \boldsymbol{\varepsilon}(\mathbf{u}_k^{i+1}))$	$(2\boldsymbol{\varepsilon}(\mathbf{v}_k), \alpha_k^{i+1} \mu_k \boldsymbol{\varepsilon}(\mathbf{u}_k^i))$	$-(2\boldsymbol{\varepsilon}(\mathbf{v}_k), \alpha_k^i \mu_k \boldsymbol{\varepsilon}(\mathbf{u}_k^i))$	
	$-(\alpha_k^i \nabla\cdot \mathbf{v}_k, p^{i+1})$	$-(\alpha_k^{i+1} \nabla\cdot \mathbf{v}_k, p^i)$	$(\alpha_k^i \nabla\cdot \mathbf{v}_k, p^i)$	
	$-(\mathbf{v}_k \cdot\nabla \alpha_k^i, p^{i+1})$	$-(\mathbf{v}_k \cdot\nabla \alpha_k^{i+1}, p^i)$	$(\mathbf{v}_k \cdot\nabla \alpha_k^i, p^i)$	
	$(\mathbf{v}_k, \rho_k \alpha_k^i \partial_t \mathbf{u}_k^{i+1})$	$(\mathbf{v}_k, \rho_k \alpha_k^{i+1} \partial_t \mathbf{u}_k^i)$	$-(\mathbf{v}_k, \rho_k \alpha_k^i \partial_t \mathbf{u}_k^i)$	
	$-(\mathbf{v}_k, \alpha_k^{i+1} \mathbf{f}_k)$			
	$-(\mathbf{v}_k, C_g b_g \alpha_k^i \alpha_\ell^i \mathbf{w}^{i+1})^\dagger$	$-(\mathbf{v}_k, C_g \alpha_k^{i+1} \alpha_\ell^i w^i \mathbf{w}^i)$	$-(\mathbf{v}_k, C_g \alpha_k^i \alpha_\ell^{i+1} w^i \mathbf{w}^i)$	$3(\mathbf{v}_k, C_g \alpha_k^i \alpha_\ell^i w^i \mathbf{w}^i)$
	$(q, \alpha_k^i \nabla\cdot \mathbf{u}_k^{i+1})$	$(q, \alpha_k^{i+1} \nabla\cdot \mathbf{u}_k^i)$	$-(q, \alpha_k^i \nabla\cdot \mathbf{u}_k^i)$	
	$(q, \mathbf{u}_k^{i+1} \cdot\nabla \alpha_k^i)$	$(q, \mathbf{u}_k^i \cdot\nabla \alpha_k^{i+1})$	$-(q, \mathbf{u}_k^i \cdot\nabla \alpha_k^i)$	
	$-t_0^{-1}(q, \alpha_k^{i+1})$			
	$t_0^{-1}(q, 1)$			
stabilization	$(\mathbf{u}_k^i \cdot\nabla \beta_k, \tau_k^\alpha \mathbf{u}_k^i \cdot\nabla \alpha_k^{i+1})$	$(\mathbf{u}_k^{i+1} \cdot\nabla \beta_k, \tau_k^\alpha \mathbf{u}_k^i \cdot\nabla \alpha_k^i)$	$(\mathbf{u}_k^i \cdot\nabla \beta_k, \tau_k^\alpha \mathbf{u}_k^{i+1} \cdot\nabla \alpha_k^i)$	$-2\,(\mathbf{u}_k^i \cdot\nabla \beta_k, \tau_k^\alpha \mathbf{u}_k^i \cdot\nabla \alpha_k^i)$
	$(\nabla \beta_k, \lvert\mathbf{u}_k\rvert^2 \tau_k^\alpha \nabla \alpha_k^{i+1})$			
	$(\nabla q, \tau_k^u \alpha_k^i \nabla p^{i+1})$	$(\nabla q, \tau_k^u \alpha_k^{i+1} \nabla p^i)$	$-(\nabla q, \tau_k^u \alpha_k^i \nabla p^i)$	
	$(\mathbf{u}_k^i \cdot\nabla \mathbf{v}_k, \rho_k^2 \tau_k^u \alpha_k^i \mathbf{u}_k^i \cdot\nabla \mathbf{u}_k^{i+1})$	$(\mathbf{u}_k^i \cdot\nabla \mathbf{v}_k, \rho_k^2 \tau_k^u \alpha_k^i \mathbf{u}_k^{i+1} \cdot\nabla \mathbf{u}_k^i)$	$(\mathbf{u}_k^i \cdot\nabla \mathbf{v}_k, \rho_k^2 \tau_k^u \alpha_k^{i+1} \mathbf{u}_k^i \cdot\nabla \mathbf{u}_k^i)$	
		$(\mathbf{u}_k^{i+1} \cdot\nabla \mathbf{v}_k, \rho_k^2 \tau_k^u \alpha_k^i \mathbf{u}_k^i \cdot\nabla \mathbf{u}_k^i)$	$-3\,(\mathbf{u}_k^i \cdot\nabla \mathbf{v}_k, \rho_k^2 \tau_k^u \alpha_k^i \mathbf{u}_k^i \cdot\nabla \mathbf{u}_k^i)$	

