

# Efficient preconditioners for large scale binary Cahn-Hilliard models

Petia Boyanova<sup>1,2</sup>, Minh Do-Quang<sup>3</sup>, Maya Neytcheva<sup>1</sup>

<sup>1</sup>Uppsala University, Box 337, 751 05 Uppsala, Sweden

<sup>2</sup>Institute for Information and Communication Technology - BAS,  
Acad. G. Bonchev Str., bl. 25A, 1113 Sofia, Bulgaria

<sup>3</sup>Department of Mechanics, Linné Flow Centre,  
Royal Institute of Technology, SE-100 44 Stockholm, Sweden

## Abstract

In this work we consider preconditioned iterative solution methods for numerical simulations of multiphase flow problems, modelled by the Cahn-Hilliard equation. We focus on diphasic flows and the construction and efficiency of a preconditioner for the algebraic systems arising from finite element discretizations in space and the  $\theta$ -method in time. The preconditioner utilizes to a full extent the algebraic structure of the underlying matrices and exhibits optimal convergence and computational complexity properties. Large scale numerical experiments are included as well as performance comparisons with other solution methods.

**Keywords** Cahn-Hilliard · finite elements · two-by-two block matrices · preconditioning

## 1 Introduction

The diffuse-interface phase-field approach has emerged as a powerful mathematical model for simulating the creation and evolution of various morphological patterns and interface motion. A brief account of the phenomena, treated using this model contains

- coarsening kinetics of two- or three-phase microstructures such as binary or multicomponent alloys, polymer systems, crystal growth and spinodal decomposition (cf. [36]);
- capillary phenomena, wetting (cf. [14])
- evolution of two components of intergalactic materials, the dynamics of two populations;
- modelling the dynamics of the biomass and the solvent components of a bacterial or other thin films (cf. [34];
- phase separation in presence of elastic interactions, see e.g [22];
- phase transition with memory, for instance, delayed response of a system to thermal gradients [37];
- inpainting the binary images (cf. [8]);

- river bed formation (cf. [34]);
- galaxy formation (cf. [34]), etc.

A relevant question is what are the features of the diffuse-interface phase-field model, which make it so broadly applicable. The main reason is that the model is based on the so-called *total free energy* of the physical system, which can be adapted to the particular system under consideration. In this way, based on one and the same theoretical fundament, we obtain the corresponding equations for a large variety of models, spanning from micro to macro scales.

As a consequence of the complexity and mutual coupling of the modelled physical phenomena, the arising numerical models are, in general, of very large size, may have complex geometries and require adaptive discretization methods to accurately track the dynamics of interface movement. This, in its turn, poses extra demands on the discretization techniques as well as on the numerical solution techniques with respect to robustness, fast convergence, efficient usage of computer resources and parallelization aspects.

To meet the latter requirements, we use as a discretization technique the Finite Element method (FEM), exploiting its flexibility with respect to handling both complex geometries and adaptively refined meshes.

The general applicability, high robustness and the available efficient implementations of sparse direct solution methods, makes these to be the preferred choice in many reported numerical simulations of complex coupled phenomena. However, when very large scale, in space and/or in time, simulations have to be performed, these may require several days or even weeks when direct solvers are used. Therefore, due to their lesser demands for computer resources, iterative solution methods become a necessity. As is well known, the utilization of iterative methods brings concurrently the additional task to construct and apply computationally efficient and robust preconditioners. In the research literature, dealing with the same physical problem, most often the numerical solution method is not even mentioned. In this work, where iterative techniques are allied, it appears, that most often a nonlinear geometric multigrid is used, applied to the whole system of CH equations, using regular refinements and a hierarchy of meshes (see, e.g. [28, 41, 13] for two-phase flow problems). In some studies, ILU-preconditioning techniques are used (cf. [42], for example). In almost all papers the authors came across, no performance (timing) information has been revealed. One exception is [41], however, no comparison with any other solution method is given in order to judge the efficiency of the solution procedure.

We consider here the task to construct preconditioners for the discrete systems of equations, arising in numerical simulations of multiphase problems, modelled by the Cahn-Hilliard equation, with emphasize to binary flows. All aspects of the preconditioners are considered - ease of construction, numerical efficiency, robustness with respect to problem, discretization and method parameters, implementation and suitability for HPC platforms. The preconditioner, described here, satisfies all of the above criteria. It does not require heavy construction phase and unnecessary hierarchy of meshes; for constant meshes it does not have to be recomputed; it allows for relatively large timesteps (of order of the space discretization parameter) without degrading the convergence rate; all its ingredients are standard matrix blocks and general parallelization techniques, as well as adaptive mesh refinement techniques are straightforwardly applicable; it could make use of any of-the-shelf generally applicable Algebraic Multigrid type of method, applied to

a matrix of size, twice smaller than the original system matrix. We also compare the suggested technique to other methods and include timing results for comparison.

This paper is organized as follows. Section 2 briefly describes the mathematical model used to simulate the physical processes, related to phase separation and interface tracking, as well as some particular formulations and properties. In Section 3 we present the space and time discretization schemes, outline the treatment of the nonlinearity using Newton's method and the structure of the matrices in the so-arising linear systems. In Section 4 we discuss suitable preconditioners for solving the discrete Cahn-Hilliard equation and analyze their convergence properties, computational complexity and suitability for parallel implementation. Section 5 contains numerical experiments. Conclusions and some open problems are found in the final Section 6.

## 2 The Cahn-Hilliard equation as a mathematical model for multiphase flow

The brief description of the model, presented below, is based on [33] and [39]. The principal concept is outlined in the case of binary fluids, i.e., fluids with two phases, each of them being incompressible, viscous and isothermal.

Consider the task to numerically simulate the evolution of the interface between two phases in time, subject to various physical processes, such as diffusion, convection, etc. To mathematically describe these processes, we use the so-called phase-field model. In contrast to other lattice-based approaches, in the phase-field framework, the interface is modelled by a function  $C(\mathbf{x}, t)$ , which represents the concentration of the fluids. The function  $C(\mathbf{x}, t)$ , also referred to as the *phase-field*, or the *order parameter*, attains a distinct constant value in each bulk phase and rapidly, but smoothly, changes in the interface region between the phases. For a binary fluid, a usual assumption is that  $C$  takes values between  $-1$  and  $1$ , or  $0$  and  $1$ . The form of the function  $C$  can be generalized for multifluids, for example, for a ternary fluid one can assume that  $\sum C_i(\mathbf{x}, t) = 1, i = 1, 2, 3$  (see, e.g., [17]).

The model goes back to a pioneering work by van der Waals in 1893 ([38]), and is based on classical thermodynamics arguments, developed by Gibbs in 1873 ([23]). The interface profiles are the minimizers of the so-called *free energy functional*

$$E(C) = \int_{\Omega} f(C) d\Omega \quad (1)$$

for some spatial domain  $\Omega \in \mathbb{R}^d, d = 1, 2, 3$  and  $f$  is the free energy density.

Depending on the particular physical phenomena, the free energy density function may have different forms. In the test problems, considered here, it has the form

$$f(C) = \frac{1}{2}\alpha|\nabla C|^2 + \beta\Psi(C), \quad (2)$$

where  $\alpha$  and  $\beta$  are some coefficients to be described below. The two terms in (2) are referred to as the *gradient energy*  $\frac{1}{2}\alpha|\nabla C|^2$  and the *bulk energy*, or *molar Gibbs energy*,  $\beta\Psi(C)$ . The

term  $|\nabla C|^2$  can be related to intermolecular interactions and can be viewed as penalizing the creation of interfaces (cf. i.e., [20]). The function  $\Psi(C)$  is a double-well potential with two stable minima. If we assume that for the two phases  $C$  varies between  $-1$  and  $1$ , these minima are at  $\pm 1$ . In the current study we consider  $\Psi(C) = \frac{1}{4}(C^2 - 1)^2$ .

Define next  $\Phi$  to be the rate of change of  $E$  with respect to  $C$ ,

$$\Phi = \frac{\partial E}{\partial C} = \int_{\Omega} [-\alpha \Delta C + \beta \Psi'(C)] d\Omega. \quad (3)$$

Then, the equilibrium profiles are the solutions of the equation

$$\eta \equiv -\alpha \Delta C + \beta \Psi'(C) = \text{const}, \quad (4)$$

where  $\eta$  is the so-called *chemical potential* of the liquid mixture.

Equation (4), together with the two stable constant solutions  $C = \pm 1$  (recall that  $\Psi'(C) = C(C^2 - 1)$ ), turns out to have a one-dimensional nonuniform solution (first found by van der Waals) in the form

$$\hat{C}(\mathbf{x}) = \tanh \left( \frac{x}{\sqrt{2}\xi} \right),$$

where  $\xi = \sqrt{\frac{\alpha}{\beta}}$  is the mean-field thickness. With the help of (4) one can find the *equilibrium surface tension*  $\sigma$ , which has the following form,

$$\sigma = \alpha \int_{-\infty}^{\infty} \left( \frac{d\hat{C}}{dx} \right)^2 dx = \frac{2\sqrt{2}}{3} \sqrt{\alpha\beta}.$$

Thus, via the choice of the parameters  $\alpha$  and  $\beta$ , one can control  $\sigma$ , as well as the so-called *equilibrium interface thickness*  $\epsilon$ , defined as the distance between  $\mathbf{x}_1$  and  $\mathbf{x}_2$ ,  $C(\mathbf{x}_1) = -0.9$  and  $C(\mathbf{x}_2) = 0.9$  (cf. e.g., [26]).

Later, in [12, 11], the following equation, referred to as the Cahn-Hilliard (C-H) equation, has been derived. It is considered to capture the dominant paradigms, which describe phase separation dynamics.

