Bachelor Thesis

Jonathan Ulmer

January 11, 2024

Contents

1 Cahn Hillard Equation Overview

Partial Differential Equation (PDE) solving the state of a 2 Phase Fluid[2]. The form of the Cahn Hillard Equation used for the remainder of this thesis is:

$$\phi_t(x,t) = \Delta\mu \quad (1)$$

$$\mu = -\varepsilon^2\Delta\phi + W'(\phi) \quad (2)$$

where ϕ is the so-called phase field. Demarking the different states of the fluids through an Interval $I = [-1, 1]$ and where $\partial I = \{-1, 1\}$ represents full state of one fluid. $\varepsilon > 0$ is a positive constant

, and μ is the chemical potential[2]. While the Cahn Hillard exist in a more general form taking the fluid's mobility $M(\Phi)$ into account, we will assume $M(\Phi) = 1$, simplifying the CH-Equations used in [2] [1] to what is stated above.

The Advantages of the Cahn Hillard Approach as compared to traditional fluid dynamics solvers are for example: "explicit tracking of the interface" [2], as well as "evolution of complex geometries and topological changes [...] in a natural way" [2] In practice it enables linear interpolation between different formulas on different phases

1.1 TODO Derivation from paper

1.1.1 Free energy

The Cahn Hillard Equations can be motivated Using a **Ginzburg Landau** type free energy equation:

$$E^{\text{bulk}} = \int_\Omega \frac{\varepsilon^2}{2}|\nabla\phi|^2 + W(\phi)\,dx$$

where $W(\phi)$ denotes the (Helmholtz) free energy density of mixing."" [2] and will be approximated in further calculations as $W(\phi) = \frac{(1-\phi^2)^2}{4}$ as used in[1]

The chemical potential then follows as derivative of Energy in respect to time.

$$\mu = \frac{\delta E_{bulk}(\phi)}{\delta\phi} = -\varepsilon^2\Delta\phi + W'(\phi)$$

1.1.2 Derivation by mass balance

The Cahn Hillard equation then can be motivated as follows: consider

$$\partial_t\phi + \nabla J = 0 \quad (3)$$

where **J** is mass flux. 3 then states that the change in mass balances the change of the phasefield. Using the no-flux boundry conditions:

$$J \cdot n = 0 \qquad \partial\Omega \times (0,T) \tag{4}$$
$$\partial_n \phi = 0 \qquad \partial\Omega \times (0,T) \tag{5}$$

conservation of mass follows see[2].

Using:

$$J = -\nabla\mu \tag{6}$$

which conceptionally sets mass flux to equalize the potential energy gradient, leads to the formulation of the CH equations as stated above. Additionally, the boundary conditions evaluate to:

$$-\nabla\mu = 0$$
$$\partial_n \phi = 0$$

ie no flow leaves and potential on the border doesn't change. Then for ϕ then follows:

$$\begin{aligned}\frac{d}{dt}E^{bulk}(\phi(t)) &= \int_\Omega (\varepsilon^2 \nabla\phi \cdot \nabla\partial_t\phi + W'(\phi)\partial_t\phi)\ dx \\ &= -\int_\Omega |\nabla\mu|^2\ dx, \qquad \forall t \in (0,T)\end{aligned}$$

hence the Free Energy is decreasing in time.

2 Baseline Multigrid solver:

As baseline for further experiments a multi grid method based on finite differences by[1]. Is used.

2.1 Discretization:

it discretizes the phasefield and potential energy ϕ, μ into a grid wise functions ϕ_{ij}, μ_{ij} and defines the partial derivatives $D_x f_{ij},\ D_y f_{ij}$ using the differential quotients:

$$D_x f_{i+\frac{1}{2}j} = \frac{f_{i+1j} - f_{ij}}{h} \qquad D_y f_{ij+\frac{1}{2}} = \frac{f_{ij+1} - f_{ij}}{h} \tag{7}$$

for $\nabla f, \Delta f$ then follows:

$$\nabla_d f_{ij} = (D_x f_{i+1j},\ D_y f_{ij+1})$$
$$\Delta_d f_{ij} = \frac{D_x f_{i+\frac{1}{2}j} - D_x f_{i-\frac{1}{2}j} + D_y f_{ij+\frac{1}{2}} - D_y f_{ij-\frac{1}{2}}}{h} = \nabla_d \cdot \nabla_d f_{ij}$$

the authors further adapt the discretized phasefield by the characteristic function of the domain Ω:

$$G(x,y) = \begin{cases} 1 & (x,y) \in \Omega \\ 0 & (x,y) \notin \Omega \end{cases}$$

