

A CutFEM method for Stefan-Signorini problems with application in pulsed laser ablation.

Susanne Claus*, Samuel Bigot, Pierre Kerfriden[†]

Cardiff University, School of Engineering,
The Parade, CF243AA Cardiff, United Kingdom.

August 15, 2018

Abstract

In this article, we develop a cut finite element method for one-phase Stefan problems with applications in laser manufacturing. The geometry of the workpiece is represented implicitly via a level set function. Material above the melting/vaporisation temperature is represented by a fictitious gas phase. The moving interface between the workpiece and the fictitious gas phase may cut arbitrarily through the elements of the finite element mesh, which remains fixed throughout the simulation, thereby circumventing the need for cumbersome re-meshing operations. The primal/dual formulation of the linear one-phase Stefan problem is recast into a primal non-linear formulation using a Nitsche-type approach, which avoids the difficulty of constructing inf-sup stable primal/dual pairs. Through the careful derivation of stabilisation terms, we show that the proposed Stefan-Signorini-Nitsche CutFEM method remains stable independently of the cut location. In addition, we obtain optimal convergence with respect to space and time refinement. Several 2D and 3D examples are proposed, highlighting the robustness and flexibility of the algorithm, together with its relevance to the field of micro-manufacturing.

CutFEM, Stefan problem, Stefan-Signorini-Nitsche formulation, pulsed laser ablation.

1 Introduction

The simulation of phase changes requires tracking the evolution of solid/liquid and liquid/gas interfaces, which is numerically challenging. In the context of the finite element

*ClausS2@cardiff.ac.uk

[†]pierre.kerfriden@gmail.com

method (FEM), two main approaches for interface tracking can be distinguished. The first family of approaches smooths the transition between phases, allowing for the existence of a mushy region in space where both phases coexist (*i.e.* enthalpy method [62, 25, 3, 24], phase field method [57, 64]). The width of this region may be thought of as a trade-off between computational cost, which is lower for fatter transition zones, and modelling accuracy, whereby the “true” model corresponds to an infinitely thin transition zone. The second approach describes the interface between phases as a sharp surface in 3D or a line in 2D. Although this may seem to be the “natural” approach to interface tracking, the sharp interface approach is difficult to handle within a finite element context. Indeed, the mesh either needs to conform to this interface, leading to a class of moving mesh algorithms such as ALE, or special finite element methods need to be developed so as to allow the interface to *cut through* the element. The latter family of methods are the so-called implicit boundary methods (see for instance [47, 6, 7, 31, 37, 12]), which are of prime interest in this paper.

The XFEM method was proposed in [47], and relies on a partition-of-unity enrichment to represent embedded kinks and discontinuities. The XFEM method has been applied to the simulation of two-phase Stefan problems in *e.g.* [45, 15, 63, 28, 53, 4, 22, 44, 39]. In this case, the interface between solid and liquid moves through a regular background mesh, which may be refined around the interface for accuracy purposes, but does not need to conform to it. XFEM methods offer increased efficiency and robustness as they do not require the cumbersome re-meshing operations that are used in mesh-moving algorithm to prevent the development of prohibitively large mesh deformations. As an alternative to XFEM, the CutFEM approach [36, 37, 12] also enriches elements in order to allow for the representation of embedded discontinuities. However, the enrichment is obtained by an overlapping domain-decomposition strategy (*i.e.* a “fictitious domain” approach). The strength of CutFEM lies in its stability, which is provided by the so-called “ghost-penalty” regularisation [9]. As far as we are aware, CutFEM has never been applied to Stefan problems.

In this paper, we are interested in a particular subclass of phase change problems with sharp interface representation: the one-phase Stefan problem. In this particular setting, only one of the phases is represented and the other phase is replaced by a fictitious material with zero specific heat. Subsequently, the fictitious phase does not contribute to the energy balance. The interface between the represented phase and the fictitious phase is moved so that the flux at the boundary of the represented domain is balanced by a latent heat term. It is possible to include a non-zero energy flux applied locally at the phase change interface. This is routinely done in laser manufacturing to simulate the irradiation of the ablated material (see [58]). The boundary conditions of the one-phase Stefan problem are ambiguous and can be treated mathematically using the same tools that are used to formulate unilateral contact in solid mechanics. Mathematical considerations relative to the Stefan-Signorini problem can be found in [30, 40, 60].

Very few implicit boundary methods have been developed for one-phase Stefan problems. One exception is the elegant Stefan-Signorini formulation proposed in [50] for the simulation of thermal plasma cutting, associated with an implicit representation of the do-

main boundary through the evolution of a level-set function. However, unilateral contact problems have been extensively studied in XFEM [26, 42, 52, 29, 32, 48] and CutFEM [19, 11]. Typical embedded interface formulations of contact laws include the penalty method, the Lagrange multiplier approach, and the Nitsche-contact formulation, which was recently proposed in [16]. Lagrange multiplier approaches are usually solved by either the augmented Lagrangian algorithm [61, 51, 13], the Uzawa algorithm or the LaTIn approach [41, 1, 19], all of them being some form of proximal algorithms. Alternatively, the Nitsche-contact formulation reformulates the KKT primal/dual contact problem as a purely primal nonlinear problem that can be solved by Newton algorithms [14, 16, 17, 18]. The Nitsche-contact algorithm promises consistency, whilst circumventing the cumbersome choice of an inf-sup-stable pair for the primal and dual finite element spaces.

In this paper, we present the first CutFEM algorithm for phase-change problems with sharp embedded representation of moving interfaces. The proposed algorithm is highly flexible owing to the implicit description of the domain geometry by a level-set function. The most novel part of the algorithm resides in rewriting the primal/dual condition associated with the interface of the one-phase Stefan-Signorini problem using a dedicated Nitsche reformulation, inspired by [14, 16] and presented in Section 2 and 3. Noteworthily, we provide the expression of the tangent operator required to deploy the Newton algorithm. In addition, our derivation of the Nitsche-Signorini formulation of the Stefan problem departs from those proposed in [18, 14], which provides new insights into this emerging approach. Consistently with the CutFEM paradigm, cut elements are regularised using ghost penalties [9], which are carefully adapted to the context of the one-phase Stefan-Signorini problem. We pay particular attention to the mathematical scaling of all stabilisation terms so as to obtain the optimal trade-off between stability and accuracy. This is described in Section 4 of the paper. Our numerical time integration procedure is relatively classical: we use an implicit Euler algorithm to solve the unsteady temperature equation. At every time step, the interface is moved by advecting the level-set function using the velocity field delivered by the Nitsche-Signorini algorithm and extended to the entire domain using the fast-marching method [56]. The advection of the level-set function is stabilised by the SUPG method [8]. This is detailed in Section 5. An alternative approach where the authors use an upwind finite-difference scheme to update the level-set can be found in [27].

Our formal developments are accompanied by a high-performance computer implementation. The core of our implementation is the finite element C++/Python library FEniCS [2, 35], which, in particular, proposes a range of high-level tools to rapidly developed finite element solvers. The CutFEM C++ library, LibCutFEM, partially described in [12], defines additional tools that are specific to unfitted finite elements. This library forms the basis for the CutFEM Stefan-Signorini code used in this article.

Several 2D and 3D examples are presented in Section 6, with particular relevance to engineers interested in the simulation of laser micro-milling. We first derive a new 2D manufactured analytical solution for the one-phase Stefan-Signorini problem, which we use to validate our numerical algorithm and show optimal convergence. In particular, we show that the convergence of the proposed algorithm is optimal: order two in space and order one in time. We then move to the 2D simulation of a pulsed laser ablation process,

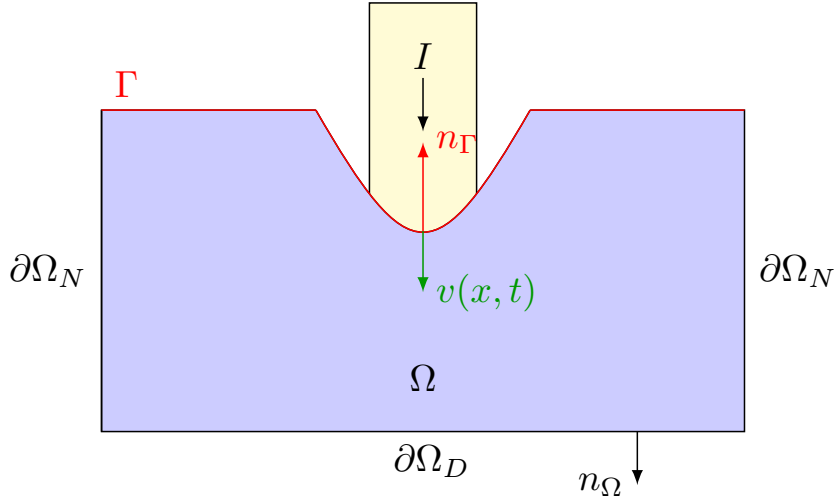


Figure 2.1: Schematics of a one-phase Stefan problem with energy flux I heating material Ω at Γ .

where we compare the effect of different pulse frequencies onto the finishing quality of the ablated surface. Our first 3D example is the simulation of a laser drilling operation, where the laser irradiates the material. Finally, we present a 3D example of laser milling, where material is removed by the laser through a complex path, following a layer-by-layer removal strategy.

2 The Stefan-Signorini problem

Let us assume that an energy flux $I(x, t)$ (laser beam) heats a piece of material occupying domain $\Omega(t)$ in \mathbb{R}^d ($d = 2$ or 3) in the time interval $t \in [t_0, t_f]$. Here, t_0 is the initial time and t_f is the final time. The boundary of $\Omega(t)$, denoted by $\partial\Omega(t)$, is decomposed into a Dirichlet part $\partial\Omega_D$ and a Neumann part $\partial\Omega_N$ as well as a moving boundary $\Gamma(t)$ (see Figure 2.1). The boundary $\Gamma(t)$ can be interpreted as an interface between a heated material and the fictitious fluid and gas phase, which will not be represented explicitly. Instead we assume that material above the melting temperature is instantaneously removed, latent heat being consumed in the process.

The Stefan-Signorini problem describing the heat conduction and removal of material can then be formulated as: For all $t \in [t_0, t_f]$, find the temperature $T : \Omega(t) \rightarrow \mathbb{R}$ such that

$$\rho c \frac{\partial T}{\partial t} - k \Delta T = f \quad \text{in } \Omega(t) \quad (2.1)$$

with boundary conditions

$$\begin{aligned} T &= T_D && \text{on } \partial\Omega_D(t), \\ k \nabla T \cdot n_\Omega &= q_N && \text{on } \partial\Omega_N(t), \end{aligned} \quad (2.2)$$

together with the evolution equation for interface $\Gamma(t)$ between the fictitious material and the heated material

$$I(x, t) \cdot n_\Gamma - k \nabla T \cdot n_\Gamma = -\rho L (v(x, t) \cdot n_\Gamma) \quad \text{on } \Gamma(t) \quad (2.3)$$

and the associated Signorini conditions

$$\begin{aligned} k \nabla T \cdot n_\Gamma - I \cdot n_\Gamma &\leq 0 && \text{on } \Gamma(t), \\ T - T_m &\leq 0 && \text{on } \Gamma(t), \\ (k \nabla T \cdot n_\Gamma - I \cdot n_\Gamma) \perp (T - T_m) &&& \text{on } \Gamma(t), \end{aligned} \quad (2.4)$$

and the initial condition

$$T(x, t_0) = T_0 \quad \text{in } \Omega(t_0), \quad (2.5)$$

where T_0 is a specified initial temperature with $T_0 < T_m$. Here, ρ is the mass density, c is the heat capacity, k is the thermal conductivity, f is a volumetric heat source, T_D is a given temperature, q_N is a given heat flux, L is the latent heat, v is the velocity of the phase boundary $\Gamma(t)$, T_m is the melting temperature, and I is the prescribed heat flux given by the profile

$$I(x, t) = A(x, t) e_{ray}(x, t), \quad (2.6)$$

where A is the amplitude of the laser beam and e_{ray} is the direction of the beam.

Remark 2.1 *The Signorini conditions (2.4) ensure that material is only removed if it reaches melting temperature and that material is only removed and not added. It can be interpreted as enforcing either*

(a) $T = T_m$ and $\rho L v(x, t) \cdot n_\Gamma \leq 0$, i.e. the material is heated to melting temperature and material is removed in the normal direction to the interface with speed $v(x, t)$

or

(b) $T < T_m$ and $\rho L v(x, t) \cdot n_\Gamma = 0$, i.e. the material does not reach melting temperature and hence no material is removed.