The Cahn-Hilliard equation for two-phase flow reads as follows:

$$\frac{\partial C}{\partial t} = \nabla \cdot [\kappa(C) \nabla (\Psi'(C) - \epsilon^2 \Delta C)], \quad (\mathbf{x}, t) \in \Omega \times \mathbb{R}^+ \quad (5)$$

$$\mathbf{n} \cdot \nabla C = 0 \quad \text{on } \partial\Omega, \quad (6)$$

$$\mathbf{n} \cdot [\kappa(C) \nabla (\Psi'(C) - \epsilon^2 \Delta C)] = 0 \quad \text{on } \partial\Omega, \quad (7)$$

$$C(\mathbf{x}, 0) = C_0(\mathbf{x}), \quad (8)$$

where  $\mathbf{n}$  is the outer normal to  $\partial\Omega$ . Here,  $\kappa(C)$  is the so-called *mobility*, which is assumed to be dependent on  $C$ . The relations (6) (the variational condition), (7) (the no-flux condition) and the initial condition (8), added to the C-H equation (5), ensure the well-posedness of the problem. To relate (5) to (4) and see the relationship between the parameters  $\epsilon, \alpha, \beta$ , it suffices to assume that  $\epsilon = \sqrt{\frac{\alpha}{\beta}}$  with  $\beta = 1$ .



### Remark 2.1

- Some available results regarding existence and uniqueness of the solution of the C-H equation, can be found in [32, 33]. A proof of the existence and uniqueness for the constant mobility case with a free energy of a polynomial form is given in [16]. Under certain conditions, some results for the degenerate mobility case in one dimension are shown in [27, 33].
- We note that (5) is a fourth order parabolic equation. However, in general, the maximum principle does not hold (see, e.g., [30]).
- Due to the assumption that the processes are isothermic, Equation (5) may seem rather simple to describe complex physics phenomena. It does not depend explicitly on temperature, thus, thermal effects are not included. Further, elastic, viscoelastic effects, as well as anisotropies are also not accounted for. As is described in [32], however, the Cahn-Hilliard equation is the limit case for much more complex models and therefore, understanding its dynamics and being able to perform efficient numerical simulations with it, creates the building blocks for tackling those more complicated and coupled models.

In practice, adjustments are done, in order to take into account other physical processes. Consider as an example temperature changes, when we want to follow the evolution and pattern formation in a binary liquid under cooling. A typical scenario is to assume that the temperature of the system rapidly decreases and the system instantaneously equilibrates to that. The later equilibration is modelled by coupling C-H with an energy balance equation (cf. [33] and the references therein).

Another example is to include anisotropies in the phases. A non-isotropic model with anisotropic surface energy is derived in [1] (see also the references therein).

Consider now some versions of the C-H equation.

- (I) A simplified version of (5) is the so-called *constant mobility* form of C-H,

$$\frac{\partial C}{\partial t} = \kappa \Delta (\Psi'(C) - \epsilon^2 \Delta C). \quad (9)$$

As mentioned in [33], in this case the solution needs not remain bounded between  $-1$  and  $1$ , even if the initial condition  $C_0(\mathbf{x})$  is.

- (II) As a cure for the latter, the so-called *degenerate mobility* formulation is introduced, in which

$$\kappa(C) = C(1 - C),$$

i.e. it may degenerate at  $C = 0$  and  $C = 1$ . Note that in the formulation used then,  $C$  has a meaning only in  $[0,1]$ , since  $\Psi(C) = \frac{T}{2} \{C \ln C + (1 - C) \ln(1 - C)\} + \tau C(1 - C)$ ,

where  $T > 0$  denotes a scaled temperature. The resulting formulation reads as

$$\begin{aligned}\frac{\partial C}{\partial t} &= \frac{T}{2}\Delta C - \nabla \cdot \kappa(C)\nabla (2\tau C + \epsilon^2\Delta C), \quad (\mathbf{x}, t) \in \Omega \times \mathbb{R}^+ \\ \mathbf{n} \cdot \nabla C &= \mathbf{n} \cdot \kappa(C)\nabla \Delta C = 0, \quad (\mathbf{x}, t) \in \partial\Omega.\end{aligned}\tag{10}$$

See [33] for more details and references.

(III) It is also relevant to note that C-H can be written in the form of a gradient flow

$$\frac{\partial C}{\partial t} = -\nabla \cdot G,\tag{11}$$

where  $G = \kappa(C)\nabla(\Psi'(C) - \epsilon^2\Delta C)$ . The latter formulation shows that C-H is fully mass-conservative. The form (11) has led to numerical schemes to solve the C-H equation using methods for general gradient flows (see [18] and [19]).

(IV) Another formulation of the model, important for the test problems and the target applications in this work is the *convective C-H equation*, where the interface develops due to both diffusion and convection

$$\frac{\partial C}{\partial t} + (\mathbf{u} \cdot \nabla)C = \nabla \cdot [\kappa(C)\nabla(\Psi'(C) - \epsilon^2\Delta C)].\tag{12}$$

Above,  $\mathbf{u}$  is the velocity vector, obtained as a solution of the time-dependent Navier-Stokes (N-S) equation,

$$\rho \frac{\partial \mathbf{u}}{\partial t} + (\rho \mathbf{u} \cdot \nabla)\mathbf{u} = -\nabla p + \nabla \cdot [\mu(\nabla)\mathbf{u} + \nabla \mathbf{u}^T] - \eta \nabla C + F,\tag{13}$$

equipped with appropriate boundary conditions. Here  $p$  is the pressure,  $\rho$  and  $\mu$  are the density and viscosity, correspondingly, and  $F$  is the force term. The term  $\eta \nabla C$ , where  $\eta = \Psi'(C) - \epsilon^2\Delta C$ , gives the coupling with (12) and represents the surface tension force in a potential form (cf. [14]). Note, that in the context of the coupled C-H–N-S system,  $\rho$  and  $\mu$  vary with time and space.

### 3 Space and time discretization of the Cahn-Hilliard equation

Here, we consider only two space dimensions and constant mobility. The C-H formulations, which are studied numerically in this paper, are the following.

(1) Evolution of flow interface and material transport due to both diffusion and hydrodynamic flow:

$$\frac{\partial C}{\partial t} + (\mathbf{u} \cdot \nabla)C = \frac{1}{Pe}\Delta (\Psi'(C) - Cn^2\Delta C),\tag{14}$$

where  $\mathbf{u}$  is the solution of the coupled time-dependent nonlinear Navier-Stokes equation for an incompressible fluid,

$$\begin{aligned} Re \left( \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) &= -\nabla p + \Delta \mathbf{u} - \frac{1}{Ca \cdot Cn} C \nabla \eta \\ \nabla \cdot \mathbf{u} &= 0, \end{aligned} \quad (15)$$

where  $\eta = \Psi'(C) - Cn^2 \Delta C$ .

Formulation (14) serves as a model in multiphase simulations, where the interface thickness ( $Cn$ ) and the diffusion speed ( $Pe$ ) play an important role.

- (2) Phase separation with coarsening of a binary mixture due to interface diffusion, i.e., the case of absence of convection ( $\mathbf{u} \equiv \mathbf{0}$  in (14)). This model is often used to simulate processes, for instance, in metallurgy. The formulation in this case reads as follows:

$$\frac{\partial C}{\partial t} - \Delta(\Psi'(C) - \epsilon^2 \Delta C) = 0, \quad (16)$$

Without going into details, we note that in this case  $Pe = 1$  is a usual assumption. In the corresponding dimensionless formulation,  $Cn$  and  $\epsilon$  are not exactly the same physical quantities. Mathematically, however, both parameters are small positive constants.

(For brevity, the details about the domain of definition and the boundary conditions are excluded from the above problem formulations.)

Equations (14), (15) and (16) are in a nondimensionalized form. The parameters  $Pe$ ,  $Cn$ ,  $Re$ ,  $Ca$  are the Peclet, Cahn, Reynolds, and Capillary numbers, respectively. Details on the nondimensionalization are given in [9]. In case (1), we assume that the coupled system (14)–(15) is solved using operator splitting, i.e., at each time step the velocity  $\mathbf{u}$  in equation (14) has already been found as a solution of (15).

Consider the more general form of C-H, (14), which, as already stated, is a fourth order nonlinear parabolic equation. In what follows, for notational simplicity, we use  $\epsilon$  to denote  $Cn$  and  $\omega$  to denote  $1/Pe$ . We choose to deal with the equation by decomposing it into a coupled system of two second-order partial differential equations (PDE) (see e.g. [15]). We consider  $\eta$  as an unknown function and obtain the following system for both the concentration  $C$  and the chemical potential  $\eta$ .

$$\begin{aligned} \eta - \Psi'(C) + \epsilon^2 \Delta C &= 0, & (\mathbf{x}, t) \in \Omega_T \equiv \Omega \times (0, T), \Omega \subset \mathbb{R}^2 \\ -\omega \Delta \eta + \frac{\partial C}{\partial t} + (\mathbf{u} \cdot \nabla) C &= 0, & (\mathbf{x}, t) \in \Omega_T \\ \frac{\partial C}{\partial \mathbf{n}} &= 0, \quad \frac{\partial \eta}{\partial \mathbf{n}} = 0, & \mathbf{x} \in \partial \Omega \\ C(x, 0) &= C_0(x), \end{aligned} \quad (17)$$

where  $0 < \epsilon \ll 1$  and  $0 < \omega \leq 1$ .

**Remark 3.1** Within the phase-field theory, the boundary condition that accounts for the free energy distribution between the different phases, sets the wetting boundary condition ( $\theta$ ) for the interface, see [14]), which is of the form

$$\mathbf{n} \cdot \nabla C = -\frac{\cos \theta}{\epsilon^2} g'(C).$$

Here,  $g(C)$  is a local surface energy and is set to  $0.75C - 0.25C^3$  in this formulation. Here we consider only the case  $\theta = \pi/2$ .

