

Preconditioning of a coupled Cahn–Hilliard Navier–Stokes system

Jessica Bosch* Christian Kahle[†] Martin Stoll[‡]

February 16, 2017

Abstract

Recently, Garcke et al. [H. Garcke, M. Hinze, C. Kahle, Appl. Numer. Math. 99 (2016), 151–171] developed a consistent discretization scheme for a thermodynamically consistent diffuse interface model for incompressible two-phase flows with different densities [H. Abels, H. Garcke, G. Grün, Math. Models Methods Appl. Sci. 22(3) (2012)]. At the heart of this method lies the solution of large and sparse linear systems that arise in a semismooth Newton method.

In this work we propose the use of preconditioned Krylov subspace solvers using effective Schur complement approximations. Numerical results illustrate the efficiency of our approach. In particular, our preconditioner is shown to be robust with respect to parameter changes.

Keywords: Navier–Stokes, Cahn–Hilliard, two-phase flow, preconditioning, Schur complement approximation

1 Introduction

In recent years numerical simulation of two-phase flow and multiphase flow has attained growing attention both from the modeling point of view and the development of schemes for the numerical realization. This especially holds for so called diffuse interface or phase-field models, where the distribution of the two phases, i.e. the regions where the two phases are located, is encoded in a smooth indicator function, called phase field. In this work we consider the particular problem of solving linear systems arising during Newton’s method for the solution of nonlinear equations appearing in an energy stable time stepping scheme for a thermodynamically consistent model for two-phase flow.

*Department of Computer Science, The University of British Columbia, Vancouver, BC, V6T 1Z4, Canada (jbosch@cs.ubc.ca).

[†]Chair of Optimal Control, Technical University München, Boltzmannstr. 3, 85748 Garching bei München (Christian.kahle@ma.tum.de).

[‡]Numerical Linear Algebra for Dynamical Systems, Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstr. 1, 39106 Magdeburg, Germany (stollm@mpi-magdeburg.mpg.de).

We use the thermodynamically consistent model proposed in [AGG12] and the numerical scheme presented in [GHK16]. The model is based on a phase field representation of the two-phase fluid structure and couples the Navier–Stokes equations to the Cahn–Hilliard equations in a thermodynamically consistent way, i.e., an energy equality holds.

Before we state the model and the scheme below in Section 2, let us briefly comment on the development of models for two-phase flows and especially on numerical schemes for these. The first model for two-phase flow using a phase field representation is the model ‘H’ proposed in [HH77]. It couples the Navier–Stokes equation and the Cahn–Hilliard equation [CH58] in a thermodynamically consistent way. It is only valid for fluids of equal density. Since then several attempts have been made for a generalization to fluids of different densities, see e.g. [LT98, SY10, Boy02]. The first thermodynamically consistent model, that contains the same coupling between the Navier–Stokes and the Cahn–Hilliard model is proposed in [AGG12]. Especially for equal density fluids, this model equals the model ‘H’.

In the few years since the invention of this model, several groups proposed schemes for its numerical realization and we refer here only to [GT14, GGM16, GK14, Ala14, HKW15, GHK16]. The first two schemes decouple the Navier–Stokes and the Cahn–Hilliard equation. This means that on each time instance, the Cahn–Hilliard equation can be solved in advance and the solution can be used to solve the Navier–Stokes equation afterwards. The systems are decoupled by using an augmented velocity field in the Cahn–Hilliard equation, that then only depends on the velocity field from the old time instance and thus can be solved before turning to the Navier–Stokes equation. This procedure adds some diffusion outside of the interface to the Cahn–Hilliard equation. In this way, the defects from using the old velocity field can be balanced with numerical dissipation that arises due to time discretization to obtain an energy stable scheme.

The other papers propose fully coupled schemes and especially in [Ala14] benefits from using a fully coupled scheme are investigated. At the heart of the numerical realization of such fully coupled schemes large and sparse linear systems have to be solved, that have a block structure, namely

$$\mathcal{A} = \left(\begin{array}{c|c} A_{NS} & C_I \\ \hline C_T & A_{CH} \end{array} \right),$$

where A_{NS} and A_{CH} are themselves block matrices arising from the discretization of the (linearized) Navier–Stokes equation (A_{NS}) and the (linearized) Cahn–Hilliard equation (A_{CH}), while C_T describes coupling through transport of the interface and C_I describes coupling through interfacial forces.

The arising sparse linear systems are usually of very large dimension, and in combination with three-dimensional experiments, the application of direct solvers such as UMFPACK [Dav16] quickly becomes infeasible. As a result, iterative methods have to be employed (see, e.g., [Gre97, Saa03] for introductions to this field). The state-of-the-art iterative schemes are the so-called Krylov subspace methods with prominent representatives found in [HS52, PS75, SS86b].

The convergence behavior of the iterative schemes typically depends on the conditioning of the problem and the clustering of the eigenvalues. These properties are usually affected by the model parameters. Various parameters of different scales are involved in \mathcal{A} : Spatial mesh sizes, the time step size, the interfacial parameter, the mobility of the interface, a penalty parameter, the Reynolds number, and the surface tension. Therefore, the convergence behavior of the iterative solver needs to be enhanced using preconditioning techniques $\mathcal{A}\mathcal{P}^{-1}\tilde{\mathbf{z}} = \mathbf{b}$ or $\mathcal{P}^{-1}\mathcal{A}\mathbf{z} = \mathcal{P}^{-1}\mathbf{b}$, where \mathcal{P} is an invertible matrix that is easy to invert and resembles \mathcal{A} . Our goal in this paper is the derivation and analysis of a preconditioner well suited for the coupled system \mathcal{A} . To the best of our knowledge this is the first time that the preconditioning of the coupled Navier-Stokes-Cahn-Hilliard system is studied. While preconditioning for the Navier-Stokes equations has been investigated for a long time (see [EJW05] and the references given therein), the development of preconditioners for Cahn-Hilliard systems is a rather new topic with several contributions found in [BSB14, BDQN11]. The coupling of both systems poses serious challenges regarding robustness of the method with respect to the crucial system parameters. Hence, we derive an efficient preconditioner \mathcal{P} tailored to the coupled Cahn-Hilliard Navier-Stokes system using effective Schur complement approximations and (algebraic) multigrid developed for elliptic systems [Fal06, Saa03, RS87]. For this we utilize recent developments for the preconditioning of the Navier-Stokes equations using a pressure-convection-diffusion approximation to the Navier-Stokes Schur complement [EJW05] and a multigrid for the (1, 1)-block of the discrete Navier-Stokes system. We then approximate the Schur complement of the overall system using the recently established techniques for the Cahn-Hilliard equations that are in turn based on an efficient approximation of the Schur complement of this block using a matching argument, well established in PDE-constrained optimization [PW12, PSW12].

The paper is organized as follows. In Section 2, we introduce the two-phase flow equation under consideration. We state the thermodynamically consistent discretization proposed in [GHK16] and review major analytical results. In Section 3, we develop and investigate our preconditioner, while in Section 4, we show its performance in a benchmark example and investigate its robustness with respect to the relevant parameters of the system under investigation. Finally, Section 5 summarizes our findings.

2 The Cahn-Hilliard Navier-Stokes model

In [AGG12], the following thermodynamically consistent diffuse interface model for the simulation of two-phase fluids with different densities is proposed:

Let $\Omega \subset \mathbb{R}^n$, $n \in \{2, 3\}$ denote an open, bounded domain with Lipschitz boundary and outer normal ν_Ω , and $I = (0, T]$ denote a time interval. Inside Ω there are two immiscible phases and we introduce a smooth indicator function φ , called phase field, such that $\varphi \approx +1$ indicates the one phase and $\varphi \approx -1$ indicates the second phase. The values $|\varphi| \leq 1$ indicate a diffuse interface

between the phases. The velocity and the pressure of the fluid are denoted by v and p ; see [AGG12] for the precise definition of these terms. Adding a chemical potential μ as first variation of the underlying Ginzburg–Landau energy, the model is given by the following set of equations:

$$\rho \partial_t v + ((\rho v + J) \cdot \nabla) v - \operatorname{div}(2\eta Dv) + \nabla p = \mu \nabla \varphi + \rho g \quad \forall x \in \Omega, \forall t \in I, \quad (1)$$

$$-\operatorname{div}(v) = 0 \quad \forall x \in \Omega, \forall t \in I, \quad (2)$$

$$\partial_t \varphi + v \cdot \nabla \varphi - b \Delta \mu = 0 \quad \forall x \in \Omega, \forall t \in I, \quad (3)$$

$$-\sigma \epsilon \Delta \varphi + \frac{\sigma}{\epsilon} W'(\varphi) - \mu = 0 \quad \forall x \in \Omega, \forall t \in I, \quad (4)$$

$$v(0, x) = v_0(x) \quad \forall x \in \Omega, \quad (5)$$

$$\varphi(0, x) = \varphi_0(x) \quad \forall x \in \Omega, \quad (6)$$

$$v(t, x) = 0 \quad \forall x \in \partial\Omega, \forall t \in I, \quad (7)$$

$$\nabla \mu(t, x) \cdot \nu_\Omega = \nabla \varphi(t, x) \cdot \nu_\Omega = 0 \quad \forall x \in \partial\Omega, \forall t \in I, \quad (8)$$

with $J = -\frac{\rho_2 - \rho_1}{2} b \nabla \mu$.

Here $\rho = \rho(\varphi) := \frac{\rho_2 - \rho_1}{2} \varphi + \frac{\rho_2 + \rho_1}{2}$ describes the interpolated density of the fluid, and $\eta = \eta(\varphi) := \frac{\eta_2 - \eta_1}{2} \varphi + \frac{\eta_2 + \eta_1}{2}$ the interpolated viscosity, where ρ_1, ρ_2 are the densities of the two pure phases, while η_1, η_2 are the corresponding viscosities. By $2Dv = \nabla v + (\nabla v)^t$ we denote the symmetrized gradient. The gravitational acceleration is denoted by g . By b we denote the constant mobility of the interface. The scaled surface tension (see [AGG12, Sec. 4.3.4]) is given by σ and the interfacial region between the phases has a thickness of order $O(\epsilon)$. The free energy density of the interface is denoted by W and here we use

$$W(\varphi) = W^s(\varphi) = \frac{1}{2} \left(1 - \varphi^2 + s \max(0, \varphi - 1)^2 + s \min(0, \varphi + 1)^2 \right),$$

where $s \gg 0$ denotes a relaxation coefficient arising from the Moreau–Yosida relaxation of the double-obstacle free energy

$$W^\infty(\varphi) = \begin{cases} \frac{1}{2}(1 - \varphi^2) & \text{if } |\varphi| \leq 1, \\ +\infty & \text{else;} \end{cases}$$

see [BE91, HHT11]. Note that the results proposed in [GHK16] are valid for a wider class of free energy densities. Finally, the initial velocity is given by v_0 and the initial phase field by φ_0 . Note that for simplicity we use a no-slip boundary condition for the velocity field, while the scheme is literally valid as long as no flux across $\partial\Omega$ appears, i.e. $v \cdot \nu_\Omega \equiv 0$ on $\partial\Omega$. This especially includes free-slip conditions

$$v \cdot \nu_\Omega = 0, \quad \nu_\Omega^\perp \cdot (2\eta Dv) \nu_\Omega = 0 \quad \text{on } \partial\Omega,$$

where ν_Ω^\perp denotes the tangential on $\partial\Omega$.