3 Signorini-Nitsche reformulation of the one-phase Stefan problem

3.1 Primal/dual weak formulation of the Stefan-Signorini problem

The weak formulation of the Stefan-Signorini problem reads: For all $t \in [t_0, t_f]$, find $T \in H_D^1(\Omega(t))$ such that, for all $\delta T \in H_0^1(\Omega(t))$,

$$\rho c \left(\frac{\partial T}{\partial t}, \delta T \right)_{\Omega(t)} + (k \nabla T, \nabla \delta T)_{\Omega(t)} = (f, \delta T)_{\Omega(t)} + (k \nabla T \cdot n_\Gamma, \delta T)_{\Gamma(t)} + (q_N, \delta T)_{\partial\Omega_N(t)} \quad (3.1)$$

Together with (2.3), (2.4) and (2.5). Here,

$$\begin{aligned} H_D^1(\Omega(t)) &= \{T \in H^1(\Omega(t)) : T = T_D \text{ on } \partial\Omega_D(t)\}, \\ H_0^1(\Omega(t)) &= \{T \in H^1(\Omega(t)) : T = 0 \text{ on } \partial\Omega_D(t)\}. \end{aligned} \quad (3.2)$$

In order to facilitate the treatment of Signorini conditions (2.4), we introduce a slack variable

$$\sigma := k \nabla T \cdot n_\Gamma - I \cdot n_\Gamma, \quad (3.3)$$

which yields the weak form

$$\begin{aligned} \rho c \left(\frac{\partial T}{\partial t}, \delta T \right)_{\Omega(t)} + (k \nabla T, \nabla \delta T)_{\Omega(t)} &= (f, \delta T)_{\Omega(t)} + (q_N, \delta T)_{\partial\Omega_N(t)} \\ &+ (\sigma, \delta T)_{\Gamma(t)} + (I \cdot n_\Gamma, \delta T)_{\Gamma(t)} \end{aligned} \quad (3.4)$$

with the modified Signorini conditions

$$\begin{aligned} \sigma &\leq 0 \quad \text{on } \Gamma(t), \\ T - T_m &\leq 0 \quad \text{on } \Gamma(t), \\ \sigma &\perp (T - T_m) \quad \text{on } \Gamma(t). \end{aligned} \quad (3.5)$$

We define bilinear form

$$a(T, \delta T) := \rho c \left(\frac{\partial T}{\partial t}, \delta T \right)_{\Omega(t)} + (k \nabla T, \nabla \delta T)_{\Omega(t)} \quad (3.6)$$

and linear form

$$l(\delta T) := (f, \delta T)_{\Omega(t)} + (q_N, \delta T)_{\partial\Omega_N(t)} + (I \cdot n_\Gamma, \delta T)_{\Gamma(t)}. \quad (3.7)$$

The weak formulation of the Stefan-Signorini problem then reads: For all $t \in [t_0, t_f]$, find $T \in H_D^1(\Omega(t))$ such that, for all $\delta T \in H_0^1(\Omega(t))$,

$$a(T, \delta T) - (\sigma, \delta T)_{\Gamma(t)} = l(\delta T) \quad (3.8)$$

with the Signorini law (3.5) and (2.3) and initial conditions (2.5). Existence and uniqueness of the Stefan-Signorini problem are discussed in [30, 49].

3.2 Nonlinear Nitsche Reformulation

First, following [23, 17, 14], let us reformulate the Signorini law (3.5) as

$$\sigma = -\frac{1}{\gamma} [(T - T_m) - \gamma \sigma]_+, \quad (3.9)$$

where γ is a positive penalty parameter and $[\cdot]_+$ denotes the positive part of a scalar quantity $x \in \mathbb{R}$, i.e.

$$[x]_+ = \begin{cases} x & \text{if } x > 0, \\ 0 & \text{otherwise.} \end{cases} \quad (3.10)$$

Remark 3.1 *The equivalence of (3.9) and (3.5) can be proved by enumeration.*

Let us first show that (3.9) implies (3.5). Consider the following two complementary cases:

Case 1: $[(T - T_m) - \gamma\sigma] \geq 0 \Rightarrow \sigma = -\frac{1}{\gamma}((T - T_m) - \gamma\sigma) \Rightarrow T - T_m = 0$. Now, returning to the first statement, $[0 - \gamma\sigma] \geq 0$ also indicates that $\sigma \leq 0$. As $\sigma \cdot (T - T_m) = \sigma \cdot 0 = 0$, the Signorini law (3.5) is satisfied in case 1.

Case 2: $[(T - T_m) - \gamma\sigma] \leq 0 \Rightarrow \sigma = 0$. Now, returning to the previous statement, $[(T - T_m) - \gamma\sigma] \leq 0$ means that quantity $T - T_m$, which can be non-zero, is necessarily negative. Finally $\sigma \cdot (T - T_m) = 0 \cdot (T - T_m) = 0$, and therefore Signorini law (3.5) is satisfied in case 2. These two cases are illustrated in Figure 3.1.

Let us now show that (3.5) implies (3.9). Consider the three following cases:

Case 1: $\sigma < 0$. Owing to the consistency condition, this can only happen when $T - T_m = 0$. Therefore, we can write $\sigma = -\frac{1}{\gamma}(-\gamma\sigma) = -\frac{1}{\gamma}[0 - \gamma\sigma]_+ = -\frac{1}{\gamma}[T - T_m - \gamma\sigma]_+$.

Case 2: $T - T_m < 0$. This can only happen when $\sigma = 0$. Therefore, we can write that $\sigma = 0 = -\frac{1}{\gamma}[T - T_m]_+ = -\frac{1}{\gamma}[T - T_m - \gamma\sigma]_+$.

Case 3: The last possible scenario is $\sigma = T - T_m = 0$. In this case, we can write that $\sigma = 0 = -\frac{1}{\gamma}[0]_+ = -\frac{1}{\gamma}[T - T_m - \gamma\sigma]_+$. \square

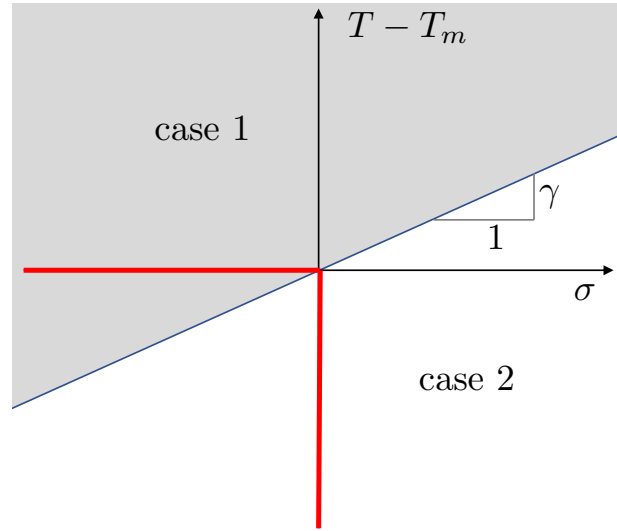


Figure 3.1: Illustration of the different formulations of the Signorini law.

A Nitsche formulation of the Stefan-Signorini problem (3.8) is obtained by replacing slack variable σ by its expression as a function of the primal variable T ,

$$\sigma(T) = -\frac{1}{\gamma} [(T - T_m) - \gamma(k\nabla T \cdot n_\Gamma - I \cdot n_\Gamma)]_+ . \quad (3.11)$$

A penalty term enforcing this expression weakly can be formulated as follows

$$s_{\heartsuit}(T, \delta T) := \int_{\Gamma(t)} ((k \nabla T \cdot n_{\Gamma} - I \cdot n_{\Gamma}) - \sigma(T)) (\theta_1 \delta T - \theta_2 \gamma k \nabla \delta T \cdot n_{\Gamma}) d\Gamma, \quad (3.12)$$

where the choice of $\theta_1 \in \{0, 1\}$ and $\theta_2 \in \{-1, 0, 1\}$ leads to a family of different methods as detailed below. Finally, the proposed *Stefan-Signorini-Nitsche* formulation reads

$$a(T, \delta T) - (k \nabla T \cdot n_{\Gamma} - I \cdot n_{\Gamma}, \delta T)_{\Gamma(t)} + s_{\heartsuit}(T, \delta T) = l(\delta T). \quad (3.13)$$

To simplify the notations, let us define

$$P_{\gamma}(T) := (T - T_m) - \gamma(k \nabla T \cdot n_{\Gamma} - I \cdot n_{\Gamma}), \quad (3.14)$$

and the parametrised variation of this quantity, which we define as

$$P_{\theta\gamma}^{\delta}(\delta T) := \theta_1 \delta T - \gamma \theta_2 k \nabla \delta T \cdot n_{\Gamma}. \quad (3.15)$$

Using these notations, the penalty term s_{\heartsuit} reads as

$$s_{\heartsuit}(T, \delta T) = \int_{\Gamma(t)} \left((k \nabla T \cdot n_{\Gamma} - I \cdot n_{\Gamma}) + \frac{1}{\gamma} [P_{\gamma}(T)]_+ \right) P_{\theta\gamma}^{\delta}(\delta T) d\Gamma. \quad (3.16)$$

And the proposed Nitsche formulation can be expressed as a sum of linear and nonlinear terms,

$$a(T, \delta T) + (k \nabla T \cdot n_{\Gamma}, P_{\theta\gamma}^{\delta}(\delta T) - \delta T)_{\Gamma(t)} + \mathcal{N}(T, \delta T) = l(\delta T) + (I \cdot n_{\Gamma}, P_{\theta\gamma}^{\delta}(\delta T) - \delta T)_{\Gamma(t)}, \quad (3.17)$$

where

$$\mathcal{N}(T, \delta T) = \frac{1}{\gamma} ([P_{\gamma}(T)]_+, P_{\theta\gamma}^{\delta}(\delta T))_{\Gamma(t)}. \quad (3.18)$$

We emphasise the fact that \mathcal{N} is nonlinear in its first argument.

Three interesting Nitsche formulations are obtained by choosing particular values for the (θ_1, θ_2) pair, as follows.

- $\theta_1 = 1$ and $\theta_2 = 1$, $\gamma > 0$: Symmetric Nitsche method.
- $\theta_1 = 1$ and $\theta_2 = -1$, $\gamma > 0$: Non-symmetric Nitsche method. A closely related formulation was proposed in [17] to solve problems of unilateral contact between deformable solids. It was shown that, as opposed to the symmetric Nitsche formulation, the stability of this non-symmetric variant is preserved irrespectively of the value of Nitsche parameter γ .
- $\theta_1 = 1$ and $\theta_2 = 0$, $\gamma > 0$: Consistent penalty formulation as described in [23].

- $\theta_1 = 0$ and $\theta_2 = -1$: Semi penalty-free Nitsche method. A closely related formulation was derived in the context of Signorini-Poisson problems in [14]. In this setting, the method is stable irrespectively of γ . The term “penalty-free” indicates that condition $T = T_m$ is enforced with the penalty-free Nitsche method (i.e. non-symmetric Nitsche method without penalty term). However, γ is the scaling of a penalty term that enforces the Neumann interface condition when $T < T_m$.

In this article, we focus on the (semi) penalty-free Nitsche method. Our motivation is that, at least in the context of equality constraints, the penalty-free Nitsche method yields better interface fluxes than the symmetric and non-symmetric Nitsche method, as was shown in [5]. Interface fluxes are of particular importance in the Stefan-Signorini problem because they drive the motion of the interface when $T = T_m$.

4 Stabilised Cut Finite Element Formulation

In this section, we introduce the spatial and temporal finite element discretisation of problem (3.17).

4.1 Discretisation in Space

4.1.1 Background Mesh and Fictitious Domain

First, let us introduce important background mesh quantities and the definition of the evolving fictitious domain. Let $\Omega(t_0)$ be our domain in \mathbb{R}^d ($d = 2, 3$) at time $t = t_0$ with Lipschitz boundary $\partial\Omega$ and let $\tilde{\mathcal{T}}_h$ be a quasi-uniform tessellation that covers the domain $\Omega(t_0)$. We define a background domain

$$\Omega_b = \bigcup_{K \in \tilde{\mathcal{T}}_h} K \quad (4.1)$$

associated with our fixed tessellation $\tilde{\mathcal{T}}_h$. The background mesh $\tilde{\mathcal{T}}_h$ will stay fixed in time while the interface $\Gamma(t)$ will move through this background mesh. For $t \in [t_0, t_f]$, we denote the elements in the background mesh $\tilde{\mathcal{T}}_h$ that have at least a small part in domain $\Omega(t)$ as

$$\mathcal{T}_h(t) = \{K \in \tilde{\mathcal{T}}_h : K \cap \Omega(t) \neq \emptyset\}, \quad (4.2)$$

which we call *active mesh* (see the blue and green shaded elements in Figure 4.1). In contrast to the background mesh, the active mesh changes in time. We denote the union of all elements in $\mathcal{T}_h(t)$ as

$$\Omega^*(t) = \bigcup_{K \in \mathcal{T}_h(t)} K. \quad (4.3)$$

$\Omega^*(t)$ is called the *fictitious domain*.