### Discretization in space

System (17) is discretized in space using the standard conforming piece-wise linear finite elements (FEM) and we use the same finite element space for both variables. The weak variational formulation of (17) reads as follows:

Find  $C, \eta \in H^1(\Omega)$  such that

$$\begin{aligned} (\eta, \nu) - (\Psi'(C), \nu) - \epsilon^2(\nabla C, \nabla \nu) &= 0 \\ \omega(\nabla \eta, \nabla \nu) + \left( \frac{\partial C}{\partial t}, \nu \right) + ((\mathbf{u} \cdot \nabla)C, \nu) &= 0 \end{aligned} \quad (18)$$

for all  $\nu \in H^1(\Omega)$ .

After discretizing (18) in space, the semi-discrete problem reads as follows.

Find  $\mathbf{c}(t) = \{c_i(t)\}_{i=1}^N$  and  $\mathbf{d}(t) = \{d_i(t)\}_{i=1}^N$  such that

$$\begin{aligned} M\mathbf{d}(t) - f(\mathbf{c}(t)) - \epsilon^2 K\mathbf{c}(t) &= 0 \\ \omega K\mathbf{d}(t) + M \frac{d\mathbf{c}(t)}{dt} + W\mathbf{c}(t) &= 0, \end{aligned} \quad (19)$$

where the elements of the vector  $f(\mathbf{c}(t)) = \{f_i(\mathbf{c}(t))\}_{i=1}^N$  are defined as  $f_i(\mathbf{c}(t)) = (\Psi'(C^h(t)), \chi_i)$  and  $M$ ,  $K$  and  $W$  are, respectively, the Gramian mass matrix, the discrete Laplacian stiffness matrix and the matrix, corresponding to the discrete convective term in (18), respectively.

### Discretization in time

For discretization of the time derivative we consider the  $\theta$ -method, and more precisely, the two schemes corresponding to the backward Euler ( $\theta = 1$ ) and Crank-Nicolson ( $\theta = 1/2$ ) methods. Both are implicit, with first and second order accuracy in time, respectively (see, e.g., [4]).

Consider a sequence of time steps,  $\{t_k\}$ ,  $k = 0, 1, \dots$ , where  $t_0 = 0$ ,  $t_k = t_{k-1} + \Delta t_k$ , and denote by  $\mathbf{c}^k = \mathbf{c}(t_k)$ ,  $\mathbf{d}^k = \mathbf{d}(t_k)$ . After applying the  $\theta$ -method,  $\theta \in [0, 1]$ , the fully discretized C-H system to be solved at each time step  $k = 1, 2, \dots$  reads as follows:

$$\begin{aligned} M\mathbf{d}^k - f(\mathbf{c}^k) - \epsilon^2 K\mathbf{c}^k &= 0 \\ \theta \omega \Delta t_k K\mathbf{d}^k + M\mathbf{c}^k + \theta \Delta t_k W\mathbf{c}^k + (1-\theta)(\omega \Delta t_k K\mathbf{d}^{k-1} + \Delta t_k W\mathbf{c}^{k-1}) - M\mathbf{c}^{k-1} &= 0. \end{aligned} \quad (20)$$

For  $\theta = 1$  we obtain the fully implicit backwards Euler form

$$\begin{aligned} M\mathbf{d}^k - f(\mathbf{c}^k) - \epsilon^2 K \mathbf{c}^k &= 0 \\ \omega \Delta t_k K \mathbf{d}^k + M \mathbf{c}^k + \Delta t_k W \mathbf{c}^k - M \mathbf{c}^{k-1} &= 0. \end{aligned} \quad (21)$$

Thus, the time stepping procedure is as follows. Starting with initial vectors  $\mathbf{c}^0 = \{C_0(x_i)\}_{i=1}^N$  and  $\mathbf{d}^0 = \{(\Psi'(C_0) - \epsilon^2 \Delta C_0)(x_i)\}_{i=1}^N$ , the vectors  $\mathbf{c}^k$  and  $\mathbf{d}^k$  that correspond to successive time steps  $t_k$ ,  $k = 1, 2, \dots$ , are computed using the already known approximations  $\mathbf{c}^{k-1}$  and  $\mathbf{d}^{k-1}$  at the previous time step. At each time step one has to solve a nonlinear system, (20).

**Remark 3.2** As is known, in certain cases, when using the Crank-Nicolson scheme ( $\theta = 1/2$ ), the approximate solutions can contain (decaying) spurious oscillations. Therefore, in particular in the beginning of the time interval, it is recommended to use the fully implicit Euler method with small time steps, or  $\theta = 1/2 + \zeta$  for some small  $\zeta$ ,  $0 < \zeta \ll 1$ . If  $\zeta = O(\Delta t)$ , the latter method preserves the second order of accuracy as for the Crank-Nicolson method, see, for instance [2].

### Handling the nonlinearity

Due to the term  $f(\mathbf{c}^k)$  in (20), the systems to be solved at each time step are nonlinear. Therefore, some nonlinear solution technique must be used and we choose here Newton's method.

For convenience, let  $\mathbf{X}^k = [\mathbf{d}^k, \mathbf{c}^k]^T$  denote the combined vector of unknowns. Then the problem of solving the system (20) can be rewritten in the following way.

Find  $\mathbf{X}^k \in \mathbb{R}^{2N}$ , such that

$$F_k(\mathbf{X}^k) = 0, \quad (22)$$

where the nonlinear operator  $F_k(\mathbf{X}^k)$  has the form

$$F_k(\mathbf{X}^k) = \begin{bmatrix} M\mathbf{d}^k - f(\mathbf{c}^k) - \epsilon^2 K \mathbf{c}^k \\ \theta \omega \Delta t_k K \mathbf{d}^k + M \mathbf{c}^k + \theta \Delta t_k W \mathbf{c}^k + (1-\theta)(\omega \Delta t_k K \mathbf{d}^{k-1} + \Delta t_k W \mathbf{c}^{k-1}) - M \mathbf{c}^{k-1} \end{bmatrix}. \quad (23)$$

We note that  $F_k(\mathbf{X}^k)$  changes from one time step to the next, since  $\mathbf{c}^k$ ,  $\mathbf{c}^{k-1}$ ,  $\mathbf{d}^k$  and  $\mathbf{d}^{k-1}$  are different for different  $k$ .

To solve (22) we apply the classical Newton method. Starting with an initial guess  $\mathbf{X}^{k,0} = \mathbf{X}^{k-1} = [\mathbf{d}^{k-1}, \mathbf{c}^{k-1}]^T$  and  $s = 0, \dots$ , until a suitable stopping criterion is met, we find an update of the solution  $\Delta \mathbf{X}^{k,s}$  by solving the system

$$F_k'(\mathbf{X}^{k,s}) \Delta \mathbf{X}^{k,s} = -F_k(\mathbf{X}^{k,s}) \quad (24)$$

and form the next approximate solution as  $\mathbf{X}^{k,s+1} = \mathbf{X}^{k,s} + \Delta \mathbf{X}^{k,s}$ . Here,  $F_k'(\mathbf{X}^k)$  is the Jacobian of  $F_k(\mathbf{X}^k)$ . Thus, at each time step we find an approximate solution to the nonlinear problem (22) by a sequence of Newton iterations, each involving a solution of a linear system with the Jacobian matrix.

From here on we omit the superscript  $k$  and use only  $\mathbf{X}$  instead. The operator  $F_k(\mathbf{X}) = \{F_{k,j}(\mathbf{X})\}_{j=1}^{2N}$  is a vector function of a vector  $\mathbf{X} = \{X_i\}_{i=1}^{2N}$ , with  $X_i = d_i, i = 1, \dots, N$  and  $X_{N+i} = c_i, i = 1, \dots, N$ .

A straightforward computation shows that

$$F'_k(\mathbf{X}) = \begin{bmatrix} \theta M & -\theta J(\mathbf{c}) - \theta \epsilon^2 K \\ \theta \omega \Delta t_k K & M + \theta \Delta t_k W \end{bmatrix}, \quad (25)$$

where  $J(\mathbf{c})$  is the Jacobian of the nonlinear term  $f(\mathbf{c})$  only. Further,  $J(\mathbf{c})$  can be assembled in the usual FEM manner from element matrices  $J_e, e \in \mathcal{T}$  that have the form  $J_e = j_e M_e$ , where  $M_e$  is the corresponding element mass matrix and  $j_e$  depends on the nodal values of the concentration vector  $\mathbf{c}$  at the previous Newton step. Since there holds  $-1 \leq c_i \leq 1$ , one can see that  $-1 \leq j_e \leq 2$ , where a value close to  $-1$  corresponds to an element  $e$  on the interface between the two phases and value of 2 corresponds to an element away from the interface, containing only one of the phases. More details are given in [9].

### Properties of the arising matrices

We observe that for a fixed mesh, the mass and stiffness matrices  $M$  and  $K$  need to be assembled only once in the beginning of the time stepping procedure. When the C-H equation is coupled with N-S, the convection matrix  $W$  depends on the velocity field and, thus, has to be recomputed at each time step. The only matrix to be recomputed at each Newton iteration is  $J(\mathbf{c}^{k,s})$ . For brevity, from here on we use  $J$  for  $J(\mathbf{c}^{k,s})$ .

At each nonlinear iteration step we need to solve systems with  $F'_k(\mathbf{X}^k)$ , as in (25). For simplicity, in the sequel we use the generic name  $A$  to denote the system matrix, namely, we let

$$F'_k(\mathbf{X}) \equiv A = \begin{bmatrix} \theta M & -\theta(J + \epsilon^2 K) \\ \theta \omega \Delta t_k K & M + \theta \Delta t_k W \end{bmatrix}, \quad (26)$$

with  $\theta = 1$  or  $1/2 + \zeta$ .

Clearly,  $A$  is nonsymmetric. We take a closer look at the problem at hand. To be specific, we consider the matrix, resulting from the implicit Euler scheme, however, the analysis is analogous for the case  $\theta \leq 1/2$ .

