NTNU | Norwegian University of Science and Technology

CUT FINITE ELEMENT METHOD FOR THE CAHN-HILLIARD EQUATION

Supervised by André Massing

Isak Hammer

May 4, 2023

Introducing Myself

- Isak Hammer, 27 year old, Lofoten
- Graduate student in Industrial Mathematics
- Research Focus: Numerical methods for Partial Differential Equations (PDEs).

Importance and Motivation of the Cahn Hilliard Equation

- Thermodynamically modelling of a two-component liquid separation[1].
- Modelling of so-called lipid rafts in biological membrane dynamics [2].
- Droplet dynamics, i.e., coalescence, breakup and movement by coupling with Navier-Stokes [3].

---

[1] John W Cahn and John E Hilliard. "Free energy of a nonuniform system. III. Nucleation in a two-component incompressible fluid". In: *The Journal of chemical physics* 31.3 (1959), pp. 688–699

[2] Vladimir Yushutin et al. "A computational study of lateral phase separation in biological membranes". In: *International journal for numerical methods in biomedical engineering* 35.3 (2019), e3181

[3] Patrick Zimmermann, Andrew Mawbey, and Tim Zeiner. "Calculation of droplet coalescence in binary liquid–liquid systems: An incompressible Cahn–Hilliard/Navier–Stokes approach using the non-random two-liquid model". In: *Journal of Chemical & Engineering Data* 65.3 (2019), pp. 1083–1094

The Cahn Hilliard Equation

The general Cahn Hilliard Equation has the form $u(x,t) : \Omega \times [0,T] \mapsto [-1,1]$ s.t.

$$\begin{aligned} u_t + \Delta\left(\varepsilon \Delta u - \frac{1}{\varepsilon} f(u)\right) &= 0 \quad \text{in } \Omega \\ \partial_n u = \partial_n \Delta u &= 0 \quad \text{on } \Gamma \\ u &= u_0 \quad \text{on } \Omega \end{aligned}$$

where $f(s) = F'(s)$ and $F(s) = \frac{1}{4}\left(s^2 - 1\right)^2$ and $\Omega \subset \mathbf{R}^d, d = 2, 3$, is a bounded domain.

Challenges

1. Highly nonlinear and stiff. Often practical applications require $\varepsilon \ll 1$.
2. 4th order system.

Why Finite Element Method (FEM)

1. **Robust mathematical framework**
2. **Can easily handle complex geometries**
3. **High flexibility of basis functions**
4. **Other:** Supports adaptive refinements, easily adaptable to multi-physics problems ++ .

The Biharmonic Problem (on a polygon)

Let $\Omega \approx \Omega_h = \mathcal{T}_h$ be a bounded polygonal domain with boundary Γ . Let the biharmonic problem have the form s.t. $u : \Omega \mapsto \mathbb{R}$,

$$\begin{aligned} \Delta^2 u + \alpha u &= f(x) \quad \text{in } \Omega, \\ \partial_n u &= 0 \quad \text{on } \Gamma, \\ \partial_n \Delta u &= 0 \quad \text{on } \Gamma. \end{aligned} \tag{1}$$

Here is $\Delta^2 = \Delta\left(\Delta\right)$ the biharmonic operator.

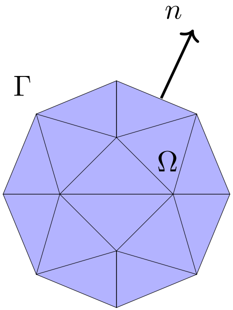


Figure: Illustration of the mesh Ω_h, the boundary Γ and the normal vector n.

C^0 Interior Penalty Method (CIP) for the Biharmonic Problem

The proposed numerical scheme is to find an $w \in V_h$ .t.

$$a_h(w, v) = l_h(v) = (f, v)_\Omega, \quad \forall v \in V_h.$$

where

$$\begin{aligned} a_h(w, v) = {} & (\alpha w, v)_\Omega + (\Delta w, \Delta v)_\Omega \\ & + (\{\!\{\Delta w\}\!\}, [\![\partial_n v]\!])_{\mathcal{F}_h} + (\{\!\{\Delta v\}\!\}, [\![\partial_n w]\!])_{\mathcal{F}_h} + \frac{\gamma}{h}([\![\partial_n w]\!], [\![\partial_n v]\!])_{\mathcal{F}_h} \end{aligned}$$

