Carnegie Mellon University

Capturing Interfaces in Multi-Phase Problems

24658: Image-Based Computational Modeling and Analysis - Final Report

January 29, 2023

Authors:
Vedant Puri vedantpuri@cmu.edu
Michael Bennington mbenning@andrew.cmu.edu

Abstract

Interfaces between different bodies and material phases define the interactions between a domain of interest and its surroundings. When studying the interactions between multiple domains, it is vital to accurately track this interface as it both defines the time-varying and spatially-varying boundary conditions of the domains and allows for the incorporation of interfacial multiphysics. These types of problems can be seen across multiple disciplines of science, engineering, and medicine, but the computation complexity of solving these problems using standard techniques makes thorough numerical modeling difficult. In this report, we will review a number of explicit and implicit boundary representation models for solving these problems. We will then apply one of these methods, the Phase Field Method, to a problem within fracture mechanics to investigate the relationship of crack propagation, stress state, and initial crack length.

1 Introduction

In many fields of physics and engineering, interfaces play an important role in prescribing spatially-varying and time-varying boundary conditions. These interfaces can be fluid-fluid in the case of multiphase fluid models, fluid-solid in the case of fluid-structure interaction problems, or solid-solid in the case of contact mechanics and fracture mechanics. The modeling of interfaces using traditional techniques requires the computationally expensive task of tracking the boundary locations of discrete bodies and searching for body penetration/interaction locations. These computational costs are exacerbated when dealing with moving, or evolving-boundary problems. Collectively referred to as Stefan problems, these complex systems present a number of difficulties for traditional numerical analysis. The interspersion of two material leads to discontinuities of material properties within the domain. The meshes that are required becoming increasingly complex as higher resolution is needed near interfaces, and as these interfaces evolve, these meshes often need to be repeatedly regenerated. This process itself can occupy the majority of the computational effort of the software particularly when the topology of the domain changes.

A number of computational approaches have been developed in attempts to capture these interfaces. The set of techniques can be broadly broken into two classes: methods that utilize an explicit representation of the interface, and methods that utilize implicit interface representations. In explicit

interface methods, the boundary is discretized along with the rest of the domain, and its location is explicitly tracked by by 'Lagrangian' boundary points. Constraints and forces are then applied at the boundary points to enforce the interface conditions. Such methods include the Immersed Boundary Method [4, 9], originally developed for fluid-structure interaction in cardiovascular biomechanics, as well as Penetration Penalty Methods and Lagrange Multiplier Methods [7, 10]. The latter two methods are popular in the field of contact mechanics due to ease of implementation, although the methods are computationally expensive.

On the other hand, in implicit interface methods, the boundary is not directly tracked by discrete points but is instead captured by an auxiliary function or field that permeates the entire domain. The boundary is then defined by some property of the auxilary function, usually its level set. The interactions occurring at the interface are the embedded in the dynamics of the auxiliary function. Such methods include the Level Set Method [4], often used in computer graphics and physics-informed image analysis techniques, as well as the related Finite Cell [2] and Phase Field Methods[4, 3], models often used for the modeling of multiphase fluids.

In this project, the general formulation of the above methods will be explained along with their application. Then, the Phase Field Method will be further explored in the context of fracture mechanics. A recent vector-valued phase field model [3] of crack growth will be implemented using the open source FEniCS [1] finite element framework, and a simple test case will be investigated. Finally, future directions for the application of the Phase Field Method within solid mechanics will be discussed.

2 Background

2.1 Explicit Interface Representations

In explicit interface representation methods, the boundary between two material phases is discretized into a Lagrangian boundary mesh. The nodes of this need will not inherently align with the nodes of a background domain grid or be embedded in a larger grid at all. The interactions that occur at these interfaces are captured by adding either displacement constraints or forces at the nodes of the boundary mesh.

2.1.1 Immersed Boundary Method

In the Immersed Boundary Method (IBM), the interface between two phases is described by a parametric function over a set of Lagrangain points at the interface. Boundary constraints are specified at the interface, and the governing equations are modified by the addition of a forcing term that would enforces said constraints.

$$\begin{aligned} \partial_t \boldsymbol{u} &= \boldsymbol{f} + \int_\Gamma \boldsymbol{f}(\boldsymbol{\xi})\delta(\boldsymbol{\xi} - \boldsymbol{x})\mathrm{d}\boldsymbol{s} \\ \boldsymbol{u}(\boldsymbol{\xi}) &= \int_\Omega \boldsymbol{u}(x)\delta(\boldsymbol{x} - \boldsymbol{\xi})\mathrm{d}\boldsymbol{x} = \boldsymbol{u}_\Gamma(\boldsymbol{\xi}) \end{aligned} \tag{1}$$

Here, $\boldsymbol{f}(\boldsymbol{\xi})$ is a force vector described on Γ, which is spanned by the variable $\boldsymbol{\xi}$. The challenge in Immersed Boundary Methods is in finding a forcing function $\boldsymbol{f}$ that, when added to the right-hand-side of the governing equation, enforces a Dirichlet boundary condition at the interface points. As the Lagrangian points do not usually coincide with the Eulerian grid, information is passed back-and-forth between the two via a smeared, continuous approximation to the Delta function, and the henceforth discretization of the convolution operation.