In general, the difference in magnitude of the entries of the Jacobian matrix  $A$  depends on the problem parameters  $\epsilon$  and  $\omega$ , the discretization parameters  $h$  and  $\Delta t$ , and the amount of convection in the problem, relative to diffusion. In turn, the choice of  $\Delta t$ , related to  $h$ , which controls the total discretization error, depends on the aim of the numerical simulation. When the task is to obtain the stationary solution,  $\Delta t$  should be taken as big as possible, within the desired discretization error bounds.

First, we estimate the order of the entries of the blocks in  $A$ , as a function of the problem and discretization parameters. We see that

$$O(A) = \begin{bmatrix} h^2 & -h^2|\alpha_e| - \epsilon^2 \\ \omega \Delta t & h^2 + \Delta t h \end{bmatrix}. \quad (27)$$

The parameter  $\epsilon$  is the thickness of the interface. Since the interface is expected to be thin, it has to be resolved reasonably well by the mesh. Thus,  $h$  has to be chosen as  $\epsilon/r$ , where  $r$  is an integer, usually between 5 – 10. The parameter  $\omega$  in many applications is of order one. We see from (27), that when  $\Delta t = O(h^2)$ , the size of the entries in all blocks in  $A$  is  $O(h^2)$ . In case the velocity field is of order  $O(1)$ , the term, corresponding to convection, becomes of order  $O(h^3)$  and is not dominating. The entries in  $J$  have different signs, thus, in general the block  $J + \epsilon^2 K$  may be indefinite.

From (27) we also see, that when  $h$  is small and  $\Delta t$  is also small relative to  $h$ , the influence of  $J$  and  $W$  becomes negligible and the matrix

$$P = \begin{bmatrix} \theta M & -\theta \epsilon^2 K \\ \theta \omega \Delta t_k K & M \end{bmatrix} \quad (28)$$

becomes a good candidate for a preconditioner of  $A$ . This proposition is further quantified in Section 4.

## 4 Preconditioners for the discrete Cahn-Hilliard equation

Next, we discuss the task to solve systems with  $F'_k(\mathbf{X}^k)$  of the form (26) using some preconditioned iterative solution method.

As a rule, efficient preconditioners and iterative methods for solving linear systems of equations utilize the properties of the corresponding system matrices. One such important property is the block structure. Due to the fact that  $A$  arises from the discretization of a system of PDEs, it admits a two-by-two block form.

We note that the blocks  $M$  and  $K$  are symmetric,  $M$  is positive definite and  $K$  is positive semi-definite,  $-\theta(J + \epsilon^2 K)$  is symmetric, possibly indefinite, and  $M + \theta \Delta t_k W$  is in general nonsymmetric. In the case of zero convection, the lower diagonal block becomes also symmetric and positive definite.

We start by considering the issue of simplification of  $A$ . As it has already been discussed, for small enough  $h$  and  $\Delta t$ , the influence of the blocks  $J$  and  $W$  diminishes and the idea to neglect those naturally arises. To this end we first simplify the notations and assume that  $\theta = 1$ , thus, let

$$A = \begin{bmatrix} M & -\epsilon^2 K - J \\ \delta K & M + \Delta t W \end{bmatrix}, \quad (29)$$

where  $\delta = \omega \Delta t_k$ . Then we consider the simplified matrix

$$A_0 = \begin{bmatrix} M & -\epsilon^2 K \\ \delta K & M \end{bmatrix}, \quad (30)$$

and analyze the properties of the spectrum of the generalized eigenvalue problem

$$A\mathbf{q} = \lambda A_0 \mathbf{q}. \quad (31)$$

First, we transform (31) to

$$(A - A_0)\mathbf{q} = \mu A_0 \mathbf{q}, \quad (32)$$

where  $\mu = \lambda - 1$ . Next, we note, that

$$A_0^{-1} = \begin{bmatrix} S_0^{-1} & \epsilon^2 M^{-1} K S_0^{-1} \\ -\delta S_0^{-1} K M^{-1} & S_0^{-1} \end{bmatrix}, \quad (33)$$

where  $S_0 = M + \epsilon^2 \delta K M^{-1} K$ . From the expression

$$A_0^{-1}(A - A_0) = \begin{bmatrix} 0 & -S_0^{-1} J + \epsilon^2 \Delta t M^{-1} K S_0^{-1} W \\ 0 & +\delta S_0^{-1} K M^{-1} J + \Delta t S_0^{-1} W \end{bmatrix} \quad (34)$$

we see that the eigenvalue problem (32) has  $N$  zero eigenvalues, corresponding to eigenvectors  $\begin{bmatrix} \mathbf{q}_1 \\ 0 \end{bmatrix}$ , where  $N$  is the size of the blocks. Thus,  $N$  of the eigenvalues of (31) are equal to 1. To analyze the rest, we write out (32) in detail.

$$\begin{aligned} -J\mathbf{q}_2 &= \mu(M\mathbf{q}_1 - \epsilon^2 K\mathbf{q}_2) \\ \Delta t W\mathbf{q}_2 &= \mu(\delta K\mathbf{q}_1 + M\mathbf{q}_2). \end{aligned} \quad (35)$$

We express  $\mathbf{q}_1$  from the first equation in (35), substitute it into the second equation and after some transformations, we obtain

$$\Delta t(M^{-1}W + \omega M^{-1}K M^{-1}J)\mathbf{q}_2 = \mu(I + \Delta t \omega \epsilon^2 (M^{-1}K)^2)\mathbf{q}_2. \quad (36)$$

We recall that  $\epsilon = rh$ , where  $r \geq 5$ ,  $0 < \omega \leq 1$ , and  $\Delta t$  is of order  $O(h)$  or smaller. Consider the matrix

$$Q = \Delta t(I + \Delta t \omega \epsilon^2 (M^{-1}K)^2)^{-1}(M^{-1}W + \omega M^{-1}K M^{-1}J).$$

Then

$$\begin{aligned} \|Q\| &\leq \Delta t \frac{\|M^{-1}W\|}{\|I + \Delta t \omega \epsilon^2 (M^{-1}K)^2\|} + \Delta t \omega \frac{\|M^{-1}K\| \|M^{-1}J\|}{\|I + \Delta t \omega \epsilon^2 (M^{-1}K)^2\|} \\ &\leq \frac{\Delta t}{2\sqrt{\Delta t \omega \epsilon}} \frac{\|M^{-1}W\|}{\|(M^{-1}K)\|} + \frac{\Delta t \omega}{2\sqrt{\Delta t \omega \epsilon}} \|M^{-1}J\| \\ &= O(\sqrt{Pe} \sqrt{\Delta t}) + O(\frac{1}{\sqrt{Pe}} \sqrt{\Delta t} h^{-1}) \\ &= O(\sqrt{\zeta} \sqrt{\Delta t} h^{-1}) + O(\frac{1}{\sqrt{\zeta}} \sqrt{\Delta t} h^{-1}) \end{aligned} \quad (37)$$

The above result is based on the following arguments:  $\|M^{-1}W\| = O(h^{-1})$ ,  $\|M^{-1}K\| = O(h^{-2})$ ,  $\|M^{-1}J\| = O(1)$ ,  $|a|/(1 + a^2 b^2) \leq 1/(2b)$ ,  $\epsilon = rh$ . Further, we recall that  $0 < \omega \leq 1$  is the inverse Peclet number, which may be large (see the appendix). We note that  $Pe h$  is the so-called *mesh Peclet number*, which should be small enough, say,  $Pe h = \zeta$  for some  $0 < \zeta \ll 1$ , in order to insure sufficient resolution of the discretization mesh.

The relations (37) show how to choose the time step  $\Delta t$  in order to ensure a high quality of the preconditioner  $A_0$ . We collect the results in a proposition.



**Proposition 1** For the eigensolutions  $(\lambda^{(s)}, \mathbf{q}^{(s)})$  of the generalized eigenvalue problem (31), there holds

$$\begin{cases} \lambda^{(s)} = 1, & \mathbf{q}^{(s)} = \begin{bmatrix} \mathbf{q}_1^{(s)} \\ \mathbf{0} \end{bmatrix}, & s = 1, \dots, N, \\ \lambda^{(s)} = 1 + \varsigma, & & s = 1, \dots, N, \end{cases} \quad (38)$$

where,

(i)  $\varsigma \rightarrow 0$  for  $\Delta t < h^2$ , when  $Pe = 1$ ,

(ii)  $\varsigma \rightarrow 0$  for  $\Delta t < h$ , when  $Pe > 1$ ,

We see from the above, that when keeping  $h$  fixed and enlarging  $Pe$ , even if we choose  $\Delta t < h$ , we can expect an increase in the iteration counts due to an increase of the corresponding proportionality constant.

An analysis of the behaviour of the eigenvalues of (31) is made also in [7], however the derivations are based on (34).

In Section 4.1, we address the question how to solve efficiently systems with the matrix  $A_0$ .

#### 4.1 Efficient solution of systems with $A_0$

It turns out that systems with  $A_0$  can be solved very efficiently by preconditioned inner iterations.

An analysis in [7], related to earlier work in [5], shows that the matrix

$$\hat{A}_0 = \begin{bmatrix} M & -\epsilon^2 K \\ \delta K & M + 2\epsilon\sqrt{\delta}K \end{bmatrix}$$

is an optimal preconditioner for  $A_0$ , and, furthermore, can be efficiently implemented.

For completeness, we include here the derivations.