To account for boundry conditions and arbitrary shaped domains. The authors [1] then define the discrete CH Equation adapted for Domain, as:

$$\frac{\phi_{i+1j} - \phi_{ij}}{\Delta t} = \nabla_d \cdot (G_{ij} \nabla_d \mu_{ij}^{n+1})$$
$$\mu_{ij}^{n+1} = 2\phi_{ij}^{n+1} - \varepsilon^2 \nabla_d \cdot (G_{ij} \nabla_d \phi_{ij}^{n+1}) + W'(\phi_{ij}^n) - 2\phi_{ij}^n$$

and derive the iteration operator $L(\phi^{n+1}, \mu^{n+\frac{1}{2}}) = (\zeta^n, \psi^n)$

$$L \begin{pmatrix} \phi^{n+1} \\ \mu^{n+\frac{1}{2}} \end{pmatrix} = \begin{pmatrix} \frac{\phi^{n+1}}{\Delta t} - \nabla_d \cdot (G_{ij} \nabla_d \mu^{n+\frac{1}{2}}) \\ \varepsilon^2 \nabla_d \cdot (G_{ij} \nabla_d \phi_{ij}^{n+1}) - 2\phi_{ij}^{n+1} + \mu_{ij}^{n+\frac{1}{2}} \end{pmatrix}$$

initialized as

$$(\zeta^n, \psi^n) = \begin{pmatrix} \frac{\phi_{ij}^{n+1}}{\Delta t} \\ W'(\phi_{ij}^n) - 2\phi_{ij}^n \end{pmatrix}$$

the algorithm is then defined as:

Wherein SMOOTH consists of point-wise Gauß Seidel Relaxation, by solving L for $\overline{\phi}, \overline{\mu}$ with the initial guess for ζ^n, ψ^n.

2.2 adaptations to the simplified problem

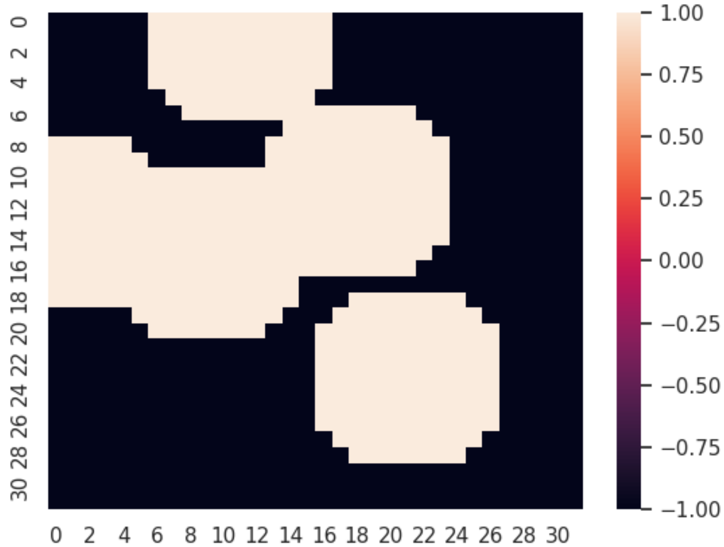
even tough this work uses rectangular domains, the adaptation of the algorithm is simplified by the domain indicator function, as well as 0 padding, in order to correctly include the boundary conditions of the CH equation. Therefore, the internal representation of the adapted algorithm considers phasefield and potential field ϕ, μ as 2D arrays of shape $(N_X + 2, N_y + 2)$ in order to accommodate padding. Where $\mathrm{N_x}$ and $\mathrm{N_y}$ are the number of steps in x- / y-Direction respectively. Hence, we define the discrete domain function as:

