



Zentrum für Technomathematik

Fachbereich 3 – Mathematik und Informatik

Numerical solution of the Stefan problem in level set formulation with the eXtended finite element method in FEniCS

Mischa Jahn

Timo Klock

Report 17-01

Berichte aus der Technomathematik

Report 17-01

January 2017

NUMERICAL SOLUTION OF THE STEFAN PROBLEM IN LEVEL SET FORMULATION WITH THE EXTENDED FINITE ELEMENT METHOD IN FENICS

M. JAHN AND T. KLOCK

ABSTRACT. In this article, we consider the Stefan problem as an example for a process with a time-dependent discontinuity. The problem is modeled in level set formulation and discretized using the extended finite element method (XFEM) with Heaviside enrichment in combination with Nitsche's method. For efficiency, a narrow band approach is used for computing the interface's velocity based on the Stefan condition as well as maintaining the level set function. The discretized problem is solved with our XFEM library `miXFEM` and the associated level set toolbox, both based upon the `FEniCS` framework. Our approach is tested by considering different model variants of the Stefan problem with known analytical solution and numerical results and convergence studies are presented.

1. INTRODUCTION

In materials science and applied physics, processes with time-dependent discontinuities are very common. From a mathematical point of view, the modeling and simulation of these type of problems is very interesting and challenging. The Stefan problem is a well known example for a such a process. Common approaches to compute a numerical solution to the Stefan problem are moving mesh methods, see e.g. [1], which are based on an explicitly defined sharp interface, and enthalpy methods, cf. for example [35], introducing the interface implicitly by considering the energy balance. Unfortunately, both methods have their drawbacks, see i.a. [9,11,21] and references therein. In moving mesh methods for example, there is not only a need for a remeshing technique but performing numerous remeshing steps during the simulation is numerically expensive, too, especially in 3D situations. Moreover, general situations including topology changes or more complex interfaces and geometries can not be considered at all. On the other hand, the enthalpy method may lack accuracy near the interface and numerical issues may arise, if problems with fluid flow and a capillary surface are considered. While there are some tricks to consider some complex situations by combining both approaches [20], a method utilizing the advantages of both methods is desirable.

A method which has proven to be very suitable for all kind of problems with arbitrary discontinuities is the extended finite element method (XFEM), see e.g. [14] for an overview. XFEM is a very flexible approach which allows for the accurate approximation of functions with strong discontinuities (jumps) and weak discontinuities (kinks) within elements by enriching the discrete function space(s) with additional basis functions at the interface location. This location is given by some indicator function whose movement is described by the level set method so it can move arbitrarily through the computational mesh.

In this article, we consider the Stefan problem in level set formulation and represent the interface by the zero level set of a signed distance function whose evolution is a-priori unknown and part of the solution. In contrast to the usual approach which is based on a weak enrichment, cf. i.a. [11,36], a Heaviside enrichment is used to enrich the function space locally for the temperature solution. As this type of enrichment generally allows for jumps in a function, we use Nitsche's method [29] to enforce internal Dirichlet conditions [12] to get a continuous temperature distribution. For efficiency, a narrow band is introduced and the level set problem as well as the interface's velocity is only computed for this region. Since the signed distance property may get lost during the evolution of the level set function, a reinitialization method and a mass correction approach are included.

The full problem is decoupled and solved using our previously developed XFEM toolbox `miXFEM` [19] and the associated level set toolbox [18] which both base the `FEniCS` framework [24]. `FEniCS` is a framework for the automated solution of PDE problems where a user can specify a problem in a specific language close to the (discretized) mathematical weak formulation and let the software generate most of the corresponding code automatically. Our toolboxes enhance the `FEniCS` framework so that problems with arbitrary time-dependent discontinuities can be tackled in the same way.

This paper is organized as follows: Starting with the governing equations for the Stefan problem, we give some brief introduction to the level set method which is a natural approach in XFEM for describing the location and movement of a discontinuity and, afterwards, state the full coupled problem. Since we use different discretization methods for the Stefan problem and the level set problem, we dedicate one section to each problem. In Section 3, we derive the time discrete formulation of the Stefan problem and define a weak formulation for each time step. The spatial discretization is based on Nitsche's method [29] treating the interface as internal Dirichlet

boundary [12]. As for the level set problem, we also derive a suitable discrete formulation and, additionally, introduce an discrete interface representation method as well as techniques to maintain the level set function in regards to the signed distance property and volume conservation. The level set method is more efficient, if used on a narrow band. This approach is also presented as well as mechanisms to compute the velocity field for the interface's evolution. In Section 5, we provide some information concerning the **FEniCS** framework, our toolboxes and the specific implementation. Results for different model examples including a convergence study are given in Section 6 together with a summary of the most important aspects of this work.

2. MATHEMATICAL SETTING

2.1. The Stefan problem. Let $\Omega \subset \mathbb{R}^d$, with $\partial\Omega$ polygonally, be a fixed domain consisting for $t \in [t_0, t_f]$ of two regions $\Omega_1(t)$ and $\Omega_2(t)$ that are separated by an interface $\Gamma(t)$. We assume $\Gamma(t)$ is sharp and sufficiently smooth for all $t \in [t_0, t_f]$ and introduce the normal vector $\vec{n}(t, \mathbf{x})$ to $\Gamma(t)$ pointing from Ω_1 into Ω_2 .

The temperature field is given by $u: \Omega \times [t_0, t_f] \rightarrow \mathbb{R}$ with $u|_{\Omega_i} = u_i$, $i \in \{1, 2\}$ and its evolution is described by

$$(2.1) \quad \rho c \frac{\partial u}{\partial t} - \nabla \cdot (\kappa \nabla u) = f, \quad \text{in } \Omega_1(t) \cup \Omega_2(t), \quad t \in (t_0, t_f).$$

For simplicity, we choose $\rho = 1$, as well as $c = 1$ and only assume that $\kappa|_{\Omega_i} = \kappa_i$ may be discontinuous with κ_i being piecewise constant in each subdomain¹. For the boundary $\partial\Omega = \Gamma_D \cup \Gamma_N$ with $\Gamma_D \cap \Gamma_N = \emptyset$, the following conditions are given

$$\begin{aligned} u &= u_D, & \text{on } \Gamma_D \times (t_0, t_f], \\ \kappa \frac{\partial u}{\partial \bar{n}} &= g_N, & \text{on } \Gamma_N \times (t_0, t_f], \end{aligned}$$

where \bar{n} denotes the outer normal to $\partial\Omega$. Initially, the temperature distribution in $\Omega_1(t_0) \cup \Omega_2(t_0)$ is given by

$$u(\cdot, t_0) = u_0$$

and, moreover, we expect the so-called isothermal interface condition

$$u(\cdot, t) = u_\Gamma \quad \text{on } \Gamma(t)$$

to hold for all times $t \in [t_0, t_f]$ and assume

$$u(\cdot, t) < u_\Gamma \quad \text{in } \Omega_1(t) \quad \text{and} \quad u(\cdot, t) > u_\Gamma \quad \text{in } \Omega_2(t).$$

As for the interface, we initially have

$$\Gamma(t_0) = \{\mathbf{x} \in \Omega : u(\mathbf{x}, t_0) = u_\Gamma\}.$$

The movement of the interface $\Gamma(t)$ in time for $t > t_0$ in terms of its normal velocity $\vec{V} \cdot \vec{n}$ is given by

$$(2.2) \quad [[\kappa \nabla u \cdot \vec{n}]] = L \vec{V} \cdot \vec{n} \quad \text{on } \Gamma,$$

with $[[\cdot]]$ denoting the jump that is defined for a function ϕ by $[[\phi]] = \phi|_{\Omega_1} - \phi|_{\Omega_2}$. Roughly speaking, this so-called Stefan condition states that the normal velocity of Γ is proportional to the jump of the temperature's gradient at the interface, with L denoting the material's latent heat.

In general, the location of the sharp interface Γ is a-priori unknown and part of the solution. Hence, a representation of Γ is needed and there are various techniques to represent it either implicitly or explicitly. As mentioned before, a very common approach in the XFEM context is to use the level set method for this purpose.

2.2. The level set method. Within the level set method [31, 34], the location of the interface Γ is given by the zero level set of a continuous function $\varphi: \Omega \times [t_0, t_f] \rightarrow \mathbb{R}$, i.e.

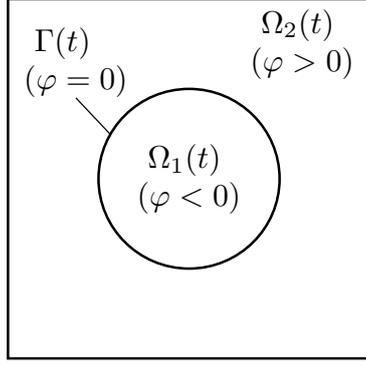
$$\Gamma(t) = \{\mathbf{x} \in \Omega : \varphi(\mathbf{x}, t) = 0\}, \quad t \in [t_0, t_f].$$

The subdomains Ω_1 and Ω_2 can be defined by $\mathbf{x} \in \Omega_1(t) \Leftrightarrow \varphi(\mathbf{x}, t) < 0$ and $\mathbf{x} \in \Omega_2(t) \Leftrightarrow \varphi(\mathbf{x}, t) > 0$. An exemplary sketch of a 2D situation where a hold-all domain Ω is divided by the sign of the function φ into subdomains $\Omega_1(t)$ and $\Omega_2(t)$ is given in Fig. 2.1a and some more level sets of φ are indicated in Fig. 2.1b.

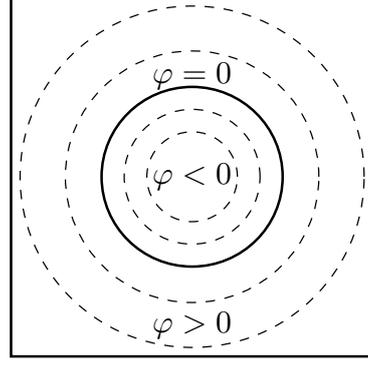
The level set method i.a. allows for an easy computation of the normal \vec{n} to Γ

$$\vec{n} = \frac{\nabla \varphi}{\|\nabla \varphi\|},$$

¹Please note that discontinuous coefficients ρ and c could easily be considered by defining the thermal diffusivity $\tilde{\kappa} := \frac{\kappa}{\rho c}$ and appropriate scaling of the right-hand-side and the boundary conditions.



(a) Domains $\Omega_1(t)$ and $\Omega_2(t)$ are separated by the zero level set $\Gamma(t)$ of φ .



(b) Visualization of some level sets of φ .

FIGURE 2.1. Visualization of the idea of the level sets method using a scalar function φ .

and the curvature K of Γ

$$K = -\operatorname{div} \vec{n} = -\operatorname{div} \frac{\nabla \varphi}{\|\nabla \varphi\|}.$$

There are various functions φ which can be defined and used within the level set method, however, from a numerical point of view it is important that $\|\nabla \varphi\|$ neither vanishes nor becomes too large, in order to get a stable computation of \vec{n} and K . Due to this, literature suggest to use a so called *signed distance function*, i.e.

$$\varphi(\mathbf{x}, t) = \begin{cases} -\min_{\tilde{\mathbf{x}} \in \Gamma(t)} \|\mathbf{x} - \tilde{\mathbf{x}}\|_2, & \text{if } \mathbf{x} \in \Omega_1(t) \\ \min_{\tilde{\mathbf{x}} \in \Gamma(t)} \|\mathbf{x} - \tilde{\mathbf{x}}\|_2, & \text{if } \mathbf{x} \in \Omega_2(t) \end{cases},$$

which satisfies $\|\nabla \varphi\| = 1$.

Given the initial value $\varphi_0(\cdot) = \varphi(\cdot, t_0)$ with zero level set $\Gamma_0 = \Gamma(t_0)$, the evolution of the level set function φ and consequently of the interface Γ in time can be described by the transport equation

$$(2.3) \quad \frac{\partial \varphi}{\partial t} + \vec{V} \cdot \nabla \varphi = 0,$$

where $\vec{V} = \vec{V}(\mathbf{x}, t)$ has to be a sufficiently smooth velocity field.