Variants of explicit interface tracking differ in the implementation of the forcing function. In continuous forcing methods, the force term is involved in the variational formulation of the problem, i.e. a modified PDE problem is discretized. Examples, are the projection approach to IBM where the forcing function can be thought of as a Lagrange multiplier that enforcing the boundary constraint at the interface[9]. In the discrete forcing case, modifications are made to the discretized PDE problem to satisfy the interface constraints. Examples are the ghost-fluid method[6], and the cut-cell method.

This method was originally designed for use in the fluid-structure interaction of cardiac valve mechanics. In such problems, the arterial valves were modeled as one-dimensional elastic elements embedded in a fluid domain. The valves and vessel walls are tracked by the IBM and the forcing functions enforce a no-slip boundary condition at the interfaces. IBM has now been applied to various fluid-structure interaction problems. This is a natural fit as in standard fluid mechanics and solid mechanics, the bodies are already captured as Eulerian and Lagrangian domains, respectively, so the governing physics of these problems only need be modified by the forcing term.

2.1.2 Penetration Penalty and Lagrange Multiplier Methods

The Penetration Penalty and Lagrange Multiplier Methods are very closely tied to each other in their application to contact mechanics and in their formulation. In both methods, two distinct elastic body meshes are generated, one being the primary body and the other the secondary. At each time step, a search is performed to determine if the boundary nodes of the secondary body have penetrated into the primary body. The level of penetration, $\delta\mathbf{u}$, is then calculated and minimized to enforce a 'no-penetration' boundary condition. In the Penetration Penalty Method, the no-penetration condition is captured approximately by means of a restoring force applied to the penetrating node. The force is proportional in magnitude to the level of penetration. The resulting elastostatic weak form for the secondary body then takes the following form

$$\int_{\Omega_2} (\boldsymbol{\nabla} \cdot \frac{\partial \Psi(\boldsymbol{u})}{\partial \boldsymbol{u}}) \cdot \boldsymbol{v} d\Omega_2 - \int_{\Omega_2} \boldsymbol{b} \cdot \boldsymbol{v} d\Omega_2 - \int_{\partial\Omega_2} \boldsymbol{t} \cdot \boldsymbol{v} d\partial\Omega_2 - \int_{\partial\Omega_2} f(\max(0, \delta\boldsymbol{u})) \cdot \boldsymbol{v} d\partial\Omega_2 \qquad (2)$$

where $\Psi(\boldsymbol{u})$ is the strain energy density function of the material, and b and t are the body forces and surface tractions respectively. The magnitude of penalty force is calculated from some prescribed function $f(\delta\boldsymbol{u})$ and is known exactly from the degree of penetration. In the Lagrange Multiplier Method, as implied by the name, the no-penetration condition is enforced exactly as the restoring force is calculated as a Lagrange multiplier associated with the constraint. This yields the weak form

$$\int_{\Omega_2} (\boldsymbol{\nabla} \cdot \frac{\partial \Psi(\boldsymbol{u})}{\partial \boldsymbol{u}}) \cdot \boldsymbol{v} d\Omega_2 - \int_{\Omega_2} \boldsymbol{b} \cdot \boldsymbol{v} d\Omega_2 - \int_{\partial\Omega_2} \boldsymbol{t} \cdot \boldsymbol{v} d\partial\Omega_2 - \int_{\partial\Omega_2} \boldsymbol{\lambda} G(\delta \boldsymbol{u}) \cdot \boldsymbol{v} d\partial\Omega_2 \tag{3}$$

where $\boldsymbol{\lambda}$ and $G(\boldsymbol{x})$ are the Lagrange multipliers and constraint functions respectively.

2.2 Implicit Interface Representations

As the name suggests in implicit interface representation methods, the interface is tracked by means of an auxiliary function ϕ that is defined all over the computational domain. The interface is then defined as the points that satisfy some condition on ϕ, for example the zero level set of ϕ. As a consequence, the properties such as smoothness of the interface, and interface width are determined by ϕ.