**Proposition 2** (Proposition 4.1 in [7]) Consider the matrices

$$\mathcal{A} = \begin{bmatrix} H & -\alpha^2 F^T \\ \beta^2 F & \gamma^2 H \end{bmatrix} \quad \text{and} \quad \mathcal{B} = \begin{bmatrix} H & -\alpha^2 F^T \\ \beta^2 F & \gamma^2 H + \alpha\beta\gamma(F + F^T) \end{bmatrix}$$

where  $H$  is spd, the symmetric part of  $F$ ,  $\frac{1}{2}(F + F^T)$  is positive semidefnite and  $\alpha$ ,  $\beta$  and  $\gamma$  are some real constants. Then, the eigenvalues of the generalized eigenvalue problem

$$\lambda \mathcal{B} \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{bmatrix} = \mathcal{A} \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{bmatrix} \quad (39)$$

satisfy the relations:

$$\begin{cases} \lambda \in [0.5, 1], \\ \lambda = 1 \quad \text{for } \mathbf{v}_2 \text{ in the null space of } F + F^T. \end{cases}$$

*Proof*

We note first, that

$$\mathcal{A} = \mathcal{B} - \begin{bmatrix} 0 & 0 \\ 0 & \alpha\beta\gamma(F + F^T) \end{bmatrix}.$$

Then, problem (39) is equivalent to

$$(\lambda - 1)\mathcal{B} \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & -\alpha\beta\gamma(F + F^T) \end{bmatrix} \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{bmatrix},$$

i.e.,

$$\begin{aligned} (\lambda - 1)(H\mathbf{v}_1 - \alpha^2 F^T \mathbf{v}_2) &= 0 \\ (\lambda - 1)(\beta^2 F \mathbf{v}_1 + (\gamma^2 H + \alpha\beta\gamma(F + F^T))\mathbf{v}_2) &= -\alpha\beta\gamma(F + F^T)\mathbf{v}_2. \end{aligned} \quad (40)$$

Clearly, if  $\mathbf{v}_2$  is in the null space of  $F + F^T$ , then  $\lambda = 1$ . Next, we consider  $\lambda \neq 1$  and express  $\mathbf{v}_1 = \alpha^2 H^{-1} F^T \mathbf{v}_2$  from the first equality in (40). Substituting it in the second equality, we obtain

$$(\lambda - 1)(\beta^2 \alpha^2 F H^{-1} F^T + \gamma^2 H + \alpha\beta\gamma(F + F^T))\mathbf{v}_2 = -\alpha\beta\gamma(F + F^T)\mathbf{v}_2.$$

We combine the similar terms, and after straight-forward transformations obtain

$$\left(\frac{1}{\lambda} - 1\right)(\beta^2 \alpha^2 F H^{-1} F^T + \gamma^2 H)\mathbf{v}_2 = \alpha\beta\gamma(F + F^T)\mathbf{v}_2.$$

Since  $H$  is spd, we transform the latter relation as

$$\left(\frac{1}{\lambda} - 1\right)(I + \tilde{F}\tilde{F}^T)\tilde{\mathbf{v}}_2 = (\tilde{F} + \tilde{F}^T)\tilde{\mathbf{v}}_2,$$

where  $\tilde{F} = \frac{\alpha\beta}{\gamma}H^{-\frac{1}{2}}FH^{-\frac{1}{2}}$  and  $\tilde{\mathbf{v}}_2 = H^{\frac{1}{2}}\mathbf{v}_2$ . Hence,

$$\frac{1}{\lambda} - 1 = \frac{\tilde{\mathbf{v}}_2^T (\tilde{F} + \tilde{F}^T) \tilde{\mathbf{v}}_2}{\tilde{\mathbf{v}}_2^T \tilde{\mathbf{v}}_2 + (\tilde{F}^T \tilde{\mathbf{v}}_2)^T (\tilde{F}^T \tilde{\mathbf{v}}_2)}.$$

We now utilize the assumption that  $F + F^T$  is positive semidefinite together with the Cauchy-Schwartz inequality, and obtain:

$$0 \leq \tilde{\mathbf{v}}_2^T (\tilde{F} + \tilde{F}^T) \tilde{\mathbf{v}}_2 = (\tilde{F}^T \tilde{\mathbf{v}}_2)^T \tilde{\mathbf{v}}_2 + \tilde{\mathbf{v}}_2^T (\tilde{F}^T \tilde{\mathbf{v}}_2) = 2\tilde{\mathbf{v}}_2^T (\tilde{F}^T \tilde{\mathbf{v}}_2) \leq \tilde{\mathbf{v}}_2^T \tilde{\mathbf{v}}_2 + (\tilde{F}^T \tilde{\mathbf{v}}_2)^T (\tilde{F}^T \tilde{\mathbf{v}}_2).$$

The latter shows that  $0 \leq \frac{1}{\lambda} - 1 \leq 1$ , or  $1 \leq \frac{1}{\lambda} \leq 2$ , that is  $\frac{1}{2} \leq \lambda \leq 1$ .  $\blacksquare$

Further,  $\hat{A}_0$  possesses the following factorization (cf. [5], [7])

$$\hat{A}_0 = \begin{bmatrix} M & -\epsilon^2 K \\ \delta K & M + 2\epsilon\sqrt{\delta}K \end{bmatrix} = \begin{bmatrix} M & 0 \\ \delta K & M + \epsilon\sqrt{\delta}K \end{bmatrix} \begin{bmatrix} I & -\epsilon^2 M^{-1}K \\ 0 & M^{-1}(M + \epsilon\sqrt{\delta}K) \end{bmatrix}. \quad (41)$$

Thus, implementing straightforwardly a block solver for (41), we see that solutions with  $\hat{A}_0$  require one solution with  $M$ , two with  $M + \epsilon\sqrt{\delta}K$  and some matrix-vector operations:

<p>Forward step:</p> <p>(1) Solve <math>M\mathbf{y}_1 = \mathbf{b}_1</math></p> <p>(2) Solve <math>(M + \epsilon\sqrt{\delta}K)\mathbf{y}_2 = \mathbf{b}_2 - \delta K\mathbf{y}_1</math></p>	<p>Backward step:</p> <p>(3) Solve <math>(M + \epsilon\sqrt{\delta}K)\mathbf{x}_2 = M\mathbf{y}_2</math></p> <p>(4) Compute <math>\mathbf{x}_1 = \mathbf{y}_1 - \frac{\epsilon}{\sqrt{\delta}}(\mathbf{y}_2 - \mathbf{x}_2)</math></p>
--	--

(42)

We envision the following computational procedures to solve linear systems with the matrix  $A$ .

Algorithm [A] (included for completeness)

[A1] Solve  $A$ , preconditioned by  $A_0$

[A2] Solve  $A_0$  by inner iterations, preconditioned by  $\hat{A}_0$ , factored as in (41)

[A3] Solve  $\hat{A}_0$  via the computational steps (42)

[A4] Solve  $M$  and  $M + \epsilon\sqrt{\delta}K$  using some suitable solution procedure

Algorithm [A0] (recommended)

[A01] Solve  $A$ , preconditioned by  $\hat{A}_0$ , factored as in (41)

[A02] Solve  $\hat{A}_0$  via the computational steps (42)

[A03] Solve  $M$  and  $M + \epsilon\sqrt{\delta}K$  using suitable solution procedures

One last issue to consider is the solution with  $M$  and  $M + \epsilon\sqrt{\delta}K$ . We see, that the matrix  $M + \epsilon\sqrt{\delta}K$  is suitable to be solved by some off-the-shelf AMG solver. We can simplify the preconditioner further and replace  $M$  by its diagonal,  $D_M = \text{diag}(M)$ , which makes solutions with it trivial. The matrix  $D_M + \epsilon\sqrt{\delta}K$  is nonsingular, easy to form explicitly and can be solved by an AMG solver. Below we analyse the effect of replacing  $M$  by  $D_M$  on the already derived condition number estimates for  $\hat{A}_0^{-1}A_0$ .

**Remark 4.1** For the solution of systems with  $\hat{A}_0$ , it is shown in [7] that the complexity of the solution with the preconditioner can be further reduced, avoiding one solution with  $M$  by rearranging the computations in the solution procedure.

## 5 Numerical experiments

The preconditioning techniques, presented above, are applicable for a general setting of problem parameters of the C-H equation. To illustrate the performance of the preconditioning techniques, discussed in Section 4, we use the following test problems.

### **Problem 1 (Phase separation and coarsening due to diffusion)**

We consider problem (17) in  $\Omega = [-1/2, 1/2] \times [0, 1]$  with parameters  $\omega = 1$ ,  $\epsilon = 0.0625$  and  $\mathbf{u} = (0, 0)$ . Thus, no convection due to fluid flow is included in the model and the process of phase separation and coarsening takes place only due to diffusion. Figure 1 illustrates the evolution of a binary mixture in time.

### **Problem 2 (Front movement due to convection)**

We consider problem (17) in  $\Omega = [-1, 1] \times [0, 1]$  with  $\omega = 1/300$ , and  $\epsilon = 0.1$ . The velocity vector is assumed to be constant in time with components  $\mathbf{u} = (1, 0)$ . The initial condition is  $C_0 = -\tanh(10x_1)$ . This problem describes the movement of the front between the two phases, as shown in Figure 2.

In order to verify our numerical approach, we perform the following two tests to solve the fully coupled convective C-H equation with the Navier-Stokes equations.