Concerning existence and uniqueness of solutions, we refer to [ADG13a, ADG13b, Grü13].

The following weak formulation of (1)–(8) is proposed in [GHK16].

Definition 1. We call v, p, φ, μ a weak solution to (1)–(8) if $v(0) = v_0, \varphi(0) = \varphi_0$, and

$$\begin{aligned} & \frac{1}{2} \int_{\Omega} (\partial_t(\rho v) + \rho \partial_t v) w \, dx + \int_{\Omega} 2\eta Dv : Dw \, dx \\ & + a(\rho v + J, v, w) - (p, \operatorname{div} w) = \int_{\Omega} \mu \nabla \varphi w + \rho g w \, dx \quad \forall w \in H_0^1(\Omega)^n, \end{aligned} \quad (9)$$

$$-(\operatorname{div} v, q) = 0 \quad \forall q \in L_{(0)}^2(\Omega), \quad (10)$$

$$\int_{\Omega} (\partial_t \varphi + v \cdot \nabla \varphi) \Psi \, dx + \int_{\Omega} b \nabla \mu \cdot \nabla \Psi \, dx = 0 \quad \forall \Psi \in H^1(\Omega), \quad (11)$$

$$\sigma \epsilon \int_{\Omega} \nabla \varphi \cdot \nabla \Phi \, dx + \frac{\sigma}{\epsilon} \int_{\Omega} W'(\varphi) \Phi \, dx - \int_{\Omega} \mu \Phi \, dx = 0 \quad \forall \Phi \in H^1(\Omega), \quad (12)$$

is satisfied for almost all $t \in I$.

Here, for $u \in L^3(\Omega)^n, v, w \in H^1(\Omega)^n$, we define the antisymmetric trilinear form

$$a(u, v, w) := \frac{1}{2} \left(\int_{\Omega} ((u \nabla) v) w \, dx - \int_{\Omega} ((u \nabla) w) v \, dx \right),$$

and by $L_{(0)}^2(\Omega)$ we denote the space of square integrable functions with mean value zero, i.e.

$$L_{(0)}^2(\Omega) := \{f \in L^2(\Omega) \mid |\Omega|^{-1} \int_{\Omega} f = 0\}.$$

We stress, that formulation (9)–(12) is based on a reformulation of (1), that requires that ρ is a linear function of φ . In [AB16] a generalization of (1)–(8) is presented, where this requirement might be dropped. This has to be investigated in the future.

Definition 2. The energy of the system is the sum of the kinetic energy of the fluid and the Ginzburg–Landau energy of the interface. Thus we define

$$E(t) := E(v(t), \varphi(t)) := \int_{\Omega} \frac{1}{2} \rho |v|^2 \, dx + \sigma \int_{\Omega} \frac{\epsilon}{2} |\nabla \varphi|^2 + \frac{1}{\epsilon} W(\varphi) \, dx.$$

The model (1)–(8) is called *thermodynamically consistent*, because the following energy identity holds.

Theorem 3 ([GHK16, Thm. 1]). Assume that there exists a sufficiently smooth solution to (9)–(12).

Then, the following energy identity holds for $0 < \tau < t$

$$\begin{aligned} E(t) &+ \int_{\tau}^t \int_{\Omega} 2\eta(\varphi(s)) |Dv(s)|^2 + b |\nabla \mu(s)|^2 \, dx \, ds \\ &= E(\tau) + \int_{\tau}^t \int_{\Omega} \rho(\varphi(s)) g v(s) \, dx \, ds. \end{aligned} \quad (13)$$

This means that the gain and loss of energy can exactly be measured and that in absence of outer forces (i.e. $g \equiv 0$) the energy can not increase with time.

In [GHK16] a discretization scheme is proposed, that resembles this property in the fully discrete setting. We summarize that scheme in the following.

2.1 The stable scheme from [GHK16]

For a numerical realization, we discretize (9)–(12) using a semi-implicit Euler discretization in time and the finite element method in space.

Let $t_{-1} < 0 = t_0 < t_1 < \dots < t_{k-2} < t_{k-1} < t_k < \dots < t_K = T$ denote an equidistant subdivision of $I = (0, T]$ with fix step size $\tau := t_1 - t_0$. Further, let $\mathcal{T}^k = \{T_1, \dots, T_{\Theta_k}\}$ denote a conforming triangulation of $\bar{\Omega}$ with closed cells $(T_i)_{i=1}^{\Theta_k}$, where we assume that $\bar{\Omega} = \bigcup_{i=1}^{\Theta_k} T_i$. Here, k refers to the time instance, and we stress that we use different triangulations on different time instances due to adaptive meshing. On \mathcal{T}^k , we define the finite element spaces

$$\begin{aligned} V_1^k &:= \{v \in C(\bar{\Omega}) \mid v|_T \in \Pi^1(\Omega) \forall T \in \mathcal{T}^k\} =: \text{span}\{b_1^i\}_{i=1}^{N_1^k}, \\ V_2^k &:= \{v \in C(\bar{\Omega})^n \mid v|_T \in (\Pi^2(\Omega))^n \forall T \in \mathcal{T}^k, v|_{\partial\Omega} = 0\} =: \text{span}\{b_2^i\}_{i=1}^{N_2^k}. \end{aligned}$$

Here, Π^l denotes the space of polynomials up to order l . We further introduce an H^1 -stable prolongation operator $P^k : H^1(\Omega) \rightarrow V_1^k$. Possible operators are, e.g., the Clément operator or Lagrangian interpolation, where we have to restrict the preimage to $C(\bar{\Omega}) \cap H^1(\Omega)$.

To state the fully discrete approximation, we introduce $\varphi_h^k \in V_1^k$ as fully discrete approximation of $\varphi(t_k)$, $\mu_h^k \in V_1^k$ as fully discrete approximation of $\mu(t_k)$, $v_h^k \in V_2^k$ as fully discrete approximation of $v(t_k)$, and $p_h^k \in V_1^k$ as fully discrete approximation of $p(t_k)$. Note that we use LBB-stable Taylor–Hood elements for the discretization of the Navier–Stokes equation. We further use the abbreviations $\rho^k := \rho(\varphi_h^k)$, $\eta^k := \eta(\varphi_h^k)$.

The fully discrete variant of (9)–(12) is as follows.

Let $\varphi^{k-2} \in V_1^{k-2}$, $\varphi^{k-1} \in V_1^{k-1}$, $\mu^{k-1} \in V_1^{k-1}$, $v^{k-1} \in V_2^{k-1}$ be given. Find $\varphi^k \in V_1^k$, $\mu^k \in V_1^k$, $v^k \in V_2^k$, and $p_h^k \in V_1^k$ such that

$$\begin{aligned} \frac{1}{\tau} \left(\frac{\rho^{k-1} + \rho^{k-2}}{2} v_h^k - \rho^{k-2} v^{k-1}, w \right) + a(\rho^{k-1} v^{k-1} + J^{k-1}, v_h^k, w) \\ + (2\eta^{k-1} Dv_h^k, Dw) - (p_h^k, \text{div}w) - (\mu_h^k \nabla \varphi^{k-1} + \rho^{k-1} g, w) = 0 \quad \forall w \in V_2^k, \end{aligned} \quad (14)$$

$$-(\text{div}v_h^k, q) = 0 \quad \forall q \in V_1^k, \quad (15)$$

$$\frac{1}{\tau} (\varphi_h^k - P^k \varphi^{k-1}, \Psi) + (b \nabla \mu_h^k, \nabla \Psi) - (v_h^k \varphi^{k-1}, \nabla \Psi) = 0 \quad \forall \Psi \in V_1^k, \quad (16)$$

$$\sigma \epsilon (\nabla \varphi_h^k, \nabla \Phi) + \frac{\sigma}{\epsilon} (W'_+(\varphi_h^k) + W'_-(P^k \varphi^{k-1}), \Phi) - (\mu_h^k, \Phi) = 0 \quad \forall \Phi \in V_1^k. \quad (17)$$

By $W_+(\varphi) = \frac{\epsilon}{2}(\max(0, \varphi - 1)^2 + \min(0, \varphi + 1)^2)$ we denote the convex part of W , and by $W_-(\varphi) = \frac{1}{2}(1 - \varphi^2)$ we denote the concave part.

Remark 4. Note that (14) – (17) is a two-step scheme since it requires data from t_{k-2} and t_{k-1} to evaluate the solution at time t_k . Especially, we require the somewhat artificial data φ^{-1} . In [GHK16], a one-step scheme with a different time discretization is proposed for the initialization of the two-step scheme (14) – (17). Here, we argue as in [HKW15] that given φ^{-1} and v^0 one can solve (16) – (17) to obtain the missing data φ^0 and μ^0 to start the two-step scheme. Compare also [GHK16, Rem. 3]

Theorem 5 ([GHK16, Thm. 2, Thm. 3]). Let $\varphi^{k-2} \in V_1^{k-2}$, $\varphi^{k-1} \in V_1^{k-1}$, $\mu^{k-1} \in V_1^{k-1}$, $v^{k-1} \in V_2^{k-1}$ be given. Then there exists a unique solution $v_h^k \in V_2^k$, $\varphi_h^k, \mu_h^k, p_h^k \in V_1^k$ to (14)–(17) that fulfills the following energy inequality

$$\begin{aligned} & \int_{\Omega} \frac{1}{2} \rho^{k-1} |v_h^k|^2 dx + \sigma \int_{\Omega} \frac{\epsilon}{2} |\nabla \varphi_h^k|^2 + \frac{1}{\epsilon} W(\varphi_h^k) dx \\ & + \frac{1}{2} \int_{\Omega} \rho^{k-2} |v_h^k - v^{k-1}|^2 dx + \frac{\sigma \epsilon}{2} \int_{\Omega} |\nabla \varphi_h^k - \nabla P^k \varphi^{k-1}|^2 dx \\ & \quad + \tau \int_{\Omega} 2\eta^{k-1} |Dv_h^k|^2 dx + \tau \int_{\Omega} b |\nabla \mu_h^k|^2 dx \quad (18) \\ & \leq \int_{\Omega} \frac{1}{2} \rho^{k-2} |v^{k-1}|^2 dx + \sigma \int_{\Omega} \frac{\epsilon}{2} |\nabla P^k \varphi^{k-1}|^2 + \frac{1}{\epsilon} W(P^k \varphi^{k-1}) dx \\ & \quad + \tau \int_{\Omega} \rho^{k-1} g v_h^k dx. \end{aligned}$$

Moreover, this solution can be found by Newton's method.

Proof. The existence of a solution is shown using Brouwer's fixpoint theorem. The energy inequality then follows from using $w = v_h^k$, $q = p_h^k$, $\Psi = \mu_h^k$ and $\Phi = \frac{1}{\epsilon}(\varphi_h^k - P^k \varphi^{k-1})$, summing the equations up and using the properties of W_+ and W_- . Then the uniqueness follows from considering two different solutions and showing that they are equal, using the energy inequality.