Implicit interface methods are often preferred over explicit interface methods, multiple meshes aren't required, thus saving heaps of computational resources spent on meshing, remeshing, and passing information back and forth between meshes. Further, properties such as the the normal vector and curvature of the interface are easily computable in terms of the gradient of ϕ. Finally, material properties that may differ across the interface can be uniformly interpolated by considering them functions of ϕ.

The dynamics of the interface are embedded in the time-evolution equation of ϕ such that relevant physics is respected. Various implicit interface methods differ in what properties of the interface are respected, and how ϕ is evolved in time.

2.2.1 Level Set Method

In the level set method, the interface is defined as the zero level set of ϕ, $\Gamma = \{\boldsymbol{x} \in \Omega | \phi(\boldsymbol{x} = 0)\}$. For smoothly varying γ. The interface is 'sharp', i.e. has zero width. Such functions are usually found by considering the signed distance function (sdf) of a surface Γ in Ω.

$$\phi(\boldsymbol{x}) = \text{sdf}(\boldsymbol{x}) = \min_{\boldsymbol{x'} \in \Gamma} \left| \boldsymbol{x} - \boldsymbol{x'} \right| \tag{4}$$

The signed distance function should satisfy Eikonal equation, which states that the distance increase at unit rate.

$$|\boldsymbol{\nabla} \phi| = 1, \, \boldsymbol{x} \in \Omega \tag{5}$$

Signed distance functions can be easily computed for most geometries. Further, the time-evolution of Γ can be given by

$$\partial_t \phi(\boldsymbol{x}) = \boldsymbol{v}(\boldsymbol{x}) |\nabla \phi| \tag{6}$$

where Γ is evolving at speed $v(\boldsymbol{x})$ at point $\boldsymbol{x}$ in the gradient normal direction. The ease with which prescribed motion can be applied to the interface makes the level set method a favorite of the computer graphics community.

2.2.2 Phase Field Method

In the Phase Field method, the auxiliary variable ϕ, referred to as the phase parameter, is a smoothly varying function that takes values between zero and one. Regions α and β may be defined as regions where ϕ is equal to either value, and the interface is defined as the region where $\Gamma = \{\boldsymbol{x} \in \Omega | 0 < \phi(\boldsymbol{x}) < 1\}$. As ϕ is smooth, the interface has a finite width.

In the Phase Field Method, the physics of the interface is elegantly captured by energy-minimization principles as follows: the time-evolution equation of ϕ is determined by the negative gradient flows of a free-energy function $F(\phi)$ which enforces relevant physics at the interface.

$$\begin{aligned} \partial_t \phi &= -\frac{\delta F(\phi)}{\delta \phi} \\ F(\phi) &= \int_\Omega f_{\text{bulk}}(\phi) + \kappa |\nabla \phi|^2 + \dots \, \mathrm{d}x \\ f_{\text{bulk}} &= \rho(\phi - 0)^2(\phi - 1)^2 \end{aligned} \tag{7}$$

The free-energy functional dictates the properties of the interface. The bulk free energy, f_{bulk}, usually written as a double-well, has minima and $\phi = 0$, 1 with an 'activation' barrier in between whose height is dictated by ρ. The gradient flows of f_{bulk} are such that, away from the interface, ϕ is either 0, or 1, i.e. the material is in either phase. Further, the amount of energy required to jump the phase barrier is dictated by ρ. Secondly, the coefficient κ to the gradient energy term dictates the width of the interface. A larger κ penalizes gradients spanning large regions in the domain, thus reducing interface width.

The Phase Field Method is an attractive choice for our problem of interest due to many natural analogues, written in the language of energy minimization, between the variational formulations in fracture mechanics phase field.

2.2.3 Fictitious Domain Methods

Another class of methods developed for solving multiphysics problems in complex geometries are called 'Fictitious Domain Methods'. Such methods extend the computational domain Ω into a rectangular bounding box (for ease of meshing). The original domain Ω is differentiated from its fictitious fictitous extension by virtue of an smeared indicator function which evaluates to unity inside Ω, and smoothly vanishes outside.

$$\phi(\boldsymbol{x}) = I_\Omega(\boldsymbol{x}) = \begin{cases} 1 & \boldsymbol{x} \in \Omega \\ 0 & |\boldsymbol{x} - \Omega| > \delta \end{cases} \tag{8}$$

Governing equations are solved over the entire domain, though indicator function zeros out all field variables in the extended domain. The smooth extension endows the interface $\partial\Omega$ with a finite width equal to δ. As the interface may arbitrarily cut across the mesh, the Finite Cell Method subdivides the cut-cells until sufficient tolerance is reached for accurate differentiation and integration.