In addition to these time-evolving domains and meshes, we have edge stabilisation quantities that change in time. For each active mesh at time $t \in [t_0, t_f]$, we will distinguish

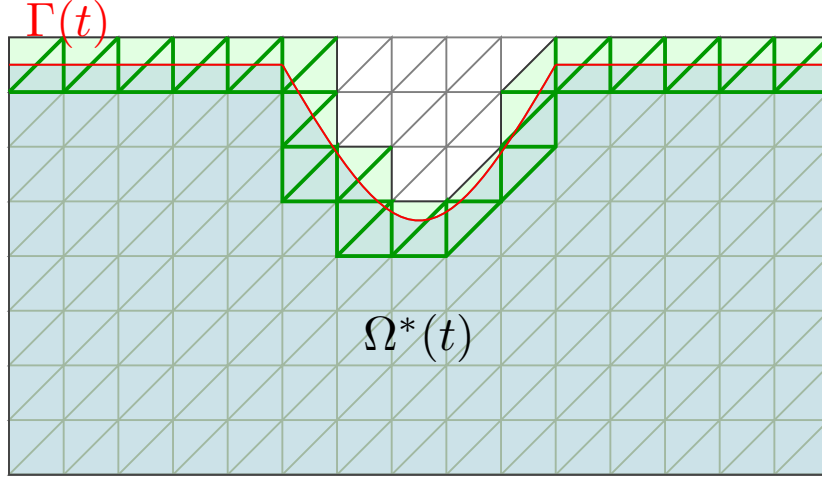


Figure 4.1: Schematics of the domain $\Omega(t)$ covered by a fixed and regular background mesh $\tilde{\mathcal{T}}_h$ and the fictitious domain $\Omega^*(t)$ consisting of all elements in $\tilde{\mathcal{T}}_h$ with at least one part in $\Omega(t)$.

between the following different sets of faces, i.e. edges in 2D and faces in 3D. The *exterior faces*, $\mathcal{F}_e(t)$, which are the faces that belong to one element only in the background mesh and that have an intersection with the active mesh. The *interior faces*, $\mathcal{F}_i(t)$, which are faces that are shared by two elements with $K \cap \Omega(t) \neq \emptyset$. To prevent ill-conditioning, we will apply stabilisation terms to the elements which are intersected by the boundary $\Gamma(t)$, i.e.

$$G_h(t) = \{K \in \mathcal{T}_h(t) : K \cap \Gamma(t) \neq \emptyset\}. \quad (4.4)$$

These stabilisation terms will be applied to so-called *ghost penalty* faces defined as

$$\mathcal{F}_\Gamma(t) = \{F \in \mathcal{F}_i(t) : K_F^+ \cap \Gamma(t) \neq \emptyset \vee K_F^- \cap \Gamma(t) \neq \emptyset\}. \quad (4.5)$$

Here, K_F^+ and K_F^- are the two elements sharing the interior face $F \in \mathcal{F}_i(t)$. The set of faces $\mathcal{F}_\Gamma(t)$ is illustrated by the dark green edges shown in Figure 4.1. To ensure that the boundary $\Gamma(t)$ is reasonably resolved by \mathcal{T}_h , we make the following assumptions:

- G1: The intersection between $\Gamma(t)$ and a face $F \in \mathcal{F}_i(t)$ is simply connected; that is, $\Gamma(t)$ does not cross an interior face multiple times.
- G2: For each element K intersected by $\Gamma(t)$, there exists a plane S_K and a piecewise smooth parametrization $\Phi : S_K \cap K \rightarrow \Gamma(t) \cap K$.
- G3: We assume that there is an integer $N > 0$ such that for each element $K \in G_h(t)$ there exists an element $K' \in \mathcal{T}_h(t) \setminus G_h(t)$ and at most N elements $\{K\}_{i=1}^N$ such that $K_1 = K$, $K_N = K'$ and $K_i \cap K_{i+1} \in \mathcal{F}_i(t)$, $i = 1, \dots, N-1$. In other words, the

number of faces to be crossed in order to “walk” from a cut element K to a non-cut element $K' \subset \Omega(t)$ is uniformly bounded.

Similar assumptions were made in [36, 10].

4.1.2 Nonconforming spatial discretisation of the Stefan-Signorini problem

Using the sets of mesh elements and faces defined above, we can formulate the discrete Stefan-Signorini problem. Firstly, we introduce the continuous linear finite element space on the *active* mesh

$$\mathcal{V}_h(t) = \{v_h \in C^0(\Omega^*(t)) : v_h|_K \in \mathcal{P}_1(K) \forall K \in \mathcal{T}_h(t)\} \quad (4.6)$$

for the temperature.

Secondly, we define a stabilisation operator on the faces $\mathcal{F}_\Gamma(t)$ for the temperature T to prevent ill-conditioning in the case of intersections of $\Gamma(t)$ near a node or face of elements as

$$s_T(T_h, \delta T_h) = \sum_{F \in \mathcal{F}_\Gamma(t)} \gamma_T k h (\llbracket \nabla T_h \rrbracket_n, \llbracket \nabla \delta T_h \rrbracket_n)_F. \quad (4.7)$$

Here, $\llbracket \nabla x \rrbracket_n$ denotes the normal jump of the quantity x over the face, F , defined as $\llbracket \nabla x \rrbracket_n = \nabla x|_{T_F^+} \cdot n_F - \nabla x|_{T_F^-} \cdot n_F$, where n_F denotes a unit normal to the face F with fixed but arbitrary orientation and γ_T is a positive penalty parameter to be determined later. We refer to the term $s_T(T_h, \delta T_h)$ as *ghost penalty* stabilisation [9]. Using the definitions above, we are now in the position to formulate our stabilised cut finite element method for the one-phase Stefan-Signorini problem. The proposed discretisation scheme reads: For all $t \in [t_0, t_f]$, find $T_h \in \mathcal{V}_h(t)$ such that for all $\delta T_h \in \mathcal{V}_h(t)$

$$A(T_h, \delta T_h) + \mathcal{N}(T_h, \delta T_h) = L(\delta T_h), \quad (4.8)$$

where

$$\begin{aligned} A(T_h, \delta T_h) &= a(T_h, \delta T_h) + a_{bc}(T_h, \delta T_h) + s_T(T_h, \delta T_h) + (k \nabla T_h \cdot n_\Gamma, P_{\theta\gamma}^\delta(\delta T_h) - \delta T_h)_{\Gamma_h(t)}, \\ L(\delta T_h) &= l(\delta T_h) + l_{bc}(T_h, \delta T_h) + (I \cdot n_\Gamma, P_{\theta\gamma}^\delta(\delta T_h) - \delta T_h)_{\Gamma_h(t)}, \\ \mathcal{N}(T_h, \delta T_h) &= \frac{1}{\gamma} ([P_\gamma(T_h)]_+, P_{\theta\gamma}^\delta(\delta T_h))_{\Gamma_h(t)}, \end{aligned} \quad (4.9)$$

with

$$a(T_h, \delta T_h) = \rho c \left(\frac{\partial T_h}{\partial t}, \delta T_h \right)_{\Omega_h(t)} + (k \nabla T_h, \nabla \delta T_h)_{\Omega_h(t)}, \quad (4.10)$$

$$l(\delta T_h) = (f, \delta T_h)_{\Omega_h(t)} + (q_N, \delta T_h)_{\partial\Omega_N(t)} + (I \cdot n_\Gamma, \delta T_h)_{\Gamma_h(t)}, \quad (4.11)$$

$$a_{bc}(T_h, \delta T_h) = - (k \nabla T_h \cdot n_\Omega, \delta T_h)_{\partial\Omega_D(t)} - (k \nabla \delta T_h \cdot n_\Omega, T_h)_{\partial\Omega_D(t)} + \frac{k\gamma_b}{h} (T_h, \delta T_h)_{\partial\Omega_D(t)}, \quad (4.12)$$

$$l_{bc}(T_h, \delta T_h) = - (k \nabla \delta T_h \cdot n_\Omega, T_D)_{\partial\Omega_D(t)} + \frac{k\gamma_b}{h} (T_D, \delta T_h)_{\partial\Omega_D(t)}. \quad (4.13)$$

Here, the positive penalty constant γ_b arises from the weak enforcement of Dirichlet boundary conditions through Nitsche's method and $h = \max_{K \in \mathcal{T}_h} h_K$ is the maximum mesh size, where h_K denotes the diameter of K . The penalty parameter γ is now scaling as $\gamma = \hat{\gamma}h$ with $\hat{\gamma} > 0$ chosen to be sufficiently small.

The discretised Stefan-Signorini problem is completed with initial conditions

$$T_h(x, t_0) = \hat{\mathcal{I}}(T_0) \quad \text{in } \Omega_h(t_0), \quad (4.14)$$

and the condition on the normal velocity of the boundary $\Gamma(t)$

$$v \cdot n_\Gamma = \frac{k \nabla T_h \cdot n_\Gamma - I \cdot n_\Gamma}{\rho L}, \quad (4.15)$$

whose discretisation will be discussed in detail in Section 5. In equation (4.14), $\hat{\mathcal{I}}$ is the standard finite element nodal interpolation operator. Note that it is implicitly assumed that field T_0 is analytically available in the entire fictitious domain.

4.2 Discretisation in Time

We decompose the time interval $[t_0, t_f]$ into n_t time steps, and we seek a sequence of solutions $\{T(t_n)\}_{n \in \llbracket 0, n_t-1 \rrbracket} =: \{T^n\}_{n \in \llbracket 0, n_t-1 \rrbracket}$. The reference time t_0 is chosen to be equal to 0. We assume that times $\{t_n\}_{n \in \{0, \dots, n_t-1\}}$ are uniformly spaced, which allows us to define the time step $\Delta t = t_1 - t_0 (= t_2 - t_1 = \dots)$. We apply a backward Euler scheme to the system (4.8) and evaluate integrals over the domain $\Omega_h(t_n)$ and the boundary $\Gamma_h(t_n)$. This time discretisation yields the fully discrete system of equations at time step $n + 1$: Find $T^{n+1} \in \mathcal{V}_h(t_n)$, such that for all $\delta T \in \mathcal{V}_h(t_n)$

$$A_\#(T_h^{n+1}, \delta T) + \mathcal{N}(T_h^{n+1}, \delta T) = L_\#(\delta T), \quad (4.16)$$

where

$$\begin{aligned} A_\#(T_h^{n+1}, \delta T_h) &= a_\#(T_h^{n+1}, \delta T_h) + a_{bc}(T_h^{n+1}, \delta T_h) \\ &\quad + s_T(T_h^{n+1}, \delta T_h) + (k \nabla T_h^{n+1} \cdot n_\Gamma, P_{\theta\gamma}^\delta(\delta T_h) - \delta T_h)_{\Gamma_h(t_n)}, \\ L_\#(\delta T_h) &= L(\delta T) + \rho c \left(\frac{T_h^n}{\Delta t}, \delta T_h \right)_{\Omega_h(t_n)} \end{aligned} \quad (4.17)$$

with

$$a_\#(T_h^{n+1}, \delta T_h) = \rho c \left(\frac{T_h^{n+1}}{\Delta t}, \delta T_h \right)_{\Omega_h(t_n)} + (k \nabla T_h, \nabla \delta T_h)_{\Omega_h(t_n)}. \quad (4.18)$$

4.3 Newton-Raphson algorithm

We solve the previous system (4.16) using a semi-smooth Newton-Raphson algorithm. We linearise the semi-linear form \mathcal{N} around a finite element reference temperature $T_h^* \in \mathcal{V}_h(t_n)$. This is done by writing the Taylor expansion, for any $\delta T \in \mathcal{V}_h(t_n)$,

$$\mathcal{N}(T_h, \delta T) = \mathcal{N}(T_h^*, \delta T) + D\mathcal{N}(T_h - T_h^*, \delta T; T_h^*), \quad (4.19)$$

where $T_h \in \mathcal{V}_h(t_n)$ and \mathcal{DN} is the Gâteaux-derivative of \mathcal{N} , which is defined, for any finite element field $dT_h \in \mathcal{V}_h(t_n)$, by

$$\mathcal{DN}(dT_h, \delta T; T_h^*) = \lim_{z \rightarrow 0} \frac{1}{z} (\mathcal{N}(T_h^* + z dT_h, \delta T) - \mathcal{N}(T_h^*, \delta T)) . \quad (4.20)$$

Identifying the temperature increment $dT_h = T_h - T_h^* \in \mathcal{V}_h(t_n)$, we find that

$$\mathcal{DN}(dT_h, \delta T; T_h^*) = \frac{1}{\gamma} (\mathcal{DG}(dT_h; T_h^*), P_{\theta\gamma}^\delta(\delta T))_{\Gamma(t_n)} , \quad (4.21)$$

where $\mathcal{DG}(\cdot; T_h^*)$ is the Gâteaux-derivative of $G(T) := [P_\gamma(T)]_+$ at T_h^* , which is given by

$$\mathcal{DG}(dT_h; T_h^*) = H(P_\gamma(T_h^*)) (dT_h - \gamma k \nabla dT_h \cdot n_\Gamma) , \quad (4.22)$$

where H is the Heaviside function

$$H(P_\gamma(T_h^*)) = \begin{cases} 1 & \text{if } P_\gamma(T_h^*) > 0, \\ 0 & \text{otherwise.} \end{cases} \quad (4.23)$$

Using these derivations, the Newton predictor for the $(k+1)$ th iterate T_h^{k+1} of T_h^{n+1} is, for any $k \in \mathbb{N}^+$ and for all $\delta T \in \mathcal{V}_h(t_n)$, given by

$$A_\sharp(dT_h, \delta T) + \mathcal{DN}(dT_h, \delta T; T_h^k) = r(\delta T; T_h^k) , \quad (4.24)$$

where the Newton increment is defined as $dT_h := T_h^{k+1} - T_h^k$ and the residual r of iterate T_h^k is such that for any $\delta T \in \mathcal{V}_h(t_n)$

$$r(\delta T; T_h^k) = L_\sharp(\delta T) - (A_\sharp(T_h^k, \delta T) + \mathcal{N}(T_h^k, \delta T)) . \quad (4.25)$$

5 Description of the Domain Movement

In this Section, we describe how domain $\Omega(t)$ is discretised and evolved in time. For each time-step, $t \in [t_0, t_f]$, we first solve (4.16) to obtain the temperature T_h^{n+1} , with which we determine the normal velocity on $\Gamma(t_n)$, i.e.

$$v^{n+1} \cdot n_\Gamma = \frac{k \nabla T^{n+1} \cdot n_\Gamma - I \cdot n_\Gamma}{\rho L} \quad \text{on } \Gamma(t_n) . \quad (5.1)$$

Then, this normal velocity on $\Gamma(t_n)$ is used to move a level-set function as detailed below.