To show that Newton's method is applicable, we have to show that system (14)–(17) is Newton-differentiable and that the derivative is continuously invertible. Beside W_+ all terms are Fréchet differentiable and thus Newton differentiable. W_+ is Newton differentiable as shown in [HIK03]. The derivative has a structure, that is very similar to (14)–(17) and the continuous invertibility follows similar to the existence of a unique solution. \square

Remark 6. Note that the energy inequality (18) bounds the energy at time instance k by the energy of $P^k \varphi^{k-1}$, i.e., the prolongation of φ^{k-1} to the triangulation \mathcal{T}^k , thus not by the energy at the old time instance. To overcome this, in [GHK16], a postprocessing step is added to the adaptive concept that guarantees that the energy does not increase through prolongation, where Lagrangian interpolation is used as prolongation P^k . Including this step, we guarantee that in absence of outer forces the energy of the system can not increase and (18) resembles a fully discrete counterpart of (13).

Remark 7. Using the unique solution proposed in Theorem 5 in [GHK16] the existence of a unique solution to a corresponding time-discrete variant of (14)–(17) is shown.

Theorem 5 states that we find the unique solution to (14)–(17) by Newton’s method. To state the algorithm, let us compactly write (14)–(17) as

$$\langle F(v_h^k, p_h^k, \varphi_h^k, \mu_h^k), (w, q, \Psi, \Phi) \rangle := \left((F^1(\dots), w), (F^2(\dots), q), (F^3(\dots), \Psi), (F^4(\dots), \Phi) \right)^\dagger,$$

where $(F^1(\dots), w)$ abbreviates (14), $(F^2(\dots), \Psi)$ abbreviates (15), $(F^3(\dots), \Phi)$ abbreviates (16), and $(F^4(\dots), \Psi)$ abbreviates (17). Then Newton’s method generates the following sequence

$$\begin{aligned} DF(x^m)\delta x &= -F(x^m), \\ x^{m+1} &= x^m + \delta x, \quad m = 1, \dots \end{aligned} \quad (19)$$

where x abbreviates $(v_h^k, p_h^k, \varphi_h^k, \mu_h^k)$. We note that F is due to our choice of W only Newton differentiable, see [HIK03, HHT11]. A Newton derivative of F is given by

$$\begin{aligned} \langle DF(v_h^k, p_h^k, \varphi_h^k, \mu_h^k) (\delta v, \delta p, \delta \varphi, \delta \mu), (w, q, \Psi, \Phi) \rangle := & \\ & \frac{1}{\tau} \left(\frac{\rho^{k-1} + \rho^{k-2}}{2} \delta v, w \right) + a(\rho^{k-1} v^{k-1} + J^{k-1}, \delta v, w) \\ & + (2\eta^{k-1} D\delta v, Dw) - (\delta p, \operatorname{div} w) - (\delta \mu \nabla \varphi^{k-1}, w) \\ & - (\operatorname{div} \delta v, q) \\ & + \frac{1}{\tau} (\delta \varphi, \Psi) + (b \nabla \delta \mu, \nabla \Psi) - (\delta v \varphi^{k-1}, \nabla \Psi) \\ & + \sigma \epsilon (\nabla \delta \varphi, \nabla \Phi) + \frac{\sigma}{\epsilon} (W'_+(\varphi_h^k) \delta \varphi, \Phi) - (\delta \mu, \Phi), \end{aligned} \quad (20)$$

where

$$W'_+(\varphi_h^k(x)) := \begin{cases} s & \text{if } |\varphi_h^k(x)| > 1, \\ 0 & \text{else} \end{cases} \quad \forall x \in \Omega.$$

Solving (19) leads to the numerical solution of a large linear system that we pose next.

Using the basis $\{b_2^i\} \subset V_2^k$ and $\{b_1^i\} \subset V_1^l$ we define the following matrices and vectors:

$$\begin{aligned} A &:= M_2 + T_a + K_2, \\ M_2 &= (m_{ij}^2)_{i,j=1}^{N_2}, & T_a &= (t_{ij}^a)_{i,j=1}^{N_2}, & K_2 &= (k_{ij}^2)_{i,j=1}^{N_2}, \\ m_{ij}^2 &:= \left(\frac{\rho^{k-1} + \rho^{k-2}}{2\tau} b_2^j, b_2^i \right), & t_{ij}^a &:= a(\rho^{k-1} v^{k-1} + J^{k-1}, b_2^j, b_2^i), & k_{ij}^2 &:= (2\eta^{k-1} D b_2^j, D b_2^i), \end{aligned}$$

$$\begin{aligned}
B &:= (b_{ij})_{i=1,\dots,N_1}^{j=1,\dots,N_2}, & U &:= (\xi_{ij})_{i=1,\dots,N_2}^{j=1,\dots,N_1}, & T &:= (t_{ij})_{i=1,\dots,N_1}^{j=1,\dots,N_2}, \\
b_{ij} &= -(\operatorname{div} b_2^j, b_1^i), & \xi_{ij} &= -(b_1^j \nabla \varphi^{k-1}, b_2^i), & t_{ij} &= (b_2^j \varphi^{k-1}, \nabla b_1^i), \\
M_1 &:= (m_{ij}^1)_{ij=1,\dots,N_1}, & K_1 &:= (k_{ij}^1)_{ij=1,\dots,N_1}, & \Lambda &:= (\lambda_{ij})_{ij=1}^{N_1}, \\
m_{ij}^1 &= (b_1^j, b_1^i), & k_{ij}^1 &= (\nabla b_1^j, \nabla b_1^i), & \lambda_{ij} &= (W_+''(\varphi^m) b_1^j, b_1^i), \\
F_1 &= (f_j^1)_{j=1}^{N_2}, \quad f_j^1 = (F^1(x^m), b_2^j), & F_2 &= (f_j^2)_{j=1}^{N_1}, \quad f_j^2 = (F^2(x^m), b_1^j), \\
F_3 &= (f_j^3)_{j=1}^{N_1}, \quad f_j^3 = (F^3(x^m), b_1^j), & F_4 &= (f_j^4)_{j=1}^{N_1}, \quad f_j^4 = (F^4(x^m), b_1^j).
\end{aligned}$$

Here, φ^m denotes the m -th iterate of Newton's method for the approximation of φ_h^k . Equation (19) can then be written as

$$\left(\begin{array}{cc|cc} A & B^t & U & 0 \\ B & 0 & 0 & 0 \\ \hline 0 & 0 & M_1 & -\sigma\epsilon K_1 - \sigma\epsilon^{-1}\Lambda \\ \tau T & 0 & \tau b K_1 & M_1 \end{array} \right) \begin{pmatrix} \delta v \\ \delta p \\ \delta \mu \\ \delta \varphi \end{pmatrix} = \begin{pmatrix} F_1 \\ F_2 \\ -F_4 \\ F_3 \end{pmatrix}. \quad (21)$$

where $\delta v, \delta p, \delta \mu, \delta \varphi$ denote the node vectors for $\delta v, \delta p, \delta \varphi, \delta \mu$. Note that here we already did some reformulation in order to maintain our later analysis. In the following, we denote the coefficient matrix in (21) by \mathcal{A} . We can further write \mathcal{A} as

$$\mathcal{A} = \left(\begin{array}{c|c} A_{\text{NS}} & \tilde{C}_I \\ \hline \tilde{C}_T & \tilde{A}_{\text{CH}} \end{array} \right), \quad (22)$$

where the blocks A_{NS} and \tilde{A}_{CH} are the discrete realizations of the Navier-Stokes and linearized Cahn-Hilliard system, respectively. Their coupling is represented by \tilde{C}_I , the coupling through the interfacial force, and \tilde{C}_T , the coupling through the transport at the interface. The matrix A is invertible by Lax-Milgram's theorem (note the antisymmetric transport term), and the $(1, 1)$ block A_{NS} then is invertible since we use LBB-stable elements. The $(2, 2)$ block \tilde{A}_{CH} is invertible; see, e.g., [HHT11]. The Schur complement of the whole system, $S = \tilde{A}_{\text{CH}} - \tilde{C}_T A_{\text{NS}}^{-1} \tilde{C}_I$, describes a Cahn-Hilliard system with additional transport. S is at least invertible for small time steps; see, e.g., [Kah14].

3 Preconditioning

As we have seen in the previous section, a large and sparse linear nonsymmetric system is at the heart of the computation. In [GHK16], the system (21) is solved by preconditioned GMRES [SS86a] with a restart after 10 iterations. The authors use the block diagonal preconditioner

$$\mathcal{P} = \left(\begin{array}{cc} \hat{A}_{\text{NS}} & \mathbf{0} \\ \mathbf{0} & \hat{A}_{\text{CH}} \end{array} \right). \quad (23)$$

The (2,2) block \tilde{A}_{CH} is inverted by LU decomposition. The (1,1) block \hat{A}_{NS} is an upper block triangular preconditioner of the form

$$\hat{A}_{\text{NS}} = \begin{pmatrix} \hat{A} & B^t \\ 0 & \hat{S}_{\text{NS}} \end{pmatrix}. \quad (24)$$

\hat{A} is composed of the diagonal blocks of A and is inverted by LU decomposition. \hat{S}_{NS} is an approximation of the exact Schur complement $S_{\text{NS}} = -BA^{-1}B^t$ of the Navier–Stokes system. Garcke et al. [GHK16] use

$$\hat{S}_{\text{NS}} = -K_p A_p^{-1} M_p, \quad (25)$$

where $M_p = M_1$ is the pressure mass matrix, $K_p = K_1$ the pressure Laplacian matrix, and A_p is the representation of A on the pressure space, i.e., $A_p := M_{2,p} + T_{a,p} + K_{2,p}$, where

$$\begin{aligned} M_{2,p} &= (m_{ij}^2)_{i,j=1}^{N_1}, & m_{ij}^2 &:= \left(\frac{\rho^{k-1} + \rho^{k-2}}{2\tau} b_1^j, b_1^i \right), \\ T_{a,p} &= (t_{ij}^a)_{i,j=1}^{N_1}, & t_{ij}^a &:= a(\rho^{k-1} v^{k-1} + J^{k-1}, b_1^j, b_1^i), \\ K_{2,p} &= (k_{ij}^2)_{i,j=1}^{N_1}, & k_{ij}^2 &:= (2\eta^{k-1} D b_1^j, D b_1^i). \end{aligned}$$

This Schur complement approximation was proposed, e.g., in [KLW02, E JW05], where it was shown to be independent of the mesh size and only mildly dependent on the Reynolds number.