3 Fracture Mechanics

One application of the Phase Field Method outside of fluid-fluid interactions is in fracture mechanics, the study of how cracks and defects in solid materials develop, propagate, and ultimately lead to failure. This is an area with solid mechanics that is of great importance, with the safety of machines, their operators, and the general public relying on known failure points and failure modes within engineered systems. However, it has also been an area that is difficult to study because of its dramatic non-linearity and computation complexity, and the study of computational fracture mechanics is still an open question. Methods like the Phase Field Method can provide computational tractable methods with which to probe the stability of materials and investigate the development of potentially catastrophic failures. In this section, we will introduce the governing mechanics behind the mechanics of the healthy material, as well as introduce a recent vector-valued phase field model. Then this model is used to investigate how crack spread through a material in tension, and how stress is distributed within the material.

3.1 Governing Mechanics

The system is governed by the conservation of momentum within the intact material. This takes the form:

$$\boldsymbol{\nabla} \cdot \sigma - \rho \boldsymbol{b} = \rho \partial_{tt} \boldsymbol{u} \tag{9}$$

where σ is the Cauchy stress tensor, $\boldsymbol{b}$ are the body forces, and $\boldsymbol{u}$ are the displacements of the solid material. Here, the material will be assumed to be linearly elastic such that the Cauchy stress can be expressed as:

$$\sigma = \lambda \operatorname{tr} \varepsilon I + 2\mu\varepsilon \tag{10}$$

where λ and μ are the Lamé constants, I is the indentity tensor, and

$$\varepsilon = \frac{1}{2}\left(\boldsymbol{\nabla u} + \boldsymbol{\nabla u}^T\right) \tag{11}$$

For this investigation, the system will be studied in equilibrium states for each applied load such that $\partial_{tt} u = 0$ and with the gravitational forces parallel to the face of fracture such that they can be neglected in the calculation. This results in a simplified strong form of the system given as:

$$\lambda \boldsymbol{\nabla} \cdot \operatorname{tr}(\varepsilon I) + 2\mu \boldsymbol{\nabla} \cdot \varepsilon = 0 \tag{12}$$

. In this quasistatic formulation, this problem can be reexamined as an global energy minimization, where the elastic energy functional for the material is calculated as:

$$E[\boldsymbol{u}] = \int_\Omega \sigma(\boldsymbol{u}) : \varepsilon(\boldsymbol{u})\, dV - \int_{\partial\Omega} \boldsymbol{t} \cdot \boldsymbol{u}\, dS \tag{13}$$

where Ω is the domain of the elastic body, $\partial\Omega$ is the boundary of that body, and t are the surface tractions. This can then be solved using variational approaches.

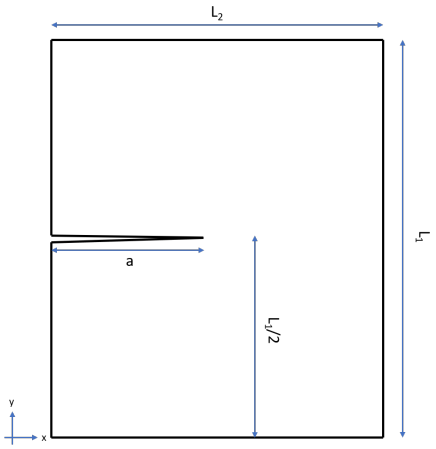

Figure 1: Fracture mechanics governing geometry.

3.2 Governing Geometry and Boundary Conditions

The system will consist of a rectangular plane of elastic material with dimension L_1xL_2 with an initial crack of length a at a height of $L_1/2$ (Fig. 1). The bottom border of the plate ($y = 0$) will be clamped in both the x and y directions ($u_y(y = 0) = u_x(y = 0) = 0$). Uniform displacements will be applied along the top surface to deform the plate ($u_y(y = L_1) = dL$). The system will begin in a uniform stress-free, deformation-free state ($\sigma(t = 0^-) = 0$, $\varepsilon(t = 0^-) = 0$), and dL will be incremented, allowing both the mechanical system and the phase field to come to steady state in between increments. Along the cracked surface, the material is unstressed normal to the surface of the crack ($\sigma_c \cdot \boldsymbol{n_c} = 0$, where c indicates surface elements along the crack); this is implicitly enforced by the phase field variable.