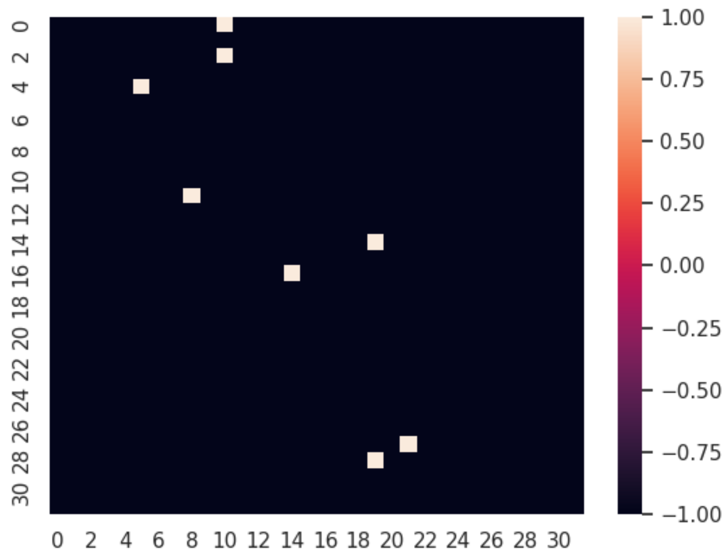
$$G_{ij} = \begin{cases} 1 & (i,j) \in [1, N_x + 1] \times [1, N_y + 1] \\ 0 & \text{else} \end{cases}$$

2.3 tests$_{\mathbf{data}}$:

2.3.1 squares

For testing and later training, a multitude o different phasefields where used. Notably an assortment of randomly placed circles, squares, and arbitrary generated values





2.4 Tests

```
test_phase = tu.k_spheres_phase(4,7)
solver = tu.setup_solver(test_phase)
solver.solve(4,10)
```

```
plt.figure()
sns.heatmap(solver.phase_small)
```

3 Relaxed Problem

In effort to decrease the order of complexity, the following relaxation to the classical Cahn Hillard Equation is proposed:

$$\partial_t \phi^\alpha = \Delta \mu$$
$$\mu = \varepsilon^2 (c^\alpha - \phi^\alpha) + W'(\phi)$$

that in turn requires solving an additional PDE each time-step to calculate c. c is the solution of the following elliptical PDE

$$-\Delta c^\alpha + \alpha c^a = \alpha \phi^\alpha$$

3.1 TODO relaxed operators:

the multi-grid solver proposed earlier is then adapted to the relaxed Problem by replacing the differential operators by their discrete counterparts as defined in ?? and expanding them

3.1.1 L Relaxed

for the reformulation of the iteration in terms of Operator L then follows:

$$L\begin{pmatrix}(\phi^{n+1})^\alpha\\ \mu^{n+1}\end{pmatrix}=\begin{pmatrix}\frac{(\phi_{ij}^{n+1,m})^\alpha}{\Delta t}-\nabla_d\cdot(G_{ji}\nabla_d\mu_{ji}^{n+\frac{1}{2},m})\\ \varepsilon^2(c^\alpha-(\phi_{ij}^{n+1,m})^\alpha)-2(\phi_{ij}^{n+1,m})^\alpha-\mu_{ji}^{n+\frac{1}{2},m}\end{pmatrix}$$

3.1.2 SMOOTH

and correspondingly the SMOOTH operation expands to:

$$SMOOTH((\phi_{ij}^{n+1,m})^\alpha,\mu_{ji}^{n+\frac{1}{2},m},L_h,\zeta^n,\psi^n)$$

$$\begin{aligned}\overline{\mu}_{ji}^{n+\frac{1}{2},m}&=\frac{(\phi_{ij}^{n+1,m})^\alpha}{\Delta t}-\zeta_{ij}^n\\&-\frac{1}{h^2}\left(G_{i+\frac{1}{2}j}\mu_{i+1j}^{n+\frac{1}{2},m}+G_{i-1j}\mu_{i-1j}^{n+\frac{1}{2},m}+G_{ij+1}\mu_{ij+1}^{n+\frac{1}{2},m}+G_{ij-1}\mu_{ij-1}^{n+\frac{1}{2},m}\right)\\&\cdot(G_{i+1j}+G_{i-1j}+G_{ij+1}+G_{ij-1})^{-1}\\ \varepsilon^2(\overline{\phi}_{ij}^{n+1,m})^\alpha+2\phi_{ij}^{n+1,m}&=\varepsilon^2c^\alpha-\mu_{ji}^{n+\frac{1}{2},m}-\psi_{ij}\end{aligned}$$