† Newton linearization: $b_g = 2$, Picard iteration: $b_g = w^i$

References

[1] M. Manninen, V. Taivassalo, On the mixture model for multiphase flow, VTT Publications (288) (1996) 3–67.

[2] E. Oñate, M. A. Celigueta, S. Latorre, G. Casas, R. Rossi, J. Rojek, Lagrangian analysis of multiscale particulate flows with the particle finite element method, Computational Particle Mechanics 1 (1) (2014) 85–102.

[3] A. Y. Varaksin, Fluid dynamics and thermal physics of two-phase flows: problems and achievements, High Temperature 51 (3) (2013) 377–407.

[4] E. Delnoij, J. A. Kuipers, W. P. Van Swaaij, Dynamic simulation of gas-liquid twophase flow: effect of column aspect ratio on the flow structure, Chemical Engineering Science 52 (21-22) (1997) 3759–3772.

[5] I. K. Park, H. K. Cho, H. Y. Yoon, J. J. Jeong, Numerical effects of the semi-conservative form of momentum equations for multi-dimensional two-phase flows, Nuclear Engineering and Design 239 (2009) 2365–2371.

[6] F. Behrangi, M. Ali Banihashemi, M. M. Namin, A. Bohluly, A new approach to solve mixture multi-phase flow model using time splitting projection method, Progress in Computational Fluid Dynamics 19 (3) (2019) 160–169.

[7] K. Hiltunen, A stabilized finite element method for particulate two-phase flow equations laminar isothermal flow, Computer Methods in Applied Mechanics and Engineering 147 (1997) 387–399.

[8] C. T. Jacobs, G. S. Collins, M. D. Piggott, S. C. Kramer, C. R. G. Wilson, Multiphase flow modelling of volcanic ash particle settling in water using adaptive unstructured meshes, Geophysical Journal International 192 (2) (2013) 647–665.

[9] T. S. Dang, G. Meschke, An ALE-PFEM method for the numerical simulation of two-phase mixture flow, Computer Methods in Applied Mechanics and Engineering 278 (2014) 599–620.

[10] C. Caia, P. Minev, A finite element method for an averaged multiphase flow model, International Journal of Computational Fluid Dynamics 18 (2) (2004) 111–123.

[11] A. W. Vreman, Stabilization of the Eulerian model for incompressible multiphase flow by artificial diffusion, Journal of Computational Physics 230 (4) (2011) 1639–1651.

[12] R. Codina, On stabilized finite element methods for linear systems of convection-diffusion-reaction equations, Computer Methods in Applied Mechanics and Engineering 188 (1-3) (2000) 61–82.

[13] T. J. R. Hughes, Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods, Computer Methods in Applied Mechanics and Engineering 127 (1995) 387–401.

[14] T. J. R. Hughes, G. R. Feijóo, L. Mazzei, J. B. Quincy, The variational multiscale method - a paradigm for computational mechanics, Computer Methods in Applied Mechanics and Engineering 166 (1998) 3–24.