3.3 Phase Field Model of Fracture Mechanics

The phase field method will be used to track the propagation of the crack for increases of the load σ_L. Traditionally, the phase field contains a scalar indicator variable that takes on a continuous value between 0 and 1, with 1 indicating full healthy material, and 0 indicating fully cracked material. Additional energy terms are added to account for the energy of fracture and to enforce the stress-free boundary condition. [3] proposed the following model, where the phase field contains the vector $\boldsymbol{d}$ capturing the surface normal of the crack, where $|\boldsymbol{d}| \approx 0$ where the material is healthy and $|\boldsymbol{d}| \approx 1$ where the material is fully cracked. In addition to the surface energy terms, a strain energy density is added for the cracked region to enforce the correct stress state for different loading conditions. In total, the energy functional takes the form:

$$E[\boldsymbol{u}, \boldsymbol{d}] = \int_\Omega \Big(((1 - |\boldsymbol{d}|)^2 + \eta_\epsilon)W(\nabla \boldsymbol{u}) + (1 - (1 - |\boldsymbol{d}|)^2)W_d(\nabla \boldsymbol{u}, \boldsymbol{d}) + G_c \left(\frac{|\boldsymbol{d}|^2}{2\epsilon} + \frac{\epsilon}{2} |\nabla \boldsymbol{d}|^2 \right) dV \tag{14}$$

Here, the first term is the strain energy in the material ($W(\nabla \boldsymbol{u}) = \sigma(\nabla \boldsymbol{u}) : \varepsilon(\nabla \boldsymbol{u})$ for linearly elastic materials, but can be an arbitrary hyperelastic model), the second term is the strain energy of the cracked region, and the third term captures the surface energy. While the material model of the healthy material can be arbitrary, [3] showed certain constraints are required on W_d to maintain physically accurate loading responses. The η_ϵ term is added to retain ellipticity where $|\mathbf{d}| \approx 1$ [3].

To solve this system for a quasistatic loading condition, [3] implemented an iterative approach, where they first solve the displacement minimization for a fixed crack and then solve the crack for this new displacement. This cycle is then repeated until the two solutions converge.

3.4 Implementation and Computational Experiments

The above problem was implemented in the open source FEniCS [1] finite elements framework. For all simulations: $L_1 = L_2 = 1$ m. The value of a was varied to investigate the effect of initial crack size on the resultant stress state. Structured triangular meshes were used for all of the simulations, with the region surrounding the midline (where the crack would be forming) being refined in the same manner as in [3].

For these simulations, the same material models from [3] were used, with the strain energy in the healthy material being a compressible Neohookean material, with shear modulus μ chosen to reflect the stiffness of PMMA, and the λ set to 0 to avoid Poisson effects. The parameters were then normalized to by μ. All parameters associated with the phase field model were borrowed from [3]. To prevent the crack in the material from healing, the same algorithm from [3] was implemented. Briefly, at each step in the displacement-crack iteration, the crack vector at each point was modified such that if the point had been previously crack, it remained cracked.

To investigate the effect of initial crack size on the stress state of the material, the value of a was varied for a given ramp displacement of the top boundary ($a \in [0.125, 0.25, 0.375, 0.5]$ m). For all values of a, the top boundary was displaced by a value of $0.003 * t$, where t is the incremented time step. The time step was decreases from 0.1 to 0.01 and then to 0.01 as the simulation progressed and the crack began to propagate quickly through the materal. The timestep was assumed converged when the maximum change in either variable was less than 10^{-6}. For each simulation, the von Mises stress state, displacement field, and phase field were saved, and the total force applied to the top edge were calculated.

4 Results and Discussion

4.1 Simulations

All simulations were run using CPU parallelization (MPI standard) on 6 cores of an AMD Ryzen 5 5600 CPU. For each time step, the iteration between solving the displacement and crack problems was continued until both functions changed by less than 10^6. For early timesteps and low forces, this was typically achieved in under 5 iterations. However, as loads increased and the crack began to propagate, the number of iterations that would be required began to grow dramatically, with the final time step before catastrophic failure often taking >1000 iterations to solve. In general, for time

steps that resulted in larger changes in the crack length, more iterations were required to converge. As the number and size of cracks being considered increased, this number of iterations per time step would likely also increased. However, the current project only considered single cracks.

4.2 von Mises Stress State

For all simulations and initial crack values, the von Mises stress was calculated at each time point. Investigating the stress state immediately before catastrophic failure (where the crack propagates to the end of the material), common visual features can be identified (Fig. 2). Inside the crack and in a triangular region spanning from the crack tip to the crack origin edge, the material is stress-free (Fig. 2 (center)). This agree with analytical models that show the stress isocurves wrapping around the tip of the crack, but not residing in the triangular region (Fig. 2(right), from [8]). Additionally, a stress concentration is developing at the crack tip in agreement with analytical analysis.

One of the isocurves can also be identified as a propagating front that aligns with the tip of the crack. One feature that was common throughout the simulation but was not present in analytical model was a ring-shaped, propogating iso-stress contour that contacts the crack tip (Fig. 2 (center)). The area of this ring increases with increased load until the crack begins to propagate to the right. This ring is at least partially an artifact of the particular phase field model used here, as view of the crack vector field shows a region of partially damaged material in this same ring shape. However, this could reflect ares of the material that have been physically weakened and are more prone to cracking. This would require experimental validation. The above features can be seen throughout the different initial crack lengths (Fig. 3 and seem to be characteristic of cracked stress states.