With additional boundary conditions on the inflow boundary $\partial\Omega_{\text{in}} := \{\mathbf{x} \in \partial\Omega : \vec{V}(\mathbf{x}, t) \cdot \vec{n}(\mathbf{x}) < 0, t \in [t_0, t_f]\}$ defined by a continuous function $\varphi_{\text{in}}: \partial\Omega_{\text{in}} \times [t_0, t_f] \rightarrow \mathbb{R}$, the level set problem in strong formulation is given by: Find $\varphi \in C^1(\Omega \times [t_0, t_f]) \cap C^0(\bar{\Omega} \times [t_0, t_f])$, s.t.

$$(2.4) \quad \begin{cases} \frac{\partial \varphi}{\partial t} + \vec{V} \cdot \nabla \varphi = 0 & \text{in } \Omega \times [t_0, t_f], \\ \varphi(\cdot, t_0) = \varphi_0(\cdot) & \text{in } \Omega, \\ \varphi(\cdot, t) = \varphi_{\text{in}}(\cdot, t) & \text{on } \partial\Omega_{\text{in}} \times [t_0, t_f]. \end{cases}$$

2.3. Coupled problem. Combining both models, the full coupled Stefan problem in level set formulation with non-prescribed interface as considered in this paper is given by:

Find $\varphi \in C^1(\Omega \times [t_0, t_f]) \cap C^0(\bar{\Omega} \times [t_0, t_f])$ and u sufficiently smooth, i.e. $u \in C^0(\bar{\Omega} \times [t_0, t_f])$, $u(\cdot, t)|_{\Omega_i} \in C^2(\Omega_i(t))$ and $\partial_t u(\cdot, t) \in C^0(\Omega_1(t) \cup \Omega_2(t))$ for $t \in [t_0, t_f]$ and $i = 1, 2$, such that

$$(2.5a) \quad \begin{cases} \frac{\partial u}{\partial t} - \nabla \cdot (\kappa \nabla u) = f, & \text{in } \Omega_1(t) \cup \Omega_2(t), t \in (t_0, t_f), \\ u = u_D, & \text{on } \Gamma_D \times (t_0, t_f), \\ \kappa \frac{\partial u}{\partial \vec{n}} = g_N, & \text{on } \Gamma_N \times (t_0, t_f), \\ u(\cdot, t_0) = u_0, & \text{in } \Omega_1(t_0) \cup \Omega_2(t_0), \\ u(\cdot, t) = u_\Gamma, & \text{on } \Gamma(t), \end{cases}$$

$$(2.6) \quad [\kappa \nabla u \cdot \vec{n}] = L \vec{V} \cdot \vec{n}, \quad \text{on } \Gamma(t),$$

with

$$(u_i, v_i)_{H^1(\Omega_i)} := \int_{\Omega_i} \nabla u_i \nabla v_i d\mathbf{x}, \quad i = 1, 2.$$

By using this definition and the L^2 -norm we end up with the norm

$$\left(\|\cdot\|_{L^2(\Omega)}^2 + \|\cdot\|_{H^1(\Omega_1 \cup \Omega_2)}^2 \right)^{1/2} =: \|\cdot\|_{H^1(\Omega_1 \cup \Omega_2)}$$

and define the corresponding Hilbert space

$$(3.11) \quad V_0 := \{v \in H_0^1(\Omega_1 \cup \Omega_2) : v|_{\Gamma} = 0\}$$

and the affine space³

$$(3.12) \quad V_{\Gamma} := \{v \in H_{u_D}^1(\Omega_1 \cup \Omega_2) : v|_{\Gamma} = u_{\Gamma}\}$$

where the interface conditions are introduced in a trace sense, as before. A weak formulation of the problem (3.6) is then given by: For $\xi, \kappa \in L_{\infty}(\Omega)$, $\tilde{f} \in L^2(\Omega)$ and $g_N \in L^2(\Gamma_N)$ find $u \in V_{\Gamma}$ s.t.

$$(3.13) \quad (\xi u, v)_{L^2(\Omega_1 \cup \Omega_2)} + (\kappa u, v)_{H^1(\Omega_1 \cup \Omega_2)} = (\tilde{f}, v)_{L^2(\Omega)} + (g_N, v)_{L^2(\Gamma_N)}$$

for all $v \in V_0$. Using the theorem of Lax-Milgram, one can show that there exists a unique solution to (3.13).

3.3. Discretization in space based on Nitsche's method. Now, let $\{\mathcal{S}_h\}_{h>0}$ be a family of shape regular triangulations consisting of d -simplices, with d denoting the dimension, and h is the maximum diameter $h = \max_{S \in \mathcal{S}_h} \text{diam}(S)$. Furthermore, let $\Omega_{i,h}$, $i = 1, 2$, be the discrete counterparts of Ω_i ⁴ separated by an (at this point arbitrary) approximation Γ_h of Γ and $S_i := S \cap \Omega_{i,h}$ the part of S in $\Omega_{i,h}$.

In our approach, the interface Γ_h is not considered explicitly as facets within the triangulation, so the condition $v|_{\Gamma_h} = u_{\Gamma}$ can not be included into the discrete function space in the same way as the outer Dirichlet condition on Γ_D . Thus, we introduce the function space

$$(3.14) \quad V_{h,u_D}^k := \{v \in H_{u_D}^1(\Omega_{1,h} \cup \Omega_{2,h}) : v_i \in C^0(\Omega_i), v|_{S_i} \in \mathbb{P}_k, i = 1, 2, S \in \mathcal{S}_h\}$$

for $k \in \mathbb{N}$, where we only request that for a function v only the restriction v_i has to be continuous on $\Omega_{i,h}$, $i = 1, 2$ but not on $\Omega_h = \Omega_{1,h} \cup \Omega_{2,h}$. Please note that the corresponding function space without Dirichlet condition is denoted by V_h^k . To consider the internal Dirichlet condition $u_h = u_{\Gamma}$ on Γ_h , Nitsche's method [29] is used to include this condition weakly into the discrete problem formulation. Hence, we end up treating (3.13) as two "independent" problems.

Spatial discretization. Following [12], the discrete formulation of (3.6) is given by: Find $u_h \in V_{h,u_D}^k$ s.t.

$$(3.15) \quad a(u_h, v_h) + a_1(u_h, v_h) + a_2(u_h, v_h) = L(v_h) + L_1(v_h) + L_2(v_h)$$

for all $v_h \in V_{h,0}^k$. The bilinear forms and linear forms are defined as

$$\begin{aligned} a(u_h, v_h) &= \int_{\Omega_{1,h} \cup \Omega_{2,h}} \xi u_h v_h d\mathbf{x} + \int_{\Omega_{1,h} \cup \Omega_{2,h}} \kappa \nabla u_h \nabla v_h d\mathbf{x} \\ a_1(u_h, v_h) &= - \int_{\Gamma_h} \kappa_1 \nabla u_{1,h} \cdot \vec{n}_h v_{1,h} d\mathbf{c} - \int_{\Gamma_h} \kappa_1 \nabla v_{1,h} \cdot \vec{n}_h u_{1,h} d\mathbf{c} + \int_{\Gamma_h} \lambda u_{1,h} v_{1,h} d\mathbf{c} \\ a_2(u_h, v_h) &= \int_{\Gamma_h} \kappa_2 \nabla u_{2,h} \cdot \vec{n}_h v_{2,h} d\mathbf{c} + \int_{\Gamma_h} \kappa_2 \nabla v_{2,h} \cdot \vec{n}_h u_{2,h} d\mathbf{c} + \int_{\Gamma_h} \lambda u_{2,h} v_{2,h} d\mathbf{c} \\ L(v_h) &= \int_{\Omega_{1,h} \cup \Omega_{2,h}} \tilde{f} v_h d\mathbf{x} + \int_{\Gamma_{N,h}} g_N v_h d\mathbf{s} \\ L_1(v_h) &= - \int_{\Gamma_h} \kappa_1 \nabla v_{1,h} \cdot \vec{n}_h u_{\Gamma} d\mathbf{c} + \int_{\Gamma_h} \lambda u_{\Gamma} v_{1,h} d\mathbf{c}, \\ L_2(v_h) &= \int_{\Gamma_h} \kappa_2 \nabla v_{2,h} \cdot \vec{n}_h u_{\Gamma} d\mathbf{c} + \int_{\Gamma_h} \lambda u_{\Gamma} v_{2,h} d\mathbf{c} \end{aligned}$$

cf. Appendix for more details. Here, $0 < \lambda \in \mathbb{R}$ is a stability parameter which has to be chosen large enough and can be derived analytically for some situations [12]. Please note that the signs of the terms in a_i and L_i , $i = 1, 2$,

³In this particular situation, one could also use $H_0^1(\Omega)$ resp. $H_{u_D}^1(\Omega)$ in the definitions of V_0 resp. V_{Γ} since no jumps are allowed across Γ and only a weak discontinuity, i.e. a jump in the gradients, is present. However, the present approach can easily be extended to allow for strongly discontinuities as they may occur in more general problems.

⁴We want to stress that due to our Ω with $\partial\Omega$ polygonal, we have $\Omega_h = \Omega$ without any further requirements on the boundary of the d -simplices.



FIGURE 3.1. Standard basis function v_i and additional basis function Hv_j when using the Heaviside enrichment for $d = 2$.

result from the direction of the normal vector \vec{n}_h pointing from $\Omega_{2,h}$ to $\Omega_{1,h}$. In contrast to the continuous situation, it is not trivial to show that the sum of all bilinear forms is coercive. In fact, this property depends heavily on the choice of λ . In this paper, we do not comment and investigate this issue further but assume that there is a unique solution of problem (3.15).

3.4. XFEM representation of the function space. The general idea of the extended finite element method is to represent a function by a standard and an enriched part, where the enrichment should be locally restricted. Thus, most of the simplices and degrees of freedom can be considered just as in the standard finite element context while only a minor subset needs special attention so that the assembled matrices and vectors are still sparse. While there are various approaches possible for a concrete representation of a basis of the function space (3.14), we choose a Heaviside enrichment, cf. i.a. [14, 26, 28], so that $u_h \in V_h^k$, is given by

$$u_h = \sum_{i \in \mathcal{N}} u_i v_i + \sum_{j \in \tilde{\mathcal{N}}} \hat{u}_j H v_j$$

with basis functions v_i , $i \in \mathcal{N}$, of the associated standard Lagrangian function space $\tilde{V}_h^k = \{v_h \in C(\Omega_h) : v_h|_S \in \mathcal{P}_k, \forall S \in \mathcal{S}_h\}$ and corresponding coefficients u_i . The index set of enriched basis functions $\tilde{\mathcal{N}}$ is defined by

$$\tilde{\mathcal{N}} := \{i \in \mathcal{N} : \text{meas}_{d-1}(\Gamma_h \cap \text{supp}(v_i)) > 0, v_i \in \tilde{V}_h^k\},$$

\hat{u}_j are the enriched coefficients, and

$$H(\mathbf{x}) = \begin{cases} 1, & \text{for } \mathbf{x} \in \Omega_{2,h} \\ 0, & \text{else} \end{cases}$$

is the Heaviside function. The advantage of using a strong enrichment, like the presented Heaviside enrichment, is its flexibility as it can be used for problems with strong and weak discontinuities by adding corresponding conditions using Nitsche's technique.

Although using curved intersection segments and the corresponding adaption of the quadrature rules are possible as shown in [10], $k = 1$ is chosen as polynomial degree in this paper so that Γ_h is an linear approximation of Γ , thus, making the intersecting segments S_i linear, too. Fig. 3.1 shows a visualization of a standard and the corresponding enriched basis function for a 2D setting.

Remark: As shown in [5], this enrichment is equivalent to the method proposed by [17] which is called *cut cell method* and based on duplicating nodes of intersected elements.

4. DISCRETIZATION OF THE LEVEL SET PROBLEM

The level set problem can be discretized by both, the *method of lines* and *Rothe's method*. Since using the method of lines is more common, we continue with this approach and, firstly, derive a suitable weak formulation before discretizing the problem. Please note that this subsection is just a brief overview of the approaches considered in our previous work [18] and the references therein.

4.1. Weak formulation. A weak formulation of the level set problem (2.4) can be easily derived using the time dependent function space

$$W(t)_{\vec{V}, \varphi_D} = \{v \in L^2(\Omega) : \vec{V}(\cdot, t) \cdot \nabla v \in L^2(\Omega) \wedge v|_{\partial\Omega_{\text{in}}(\cdot, t)} = \varphi_D(\cdot, t)\}.$$

By multiplying (2.3) with an arbitrary test function $v \in L^2(\Omega)$ and integrating over Ω , we end up with the weak formulation of the level set problem (2.4): For $t \in (t_0, t_f)$ find $\varphi(\cdot, t) \in W_{\vec{V}, \varphi_D}(t)$ with $\frac{\partial \varphi}{\partial t} \in L^2(\Omega)$ s.t.

$\varphi(\cdot, t_0) = \varphi_0$ and

$$(4.1) \quad \left(\frac{\partial \varphi}{\partial t}, v \right)_{L^2} + (\vec{V} \cdot \nabla \varphi, v)_{L^2} = 0, \quad \forall v \in L^2(\Omega), \quad t \in [t_0, t_f].$$

4.2. Discretization in space. For the triangulations $\{\mathcal{S}_h\}_{h>0}$ we introduce the standard Lagrangian finite element space

$$(4.2) \quad W_h^l = \{v_h \in C(\Omega_h) : v_h|_S \in \mathcal{P}_l, \forall S \in \mathcal{S}_h\},$$

and for functions with Dirichlet boundary conditions we define for $[t_0, t_f]$ the affine space

$$(4.3) \quad W_{h, \varphi_D}^l(t) = \{v_h \in C(\Omega_h) : v_h|_S \in \mathcal{P}_l, \forall S \in \mathcal{S}_h, v(\mathbf{x}) = \varphi_D(\mathbf{x}, t), \forall \mathbf{x} \in \partial\Omega_{\text{in}, h}(t)\},$$

with $l \geq 1$ and $\partial\Omega_{\text{in}, h}(t)$ being the discrete influx boundary⁵. Using these function spaces, (4.1) discretized in space reads: For $t \in [t_0, t_f]$ find $\varphi(\cdot, t) \in W_{h, \varphi_D}^l$ with $\vec{V}(t) \in L^\infty(\Omega_h)$ and $\frac{\partial \varphi_h}{\partial t} \in L^2(\Omega_h)$ such that

$$(4.4) \quad \sum_{S \in \mathcal{S}_h} \left(\frac{\partial \varphi_h}{\partial t} + \vec{V} \cdot \nabla \varphi_h, v_h \right)_{L^2(S)} = 0, \quad \forall v_h \in W_h^l.$$

In this paper as well as in many other applications like multi-phase flow, the polynomial degree $l = 2$ is chosen for the finite-dimensional function space (4.3). This is due to different reasons, for example the quality of the curvature approximation of the level set function containing second derivatives, as pointed out in [16]. Moreover, using quadratic basis functions has the additional advantage that the degrees of freedom coincide with the degrees of freedom of linear basis functions on a regularly refined mesh. This will be extensively exploited for characterizing the interface Γ discretely and by the reinitialization technique, see Section 4.5.

