SIAM J. IMAGING SCIENCES
Vol. xx, pp. x

Fast Solvers for Cahn-Hilliard Inpainting

Jessica Bosch[†‡], David Kay[§], Martin Stoll[†¶], and Andrew J. Wathen[‖]

Abstract. We consider the efficient solution of the modified Cahn-Hilliard equation for binary image inpainting using convexity splitting, which allows an unconditionally gradient stable time-discretization scheme. We look at a double-well as well as a double obstacle potential. For the latter we get a nonlinear system for which we apply a semi-smooth Newton method combined with a Moreau-Yosida regularization technique. At the heart of both methods lies the solution of large and sparse linear systems. We introduce and study block-triangular preconditioners using an efficient and easy to apply Schur complement approximation. Numerical results indicate that our preconditioners work very well for both problems and show that qualitatively better results can be obtained using the double obstacle potential.

Key words. Cahn-Hilliard equation, image inpainting, binary images, preconditioning, Schur complement approximation

AMS subject classifications. 68U90, 94A08, 65F08, 65F10, 65N22, 65F50, 65M60, 93C20, 35R35, 74S05, 35K55, 80A22

1. Introduction.

Image inpainting is the process of filling in missing or damaged parts of images based on information from surrounding areas. Bertalmio et al. [3] introduced image inpainting for digital image processing. Their model is based on nonlinear PDEs. A number of variational- and PDE-based approaches have been considered, like the total variation (TV) model [48, 47, 41, 42], Eulers elastica model [11, 49, 35], the active contour model based on Mumford and Shahs segmentation [51], the inpainting scheme based on the Mumford-Shah-Euler image model [20], inpainting with the Navier-Stokes equation [2] or wavelet-based inpainting [12, 15].

Second order variational inpainting methods have drawbacks as in the connection of edges over large distances or the smooth propagation of level lines into the damaged domain, see e.g. [46]. A new approach in the class of fourth order inpainting is the application of a modified Cahn-Hilliard equation, proposed by Bertozzi et al. [5, 4].

The Cahn-Hilliard equation is a PDE of fourth order, which is used in materials science [37, 26], image processing [16] and chemistry [56]. It was originally introduced to model phase separation in binary alloys [31, 10] and follows the evolution of a function $u(x)$ known as the order parameter, which smoothly varies between the values 0 and 1 across an interface. This represents the two different phases or especially in this work the colours black and white.

[†]Computational Methods in Systems and Control Theory, Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstr. 1, 39106 Magdeburg, Germany

[‡]bosch@mpi-magdeburg.mpg.de. Questions, comments, or corrections to this document may be directed to that email address.

[§]Oxford Department of Computer Science, University of Oxford, Wolfson Building, Parks Road, Oxford OX1 3QD, United Kingdom (dkay@cs.ox.ac.uk)

[¶]stollm@mpi-magdeburg.mpg.de

[‖]Numerical Analysis Group, Mathematical Institute, 24-29 St Giles', Oxford, OX1 3LB, United Kingdom (Andy.Wathen@maths.ox.ac.uk)

Phase separation is modelled either by a smooth free energy, e.g. double-well potentials [17] such as

$$\psi_s(u) := u^2(u-1)^2, \tag{1.1}$$

logarithmic potentials [13], or by non-smooth double obstacle potentials [6, 7] such as

$$\psi_{ns}(u) := \begin{cases} \frac{1}{2}u(1-u), & 0 \leq u \leq 1 \\ \infty, & \text{otherwise.} \end{cases} \tag{1.2}$$

The Cahn-Hilliard equation combined with an additional fidelity term acting on the known parts of an image provides a nice tool for binary image inpainting. The smoothing property of the standard Cahn-Hilliard equation allows a natural connection of the contours across damaged parts. In particular, one can control this connection subject to the size of the damaged region by the interfacial parameter ε. The additional fidelity term keeps the new image close to the original one in those parts where picture information is available.

So far, we have observed only the use of double-well potentials for Cahn-Hilliard inpainting [27, 4, 9]. We additionally consider the non-smooth one in (1.2) and will show that qualitatively better results can be obtained using the latter, see Section 7.2. Moreover, using this potential, we can control the restriction $0 \leq u \leq 1$ more precisely. All in all, we consider the efficient solution of Cahn-Hilliard inpainting with respect to preconditioning for both types of potential.

For image inpainting, efficient numerical schemes for higher-order methods are an active area of research. As discussed in [11], one of the most interesting open problems in digital inpainting is, in fact, the fast and exact digital realization. In the case of Cahn-Hilliard inpainting, Bertozzi et al. proposed a semi-implicit time discrete scheme, the so called convexity splitting [54, 46], that is guaranteed to be unconditionally stable [5]. The convexity splitting was originally introduced by Elliott and Stuart [18] and is often attributed to Eyre [21, 22]. For the space discretization, a common way is the use of a Fast-Fourier-Transform (FFT) method to compute the finite differences for the derivatives, see e.g. [27, 5, 9]. We are not focussed on FFT but we compare our proposed technique with the use of FFT methods in Section 7.1. We apply the idea of convexity splitting, propose a splitting of the fourth order operator into two second order ones and the use of finite elements for the discretization in space, as adopted in our previous work [8]. Using the non-smooth potential, we additionally have to take care of the non-smoothness and nonlinearity for which we suggest a semi-smooth Newton method [32, 52, 34, 8].

As we show in the course of this paper the solution of a linear system $Ax = b$ with A being a real, sparse matrix is at the heart of our method for both variants, using the smooth and non-smooth potential. For huge linear systems the application of direct solvers such as UMFPACK [14] becomes infeasible. As a result iterative methods have to be employed (see e.g. [28, 44] for introductions to this field). We propose the use of Krylov subspace solvers. The convergence behaviour of the iterative scheme typically depends on the conditioning of the problem and the clustering of the eigenvalues. These properties can be enhanced using preconditioning techniques $P^{-1}Ax = P^{-1}b$, where P is an invertible matrix that is easy to invert and resembles A. In this paper, we provide an efficient preconditioner P for the solution of the smooth and non-smooth modified Cahn-Hilliard equation using an effective Schur complement approximation and (algebraic) multigrid developed for elliptic systems [23, 44, 43].

The paper is organized as follows. In Section 2 we derive the time-discrete numerical Cahn-Hilliard inpainting scheme for the non-smooth potential (the smooth version with the double-well potential follows analogously). We consider in Section 3 a semi-smooth Newton (SSN) method to solve the regularized subproblems and show superlinear convergence in function space. We derive the linear systems arising from a discretization using finite elements in Section 4. In Sections 5–6, we analyse the linear systems for the double-well as well as double obstacle potential, respectively, and propose preconditioning strategies. Section 7 illustrates the competitiveness of our preconditioners for both problem setups. Section 8 summarizes our findings.

2. The modified Cahn-Hilliard equation. The Cahn-Hilliard equation is derived as the H^{-1}-gradient flow of the Ginzburg-Landau energy

$$E(u) = \int_\Omega \frac{\gamma\varepsilon}{2}|\nabla u|^2 + \frac{1}{\varepsilon}\psi(u)\,\mathrm{d}x,$$

where $\varepsilon > 0$ is proportional to the thickness of the interfacial region and $\gamma > 0$ is a constant related to the interfacial energy density. As mentioned in the introduction, $\psi(u)$ can be the double-well potential in (1.1), see e.g. [17], or the non-smooth double obstacle potential in (1.2) as in [6, 7]. Existence and uniqueness of weak solutions was shown in [19] and [6] for both problem setups.

Now, we formulate the modified version of the non-smooth Cahn-Hilliard equation for the inpainting problem. Let $f(x)$ be a given binary image in a domain Ω and $D \subset \Omega$ the inpainting domain (damaged or missing parts). Then the variable $u(x,t)$ evolves in time to become a fully inpainted version of $f(x)$ under the following system:

$$\partial_t u = -\Delta(\gamma\varepsilon\Delta u - \frac{1}{\varepsilon}(\psi_0'(u) + \mu)) + \omega(x)(f-u) \tag{2.1}$$

$$\mu \in \partial\beta_{[0,1]}(u) \tag{2.2}$$

$$0 \le u \le 1 \tag{2.3}$$

$$\frac{\partial u}{\partial n} = \frac{\partial \Delta u}{\partial n} = 0 \quad \text{on } \partial\Omega, \tag{2.4}$$

where

$$\omega(x) = \begin{cases} 0, & \text{if } x \in D \\ \omega_0, & \text{if } x \in \Omega \setminus D. \end{cases} \tag{2.5}$$

Here we have written ψ_{ns} in (1.2) via the indicator function as

$$\psi_{ns}(u) = \psi_0(u) + I_{[0,1]}(u),$$

where $\psi_0(u) = \frac{1}{2}u(1-u)$ and $\partial\beta_{[0,1]}(u)$ denotes the subdifferential of the non-smooth part $\beta_{[0,1]}(u) := \int_\Omega I_{[0,1]}(u)$ of the energy E. The system (2.1)–(2.4) is identical to the standard Cahn-Hilliard equation except for the second term on the right hand side of (2.1), which is the so-called fidelity term that keeps the solution close to the given image $f(x)$ in those areas where image information is available.

The new modified Cahn-Hilliard equation is not strictly a gradient flow. As mentioned in the beginning, the original Cahn-Hilliard equation, (2.1)–(2.4) with $\omega \equiv 0$, is indeed the H^{-1}-gradient flow of the energy $E(u)$ while the fidelity term in equation (2.1) is derived as the L^2-gradient flow of the energy

$$E_2(u) = \frac{1}{2}\int_\Omega \omega(f-u)^2 \,\mathrm{d}x. \tag{2.6}$$

But in total, the modified Cahn-Hilliard equation is neither a gradient flow in H^{-1} nor in L^2. In [5], the authors propose the application of convexity splitting to these energy functionals leading to an efficient numerical scheme which is unconditionally gradient stable.

Before we apply convexity splitting, we handle the pointwise constraints in (2.3) with a Moreau-Yosida regularization technique as in [8, 32]. Instead of the energy functional

$$E(u) = \int_\Omega \frac{\gamma\varepsilon}{2}|\nabla u|^2 + \frac{1}{\varepsilon}(\psi_0(u) + I_{[0,1]}(u)) \,\mathrm{d}x \tag{2.7}$$

we consider

$$E_1(u_\nu) = \int_\Omega \frac{\gamma\varepsilon}{2}|\nabla u_\nu|^2 + \frac{1}{\varepsilon}\psi_0(u_\nu) + \frac{1}{2\nu}|\max(0,u_\nu-1)|^2 + \frac{1}{2\nu}|\min(0,u_\nu)|^2 \,\mathrm{d}x,$$

such that we obtain

$$\partial_t u_\nu = -\Delta(\gamma\varepsilon\Delta u_\nu - \frac{1}{\varepsilon}\psi_0'(u_\nu) - \theta_\nu(u_\nu)) + \omega(x)(f-u_\nu) \tag{2.8}$$

$$\frac{\partial u_\nu}{\partial n} = \frac{\partial \Delta u_\nu}{\partial n} = 0 \quad \text{on } \partial\Omega, \tag{2.9}$$

where

$$\theta_\nu(u_\nu) := \frac{1}{\nu}\max(0,u_\nu-1) + \frac{1}{\nu}\min(0,u_\nu)$$

and $0 < \nu \ll 1$ denotes the regularization/penalty parameter.

In the convexity splitting scheme [21, 22, 18, 54, 27, 4, 9], the energy functionals are divided into two parts, respectively – a convex plus a concave one. The convex part is then treated implicitly whilst the concave part is treated explicitly. In our case, we split $E_1(u_\nu)$ as $E_1(u_\nu) = E_{11}(u_\nu) - E_{12}(u_\nu)$, where

$$E_{11}(u_\nu) = \int_\Omega \frac{\gamma\varepsilon}{2}|\nabla u_\nu|^2 + \frac{C_1}{2}|u_\nu|^2 + \frac{1}{2\nu}|\max(0,u_\nu-1)|^2 + \frac{1}{2\nu}|\min(0,u_\nu)|^2 \,\mathrm{d}x$$

$$E_{12}(u_\nu) = \int_\Omega -\frac{1}{\varepsilon}\psi_0(u_\nu) + \frac{C_1}{2}|u_\nu|^2 \,\mathrm{d}x$$

as well as $E_2(u_\nu) = E_{21}(u_\nu) - E_{22}(u_\nu)$, where

$$E_{21}(u_\nu) = \int_\Omega \frac{C_2}{2}|u_\nu|^2 \,\mathrm{d}x$$

$$E_{22}(u_\nu) = \int_\Omega -\frac{\omega}{2}(f-u_\nu)^2 + \frac{C_2}{2}|u_\nu|^2 \,\mathrm{d}x.$$

The constants C_1 and C_2 are positive and need to be chosen large enough such that the energies $E_{11}(u_\nu), E_{12}(u_\nu), E_{21}(u_\nu)$ and $E_{22}(u_\nu)$ are convex. Therefore, we have to choose $C_1 > 0$ and $C_2 > \omega_0$.