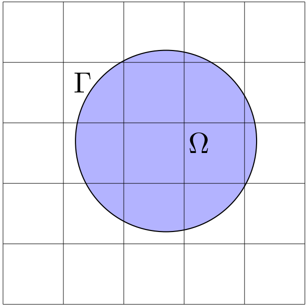
Which is inspired from Brenner2012 [1]

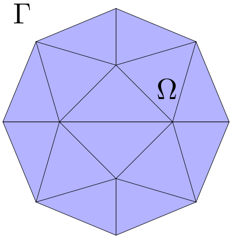
[1] Susanne Brenner. *C0 Interior Penalty Methods*. Springer International Publishing, 2012. URL: https://link.springer.com/content/pdf/10.1007/978-3-642-23914-4_2.pdf

Cut Finite Element Method (CutFEM)

Unfitted mesh vs fitted mesh

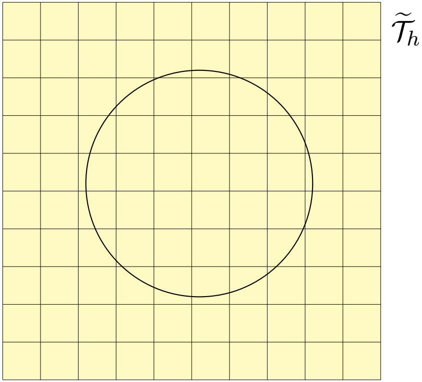
CutFEM is a numerical method for solving partial differential equations (PDEs) using an unfitted mesh.





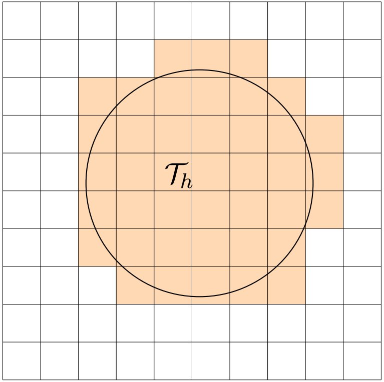
Cut Finite Element Method

Background Mesh



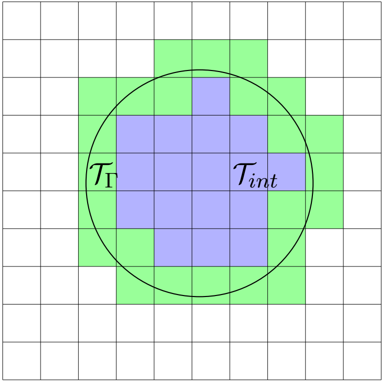
Cut Finite Element Method

Active Mesh



Cut Finite Element Method

Interior Mesh and Cut Cells



Cut Finite Element Method

A recent and promising numerical technique for PDEs, has gained significant momentum in the past decade [1,2].

- Complex domains and moving domains efficiently.
- Utilizing so-called ghost penalties to ensure well-posedness.

---

[1] Erik Burman et al. "CutFEM: discretizing geometry and partial differential equations". In: *International Journal for Numerical Methods in Engineering* 104.7 (2015), pp. 472–501

[2] Ceren Gürkan and André Massing. "A stabilized cut discontinuous Galerkin framework for elliptic boundary value and interface problems". In: *Computer Methods in Applied Mechanics and Engineering* 348 (2019), pp. 466–499

Cut C^0 Interior Penalty Method (CutCIP)

The discretized numerical problem is to solve $w \in V_h$ such that

$$A(w, v) = a_h(w, v) + g_h(w, v) = l_h(v), \quad \forall v \in V_h.$$

Where the additional bilinear term $g_h(w, v) : V_h \times V_h \to \mathbb{R}$ is the so-called **ghost penalty**, which does the numerical regularization to ensure stability on cut cells.

Cut C^0 Interior Penalty Method

My master's thesis is dedicated to demonstrating that the relevant properties remain valid for CutCIP formulation still holds.

Well-posedness

The discrete bilinear form a_h is wellposed on V_h if this holds;

$$
\begin{aligned}
(Coercivity) \quad & A(v,v) \gtrsim \|v\|_A^2 \quad \forall v \in V_h \\
(Boundedness) \quad & A(v,w) \lesssim \|v\|_A \|w\|_{a_h} \quad \forall v, w \in V_h
\end{aligned}
$$

Cut C^0 Interior Penalty Method Results

Manufactured solution

In the experiments will we only consider polynomial order $k = 2$. We consider the manufactured solution:

$$u_{ex}(\mathbf{x}) = \left(x_1^2 + x_2^2 - 1\right)^2 \cos(2\pi x_1)\cos(2\pi x_2)$$

where $\mathbf{x} = (x_1, x_2)$ and $\Omega = \{(x_1, x_2) : x_1^2 + x_2^2 \leq 1\}$. This manufactured solution can be used to test the accuracy of numerical methods for solving the above differential equation.