5.1 Level-set description of the moving domain and level-set advection

We track the motion of the boundary $\Gamma(t)$ using a continuous level-set function $\phi : \Omega_b \times [t_0, t_f] \rightarrow \mathbb{R}$, whose zero level set describes the location of the boundary $\Gamma(t) = \{x \in \Omega_b :$

$\phi(x, t) = 0, t \in [t_0, t_f]$. The material domain $\Omega(t)$ is implicitly defined by $\phi(x, t) < 0$, and the fictitious domain by $\phi(x, t) > 0$.

To satisfy equation (5.1), the zero level-set contour is required to move with $v^{n+1} \cdot n_\Gamma$. Furthermore, as the level-set function is defined over the entire background domain Ω_b , the velocity at the interface (5.1) needs to be extended to the remainder of the domain (or at least to a band around the zero isoline of the level-set) to evolve the level-set function. We denote this extension of the velocity with v_{ext} .

Then, the level-set function is moved by solving the advection problem

$$\frac{\partial \phi}{\partial t} + v_{ext} \cdot \nabla \phi = 0 \quad (5.2)$$

with initial condition $\phi(x, 0) = \phi_0$. Here, ϕ_0 is a given initial level-set description of the domain $\Omega(t_0)$. We discretise the level set function using a continuous quadratic finite element space defined on the entire background mesh $\tilde{\mathcal{T}}_h$ denoted by

$$\mathcal{W}_h := \left\{ v_h \in C^0(\Omega_b) : v_h|_K \in \mathcal{P}_2(K) \forall K \in \tilde{\mathcal{T}}_h \right\}. \quad (5.3)$$

To solve the advection equation (5.2), we use a θ -scheme in time and streamline diffusion (SUPG) in space. The discretised advection problem reads: Find $\phi_h^{n+1} \in \mathcal{W}_h$, such that for all $\delta \phi \in \mathcal{W}_h$

$$a_\phi(\phi_h^{n+1}, \delta \phi) = l_\phi(\delta \phi) \quad (5.4)$$

with

$$\begin{aligned} a_\phi(\phi_h^{n+1}, \delta \phi) &= \left(\frac{\phi_h^{n+1}}{\Delta t} + \theta v_{ext}^{n+1} \cdot \nabla \phi_h^{n+1}, \delta \phi + \tau_{SD}(v_{ext}^{n+1} \cdot \nabla \delta \phi) \right)_{\Omega_b}, \\ l_\phi(\delta \phi) &= \left(\frac{\phi_h^n}{\Delta t} + (1 - \theta)v_{ext}^n \cdot \nabla \phi_h^n, \delta \phi + \tau_{SD}(v_{ext}^{n+1} \cdot \nabla \delta \phi) \right)_{\Omega_b} \end{aligned} \quad (5.5)$$

with the streamline diffusion parameter (see [38])

$$\tau_{SD} = 2 \left(\frac{1}{\Delta t^2} + \frac{v_{ext} \cdot v_{ext}}{h^2} \right)^{-\frac{1}{2}} \quad (5.6)$$

and initial condition $\phi_h^0 = \phi_0$. Throughout this contribution, θ is set to 0.5.

5.2 Description of the geometry

The quadratic level-set function is used to define the geometry of our problem including the discretised material domain $\Omega_h(t)$, the discretised interface $\Gamma_h(t)$ and the normal $n_\Gamma(t)$. In the rest of this section we choose a fixed time t and suppress the time dependence to ease the notation.

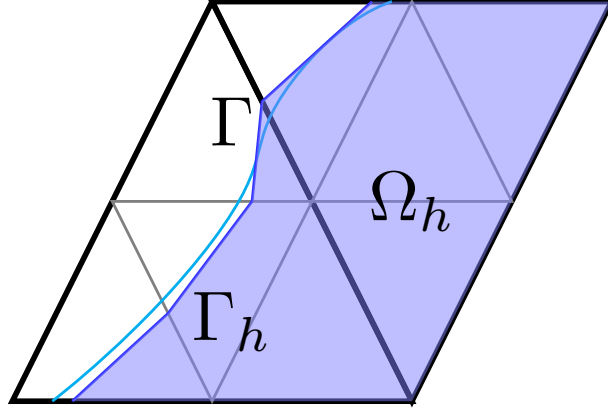


Figure 5.1: Illustration of linear approximation of interface Γ and domain Ω on the refined mesh $\tilde{\mathcal{T}}_{h/2}$ with respect to the mesh $\tilde{\mathcal{T}}_h$.

Normal Computation The normal pointing from the domain $\Omega(t)$ into the fictitious material at the interface $\Gamma(t)$ can be obtained from the level-set function using

$$n_\Gamma(x, t) = \frac{\nabla\phi(x, t)}{\|\nabla\phi(x, t)\|}. \quad (5.7)$$

In this contribution, we determine the normal n_Γ from the level set function through a L^2 -projection onto the continuous piecewise linear space

$$\mathcal{X}_h^d := \left\{ v_h \in [C^0(\Omega_b)]^d : v_h|_K \in \mathcal{P}_1(K) \forall K \in \tilde{\mathcal{T}}_h \right\}. \quad (5.8)$$

Here, $d = 2, 3$ is the geometrical dimension. We determine the normal by finding $n_\Gamma \in \mathcal{X}_h^d$ such that for all $\delta n_\Gamma \in \mathcal{X}_h^d$

$$(n_\Gamma, \delta n_\Gamma)_{\Omega_b} = \left(\frac{\nabla\phi}{|\nabla\phi|}, \delta n_\Gamma \right)_{\Omega_b}. \quad (5.9)$$

Discrete Geometrical Domains To define our discrete domains Ω_h and Γ_h , we use a two-grid solution proposed by [34, 33] which is outlined in the following. First, we interpolate the piecewise quadratic level-set function onto a piecewise linear function, $\mathcal{I}(\phi_h)$, on a regularly refined mesh $\tilde{\mathcal{T}}_{h/2}$, such that

$$\mathcal{I}(\phi_h(\mathbf{v})) = \phi_h(\mathbf{v}) \text{ for all nodes } \mathbf{v} \text{ in } \tilde{\mathcal{T}}_{h/2}. \quad (5.10)$$

We then use this piecewise linear interpolation to determine the intersection between $\mathcal{I}(\phi_h)$ with the refined grid to obtain the piecewise linear approximation of Ω_h and Γ_h as illustrated in Figure 5.1.

5.3 Interface velocity smoothing and bulk extension

To enable a smooth domain movement, we construct a continuous piecewise linear normal velocity approximation in the vicinity of the interface in the following way. We first recover a smoothed gradient of the temperature using the stabilised projection: Find $G_T^h \in \mathcal{V}_h^d(t)$ such that for all $\delta G_T \in \mathcal{V}_h^d(t)$

$$(G_T^h, \delta G_T)_{\Omega_h} + s_{G_T}(G_T^h, \delta G_T) = (\nabla T_h^{n+1}, \delta G_T)_{\Omega_h}, \quad (5.11)$$

$$s_{G_T}(G_T^h, \delta G_T) = \sum_{F \in \mathcal{F}_\Gamma(t)} \gamma_{G_T} h (\llbracket \nabla G_T^h \rrbracket_n, \llbracket \nabla \delta G_T \rrbracket_n)_F, \quad (5.12)$$

where

$$\mathcal{V}_h^d(t) = \{v_h \in [C^0(\Omega^*(t))]^d : v_h|_K \in \mathcal{P}_1(K) \forall K \in \mathcal{T}_h(t)\} \quad (5.13)$$

is the vector-valued space of continuous piecewise linear functions on the fictitious domain with $d = 2, 3$. Here, $\gamma_{G_T} > 0$ is a positive penalty parameter to recover a continuous piecewise linear temperature gradient over patches of elements in the interface region from the piecewise constant temperature gradient ∇T_h^{n+1} . The normal velocity is then determined from the following L^2 projection: Find $v_n^h := v_h^{n+1} \cdot n_\Gamma \in \mathcal{V}_h(t)$ such that for all $\delta v_n \in \mathcal{V}_h(t)$

$$(v_n^h, \delta v_n)_{\Omega^*} = H(P_\gamma(T_h^{n+1})) \left[\left(\frac{(kG_T^h - I) \cdot n_\Gamma}{\rho L}, \delta v_n \right)_{\Omega^*} - \frac{\theta_1}{\gamma \rho L} (T_h^{n+1} - T_m, \delta v_n)_{\Gamma_h} \right]. \quad (5.14)$$

We extend this normal velocity field onto the entire background domain Ω_b using a fast marching scheme as detailed in [46, 43, 33]. The principle of this technique relies on two steps: a near-field step and a far-field step. In the near-field step, all nodes of elements which are intersected by the interface obtain the normal velocity value determined by a closest point projection onto the discretised interface Γ_h , i.e. $v_n^{ext}(\mathbf{v}) = v_n(P_W(\mathbf{v}))$, where \mathbf{v} is an element node and P_W is the closest point projection onto $p_\Gamma(\mathbf{v}) \in \Gamma_h$ given by the shortest distance between \mathbf{v} and Γ_h . In the far-field step the information of the intersected elements is propagated through evaluation of known extended velocity function values. For a detailed description see [46, 43, 33].

This extended normal velocity field, v_n^{ext} , is then used to obtain the vectorial extended velocity field as

$$v_{ext} = v_n^{ext} \cdot n_\Gamma, \quad (5.15)$$

which is then used to propagate the level-set function using equation (5.5).

Remark 5.1 *This extended velocity field enables the transport of the level-set function in a way that prevents a large deviation of the level set away from a signed distance function. Therefore reinitialisation of the level set is rarely required. However, in examples with strong interface deformation reinitialisation may become necessary. In these rare cases, we use a fast marching redistancing technique described in [33, 34], which relies on a fast marching scheme of the linearly interpolated level set function on the regularly refined grid $\tilde{\mathcal{T}}_{h/2}$.*

Algorithm 1 summaries the CutFEM Stefan-Signorini algorithm presented in the previous sections.

Algorithm 1 CutFEM Stefan-Signorini Algorithm

```

1: Set  $t = t_0$ ,  $T_h^0 = T_0$ ,  $\phi_h^0 = \phi_0$ .
2: while  $t \leq t_f$  do  $\triangleright t_f$  is the final time
3:   Determine  $\Omega_h$  and  $\Gamma_h$  through intersection computations of zero level-set with back-
   ground mesh.
4:   Compute normal  $n_\Gamma$  using (5.9).
5:   procedure STEFAN-SIGNORINI-NITSCHKE( $T_h^n$ )
6:     Solve (4.16) using Newton-Raphson algorithm.
7:     return  $T_h^{n+1}$ .
8:   end procedure
9:   procedure VELOCITY( $T_h^{n+1}$ )
10:    Compute smoothed temperature gradient  $G_T$  using (5.12).
11:    Determine normal velocity on  $\Gamma_h$  using (5.14).
12:    Extend normal velocity to obtain  $v_{ext}$ .
13:    return  $v_{ext}$ .
14:   end procedure
15:   procedure LEVEL SET ADVECTION( $v_{ext}$ )
16:    Solve advection problem (5.4) for level set.
17:    return  $\phi_h^{n+1}$ .
18:   end procedure
19:    $t = t + \Delta t$ ,  $\phi_h^n = \phi_h^{n+1}$ ,  $T_h^n = T_h^{n+1}$ .
20: end while

```

6 Numerical results

In this Section, we present numerical results for a manufactured solution, and for several thermal ablation problems in 2D and 3D. The penalty parameters are set to $\gamma_{GT} = 10^{-3}$, $\gamma_T = 10^{-1}$, $\hat{\gamma} = 1$, $\gamma_b = 100$ and we choose the penalty-free, non-symmetric Nitsche method, i.e. $\theta_1 = 0$, $\theta_2 = -1$ in all the presented results.