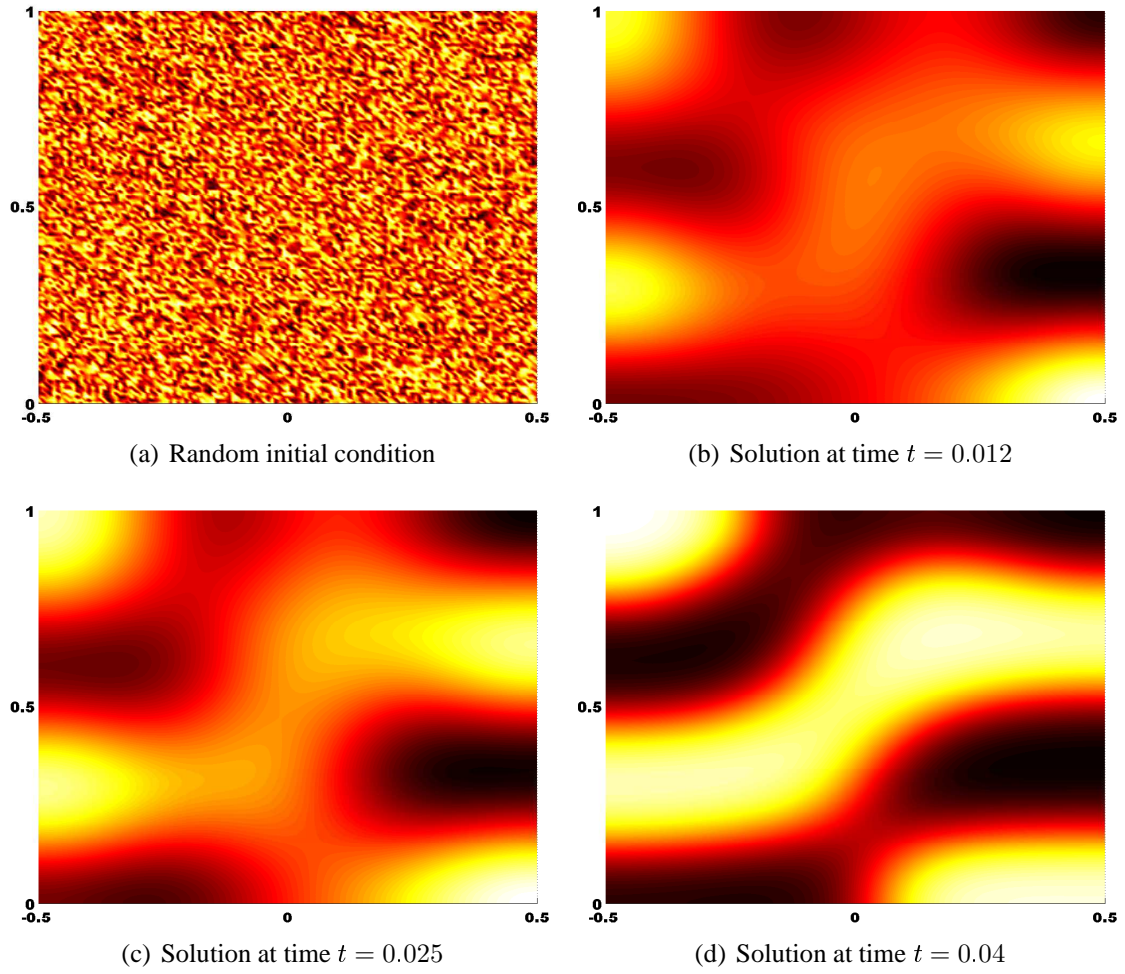


Figure 1: Phase separation and coarsening with no convection

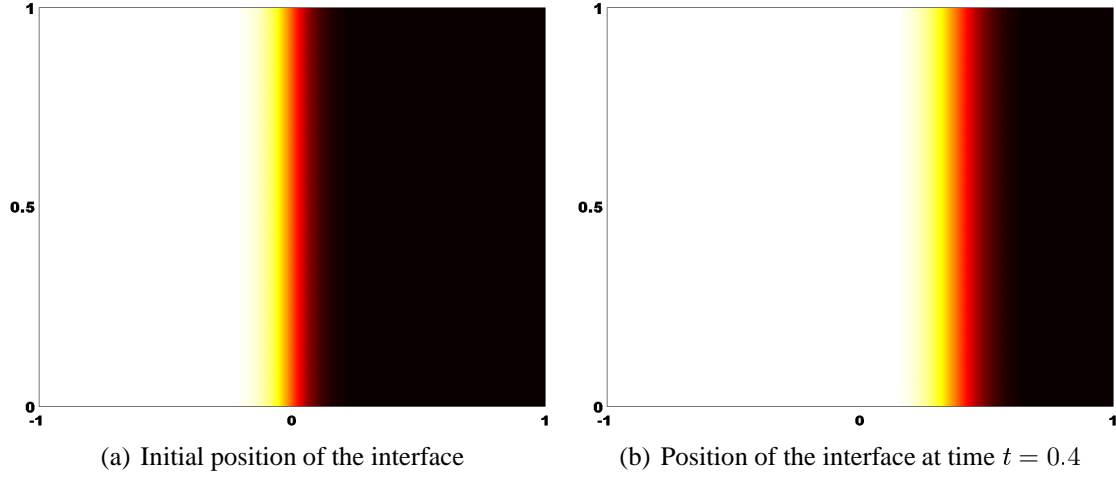


Figure 2: Interface movement for convective Cahn-Hilliard

**Problem 3 (The static bubble test)**

Consider a circular bubble in static equilibrium. In this setting, the net surface force should be zero, since at each point on the bubble surface the tension force is counteracted by a force, equal in size and opposite in direction, acting at a diametrically opposed point. The correct solution is a zero velocity field and a pressure field that rises from a constant value of  $p_{out}$  outside the bubble to a value of  $p_{in} = p_{out} + \sigma/R$  inside the bubble, according to the Laplace-Young law ( $R$  is the bubble radius). Figure 4 illustrates the correctness of the computed solution.

**Problem 4 (The dynamic bubble test)**

We consider an elliptic droplet with main axes  $a = 0.16$  and  $b = 0.14$ , Figure 5(a). The non-dimensional numbers are  $Re = 10$  and  $Ca = 0.1$  and  $Cn = 0.00625$ . The droplet oscillates due to the curvature until it reaches a circular shape with radius  $R = 0.15$ , Figure 5(b). As a result of numerical simulations one can observe spurious velocity currents of amplitude  $U < 1 \cdot 10^{-10}$  near the bubble interface. There are few vortices with  $|U| < 1 \cdot 10^{-5}$  outside of the interface.

The numerical tests in Tables 1-4 are performed in Matlab and those in Tables 5-7 - in Fortran. To solve systems with  $A$ , a nonsymmetric iterative solution method is applied. In Matlab we use the Generalized Conjugate Gradient - Minimal Residual (GCG-MR) method, see e.g. [3] and GMRES in the rest of the experiments.

In all numerical tests, presented throughout the paper, we use a regular triangular mesh with a characteristic mesh size  $h$ . Note that due to the smaller interface thickness, we need a better resolution for Problem 1 than for Problem 2.

The stopping criterion for the Newton method at each time step is always taken to be  $\|\Delta \mathbf{X}^{k,s}\| < 10^{-6}$ . The iterative method used to solve the arising systems is GCG-MR, and the solution process is stopped when the norm of the residual is reduced by a factor  $10^{-6}$  or the norm itself is smaller than  $10^{-12}$ .

Size	$\Delta t$				
	$h$	$h/2$	$h/4$	$h/10$	$h^2$
$M$					
8450	7 / 28	3 / 12	3 / 9	3 / 8	3 / 8
33282	3 / 11	3 / 8	3 / 8	3 / 7	3 / 7
132098	3 / 8	3 / 7	3 / 7	3 / 7	3 / 7
$diag(M)$					
8450	7 / 26	3 / 13	3 / 12	3 / 11	3 / 12
33282	3 / 12	3 / 11	3 / 10	3 / 11	3 / 12
132098	3 / 9	3 / 9	3 / 10	3 / 10	3 / 12

Table 1: Problem 1, no convection:  $A$  is preconditioned with  $\hat{A}_0$

In Tables 1-4 we monitor the convergence of the nonlinear method, as well as that of various combinations of inner solvers and their influence on the number of the nonlinear iterations. We report results, averaged over ten time steps. Each table cell contains Two or three integer digits of the form  $N_1/N_2$  or  $N_1/N_2/N_3$ , where  $N_1$  denotes the average number of Newton iterations per time step,  $N_2$  is the average number of GCG-MR iterations per Newton iteration and  $N_3$ , whenever present, shows the average number of AMG-preconditioned conjugate gradient (PCG) iterations to solve iteratively systems with  $M + \epsilon\sqrt{\delta}K$ .

As predicted, for relatively small values of  $\Delta t$ , which is achieved already for  $h/4$  and  $h/10$ , the performance of the preconditioner is stable with respect to both discretization parameters.

Tables 1 to 4 illustrate the numerical performance of Algorithm [A0]. We use  $\hat{A}_0$  as a preconditioner for  $A$ , solved as shown in (42). We test first the convergence when systems with  $M + \epsilon\sqrt{\delta}K$  are solved via direct method (Tables 1 and 3) and next - when  $M + \epsilon\sqrt{\delta}K$  is solved by an AMG-preconditioned conjugate gradient method (Tables 2 and 4). We also illustrate the effect of replacing  $M$  by its diagonal.

The tables consist of two parts. In the upper part of each table, we present the iteration counts where the original mass matrix  $M$  is used. In the lower part, we see the effect of replacing  $M$  by its diagonal in all blocks of  $\hat{A}_0$  - the linear iterations increase approximately by a factor of two.

In Tables 1 and 3 we see the averaged nonlinear and linear iterations to solve the Jacobian matrix  $A$ , preconditioner by  $\hat{A}_0$ , where systems with  $M + \epsilon\sqrt{\beta}K$  are solved directly.

In Tables 2 and 4,  $\hat{A}_0$  is considered in its factored form (41) and systems with it are solved via Algorithm (42). The matrix blocks  $M + \epsilon\sqrt{\beta}K$  are solved by an inner PCG solver with AMG from [25] as a preconditioner. The inner stopping criterion is  $10^{-3}$ , which results in about two to three iterations. We see that the inner solver does not influence the outer convergence of both the nonlinear and the linear solution methods.