Remark: It is well known, that solving hyperbolic PDEs with standard finite element methods can be unstable, especially for high velocities \vec{V} . An approach to overcome this issue is using a stabilization method [33] to slightly reformulate the discretized problem to enforce stability. A method well known in literature is the *Streamline-Upwind/Petrov-Galerkin* (SUPG) stabilization [8]. In this paper however, we do not use any stabilization technique due to the small velocities and the absence of an advection term in (2.1). Further reference on this topic and the adjusted formulation of (4.4) can be found in [16] and [18].

4.3. Discretization in time. For time discretization of (4.4), the so-called θ -scheme is used. Since the time discretization of the level set problem may differ in comparison to the discretization described in Section 3.1, we now discretize the interval $[t_0, t_f]$ by $\tilde{N}_t + 1$ time steps $t_n = t_0 + n\tilde{\Delta}t$, $n = 0, \dots, \tilde{N}_t$ with $\tilde{\Delta}t$ denoting the time step. Let $\theta \in [0, 1]$ be a parameter⁶ and $\varphi_h^n(\cdot) \approx \varphi(\cdot, t_n)$ be an approximation of the level set function φ at time t_n . The completely discretized level set problem reads

$$(4.5) \quad \sum_{S \in \mathcal{S}_h} \left(\frac{\varphi_h^{n+1} - \varphi_h^n}{\tilde{\Delta}t} + \theta \vec{V}^{n+1} \cdot \nabla \varphi_h^{n+1} + (1 - \theta) \vec{V}^n \cdot \nabla \varphi_h^n, v_h \right)_{L^2(S)} = 0, \quad \forall v_h \in W_h^2.$$

4.4. Representation of Γ . An important aspect of the discretization is the discrete approximation of the interface Γ . While we assumed Γ_h to be an arbitrary approximation of Γ for the formal definition of the function spaces and the introduction of the discrete formulation of the Stefan problem in Section 3, we now describe the approach chosen in this article which follows the idea presented in [16].

For $t_n = t_0 + n\tilde{\Delta}t$, $n = 0, \dots, \tilde{N}_t$, let $\varphi_h(\cdot, t_n) \in W_h^2$ be the finite element approximation of the level set function φ , $\bar{\Gamma}_h$ its zero level, and

$$(4.6) \quad \mathcal{S}_h^\Gamma := \{S \in \mathcal{S}_h : \text{meas}_{d-1}(S \cap \bar{\Gamma}_h) > 0\}$$

the set of simplices containing $\bar{\Gamma}_h$. We drop the time t_n as an argument in this paragraph in our notation for simplicity and define $\mathcal{S}_{h/2}^\Gamma$ as the set consisting of all simplices that are obtained, if the elements in \mathcal{S}_h^Γ are regularly refined.

The finite element approximation φ_h of φ is then linearly interpolated by $I\varphi_h$ using standard Lagrange interpolation on the patch of refined elements $S \in \mathcal{S}_{h/2}^\Gamma$ and the discrete approximation of Γ is given by

$$\Gamma_h := \{\mathbf{x} \in \Omega : I\varphi_h(\mathbf{x}) = 0\},$$

as shown in Fig. 4.1 for a 2D setting. A detailed investigation about the approximation quality and the discretization error of this discrete interface representation is given in [16].

⁵Note that the time dependency of this function space is only introduced by the influx boundary and the corresponding boundary condition. If both are time independent, the function space W_{h, φ_D}^l is also time independent.

⁶Note that $\theta = 0$ leads to the explicit Euler-scheme while $\theta = 1$ results in the implicit Euler-scheme.

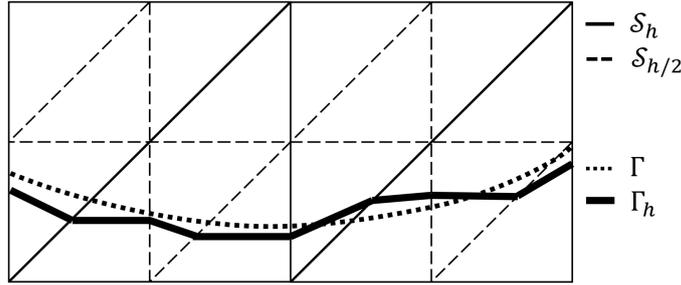


FIGURE 4.1. Construction of the discrete representation Γ_h of an interface Γ for $d = 2$

Remark: While a high order approximation of the interface Γ is possible, cf. [10], using the presented approach to get a discrete representation Γ_h has several advantages. First of all, the degrees of freedom (DOFs) of φ_h and $I\varphi_h$ coincide due to the choice of $\varphi_h \in W_h^2$, allowing for a fast interpolation. More important, the segments of Γ_S , $S \in \mathcal{S}_{h/2}$, are straight resp. planar which makes the computation of intersection points very easy. This fact is heavily utilized during the computation of the distances in the reinitialization method presented in the following section. Another advantage of this approach is that the discrete counterparts

$$\Omega_{1,h} := \Omega_{1,h}(\varphi_h) = \{\mathbf{x} \in \Omega : I\varphi_h(\mathbf{x}, t_n) < 0\}$$

and

$$\Omega_{2,h} := \Omega_{2,h}(\varphi_h) = \{\mathbf{x} \in \Omega : I\varphi_h(\mathbf{x}, t_n) > 0\}$$

at time $t_n = t_0 + n\tilde{\Delta}t$, $n = 0, \dots, \tilde{N}_t$, can be easily (re-) constructed.

4.5. Maintaining techniques. As mentioned earlier, it is beneficial to have a level set function φ_h which is close to a signed distance function. Unfortunately, this property may be lost during the evolution of the level set function in time due to various reasons, e.g. discretization errors, insufficient approximation of the curvature and topological changes. To regain the signed distance property, the level set function is reinitialized with a variant [16] of the Fast Marching Method (FMM) [34], providing a signed distance approximation $\tilde{\varphi}_h$ of φ_h . Since the FMM slightly distorts the interface Γ_h and, consequently, is not volume-preserving, we present a volume correction algorithm in Section 4.5.2 which can be applied during the reinitialization process. Please note that a more detailed description of these techniques can also be found in [16] and our previous work [18].

4.5.1. Reinitialization via FMM. Given $\varphi_h \in W_h^2$ on \mathcal{S}_h , we firstly compute the linear interpolation $I\varphi_h$ of φ_h on the regularly refined triangulation $\mathcal{S}_{h/2}$, cf. Section 4.4. Let $\mathcal{V}(S)$ denote the set of vertices given on a simplex $S \in \mathcal{S}_{h/2}$ and $\mathcal{V} := \mathcal{V}(\mathcal{S}_{h/2})$ be the (discrete) set of all vertices of $\mathcal{S}_{h/2}$ ⁷. The patch of order $l + 1$, $l \geq 1$, of elements related to a vertex $v \in \mathcal{V}(S)$, $S \in \mathcal{S}_{h/2}$ is given by

$$\mathcal{P}^{l+1}(v) := \{S \in \mathcal{S}_{h/2} : \mathcal{V}(S) \cup \mathcal{V}(\mathcal{P}^l(v)) \neq \emptyset\}$$

with

$$\mathcal{P}^1(v) := \{S \in \mathcal{S}_{h/2} : v \in \mathcal{V}(S)\}.$$

The basic idea of the FFM, consisting of two phases, is to compute distance values $\hat{d}(v)$ for any vertex $v \in \mathcal{V}$ taking thereby advantage of the fact that all information will only propagate outwards from the zero level set of the function.

In the first, so-called *initialization* phase, distance values $\hat{d}(v)$ are then calculated for $v \in \mathcal{V}(\mathcal{S}_{h/2}^\Gamma)$ by

$$\hat{d}(v) = \min_{S \in \mathcal{P}^2(v)} \text{dist}(v, \Gamma_{h,S}).$$

Please note that for stability reasons, cf. [16], we do not consider the patch $\mathcal{P}^1(v)$ but the extended patch $\mathcal{P}^2(v)$ including also all second neighbor simplices for this computation.

In the second phase referred to as *extension phase*, the initially computed values are propagated into the far field. Therefore, we introduce the finished set

$$\mathcal{V}_F = \{v \in \mathcal{V}(\mathcal{S}_{h/2}) : \hat{d}(v) \text{ is computed}\},$$

which stores already processed vertices, as well as an active set

$$\mathcal{V}_A = \{v \in \mathcal{V}(\mathcal{S}_{h/2}) : \text{At least one neighbor vertex of } v \text{ is in } \mathcal{V}_F\}$$

⁷Please note that by choosing a linear Lagrangian basis, vertices and degrees of freedom coincide.

storing vertices, which are likely to be dealt with next since they are neighbors of already processed vertices. The process of choosing the vertex $v \in \mathcal{V}_A$ which is considered next is based on a tentative distance function

$$\tilde{d}(v) = \min\{\tilde{d}_S(v) : S \in \mathcal{P}^1(v) \text{ with } \mathcal{V}(S) \cap \mathcal{V}_F \neq \emptyset\}$$

with $\tilde{d}_S(v)$ defined as

$$\tilde{d}_S(v) = \hat{d}(P_{W(S)}(v)) + \|v - P_{W(S)}(v)\|.$$

Thereby, the function $P_{W(S)}(v)$ is the minimum distance projection of v onto the convex hull $W(S) = \mathcal{V}(S) \cap \mathcal{V}_F$, i.e.

$$P_{W(S)}(v) = \operatorname{argmin}_{x \in \operatorname{conv}(W(S))} \|v - x\|.$$

With this construction, the value $\tilde{d}_S(v)$ approximates the distance of v to the discrete interface Γ_h by using the simplex S (which has at least one vertex in \mathcal{V}_F) as an information propagator. Out of all these possible values, the minimum value $\tilde{d}_S(v)$ is used as the tentative distance value respectively the most likeliest distance value, if all processed vertices/simplices are considered as information propagators.

Once values $\tilde{d}(v)$ can be calculated, the extension phase works by extracting the current nearest vertex $v^* = \operatorname{argmin}_{v \in \mathcal{V}_A} \tilde{d}(v)$, setting $\hat{d}(v^*) = \tilde{d}(v^*)$, removing this vertex from the active set \mathcal{V}_A and adding it to the finished set \mathcal{V}_F . Then the active set \mathcal{V}_A and the tentative values $\tilde{d}(v)$, $v \in \mathcal{V}_A$ are updated according to the updated finished set and the procedure is repeated until all vertices are processed.

As the result of the FMM, we obtain unsigned distance values $\tilde{d}(v)$ for every vertex v of the refined triangulation. These uniquely define a reinitialized piecewise quadratic level set function $\tilde{\varphi}_h \in W_h^2$ due to the one-to-one relation between vertices in $\mathcal{S}_{h/2}$ and DOFs in W_h^2 . Note that the values \tilde{d} are unsigned, hence a multiplication with the correct sign is necessary before using them as new DOFs of $\tilde{\varphi}_h$.

4.5.2. Volume conservation. While the reinitialized level set function $\tilde{\varphi}_h$ is close to a signed distance function, its zero level set $\tilde{\Gamma}_h$ does not coincide with the zero level set Γ_h of the original level set function φ_h any more. Consequently, the volume is not preserved during reinitialization. This is also true for the linearized level set functions $I\varphi_h$ and $I\tilde{\varphi}_h$ on $\mathcal{S}_{h/2}$.

One way to overcome this issue is to apply a local volume conservation algorithm after the initialization phase of the FMM in which the new (unsigned) distance values $\hat{d}(v)$ for $v \in \mathcal{V}(\mathcal{S}_{h/2}^\Gamma)$ are computed. In this article, we use the localized correction algorithm of [6] that is briefly explained in the following. For a more detailed description, we refer to [6] and [18].