Remark 2.1. *Note that we do not need the C_1-term as $E_{12}(u_\nu)$ is already convex without it. But with respect to the preconditioning we obtain better results by keeping this term.*

For this splitting, together with the backward Euler discretization for the time derivative $\partial_t u_\nu$, the resulting time-stepping scheme is:

$$\frac{u_\nu^{(n)} - u_\nu^{(n-1)}}{\tau} = -\Delta_{H^{-1}}(E_{11}(u_\nu^{(n)}) - E_{12}(u_\nu^{(n-1)})) - \Delta_{L^2}(E_{21}(u_\nu^{(n)}) - E_{22}(u_\nu^{(n-1)})),$$

where $\Delta_{H^{-1}}$ and Δ_{L^2} represent the gradient descent with respect to the H^{-1}- and L^2-inner product, respectively. Here, $\tau > 0$ denotes the time step size and $n \in \mathbb{N}$ the time step. This translates to a numerical scheme of the form

$$\begin{aligned}\frac{u_\nu^{(n)} - u^{(n-1)}}{\tau} + \gamma\varepsilon\Delta^2 u_\nu^{(n)} - C_1\Delta u_\nu^{(n)} + C_2 u_\nu^{(n)} - \Delta\theta_\nu(u_\nu^{(n)}) \\ = \frac{1}{\varepsilon}\Delta(\psi_0'(u^{(n-1)})) + \omega(f - u^{(n-1)}) - C_1\Delta u^{(n-1)} + C_2 u^{(n-1)}. \quad (2.10)\end{aligned}$$

By operator splitting, we obtain the weak formulation of (2.10) as

$$\begin{aligned}(\frac{1}{\tau} + C_2)(u_\nu, v) + (\nabla w_\nu, \nabla v) + C_1(\nabla u_\nu, \nabla v) \\ = (\omega(f - u^{(n-1)}), v) + C_1(\nabla u^{(n-1)}, \nabla v) + (\frac{1}{\tau} + C_2)(u^{(n-1)}, v) \quad (2.11)\end{aligned}$$

$$(w_\nu, v) - \gamma\varepsilon(\nabla u_\nu, \nabla v) - (\theta_\nu(u_\nu), v) = \frac{1}{\varepsilon}(\psi_0'(u^{(n-1)}), v) \quad (2.12)$$

$\forall v \in H^1(\Omega)$, where $u_\nu = u_\nu^{(n)}$ and $w_\nu = w_\nu^{(n)}$. In the following, $(\cdot,\cdot)$ and $\langle\cdot,\cdot\rangle$ stand for the $L^2(\Omega)$-inner product and the duality pairing of $H^1(\Omega)$ and $H^1(\Omega)^*$, respectively.

Remark 2.2. *Using the double-well potential, the authors proved in [4] global existence and uniqueness of a weak solution of the modified continuous fourth-order system. Moreover, in [9] the existence of a weak solution of the stationary equation for this system is shown. The application of the convexity splitting scheme leads to the conditions $C_1 > \frac{1}{\varepsilon}$ and $C_2 > \omega_0$. In [9], the authors proved that this time-stepping scheme is unconditionally stable in the sense that the numerical solution $u^{(n)}$ is uniformly bounded on a finite time interval. Moreover, the discrete solution converges to the exact solution of the continuous fourth-order system as $\tau \to 0$. These properties make the time-discrete scheme a stable and reliable discrete approximation of the continuous fourth-order equation.*

3. Semi-smooth Newton method. We apply the function space-based algorithm motivated in [32, 8] for solving the non-smooth time-discrete Cahn-Hilliard problem. For a specified sequence $\nu \to 0$ we solve the optimality system (2.11)–(2.12), compactly written as

$$F_\nu(u_\nu, w_\nu) = (F_\nu^{(1)}(u_\nu, w_\nu), F_\nu^{(2)}(u_\nu, w_\nu)) = 0, \tag{3.1}$$

for every ν by a semi-smooth Newton algorithm. In (3.1), the components are defined by

$$\begin{aligned}\left\langle F_\nu^{(1)}(u,w), v\right\rangle = (\frac{1}{\tau} + C_2)(u,v) + (\nabla w, \nabla v) + C_1(\nabla u, \nabla v) \\ - (\omega(f - u^{(n-1)}), v) - C_1(\nabla u^{(n-1)}, \nabla v) - (\frac{1}{\tau} + C_2)(u^{(n-1)}, v)\end{aligned}$$

$$\left\langle F_\nu^{(2)}(u,w), v\right\rangle = (w,v) - \gamma\varepsilon(\nabla u, \nabla v) - (\theta_\nu(u), v) - \frac{1}{\varepsilon}(\psi_0'(u^{(n-1)}), v)$$

for all $u, w, v \in H^1(\Omega)$. Due to the presence of the max- and min-operators in the definition of θ_ν, F_ν is not Fréchet-differentiable. However, it satisfies the weaker notion of Newton differentiability, see [33, 32].

Definition 3.1. *Let X and Z be Banach spaces, $E \subset X$ an open subset. A mapping $F\colon E \to Z$ is called Newton-differentiable in $U \subset E$ if there exists a family of mappings $G\colon U \to Z$ such that*

$$\lim_{h\to 0} \frac{\|F(x+h) - F(x) - G(x+h)h\|_Z}{\|h\|_X} = 0 \quad \forall x \in U.$$

The operator G is called a Newton derivative of F on U.

For such mappings, the following convergence result for the (semi-smooth) Newton iteration

$$x^{(k+1)} = x^{(k)} - G(x^{(k)})^{-1} F(x^{(k)}), \quad k = 0, 1, \dots \tag{3.2}$$

holds true, where G is a Newton derivative of F. For its proof we refer to Theorem 1.1 in [33].

Theorem 3.2. *Let x^* be a solution of $F(x) = 0$ and suppose that $F\colon E \subset X \to Z$ is Newton-differentiable in a neighbourhood U of x^* with $\{\|G(x)^{-1}\|_{\mathcal{L}(Z,X)} : x \in U\}$ bounded. Then the sequence $\{x^{(k)}\}_{k\in\mathbb{N}}$ generated by (3.2) converges superlinearly to x^* provided that $\|x^{(0)} - x^*\|_X$ is sufficiently small.*

Lemma 3.3. *The mapping $F_\nu\colon H^1(\Omega)\times H^1(\Omega) \to H^1(\Omega)^* \times H^1(\Omega)^*$ is Newton-differentiable. Furthermore, the operator $G_\nu(u,w)$ given by*

$$\langle G_\nu(u,w)(\delta u, \delta w), (\phi, \psi)\rangle = \begin{pmatrix} (\frac{1}{\tau} + C_2)(\delta u, \phi) + (\nabla \delta w, \nabla \phi) + C_1(\nabla \delta u, \nabla \phi) \\ (\delta w, \psi) - \gamma\varepsilon(\nabla \delta u, \nabla \psi) - \frac{1}{\nu}(\chi_{\mathcal{A}(u)}\delta u, \psi) \end{pmatrix}$$

serves as a Newton-derivative for F_ν, where $\chi_{\mathcal{A}(u)}$ is the characteristic function of the set

$$\mathcal{A}(u) := \{x \in \Omega : u(x) > 1 \text{ or } u(x) < 0\}.$$

For the proof we refer to Lemma 5.3 in [32] and Proposition 4.1 in [33]. Because of the explicit treatment of the additionally term $(\omega(f - u^{(n-1)}))$ we do not have to take care of it.

Lemma 3.4. *The semi-smooth Newton method (3.2) (with F and G replaced by F_ν and G_ν) converges superlinearly to (u_ν, w_ν), the solution of (3.1), provided that $\|(u^{(0)}, w^{(0)}) - (u_\nu, w_\nu)\|_{H^1(\Omega)\times H^1(\Omega)}$ is sufficiently small.*

Proof. First of all, we show that $G_\nu(u, w)$ is invertible for all $(u, w) \in H^1(\Omega) \times H^1(\Omega)$, i.e. that a unique solution $(\delta u, \delta w) \in H^1(\Omega) \times H^1(\Omega)$ of the following linear system exists

$$(\frac{1}{\tau} + C_2)(\delta u, \phi) + (\nabla \delta w, \nabla \phi) + C_1(\nabla \delta u, \nabla \phi) = 0 \quad \forall \phi \in H^1(\Omega) \tag{3.3}$$

$$(\delta w, \phi) - \gamma\varepsilon(\nabla \delta u, \nabla \phi) - \frac{1}{\nu}(\chi_{\mathcal{A}(u)}\delta u, \phi) = 0 \quad \forall \phi \in H^1(\Omega). \tag{3.4}$$

Multiplying (3.4) with $-(\frac{1}{\tau} + C_2)$, testing (3.3) with δw, (3.4) with δu and adding the two equations, we obtain

$$0 = \|\nabla \delta w\|^2 + C_1(\nabla \delta u, \nabla \delta w) + \gamma\varepsilon(\frac{1}{\tau} + C_2)\|\nabla \delta u\|^2 + \frac{1}{\nu}(\frac{1}{\tau} + C_2)(\chi_{\mathcal{A}(u)}\delta u, \delta u). \tag{3.5}$$

Applying Young's inequality with $\alpha > 0$ to

$$(\nabla \delta u, \nabla \delta w) \geq -\alpha \|\nabla \delta u\|^2 - \frac{1}{4\alpha}\|\nabla \delta w\|^2 \tag{3.6}$$

yields in (3.5)

$$0 \geq (1 - \frac{C_1}{4\alpha})\|\nabla \delta w\|^2 + (\gamma\varepsilon(\frac{1}{\tau} + C_2) - C_1\alpha)\|\nabla \delta u\|^2 + \frac{1}{\nu}(\frac{1}{\tau} + C_2)(\chi_{\mathcal{A}(u)}\delta u, \delta u) \geq 0,$$

where we choose $\alpha \in (0, \frac{\gamma\varepsilon(\frac{1}{\tau}+C_2)}{C_1})$. We obtain $\|\nabla \delta w\|^2 = \|\nabla \delta u\|^2 = 0$, which implies that δu and δw are constant. Then (3.3) gives $\delta u = 0$ and finally (3.4) results in $\delta w = 0$. We have proved that $G_\nu(u, w)$ is invertible for all $(u, w) \in H^1(\Omega) \times H^1(\Omega)$. Now, we show the boundedness of $\left\|G_\nu^{-1}(u, w)\right\|_{\mathcal{L}((H^1(\Omega)^*)^2,(H^1(\Omega))^2)}$ for all $(u, w) \in H^1(\Omega) \times H^1(\Omega)$.