Our aim here is to further improve the performance of the preconditioner as a tailored iterative method for the system (21) can reduce the computing time dramatically. In order to achieve this, we first derive a general strategy for the coupled system and then show that for this to work well both the Navier–Stokes part of the discretized equation as well as the Cahn–Hilliard part need to be approximated by sophisticated strategies. This is based on the preconditioning techniques that have been developed in [BSB14] for the nonsmooth Cahn–Hilliard system together with the methods that have been developed in [KLW02, E JW05] for the Navier–Stokes equations and where partly previously used in [GHK16]. Let us restate the linear system (21) for convenience here

$$\left(\begin{array}{cc|cc} A & B^t & U & 0 \\ B & 0 & 0 & 0 \\ \hline 0 & 0 & M_1 & -\sigma\epsilon K_1 - \sigma\epsilon^{-1}\Lambda \\ \tau T & 0 & \tau b K_1 & M_1 \end{array} \right) \begin{pmatrix} \delta v \\ \delta p \\ \delta \mu \\ \delta \varphi \end{pmatrix} = \begin{pmatrix} F_1 \\ F_2 \\ -F_4 \\ F_3 \end{pmatrix}, \quad (26)$$

and we recall that we denote the coefficient matrix in (26) by \mathcal{A} that we can partition as

$$\mathcal{A} = \left(\begin{array}{c|c} A_{\text{NS}} & C_I \\ \hline C_T & A_{\text{CH}} \end{array} \right). \quad (27)$$

Our basis for preconditioning is the upper block triangular preconditioner

$$\mathcal{P} = \begin{pmatrix} A_{\text{NS}} & C_I \\ \mathbf{0} & S \end{pmatrix},$$

motivated in [EJW05, MGW00], where $S = A_{\text{CH}} - C_T A_{\text{NS}}^{-1} C_I$ is the Schur complement of the whole system as introduced above after Equation (22). The right-preconditioned matrix becomes

$$\mathcal{A}\mathcal{P}^{-1} = \left(\begin{array}{c|c} I & \mathbf{0} \\ \hline C_T A_{\text{NS}}^{-1} & I \end{array} \right),$$

where I is the identity matrix. Hence, $\mathcal{A}\mathcal{P}^{-1}$ has only a single eigenvalue of 1, and \mathcal{P} is called a theoretical optimal preconditioner.¹ This preconditioner needs the application of the inverse of A_{NS} and of S , which cannot be explicitly used. Hence, our aim is the development of practical approximations $\hat{A}_{\text{NS}} \approx A_{\text{NS}}$ and $\hat{S} \approx S$, which leads to our practical preconditioner

$$\mathcal{P}_{\text{out}} = \begin{pmatrix} \hat{A}_{\text{NS}} & C_I \\ \mathbf{0} & \hat{S} \end{pmatrix}. \quad (28)$$

Note that the index of \mathcal{P}_{out} marks an outer preconditioner. Below in Section 3.2, we will introduce an inner preconditioner for the solution of linear systems with \hat{S} . For good approximations $\hat{A}_{\text{NS}} \approx A_{\text{NS}}$ and $\hat{S} \approx S$, the preconditioned matrix $\mathcal{A}\mathcal{P}_{\text{out}}^{-1}$ has only a small number of different eigenvalue clusters. This in turn is known to result in only a few iterations of suitable Krylov subspace solvers until convergence [EJW05, MGW00].

3.1 Approximation of A_{NS}

Similar to Garcke et al. [GHK16] in (24), we choose \hat{A}_{NS} as

$$\hat{A}_{\text{NS}} = \begin{pmatrix} \hat{A} & B^t \\ 0 & -\hat{S}_{\text{NS}} \end{pmatrix}. \quad (29)$$

As above, \hat{A} is composed of the diagonal blocks of A . We use an algebraic multigrid (AMG) preconditioner² for the approximation of the inverse of \hat{A} . As shown in [Ram99], multigrid is a good preconditioner for convection-diffusion problems if the Reynolds number is not too large. \hat{S}_{NS} is given in (25). The action of the inverse of M_p and K_p are performed with an AMG each. Note that

¹Note, that the left-preconditioned system $\mathcal{P}^{-1}\mathcal{A}$ has the same spectrum as the right-preconditioned system $\mathcal{A}\mathcal{P}^{-1}$.

²AMG methods typically exhibit geometric-like properties for positive definite elliptic type operators but use only algebraic information. This has the advantage that AMG can work well even for complicated geometries and meshes. We refer to [RS87, Fal06] for more information on AMG. We also want to emphasize that geometric multigrid (see, e.g., [Wes92, Hac85]) approximations are also well suited to approximate \hat{A} provided they can be readily applied.

the conjugate gradient method with Jacobi preconditioning provides a good approximation to the inverse of M_p as well, see e.g. [KLW02]. K_p is a discrete Laplacian for which multigrid provides a good approximation to the inverse; see [Wes92].

3.2 Approximation of S

Now, let us consider the Schur complement $S = A_{\text{CH}} - C_T A_{\text{NS}}^{-1} C_I$ of the whole system. Using the preconditioner \hat{A}_{NS} from the previous section, we approximate S as

$$S \approx A_{\text{CH}} - C_T \hat{A}_{\text{NS}}^{-1} C_I = \begin{pmatrix} M_1 & -\sigma\epsilon K_1 - \sigma\epsilon^{-1}\Lambda \\ \tau b K_1 - \tau T \hat{A}^{-1} U & M_1 \end{pmatrix} =: \hat{S}.$$

We propose to apply a preconditioned GMRES iteration to the system of the form $\hat{S}y = f$. We call this iteration the inner iteration. For the construction of the preconditioner \mathcal{P}_{in} for the inner iteration, we make use of the following formulation:

$$\hat{S} = \begin{pmatrix} M_1 & -\sigma\epsilon K_1 - \sigma\epsilon^{-1}\Lambda \\ \tau b K_1 & M_1 \end{pmatrix} - \begin{pmatrix} 0 & 0 \\ \tau T \hat{A}^{-1} U & 0 \end{pmatrix} = A_{\text{CH}} - \begin{pmatrix} 0 & 0 \\ \tau T \hat{A}^{-1} U & 0 \end{pmatrix} \approx A_{\text{CH}}.$$

Hence, we build the preconditioner \mathcal{P}_{in} for the inner iteration on the basis of the simplification A_{CH} of \hat{S} .

In the following, we study the importance of taking the whole block A_{CH} as preconditioner \mathcal{P}_{in} . Remember from Equation (26)

$$A_{\text{CH}} = \begin{pmatrix} M_1 & -\sigma\epsilon K_1 - \sigma\epsilon^{-1}\Lambda \\ \tau b K_1 & M_1 \end{pmatrix}.$$

Now, if we would further simplify A_{CH} to

$$\hat{A}_{\text{CH}} = \begin{pmatrix} M_1 & -\sigma\epsilon K_1 \\ \tau b K_1 & M_1 \end{pmatrix},$$

we could easily derive an optimal preconditioner for \hat{A}_{CH} ; see, e.g., [AN11, BDQN11]. However, Theorem 8 and Corollary 9 below state the severe influence of the penalty parameter s contained in Λ ; see also [Bos16]. Hence, a simplification to \hat{A}_{CH} would give a worse approximation for large values of s . This justifies our approach of taking A_{CH} as preconditioner \mathcal{P}_{in} .

For the following theorem, we make use of the symmetric positive definiteness of M_1 as well as of the symmetric positive semidefiniteness of K_1 and Λ .

Theorem 8. With the simplified notation $M := M_1$, $K := \tau b K_1$, $\alpha = \frac{\sigma \varepsilon}{\tau b}$, $\beta = \frac{\sigma}{\varepsilon}$, let

$$\mathbf{X} = \begin{pmatrix} M & -\alpha K \\ K & M \end{pmatrix}, \quad \mathbf{Y} = \begin{pmatrix} M & -\alpha K - \beta \Lambda \\ K & M \end{pmatrix}.$$

Note that $\mathbf{X} = \mathring{A}_{\text{CH}}$ and $\mathbf{Y} = A_{\text{CH}}$. Then,

$$\text{sp}(\mathbf{X}^{-1}\mathbf{Y}) \subseteq B_\zeta(1),$$

where $\text{sp}(\cdot)$ denotes the spectrum of a matrix, and $B_\zeta(1)$ is a circle in the complex plane around one with radius ζ . The radius is bounded by $\frac{\beta}{2\sqrt{\alpha}} \rho(\tilde{\Lambda})$, where $\tilde{\Lambda} = M^{-\frac{1}{2}} \Lambda M^{-\frac{1}{2}}$.

Proof. It holds $\text{sp}(\mathbf{X}^{-1}\mathbf{Y}) \subseteq B_\zeta(1)$ if and only if $\text{sp}(\mathbf{I} - \mathbf{X}^{-1}\mathbf{Y}) \subseteq B_\zeta(0)$ if and only if $\rho(\mathbf{X}^{-1}(\mathbf{X} - \mathbf{Y})) < \zeta$. Here, $\rho(\cdot)$ denotes the spectral radius of a matrix, and \mathbf{I} is the identity matrix of appropriate size.

Let us start with finding an equivalent formulation for $\rho(\mathbf{X}^{-1}(\mathbf{X} - \mathbf{Y}))$. With the notation $\mathbf{C} = \mathbf{K}\mathbf{M}^{-1}$ and $\mathbf{W} = \mathbf{I} + \alpha\mathbf{C}^2$, where \mathbf{I} is again the identity matrix of appropriate size, we have

$$\mathbf{X} = \begin{pmatrix} \mathbf{I} & -\alpha\mathbf{C} \\ \mathbf{C} & \mathbf{I} \end{pmatrix} \begin{pmatrix} \mathbf{M} & 0 \\ 0 & \mathbf{M} \end{pmatrix},$$

$$\mathbf{X}^{-1} = \begin{pmatrix} \mathbf{M}^{-1} & 0 \\ 0 & \mathbf{M}^{-1} \end{pmatrix} \begin{pmatrix} \mathbf{I} & -\alpha\mathbf{C} \\ \mathbf{C} & \mathbf{I} \end{pmatrix}^{-1}.$$

Using block matrix inversion, we have

$$\begin{pmatrix} \mathbf{I} & -\alpha\mathbf{C} \\ \mathbf{C} & \mathbf{I} \end{pmatrix}^{-1} = \begin{pmatrix} (\mathbf{I} + \alpha\mathbf{C}^2)^{-1} & \alpha(\mathbf{I} + \alpha\mathbf{C}^2)^{-1}\mathbf{C} \\ -(\mathbf{I} + \alpha\mathbf{C}^2)^{-1}\mathbf{C} & (\mathbf{I} + \alpha\mathbf{C}^2)^{-1} \end{pmatrix} = \begin{pmatrix} \mathbf{W}^{-1} & \alpha\mathbf{W}^{-1}\mathbf{C} \\ -\mathbf{W}^{-1}\mathbf{C} & \mathbf{W}^{-1} \end{pmatrix}$$

Hence,

$$\mathbf{X}^{-1}(\mathbf{X} - \mathbf{Y}) = \begin{pmatrix} \mathbf{M}^{-1} & 0 \\ 0 & \mathbf{M}^{-1} \end{pmatrix} \begin{pmatrix} \mathbf{W}^{-1} & \alpha\mathbf{W}^{-1}\mathbf{C} \\ -\mathbf{W}^{-1}\mathbf{C} & \mathbf{W}^{-1} \end{pmatrix} \begin{pmatrix} 0 & \beta\Lambda \\ 0 & 0 \end{pmatrix} = \beta \begin{pmatrix} 0 & (\mathbf{W}\mathbf{M})^{-1}\Lambda \\ 0 & -(\mathbf{W}\mathbf{M})^{-1}\mathbf{C}\Lambda \end{pmatrix}$$

and therefore

$$\rho(\mathbf{X}^{-1}(\mathbf{X} - \mathbf{Y})) = \beta \rho((\mathbf{W}\mathbf{M})^{-1}\mathbf{C}\Lambda).$$