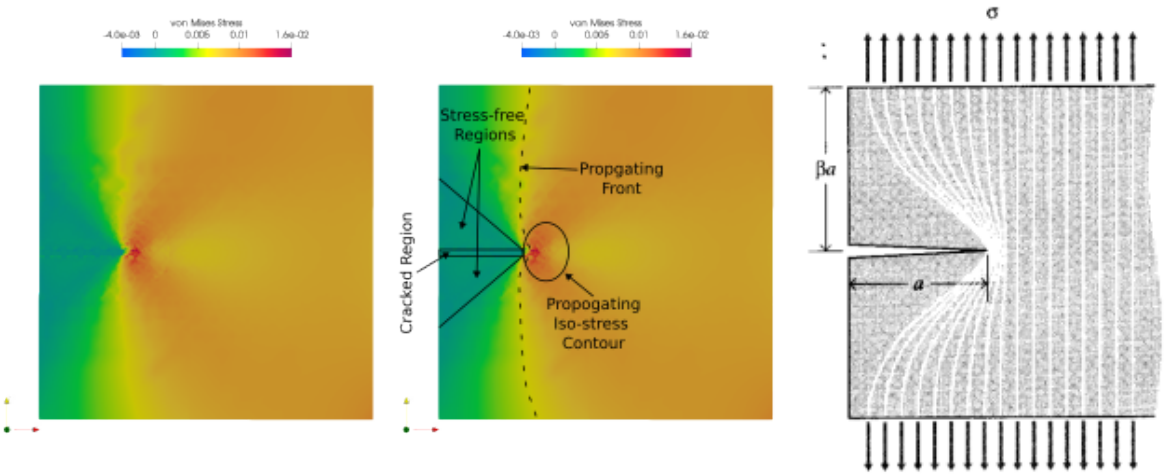

Figure 2: Structure of the Fractured Material Stress State. (left) Von Mises Stress in the material with an initial crack of 25 cm in the time step before catastrophic failure. (center) Annotated version of (left). (right) Idealized stress state for the case of a perfectly sharp crack.

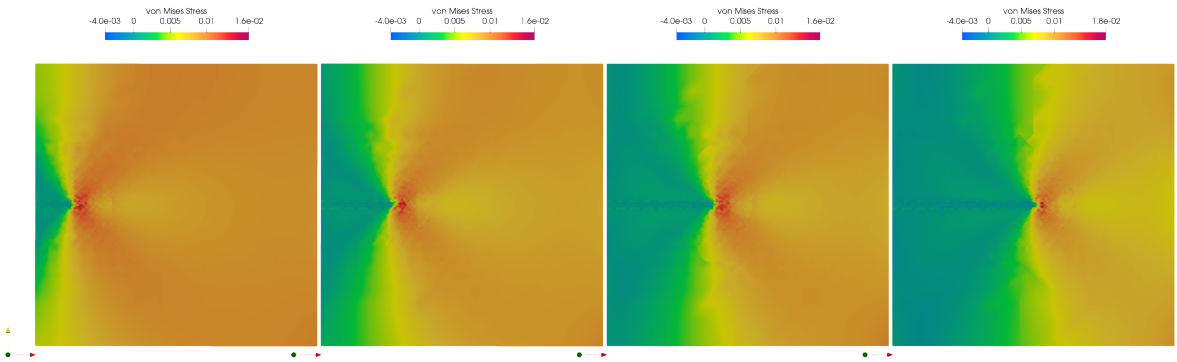

Figure 3: Comparison of stress states immediately before catastrophic failure for solids with different initial crack lengths a. (left to right) $a = [12.5, 25, 37.5, 50]$cm.

4.3 Crack Propagation

Through these simulations, the propagation of formed cracks can be followed. Fig. 4 shows the four frames leading up through the catastrophic failure of a sample. Here the initial crack length was 25 cm. Moving from left to right, it is observed that the previously identified propagating stress front maintains a mostly consistant shape as it travels with the moving crack tip. The area of the triangular stress-free region also continues to expand, with the legs of the triangle maintaining a constant angle relative to the crack face. When the material finally fails (as shown in the far right panel), all stress in the material is relieved. These simulations also capture the speed with which these fractures propagate, with the material failing within four frames (< 40ms). Similar rates of failure were seen for different initial crack lengths.