1. Proposal Since the resulting system no longer is linear, (albeit simpler in Dimension), we propose a newton method to solve second equation (in conjunction with the first one) hopefully solving this converges faster than the original multiple SMOOTH Iterations. The iteration solves for $(\phi_{ij}^{n+1,m})^\alpha=x$ as free variable. Therefore, it follows for $F(x)$

$$\begin{aligned}F(x)&=\varepsilon^2x^\alpha+2x-\varepsilon^2c^\alpha+y+\psi_{ij}\\ y&=\frac{x}{\Delta t}-\zeta_{ij}^n\\&-\frac{1}{h^2}\left(G_{i+\frac{1}{2}j}\mu_{i+1j}^{n+\frac{1}{2},m}+G_{i-1j}\mu_{i-1j}^{n+\frac{1}{2},m}+G_{ij+1}\mu_{ij+1}^{n+\frac{1}{2},m}+G_{ij-1}\mu_{ij-1}^{n+\frac{1}{2},m}\right)\\&\cdot(G_{i+1j}+G_{i-1j}+G_{ij+1}+G_{ij-1})^{-1}\end{aligned}$$

And the derivative for the iteration is

$$\frac{d}{dx}F(x) = \alpha\varepsilon^2 x^{\alpha-1} + 2 + \frac{d}{dx}y$$
$$\frac{d}{dx}y = \frac{1}{\Delta t}$$

3.2 Elliptical PDE:

on order to solve the relaxed CH Equation the following PDE as to be solved in Each additional time step: or in terms of the characteristic function:

$$-\nabla \cdot (G\nabla c^\alpha) + \alpha c^\alpha = \alpha\phi^\alpha$$

Similarly to the first solver this PDE is solved with a finite difference scheme using the same discretisations as bevore:

3.2.1 Discretization

the Discretization of the PDE expands the differential opperators in the same way

$$\begin{aligned}
-\nabla_d \cdot (G_{ij}\nabla_d c^\alpha_{ij}) + \alpha c^\alpha_{ij} &= \alpha\phi^\alpha_{ij} \\
-(\frac{1}{h}(G_{i+\frac{1}{2}j}\nabla c^\alpha_{i+\frac{1}{2}j} + G_{ij+\frac{1}{2}}\nabla c^\alpha_{ij+\frac{1}{2}}) \\
-(G_{i-\frac{1}{2}j}\nabla c^\alpha_{i-\frac{1}{2}j} + G_{ij-\frac{1}{2}}\nabla c^\alpha_{ij-\frac{1}{2}})) + \alpha c^\alpha_{ij} &= \alpha\phi^\alpha_{ij} \\
-\frac{1}{h^2}(G_{i+\frac{1}{2}j}(c^\alpha_{i+1j} - c^\alpha_{ij}) \\
+G_{ij+\frac{1}{2}}(c^\alpha_{ij+1} - c^\alpha_{ij}) \\
+G_{i-\frac{1}{2}j}(c^\alpha_{i-1j} - c^\alpha_{ij}) \\
+G_{ij-\frac{1}{2}}(c^\alpha_{ij-1} - c^\alpha_{ij})) + \alpha c^\alpha_{ij} &= \alpha\phi^\alpha_{ij}
\end{aligned}$$

proposed simple solver: Let F, dF be:

$$\begin{aligned}
F(x) = -\frac{1}{h^2}(\\
G_{i+\frac{1}{2}j}c^\alpha_{i+1j} + G_{i-\frac{1}{2}j}c^\alpha_{i-1j} \\
+G_{ij+\frac{1}{2}}c^\alpha_{ij+1} + G_{ij-\frac{1}{2}}c^\alpha_{ij-1}) \\
+\frac{1}{h^2}(G_{i+\frac{1}{2}j} + G_{i-\frac{1}{2}j} + G_{ij+\frac{1}{2}} + G_{ij-\frac{1}{2}})x^\alpha \\
+\alpha x^\alpha - \alpha\phi^\alpha_{ij}
\end{aligned}$$

and $dF(x)$

$$dF(x) = - \cdot (G_{i+\frac{1}{2}j} + G_{i-\frac{1}{2}j} + G_{ij+\frac{1}{2}} + G_{ij-\frac{1}{2}})\alpha x^{\alpha-1} + \alpha^2 x^{\alpha-1}$$

solve equation for c_{ij}^{α} and then iteratively update the entire phasefield using the already updated values:

4 References