For a further reformulation we introduce the rational function $r(x) = \frac{x}{1+\alpha x^2}$, and define $\tilde{\mathbf{C}} = \mathbf{M}^{-\frac{1}{2}}\mathbf{K}\mathbf{M}^{-\frac{1}{2}}$ and $\tilde{\Lambda} = \mathbf{M}^{-\frac{1}{2}}\Lambda\mathbf{M}^{-\frac{1}{2}}$, which are symmetric positive semidefinite. Then we have

$$\begin{aligned} r(\tilde{\mathbf{C}}) &= (\mathbf{I} + \alpha\tilde{\mathbf{C}}^2)^{-1} \tilde{\mathbf{C}} = (\mathbf{I} + \alpha\mathbf{M}^{-\frac{1}{2}}\mathbf{K}\mathbf{M}^{-1}\mathbf{K}\mathbf{M}^{-\frac{1}{2}})^{-1} \mathbf{M}^{-\frac{1}{2}}\mathbf{K}\mathbf{M}^{-\frac{1}{2}} \\ &= \left[\mathbf{M}^{\frac{1}{2}} (\mathbf{I} + \alpha\mathbf{M}^{-\frac{1}{2}}\mathbf{K}\mathbf{M}^{-1}\mathbf{K}\mathbf{M}^{-\frac{1}{2}}) \right]^{-1} \mathbf{K}\mathbf{M}^{-\frac{1}{2}} = (\mathbf{M}^{\frac{1}{2}} + \alpha\mathbf{K}\mathbf{M}^{-1}\mathbf{K}\mathbf{M}^{-\frac{1}{2}})^{-1} \mathbf{K}\mathbf{M}^{-\frac{1}{2}} \\ &= \left[(\mathbf{I} + \alpha\mathbf{K}\mathbf{M}^{-1}\mathbf{K}\mathbf{M}^{-1}) \mathbf{M}^{\frac{1}{2}} \right]^{-1} \mathbf{K}\mathbf{M}^{-\frac{1}{2}} = \mathbf{M}^{-\frac{1}{2}} (\mathbf{I} + \alpha\mathbf{C}^2)^{-1} \mathbf{K}\mathbf{M}^{-\frac{1}{2}} \\ &= \mathbf{M}^{-\frac{1}{2}} \mathbf{W}^{-1} \mathbf{K}\mathbf{M}^{-\frac{1}{2}} \end{aligned}$$

and hence

$$\begin{aligned} (WM)^{-1}C\Lambda &= M^{-1}W^{-1}KM^{-1}\Lambda = M^{-\frac{1}{2}}\left(M^{-\frac{1}{2}}W^{-1}KM^{-\frac{1}{2}}\right)\left(M^{-\frac{1}{2}}\Lambda M^{-\frac{1}{2}}\right)M^{\frac{1}{2}} \\ &= M^{-\frac{1}{2}}r(\tilde{C})\tilde{\Lambda}M^{\frac{1}{2}}. \end{aligned}$$

It follows

$$\begin{aligned} \rho\left((WM)^{-1}C\Lambda\right) &= \rho\left(r(\tilde{C})\tilde{\Lambda}\right) \leq \|r(\tilde{C})\tilde{\Lambda}\| \leq \|r(\tilde{C})\| \|\tilde{\Lambda}\| = \rho\left(r(\tilde{C})\right) \rho\left(\tilde{\Lambda}\right) \leq \max_{\lambda \in \text{sp}(\tilde{C})} \{r(\lambda)\} \rho\left(\tilde{\Lambda}\right) \\ &\leq \max_x \{r(x)\} \rho\left(\tilde{\Lambda}\right) = \frac{1}{2\sqrt{\alpha}} \rho\left(\tilde{\Lambda}\right), \end{aligned} \quad (30)$$

where we have taken into account that the eigenvalues of $r(Z)$ for a matrix Z with eigenvalues $\lambda_1, \dots, \lambda_q$ are given by $r(\lambda_1), \dots, r(\lambda_q)$. \square

Applied to our situation, we have

$$\text{sp}(\hat{A}_{\text{CH}}^{-1}A_{\text{CH}}) \subseteq B_\varsigma(1),$$

where $\varsigma \leq \frac{\sqrt{\tau\sigma b}}{2\epsilon\sqrt{\epsilon}} \rho(\tilde{\Lambda})$.

Corollary 9. *As a result from Theorem 8, we get $\varsigma \leq 0.5$ when $\tau \leq \epsilon^3/(s^2\sigma b \rho(\tilde{\Lambda}_0)^2)$, where $\tilde{\Lambda}_0 = s^{-1}M^{-\frac{1}{2}}\Lambda M^{-\frac{1}{2}}$. In particular, in the case of lumped mass matrices, we have $\rho(\tilde{\Lambda}_0) = 1$ and hence, the circle radius is bounded by $\varsigma \leq \frac{s\sqrt{\tau\sigma b}}{2\epsilon\sqrt{\epsilon}}$. Hence, we get $\varsigma \leq 0.5$ when $\tau \leq \epsilon^3/(s^2\sigma b)$.*

Proof. In order to illustrate the influence of the penalty parameter s , we reformulate $\tilde{\Lambda}$ as

$$\tilde{\Lambda} = M^{-\frac{1}{2}}\Lambda M^{-\frac{1}{2}} = sM^{-\frac{1}{2}}\Lambda_0 M^{-\frac{1}{2}} = s\tilde{\Lambda}_0,$$

where $\Lambda_0 = (\lambda_{ij}^0)_{ij=1}^{N_1}$, $\lambda_{ij}^0 = (W''_{+,0}(\varphi^m)b_1^j, b_1^i)$, and

$$W''_{+,0}(\varphi_h^k(x)) := \begin{cases} 1 & \text{if } |\varphi_h^k(x)| > 1, \\ 0 & \text{else} \end{cases} \quad \forall x \in \Omega.$$

Hence, we have in (30)

$$\rho\left((WM)^{-1}C\Lambda\right) \leq \frac{s}{2\sqrt{\alpha}} \rho(\tilde{\Lambda}_0) = \frac{s\sqrt{\tau\sigma b}}{2\epsilon\sqrt{\epsilon}} \rho(\tilde{\Lambda}_0), \quad (31)$$

where $\rho(\tilde{\Lambda}_0)$ depends on the spatial mesh size and φ_h^k . Therefore, for $\tau \leq \epsilon^3/(s^2\sigma b \rho(\tilde{\Lambda}_0)^2)$, it holds $\text{sp}(\hat{A}_{\text{CH}}^{-1}A_{\text{CH}}) \subseteq B_{0.5}(1)$.

In the case of lumped mass matrices, $\tilde{\Lambda}_0$ becomes a diagonal matrix with entries that are either zero or one. Thus, we obtain in (31)

$$\rho\left((WM)^{-1}C\Lambda\right) \leq \frac{s}{2\sqrt{\alpha}} = \frac{s\sqrt{\tau\sigma b}}{2\epsilon\sqrt{\epsilon}}.$$

Therefore, for $\tau \leq \varepsilon^3/(s^2\sigma b)$, it holds $\text{sp}(\hat{A}_{CH}^{-1}A_{CH}) \subseteq B_{0.5}(1)$. \square

Hence, neglecting the block Λ in A_{CH} would only be satisfying for tiny time step sizes τ , which is far away from being practical.

As preconditioner for the inner iteration, we propose the upper block triangular preconditioner

$$\mathcal{P}_{\text{in}} = \begin{bmatrix} M_1 & -\sigma\varepsilon K_1 - \sigma\varepsilon^{-1}\Lambda \\ 0 & -\hat{S}_{CH} \end{bmatrix}. \quad (32)$$

\hat{S}_{CH} is an approximation of the exact Schur complement

$$S_{CH} = M_1 + \tau(bK_1 - T\hat{A}^{-1}U)M_1^{-1}(\sigma\varepsilon K_1 + \sigma\varepsilon^{-1}\Lambda)$$

of \hat{S} . Our procedure for the approximation of the Schur complement S_{CH} originates in the work of Pearson and Wathen [PW12], who developed preconditioners for PDE-constrained optimization. Their matching strategy, applied to our problem, is the following: Construct a preconditioner of the form $\hat{S}_{CH} = S_1M_1^{-1}S_2$, which captures the exact Schur complement S_{CH} as close as possible. Note that we need \hat{S}_{CH} to be nonsingular. We design \hat{S}_{CH} as

$$\begin{aligned} \hat{S}_{CH} &= S_1M_1^{-1}S_2 \\ &= (M_1 + \sqrt{\tau\sigma b}K_1)M_1^{-1}(M_1 + \sqrt{\tau b\sigma^{-1}}[\sigma\varepsilon K_1 + \sigma\varepsilon^{-1}\Lambda]) \\ &= M_1 + \tau bK_1M_1^{-1}(\sigma\varepsilon K_1 + \sigma\varepsilon^{-1}\Lambda) + \sqrt{\tau b\sigma}K_1 + \sqrt{\tau b\sigma}(\varepsilon K_1 + \varepsilon^{-1}\Lambda). \end{aligned} \quad (33)$$

The first term in (33) matches the first term in the exact Schur complement. The second term in (33) approximates the second term in the exact Schur complement. Due to the factor $\sqrt{\tau b\sigma}$, the influence of both remainder terms in (33) is reduced. We refer the reader to [BSB14, BKSW14, BS15, Bos16] for Schur complement approximations to other Cahn–Hilliard problems. The action of the inverse of S_1 and S_2 is performed with an AMG each since both form the discretization of an elliptic operator. We also apply AMG for the action of the inverse of the $(1, 1)$ block M_1 in \mathcal{P}_{in} .³

Here, we finish the theoretical discussion about the preconditioner. In the next section, we illustrate its efficiency via various numerical experiments.

4 Numerical examples

In this section, we show numerical results for the presented coupled Cahn–Hilliard Navier–Stokes problem. First, we explain our implementation framework.

³For consistent mass matrices, the Chebyshev-iteration [GV61a, GV61b] provides a powerful preconditioner [WR09, RS10].

Garcke et al. [GHK16] have implemented the whole numerical simulation in C++. We use their code, but the iterative solution of the linear system is executed in MATLAB® R2012a on a 32-bit server with CPU type Intel® Core™ E6850 @3.00 GHz with 2 CPUs. We use the MATLAB Engine API in order to call MATLAB® from C++.

As we use GMRES as the inner iteration, we apply FGMRES as the outer iteration. FGMRES is a variant of GMRES and was introduced by Saad [Saa93]. This method allows changes in the preconditioner at every step. Note again that we apply right preconditioning here. We use FGMRES (outer solver) with a restart after 30 iterations and set the initial guess to the zero vector. As stopping criterion, we use

$$\|\mathbf{b} - \mathcal{A}\mathbf{z}^{(l)}\| \leq \min(10^{-6} \|\mathbf{b}\|, 10^{-6}),$$

where \mathbf{b} denotes the corresponding right-hand side and $\mathbf{z}^{(l)}$ the calculated solution at FGMRES step l . We use GMRES (inner solver) without restart and set the initial guess to the zero vector, the tolerance for the preconditioned relative residual to 10^{-1} , and the maximum number of iterations to 50.