4.4 Effect of Initial Crack Length

From these initial simulations, some effect of initial crack length can be observed. As would be expected, for increased initial crack length, the effective spring stiffness of the material decreases (proportional to the slope of the lines in Fig. 5 (left)). This is expected as spring stiffness will be proportional to the Young's modulus times the cross-sectional area of the sample, and will be dominated by the weakest area (for this series elasticity set up). This means that the area adjacent to the crack (with the narrowest cross-section) will be very impactful to the overall spring stiffness, with a larger crack leading to a smaller cross-section and therefore a lower spring stiffness. This spring stiffness as decreases relative to the initial value as the crack propagates, which can be seen in the force response diverging from the initial slope (shown in a dashed line of the same color, Fi5). These trends were consistent across all initial crack lengths.

What was not consistent was the onset of propagation (Fig. 5 (right)). Here, it would be expected that the crack would begin to propagate for lower forces with increasing initial crack length (for longer cracks the spring stiffness is lower in the adjacent region relative to the intact material, so stresses would be greater, and the tension has a larger moment arm against the crack tip). While this trend was seen in general, one of the initial crack lengths differed. Specifically, in the case of an initial crack length of 37.5 cm, the crack widened at a lower force than in the case of a 50 cm initial crack. However, this initial widening did not lead to catastrophic failure, and the sample

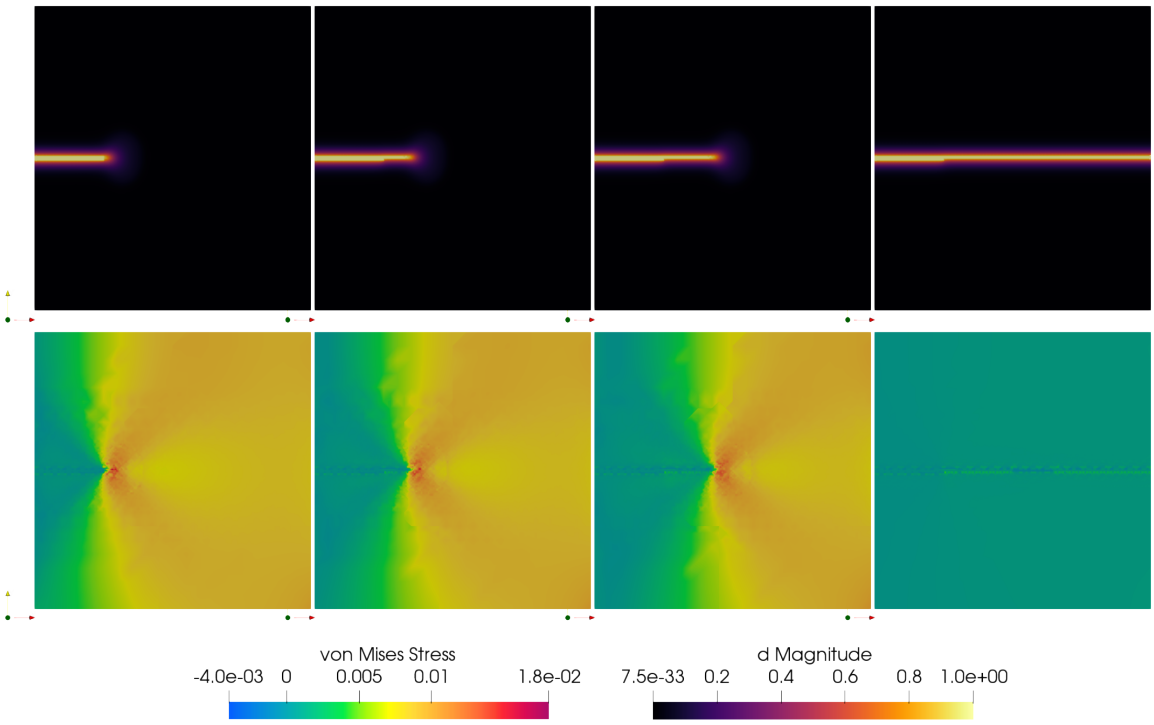

Figure 4: Propagation of a crack leading to catastrophic failure. All frames come from the simulation of a sample with an initial crack of 25 cm. (top) Magnitude of phase field vector. (bottom) von Mises stress state associated with the above crack vector. Each column shows a sequential time step leading to catastrophic failure (far right).

was able to hold on for longer than the 50 cm initial crack sample. This suggests that increase of the onset of any crack propogation being related to initial crack length, it may instead be related to the force leading to catastrophic failure.