$G_\nu(u, w)$ is invertible for all $(u, w) \in H^1(\Omega) \times H^1(\Omega)$, i.e. for given $(y_1, y_2) \in H^1(\Omega)^* \times H^1(\Omega)^*$, there exists a unique pair $(\delta u, \delta w) \in H^1(\Omega) \times H^1(\Omega)$, such that

$$(\frac{1}{\tau} + C_2)(\delta u, \phi) + (\nabla \delta w, \nabla \phi) + C_1(\nabla \delta u, \nabla \phi) = \langle y_1, \phi\rangle \quad \forall \phi \in H^1(\Omega) \tag{3.7}$$

$$(\delta w, \psi) - \gamma\varepsilon(\nabla \delta u, \nabla \psi) - \frac{1}{\nu}(\chi_{\mathcal{A}(u)}\delta u, \psi) = \langle y_1, \phi\rangle \quad \forall \psi \in H^1(\Omega) \tag{3.8}$$

is satisfied. Taking $(\phi, \psi) = (\delta w, \delta u)$ in (3.7)–(3.8) and adding the two equations, we obtain

$$\gamma\varepsilon(\frac{1}{\tau} + C_2)\|\nabla \delta u\|^2 + \|\nabla \delta w\|^2 = \langle y_1, \delta w\rangle + \langle y_2, \delta u\rangle - C_1(\nabla \delta u, \nabla \delta w) - \frac{1}{\nu}\underbrace{(\chi_{\mathcal{A}(u)}\delta u, \delta u)}_{\geq 0}$$

$$\leq \langle y_1, \delta w\rangle + \langle y_2, \delta u\rangle - C_1(\nabla \delta u, \nabla \delta w).$$

Using (3.6) with $\alpha \in (0, \frac{\gamma\varepsilon(\frac{1}{\tau}+C_2)}{C_1})$, we get

$$\begin{aligned}\underbrace{\left(\gamma\varepsilon(\frac{1}{\tau}+C_2)-C_1\alpha\right)}_{\geq 0}\|\nabla\delta u\|^2+\underbrace{\left(1-\frac{C_1}{4\alpha}\right)}_{\geq 0}\|\nabla\delta w\|^2 \\ \leq \langle y_1,\delta w\rangle+\langle y_2,\delta u\rangle \\ \leq \frac{1}{4\beta_1}\left(\|\delta w\|^2+\|\nabla\delta w\|^2\right)+\frac{1}{4\beta_2}\left(\|\delta u\|^2+\|\nabla\delta u\|^2\right) \\ +C\left(\|y_1\|^2_{H^1(\Omega)^*}+\|y_2\|^2_{H^1(\Omega)^*}\right),\end{aligned} \tag{3.9}$$

where we used Cauchy-Schwarz's and Young's inequality for the last result and the fact that the constant $C>0$ does not depend on δu or δw. Taking $(\phi,\psi)=(1,1)$ in (3.7)–(3.8), we get

$$(\delta u,1)=\frac{1}{\frac{1}{\tau}+C_2}\langle y_1,1\rangle \tag{3.10}$$

$$(\delta w,1)=\frac{1}{\nu}(\chi_{\mathcal{A}(u)}\delta u,1)-\frac{1}{\frac{1}{\tau}+C_2}\langle y_2,1\rangle. \tag{3.11}$$

From (3.10)–(3.11) and (3.9) we obtain by using Poincaré-Friedrichs and again Young's inequality that

$$\|(\delta u,\delta w)\|_{H^1(\Omega)\times H^1(\Omega)}\leq C\left(\|y_1\|_{H^1(\Omega)^*}+\|y_2\|_{H^1(\Omega)^*}\right).$$

For $\max\left(\|y_1\|_{H^1(\Omega)^*},\|y_2\|_{H^1(\Omega)^*}\right)\leq\beta$ for some constant $\beta>0$, we consequently have

$$\left\|G_\nu^{-1}(u,w)\right\|_{\mathcal{L}((H^1(\Omega)^*)^2,(H^1(\Omega))^2)}\leq\hat{C}\quad\forall(u,w)\in H^1(\Omega)\times H^1(\Omega)$$

with some constant $\hat{C}>0$ possibly depending on $\varepsilon,\gamma,\tau,\nu,\beta_1,\beta_2,\beta_3$ or β, but not on u or w. Thus F_ν with associated Newton derivative G_ν fulfills the conditions of Theorem 3.2, which completes the proof. ■

We have shown local superlinear convergence in function space of the semi-smooth Newton method for solving the regularized subproblem (3.1). We now want to discretize the problem in space and then discuss its efficient solution.

4. Finite-element approximation. We discretize (3.1) by finite elements [50]. In the following, we assume for simplicity that Ω is a polyhedral domain. Generalizations to curved domains are possible using boundary finite elements with curved faces. Let $\{\mathcal{R}_h\}_{h>0}$ be a triangulation of Ω into disjoint open rectangular elements. The use of rectangles is motivated by performing the implementation with deal.II [1]. Furthermore, we define $\mathcal{R}_h$ to have maximal element size $h:=\max_{R\in\mathcal{R}_h}\{\mathrm{diam}(R)\}$ and we set J_h to be the set of nodes of $\mathcal{R}_h$ and $p_j\in J_h$ to be the coordinates of these nodes. We approximate the infinite dimensional space $H^1(\Omega)$ by the finite dimensional space

$$S_h:=\{\phi\in C^0(\bar{\Omega}):\phi\,|_R\in Q_1(R)\ \ \forall R\in\mathcal{R}_h\}\subset H^1(\Omega),$$

of continuous, piecewise multi-linear functions, where e.g. for dimension 2 we have $Q_1 = \operatorname{span}\{x^{\alpha_i} y^{\alpha_i} : \alpha_i \in \{0,1\},\ i = 1,2\}$ are bilinear. To each $p_j \in J_h$ we associate the nodal basis function $\varphi_j \in S_h$ with the property $\varphi_j(p_i) = \delta_{ij}$, $i, j = 1, \ldots, N$.

The discretized version of the penalized problem (3.1) consists in finding $(u_{\nu,h}, w_{\nu,h}) \in S_h \times S_h$ such that

$$\left\langle F^{(1)}_{\nu,h}(u_{\nu,h}, w_{\nu,h}), v_h \right\rangle = 0 \quad \forall v_h \in S_h \tag{4.1}$$

$$\left\langle F^{(2)}_{\nu,h}(u_{\nu,h}, w_{\nu,h}), v_h \right\rangle = 0 \quad \forall v_h \in S_h, \tag{4.2}$$

where the components are

$$\begin{aligned}\left\langle F^{(1)}_{\nu,h}(u_{\nu,h}, w_{\nu,h}), v_h \right\rangle &= (\frac{1}{\tau} + C_2)(u_{\nu,h}, v_h)_h + (\nabla w_{\nu,h}, \nabla v_h) + C_1(\nabla u_{\nu,h}, \nabla v_h) \\ &\quad - (\omega(f_h - u_h^{(n-1)}), v_h)_h - C_1(\nabla u_h^{(n-1)}, \nabla v_h) - (\frac{1}{\tau} + C_2)(u_h^{(n-1)}, v_h)_h\end{aligned}$$

$$\left\langle F^{(2)}_{\nu,h}(u_{\nu,h}, w_{\nu,h}), v_h \right\rangle = (w_{\nu,h}, v_h)_h - \gamma\varepsilon(\nabla u_{\nu,h}, \nabla v_h) - (\theta_\nu(u_{\nu,h}), v_h)_h - \frac{1}{\varepsilon}(\psi_0'(u_h^{(n-1)}), v_h)_h.$$

The semi-inner product $(\cdot,\cdot)_h$ on $C_0(\bar\Omega)$ is defined by

$$(g_1, g_2)_h := \int_\Omega \pi_h(g_1(x) g_2(x))\, \mathrm{d}x = \sum_{i=1}^N (1, \varphi_i) g_1(p_i) g_2(p_i) \quad \forall g_1, g_2 \in C_0(\bar\Omega),$$

where $\pi_h \colon C_0(\bar\Omega) \to S_h$ is the Lagrange interpolation operator. Within our finite-element framework, for a given $(u_h, w_h) \in S_h \times S_h$, every step of the semi-smooth Newton method for solving (4.1)–(4.2) requires to compute $(\delta u_h, \delta w_h) \in S_h \times S_h$ satisfying

$$(\frac{1}{\tau} + C_2)(\delta u_h, v_h)_h + (\nabla \delta w_h, \nabla v_h) + C_1(\nabla \delta u_h, \nabla v_h) = -F^{(1)}_{\nu,h}(u_h, w_h) \tag{4.3}$$

$$\gamma\varepsilon(\delta w_h, v_h)_h - \gamma\varepsilon(\nabla \delta u_h, \nabla v_h) - \frac{1}{\nu}(\chi^h_{\mathcal{A}(u_h)} \delta u_h, v_h)_h = -F^{(2)}_{\nu,h}(u_h, w_h) \tag{4.4}$$

for all $v_h \in S_h$, where $\chi^h_{\mathcal{A}(u_h)} := \sum_{i=1}^N \chi^h_{\mathcal{A}(u_h)}(p_i)\varphi_i$ with $\chi^h_{\mathcal{A}(u_h)}(p_i) = 0$ if $0 \le u_h(p_i) \le 1$ and $\chi^h_{\mathcal{A}(u_h)}(p_i) = 1$ otherwise. In matrix form, the linear system (4.3)–(4.4) reads

$$\begin{bmatrix} M & -\gamma\varepsilon K - \frac{1}{\nu} G_A M G_A \\ K & (\frac{1}{\tau} + C_2) M + C_1 K \end{bmatrix} \begin{bmatrix} w^{(k+1)} \\ u^{(k+1)} \end{bmatrix} = \begin{bmatrix} \frac{1}{\varepsilon} M \psi_0'(u^{\mathrm{old}}) - \frac{1}{\nu} G_{A^+} M G_{A^+} \bar u \\ C_1 K u^{\mathrm{old}} + (\frac{1}{\tau} + C_2) M u^{\mathrm{old}} + \omega_0 G_D M G_D (f - u^{\mathrm{old}}) \end{bmatrix}, \tag{4.5}$$

where $u^{(k+1)}, w^{(k+1)}, u^{\mathrm{old}}, \bar u = (1, \ldots, 1)^T \in \mathbb{R}^N$ (u^{old} is the solution from the previous time step) and for $i = 1, \ldots, N$

$$G_A = G_A(u^{(k)}) = \operatorname{diag} \begin{pmatrix} 1, & \text{if } u^{(k)}(p_i) < 0 \text{ or } u^{(k)}(p_i) > 1 \\ 0, & \text{otherwise} \end{pmatrix}$$

$$G_{A^+} = G_{A^+}(u^{(k)}) = \operatorname{diag} \begin{pmatrix} 1, & \text{if } u^{(k)}(p_i) > 1 \\ 0, & \text{otherwise} \end{pmatrix}$$

describe the matrix representations of the generalized derivatives. The lumped mass matrix and the stiffness matrix are defined as

$$M_{i,j} := (\varphi_i, \varphi_j)_h \quad \text{and} \quad K_{i,j} := (\nabla\varphi_i, \nabla\varphi_j) \quad \forall i,j = 1,\dots,N,$$

respectively, and

$$G_D = \operatorname{diag}\begin{pmatrix} 1, & \text{if } x \in \Omega \setminus D \\ 0, & \text{if } x \in D \end{pmatrix}.$$

Note that M is a diagonal symmetric positive definite matrix and K is symmetric and positive semi-definite.

Remark 4.1. *Using the smooth double-well potential, the linear system reads*

$$\begin{bmatrix} M & \gamma\varepsilon K \\ \gamma\varepsilon K & -\gamma\varepsilon((\frac{1}{\tau}+C_2)M + C_1K) \end{bmatrix} \begin{bmatrix} w \\ -u \end{bmatrix} = \begin{bmatrix} \frac{1}{\varepsilon}M\psi_s'(u^{\text{old}}) \\ \gamma\varepsilon(C_1Ku^{\text{old}} + (\frac{1}{\tau}+C_2)Mu^{\text{old}} + \omega_0 G_D M G_D(f-u^{\text{old}})) \end{bmatrix}. \quad (4.6)$$

5. Preconditioning for the smooth modified Cahn-Hilliard inpainting. For the system matrix in (4.6)

$$A = \begin{bmatrix} M & \gamma\varepsilon K \\ \gamma\varepsilon K & -\gamma\varepsilon((\frac{1}{\tau}+C_2)M + C_1K) \end{bmatrix}, \quad (5.1)$$

we propose the block-triangular preconditioner

$$P = \begin{bmatrix} M & 0 \\ \gamma\varepsilon K & -S \end{bmatrix}.$$

Here, S is the Schur complement $-\gamma\varepsilon((\frac{1}{\tau}+C_2)M + C_1K) - \gamma^2\varepsilon^2 K M^{-1} K$. Preconditioners of this form have proven to perform well based on the simple observation that the preconditioned matrix $P^{-1}A$ has a small number of distinct eigenvalues [36]. It is obvious that we never want to form the Schur complement explicitly as the storage requirements for realistic scenarios would not be feasible. In order to represent most of the terms in S, we suggest the following approximation

$$\begin{aligned} \hat{S} &= -\hat{S}_1 M^{-1} \hat{S}_1 \\ &= -\left(\sqrt{\gamma\varepsilon(C_2+\frac{1}{\tau})}M + \gamma\varepsilon K\right) M^{-1} \left(\sqrt{\gamma\varepsilon(C_2+\frac{1}{\tau})} + \gamma\varepsilon K\right), \end{aligned}$$

where we use an algebraic multigrid (AMG) preconditioner for the approximation of the inverse of $\hat{S}_1$. Algebraic multigrid methods typically exhibit geometric-like properties for positive definite elliptic type operators such as $\hat{S}_1$, but use only algebraic information. This has the advantage that AMG can work well even for complicated geometries and meshes. We refer to [43, 23] for more information on AMG. We also want to emphasize that geometric multigrid

(GMG) (see e.g. [55, 29]) approximations are also well suited to approximate $\hat{S}_1$ provided they can be readily applied. The suitability of $\hat{S}$ as a preconditioner for S is guaranteed by the following Lemma.