For the application of AMG, we employ the HSL (formerly the Harwell Subroutine Library) Mathematical Software Library, a collection of Fortran codes for large-scale scientific computation; see <http://www.hsl.rl.ac.uk/> and [BMS10]. In particular, we make use of the *HSL_M120* package for the approximation of the inverse of \hat{A} (the two diagonal blocks of the discrete convection-diffusion operator) and of K_p (the pressure Laplacian matrix). For \hat{A} , we apply *HSL_M120* with two coarse levels in the multigrid structure, two V-cycles with symmetric Gauss–Seidel smoothing (two pre- and two post-smoothing iterations), and 10 Gauss-Seidel iterations on the coarsest level. For K_p , we apply *HSL_M120* with at most 100 coarse levels in the multigrid structure, one V-cycle with symmetric Gauss–Seidel smoothing (two pre- and two post-smoothing iterations), and the sparse direct solver *HSL_MA48* on the coarsest level.

Moreover, we employ the software package *AGMG* version 3.2.0, a program which implements AMG described in [Not10] with further improvements from [NN12] and [Not12]; see also <http://homepages.ulb.ac.be/~ynotay/AGMG>. It introduces K-cycle multigrid meaning that the iterative solution of the residual equation at each level is accelerated with a Krylov subspace method. We use the default, which is a preconditioned variant of the generalized conjugate residual method (GCR) [EES83] restarted every 10 iterations. We set the maximum number of iterations to 50. The coarsening is stopped when the coarse grid matrix has 200 or less rows, allowing fast direct inversion with LAPACK routines [ABB⁺99]. Symmetric Gauss-Seidel smoothing is used with one pre- and one post-smoothing iteration. We make use of the *AGMG* MATLAB interface for the approximation of the inverse of M_p (the pressure mass matrix), of M_1 (the (1,1) block of the inner preconditioner \mathcal{P}_{in}), as well as of S_1 and S_2 (the two blocks forming the Schur complement approximation \hat{S}_{CH}). If not mentioned

otherwise, we set the tolerance on the relative residual norm to 10^{-3} , 10^{-2} , 10^{-5} , and 10^{-5} for M_p , M_1 , S_1 , and S_2 , respectively.

4.1 Parameter study: A rising bubble

In this section, we demonstrate the robustness of our proposed preconditioner regarding relevant model parameters. As test example, we use a quantitative benchmark for rising bubble dynamics; see the first benchmark test case in [HTK⁺09]. A simulation is illustrated in Figure 1.

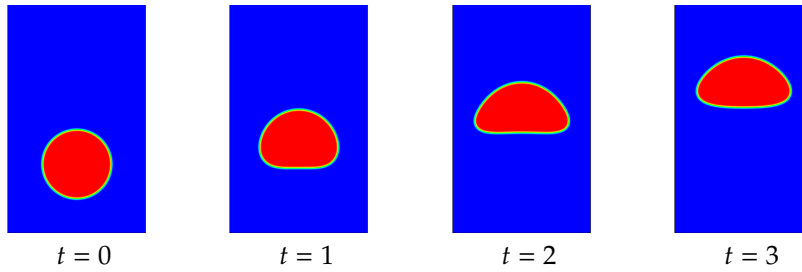


Figure 1: Simulation of a rising bubble using a coupled Cahn–Hilliard Navier–Stokes model.

The initial configuration is described as follows; see also [GHK16, p. 168]. The spatial domain is $\Omega = (0, 1) \times (0, 2)$ with no-slip boundary conditions for the velocity field on the top and bottom wall and free-slip boundary conditions on the left and right wall. The initial state consists of a bubble of radius $r = 0.25$ centered at the spatial point $(0.5, 0.5)$. The initial velocity is zero. In the following, we denote by N_1^0 and N_2^0 the values of N_1 and N_2 at time 0. Note that during the simulation, the values of N_1 and N_2 stay about their initial sizes, i.e., there is no drastic change. The fixed parameters in all tests are given as $\rho_2 = 100$, $\eta_1 = 10$, $\eta_2 = 1$, $g = (0, -0.98)^t$. The remaining parameters are given below for each individual experiment. Moreover, the Reynolds number is given as

$$Re = \frac{0.35 \rho_1}{\eta_1}.$$

Figures 2 and 3 demonstrate the robustness with respect to different model parameters. Table 1 shows the values of all parameters. In Figure 2(a), we simultaneously vary the mesh sizes via refinements of the initial spatial mesh \mathcal{T}^0 , the interfacial parameter ϵ , the time step size τ , as well as the mobility b . In fact, this is the practical procedure: Choose an ϵ and adjust the mesh sizes. The time step size is adapted to fulfill the CFL-condition

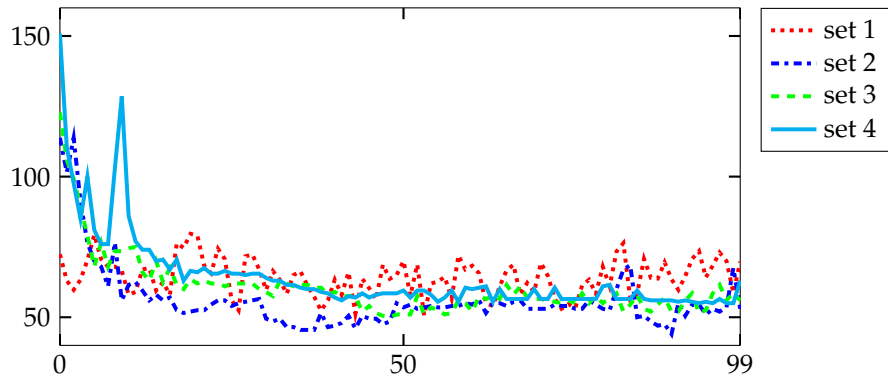
$$\max_T \left\{ \frac{\tau |v^k|_T}{\text{diam}(T)} \right\} \leq 1.$$

Moreover, the mobility is chosen to be $b = 10^{-3}\epsilon$, as used in [GHK16, p. 168]. Except for the initial time frame, we observe a robust behavior of iteration numbers. For the first time step the initial data for Newton's method, which is a discrete approximation of φ_0 , seems not to be appropriate and thus result in a slow convergence. In Figure 2(b), we vary the scaled surface tension σ . Although the iteration numbers behave quite chaotic, they mostly stay in the range between 50 and 65. In Figure 2(c), we vary the Reynolds number via increasing the density ρ_1 . We observe a small increase of iterations numbers as the Reynolds number increases. In Figure 3(a), we vary the mobility b . We observe a benign increase of iterations numbers as the mobility decreases. In Figure 3(b), we vary the penalty parameter s . Comparing the iteration numbers for $s = 10^4$ with the ones for $s \in \{10^6, 10^8, 10^9\}$, we even obtain better results for the larger penalty parameters. Finally, Table 1 illustrates the maximum and average number of semismooth Newton iterations for each of the five subplots, respectively.

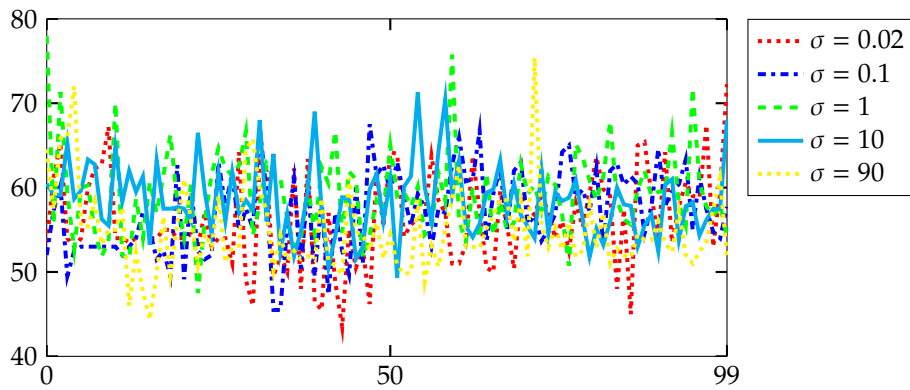
Simulation		Newton		Parameters								
Figure	Plot	Max	Avg	N_1^0	N_2^0	Re	ρ_1	s	σ	ϵ	τ	b
2(a)	(•••••)	6	3	6599	26213	35	1000	10^4	15.60	0.040	$2.000 \cdot 10^{-3}$	$4 \cdot 10^{-5}$
	(•••••)	6	2	10399	41413	35	1000	10^4	15.60	0.020	$5.000 \cdot 10^{-4}$	$2 \cdot 10^{-5}$
	(•••••)	6	2	17831	71141	35	1000	10^4	15.60	0.010	$1.250 \cdot 10^{-4}$	$1 \cdot 10^{-5}$
	(—)	6	2	32527	129925	35	1000	10^4	15.60	0.005	$3.125 \cdot 10^{-5}$	$5 \cdot 10^{-6}$
2(b)	(•••••)	8	4	6599	26213	35	1000	10^6	0.02	0.040	$2.000 \cdot 10^{-3}$	$4 \cdot 10^{-5}$
	(•••••)	9	5	6599	26213	35	1000	10^6	0.10	0.040	$2.000 \cdot 10^{-3}$	$4 \cdot 10^{-5}$
	(•••••)	9	6	6599	26213	35	1000	10^6	1.00	0.040	$2.000 \cdot 10^{-3}$	$4 \cdot 10^{-5}$
	(—)	9	6	6599	26213	35	1000	10^6	10.00	0.040	$2.000 \cdot 10^{-3}$	$4 \cdot 10^{-5}$
	(•••••)	10	6	6599	26213	35	1000	10^6	90.00	0.040	$2.000 \cdot 10^{-3}$	$4 \cdot 10^{-5}$
2(c)	(•••••)	10	6	6599	26213	35	1000	10^6	15.60	0.040	$2.000 \cdot 10^{-3}$	$4 \cdot 10^{-5}$
	(•••••)	10	7	6599	26213	70	2000	10^6	15.60	0.040	$2.000 \cdot 10^{-3}$	$4 \cdot 10^{-5}$
	(•••••)	31	7	6599	26213	140	4000	10^6	15.60	0.040	$2.000 \cdot 10^{-3}$	$4 \cdot 10^{-5}$
	(—)	44	8	6599	26213	280	8000	10^6	15.60	0.040	$2.000 \cdot 10^{-3}$	$4 \cdot 10^{-5}$
	(•••••)	10	7	6599	26213	560	16000	10^6	15.60	0.040	$2.000 \cdot 10^{-3}$	$4 \cdot 10^{-5}$
3(a)	(•••••)	12	7	6599	26213	35	1000	10^6	15.60	0.040	$2.000 \cdot 10^{-3}$	$7 \cdot 10^{-5}$
	(•••••)	10	6	6599	26213	35	1000	10^6	15.60	0.040	$2.000 \cdot 10^{-3}$	$4 \cdot 10^{-5}$
	(•••••)	10	7	6599	26213	35	1000	10^6	15.60	0.040	$2.000 \cdot 10^{-3}$	$1 \cdot 10^{-4}$
	(—)	11	7	6599	26213	35	1000	10^6	15.60	0.040	$2.000 \cdot 10^{-3}$	$3 \cdot 10^{-4}$
3(b)	(•••••)	6	3	6599	26213	35	1000	10^4	15.60	0.040	$2.000 \cdot 10^{-3}$	$4 \cdot 10^{-5}$
	(•••••)	10	6	6599	26213	35	1000	10^6	15.60	0.040	$2.000 \cdot 10^{-3}$	$4 \cdot 10^{-5}$
	(•••••)	18	8	6599	26213	35	1000	10^8	15.60	0.040	$2.000 \cdot 10^{-3}$	$4 \cdot 10^{-5}$
	(—)	23	8	6599	26213	35	1000	10^9	15.60	0.040	$2.000 \cdot 10^{-3}$	$4 \cdot 10^{-5}$

Table 1: The maximum and average number of semismooth Newton iterations for each parameter test.