6.1 Manufactured solution: convergence analysis

We have constructed a two-dimensional manufactured solution inspired by the manufactured solution of a two-phase Stefan problem in [46]. We consider a rectangular domain Ω with a circular hole. The circular hole gets heated by a heat flux $I(x, t)$. We choose $T_m = -0.01$, $\rho = c = k = 1.0$, $L = 1.0$. We consider an analytical temperature distribution

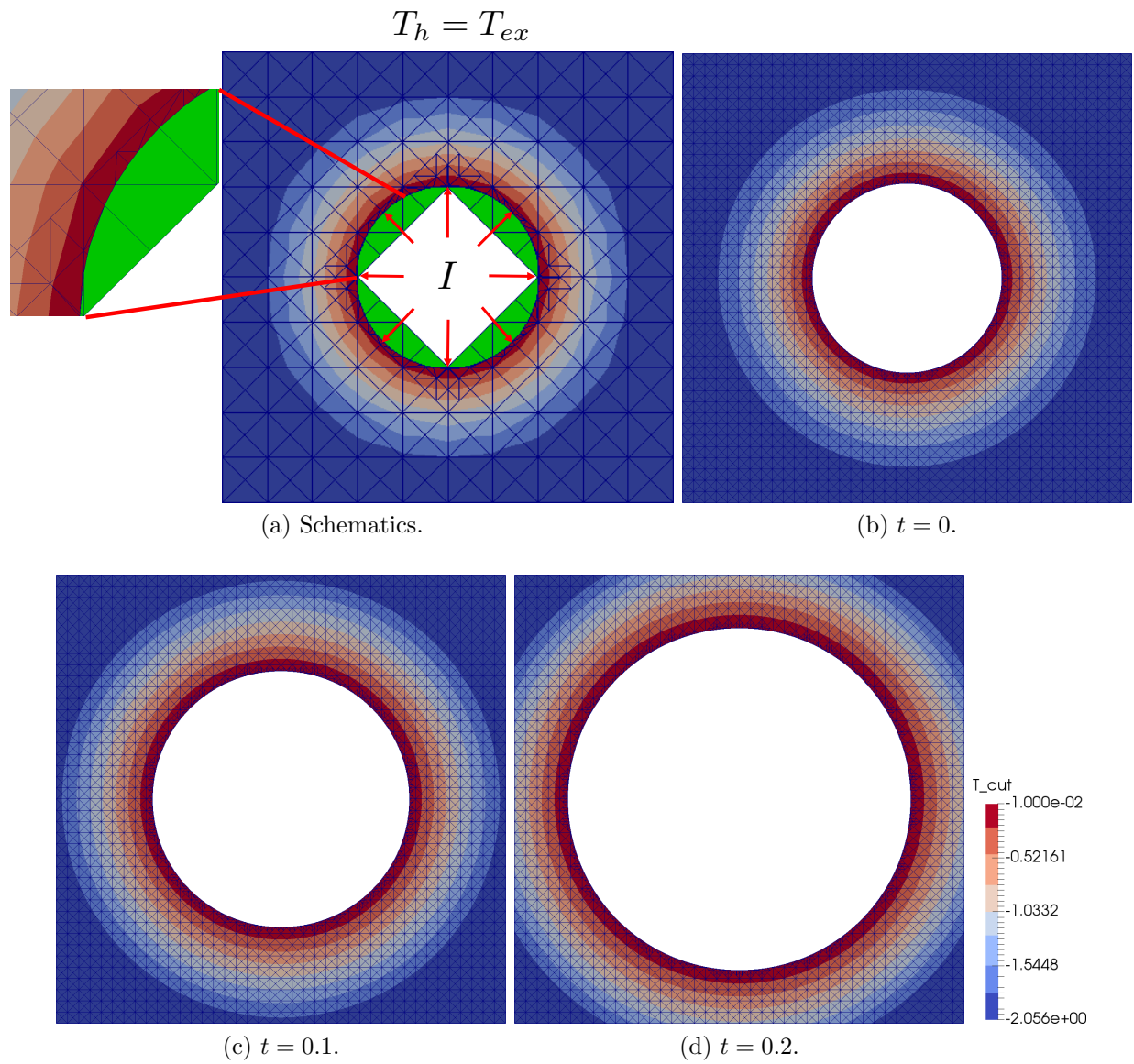


Figure 6.1: Schematics of the manufactured solution and numerical solution at time $t = \{0, 0.1, 0.2\}$ for $h = 1/40$.

given by

$$T_{ex}(x, t) = -e^{r(x)} + \cos\left(\frac{\pi r(x)}{2 \log(\alpha(t))}\right) - T_m + \alpha(t). \quad (6.1)$$

Here, $r(x) = \sqrt{x^2 + y^2}$ and

$$\alpha(t) = \frac{3}{2 - 3t}. \quad (6.2)$$

At $r(x) = R(t)$ with

$$R(t) = \log(\alpha(t)) \quad (6.3)$$

the analytical solution is at melting temperature $T = T_m$. For $t = 0$, we obtain the initial condition with $\alpha(0) = \frac{3}{2}$ as

$$T_0 = -e^{r(x)} + \cos\left(\frac{\pi r(x)}{2 \log(1.5)}\right) + \frac{3}{2} - T_m. \quad (6.4)$$

The volume source term f can now be determined from T_{ex} ¹. A level-set describing the location of the melting temperature is given by

$$\phi(x, t) = R(t) - r(x). \quad (6.5)$$

This level-set describes the motion of the circular hole and the normal velocity of the hole is given by

$$v(x, t) \cdot n_\Gamma = -\frac{\partial R(t)}{\partial t} = -\alpha(t). \quad (6.6)$$

The expression for the beam at $\Gamma(t)$ can now be determined from (2.3) which yields

$$\begin{aligned} I_{ex}(x, t) &= A_{ex}(t) e_{ray,ex}(x), \\ A_{ex}(t) &= -\left[(\rho L + 1)\alpha(t) + \frac{\pi}{2R(t)}\right], \\ e_{ray,ex}(x) &= -n_{\Gamma,ex} = \frac{1}{r(x)} \begin{pmatrix} x \\ y \end{pmatrix}. \end{aligned} \quad (6.7)$$

To test our numerical scheme, we set the temperature $T = T_{ex}$ on $\partial\Omega$ and apply the heat flux expression (6.7) on $\Gamma(t)$. Figure 6.1 shows the numerical solution at time $t = 0, 0.1, 0.2$ and shows the removal of material with time. We test convergence with mesh refinement and with time step refinement. We evaluate the error of our numerical solution with respect to the analytical solution in the following relative error norms. For each time-step,

¹the corresponding symbolic derivation using an IPython notebook can be found in [21].

$t_n \in t_0, \dots, t_{n_t}$, we determine

$$\begin{aligned}
e_{L^2(\Omega_h)}(u, t_n) &:= \frac{\|u_h - u_{ex}\|_{L^2(\Omega_h(t_n))}}{\|u_{ex}\|_{L^2(\Omega_h(t_n))}} = \frac{\sqrt{\int_{\Omega_h(t_n)} (u_h - u_{ex})^2 dx}}{\sqrt{\int_{\Omega_h(t_n)} (u_{ex})^2 dx}}, \\
e_{H^1(\Omega_h)}(u, t_n) &:= \frac{\|u_h - u_{ex}\|_{H^1(\Omega_h(t_n))}}{\|u_{ex}\|_{H^1(\Omega_h(t_n))}} = \frac{\sqrt{\int_{\Omega_h(t_n)} (u_h - u_{ex})^2 + (\nabla(u_h - u_{ex}))^2 dx}}{\sqrt{\int_{\Omega_h(t_n)} (u_{ex})^2 + \nabla u_{ex}^2 dx}}, \\
e_{L^2(\Gamma_h)}(u, t_n) &:= \frac{\|u_h - u_{ex}\|_{L^2(\Gamma_h(t_n))}}{\|u_{ex}\|_{L^2(\Gamma_h(t_n))}} = \frac{\sqrt{\int_{\Gamma_h(t_n)} (u_h - u_{ex})^2 dx}}{\sqrt{\int_{\Gamma_h(t_n)} (u_{ex})^2 dx}}.
\end{aligned} \tag{6.8}$$

Here, u_h is the numerical solution and u_{ex} is the analytical solution, which in the following will be the temperature, the radius or the interface velocity. To average the error over time, we define the l^2 -error norm over the time interval as

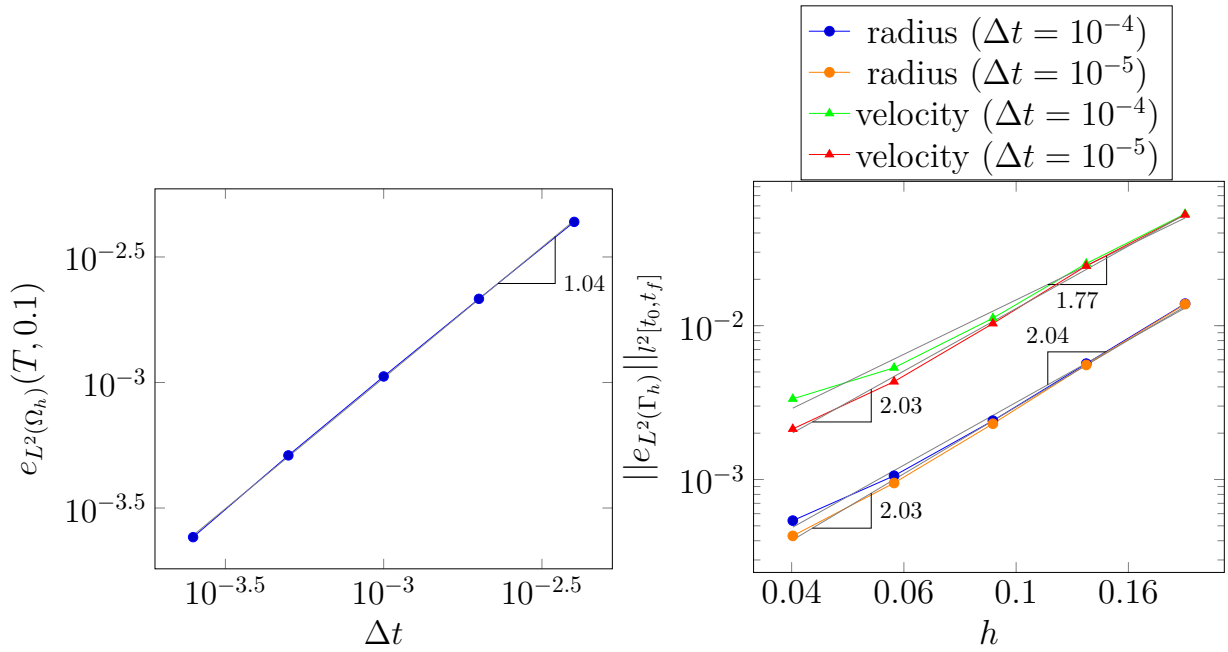
$$\|e(u)\|_{l^2[t_0, t_f]} = \sqrt{\frac{1}{n_t} \sum_{i=0}^{n_t} e(u, t_i)^2}. \tag{6.9}$$

Here, $e(u)$ is any of the error measures defined in equation (6.8).