Lemma 5.1. *The eigenvalues of $\hat{S}^{-1}S$ are contained within the following interval:*

$$\lambda(\hat{S}^{-1}S) \in \left[\frac{1}{2}, 1 + \frac{C_1}{2\sqrt{\gamma\varepsilon(C_2 + \frac{1}{\tau})}}\right].$$

Proof. As both S and $\hat{S}$ are symmetric matrices, we may prove the result using a Rayleigh quotient argument. We write that

$$\begin{aligned}\frac{v^T S v}{v^T \hat{S} v} &= \frac{\gamma^2\varepsilon^2 v^T K M^{-1} K v + \gamma\varepsilon(C_2 + \frac{1}{\tau})v^T M v + \gamma\varepsilon C_1 v^T K v}{\gamma^2\varepsilon^2 v^T K M^{-1} K v + \gamma\varepsilon(C_2 + \frac{1}{\tau})v^T M v + 2\gamma\varepsilon\sqrt{\gamma\varepsilon(C_2 + \frac{1}{\tau})}v^T K v} \\ &= \frac{1 + \frac{\gamma\varepsilon C_1 v^T K v}{\gamma^2\varepsilon^2 v^T K M^{-1} K v + \gamma\varepsilon(C_2+\frac{1}{\tau})v^T M v}}{1 + \frac{2\gamma\varepsilon\sqrt{\gamma\varepsilon(C_2+\frac{1}{\tau})}v^T K v}{\gamma^2\varepsilon^2 v^T K M^{-1} K v + \gamma\varepsilon(C_2+\frac{1}{\tau})v^T M v}}.\end{aligned}$$

The quantity $\frac{2\gamma\varepsilon\sqrt{\gamma\varepsilon(C_2+\frac{1}{\tau})}v^T K v}{\gamma^2\varepsilon^2 v^T K M^{-1} K v + \gamma\varepsilon(C_2+\frac{1}{\tau})v^T M v}$ arising in the denominator may be written as $\frac{2a^T b}{a^T a + b^T b}$, $a = \sqrt{\gamma\varepsilon(C_2 + \frac{1}{\tau})}M^{\frac{1}{2}}v$ and $b = \gamma\varepsilon M^{-\frac{1}{2}}v$. Now, as in the work of Pearson et al. [38, 40, 39], since $a^T a > 0$ (because of the positive definiteness of M), we may use that $(a-b)^T(a-b)$ is non-negative to bound this quantity above by one. On the other hand, the quantity is clearly bounded below by zero. The quantity $\frac{\gamma\varepsilon C_1 v^T K v}{\gamma^2\varepsilon^2 v^T K M^{-1} K v + \gamma\varepsilon(C_2+\frac{1}{\tau})v^T M v}$ arising in the numerator may be written as $\frac{2C_1 a^T b}{2\sqrt{\gamma\varepsilon(C_2+\frac{1}{\tau})}(a^T a + b^T b)}$, so it is bounded above by $\frac{C_1}{2\sqrt{\gamma\varepsilon(C_2+\frac{1}{\tau})}}$ and below by zero. Putting all our working together, we have proved that

$$\lambda(\hat{S}^{-1}S) \in \left[\frac{1}{2}, 1 + \frac{C_1}{2\sqrt{\gamma\varepsilon(C_2 + \frac{1}{\tau})}}\right].$$

■

6. Preconditioning for non-smooth modified Cahn-Hilliard inpainting. For the system matrix in (4.5), which we write as

$$A = \begin{bmatrix} M & -L \\ K & (\frac{1}{\tau} + C_2)M + C_1 K \end{bmatrix}, \tag{6.1}$$

we propose as in the previous section the block-triangular preconditioner

$$P = \begin{bmatrix} M & 0 \\ K & -S \end{bmatrix}.$$

Here, S is the Schur complement $(\frac{1}{\tau}+C_2)M+C_1K+KM^{-1}L$. For S we suggest the following approximation

$$\hat{S} = \hat{S}_2 M^{-1} \hat{S}_3 = \left(\sqrt{\frac{1}{\tau}+C_2}M + K\right) M^{-1} \left(\sqrt{\frac{1}{\tau}+C_2}M + L\right),$$

where we use as in the previous section an algebraic multigrid (AMG) preconditioner for the approximation of the inverse of $\hat{S}_2$ and $\hat{S}_3$. Because we are dealing with nonsymmetry here, it is hard to prove eigenvalue bounds as in the symmetric case from the previous Section. In the following, we want to illustrate the performance of $\hat{S}^{-1}S$. We therefore consider the eigenvalue problem

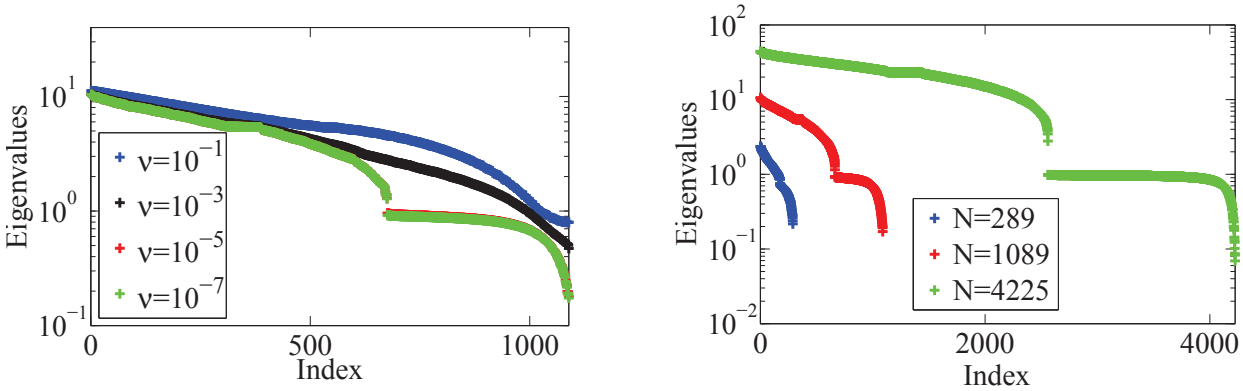
$$Sv = \lambda \hat{S} v$$

via Matlab® and analyse the robustness of the Schur complement approximation with respect to the penalty parameter c and the mesh parameter h. For the former we vary c by the sequence $\nu \in \{10^{-1}, 10^{-3}, 10^{-5}, 10^{-7}\}$ and for the latter we consider $h \in \{2^{-4}, 2^{-5}, 2^{-6}\}$. Moreover, we choose $\tau = 1$, $\gamma = 1$, $C_1 > \frac{1}{\varepsilon}$ and $C_2 > \omega_0$. In the following we present some eigenvalue distributions for different values of ε.

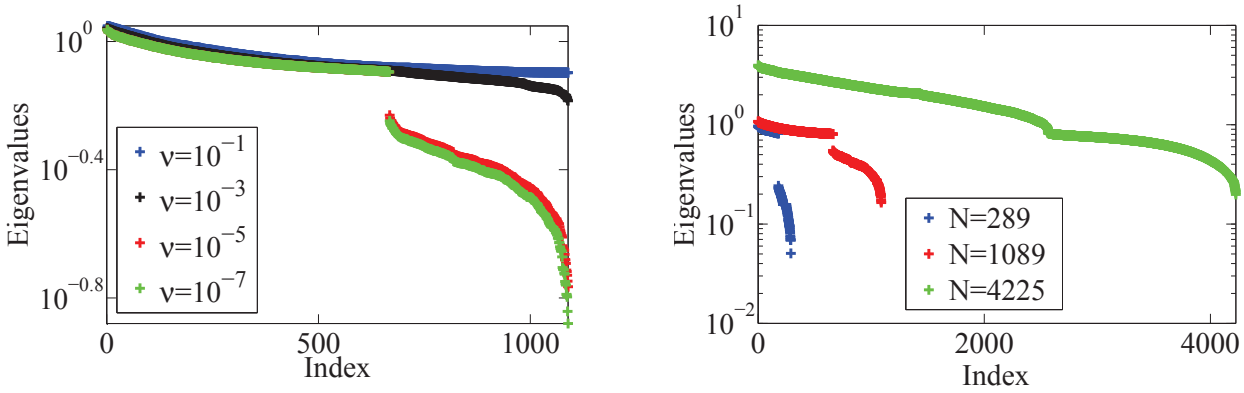
We start with the rather large choice $\varepsilon = 0.8$ which is typical when having a large inpainting gap, see Figure 6.1(a). For this case we only obtain real eigenvalues and a few clusters whose number stays almost constant for varying ν and h. Increasing the parameter C_2 leads to a better clustering because the eigenvalues move closer together, see Figure 6.1(b). Next, we decrease ε to 0.1 in Figure 6.1(c) and get complex eigenvalues. The imagnary parts are clustered in a circle centred on 0.6 and of radius about 0.1 for varying ν and h. The real parts are considered separately in Figure 6.1(d). Comparing to the eigenvalues in Figure 6.1(b), the values here look nearly the same but with a hint to a little improvement. Decreasing ε to 0.01 also leads to complex eigenvalues which are shown in Figure 6.2(a) and again, considering the real parts in Figure 6.2(b) the clustering has improved. The imaginary parts are clustered in a circle centred on 0.7 and of radius about 0.2 and the real parts are mostly distributed about $10^{-0.1}$ for varying ν and about 1 for varying h. In the last example we switch to $\varepsilon = 0.001$. Figure 6.2(c) shows for varying the parameter ν real eigenvalues around $1, 3$ and 5. For varying h we obtain complex eigenvalues whose imaginary parts are clustered like in the previous example. Considering the real parts in Figure 6.2(d), we recognize again the small number of clusters for varying h.

We have seen that we have obtained different eigenvalue distributions for several examples. However, we also observed only a small number of eigenvalue clusters, which justifies our choice of $\hat{S}$.

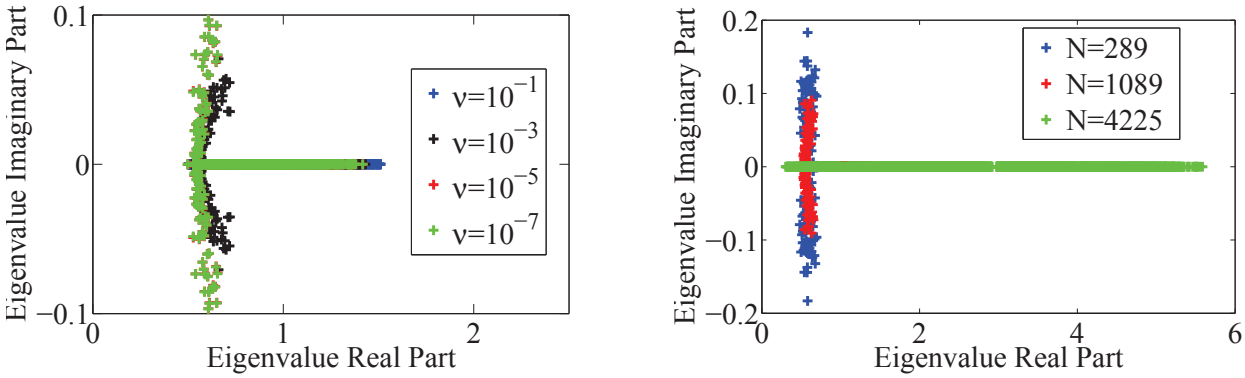
For the application of P it remains to analyse the computational cost of solving a linear system with the block M. The aim of a preconditioner is to resemble the original matrix by also being easy to invert. As M is a diagonal matrix it is a simple task to apply the inversion.