Table 5 contains comparisons of the performance of the iterative solver with that of a fast sparse direct method (MUMPS, see [31], the same solver is also used in [14]), shows the computational time and the memory consumption for four different problem sizes. When the size of the problem is small enough, the direct solver is faster than our block preconditioned iterations. For

Size	$\Delta t$				
	$h$	$h/2$	$h/4$	$h/10$	$h^2$
$M$					
8450	7 / 22 / 3	3 / 14 / 3	3 / 11 / 3	3 / 8 / 3	3 / 8 / 3
33282	3 / 13 / 3	3 / 10 / 3	3 / 9 / 3	3 / 8 / 3	3 / 7 / 3
132098	3 / 18 / 3	3 / 9 / 3	3 / 8 / 3	3 / 8 / 3	3 / 7 / 3
$diag(M)$					
8450	7 / 23 / 3	3 / 14 / 3	3 / 13 / 3	3 / 12 / 3	3 / 12 / 3
33282	3 / 15 / 3	3 / 13 / 3	3 / 12 / 3	3 / 12 / 3	3 / 12 / 3
132098	3 / 16 / 3	3 / 12 / 3	3 / 11 / 3	3 / 12 / 3	3 / 12 / 3

Table 2: Problem 1, no convection:  $A$  is preconditioned with  $\hat{A}_0$ , AMG used for the matrix  $M + \epsilon\sqrt{\beta}K$

Size	$\Delta t$				
	$h$	$h/2$	$h/4$	$h/10$	$h^2$
$M$					
4290	4 / 10	4 / 9	3 / 9	3 / 8	3 / 7
16770	4 / 8	3 / 9	3 / 8	3 / 7	3 / 6
66306	4 / 7	3 / 8	3 / 7	3 / 6	3 / 5
$diag(M)$					
4290	4 / 15	4 / 14	3 / 15	3 / 15	3 / 14
16770	4 / 13	3 / 14	3 / 13	3 / 12	3 / 11
66306	3 / 14	3 / 12	3 / 11	3 / 10	3 / 9

Table 3: Problem 2, convection-diffusion:  $A$  is preconditioned with  $\hat{A}_0$

Size	$\Delta t$				
	$h$	$h/2$	$h/4$	$h/10$	$h^2$
$M$					
4290	4 / 10 / 3	4 / 9 / 3	3 / 9 / 3	3 / 8 / 2	3 / 7 / 2
16770	4 / 8 / 3	3 / 9 / 3	3 / 8 / 3	3 / 7 / 3	3 / 6 / 3
66306	4 / 7 / 3	3 / 8 / 3	3 / 7 / 3	3 / 6 / 3	3 / 5 / 3
$diag(M)$					
4290	4 / 15 / 3	4 / 14 / 3	3 / 15 / 3	3 / 15 / 3	3 / 14 / 3
16770	4 / 13 / 3	3 / 14 / 3	3 / 13 / 3	3 / 12 / 3	3 / 11 / 3
66306	3 / 14 / 3	3 / 12 / 3	3 / 11 / 3	3 / 10 / 3	3 / 9 / 3

Table 4: Problem 2, convection-diffusion:  $A$  is preconditioned with  $\hat{A}_0$ , AMG used for the matrix  $M + \epsilon\sqrt{\beta}K$

$h$	$DOF$	Block precondition			Direct solver [14]		
		$N1/N2$	$time(s)$	$Mem(MB)$	$N1$	$time(s)$	$Mem(MB)$
1/256	131 072	4/10	16.98	185	3	7.2	352
1/512	528 392	4/10	72.61	646	3	53.4	1 409
1/768	1 176 578	4/10	170	1 429	3	193.75	3 126
1/1024	2 097 152	4/10	306.05	2 587	<i>Out of memory</i>		

Table 5: Problem 2: Memory consumption and run time for two solution methods

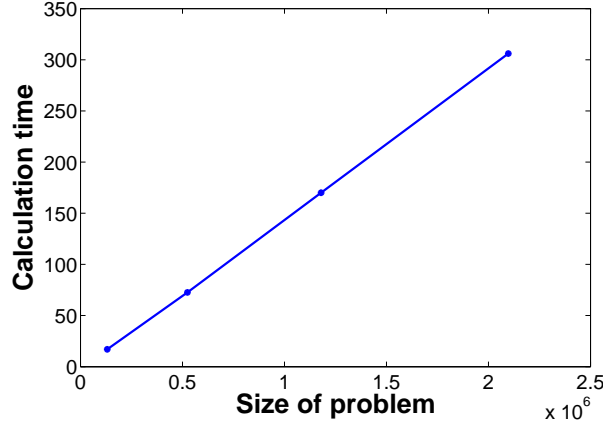


Figure 3: Problem 2: Computing time per linear step vs. problem size

problem sizes, larger than a million degrees of freedom, the iterative method becomes superior over the direct solver in terms of performance. Figure 3 illustrates that the computing time of the iterative solver increases linearly with the problem size.

Tables 6 and 7 contain comparisons between three solution techniques, including ILU-preconditioned GMRES (the same as used in [42]). The ILU preconditioner is constructed for the whole system  $A$ . As can be seen from the tables, the ILU preconditioner is rather efficient in some cases while in others it diverges. The block preconditioner  $\hat{A}_0$  shows a very robust behaviour in all cases.

## 6 Concluding remarks

In this paper we address preconditioning techniques for the iterative solution methods for numerical simulations of multiphase flow problems. The model is based on the Cahn-Hilliard equation. In its original form, it is a fourth order parabolic partial differential equation, however, in this study it is reformulated as a coupled nonlinear system of two second order equations, one of which is time-dependent.

As a space discretization method, we consider finite elements, triangular meshes and con-



$dt$	Precond.with $\hat{A}_0$		Direct solver [14]		ILU[42]	
	$N1/N2$	$time(s)$	$N1$	$time(s)$	$N1/N2$	$time(s)$
$h$	4/10	14.58	3	7.19		
$h/4$	4/10	16.98	3	7.23	<i>no convergence</i>	
$h/5$	4/10	16.77	3	7.16		
$h/10$	4/10	14.67	3	7.21		
$h/20$	4/10	14.62	3	7.21	4/42	23.66
$h/40$	4/10	14.11	3	7.23	4/13	13.23
					4/10	10.55

Table 6: Computing time and number of iteration for three different methods,  $Pe = 1000$

$dt$	Precond.with $\hat{A}_0$		Direct solver [14]		ILU[42]	
	$N1/N2$	$time(s)$	$N1$	$time(s)$	$N1/N2$	$time(s)$
$h$	3/10	16.66	3	7.25		
$h/4$	3/10	16.54	3	7.23	<i>no convergence</i>	
$h/5$	3/10	16.53	3	7.22		
$h/10$	3/10	16.28	3	7.21		
$h/20$	3/10	15.82	3	7.19		
$h/40$	3/10	15.59	3	7.18		

Table 7: Computing time and number of iteration for three different methods,  $Pe = 1$ .

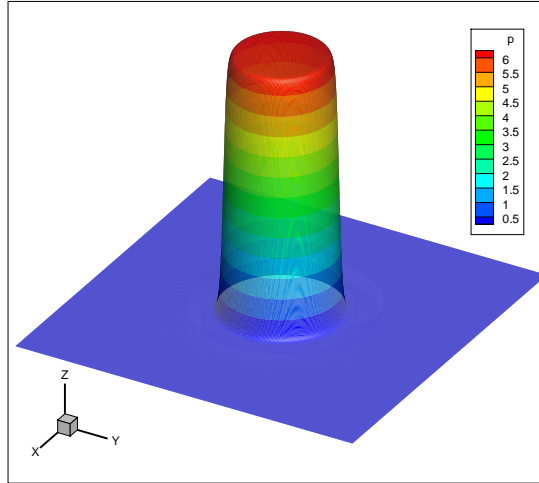


Figure 4: Problem 3: The pressure profile, a simulation with  $Ca = 0.1$ ,  $Re = 10$  and  $Cn = 0.00625$ .

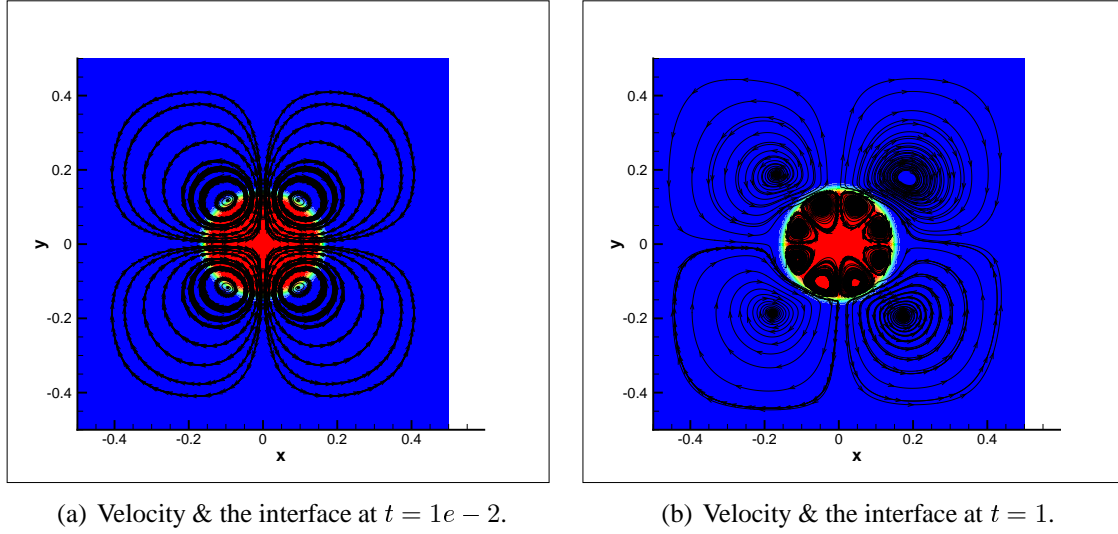


Figure 5: Problem 4: The Interface movement for convective Cahn-Hilliard and the 'spurious currents' around the bubble in a stationary state

forming bilinear basis functions. The system matrix is nonsymmetric, due to the Jacobian of the nonlinear term and, if present, a discrete convection operator. The time discretization is implicit, using the  $\theta$ -method.