We compute the numerical solution in the time interval $t \in [0, 0.1]$ for time step sizes $\Delta t = \{10^{-4}, 10^{-5}\}$. Figure 6.2 shows the convergence of the temperature to the analytical solution with mesh refinement and the convergence of the temperature with time step refinement. As can be seen in Figure 6.2, we obtain optimal convergence orders of second order for the L^2 -norm and of first order for the H^1 -norm in space and first order convergence in the L^2 -norm in time for the temperature. For the convergence of velocity and radius of the circular hole, we obtain a convergence rate of second order. The convergence rates for the L^2 -errors in temperature, velocity and radius show a slight improvement for the time step $\Delta t = 10^{-5}$ in comparison to time step $\Delta t = 10^{-4}$ for finer meshes. This is to be expected as the discretisation error in time is starting to dominate the total error for finer meshes and is impacting the convergence rate. Figure 6.3 shows the averaged computed velocity and the averaged computed radius and their analytical expression. The average is computed for all $t_n \in t_0, \dots, t_{n_t}$ as

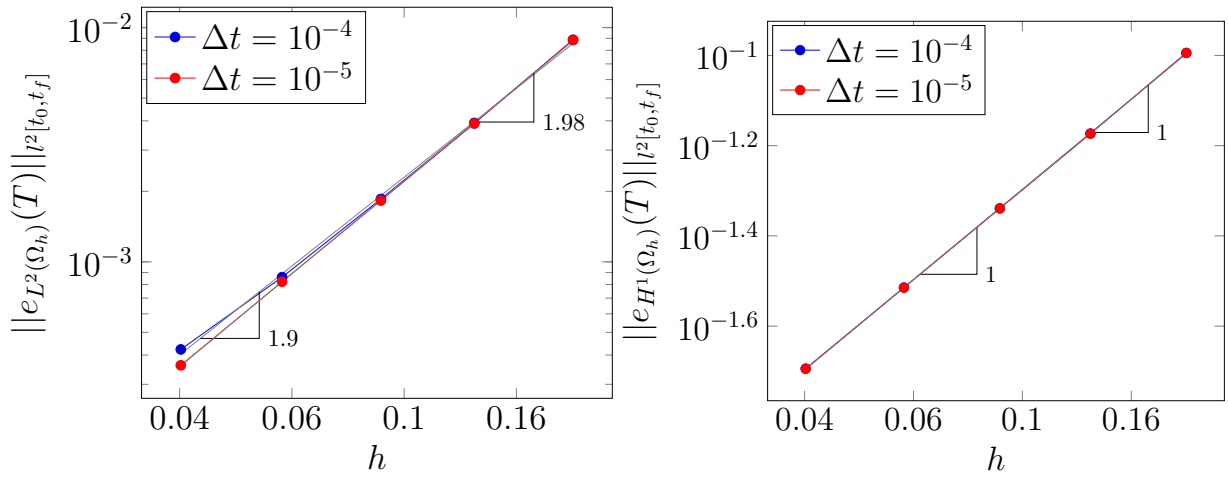
$$\begin{aligned}
v_{avg}(t_n) &= \frac{\int_{\Gamma_h(t_n)} v_n^h ds}{\int_{\Gamma_h(t_n)} ds}, \\
r_{avg}(t_n) &= \frac{\int_{\Gamma_h(t_n)} \sqrt{x^2 + y^2} ds}{\int_{\Gamma_h(t_n)} ds}.
\end{aligned} \tag{6.10}$$

It is clear that both the velocity and radius approach the exact solution as the mesh is refined. The convergence in position seems to be monotonic, at any time of the analysis,



(a) L^2 error in temperature with time step size for $h = 1/160$ at time $t = 0.1$.

(b) L^2 error in velocity and radius.



(c) Temperature L^2 -error.

(d) Temperature H^1 -error.

Figure 6.2: Convergence rates for L^2 and H^1 errors with mesh refinement and with time step refinement.

while the instantaneous convergence in velocity appears to be much more erratic, which is to be expected, given the fact that the velocity is the time derivative of the position.

6.2 Thermal ablation using a moving laser beam

In this Section, we will set-up several numerical examples describing a laser beam heating a workpiece alongside a predefined machining path. We define the intensity of the spatially Gaussian-distributed beam as

$$\begin{aligned} I(x, t) &= -A_p(\theta) f(x, t) e_{ray}, \\ f(x, t) &= f_x(x, t) f_t(t) := A_{amp} \frac{1}{\sqrt{2\pi^{d-1}\sigma^2}} e^{-\frac{p(x,t) \cdot p(x,t)}{2\sigma^2}} f_t(t), \\ p(x, t) &= (x - F(t)) - ((x - F(t)) \cdot e_{ray}) e_{ray}, \end{aligned} \quad (6.11)$$

where σ is the width of the beam, A_{amp} is the amplitude of the beam, $F(t)$ is the focal point of the beam that describes the path of the laser beam. In the following, we choose the direction of the beam, e_{ray} , to be constant in time. The beam is scaled with the absorption coefficient ([54], [59], [55]) given by

$$A_p(\theta) = \begin{cases} 1 - \frac{2 \cos(\theta)^2 - 2\varepsilon \cos(\theta) + \varepsilon^2}{2 \cos(\theta)^2 + 2\varepsilon \cos(\theta) + \varepsilon^2} & \cos(\theta) > 0, \\ 0 & \text{otherwise,} \end{cases} \quad (6.12)$$

where the angle of incidence θ of the laser beam with respect to the inside surface normal $-n_\Gamma$ appears in the equation through trigonometric function $\cos(\theta) = -n_\Gamma(x, t) \cdot e_{ray}$. Here, ε is a material-dependent quantity, which we choose as $\varepsilon = 1$. We choose to represent a pulsed laser beam whose periodic on/off behaviour can be described by using the pulse function

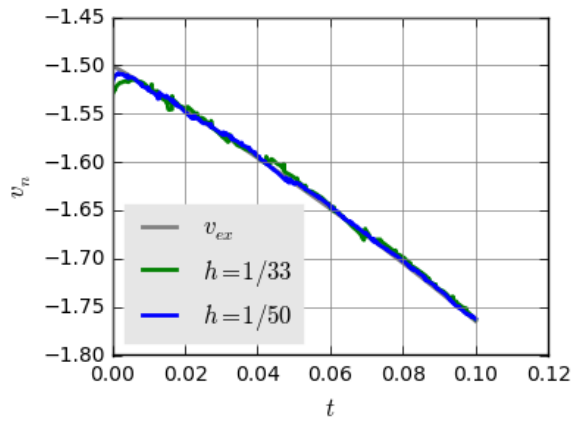
$$f_t(t) = \begin{cases} 1 & \text{if } t - \lfloor \frac{t}{P_0} \rfloor P_0 \leq \frac{P_0}{2}, \\ 0 & \text{else,} \end{cases} \quad (6.13)$$

where $\lfloor \cdot \rfloor$ denotes the floor operation, and P_0 is the total period, which is the sum of an "on" phase of duration t_{ON} and an "off" phase of duration t_{OFF} during which the workpiece does not receive any energy from the thermal ablation device.

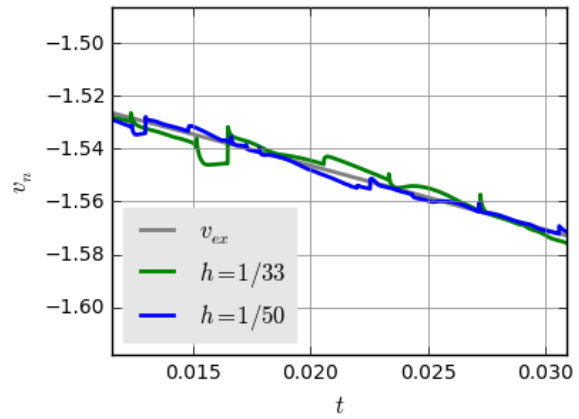
In the following sections, we consider rectangular workpieces for which the top boundary is the moving boundary Γ (i.e. thermally ablated surface). Homogenous Dirichlet boundary conditions, i.e. $T|_{\partial\Omega_D} = 0$, will be applied to the bottom boundary, and homogeneous Neumann boundary conditions will be applied to the remaining sides (see Figure 2.1).

6.2.1 Pulsed thermal ablation in 2D

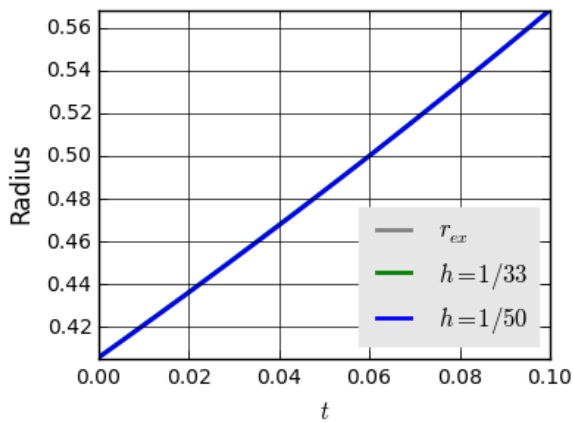
Consider a rectangular background domain $\Omega_b = (0, 3) \times (0, 1.2)$ and a time interval $t \in [0, 1.6]$. We consider an initial level-set of $\phi(x, 0) = y - 1.0$ leaving a rectangular block of material $\Omega_h = (0, 3) \times (0, 1)$. The workpiece Ω_h is heated by a laser beam described by



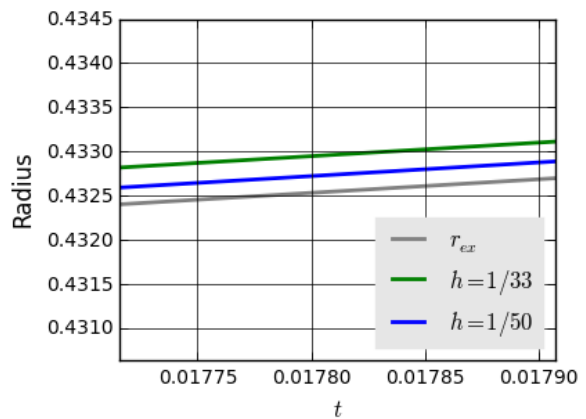
(a) Velocity.



(b) Velocity.



(c) Radius.



(d) Radius.

Figure 6.3: Computed average velocity and average radius versus analytical solution for $\Delta t = 10^{-5}$.

equation (6.11) with width $\sigma = 0.1$, amplitude $A_{amp} = 2$ and beam direction $e_{ray} = (0, -1)$. The time evolution of the laser beam is described by the path of focal point $F(t)$, which is defined for all times $\{t_n\}_{n \in \llbracket 0, n_t-1 \rrbracket}$ by

$$F(t_{n+1}) = F(t_n) + v_F(t)\Delta t \quad (6.14)$$

with initial beam focal point $F_0 = (0.5 \ 1)^T$ and initial velocity $v_F(t) = (5 \ 0)^T$. Velocity v_F conserves its magnitude throughout the simulation, but changes direction every $t_{change} = 0.4$ units of time. Changing the sign of the velocity vector causes the laser beam to pass over the block of material four times. We choose the material parameters as $T_m = 0.1$, $L = 1$, $\rho = 1$, $k = 1$, $c = 1$. We choose two different pulse periods $P_0 \in \{0.1, 0.01\}$ and compare the corresponding results. We choose a fixed time step of $\Delta t = 5 \cdot 10^{-4}$ and a fixed mesh size $h = 0.048$. Figure 6.4 shows the temperature contour at times $t = 0.4, 0.8, 1.2, 1.6$. The short pulsed beam ($P_0 = 0.01$) removes material in an even manner, leaving no visible crater on the surface of the workpiece, while the long pulsed beam ($P_0 = 0.1$) leaves a wavy surface with visible craters. For these two simulations, the amplitude and spatial distribution of the energy is the same, and the ratio between the on and off time are also equal. As a result, the average power received by the workpiece over one period is the same in both cases, which explains why the depths of the resulting cavities are similar (see Figure 6.5). Note however that there is no theoretical reason for a strict equality between the volumes removed during the process as the amount of energy lost through the Dirichlet conditions and the quantity of thermal energy remaining in the workpiece at the end of the simulation may differ (slightly) in the two examples.

6.2.2 Laser beam in 3D

In this Section, we consider two 3D examples. The first example describes the formation of a single crater, for a spatially fixed laser beam, while the second example describes a complex ablation process designed to manufacture a rectangular cavity through the continuous motion of the laser beam.

Single Crater formation We compute the formation of a single crater considering a rectangular background domain $\Omega_b = (-0.5, 0.5) \times (-0.5, 0.5) \times (-0.5, 0.01)$ with an initial level set $\phi(x, 0) = y$ in a time interval $t = [0, 0.2]$. We fix the time step size to $\Delta t = 0.005$ and the mesh size to $h = 0.029$. We choose the material parameters as $T_m = 0.01$, $L = 1$, $\rho = 1$, $k = 1$, $c = 1$. The focal point of the laser beam is fixed in time, $F(t) = (0, 0, 0)$ and we set $e_{ray} = (0, 0, -1)$, $A_{amp} = 3$ and $\sigma = 0.1$. The laser beam is switched on over the entire time period. Figure 6.6 shows the crater profile, together with several temperature isolines, at time $t = 0.2$. The laser beam causes the formation of a single deep crater in a cone shape.