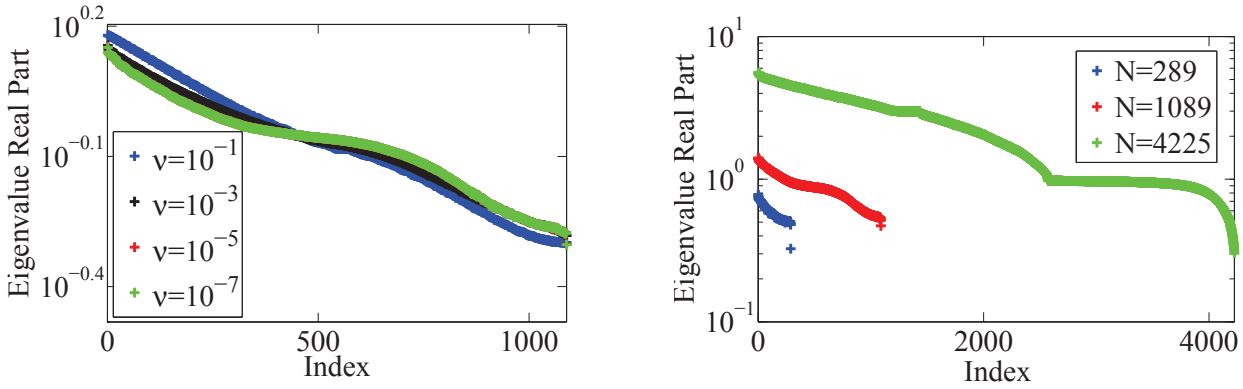
(a) $\varepsilon = 0.8$, $C_1 = \frac{3}{\varepsilon}$, $C_2 = 3 \cdot 10^5$.



(b) $\varepsilon = 0.8$, $C_1 = \frac{3}{\varepsilon}$, $C_2 = 3 \cdot 10^7$.

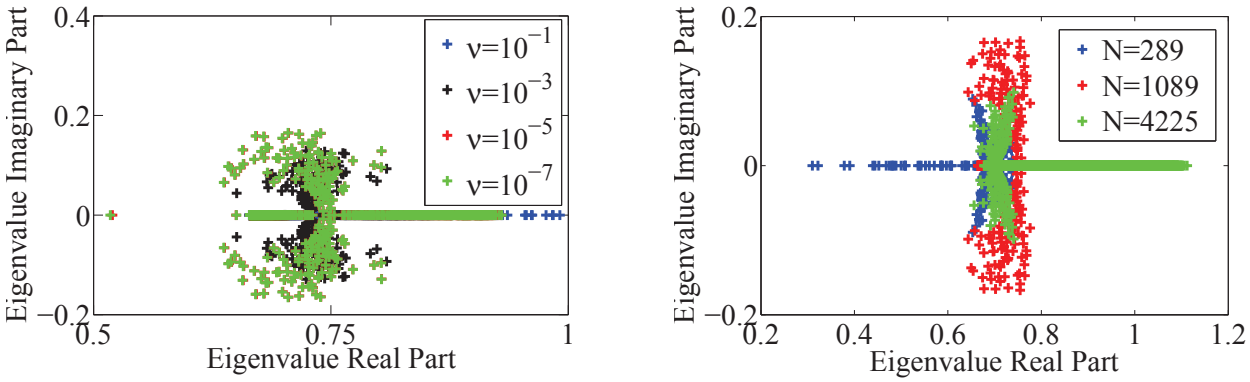


(c) $\varepsilon = 0.1$, $C_1 = \frac{3}{\varepsilon}$, $C_2 = 3 \cdot 10^5$.

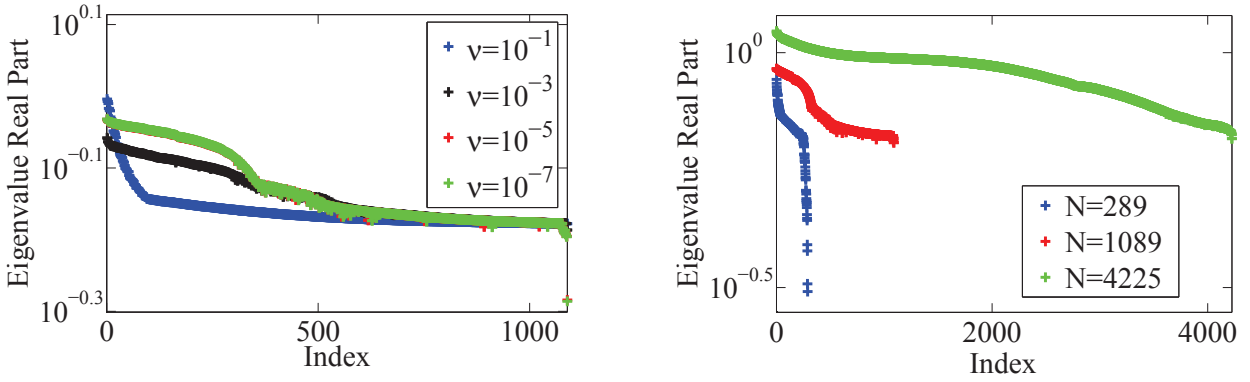


(d) $\varepsilon = 0.1$, $C_1 = \frac{3}{\varepsilon}$, $C_2 = 3 \cdot 10^5$.

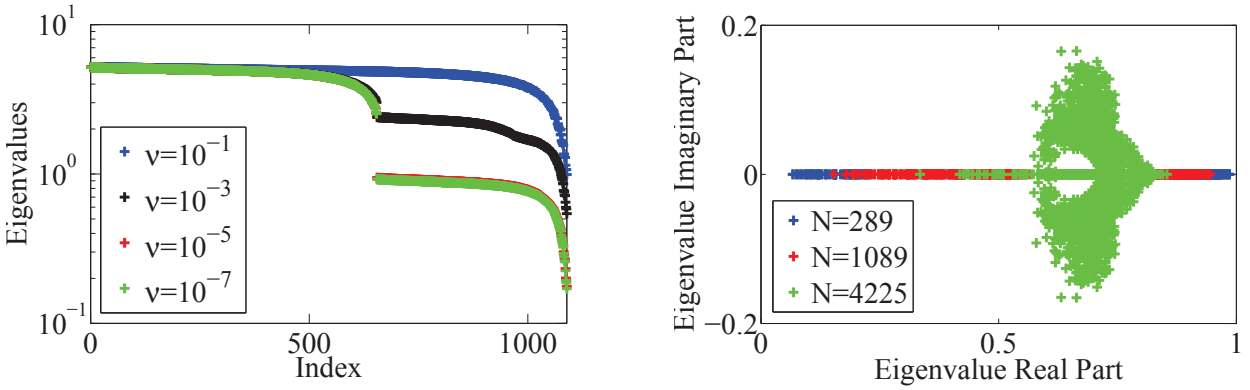
Figure 6.1: Eigenvalues for the Schur complement approximation.



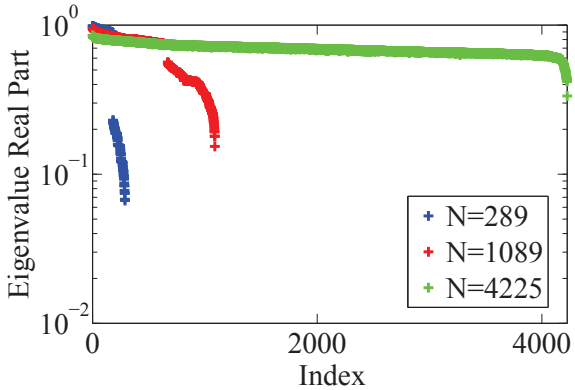
(a) $\varepsilon = 0.01$, $C_1 = \frac{3}{\varepsilon}$, $C_2 = 3 \cdot 10^5$.



(b) $\varepsilon = 0.01$, $C_1 = \frac{3}{\varepsilon}$, $C_2 = 3 \cdot 10^5$.



(c) $\varepsilon = 0.001$, $C_1 = \frac{3}{\varepsilon}$, $C_2 = 3 \cdot 10^5$.



(d) $\varepsilon = 0.001$, $C_1 = \frac{3}{\varepsilon}$, $C_2 = 3 \cdot 10^5$.

Figure 6.2: Eigenvalues for the Schur complement approximation.

7. Numerical results. In this section we show results for the modified Cahn-Hilliard equation. The preconditioners we present can be embedded into various Krylov subspace solvers. For the non-symmetric matrix (6.1) we propose the use of a non-symmetric short-term recurrence method, namely BiCG [24], but note that also other solvers such as QMR [25], BiCGSTAB [53] or GMRES [45] can be used with this preconditioner. For the numerical results presented in this section we set the BiCG tolerance to be 10^{-7} for the preconditioned residual in all examples. For the multilevel approximations we choose Trilinos AMG approximations [30]. For one application of the preconditioner we take 10 steps of a Chebyshev smoother and two V-cycles. The discretization is performed with the finite element package deal.II [1], which allows the use of the Trilinos library. All numerical experiments listed here are generated with finite elements on rectangles. For the semi-smooth Newton method we use the stopping criterion in [32, 8], given by

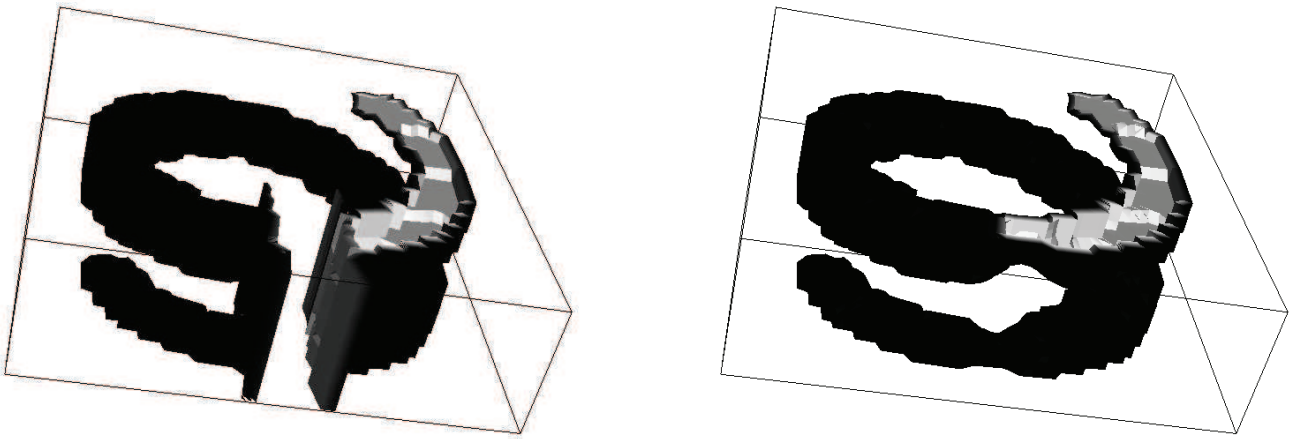
$$\|F_\nu(u_h^{(k)}, w_h^{(k)})\|_2 \leq \epsilon_{\text{rel}} \|F_\nu(u_h^{(0)}, w_h^{(0)})\|_2 + \epsilon_{\text{abs}}, \quad k = 1, \ldots, k_{\max},$$

where we set $k_{\max} = 100$, $\epsilon_{\text{rel}} = 10^{-12}$ and $\epsilon_{\text{abs}} = 10^{-6}$ in all examples. For the handling of the parameter ν we follow [32, 8] and solve (4.1)–(4.2) for the sequence $\nu_1 = 10^{-1} \geq \nu_2 = 10^{-2} \geq \ldots \geq \nu_{\max}$, where we initialize each Newton method by the approximate solution of the previously used ν value. After a few time steps, we fix $\nu = \nu_{\max}$. This is because the initial solutions at the beginning might not be a good starting point for the semi-smooth Newton methods. In all examples we denote the inpainting region in gray. For some pictures, we use the two-scale approach described in [27, 5] which is successful in connecting edges across large inpainting regions. There, the first simulation is run close to steady state for a large choice of ε. We use as stopping criterion

$$\|u_h^{(n)} - u_h^{(n-1)}\|_2 \leq \epsilon,$$

where we set $\epsilon = 10^{-2}$. Then, the obtained approximate solution is used as initial state for the second run with a small ε to sharpen the edges. For this, we use the same stopping criterion as before but with $\epsilon = 10^{-4}$. Moreover, we set $\tau = 1$ and $C_2 = 3\omega_0$ in all examples (ω_0 defined in 2.5 is the large parameter making sure that the result is close enough to the original picture).