Cut C^0 Interior penalty method (CutCIP) Results

n	$\|e\|_{L^2}$	EOC	$\|e\|_{H^1}$	EOC	$\|e\|_{a_h,*}$	EOC	Cond number	ndofs
4	2.4E+00		3.3E+00		6.2E+01		8.7E+04	8.1E+01
8	3.6E-01	2.72	1.1E+00	1.60	3.9E+01	0.68	5.1E+05	2.4E+02
16	2.2E-02	4.06	2.5E-01	2.12	1.4E+01	1.51	3.7E+06	8.3E+02
32	5.6E-03	1.97	6.0E-02	2.04	3.6E+00	1.93	2.8E+07	3.0E+03
64	1.4E-03	2.00	1.5E-02	2.02	9.2E-01	1.96	2.1E+08	1.1E+04
128	3.5E-04	2.00	3.7E-03	2.01	2.4E-01	1.94	1.7E+09	4.3E+04

Cut C^0 Interior penalty method (CutCIP) Results

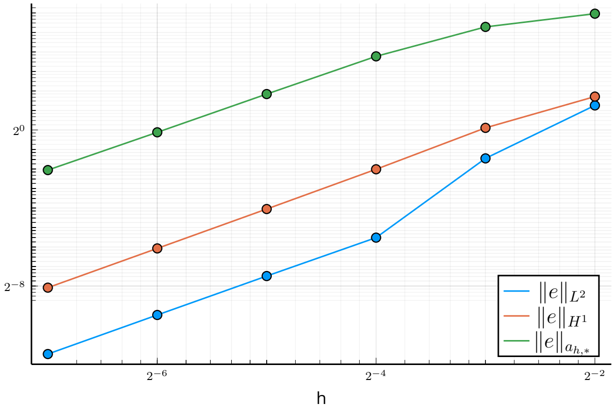


Figure: The plot presents the L^2 and H^1 error norms and the error in the energy norm ($\|e\|_{a_h,*}$).

The Cahn Hilliard Equation

Recall

The problem has the form
$u(x,t) : \Omega \times [0,T] \mapsto [-1,1]$ s.t.

$$
\begin{aligned}
u_t + \Delta\left(\varepsilon \Delta u - \frac{1}{\varepsilon} f(u)\right) &= 0 \quad \text{in } \Omega \\
\partial_n u = \partial_n \Delta u &= 0 \quad \text{on } \Gamma \\
u &= u_0 \quad \text{on } \Omega
\end{aligned}
$$

where $f(u)$ is a nonlinear function.

Plan forward

1. We have now a tool to solve the $\Delta(\Delta u)$ operator
2. Will utilize the time-iteration scheme to solve non-linearity

The CutCIP Cahn-Hilliard Formulation

Drawing upon the concepts delineated in Feng[1], the most efficient approach to address the nonlinear term is by employing an implicit-explicit (IMEX) scheme.

IMEX method on the CutCIP formulation

Let $u_h^m \in V_h$ for the timesteps $m = 0, 1, \ldots, M$. Let $u_h^0 = u_0$ be the initial timestep, then is.

$$(\overline{\partial}_t u_h^m, v_h) + \varepsilon A(u_h^m, v_h) + \frac{1}{\varepsilon} c_h(u_h^{m-1}, v_h) = 0 \quad \forall v_h \in V_h^m.$$

Here is $c_h(.,.)$ an the nonlinear terms handled in a implicit fashion. The $\overline{\partial}_t$ operator is simply a finite difference scheme in time-dimension.

[1] Xiaobing Feng and Ohannes Karakashian. “Fully discrete dynamic mesh discontinuous Galerkin methods for the Cahn-Hilliard equation of phase transition”. In: *Mathematics of computation* 76.259 (2007), pp. 1093–1117

The CutCIP Cahn-Hilliard Experiments

Implemented using the Gridap FEM framework written in Julia [1].

Simulation parameters

- Physical domain Ω is a 4 discs of radius $R = 1$ with distance $d = 0.999$, i.e. they are touching!
- Inital data is $u_0 = random(-1, 1)$ in physical domain Ω.