[15] R. Codina, Stabilization of incompressibility and convection through orthogonal sub-scales in finite element methods, Computer Methods in Applied Mechanics and Engineering 190 (2000) 1579–1599.

[16] E. Castillo, R. Codina, Finite element approximation of the viscoelastic flow problem: A non-residual based stabilized formulation, Computers and Fluids 142 (2017) 72–78.

[17] R. Codina, Analysis of a stabilized finite element approximation of the Oseen equations using orthogonal subscales, Applied Numerical Mathematics 58 (3) (2008) 264–283.

[18] H. Coppola-Owen, R. Codina, A free surface finite element model for low Froude number mould filling problems on fixed meshes, International Journal for Numerical Methods in Fluids 66 (2011) 833–851.

[19] E. Castillo, R. Codina, Dynamic term-by-term stabilized finite element formulation using orthogonal subgrid-scales for the incompressible Navier-Stokes problem, Computer Methods in Applied Mechanics and Engineering 349 (2019) 701–721.

[20] R. Codina, Stabilized finite element approximation of transient incompressible flows using orthogonal subscales, Computer Methods in Applied Mechanics and Engineering 191 (2002) 4295–4321.

[21] R. Codina, J. Principe, O. Guasch, S. Badia, Time dependent subscales in the stabilized finite element approximation of incompressible flow problems, Computer Methods in Applied Mechanics and Engineering 196 (21-24) (2007) 2413–2430.

[22] O. Colomés, S. Badia, R. Codina, J. Principe, Assessment of variational multiscale models for the large eddy simulation of turbulent incompressible flows, Computer Methods in Applied Mechanics and Engineering 285 (2015) 32–63.

[23] K. Hiltunen, A. Jäsberg, S. Kallio, H. Karema, M. Kataja, A. Koponen, M. Manninen, V. Taivassalo, Multiphase Flow Dynamics - Theory and Numerics, VTT Publilcations, 2009.

[24] H. Städtke, Gasdynamic Aspects of Two-Phase Flow, Wiley-VCH, 2006.

[25] R. Codina, S. Badia, J. Baiges, J. Principe, Variational Multiscale Methods in Computational Fluid Dynamics, in: E. Stein, R. Borst, T. Hughes (Eds.), Encyclopedia of Computational Mechanics Second Edition, 2nd Edition, 2017, pp. 1–28.

[26] J. Baiges, R. Codina, Variational Multiscale error estimators for solid mechanics adaptive simulations: An Orthogonal Subgrid Scale approach, Computer Methods in Applied Mechanics and Engineering 325 (2017) 37–55.
[27] L. Moreno, R. Codina, J. Baiges, Solution of transient viscoelastic flow problems approximated by a term-by-term VMS stabilized finite element formulation using time-dependent subgrid-scales, Computer Methods in Applied Mechanics and Engineering 367 (2020) 113074.
[28] R. Codina, J. Principe, Dynamic subscales in the finite element approximation of thermally coupled incompressible flows, International Journal for Numerical Methods in Fluids 54 (2007) 707–730.
[29] J. Principe, R. Codina, On the stabilization parameter in the subgrid scale approximation of scalar convection-diffusion-reaction equations on distorted meshes, Computer Methods in Applied Mechanics and Engineering 199 (2010) 1386–1402.
[30] R. Codina, Comparison of some finite element methods for solving the diffusion-convection-reaction equation, Computer Methods in Applied Mechanics and Engineering 156 (1998) 185–210.
[31] S. S. Karunarathne, L.-A. Tokheim, Comparison of the influence of drag models in CFD simulation of particle mixing and segregation in a rotating cylinder, in: 58th Conference on Simulation and Modelling, Vol. 138, 2017, pp. 151–156.
[32] C. Jacobs, Modelling of Multiphase Flows on Adaptive Unstructured Meshes with Applications to the Dynamics of Volcanic Ash Plumes, Ph.D. thesis, Imperial College London (2013).
[33] R. Codina, Pressure Stability in Fractional Step Finite Element Methods for Incompressible Flows, Journal of Computational Physics 170 (1) (2001) 112–140.