As before, we only consider the linear level set functions $I\varphi_h$ and $I\tilde{\varphi}_h$ on $\mathcal{S}_{h/2}$. For arbitrary functions ϕ, ψ , we define the volume functional

$$\Delta V(\phi_h, \psi_h, S) = \int_{S \cap \{x \in \Omega_h : \phi < 0\}} dx - \int_{S \cap \{x \in \Omega_h : \psi < 0\}} dx, \quad S \in \mathcal{S}_{h/2}.$$

Let $\Omega_{\Gamma_h} = \cup_{S \in \mathcal{S}_{h/2}^{\Gamma_h}} S$ be the domain of all intersected simplices and $W_{h/2}^1(\Omega_{\Gamma_h})$ be the corresponding function space, cf. Eq. (4.2). The values $\hat{d}(v)$ computed in the initialization phase of the FMM then define a tentative level set function $\tilde{\phi}^{\text{ten}} \in W_{h/2}^1(\Omega_{\Gamma_h})$. To apply the volume conservation, we adjust the values of this function in the following four steps:

- (1) Calculate an offset $C_S \in \mathbb{R}$ for every $S \in \mathcal{S}_{h/2}^{\Gamma_h}$ such that

$$\Delta V(\mathcal{I}(\varphi_h), \tilde{\phi}^{\text{ten}}(\cdot) + C_S, S) = 0$$

is satisfied. Thereby the addition $\tilde{\phi}^{\text{ten}}(\cdot) + C_S$ is understood as a DOF-wise addition, i.e. C_S is added to every DOF of $\tilde{\phi}^{\text{ten}}$.

- (2) Compute a continuous, piecewise linear offset function $\phi^{\text{corr}} \in W_{h/2}^1(\Omega_{\Gamma_h})$ by averaging values C_S on $\mathcal{P}^1(v)$ so that a value $\phi^{\text{corr}}(v)$ for v connected to $S \in \mathcal{S}_{h/2}^{\Gamma_h}$ is given as

$$\phi^{\text{corr}}(v) = \frac{1}{|\mathcal{S} \in \mathcal{P}^1(v) \cap \mathcal{S}_{h/2}^{\Gamma_h}|} \sum_{S \in \mathcal{P}^1(v) \cap \mathcal{S}_{h/2}^{\Gamma_h}} C_S.$$

- (3) Search a global multiplier $C_\Omega \in \mathbb{R}$ such that the equation

$$\Delta V(\mathcal{I}(\varphi_h), \tilde{\phi}^{\text{ten}}(\cdot) + C_\Omega \phi^{\text{corr}}(\cdot), \Omega_h) = 0$$

is satisfied. Since this is already the second optimization, the resulting C_Ω is usually close to 1.

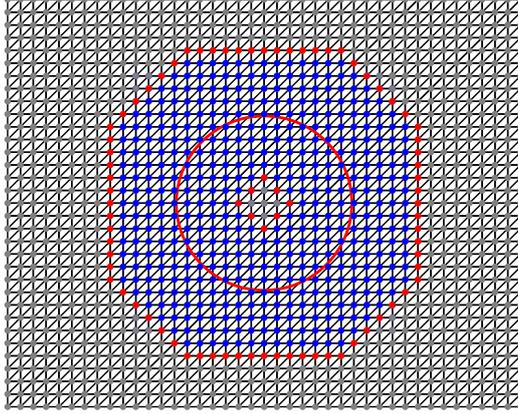


FIGURE 4.2. Different vertices in the narrow band level set method for a two-dimensional problem: \mathcal{V}_{INB} in blue, \mathcal{V}_{ONB} in blue and red.

- (4) Adjust the $\hat{d}(v)$ values for vertices that are processed in the FMM initialization phase by setting them to

$$\hat{d}(v) = \begin{cases} \hat{d}(v) - C_{\Omega} \phi^{\text{corr}}(v), & \text{if } v \in \Omega_{1,h}, \\ \hat{d}(v) + C_{\Omega} \phi^{\text{corr}}(v), & \text{if } v \in \Omega_{2,h}, \end{cases}$$

and proceed the extension phase of the FMM with these modified $\hat{d}(v)$ values.

The roots in steps 1 and 3 can e.g. be computed by using the regula falsi algorithm in the Anderson/Björk variant [4] as shown in [18].

4.6. Narrow band approach. A major drawback of the level set method as an interface representation technique is the inherent computational effort which is caused by using a higher dimensional object (the level set function) to represent the lower dimensional object (the interface). To overcome this drawback, the narrow band level set method [32] can be used. The basic idea in this is to restrict the steps of the interface evolution, namely the PDE solution and the reinitialization, to a small narrow band around the current interface.

4.6.1. Construction of the narrow band(s). Two sets of vertices are defined via a small $\gamma \in \mathbb{Z}^+$, namely the *inner* narrow band

$$(4.7) \quad \mathcal{V}_{\text{INB}} = \{v \in \mathcal{V}(\mathcal{S}_h) : \varphi_h(v) < \gamma_h\},$$

with $\gamma_h = \gamma h$ and $h = \max_{S \in \mathcal{S}_h} \text{diam}(S)$, and the *outer* narrow band

$$\mathcal{V}_{\text{ONB}} = \mathcal{V}_{\text{INB}} \cup \left(\bigcup_{v \in \mathcal{V}_{\text{INB}}} \bigcup_{S \in \mathcal{P}^1(v)} \mathcal{V}(S) \right)$$

which corresponds to all vertices of the inner narrow band set as well as all vertices of the first neighbor patch of all simplices in the inner narrow band domain. An exemplary visualization of both sets can be seen in Figure 4.2. Using these set, we define the corresponding domains $\Omega_{\text{INB}} := \{S \in \mathcal{S}_h : \mathcal{V}(S) \subset \mathcal{V}_{\text{INB}}\}$ and $\Omega_{\text{ONB}} := \{S \in \mathcal{S}_h : \mathcal{V}(S) \subset \mathcal{V}_{\text{ONB}}\}$.

Remark: The narrow band method is based on the assumption that the level set function $\varphi_h(v)$ is an approximate signed distance function, cf. (4.7) so that the DOF values in the inner narrow band are be approximately equal to the exact distance from a vertex to the interface, making reinitialization even more important.

4.6.2. Modifications to the level set problem. Based on these definitions, the concept of the narrow band level set method is to solve (4.5) on Ω_{INB} , reinitialize the solution on Ω_{ONB} and extend the function with a constant value $\pm(\gamma_h + \varepsilon)$ outside of Ω_{ONB} . Unfortunately, solving the original level set problem (4.5) on Ω_{INB} and reinitializing the solution on Ω_{ONB} often exhibits oscillations at the boundary $\partial\Omega_{\text{INB}}$ as shown in [32]. Due to this, a modified level set problem is introduced reading

$$(4.8) \quad \frac{\partial \varphi}{\partial t} + \zeta(\varphi) \vec{V} \cdot \nabla \varphi = 0 \quad \text{in } \Omega \times [t_0, t_f],$$

where

$$(4.9) \quad \zeta(\varphi) = \zeta(\varphi(x, t)) = \begin{cases} 1 & \text{if } |\varphi(x, t)| \leq \beta_{I,h}, \\ (\varphi(x, t) - \beta_{O,h})^2 \frac{2\varphi(x, t) + \beta_{O,h} - 3\beta_{I,h}}{(\beta_{O,h} - \beta_{I,h})^3} & \text{if } \beta_{I,h} < |\varphi(x, t)| \leq \beta_{O,h}, \\ 0 & \text{if } |\varphi(x, t)| > \beta_{O,h}, \end{cases}$$

is a cutoff function that slowly decreases the influence of the advection term towards the boundary of Ω_{INB} . The parameters $\beta_{I,h} = \beta_I h$ and $\beta_{O,h} = \beta_O h$ with $\beta_I < \beta_O < \gamma$ divide the inner narrow band layer into three sublayers, s.t. the cutoff parameter is equal to 1 in the innermost layer, tends to zero in the middle layer and is equal to zero in the outermost layer of the inner narrow band⁸.

As for the discretization of (4.8), we treat (4.9) explicitly w.r.t. time by defining $\zeta_h^{n+1} = \zeta(\varphi_h^n)$ to avoid the task of solving a non-linear equation. For doing so, the innermost narrow band layer width $\beta_I h$ has to be chosen to be sufficiently large. Using this approach, the discretized level set problem on the narrow domain Ω_{INB} is given as

$$(4.10) \quad \sum_{S \in \mathcal{S}_h} \left(\frac{\varphi_h^{n+1} - \varphi_h^n}{\Delta t} + \theta \zeta_h^{n+1} \vec{V}^{n+1} \cdot \nabla \varphi_h^{n+1} + (1 - \theta) \zeta_h^n \vec{V}^n \cdot \nabla \varphi_h^n, v_h \right)_{L^2(S)} = 0,$$

for $v_h \in W_h^2$.

4.6.3. *CFL conditions.* When using the narrow band level set method, one has to consider two CFL conditions:

- Since ζ decreases the transport of the level set function everywhere but in the most inner band, the velocity \vec{V} must not exceed a value which would make the interface Γ_h leave this region. Therefore, the CFL condition

$$(4.11) \quad \Delta t \| \|\vec{V}^n\|_2 \|_{L^\infty(\Omega_{\text{INB}})} < \beta_{I,h}, \quad \forall n \in \{0, \dots, \tilde{N}_t\}$$

must hold.

- For constructing $\Omega_{\text{INB}}^{n+1} \subset \Omega_{\text{ONB}}^n$, we need φ_h to be close to a signed distance function on Ω_{ONB}^n . If the velocity transporting the interface is too big, we may end up considering the constant values $\pm(\gamma h + \varepsilon)$ during the solution and reinitialization process. To avoid this, the condition

$$(4.12) \quad \Delta t \| \|\vec{V}^n\|_2 \|_{L^\infty(\Omega_{\text{INB}})} < h, \quad \forall n \in \{0, \dots, \tilde{N}_t\}.$$

has to be respected.

Remark: A typical parameter choice includes $\beta_I > 1$ so that (4.11) is automatically fulfilled, if (4.12) holds, making this the limiting condition. Please also note that even though the method's description assumes the reinitialization procedure to be applied after *every* time step, it might be better to apply reinitialization and update the narrow band after every m -th time step instead. This results in a more restrictive CFL condition given by

$$(4.13) \quad m \Delta t \| \|\vec{V}^n\|_2 \|_{L^\infty(\Omega_{\text{INB}})} < h, \quad \forall n \in \{0, \dots, \tilde{N}_t\}.$$

4.7. **Construction of a velocity field.** The solution of the level set problem is based on knowing the velocity \vec{V} . For the Stefan problem, this velocity field $\vec{V}^n \in (W_h^1)^d$ can be computed in two steps by using the Stefan condition (2.2), which discretely reads

$$(4.14) \quad [\kappa \nabla u_h^n \cdot \vec{n}_h] = L \vec{V}^n \cdot \vec{n}_h \quad \text{on } \Gamma^n, \quad n \in \{1, \dots, N_t\}.$$

In the first step, (4.14) is used to compute the velocity at the interface which is then extended to the whole narrow band in a second step, making this approach very similar to the previously presented Fast Marching Method. However, please note that the velocity field is not calculated on the regularly refined mesh since we do not need a piecewise quadratic velocity function.

4.7.1. *Initialization phase.* First of all, we compute the projections w_j , $j \in \mathcal{N} \subset \mathbb{N}$, of v onto the discrete interface Γ_h such that $\|v - w_j\| = \text{dist}(v, \Gamma_h)$ holds for all j . As there can be multiple points that satisfy the minimum distance requirement, we may have multiple projections w_j . Now, we present two approaches that can be used to compute the corresponding values $\vec{V}^n(v)$ for $v \in \mathcal{V}(S)$, $S \in \mathcal{S}_h^\Gamma$:

⁸A viable choice for these parameters is $\beta_I = 2$, $\beta_O = 4$, $\gamma = 6$.

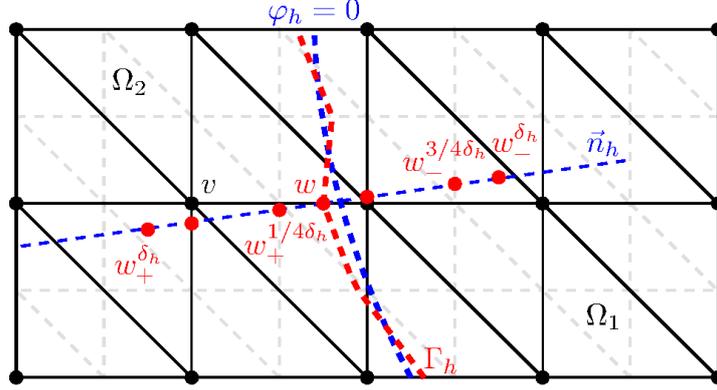


FIGURE 4.3. Evaluation points of in the DSCE velocity calculation method

- (1) **Direct gradient evaluation (DGE)** We compute the discrete temperature gradient⁹ ∇u_h^n , which is a piecewise constant, vectorial XFEM function, and the velocity field at the projections w_j *directly* using

$$\left(\vec{V}^n(w_j)\right)_k = (\nabla(u_{1,h}^n)(w_j))_k - (\nabla(u_{2,h}^n)(w_j))_k,$$

with index $k = 1, \dots, d$ denoting the respective component. The velocity vector at v is then defined by averaging over all contributions $\left(\vec{V}^n(w_j)\right)_k$ of all projections w_j that are found in the previous step.