7.1. Comparison of our model and the use of Fourier transforms. Solving ODEs or PDEs via spectral methods is an important tool in image processing. As the FFT basis functions are eigenvectors of the difference operators which form the discrete Laplacian, FFT methods rapidly solve divergence equations on simple domains. For the smooth modified Cahn-Hilliard equation using the double-well potential, Bertozzi et al. [27, 5, 9] proposed a two-dimensional FFT method and achieved fast inpainting. In fact, this method is hard to beat but with respect to more complex inpainting-type problems, spectral methods on complicated domains are difficult. Figure 7.1 shows a 3D example on a cube with a cylindrical hole inside, see the mesh in Figure 7.2, for which our method can be applied without problems. The usage of finite elements typically allows us to compute missing information on arbitrary domains.



(a) $n = 0$ (b) $n = 880$

Figure 7.1: Smooth modified Cahn-Hilliard evolution for a 3D spiral helix on a complicated domain.

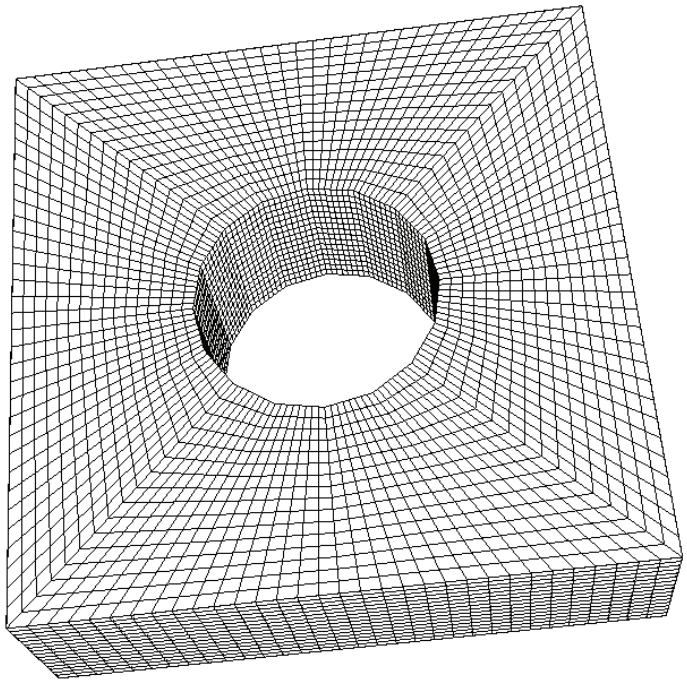


Figure 7.2: Mesh of a cube with a cylindrical hole.

Extending the idea of using an FFT based solution scheme to the non-smooth problem is a challenge. An efficient FFT based implementation employing the non-smooth potential would typically suffer from the non-constant coefficient matrix coming from the term $G_A M G_A$ that

originates in the discretization of the max- and min-Terms, see Eyre [22] for an application with a smooth potential for the standard Cahn-Hilliard equation. This poses a challenge as the optimal approximation of the non-constant term needs to be of good quality to achieve small iteration numbers of the linear solver. In general it holds that for spectral methods the smoother the function the faster the convergence. Figure 7.3 shows how the iteration numbers increase with the non-smoothness obtained by varying the penalty parameter ν. The decrease of iteration numbers after around 100 time steps is a result of the ε-jump. As another motivation to focus on a discretization via finite elements for the non-smooth potential function.

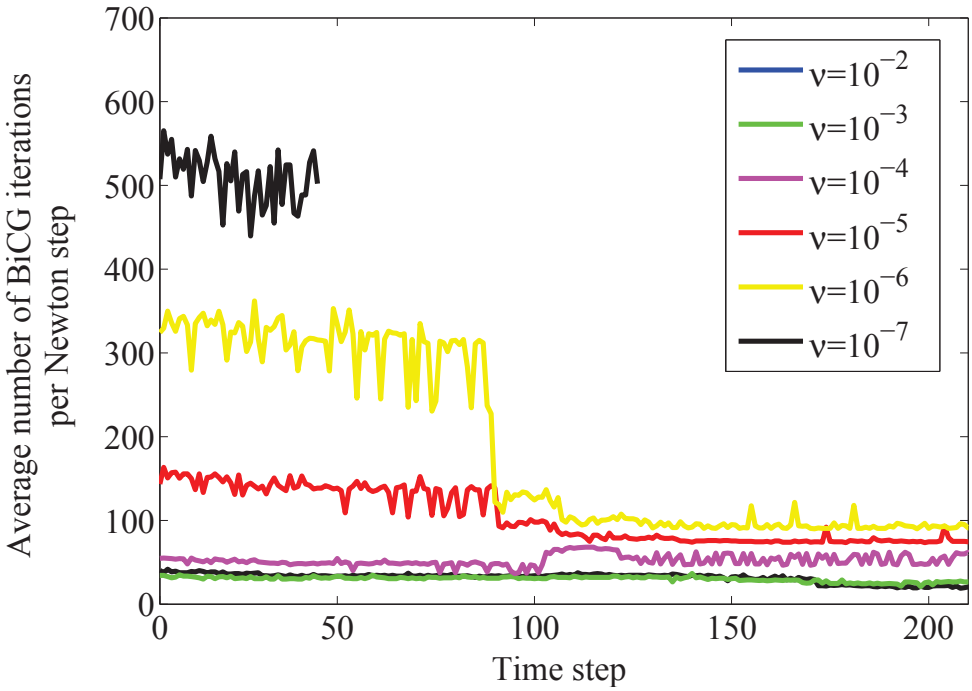
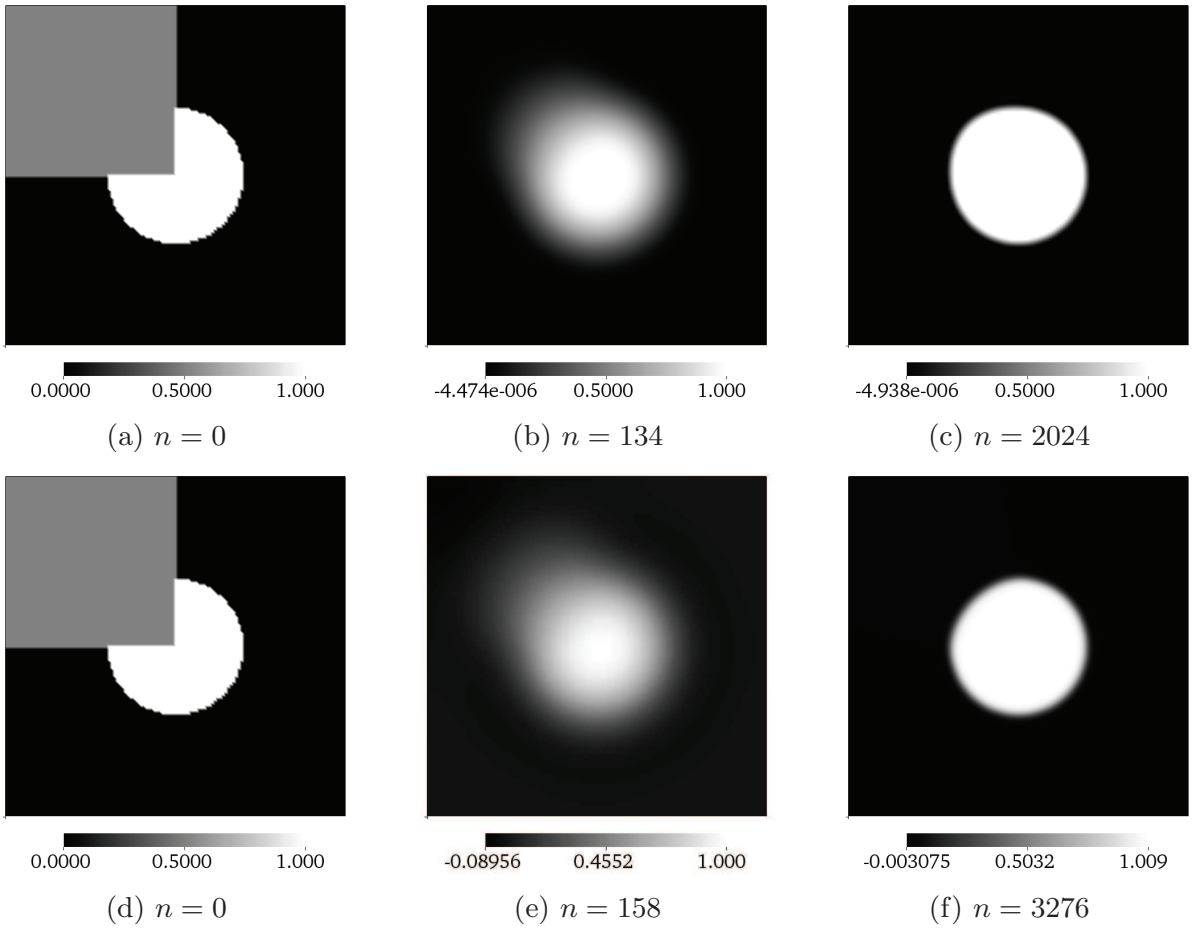


Figure 7.3: Iteration numbers for the non-smooth model using FFT for the application of the preconditioner.

7.2. Comparison of the smooth and non-smooth model. We want to compare the performance of the smooth and non-smooth Cahn-Hilliard inpainting model on the damaged circle in Figure 7.4. Because of the large inpainting domain we use the two-scale approach for the parameter ε. At first, the modified Cahn-Hilliard equations are run for the large interface parameter $\varepsilon = 0.8$. Figure 7.4(b) and 7.4(e) show the successful connection of the edges, respectively. Then, we switch to $\varepsilon = 0.01$ to sharpen the edges. The results of both models are shown in Figure 7.4(c) and 7.4(f), respectively. The parameters are set as $h = 2^{-7}$, $\omega_0 = 100000$ and $C_1 = 300$. This is a well suited example because of the smoothing property of the Cahn-Hilliard equation. Comparing both results, the rounder circle as well as intenser colors (which can be seen by comparing the minimum and maximum value of the colors) are obtained in the non-smooth model. This verifies our preference for using a non-smooth potential