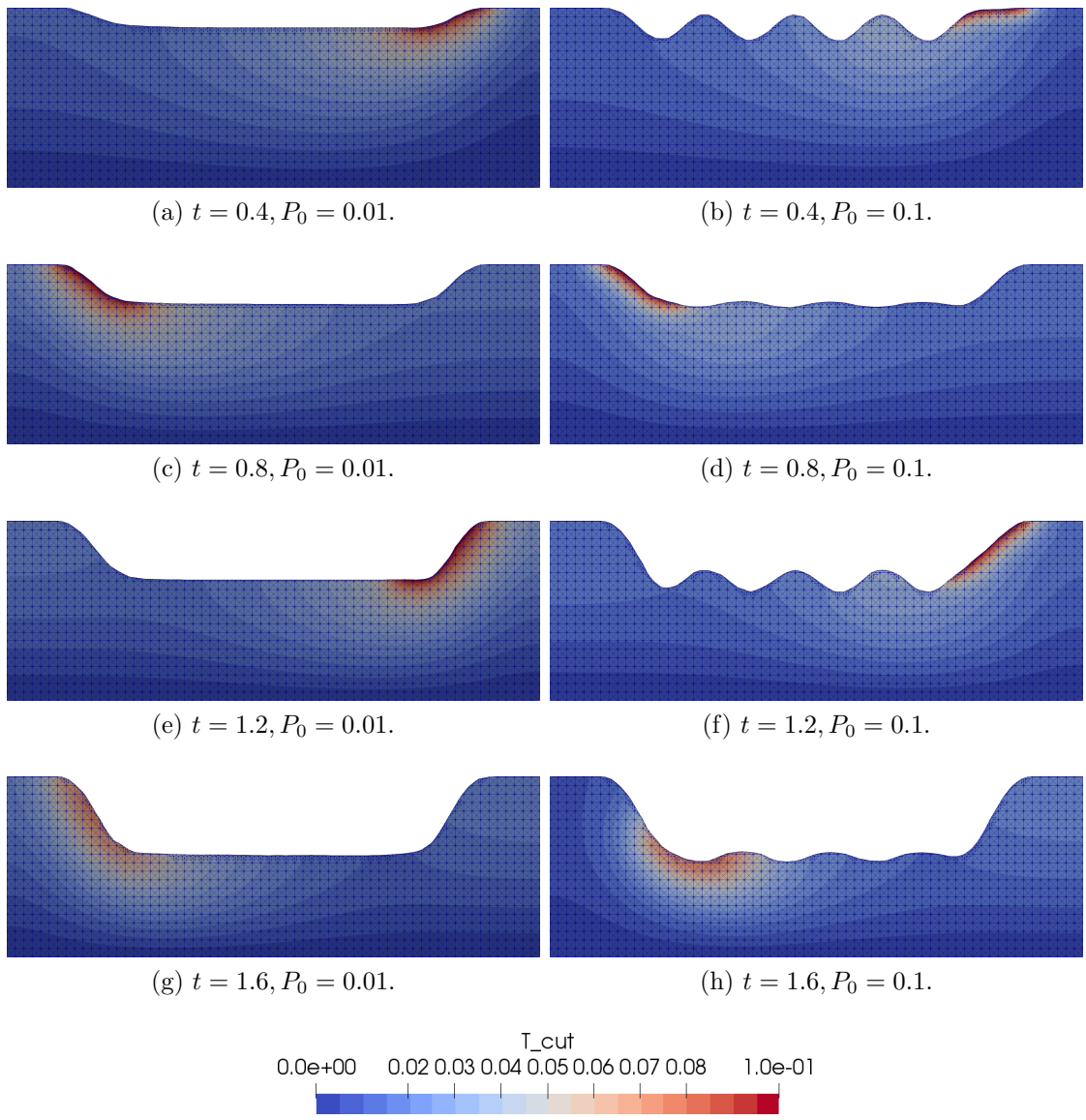


Figure 6.4: Pulsed laser beam for periods $P_0 = 0.01$ on the left and $P_0 = 0.1$ on the right for time $t = 0.4, 0.8, 1.2, 1.6$.

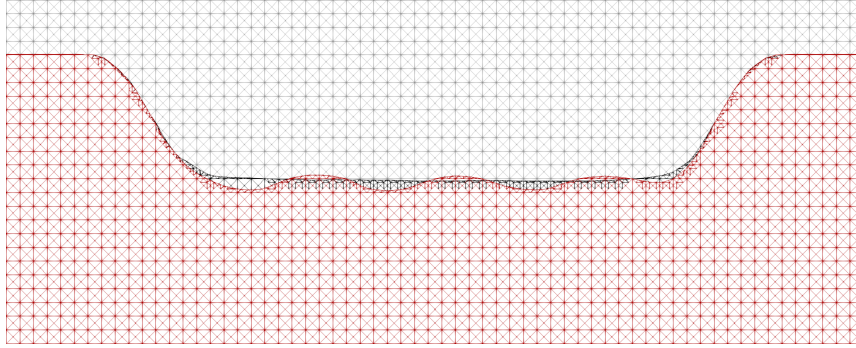


Figure 6.5: Background mesh and cut-meshes at final time $t = 1.6$ for pulsed beam $P_0 = \{0.01, 0.1\}$.

Complex 3D machining path We consider a complex machining path specified as shown in Figure 6.7. We aim to form a rectangular cavity, using a machining strategy that is typical of what could be generated by a CAM software featuring thermal milling capabilities. The background domain is set to $\Omega_b = (-1, 1) \times (-1.5, 1.5) \times (-0.6, 0.01)$ with an initial level set of $\phi(x, 0) = z$. We set the time step to $\Delta t = 0.005$ and the mesh size to $h = 0.06$. The ablation strategy is described through the motion of the focal point $F(t)$ in time interval $t \in [0, 3]$. The top layer is machined first, and deeper layers as represented in Figure 6.7 are applied subsequently. The remaining parameters of the laser beam are set to $e_{ray} = (0, 0, -1)$, $A_{amp} = 3$ and $\sigma = 0.1$. We choose the material parameters as $T_m = 0.01$, $L = 1$, $\rho = 1$, $k = 1$, $c = 1$. For this particular example, function $f_t(t)$ is always equal to one (i.e. the laser fires continuously). As shown in Figure 6.8, the manufacturing process creates the expected rectangular cavity. The cylinder displayed in Figure 6.8 represents the contour line $I = 2$ of the laser beam. As the workpiece receives energy in a continuous way, no crater is formed. However, we can clearly see the streaks left by the laser beam, owing to a rather large hatch distance (i.e. distance between two consecutive straight lines of the machining path, within one particular layer).

The results of this simulation can be played by paraview, using the .vtk files archived on zenodo.org by the authors [20].

7 Conclusions

We have presented the first CutFEM algorithm dedicated to the solution of unsteady, one-phase Stefan-Signorini problems. The geometry of the domain is represented implicitly through the negative values of a continuous, piecewise linear level-set function defined using a regular, fixed finite element mesh. The boundary of the thermally ablated material can move arbitrarily and cut through the bulk of the elements, which circumvents the need

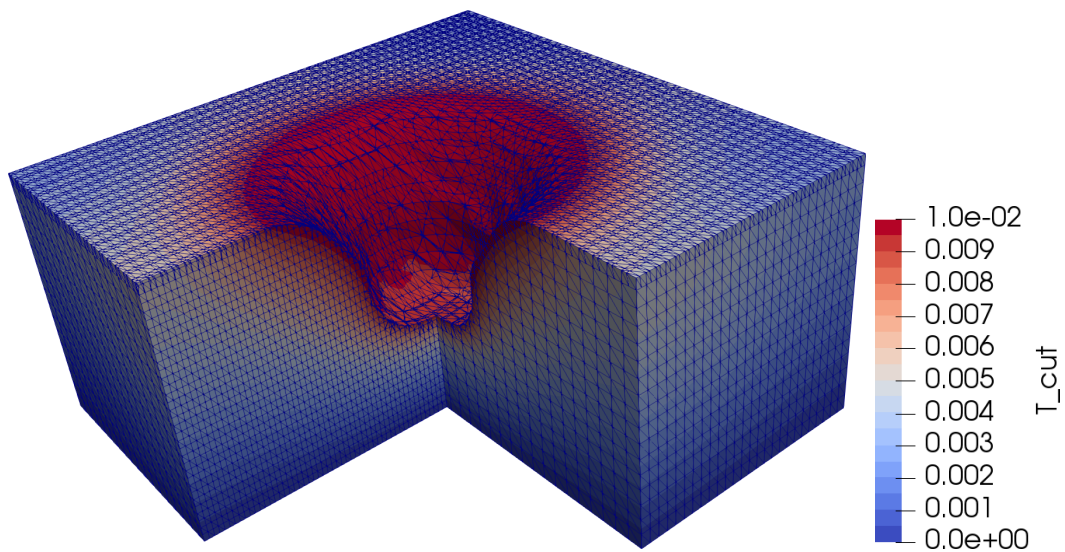


Figure 6.6: One crater formed by a laser beam at a fixed spatial location, and the distribution of temperature computed at $t = 0.2$ units of time.

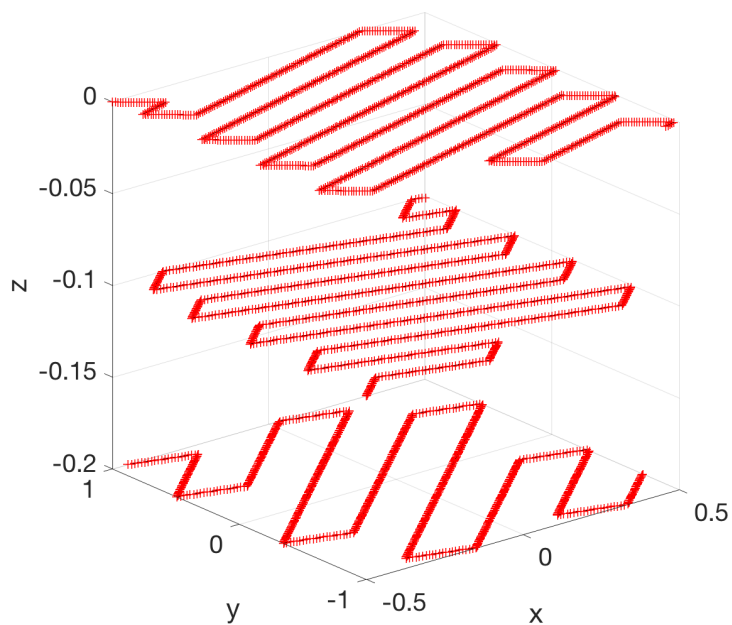


Figure 6.7: Prescribed path of the focal point of a laser beam designed to create a rectangular cavity. The depth is not significant here as the laser beam is invariant in the z direction.

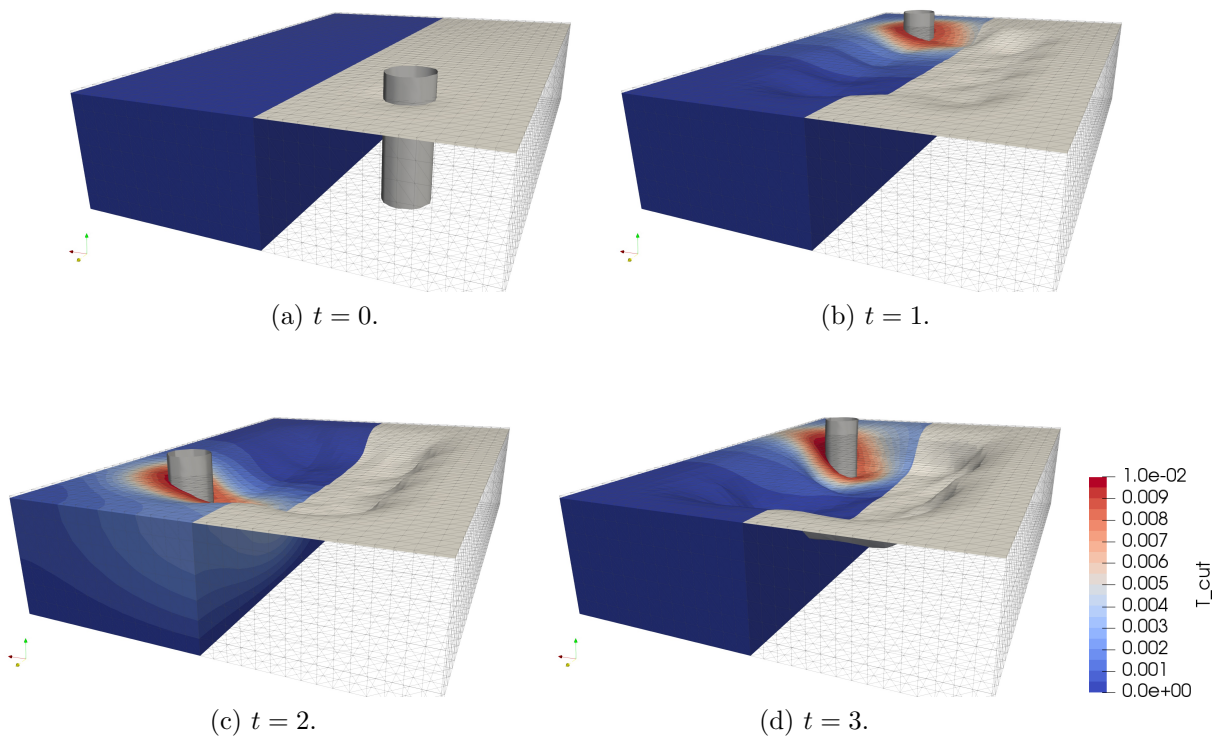


Figure 6.8: Laser beam going through the machining path shown in Figure 6.7 to create the desired rectangular cavity. The results correspond to analysis times $t = [0, 1, 2, 3]$.

for any remeshing operation during the simulation of phase change. We showed that the primal/dual formulation of the one-phase thermal ablation problem could be reformulated as a purely primal, nonlinear problem, using the Nitsche-Signorini idea, which avoids the need to introduce a Lagrange multiplier field for the interface velocity, and circumvents the need to design an inf-sup stable primal/dual discretisation strategy. Through the addition of stabilisation terms associated with the cut region, we proved that the method remains stable independently of the cut location. In addition, by carefully h-weighting several terms of the weak form associated with the proposed Stefan-Signorini-Nitsche method, we obtained optimal convergence with respect to spatial and temporal refinement. As a further contribution, we developed a 2D benchmark to test the convergence of numerical methods for one-phase Stefan problems. We hope that the new manufactured solution will be of use to researchers in the future.