- (2) **Discretized Stefan condition evaluation (DSCE)** [7] For every projection $w \in \{w_j : j \in \mathcal{N}\}$, we use point-value tuples

$$\left(w_{-}^{\frac{l}{4}\delta_h}, u_{1,h}^n\left(w_{-}^{\frac{l}{4}\delta_h}\right)\right) \quad \text{and} \quad \left(w_{+}^{\frac{l}{4}\delta_h}, u_{2,h}^n\left(w_{+}^{\frac{l}{4}\delta_h}\right)\right), \quad l = 0, \dots, 4,$$

with

$$w_{\pm}^{r\delta_h} = w \pm r\delta_h \vec{n}_h(w)$$

and $\delta_h = \delta h_{\max}$ a step-width parameter to perform a linear least-squares regression through these five points on each separate side, cf. Figure 4.3. The slope of these regressions is taken to approximate the gradient in normal direction at w and the resulting normal velocity is given as

$$\begin{aligned} (\vec{V}^n \cdot \vec{n}_h)(w) = & \frac{2}{L} \left[\frac{\kappa_1}{5} \frac{2u_{1,h}^n(w) + u_{1,h}^n\left(w_{-}^{\frac{1}{4}\delta_h}\right) - u_{1,h}^n\left(w_{-}^{\frac{3}{4}\delta_h}\right) - 2u_{1,h}^n\left(w_{-}^{\delta_h}\right)}{\delta_h} \right. \\ & \left. + \frac{\kappa_2}{5} \frac{2u_{2,h}^n(w) + u_{2,h}^n\left(w_{+}^{\frac{1}{4}\delta_h}\right) - u_{2,h}^n\left(w_{+}^{\frac{3}{4}\delta_h}\right) - 2u_{2,h}^n\left(w_{+}^{\delta_h}\right)}{\delta_h} \right]. \end{aligned}$$

By multiplying with \vec{n}_h a velocity field $\vec{V}^n(w) = (\vec{V}^n \cdot \vec{n}_h)(w) \cdot \vec{n}_h(w)$ can be obtained from this expression. At the end, we set the velocity field at v to the average of all contributions from the projections $w \in \{w_j : j \in \mathcal{N}\}$.

Remark: In numerical studies, we sometimes observe stability issues for the DGE method in situations with “barely intersected” simplices, i.e. for situations where we have either a very large or a very small volume ratio $|S \cap \Omega_{2,h}|/|S \cap \Omega_{1,h}|$. To overcome these problems, we neglect any simplex S and the corresponding discrete interface $\Gamma_{h,S}$ where the ratio of one subvolume to the complete simplex volume is below a small tolerance.

4.7.2. *Extension phase.* To propagate the initialized velocity values into the far field, we can make extensive use of the already presented FMM algorithm. Since the algorithm propagates (distance) values into the far field in the reinitialization, we can basically step through the same procedure and propagate velocity values additionally to the distance values. Concretely, we conduct the following 2 steps:

- (1) Calculate distance values for vertices $v \in \mathcal{V}(S)$, $S \in \mathcal{S}_h^\Gamma$ so that initialized distance and velocity field values are given after this step.

⁹Note that this gradient is a piecewise constant, vectorial XFEM function.

- (2) Propagate the distance values into the far field through the FMM extension phase and additionally propagate the velocity values. If a vertex v^* regarding \hat{d} in the distance propagation is processed, we set the velocity to

$$\vec{V}^n(v^*) = \vec{V}^n(P_{W(S_{\min})}(v^*))$$

where the value $\vec{V}^n(P_{W(S_{\min})}(v^*))$ is again calculated through the barycentric coordinates of the projection $P_{W(S_{\min})}(v^*)$ and already known velocity values.

Remark: Note that though (2) coincides highly with what we do in the reinitialization process, there is actually a minor difference that needs to be considered: There can be multiple minimizing simplices S_{\min} that minimize the current $\tilde{d}(v^*)$ function for the vertex that is processed next. While this does not matter in the reinitialization since the distance value coincide for all these minimizing simplices, the velocity values can differ. In this case, we need to set the velocity to be the average of all contributions, i.e.

$$\vec{V}^n(v^*) = \frac{1}{|\mathcal{S}_{\min}|} \sum_{S \in \mathcal{S}_{\min}} \vec{V}^n(P_{W(S)}(v^*)),$$

where \mathcal{S}_{\min} captures all simplices that minimize the tentative distance function. Also note that there is no such thing as a tentative velocity function in this procedure. The vertex that is processed next is still defined by the vertex that minimizes the tentative distance function \tilde{d} on the current active set \mathcal{V}_A . In other words, the velocity field values is information that is propagated alongside but does not interfere with the FMM procedure itself.

5. IMPLEMENTATION ASPECTS

The considered problem is solved with `miXFEM` and a level set toolbox, both developed within our work group for the `FEniCS` framework.

5.1. The FEniCS project. `FEniCS` is a collaborative project of researchers who develop tools for automated scientific computing, especially in the field of finite element methods for the solution of partial differential equations [24]. It consists of a collection of core components such as

- (1) the Unified Form Language `UFL` [3], which is a domain-specific language to specify finite element discretizations of differential equations using variational formulations close to the mathematical notation,
- (2) the `FEniCS` Form Compiler `FFC` [23,30], which analyzes given `UFL` code and, in combination with `Instant` and `FIAT` [22], generates `UFC` [2] code for arbitrary finite elements on simplices based on the variational forms specified in the `UFL` file,
- (3) `DOLFIN` [25], the main problem solving environment and user interface whose functionality integrates the other `FEniCS` components and handles communication with external libraries or toolboxes such as `miXFEM`.

5.2. miXFEM - an XFEM toolbox for FEniCS. `FEniCS` provides a lot of useful classes, structures and other utilities for solving PDE based problems with FEM. However, the `FEniCS` framework has to be extended by new modules for considering non-standard problems. In order to solve problems with arbitrary time-dependent discontinuities that may evolve and also intersect each other, we developed an XFEM toolbox [19] partly based on the PUM toolbox [27,28].

`miXFEM` adds features to the domain specific language `UFL` in order to define enriched function spaces. Additionally a new syntax for integrals on arbitrary interfaces is introduced. The `UFL` file is compiled using an extended `FEniCS` Form Compiler, which understands and interprets the new features, to generate the corresponding C++ code. Based on this code, the problems can be solved numerically using an extension of the `DOLFIN` library, implemented in C++. While some key features are presented in [19], a detailed technical description of all features will be given in an upcoming publication.

5.3. Level set toolbox. Another extension to `FEniCS` used to solve problem (2.5)-(2.7) is a level set toolbox, which is used to compute the evolution of the level set function resp. the discontinuity. The toolbox consists of discretized weak formulations for different time stepping schemes formulated in `UFL` and compiled with the `FFC` which are used by a C++ library. This library provides an implementation of the presented Fast Marching Method and the volume correction approach which can be used for various problems. A detailed description is given in [18]. Recently, the narrow band approach has been included to improve the efficiency of the implemented methods.

5.4. Problem related aspects for solving the Stefan problem in level set formulation.

Decoupling of thermal problem and level set problem. Since the problems (2.5) and (2.7) are coupled by the Stefan condition (2.6), a numerical decoupling strategy is needed to solve the complete problem. In this article, we use the simple approach to solve the problems in succession:

Given all data for t_n , we firstly solve the level set problem to obtain φ_h^{n+1} and the new interface Γ_h^{n+1} based on the old data of the thermal problem. The computed interface Γ_h^{n+1} is then used to construct new XFEM function spaces and to solve the thermal problem for u^{n+1} . Using the approaches presented in Section 4.7, the interface's normal velocity $\vec{V}^{n+1} \cdot \vec{n}$ and the velocity field \vec{V} are computed. In our approach, we use the implicit Euler scheme for time discretization in both subproblems, however, we still may need intermediate time steps for solving the level set problem due to the CFL conditions, as explained in the next paragraph.

Time stepping. As mentioned before, the discretization of the time interval $[t_0, t_f]$ for the thermal problem (2.5) in Section 3.1 and for the transport problem (2.7) in Section 4.3 do not necessarily have to coincide. The reason for this can be that, firstly, we may use different time stepping methods for both subproblems, e.g. the Crank-Nicolson method for the level set problem and the implicit Euler method for the thermal problem, and, secondly, we may need smaller time steps due to the arising CFL conditions when using the narrow band approach. Hence, we have to synchronize the time step sizes for the subproblems in order to compute a numerical solution of the coupled problem.

For this purpose, we use the time step size Δt of the thermal problem as major time step size and the values $t_n = t_0 + n\Delta t$ with $n \in \{0, \dots, N_t\}$, cf. Section 3.1, as so-called synchronization points. Based on this, we adjust the time step size Δt_φ for the level set problem, if necessary, so that the discretization method with time step size $\tilde{\Delta}t$ and the CFL condition(s) with time step size Δt_{CFL} are respected. This means, we may have to introduce potentially non-equidistant intermediate time steps $t_{n,i}$ in order to reach the (next) synchronization point(s). The described procedure is illustrated in Fig. 5.1. As we will see in Example 2, this can influence the solution process and the convergence behavior.

Now, the complete numerical approach for solving the Stefan problem in level set formulation is shown in Algorithm 1.

6. RESULTS

The presented numerical approach consists of a level set solver including maintaining techniques, an XFEM framework, and some methods for tackling the Stefan problem. Since we already verified the level set toolbox, cf. [18], and the XFEM framework for time-dependent problems, see [19], we now focus on the overall results of the complete approach. In particular, a numerical convergence study is performed with respect to different time step sizes Δt and varied maximum cell diameters h . Similar to our previous work with prescribed interface [19], two academical examples with known analytical solutions u are examined and the considered errors for $u - u_h$ are the L^2 -error

$$\|u - u_h\|_{L^\infty(L^2)} := \max_{t \in (t_0, t_f)} \|u(\cdot, t) - u_h(\cdot, t)\|_{L^2(\Omega)}$$

and the (semi) H^1 -error

$$\|\nabla u - \nabla u_h\|_{L^2(L^2)} := \sqrt{\int_{t_0}^{t_f} \|\nabla u(\cdot, t) - \nabla u_h(\cdot, t)\|_{L^2(\Omega)}^2 dt.}$$

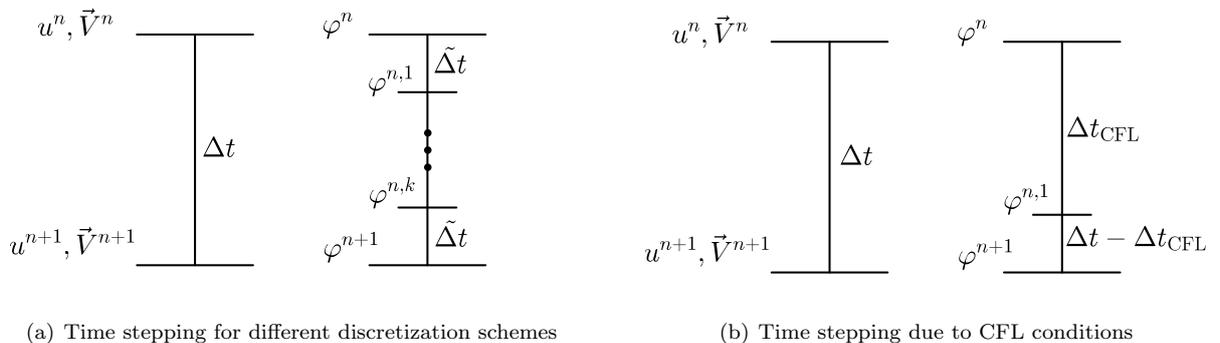


FIGURE 5.1. Time stepping synchronization: Intermediate time steps for the solution of the level set problem are synchronized with the time step size for the thermal problem.

Algorithm 1 Solver for the two-phase Stefan problem in level set formulation

Input: $\Omega, \Gamma_D, \Gamma_N, \mathcal{S}_h, \varphi_0, u_0, u_\Gamma, f, \kappa, L, u_D, g_N, \Delta t, \beta_O, \beta_I, \gamma$.

Output: $u_h^n, \Gamma_h^n, \varphi_h^n, \vec{V}_h^n$ for $n = 0, \dots, N_T$.

Initialization:

Obtain φ_h^0 by reinitializing the given level set function φ_0 .

Construct the function space $V_{h,u_D}^1(t_0)$ with φ_h^0 .

if the narrow band method is applied **then**

Initialize Ω_{INB} and Ω_{ONB} with φ_h^0 .

end if

Interpolate u_0 onto V_{h,u_D}^1 to obtain the initial temperature u_h^0 .

Time stepping:

for $n = 0, \dots, N_t - 1$ **do**

Derive the velocity field \vec{V}_h^n from u_h^n with one of the methods that are presented in Section 4.7.