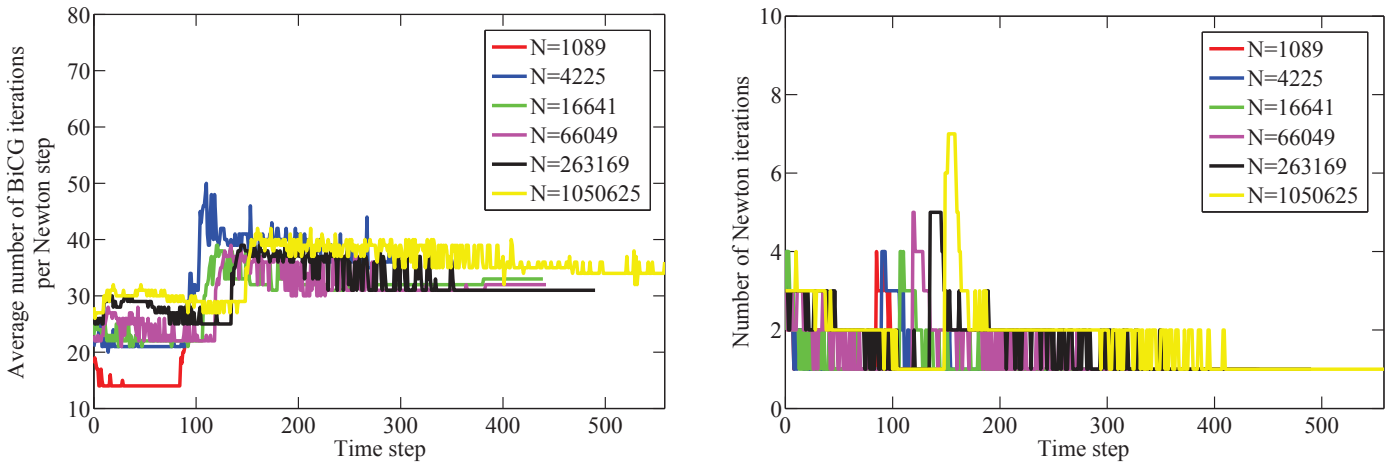


(a) $n = 0$ (b) $n = 134$ (c) $n = 2024$

(d) $n = 0$ (e) $n = 158$ (f) $n = 3276$

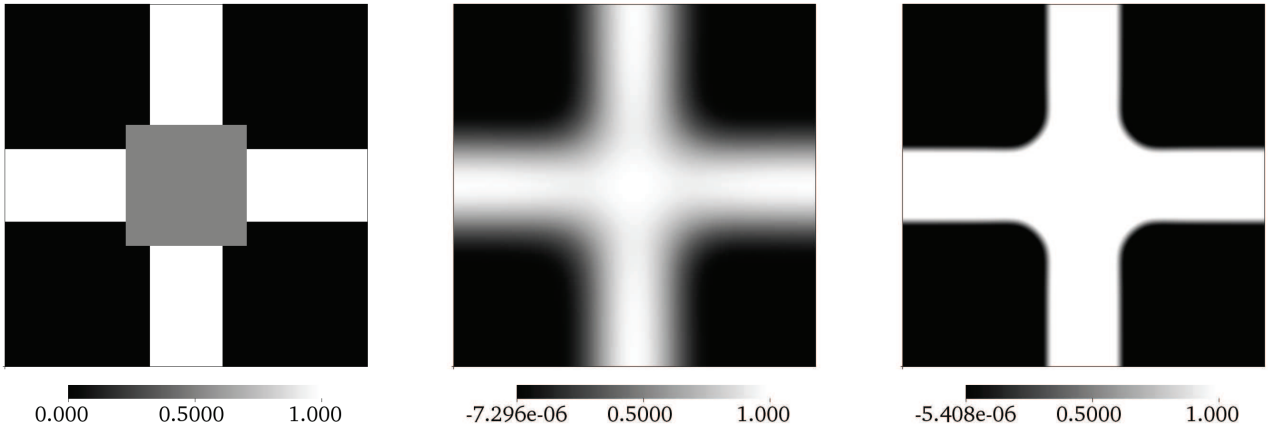
Figure 7.4: Non-smooth (above) and smooth (below) modified Cahn-Hilliard evolution for the circle computation.

7.3. Iteration numbers. We want to look at the number of BiCG iterations as well as the number of Newton iterations (in the non-smooth case) for different uniform mesh sizes.



(a) BiCG (b) Newton

Figure 7.5: Iteration numbers for the non-smooth model.



(a) $n = 0$ (b) $n = 149$ (c) $n = 558$

(d) $n = 0$ (e) $n = 148$ (f) $n = 475$

Figure 7.6: Non-smooth (above) and smooth (below) modified Cahn-Hilliard evolution for the cross computation.

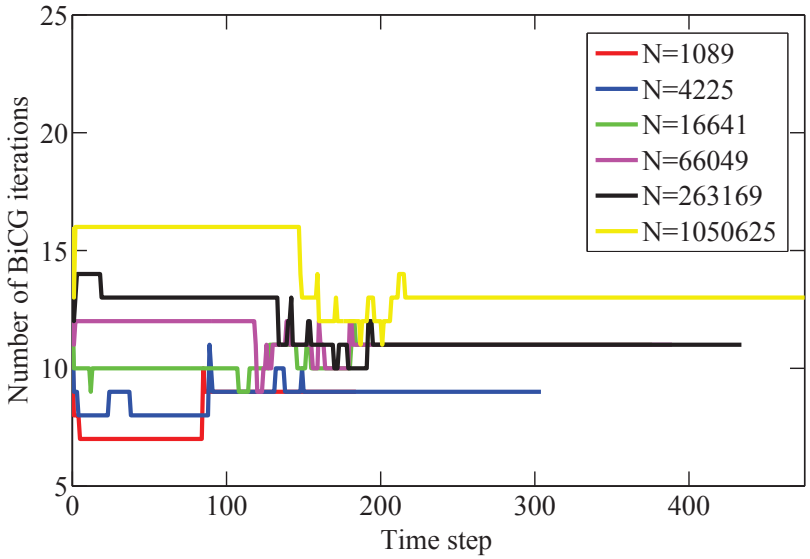
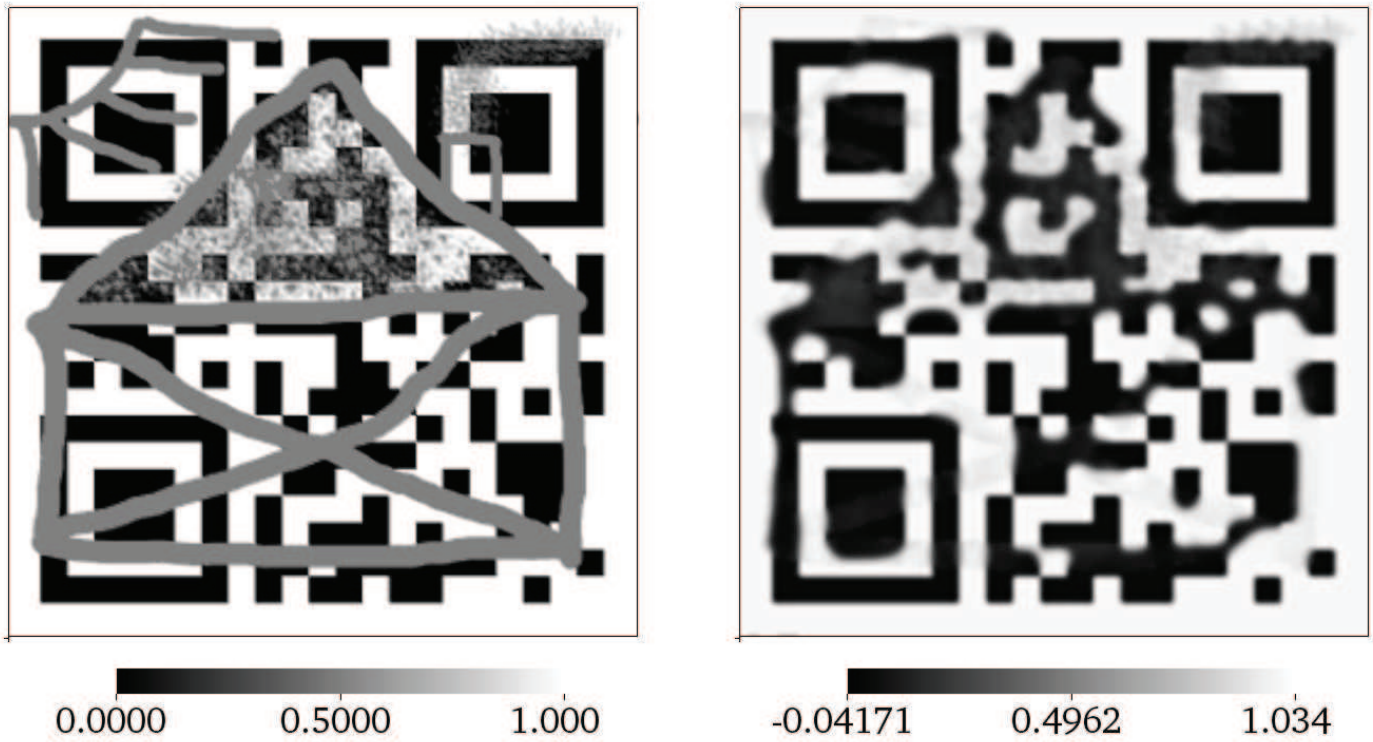


Figure 7.7: Iteration numbers for the smooth model.

One simulation is shown in Figure 7.6 (here $h = 2^{-10}$). As in the previous subsection, we use the two-scale approach with the same choice of all parameters. Figure 7.5 shows the number of BiCG iterations per Newton step and the number of Newton iterations per time step for the non-smooth model. In both figures, the number of unknowns N is listed. We observe a little growth after the ε-jump. For the SSN, we use the sequence with $\nu_{\max} = 10^{-7}$ for the meshes with $N \geq 16641$ for first 10 time steps. We can reduce the first few Newton iteration numbers after the ε-jump if we also use the ν-sequence here for some time steps. The BiCG iteration numbers for the smooth model are illustrated in Figure 7.7. We only have a benign increase for the coarser grids. In fact, we recognize a decrease of iteration numbers for the finer grids. All in all, the number of BiCG and Newton iterations stay low and we observe nearly mesh independent iteration numbers.

7.4. QR code. Now, we test the inpainting model for a QR code. Clearly, this is a very difficult example for Cahn-Hilliard inpainting because of the rectangular structures. Nevertheless, the result is satisfying, see Figure 7.8. Here, we use $\varepsilon = 0.005$ during the whole computation as well as $C_1 = 2\sqrt{\gamma\varepsilon(\frac{1}{\tau} + C_2)}$, $\omega_0 = 5 \cdot 10^7$. This choice for C_1 is optimal in the sense that we need only 2 BiCG iterations in every time step, see Section 5.



(a) $n = 0$ (b) $n = 16715$

Figure 7.8: Smooth modified Cahn-Hilliard evolution for the QR code computation.

7.5. Zebra. We now look at a real and more complex picture. For binary images, a very well suitable example is a photograph of a zebra. Figure 7.9 shows how the non-smooth modified Cahn-Hilliard model may be applied to the inpainting of an extract of a plains zebra

photo[1] for three different sets of parameters. The initial value for ε is 0.8 and we use $h = 2^{-8}$ in all simulations.

(a) $n = 0$ (b) $n = 53$ (c) $n = 1626$

(d) $n = 0$ (e) $n = 57$ (f) $n = 758$

(g) $n = 0$ (h) $n = 125$ (i) $n = 3311$

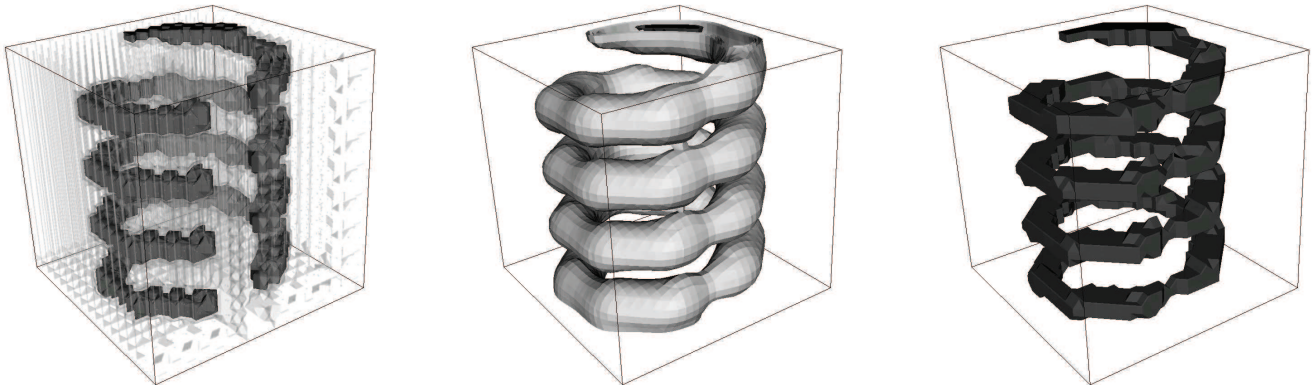
Figure 7.9: Non-smooth modified Cahn-Hilliard evolution for the zebra computation.

In the first example (first row in 7.9), we execute an ε-jump to 0.005 and use $C_1 = \frac{3}{\varepsilon}$ as well as $\omega_0 = 10^6$. We observed at most 41 average BiCG iterations. The second row

[1] ©2012 Thomas Rolle from the Zoo Magdeburg, Germany.

shows a simulation with an ε-jump to 0.01 as well as $C_1 = 1000$ and $\omega_0 = 10^6$ where we obtained at most 30 average BiCG iterations. In the last example we switch to the ε-jump 0.007 and set $C_1 = 7000$ and $\omega_0 = 1.5 \cdot 10^7$ and gained at most 41 BiCG iterations. The three simulations show the influence of different parameter settings. We get the most intensive colours in the first example when ε and C_1 are relatively small. Increasing ω_0 results in a more exact inpainting version in the area $\Omega \setminus D$ in the sense that thin black/white areas are captured more accurately, see the border of the zebra below the eye in the last example. By means of time steps, it can be observed that the larger C_1 the more time steps are needed.