[1] Santiago Badia and Francesc Verdugo. "Gridap: An extensible finite element toolbox in julia". In: *Journal of Open Source Software* 5.52 (2020), p. 2520

The CutCIP Cahn-Hilliard Experiments

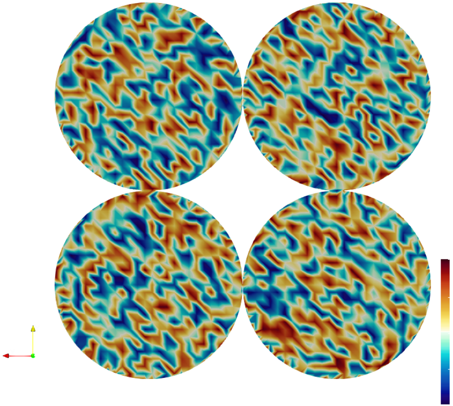


Figure: Iteration 0

The CutCIP Cahn-Hilliard Experiments

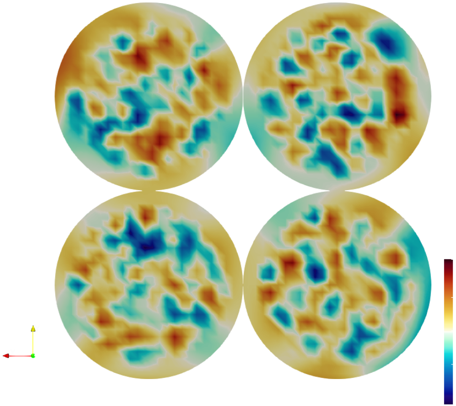


Figure: Iteration 1

The CutCIP Cahn-Hilliard Experiments

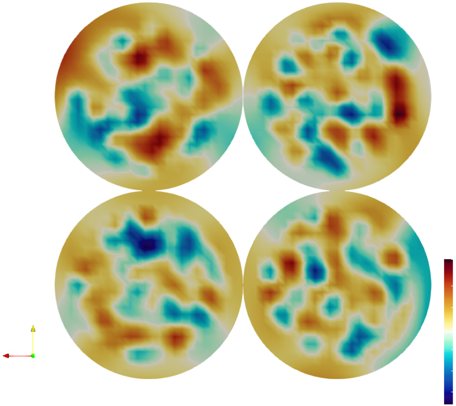


Figure: Iteration 10

The CutCIP Cahn-Hilliard Experiments

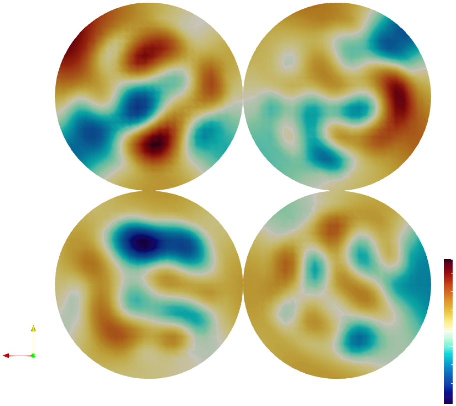


Figure: Iteration 50

The CutCIP Cahn-Hilliard Experiments

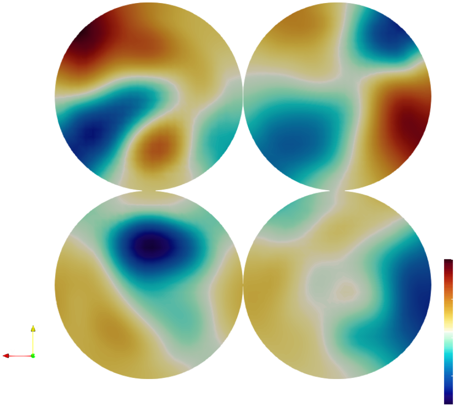


Figure: Iteration 200

The CutCIP Cahn-Hilliard Experiments

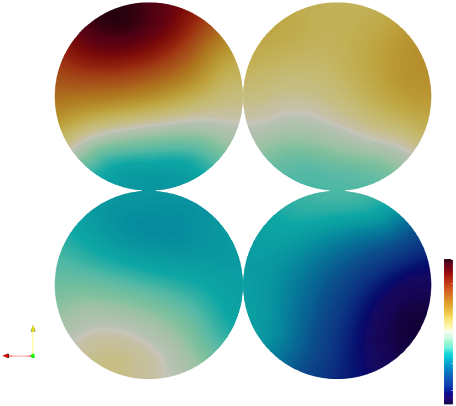


Figure: Iteration 1000

Further work

1. Adaptive time steps.
2. Further numerical validation.
3. Extend the method to handle moving domains.

Questions?