Assign $t_\varphi = n\Delta t$ (current simulation time) and $\varphi_h^{t_\varphi} = \varphi_h^n$.

while $t_\varphi < (n+1)\Delta t$ **do**

Assign largest allowed time step to Δt_{CFL} such that CFL conditions (4.11) and (4.13) are satisfied.

Calculate time step for level set propagation $\Delta t_\varphi = \min\{\Delta t_{\text{CFL}}, (n+1)\Delta t - t_\varphi\}$.

Propagate level set function to obtain an updated level set function $\varphi_h^{t_\varphi + \Delta t_\varphi}$.

if reinitialization is necessary **then**

Replace $\varphi_h^{t_\varphi + \Delta t_\varphi}$ by its reinitialized version.

if narrow band method is applied **then**

Update inner and outer narrow band regions $\Omega_{\text{INB}}, \Omega_{\text{ONB}}$ with the reinitialized function $\varphi_h^{t_\varphi + \Delta t_\varphi}$.

end if

end if

Set $t_\varphi = t_\varphi + \Delta t_\varphi$.

end while

Set $\varphi_h^{n+1} = \varphi_h^{t_\varphi}$.

Construct the new discrete interface Γ_h^{n+1} from φ_h^{n+1} .

Construct the new XFEM space $V_{h,u_D}^1(t_{n+1})$ with Γ_h^{n+1} .

Solve for the new temperature approximation u_h^{n+1} .

end for

The order of convergence for each error is as usual determined for varied Δt and h and the results are compared to the convergence rates using a standard finite element method for the heat equation using the implicit Euler scheme which at best, cf. [13], are given by

$$\|u - u_h\|_{L^\infty(L^2)} = \begin{cases} \mathcal{O}(h^{k+1}) \\ \mathcal{O}(\Delta t) \end{cases}$$

and

$$\|\nabla u - \nabla u_h\|_{L^2(L^2)} = \begin{cases} \mathcal{O}(h^k) \\ \mathcal{O}(\Delta t) \end{cases}.$$

For both examples, a regular structured triangular mesh is used. The range of chosen cell diameters h and time step sizes Δt are specified for each example individually in the respective subsection. In regards to the time stepping scheme, we use the implicit Euler method in both examples for all subproblems. While we present examples for 2D situations only, please note that the same methods can be used in a straight forward way for 3D problems as well.

6.1. Example 1: Straight interface. Choosing $\Omega := (0,1)^2$ with $\Gamma_D := \{(x,y) \in \partial\Omega \mid y = 0 \vee y = 1\}$ and $\Gamma_N := \partial\Omega \setminus \Gamma_D$ for the geometry and $\kappa_1 := 1, \kappa_2 := 2, L := 2, u_\Gamma := 0$ for the material parameters, an analytical

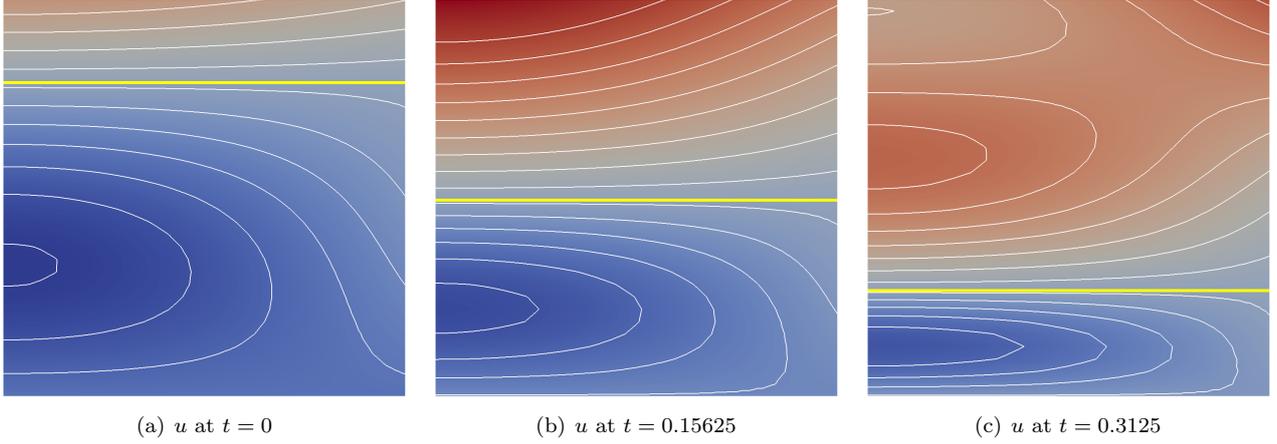


FIGURE 6.1. Example 1: Different level sets including the zero level set (thick yellow line) of the analytical solution u at different time instants t .

solution to problem (2.5)-(2.7) for $[t_0, t_f] := [0, 5 \cdot 2^{-4}]$ is given by

$$u(\mathbf{x}, t) = \begin{cases} \cos\left(\frac{\pi x}{2}\right) \sin\left(\frac{\pi \varphi(\mathbf{x}, t)}{y - \varphi(\mathbf{x}, t)}\right) + \varphi(\mathbf{x}, t) & \text{on } \Omega_1(t) \\ \cos\left(\frac{\pi x}{2}\right) \sin\left(\frac{\pi \varphi(\mathbf{x}, t)}{2(y - \varphi(\mathbf{x}, t))}\right) + \frac{1}{2}\varphi(\mathbf{x}, t) + e^y + \frac{11}{11t+5} & \text{on } \Omega_2(t) \end{cases},$$

with $\mathbf{x} = (x, y) \in \Omega$. The corresponding interface $\Gamma(t)$ to this solution is a straight horizontal line moving downwards which is characterized by the zero level set of the level set function

$$\varphi(\mathbf{x}, t) := y - \ln\left(\frac{11}{11t+5}\right).$$

The source term f for the right-hand-side, the boundary functions u_D and g_N and the initial conditions have to be chosen with respect to the specified analytical solution and can be easily computed. The analytical solution u is shown in Figure 6.1 at different time instants and the results of the convergence analysis are shown in Figure 6.2.

In the latter, one can see that the problem is dominated by the spatial error reaching the optimal convergence order 2, cf. Fig 6.2(a). Moreover, an interesting effect can be seen in Fig. 6.2(b) (left) for the coarsest mesh size where the error $\|u - u_h\|_{L^\infty(L^2)}$ increases for decreasing time step sizes. This behavior is an indirect consequence of the narrow band approach and the reinitialization method. As we have to reinitialize the level set function after every time step, the total number of reinitialization procedures increases with decreasing time step size Δt . As mentioned, every reinitialization slightly alters the interface position leading to a different normal velocity, provided by evaluating the Stefan condition, thereby influencing the complete solution process. Therefore, the errors may not decrease but sometimes increase for very small time step sizes, if the spatial discretization is coarse and fixed.

6.2. Example 2: Circular interface. On $\Omega := (-1, 1)^2$ with $\Gamma_N := \partial\Omega$ consider for $[t_0, t_f] := [0, \frac{3}{4}]$ the function

$$u(\mathbf{x}, t) = \begin{cases} A\left(\|\mathbf{x}\|^2 - R(t)^2\right) & \text{on } \Omega_1(t) \\ A\left(\|\mathbf{x}\|^2 - R(t)^2\right) - R(t) - \dot{R}(t)\left(\|\mathbf{x}\| - R(t)\right) + \frac{\dot{R}(t)\left(\dot{R}(t) + \frac{1}{R(t)}\right)}{2\left(\|\mathbf{x}\| - R(t)\right)^2} & \text{on } \Omega_2(t) \end{cases}.$$

with $A > \frac{\dot{R}(t)}{2R(t)}$, to ensure $u < 0$ on $\Omega_1(t)$ and $u > 0$ on $\Omega_2(t)$, and $R(t) := R_0 + \frac{1}{2} \sin(\pi t)$. Choosing $R_0 = 0.3$, $A = 4.1$ as well as $\kappa_1 := \kappa_2 := 1$, $L := 1$ and $u_\Gamma := 0$ for the parameters, this is a solution to problem (2.5)-(2.7) where the interface $\Gamma(t)$ corresponds to a circle with radius $R(t)$, which is centered at the origin and is expanding till $t = 0.5$ and then shrinking again. $\Gamma(t)$ can be characterized by the level set function

$$\varphi(\mathbf{x}, t) := R^2(t) - \|\mathbf{x}\|^2.$$

As before, the right-hand-side term f , the Neumann boundary function g_N and the initial conditions have to be chosen with respect to the specified analytical solution and can be derived easily. The analytical solution u is shown in Figure 6.3 at different time instants.

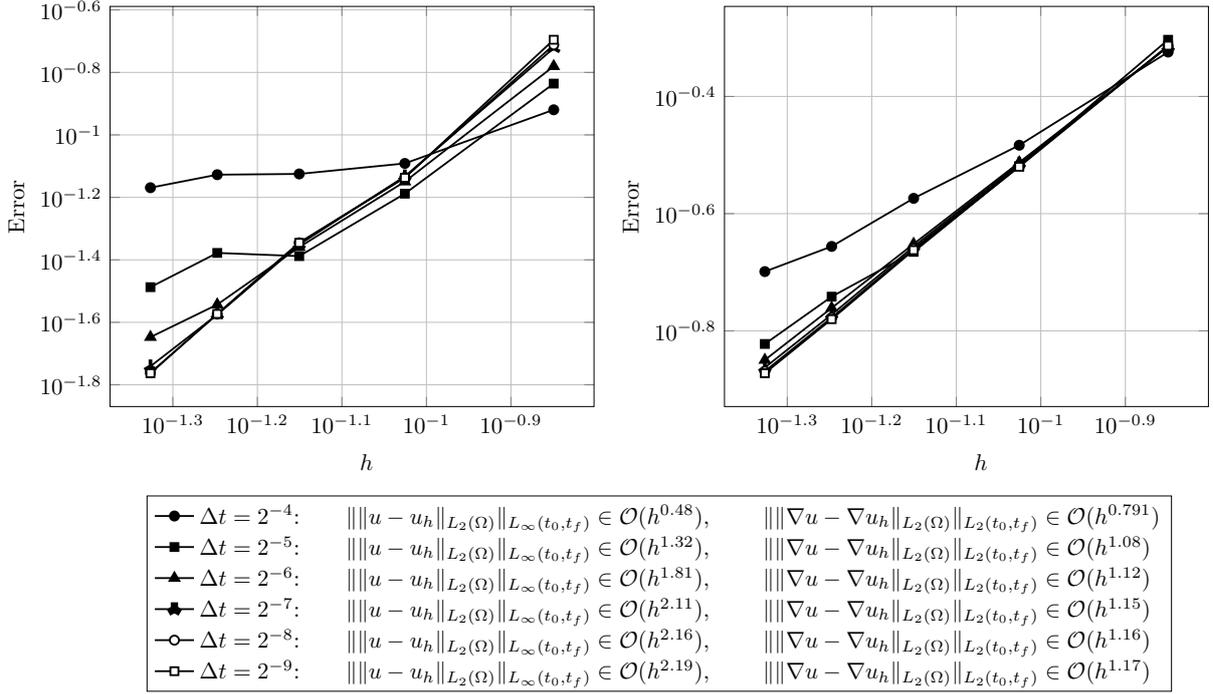
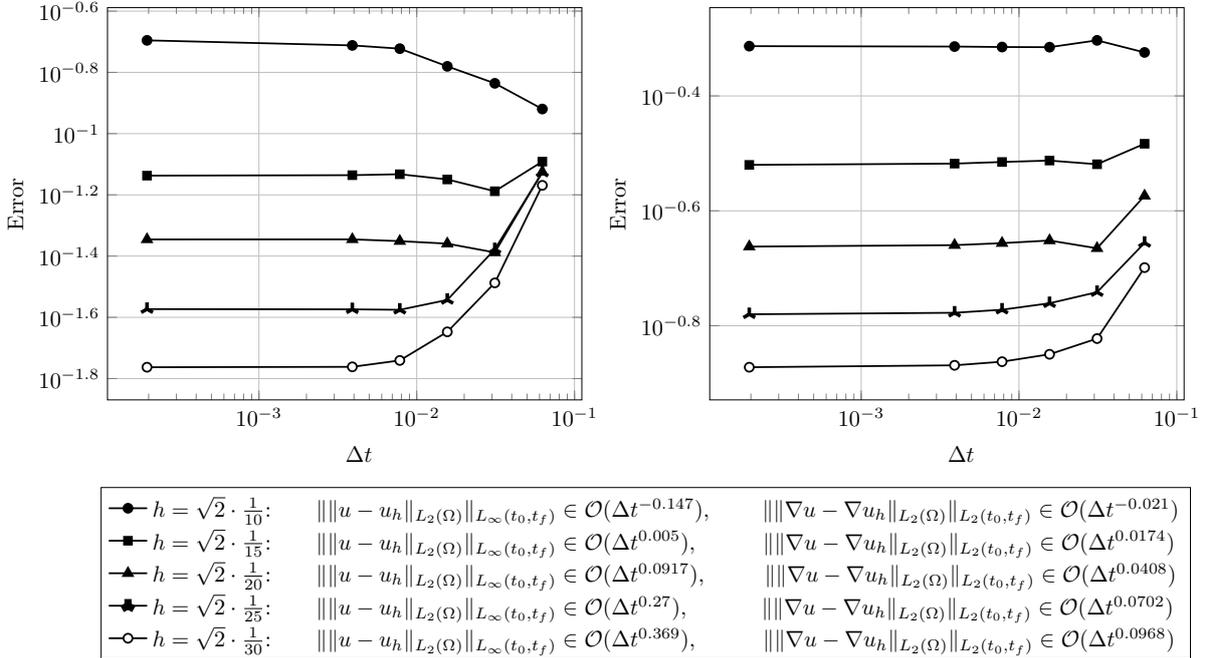

 (a) $\|u - u_h\|_{L^\infty(L^2)}$ (left) and $\|\nabla u - \nabla u_h\|_{L^2(L^2)}$ (right) over h for different time step sizes

 (b) $\|u - u_h\|_{L^\infty(L^2)}$ (left) and $\|\nabla u - \nabla u_h\|_{L^2(L^2)}$ (right) over Δt for different h

FIGURE 6.2. Convergence tests for example 1 including approximated orders of convergences.