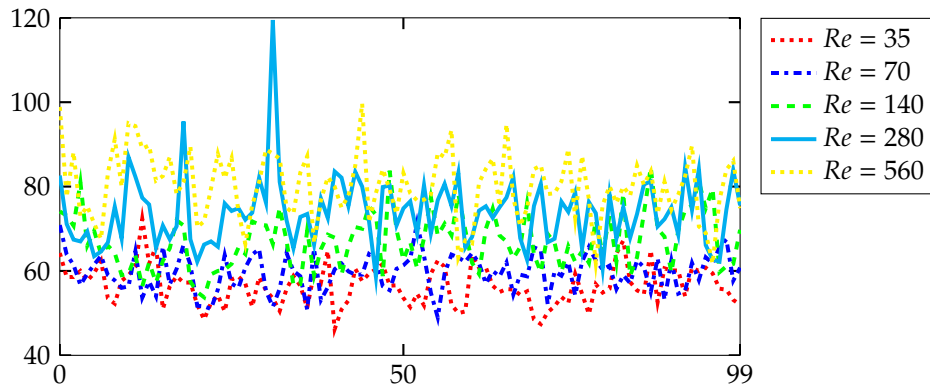
In summary, we observe a rather robust behavior of the iterative method with respect to changes of the parameters of the system. Especially, the method is robust with respect to the parameter ϵ , that heavily influences the number of degrees of freedom and therefore the size of the linear system, and the



(a) Simultaneous variation of the initial spatial mesh \mathcal{T}^0 , the interfacial parameter ϵ , the time step size τ , as well as the mobility b .

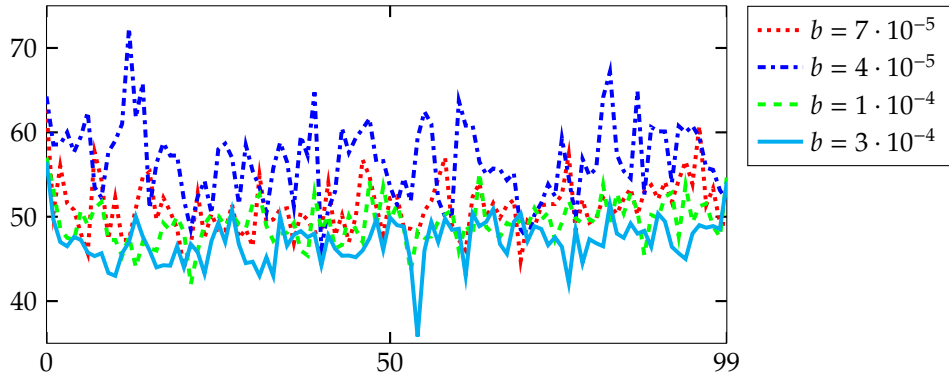


(b) Variation of the scaled surface tension σ .

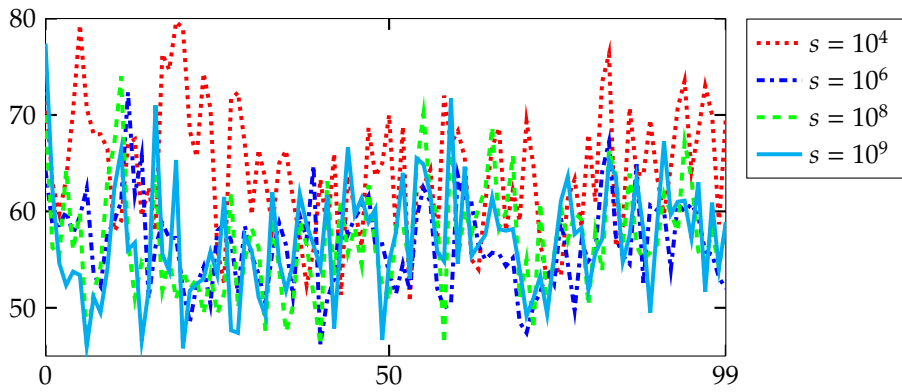


(c) Variation of the Reynolds number via increasing the density ρ_1 .

Figure 2: Iteration numbers for the parameter study. The x-axis shows the time step and the y-axis the average number of FGMRES iterations per semismooth Newton step.



(a) Variation of the mobility b .



(b) Variation of the penalty parameter s .

Figure 3: Iteration numbers for the parameter study. The x-axis shows the time step and the y-axis the average number of FGMRES iterations per semismooth Newton step.

parameter s for which we typically observe a severe increase of the condition number.

4.2 A topology change

In this section, we demonstrate the behavior of our preconditioner under topology changes. As test example, we use a quantitative benchmark for rising bubble dynamics; see the second benchmark in [HTK⁺09]. It considers a bubble with a very low density compared to that of the surrounding fluid. The initial configuration is the same as in the previous benchmark example. The parameters are given as $\rho_1 = 1000$, $\rho_2 = 1$, $\eta_1 = 10$, $\eta_2 = 0.1$, $Re = 35$, $N_1^0 = 4513$, $N_2^0 = 17929$, $\epsilon = 0.04$, $\tau = 0.002$, $b = 4 \cdot 10^{-5}$, $\sigma = 1.24777$, $s = 10^6$. As stated in [AV12, p. 756], the decrease in surface tension causes the bubble to develop a more non-convex shape and thin filaments, which eventually break off. A simulation of this benchmark test example is illustrated in Figure 4.

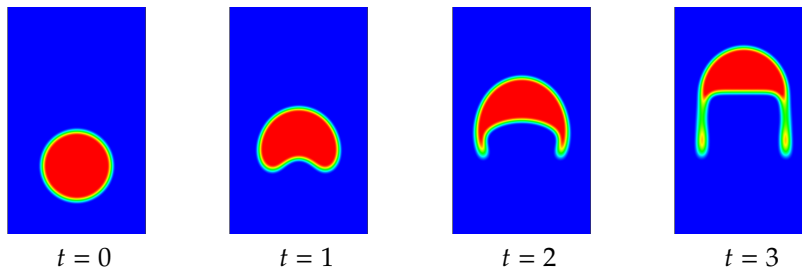


Figure 4: Simulation of a rising bubble under topology changes using a coupled Cahn–Hilliard Navier–Stokes model.

In this example, we set the tolerance on the relative residual norm for S_1 and S_2 to 10^{-6} . Figure 5 demonstrates the average number of FGMRES iterations per semismooth Newton step during the time interval $[0, 0.56]$, which consists of 280 time steps. Since the crucial part is the period during the topology change, we show the corresponding iteration numbers in Table 2. Note that for the results in Table 2, we do not call MATLAB[®] from C++. Instead, we use Garcke et al’s [GHK16] original C++ implementation, write the matrices and right-hand sides into text files, and solve the problems in MATLAB using our developed preconditioner. There are two reasons for this procedure: First, it takes a lot of time to get to the crucial instant of time using C++ with the MATLAB Engine API. Second, we observe abnormal terminations after long program runs. Hence, an important step for future research is an implementation of our preconditioner in C++ such that the MATLAB Engine API would be no longer required. Further, we set the GMRES (inner solver) tolerance for the preconditioned relative residual to 10^{-2} . Table 2 shows that the FGMRES iteration numbers stay quite robust under topology changes. As topology changes are only implicitly defined by a phase field approach, this is what we expected.

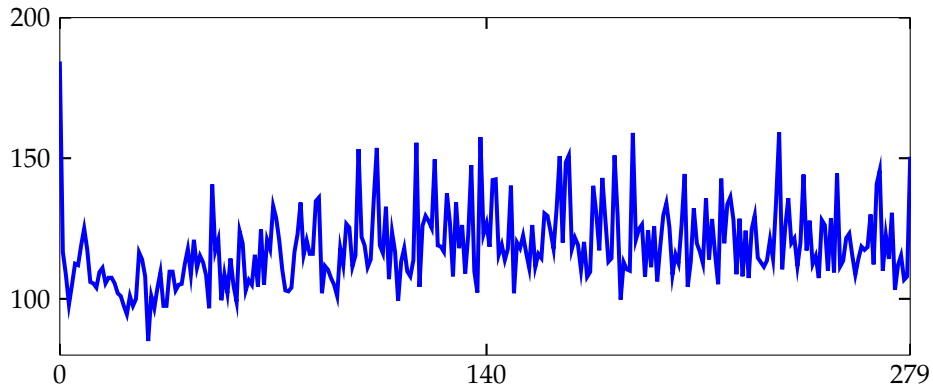


Figure 5: Iteration numbers for the second benchmark example. The x-axis shows the time step and the y-axis the average number of FGMRES iterations per semismooth Newton step.

t	1.9990	1.9995	2.0000	2.0005	2.0010	2.9985	2.9990	2.9995	3.0000
Iter	179	167	127	136	144	173	202	152	151

Table 2: Iteration numbers for the rising bubble under topology changes. The table shows the time with the average number of FGMRES iterations per semismooth Newton step.

5 Conclusions

In this paper, we have investigated the efficient iterative solution of linear systems that arise in the numerical simulation of a coupled Navier–Stokes Cahn–Hilliard system, like the model ‘H’ from [HH77] or its generalization to fluids of different densities in [AGG12]. As an example we have used the energetically consistent discretization scheme proposed in [GHK16].

At the heart of this method lies the solution of large and sparse linear systems that arise in a semismooth Newton method. The systems to be solved are fully coupled. We have introduced and studied block-triangular preconditioners using efficient Schur complement approximations. For these approximations, we have used multilevel techniques, algebraic multigrid in our case. Extensive numerical experiments show a nearly parameter independent behavior of our developed preconditioners. Together with Garcke et al’s [GHK16] adaptive spatial discretization scheme, this allows us to perform three-dimensional experiments in an efficient way. This will be subject to future work.

Acknowledgments

The authors would like to thank the anonymous referees for their helpful comments and suggestions.