5 Conclusions

In this project, we reviewed six different computational methods for tracking the evolving boundaries between two material phases. These methods were categorized into explicit- and implicit-boundary representations. We then implemented a recently developed phase field model within the realm of fracture mechanics. This model was able to successfully (though here qualitatively) replicate results from analytical fracture mechanics. It also suggested relationships between the initial crack length in a material and the force required to begin catastrophic crack propagation. While phase field methods like the one presented here have been utilized multiple times in the study of fracture mechanics and small deformations, there is still much room for the application of phase field models within solid mechanics and solid-solid boundary interactions, particularly within contact mechanics.

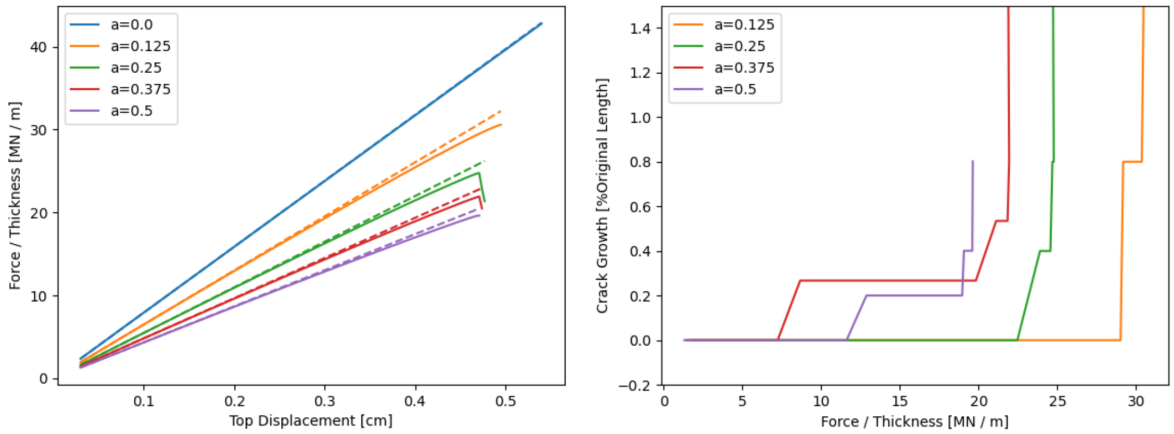

Figure 5: Comparison of Different Initial Crack Length Simulations. (left) Force v. Displacement response. (right) Crack Growth v. Force response. For both figures, the force is calculated from 2D stress states, and is therefore reported as force per unit thickness. Legends indicate the initial crack length in m.

References

[1] Martin S Alnaes et al. *The FEniCS Project Version 1.5*. URL: http://fenicsproject.org..

[2] Tino Bog et al. "Normal contact with high order finite elements and a fictitious contact material". In: *Computers and Mathematics with Applications* 70 (7 2015), pp. 1370–1390. ISSN: 08981221. DOI: 10.1016/j.camwa.2015.04.020. URL: http://dx.doi.org/10.1016/j.camwa.2015.04.020.

[3] Maryam Hakimzadeh et al. "Phase-field finite deformation fracture with an effective energy for regularized crack face contact". In: *Journal of the Mechanics and Physics of Solids* 167 (c 2022), pp. 1–45. ISSN: 00225096. DOI: 10.1016/j.jmps.2022.104994.

[4] Haobo Hua, Jaemin Shin, and Junseok Kim. "Level Set, Phase-Field, and Immersed Boundary Methods for Two-Phase Fluid Flows". In: *ASME, Journal of Fluids Engineering* 136 (Feb. 2014), p. 201301. DOI: 10.1115/1.4025658.

[5] Taejin Jang et al. "BattPhase—A Convergent, Non-Oscillatory, Efficient Algorithm and Code for Predicting Shape Changes in Lithium Metal Batteries Using Phase-Field Models: Part I. Secondary Current Distribution". In: *Journal of The Electrochemical Society* 169 (Aug. 2022). DOI: 10.1149/1945-7111/ac86a7.

[6] Xu-Dong Liu, Ronald Fedkiw, and Myung joo Kang. "A Boundary Condition Capturing Method for Poisson's Equation on Irregular Domains". In: *Journal of Computational Physics* 160 (2000), pp. 151–178.

[7] P. Papadopoulos and J. M. Solberg. "A Lagrange multiplier method for the finite element solution of frictionless contact problems". In: *Mathematical and Computer Modelling* 28 (4-8 1998), pp. 373–384. ISSN: 08957177. DOI: 10.1016/S0895-7177(98)00128-9.

[8] David Roylance. *Introduction to Fracture Mechanics*. 2001.

[9] Kunihiko Taira and Tim Colonius. "The immersed boundary method: a projection approach". In: *Journal of Computational Physics* 225.2 (2007), pp. 2118–2137.

[10] P. Wriggers. *Computational contact mechanics, second ed., Springer*. 2006. ISBN: 9783540326083.