The convergence behavior is visualized in Fig. 6.4. Therein one can see that this problem is also highly dominated by the spacial error and qualitatively similar to Example 1. In regards to the convergence order, however, we can only archive suboptimal results. One reason for this is that although φ can be exactly approximated by φ_h if using quadratic basis functions, the linear polygonal approximation $\Gamma_h(t)$ of the circular interface $\Gamma(t)$ introduces an additional error propagating through the solution process, which is different compared to the

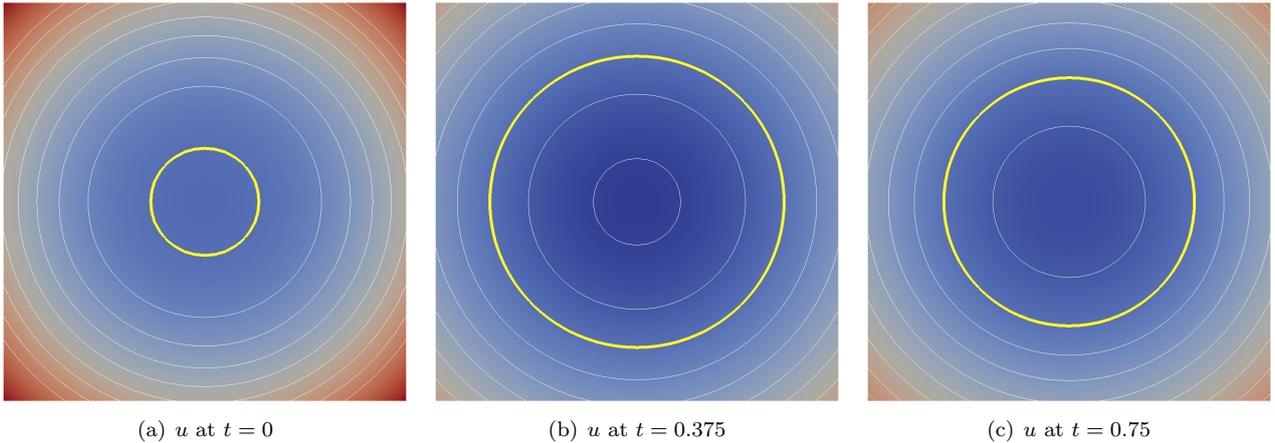


FIGURE 6.3. Example 2: Different level sets including the zero level set (thick yellow line) of the analytical solution u at different time instants t .

situation in Example 1. Additionally, the same problem as described in the previous section arises, i.e. for small time steps we have to reinitialize more often introducing thereby a secondary approximation error. However, an even more significant impact on the convergence behavior has the narrow band approach, respectively the CFL condition (4.12): In addition to using the reinitialization procedure after each time step, the CFL condition may also request intermediate time steps, see Section 5.4, especially for big Δt . In such a situation, the narrow band has to be updated and the update process relies on the signed distance property making an additional reinitialization step mandatory *for every intermediate time step* as well as very regular one. As a consequence, the introduced errors also propagate through the solution process and, for some situations, prevent a convergent behavior.

Consider exemplary the graph of the error $\|u - u_h\|_{L^\infty(L^2)}$ for $\Delta t = 2^{-4}$ in Fig. 6.4(a)(left): As the mesh size decreases, (4.12) requests more intermediate time steps introducing thereby more reinitialization procedures which in turn modify the interface position and, hence, the solution. Due to the complex character of the example, small deviations of the interface position have a high impact on the solution of the thermal problem so that, finally, we end up with bigger errors on finer meshes than on coarser meshes. Alternatively, it can be seen in Fig. 6.4(b) (left) that for a fixed spacial discretization at some point the error $\|u - u_h\|_{L^\infty(L^2)}$ does not decrease any more but slightly increase as a results of the more and more reinitialization procedures.

7. SUMMARY

In this article, we present a numerical approach to solve the Stefan problem using the level set method and XFEM. The problem is decoupled by solving the level set problem and the thermal problem in succession. For this purpose, the thermal problem is discretized using Nitsche’s method for internal Dirichlet boundaries and a (local) strong enrichment based on a Heaviside function. The evolution of the interface is described by a level set problem restricted to a narrow band region. In addition to maintaining techniques for the shape of the level set function, two approaches are presented to compute the corresponding propagation velocity field using the Stefan condition.

All described methods are implemented using our toolbox `miXFEM` for the `FEniCS` framework and are used for solving two academical examples of different complexities with known analytical solutions. In both examples, good results can be achieved with optimal rates of convergence (Example 1) or suboptimal rates of convergence (Example 2).

Due to the modular implementation, maintaining methods for the level set problem, more precisely, the reinitialization, which is mandatory for the narrow band approach, could be identified as “the weakness of the approach”. However, this is just a technical issue arising within the convergence studies where very different mesh sizes and time step sizes are considered.

Last but not least, we want to point out that due to the general approach, this numerical method is not limited to the Stefan problem but works for all problems with arbitrary time-dependent discontinuities, e.g. multi-phase flow.

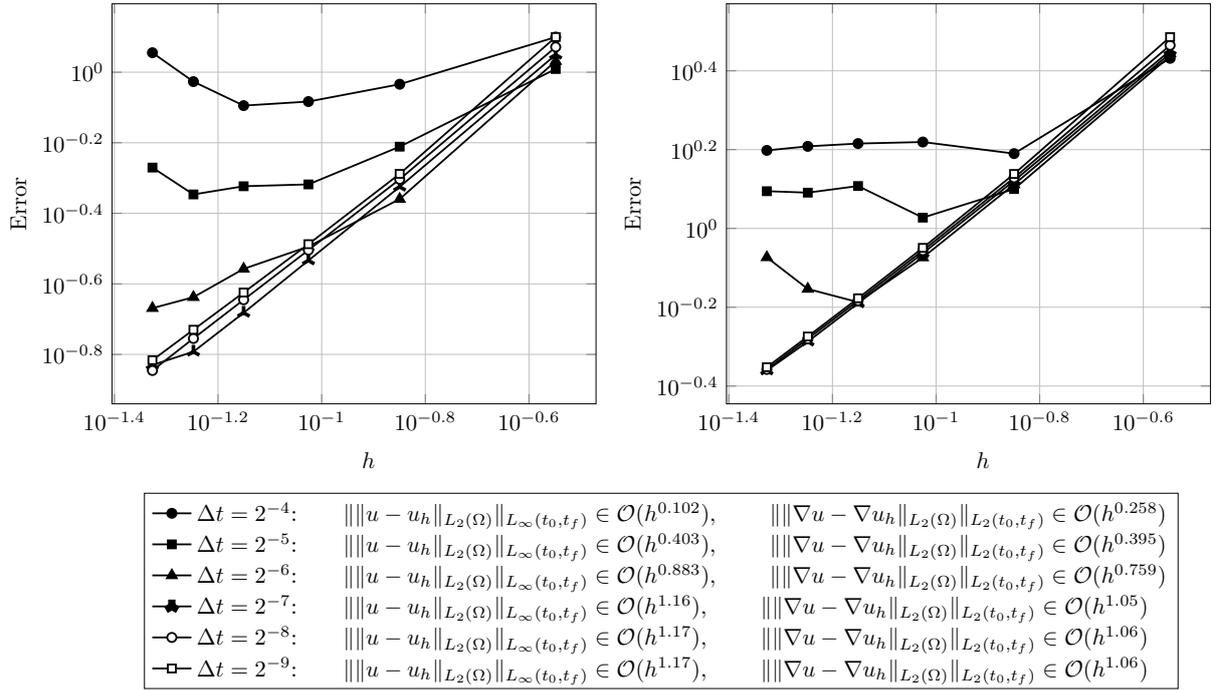
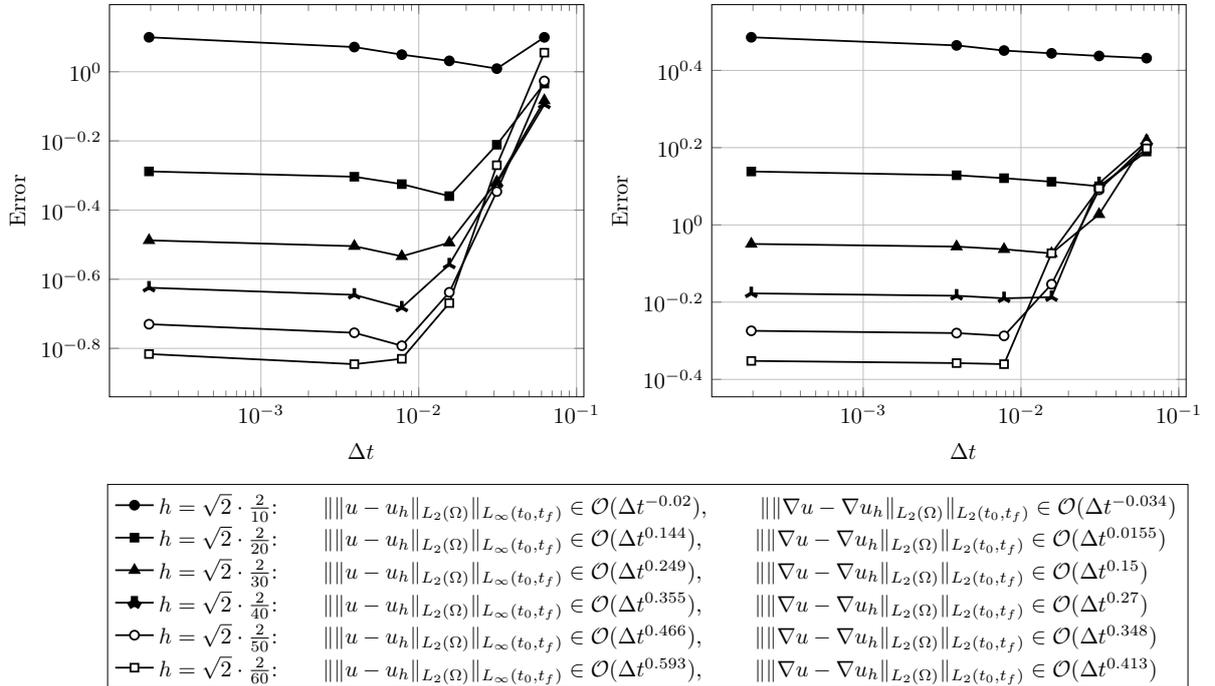

 (a) $\|u - u_h\|_{L^\infty(L^2)}$ (left) and $\|\nabla u - \nabla u_h\|_{L^2(L^2)}$ (right) over h for different time step sizes

 (b) $\|u - u_h\|_{L^\infty(L^2)}$ (left) and $\|\nabla u - \nabla u_h\|_{L^2(L^2)}$ (right) over Δt for different h

FIGURE 6.4. Convergence tests for example 2 including approximated orders of convergences.

ACKNOWLEDGEMENT

The authors gratefully acknowledge the financial support by the DFG (German Research Foundation) for the subproject A3 within the Collaborative Research Center SFB 747 ‘‘Mikrokaltumformen - Prozesse, Charakterisierung, Optimierung’’.