References

- [AB16] H. Abels and D. Breit. Weak Solutions for a Non-Newtonian Diffuse Interface Model with Different Densities. *Nonlinearity*, 29:3426–3453, 2016.
- [ABB⁺99] E. Anderson, Z. Bai, C. Bischof, L. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen. *LAPACK Users’ Guide*. SIAM, 3rd edition, 1999.
- [ADG13a] H. Abels, D. Depner, and H. Garcke. Existence of weak solutions for a diffuse interface model for two-phase flows of incompressible fluids with different densities. *J. Math. Fluid Mech.*, 15(3):453–480, September 2013.
- [ADG13b] H. Abels, D. Depner, and H. Garcke. On an incompressible Navier–Stokes / Cahn–Hilliard system with degenerate mobility. *Ann. Inst. H. Poincaré (C) Non Linear Anal.*, 30(6):1175–1190, 2013.
- [AGG12] H. Abels, H. Garcke, and G. Grün. Thermodynamically consistent, frame indifferent diffuse interface models for incompressible two-phase flows with different densities. *Math. Models Methods Appl. Sci.*, 22(3):40, March 2012.

- [Ala14] S. Aland. Time integration for diffuse interface models for two-phase flow. *J. Comput. Phys.*, 262:58–71, April 2014.
- [AN11] O. Axelsson and M. Neytcheva. Operator splittings for solving nonlinear, coupled multiphysics problems with an application to the numerical solution of an interface problem. Technical Report 2011-009, Department of Information Technology, Uppsala University, 2011.
- [AV12] S. Aland and A. Voigt. Benchmark computations of diffuse interface models for two-dimensional bubble dynamics. *Internat. J. Numer. Methods Fluids*, 69(3):747–761, 2012.
- [BDQN11] P. Boyanova, M. Do-Quang, and M. Neytcheva. Solution methods for the Cahn–Hilliard equation discretized by conforming and non-conforming finite elements. Technical Report 2011-004, Department of Information Technology, Uppsala University, 2011.
- [BE91] J. F. Blowey and C. M. Elliott. The Cahn–Hilliard gradient theory for phase separation with non-smooth free energy. Part I: Mathematical analysis. *European J. Appl. Math.*, 2:233–280, 1991.
- [BKSW14] J. Bosch, D. Kay, M. Stoll, and A. J. Wathen. Fast solvers for Cahn–Hilliard inpainting. *SIAM J. Imaging Sci.*, 7(1):67–97, 2014.
- [BMS10] J. Boyle, M. Mihajlović, and J. Scott. HSL_MI20: An efficient AMG preconditioner for finite element problems in 3D. *Internat. J. Numer. Methods Engrg.*, 82(1):64–98, 2010.
- [Bos16] J. Bosch. *Fast iterative solvers for Cahn–Hilliard problems*. PhD thesis, Otto-von-Guericke-Universität Magdeburg, 2016.
- [Boy02] F. Boyer. A theoretical and numerical model for the study of incompressible mixture flows. *Computers & Fluids*, 31(1):41–68, January 2002.
- [BS15] J. Bosch and M. Stoll. Preconditioning for vector-valued Cahn–Hilliard equations. *SIAM J. Sci. Comput.*, 37(5):S216–S243, 2015.
- [BSB14] J. Bosch, M. Stoll, and P. Benner. Fast solution of Cahn–Hilliard variational inequalities using implicit time discretization and finite elements. *J. Comput. Phys.*, 262:38–57, 2014.
- [CH58] J. W. Cahn and J. E. Hilliard. Free energy of a nonuniform system. I. Interfacial free energy. *J. Chem. Phys.*, 28(2):258–267, 1958.
- [Dav16] T. A. Davis. UMFPACK User Guide. Technical Report TR-04-003 (revised), Department of Computer Science and Engineering, Texas A&M University, 2016.

- [EES83] S. C. Eisenstat, H. C. Elman, and M. H. Schultz. Variational iterative methods for nonsymmetric systems of linear equations. *SIAM J. Numer. Anal.*, 20(2):345–357, 1983.
- [EJW05] H. C. Elman, D. J. Silvester A. J., and Wathen. *Finite Elements and Fast Iterative Solvers: With Applications in Incompressible Fluid Dynamics*. Numer. Math. Sci. Comput. Oxford Univ. Press, Oxford, 2005.
- [Fal06] R. D. Falgout. An introduction to algebraic multigrid computing. *Comput. Sci. Eng.*, 8(6):24–33, 2006.
- [GGM16] G. Grün, F. Guillén-González, and S. Metzger. On Fully Decoupled Convergent Schemes for Diffuse Interface Models for Two-Phase Flow with General Mass Densities. *Communications in Computational Physics*, 19(5):1473–1502, May 2016.
- [GHK16] H. Garcke, M. Hinze, and C. Kahle. A stable and linear time discretization for a thermodynamically consistent model for two-phase incompressible flow. *Appl. Numer. Math.*, 99:151–171, January 2016.
- [GK14] G. Grün and F. Klingbeil. Two-phase flow with mass density contrast: Stable schemes for a thermodynamic consistent and frame indifferent diffuse interface model. *J. Comput. Phys.*, 257(A):708–725, January 2014.
- [Gre97] A. Greenbaum. *Iterative Methods for Solving Linear Systems*, volume 17 of *Frontiers Appl. Math.* SIAM, Philadelphia, PA, 1997.
- [Grü13] G. Grün. On convergent schemes for diffuse interface models for two-phase flow of incompressible fluids with general mass densities. *SIAM J. Numer. Anal.*, 51(6):3036–3061, 2013.
- [GT14] F. Guillén-González and G. Tierra. Splitting schemes for a Navier–Stokes–Cahn–Hilliard model for two fluids with different densities. *Journal of Computational Mathematics*, 32(6):643–664, 2014.
- [GV61a] G. H. Golub and R. S. Varga. Chebyshev semi-iterative methods, successive overrelaxation iterative methods, and second order Richardson iterative methods. I. *Numer. Math.*, 3:147–156, 1961.
- [GV61b] G. H. Golub and R. S. Varga. Chebyshev semi-iterative methods, successive overrelaxation iterative methods, and second order Richardson iterative methods. II. *Numer. Math.*, 3:157–168, 1961.
- [Hac85] W. Hackbusch. *Multi-Grid Methods and Applications*, volume 4 of *Springer Ser. Comput. Math.* Springer, Berlin Heidelberg, 1985.
- [HH77] P. C. Hohenberg and B. I. Halperin. Theory of dynamic critical phenomena. *Reviews of Modern Physics*, 49(3):435–479, 1977.

- [HHT11] M. Hintermüller, M. Hinze, and M. H. Tber. An adaptive finite element Moreau–Yosida-based solver for a non-smooth Cahn–Hilliard problem. *Optim. Methods Softw.*, 25(4-5):777–811, 2011.
- [HIK03] M. Hintermüller, K. Ito, and K. Kunisch. The primal-dual active set strategy as a semi-smooth Newton method. *SIAM Journal on Optimization*, 13(3):865–888, 2003.
- [HKW15] M. Hintermüller, T. Keil, and D. Wegner. Optimal Control of a Semidiscrete Cahn–Hilliard–Navier–Stokes System with Non-Matched Fluid Densities. *arXiv: 1506.03591*, 2015.
- [HS52] Magnus R. Hestenes and Eduard Stiefel. Methods of conjugate gradients for solving linear systems. *J. Research Nat. Bur. Standards*, 49:409–436 (1953), 1952.
- [HTK⁺09] S. Hysing, S. Turek, D. Kuzmin, N. Parolini, E. Burman, S. Ganesan, and L. Tobiska. Quantitative benchmark computations of two-dimensional bubble dynamics. *Internat. J. Numer. Methods Fluids*, 60(11):1259–1288, 2009.
- [Kah14] C. Kahle. *Simulation and Control of Two-Phase Flow Using Diffuse Interface Models*. PhD thesis, Universität Hamburg, 2014.
- [KLW02] D. Kay, D. Loghin, and A. J. Wathen. A preconditioner for the steady-state Navier–Stokes equations. *SIAM J. Sci. Comput.*, 24(1):237–256, 2002.
- [LT98] J. Lowengrub and L. Truskinovsky. Quasi-incompressible Cahn–Hilliard fluids and topological transitions. *Proceedings of the royal society A*, 454(1978):2617–2654, 1998.
- [MGW00] M. F. Murphy, G. H. Golub, and A. J. Wathen. A note on preconditioning for indefinite linear systems. *SIAM J. Sci. Comput.*, 21(6):1969–1972, 2000.
- [NN12] A. Napov and Y. Notay. An algebraic multigrid method with guaranteed convergence rate. *SIAM J. Sci. Comput.*, 34(2):A1079–A1109, 2012.
- [Not10] Y. Notay. An aggregation-based algebraic multigrid method. *Electron. Trans. Numer. Anal.*, 37:123–146, 2010.
- [Not12] Y. Notay. Aggregation-based algebraic multigrid for convection-diffusion equations. *SIAM J. Sci. Comput.*, 34(4):A2288–A2316, 2012.
- [PS75] C. C. Paige and M. A. Saunders. Solutions of sparse indefinite systems of linear equations. *SIAM J. Numer. Anal.*, 12(4):617–629, 1975.

- [PSW12] John W. Pearson, Martin Stoll, and Andrew J. Wathen. Regularization-robust preconditioners for time-dependent PDE-constrained optimization problems. *SIAM J. Matrix Anal. Appl.*, 33(4):1126–1152, 2012.
- [PW12] J. W. Pearson and A. J. Wathen. A new approximation of the Schur complement in preconditioners for PDE-constrained optimization. *Numer. Linear Algebra Appl.*, 19(5):816–829, 2012.
- [Ram99] A. Ramage. A multigrid preconditioner for stabilised discretisations of advection-diffusion problems. *J. Comput. Appl. Math.*, 110(1):187–203, 1999.
- [RS87] J. W. Ruge and K. Stüben. Algebraic multigrid. In *Multigrid methods*, volume 3 of *Frontiers Appl. Math.*, pages 73–130. SIAM, Philadelphia, PA, 1987.
- [RS10] T. Rees and M. Stoll. Block-triangular preconditioners for PDE-constrained optimization. *Numer. Linear Algebra Appl.*, 17(6):977–996, 2010.
- [Saa93] Y. Saad. A flexible inner-outer preconditioned GMRES algorithm. *SIAM J. Sci. Comput.*, 14(2):461–469, 1993.
- [Saa03] Y. Saad. *Iterative methods for sparse linear systems*. SIAM, Philadelphia, PA, 2nd edition, 2003.
- [SS86a] Y. Saad and M. H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. and Stat. Comput.*, 7(3):856–869, 1986.
- [SS86b] Youcef Saad and Martin H. Schultz. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Statist. Comput.*, 7(3):856–869, 1986.
- [SY10] J. Shen and X. Yang. A Phase-Field Model and its Numerical Approximation for Two-Phase Incompressible Flows with Different Densities and Viscosities. *SIAM Journal on Scientific Computing*, 32(3):1159–1179, 2010.
- [Wes92] P. Wesseling. *An Introduction to Multigrid Methods*. Pure Appl. Math. (New York). Wiley, Ltd., Chichester, 1992.
- [WR09] A. J. Wathen and T. Rees. Chebyshev semi-iteration in preconditioning for problems including the mass matrix. *Electron. Trans. Numer. Anal.*, 34:125–135, 2009.