7.6. Inpainting in 3D. Last but not least, we perform inpainting in 3D on a damaged spiral helix contained in a noisy box, see Figure 7.10(a). We set $\varepsilon = 1.8 \to 0.01$, $h = 2^{-5}$, $C_1 = \frac{3}{\varepsilon}$ and $\omega_0 = 10^5$ and apply the non-smooth Cahn-Hilliard model. We observe at most 42 BiCG iterations.



(a) $n = 0$ (b) $n = 82$ (c) $n = 160$

Figure 7.10: Non-smooth modified Cahn-Hilliard evolution for 3D inpainting.

8. Conclusions. In this paper, we have shown the practicability of a non-smooth potential for the modified Cahn-Hilliard equation. We have proposed a Moreau-Yosida regularization technique for the pointwise constraints and proved superlinear convergence of the semi-smooth Newton method in function space. Moreover, we have analysed the linear systems arising from the SSN method as well as from those appearing in the smooth modified Cahn-Hilliard equation. We have introduced and studied block-triangular preconditioners using an efficient and cheap to apply Schur complement approximation. This approximation can be done using multilevel techniques, algebraic multigrid in our case. In fact, we have found an optimal preconditioner for the smooth model that is independent of the mesh size. A comparison between the smooth and non-smooth model has shown the better result for the latter.

Acknowledgements. Parts of this work were performed while the first author was visiting the Oxford Center for Collaborative Applied Mathematics (OCCAM), University of Oxford. This publication was based on work supported in part by Award No. KUK-C1-013-04, made by King Abdullah University of Science and Technology (KAUST). The authors would like to thank Thomas Rolle for providing his photograph for this work.

REFERENCES

[1] W. Bangerth, R. Hartmann, and G. Kanschat, *deal.II – a general purpose object oriented finite element library*, ACM Trans. Math. Softw., 33 (2007), pp. 24/1–24/27.

[2] M. Bertalmio, A.L. Bertozzi, and G. Sapiro, *Navier-Stokes, fluid dynamics, and image and video inpainting*, in Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, 2001, pp. 355 – 362.

[3] M. Bertalmío, G. Sapiro, V. Caselles, and C. Ballester, *Image inpainting*, in SIGGRAPH, 2000, pp. 417–424.

[4] A.L. Bertozzi, S. Esedoḡlu, and A. Gillette, *Analysis of a two-scale Cahn–Hilliard model for binary image inpainting*, Multiscale Model. Simul., 6 (2007), pp. 913–936.

[5] A.L. Bertozzi, S. Esedoglu, and A. Gillette, *Inpainting of Binary Images Using the Cahn-Hilliard Equation*, IEEE Transactions on Image Processing, 16 (2007), pp. 285–291.

[6] J.F. Blowey and C.M. Elliott, *The Cahn-Hilliard gradient theory for phase separation with non-smooth free energy Part I: Mathematical analysis*, Eur. J. Appl. Math., 2 (1991), pp. 233–280.

[7] ———, *The Cahn-Hilliard gradient theory for phase separation with non-smooth free energy Part II: Numerical analysis*, Eur. J. Appl. Math., 3 (1992), pp. 147–179.

[8] J. Bosch, M. Stoll, and P. Benner, *Fast solution of Cahn-Hilliard variational inequalities using implicit time discretization and finite elements*, MPI Magdeburg Preprint MPIMD/13-01, January 2013.

[9] M. Burger, L. He, and C.-B. Schönlieb, *Cahn-Hilliard Inpainting and a Generalization for Grayvalue Images*, SIAM J. Img. Sci., 2 (2009).

[10] J.W. Cahn and J.E. Hilliard, *Free Energy of a Nonuniform System. I. Interfacial Free Energy*, J. Chem. Phys., 28 (1958), pp. 258–267.

[11] T.F. Chan and J. Shen, *Variational image inpainting*, Commun. Pure Applied Math., 58 (2005), pp. 579–619.

[12] T.F. Chan, J.H. Shen, and H.M. Zhou, *Total variation wavelet inpainting*, J. Math. Imaging Vis., 25 (2006), pp. 107–125.

[13] M.I.M. Copetti and C.M. Elliott, *Numerical analysis of the Cahn-Hilliard equation with a logarithmic free energy*, Numer. Math., 63 (1992), pp. 39–65.

[14] T.A. Davis, *UMFPACK version 4.4 user guide*, tech. report, Dept. of Computer and Information Science and Engineering Univ. of Florida, Gainesville, FL, 2005.

[15] J.A. Dobrosotskaya and A.L. Bertozzi, *A Wavelet-Laplace Variational Technique for Image Deconvolution and Inpainting*, Trans. Img. Proc., 17 (2008), pp. 657–663.

[16] I.C. Dolcetta, S.F. Vita, and R. March, *Area-preserving curve-shortening flows: From phase separation to image processing*, Interfaces Free Bound., 4 (2002), pp. 325–343.

[17] C.M. Elliott, *The Cahn-Hilliard model for the kinetics of phase separation*, in Mathematical Models for Phase Change Problems, J.F. Rodrigues, ed., vol. 88 of Int. Ser. Numer. Math., Birkhäuser, Basel, 1989, pp. 35–73.

[18] C.M. Elliott and A.M. Stuart, *The global dynamics of discrete semilinear parabolic equations*, SIAM J. Num. Anal., 30 (1993), pp. 1622–1663.

[19] C.M. Elliott and S. Zheng, *On the Cahn-Hilliard equation.*, Arch. Ration. Mech. Anal., 96 (1986), pp. 339–357.

[20] S. Esedoglu and J.H. Shen, *Digital inpainting based on the Mumford-Shah-Euler image model*, Eur. J. Appl. Math., 13 (2002), pp. 353–370.

[21] D.J. Eyre, *An unconditionally stable one-step scheme for gradient systems*, tech. report, Department of Mathematics, University of Utah, Salt Lake City, Utah, USA, 1997. unpublished.

[22] ———, *Unconditionally gradient stable time marching the Cahn-Hilliard equation*, in Computational and Mathematical Models of Microstructural Evolution, Mater. Res. Soc. Symp. Proc., ed., vol. 529, Bullard, J.W. and Chen, L.-Q. and Kalia, R.K. and Stoneham, A.M., 1998, pp. 39–46.

[23] R.D. Falgout, *An Introduction to Algebraic Multigrid*, Comput. Sci. Eng., 8 (2006), pp. 24–33. Special Issue on Multigrid Computing.

[24] R. Fletcher, *Conjugate gradient methods for indefinite systems*, in Numerical Analysis, G.A. Watson, ed., vol. 506 of Lect. Notes in Math., Springer Berlin Heidelberg, 1976, pp. 73–89.

[25] R.W. Freund and N.M. Nachtigal, *QMR: a quasi-minimal residual method for non-Hermitian linear systems*, Numer. Math., 60 (1991), pp. 315–339.
[26] H. Garcke, *Mechanical effects in the Cahn-Hilliard model: A review on mathematical results*, in Mathematical methods and models in phase transitions, A. Miranville, ed., Nova Science Publishers, New York, 2005, pp. 43–77.
[27] A. Gillette, *Image Inpainting Using a Modified Cahn-Hilliard Equation*, PhD thesis, University of California, Los Angeles, 2006.
[28] A. Greenbaum, *Iterative methods for solving linear systems*, vol. 17 of Frontiers in Applied Mathematics, SIAM, Philadelphia, PA, 1997.
[29] W. Hackbusch, *Multigrid methods and applications*, vol. 4 of Springer Ser. Comput. Math., Springer-Verlag, Berlin, 1985.
[30] M. Heroux, R. Bartlett, V.H.R. Hoekstra, J. Hu, T. Kolda, R. Lehoucq, K. Long, R. Pawlowski, E. Phipps, A. Salinger, H. Thornquist, R. Tuminaro, J. Willenbring, and A. Williams, *An Overview of Trilinos*, Tech. Report SAND2003-2927, Sandia National Laboratories, 2003.
[31] J.E. Hilliard and J.W. Cahn, *An evaluation of procedures in quantitative metallography for volume-fraction analysis*, vol. 221, General Electric Research Laboratory, Schenectady, N.Y., 1961.
[32] 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., 26 (2011), pp. 777–811.
[33] M. Hintermüller, K. Ito, and K. Kunisch, *The primal-dual active set strategy as a semismooth Newton method*, SIAM J. Optim., 13 (2002), pp. 865–888.
[34] M. Hintermüller and M. Ulbrich, *A mesh-independence result for semismooth newton methods*, Math. Program., 101 (2004), pp. 151–184.
[35] S. Masnou and J.-M. Morel, *Level lines based disocclusion*, in 5th IEEE International Conf. on Image Processing, vol. 3, oct 1998, pp. 259 –263.
[36] M.F. Murphy, G.H. Golub, and A.J. Wathen, *A note on preconditioning for indefinite linear systems*, SIAM J. Sci. Comput, 21 (2000), pp. 1969–1972.
[37] A. Novick-Cohen, *The Cahn-Hilliard equation: Mathematical and modeling perspectives*, Adv. Math. Sci. Appl., 8 (1998), pp. 965–985.
[38] J.W. Pearson and M. Stoll, *Fast Iterative Solution of Reaction-Diffusion Control Problems Arising from Chemical Processes*, Preprint, (2012).
[39] J. Pearson, M. Stoll, and A.J. Wathen, *Regularization-robust preconditioners for time-dependent pde-constrained optimization problems*, SIAM J. Matrix Anal. Appl., 33 (2012), pp. 1126–1152.
[40] J.W. Pearson and A.J. Wathen, *Fast iterative solvers for convection-diffusion control problems*, Elect. Trans. Numer. Anal., (2013). to appear.
[41] L.I. Rudin and S. Osher, *Total variation based image restoration with free local constraints*, in Proc. 1st IEEE ICIP, vol. 1, nov 1994, pp. 31–35.
[42] L.I. Rudin, S. Osher, and E. Fatemi, *Nonlinear total variation based noise removal algorithms*, Physica D: Nonlinear Phenomena, 60 (1992), pp. 259 – 268.
[43] J.W. Ruge and K. Stüben, *Algebraic multigrid*, in Multigrid methods, vol. 3 of Frontiers Appl. Math., SIAM, Philadelphia, PA, 1987, pp. 73–130.
[44] Y. Saad, *Iterative methods for sparse linear systems*, SIAM, Philadelphia, PA, 2003.
[45] Y. Saad and M.H. Schultz, *GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 7 (1986), pp. 856–869.
[46] C.-B. Schönlieb and A. Bertozzi, *Unconditionally stable schemes for higher order inpainting*, Commun. Math. Sci., 9 (2011), pp. 413–457.
[47] J. Shen and T. Chan, *Variational restoration of nonflat image features: Models and algorithms*, SIAM J. Appl. Math., 61 (2001), pp. 1338–1361.
[48] ———, *Mathematical models for local nontexture inpaintings*, SIAM J. Appl. Math., 62 (2002), pp. 1019–1043.
[49] J. Shen, S. Kang, and T. Chan, *Euler's elastica and curvature-based inpainting*, SIAM J. Appl. Math., 63 (2003), pp. 564–592.
[50] G. Strang and G. Fix, *An Analysis of the Finite Element Method 2nd Edition*, Wellesley-Cambridge, 2nd ed., May 2008.

[51] A. Tsai, J.A. Yezzi, and A.S. Willsky, *Curve evolution implementation of the mumford-shah functional for image segmentation, denoising, interpolation, and magnification*, Trans. Img. Proc., 10 (2001), pp. 1169–1186.

[52] M. Ulbrich, *Semismooth Newton methods for operator equations in function spaces*, SIAM J. Optim., 13 (2002), pp. 805–842.

[53] H.A. Van der Vorst, *BiCGSTAB: A fast and smoothly converging variant of biCG for the solution of nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 13 (1992), pp. 631–644.

[54] B.P. Vollmayr-Lee and A.D. Rutenberg, *Fast and accurate coarsening simulation with an unconditionally stable time step*, Phys. Rev. E, 68 (2003).

[55] P. Wesseling, *An introduction to multigrid methods*, Pure and Applied Mathematics (New York), John Wiley & Sons Ltd., Chichester, 1992.

[56] X. Wu and Y. Dzenis, *Phase-field modeling of the formation of lamellar nanostructures in diblock copolymer thin films under inplanar electric fields*, Phys. Rev. E: Stat., Nonlinear, Soft Matter Phys., 77 (2008), p. 031807.