The numerical simulation consists of a time-stepping procedure, where during each time step we solve a nonlinear system of algebraic equations using full Newton method. That, in turn, during each nonlinear iteration requires the solution of a linear system with the corresponding Jacobian matrix.

The target goal of this work is the efficient solution of systems with the Jacobian matrix using preconditioned iterative solution methods and the construction and analysis of the convergence properties of the proposed preconditioners.

We consider a preconditioner of block form, which utilizes the available two-by-two block matrix structure. We proceed in two steps. First we simplify the original system by dropping the nonsymmetric matrix blocks. We show that for small enough time step, relative to the space discretization parameter, the resulting matrix is a high quality preconditioner for the original system. In its turn, that resulting system is of special form and can be preconditioned by an optimal preconditioner. Solution of systems of the latter require some vector operations and off-shelf solvers such as AMG.

The approach is straightforwardly applicable in two or three dimensions as well as in an adaptive finite element framework. We note, that for constant meshes, the preconditioner needs not to be recomputed.

We provide extensive numerical evidence of the efficiency of the derived preconditioner. Comparisons with a direct method regarding execution times and memory requirements confirm that the performance of the suggested preconditioning technique is superior over a fast sparse direct solver and also shows a very robust convergence behaviour.

The preconditioner is applicable also to other problems, which exhibit the same matrix structure, such as in constraint optimization with a PDE constraint.

## Acknowledgments

The work of the first author (fully) and the second and the third author (partly) is supported by the Swedish Research Council (VR) via grant VR 2008-5072 '*Finite element preconditioners for algebraic problems as arising in modelling of multiphase microstructures*'. The support is hereby gratefully acknowledged.

The authors express their gratitude to Prof. Owe Axelsson for his comments on the presentation of this paper.

Some of the computations were performed on resources provided by SNIC through Uppsala Multidisciplinary Center for Advanced Computational Science (UPPMAX) under Project p2009040.

We also thank the developers of the HSL MI-20 AMG package for kindly providing us with the AMG-Matlab interface.

## References

- [1] D. M. Anderson, G. B. McFadden, A. A. Wheeler, A phase-field model of solidification with convection, *Physica D*, 135 (2000), 175–194.
- [2] O. Axelsson, High-order methods for parabolic problems, *Journal of Computational and Applied Mathematics*, 1(1) 1975, 5–16.
- [3] O. Axelsson, *Iterative Solution Methods*, Cambridge University Press, 1994.
- [4] O. Axelsson, Finite Difference Methods, *The Encyclopedia of Computational Mechanics, Part I Fundamentals*, Ch. 2, Wiley, 2004.
- [5] O. Axelsson, A. Kucherov, Real valued iterative methods for solving complex symmetric linear systems. *Numerical Linear Algebra with Applications* 7 (2000), 197–218.
- [6] O. Axelsson, M. Neytcheva, A general approach to analyse preconditioners for two-by-two block matrices TR 2010-029, Institute for Information Technology, Uppsala University, November 2010.
- [7] O. Axelsson, M. Neytcheva, Operator splittings for solving nonlinear, coupled multi-physics problems with an application to the numerical solution of an interface problem. TR 2011-009, Institute for Information Technology, Uppsala University, April 2011.
- [8] A. Bertozzi, S. Essedoglu, A. Gillette, Impainting of binary images using the Cahn-Hilliard equation, *IEEE Trans Image Process.* 16(1), 2007, 285–91.

- [9] P. Boyanova, M. Do-Quang, M. Neytcheva, On solution methods for Cahn–Hilliard equation discretized by conforming and non-conforming FEM, TR 2011-004, Institute for Information Technology, Uppsala University, March 2011.
- [10] D. Braess, *Finite Elements: Theory, Fast Solvers, and Applications in Solid Mechanics*, Cambridge University Press, 1997.
- [11] J.W. Cahn, On spinoidal decomposition, *Acta Metallurgica* 9 (1961) 795–801.
- [12] J.W. Cahn, J. Hilliard, Free energy of a nonuniform system. I. Interfacial free energy, *Journal of Chemical Physics* 28 (1958) 258–267.
- [13] H.D. Ceniceros, R.L. N  s, A.M. Roma, Three-dimensional, fully adaptive simulations of phase-field fluid models, *Journal of Computational Physics*, 29 (2010), 6135-6155.
- [14] M. Do-Quang, G. Amberg, The splash of a ball hitting a liquid surface: Numerical simulation of the influence of wetting, *Physics of Fluids*, 2008.
- [15] C.M. Elliott, D.A. French, F.A. Milner, A second order splitting method for the Cahn-Hilliard Equation, *Numerische Mathematik*, 54 (1989), 575–590.
- [16] C.M. Elliott, Z. Songmu, On the Cahn-Hilliard equation, *Archive for Rational Mechanics and Analysis* 96 (1986) 339-357.
- [17] D. Eyre, Systems of Cahn-Hilliard equations, *SIAM Journal on Applied Mathematics*, 53 (1993), 1686–1712.
- [18] D. Eyre, An unconditionally stable one-step scheme for gradient systems. Unpublished article, 1998.
- [19] D. Eyre, Unconditionally gradient stable time marching the Cahn-Hilliard equation, *Computational and mathematical Models of Microstructural Evolution*, 1998.
- [20] D. Eyre, M. Grinfeld, A. Novick-Cohen, The PDE Cahn-Hilliard equation: phase separation and coarsening. Intended for *The Coffee Table Book*. Editor: N. Trefethen, in preparation.
- [21] X. Feng, A. Prohl, Error analysis of a mixed finite element method for the Cahn-Hilliard equation, *Numer. Math.*, 99 (2004), 47–84.
- [22] H. Garcke, On Cahn-Hilliard systems with elasticity, *Proceedings of the Royal Society of Edinburgh* 133 A (2003), 307–331.
- [23] J.W. Gibbs, A method of geometrical representation of the thermodynamic properties of substances by means of surfaces”, *Transactions of the Connecticut Academy*, Vol. II, 1873, 382–404.

- [24] Xin He, Maya Neytcheva, and Stefano Serra Capizzano. On an Augmented Lagrangian-Based Preconditioning of Oseen Type Problems TR 2010-026, Institute for Information Technology, Uppsala University, November 2010.
- [25] The HSL Mathematical Software Library, <http://www.hsl.rl.ac.uk/>
- [26] D. Jacqmin, Contact-line dynamics of a diffuse fluid interface, *Journal of Fluid Mechanics* 402 (2000), 57–88
- [27] Y. Jingxue, On the existence of nonnegative continuous solutions of the Cahn-Hilliard equation, *Journal of Differential Equations* 97 (1992), 310–327.
- [28] D. Kay, R. Welford, A multigrid finite element solver for the Cahn-Hilliard equation, *Journal of Computational Physics*, 212 (2006) 288–304.
- [29] K. Kim, A posteriori error estimators for locally conservative methods of nonlinear elliptic problems, *Applied Numerical Mathematics*, 57 (2007), 1065–1080.
- [30] V. Maz’ya, J. Rossmann, On the Agmon-Miranda maximum principle for solutions of strongly elliptic equations in domains of  $R^n$  with conical points, *Annals of Global Analysis and Geometry* 10 (1992) 125–150.
- [31] MUMPS, <http://graal.ens-lyon.fr/MUMPS/>
- [32] A. Novick-Cohen, The Cahn-Hilliard equation: mathematical and modelling perspectives. *Advances in Mathematical Sciences and Applications*, 8 (1998), 965–985.
- [33] A. Novick-Cohen, The Cahn-Hilliard Equation, *Handbook of Differential Equations. IV Evolutionary Partial Differential Equations*, Editors: C. Dafermos and M. Pokorný, Elsevier 2008, 201–228.
- [34] A. Novick-Cohen, *The Cahn-Hilliard Equation: From Backwards Diffusion to Surface Diffusion*. Draft. in preparation under contract with Cambridge University Press. <http://www.math.technion.ac.il/~amync/>
- [35] J. Pearson, A.J. Wathen, A New Approximation of the Schur Complement in Preconditioners for PDE Constrained Optimization. The Mathematical institute, University of oxford. Technical report, 24 Nov 2010 <http://eprints.maths.ox.ac.uk/1021/>
- [36] Jose-Francisco Rodrigues (Editor). *Mathematical Models for Phase Change Problems: Proceedings European Workshop, Obidos, Portugal, October 120133, 1988*. Basel etc., Birkhuser Verlag 1989. 408 pp., ISBN 3-7643-2309-4 (International Series of Numerical Mathematics 88)
- [37] H.G. Rotstein, S. Brandon, A. Novic-Hohen, A. Nepomnyashchik, Phase-field equations with memory: The Hyperbolic Case, *SIAM Journal on Applied Mathematics*, 62 (2001), 264–282.

- [38] J.D. van der Waals, Thermodynamische theorie der capillariteit in de onderstelling van continue dichtheidsverandering, *Verhandlingen der Koninklijke Akademie van Wetenschappen te Amsterdam, Sect. 1* (Dutch; English translation in *Journal of Statistical Physics*, 1979, 20:197).
- [39] W. Villanueva, G. Amberg, Some generic capillary-driven flows, *International Journal of Multiphase Flow*, 32 (2006), 1072–1086..
- [40] A.J. Wathen, Realistic eigenvalue bounds for the Galerkin mass matrix, *IMA Journal of Numerical Analysis*, 7 (1987), 449–457.
- [41] Steven Wise, Junseok Kim, J. Lowengrub, Solving the regularized, strongly anisotropic. Cahn-Hilliard equation by an adaptive nonlinear multigrid method, *Journal of Computational Physics*, 226 (2007) 414-446.
- [42] P. Yue, C. Zhou, J.J. Feng, C.F. Ollivier-Gooch and H.H. Hu, Phase-field simulations of interfacial dynamics in viscoelastic fluids using finite elements with adaptive meshing, *Journal of Computational Physics*, 219 (2006), 47-67.