The robustness and versatility of the proposed algorithm was demonstrated through several representative examples in 2D and 3D. Although the method is general, our example section targeted realistic applications in laser micro-manufacturing, including the simulation of laser drilling and laser milling operations.

Acknowledgements

The authors gratefully acknowledge the financial support provided by the Welsh Government and Higher Education Funding Council for Wales through the Sêr Cymru National Research Network in Advanced Engineering and Materials under grants NRNG06 and NRN102.

References

- [1] O. Allix, P. Kerfriden, and P. Gosselet. On the control of the load increments for a proper description of multiple delamination in a domain decomposition framework. *International Journal for Numerical Methods in Engineering*, 83:1518–1540, 2010.
- [2] M. Alnæs, J. Blechta, J. Hake, A. Johansson, B. Kehlet, A. Logg, C. Richardson, J. Ring, M.E. Rognes, and G.N. Wells. The FEniCS project version 1.5. *Archive of Numerical Software*, 3(100):9–23, 2015.
- [3] Y. Belhamadia, A.S. Kane, and A. Fortin. An enhanced mathematical model for phase change problems with natural convection. *Int. J. Numer. Anal. Model*, 3(2):192–206, 2012.
- [4] M.K. Bernauer and R. Herzog. Optimal control of the classical two-phase Stefan problem in level set formulation. *SIAM Journal on Scientific Computing*, 33(1):342–363, 2011.
- [5] T. Boiveau, E. Burman, and S. Claus. Penalty-free nitsche method for interface problems. In *Geometrically Unfitted Finite Element Methods and Applications*, pages 183–210. Springer, 2017.
- [6] S. Bordas and B. Moran. Enriched finite elements and level sets for damage tolerance assessment of complex structures. *Engineering Fracture Mechanics*, 73(9):1176–1201, 2006.
- [7] S.P.A. Bordas, T. Rabczuk, J.J. Rodenas, P. Kerfriden, M. Moumnassi, and S. Belouettar. Recent advances towards reducing the meshing and re-meshing burden in computational sciences. *Computational technology reviews*, 2:51–82, 2010.
- [8] A.N. Brooks and T. J.R. Hughes. Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 32(1-3): 199–259, 1982.
- [9] E. Burman. Ghost penalty. *Comptes Rendus Mathematique*, 348(21-22):1217–1220, 2010.

- [10] E. Burman and P. Hansbo. Fictitious domain finite element methods using cut elements: II. A stabilized Nitsche method. *Appl. Numer. Math.*, 62(4):328–341, 2012.
- [11] E. Burman and P. Hansbo. Deriving robust unfitted finite element methods from augmented Lagrangian formulations. In *Geometrically Unfitted Finite Element Methods and Applications*. Springer, 2018.
- [12] E. Burman, S. Claus, P. Hansbo, M.G. Larson, and A. Massing. CutFEM: discretizing geometry and partial differential equations. *Int. J. Numer. Meth. Eng.*, 104(7):472–501, 2015.
- [13] E. Burman, P. Hansbo, and M.G. Larson. Augmented Lagrangian finite element methods for contact problems. *arXiv preprint arXiv:1609.03326*, 2016.
- [14] E. Burman, P. Hansbo, and M.G. Larson. The penalty-free Nitsche method and nonconforming finite elements for the Signorini problem. *SIAM Journal on Numerical Analysis*, 55(6):2523–2539, 2017.
- [15] J. Chessa, P. Smolinski, and T. Belytschko. The extended finite element method (XFEM) for solidification problems. *International Journal for Numerical Methods in Engineering*, 53(8):1959–1977, 2002.
- [16] F. Chouly and P. Hild. A Nitsche-based method for unilateral contact problems: numerical analysis. *SIAM Journal on Numerical Analysis*, 51(2):1295–1307, 2013.
- [17] F. Chouly, P. Hild, and Y. Renard. Symmetric and non-symmetric variants of Nitsches method for contact problems in elasticity: theory and numerical experiments. *Mathematics of Computation*, 84(293):1089–1112, 2015.
- [18] F. Chouly, M. Fabre, P. Hild, R. Mlika, J. Pousin, and Y. Renard. An overview of recent results on Nitsches method for contact problems. In *Geometrically Unfitted Finite Element Methods and Applications*, pages 93–141. Springer, 2017.
- [19] S. Claus and P. Kerfriden. A stable and optimally convergent LaTIn-CutFEM algorithm for multiple unilateral contact problems. *International Journal for Numerical Methods in Engineering*, 113(6):938–966, 2018.
- [20] S. Claus, S. Bigot, and P. Kerfriden. Vtk files: Thermal ablation using CutFEM, April 2018. URL <https://doi.org/10.5281/zenodo.1213279>.
- [21] S. Claus, S. Bigot, and P. Kerfriden. Dataset: CutFEM/Signorini, July 2018. URL <https://doi.org/10.5281/zenodo.1311829>.
- [22] A. Cosimo, V. Fachinotti, and A. Cardona. An enrichment scheme for solidification problems. *Computational Mechanics*, 52(1):17–35, 2013.

- [23] A. Curnier and P. Alart. A generalized Newton method for contact problems with friction. *Journal de mécanique théorique et appliquée*, 1988.
- [24] I. Danaïla, R. Moglan, F. Hecht, and S. Le Masson. A Newton method with adaptive finite elements for solving phase-change problems with natural convection. *Journal of Computational Physics*, 274:826–840, 2014.
- [25] A.W. Date. Novel strongly implicit enthalpy formulation for multidimensional Stefan problems. *Numerical Heat Transfer, Part B Fundamentals*, 21(2):231–251, 1992.
- [26] J. Dolbow, N. Moës, and T. Belytschko. An extended finite element method for modeling crack growth with frictional contact. *Computer Methods in Applied Mechanics and Engineering*, 190(51):6825–6846, 2001.
- [27] R. Duddu. Numerical modeling of corrosion pit propagation using the combined extended finite element and level set method. *Computational Mechanics*, 54(3):613–627, 2014.
- [28] R. Duddu, S.P.A Bordas, D. Chopp, and B. Moran. A combined extended finite element and level set method for biofilm growth. *International Journal for Numerical Methods in Engineering*, 74(5):848–870, 2008.
- [29] T. Elguedj, A. Gravouil, and A. Combescure. A mixed augmented Lagrangian-extended finite element method for modelling elastic–plastic fatigue crack growth with unilateral contact. *International Journal for Numerical Methods in Engineering*, 71(13):1569–1597, 2007.
- [30] A. Friedman and L. Jiang. A Stefan-Signorini problem. *Journal of differential equations*, 51(2):213–231, 1984.
- [31] T.P. Fries, A. Zilian, and N. Moës. Extended Finite Element Method. *International Journal for Numerical Methods in Engineering*, 86(4-5):403–403, 2011.
- [32] A. Gravouil, E. Pierres, and M. C. Baietto. Stabilized global–local X-FEM for 3d non-planar frictional crack using relevant meshes. *International Journal for Numerical Methods in Engineering*, 88(13):1449–1475, 2011.
- [33] S. Groß and A. Reusken. *Numerical methods for two-phase incompressible flows*. Springer, 2011.
- [34] S. Groß, V. Reichelt, and A. Reusken. A finite element based level set method for two-phase incompressible flows. *Computing and visualization in science*, 9(4):239–257, 2006.
- [35] J.S. Hale, L. Li, C.N. Richardson, and G.N. Wells. Containers for portable, productive, and performant scientific computing. *Computing in Science & Engineering*, 19(6):40–50, 2017.

- [36] A. Hansbo and P. Hansbo. An unfitted finite element method, based on Nitsche’s method, for elliptic interface problems. *Comput. Methods Appl. Mech. Eng.*, 191(47-48):5537–5552, 2002.
- [37] A. Hansbo and P. Hansbo. A finite element method for the simulation of strong and weak discontinuities in solid mechanics. *Comput. Methods Appl. Mech. Eng.*, 193(33-35):3523–3540, 2004.
- [38] P. Hansbo, M.G. Larson, and S. Zahedi. A cut finite element method for coupled bulk-surface problems on time-dependent domains. *Computer Methods in Applied Mechanics and Engineering*, 307:96–116, 2016.
- [39] M. Jahn and A. Luttmann. Solving the Stefan problem with prescribed interface using an XFEM toolbox for FEniCS. Technical report, Technical Report 16-03, ZeTeM, Bremen, 2016.
- [40] L.S. Jiang. Remarks on the Stefan-Signorini problem, Free Boundary Problems: Applications & Theory (III), A Bossavit, A. Damlamian, M. Premond. *Research Notes in Math.(120) Pitman*, pages 13–19, 1985.
- [41] P. Kerfriden, O. Allix, and P. Gosselet. A three-scale domain decomposition method for the 3D analysis of debonding in laminates. *Computational Mechanics*, 44(3):343–362, 2009.
- [42] A.R. Khoei and M. Nikbakht. Contact friction modeling with the extended finite element method (X-FEM). *Journal of Materials Processing Technology*, 177(1):58–62, 2006.
- [43] T. Klock. M. Jahn. A level set toolbox including reinitialization and mass correction algorithms for FEniCS. Technical report, Universitt Bremen, 2016.
- [44] D. Martin. *Multiphase Modeling of Melting: Solidification with High Density Variations Using XFEM*. PhD thesis, Université Laval, 2016.
- [45] R. Merle and J. Dolbow. Solving thermal and phase change problems with the extended finite element method. *Computational Mechanics*, 28(5):339–350, 2002.
- [46] J. Mischa and T. Klock. Numerical solution of the stefan problem in level set formulation with the extended finite element method in fenics. Technical report, Universitaet Bremen, 2017.
- [47] N. Moës, J. Dolbow, and T. Belytschko. A finite element method for crack growth without remeshing. *International Journal for Numerical Methods in Engineering*, 46(1):131–150, 1999.
- [48] D.S. Mueller-Hoeppe, P. Wriggers, and S. Loehnert. Crack face contact for a hexahedral-based XFEM formulation. *Computational Mechanics*, 49(6):725–734, 2012.

- [49] A. Narimanyan. *Stefan-Signorini moving boundary problem arisen from thermal plasma cutting: mathematical modelling, analysis and numerical solution*. PhD thesis, 2006.
- [50] A. Narimanyan. Unilateral conditions modelling the cut front during plasma cutting: FEM solution. *Applied Mathematical Modelling*, 33(1):176–197, 2009.
- [51] J. M. Navarro-Jiménez, M. Tur, J. Albelda, and J. J. Ródenas. Large deformation frictional contact analysis with immersed boundary method. *Computational Mechanics*, pages 1–18, 2017.
- [52] R. Riebaucourt, M-C. Baietto-Dubourg, and A. Gravouil. A new fatigue frictional contact crack propagation model with the coupled X-FEM/LATIN method. *Computer Methods in Applied Mechanics and Engineering*, 196(33):3230–3247, 2007.
- [53] L. Salvatori and N. Tosi. Stefan problem through extended finite elements: review and further investigations. *Algorithms*, 2(3):1177–1220, 2009.
- [54] W. Schulz, G. Simon, H.M. Urbassek, and I. Decker. On laser fusion cutting of metals. *Journal of Physics D: Applied Physics*, 20(4):481–488, 1987.
- [55] W. Schulz, M. Nießen, U. Eppelt, and K. Kowalick. Simulation of laser cutting. In *The Theory of Laser Materials Processing*, pages 25–72. Springer, 2017.
- [56] J.A. Sethian. *Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science*. Cambridge University Press, 1999.
- [57] I. Steinbach. Phase-field models in materials science. *Modelling and simulation in materials science and engineering*, 17(7):073001, 2009.
- [58] M. Storti. Numerical modeling of ablation phenomena as two-phase Stefan problems. *International Journal of Heat and Mass Transfer*, 38(15):2843–2854, 1995.
- [59] J.A. Stratton. *Electromagnetic theory*. international series in pure and applied physics. McGraw-Hill Book Company: New York, 1941.
- [60] B.A. Ton. A Stefan-Signorini problem with set-valued mappings in domains with intersecting fixed and free boundaries. *Bollettino della Unione Matematica Italiana-B*, (1), 1994.
- [61] M. Tur, J. Albelda, J.M. Navarro-Jimenez, and J.J. Rodenas. A modified perturbed Lagrangian formulation for contact problems. *Computational Mechanics*, 55(4):737–754, 2015.
- [62] V.R. Voller. An implicit enthalpy solution for phase change problems: with application to a binary alloy solidification. *Applied Mathematical Modelling*, 11(2):110–116, 1987.

- [63] N. Zabararas, B. Ganapathysubramanian, and L. Tan. Modelling dendritic solidification with melt convection using the extended finite element method. *Journal of Computational Physics*, 218(1):200–227, 2006.
- [64] Y. Zhao, C.Y. Zhao, and Z.G. Xu. Numerical study of solid-liquid phase change by phase field method. *Computers & Fluids*, 164:94–101, 2017.