8. APPENDIX

The discrete formulation of (3.6) is derived as follows: Multiply (3.6) with a test function $v \in V_h$ and integrate over the domain $\Omega_{1,h} \cup \Omega_{2,h}$

$$\int_{\Omega_{1,h} \cup \Omega_{2,h}} \xi u_h v_h d\mathbf{x} - \int_{\Omega_{1,h} \cup \Omega_{2,h}} \nabla \cdot (\kappa \nabla u_h) v_h d\mathbf{x} = \int_{\Omega_{1,h} \cup \Omega_{2,h}} \tilde{f} v_h d\mathbf{x}.$$

Integration by parts leads to

$$\begin{aligned} & \int_{\Omega_{1,h} \cup \Omega_{2,h}} \xi u_h v_h d\mathbf{x} + \int_{\Omega_{1,h} \cup \Omega_{2,h}} \kappa \nabla u_h \nabla v_h d\mathbf{x} - \int_{\partial(\Omega_{1,h} \cup \Omega_{2,h})} \kappa \nabla u_h \cdot \vec{n}_h v_h d\mathbf{c} \\ &= \int_{\Omega_{1,h} \cup \Omega_{2,h}} \xi u_h v_h d\mathbf{x} + \int_{\Omega_{1,h} \cup \Omega_{2,h}} \kappa \nabla u_h \nabla v_h d\mathbf{x} - \int_{\Gamma_D} \kappa \nabla u_h \cdot \vec{n} \underbrace{v_h}_{=0} d\mathbf{c} \\ & - \int_{\Gamma_N} \underbrace{\kappa \nabla u_h \cdot \vec{n}}_{=g_N} v_h d\mathbf{c} - \int_{\Gamma_h} \kappa_1 \nabla u_{1,h} \cdot \vec{n}_h v_{1,h} d\mathbf{c} + \int_{\Gamma_h} \kappa_2 \nabla u_{2,h} \cdot \vec{n}_h v_{2,h} d\mathbf{c} \\ &= \int_{\Omega_{1,h} \cup \Omega_{2,h}} \tilde{f} v_h d\mathbf{x}. \end{aligned}$$

Now, we add the ‘‘artificial’’ terms

$$0 = \mp \int_{\Gamma_h} \kappa_{\{1,2\}} \nabla v_{\{1,2\},h} \cdot \vec{n}_h u_{\{1,2\},h} d\mathbf{c} \pm \int_{\Gamma_h} \kappa_{\{1,2\}} \nabla v_{\{1,2\},h} \cdot \vec{n}_h u_{\{1,2\},h} d\mathbf{c}$$

and

$$0 = \int_{\Gamma_h} \lambda_{\{1,2\}} u_{\{1,2\},h} v_{\{1,2\},h} d\mathbf{c} - \int_{\Gamma_h} \lambda_{\{1,2\}} u_{\{1,2\},h} v_{\{1,2\},h} d\mathbf{c}$$

so the equation reads

$$\begin{aligned} & \int_{\Omega_{1,h} \cup \Omega_{2,h}} \xi u_h v_h d\mathbf{x} + \int_{\Omega_{1,h} \cup \Omega_{2,h}} \kappa \nabla u_h \nabla v_h d\mathbf{x} \\ & - \int_{\Gamma_h} \kappa_1 \nabla u_{1,h} \cdot \vec{n}_h v_{1,h} d\mathbf{c} - \int_{\Gamma_h} \kappa_1 \nabla v_{1,h} \cdot \vec{n}_h u_{1,h} d\mathbf{c} + \int_{\Gamma_h} \kappa_1 \nabla v_{1,h} \cdot \vec{n}_h u_{1,h} d\mathbf{c} \\ & + \int_{\Gamma_h} \kappa_2 \nabla u_{2,h} \cdot \vec{n}_h v_{2,h} d\mathbf{c} + \int_{\Gamma_h} \kappa_2 \nabla v_{2,h} \cdot \vec{n}_h u_{2,h} d\mathbf{c} - \int_{\Gamma_h} \kappa_2 \nabla v_{2,h} \cdot \vec{n}_h u_{2,h} d\mathbf{c} \\ & + \int_{\Gamma_h} \lambda_1 u_{1,h} v_{1,h} d\mathbf{c} - \int_{\Gamma_h} \lambda_1 u_{1,h} v_{1,h} d\mathbf{c} + \int_{\Gamma_h} \lambda_2 u_{2,h} v_{2,h} d\mathbf{c} - \int_{\Gamma_h} \lambda_2 u_{2,h} v_{2,h} d\mathbf{c} \\ & = \int_{\Omega_{1,h} \cup \Omega_{2,h}} \tilde{f} v_h d\mathbf{x} + \int_{\Gamma_N} g_N v_h d\mathbf{c} \end{aligned}$$

Now, we put one of each artificial term onto the right-hand-side and make use of the condition $u_{\{1,2\},h} = u_\Gamma$ on Γ_h and define $\lambda_1 = \lambda_2$

$$\begin{aligned} & \underbrace{\int_{\Omega_{1,h} \cup \Omega_{2,h}} \xi u_h v_h d\mathbf{x} + \int_{\Omega_{1,h} \cup \Omega_{2,h}} \kappa \nabla u_h \nabla v_h d\mathbf{x}}_{=:a(u_h, v_h)} \\ & - \underbrace{\int_{\Gamma_h} \kappa_1 \nabla u_{1,h} \cdot \vec{n}_h v_{1,h} d\mathbf{c} - \int_{\Gamma_h} \kappa_1 \nabla v_{1,h} \cdot \vec{n}_h u_{1,h} d\mathbf{c} + \int_{\Gamma_h} \lambda u_{1,h} v_{1,h} d\mathbf{c}}_{=:a_1(u_h, v_h)} \\ & + \underbrace{\int_{\Gamma_h} \kappa_2 \nabla u_{2,h} \cdot \vec{n}_h v_{2,h} d\mathbf{c} + \int_{\Gamma_h} \kappa_2 \nabla v_{2,h} \cdot \vec{n}_h u_{2,h} d\mathbf{c} + \int_{\Gamma_h} \lambda u_{2,h} v_{2,h} d\mathbf{c}}_{=:a_2(u_h, v_h)} \\ & = \underbrace{\int_{\Omega_{1,h} \cup \Omega_{2,h}} \tilde{f} v_h d\mathbf{x} + \int_{\Gamma_N} g v_h d\mathbf{c}}_{=:L(v_h)} \\ & - \underbrace{\int_{\Gamma_h} \kappa_1 \nabla v_{1,h} \cdot \vec{n}_h u_\Gamma d\mathbf{c} + \int_{\Gamma_h} \lambda u_\Gamma v_{1,h} d\mathbf{c}}_{=:L_1(v_h)} - \underbrace{\int_{\Gamma_h} \kappa_2 \nabla v_{2,h} \cdot \vec{n}_h u_\Gamma d\mathbf{c} + \int_{\Gamma_h} \lambda u_\Gamma v_{2,h} d\mathbf{c}}_{=:L_2(v_h)}. \end{aligned}$$

REFERENCES

- [1] M. R. Albert and K. O'Neill. Moving boundary-moving mesh analysis of phase change using finite elements with transfinite mappings. *International Journal for Numerical Methods in Engineering*, 23(4):591–607, 1986.
- [2] M. S. Alnaes, A. Logg, K.-A. Mardal, O. Skavhaug, and H. P. Langtangen. Unified framework for finite element assembly. *International Journal of Computational Science and Engineering*, 4(4):231–244, 2009.
- [3] M. S. Alnaes, A. Logg, K. B. Olgaard, M. E. Rognes, and G. N. Wells. Unified form language: A domain-specific language for weak formulations of partial differential equations. *ACM Trans. Math. Softw.*, 40(2):9:1–9:37, March 2014.
- [4] N. Anderson and Å Björck. A new high order method of regula falsi type for computing a root of an equation. *BIT Numerical Mathematics*, 13(3):253–264, 1973.
- [5] P. Areias and T. Belytschko. A comment on the article “A finite element method for simulation of strong and weak discontinuities in solid mechanics” by A. Hansbo and P. Hansbo [Comput. Methods Appl. Mech. Engrg. 193 (2004) 3523–3540]. *Computer Methods in Applied Mechanics and Engineering*, 195:1275–1276, February 2006.
- [6] R. F. Ausas, E. A. Dari, and G. C. Buscaglia. A geometric mass-preserving redistancing scheme for the level set function. *International Journal for Numerical Methods in Fluids*, 65(8):989–1010, 2011.
- [7] M. Bernauer. Motion Planning for the Two-Phase Stefan problem in Level Set Formulation. *Dissertation*, 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. *Comput. Methods Appl. Mech. Eng.*, pages 199–259, 1990.
- [9] E. Bänsch, J. Paul, and A. Schmidt. An ale finite element method for a coupled stefan problem and navier–stokes equations with free capillary surface. *International Journal for Numerical Methods in Fluids*, 71(10):1282–1296, 2013.
- [10] K. W. Cheng and T.-P. Fries. Higher-order XFEM for curved strong and weak discontinuities. *International Journal for Numerical Methods in Engineering*, 82(5):564–590, 2010.
- [11] 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.
- [12] J. Dolbow and I. Harari. An efficient finite element method for embedded interface problems. *International Journal for Numerical Methods in Engineering*, 78(2):229–252, 2009.
- [13] G. Dziuk. *Theorie und Numerik partieller Differentialgleichungen*. De-Gruyter-Studium. De Gruyter, 2010.
- [14] T.-P. Fries and T. Belytschko. The extended/generalized finite element method: An overview of the method and its applications. *International Journal for Numerical Methods in Engineering*, 84(3):253–304, 2010.
- [15] T.-P. Fries and A. Zilian. On time integration in the xfem. *International Journal for Numerical Methods in Engineering*, 79(1):69–93, 2009.
- [16] S. Gross and A. Reusken. *Numerical Methods for Two-phase Incompressible Flows*. Springer Series in Computational Mathematics. Springer, 2011.
- [17] A. Hansbo and P. Hansbo. An unfitted finite element method, based on nitsche’s method, for elliptic interface problems. *Computer Methods in Applied Mechanics and Engineering*, 191(47–48):5537 – 5552, 2002.
- [18] M. Jahn and T. Klock. A level set toolbox including reinitialization and mass correction algorithms for FEniCS. Technical Report 16-01, ZeTeM, Bremen, 2016.
- [19] M. Jahn and A. Luttmann. Solving the stefan problem with prescribed interface using an XFEM toolbox for FEniCS. Technical Report 16-03, ZeTeM, Bremen, 2016.
- [20] M. Jahn, A. Luttmann, A. Schmidt, and J. Paul. Finite element methods for problems with solid-liquid-solid phase transitions and free melt surface. *PAMM*, 12(1):403–404, 2012.
- [21] M. Jahn and A. Schmidt. Finite element simulation of a material accumulation process including phase transitions and a capillary surface. Technical Report 12-03, ZeTeM, Bremen, 2012.
- [22] R. C. Kirby. Algorithm 839: Fiat, a new paradigm for computing finite element basis functions. *ACM Trans. Math. Softw.*, 30(4):502–516, December 2004.
- [23] R. C. Kirby and A. Logg. A compiler for variational forms. *ACM Trans. Math. Softw.*, 32(3):417–444, September 2006.
- [24] A. Logg, K.-A. Mardal, and G. N. Wells, editors. *Automated Solution of Differential Equations by the Finite Element Method*, volume 84 of *Lecture Notes in Computational Science and Engineering*. Springer, 2012.
- [25] A. Logg and G. N. Wells. DOLFIN: Automated finite element computing. *ACM Trans Math Software*, 37(2):20:1–20:28, 2010.
- [26] R. Merle and J. Dolbow. Solving thermal and phase change problems with the extended finite element method. *Computational Mechanics*, 28(5):339–350.
- [27] M. Nikbakht. *Automated Solution of Partial Differential Equations with Discontinuities using the Partition of Unity Method*. PhD thesis, 2012.
- [28] M. Nikbakht and GN Wells. Automated modelling of evolving discontinuities. *Algorithms*, 2:1008–1030, 2009.
- [29] J. Nitsche. Über ein variationsprinzip zur lösung von dirichlet-problemen bei verwendung von teilräumen, die keinen randbedingungen unterworfen sind. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 36(1):9–15, 2013.
- [30] K. B. Olgaard and G. N. Wells. Optimizations for quadrature representations of finite element tensors through automated code generation. *ACM Trans. Math. Softw.*, 37(1):8:1–8:23, January 2010.
- [31] S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations. *J. Comput. Phys.*, 79(1):12–49, November 1988.
- [32] D. Peng, B. Merriman, S. Osher, H. Zhao, and M. Kang. A PDE-Based Fast Local Level Set Method. *Journal of Computational Physics*, 155:410–438, November 1999.
- [33] H.G. Roos, M. Stynes, and L. Tobiska. *Robust Numerical Methods for Singularly Perturbed Differential Equations: Convection-Diffusion-Reaction and Flow Problems*. Springer Series in Computational Mathematics. Springer, 2008.
- [34] J. A. Sethian. A fast marching level set method for monotonically advancing s. *Proceedings of the National Academy of Sciences*, 93(4):1591–1595, 1996.
- [35] V. R. Voller, C. R. Swaminathan, and B. G. Thomas. Fixed grid techniques for phase change problems: A review. *International Journal for Numerical Methods in Engineering*, 30(4):875–898, 1990.
- [36] N. Zabarav, B. Ganapathysubramanian, and L. Tan. Modelling dendritic solidification with melt convection using the extended finite element method. *J. Comput. Phys.*, 218(1):200–227, October 2006.

THE CENTER FOR INDUSTRIAL MATHEMATICS AND MAPEX CENTER FOR MATERIALS AND PROCESSES, UNIVERSITY OF BREMEN,
BREMEN, GERMANY

E-mail address: `mischa@math.uni-bremen.de`

SIMULA RESEARCH LABORATORY, OSLO, NORWAY

E-mail address: `timo@simula.no`