An Enriched Shifted Boundary Method to Account For Moving Fronts

Tiffanie Carlier, Léo Nouveau, Heloise Beaugendre, Mathieu Colin, Mario Ricchiuto

An Enriched Shifted Boundary Method to Account For Moving Fronts

T. Carlier[a], L. Nouveau[b], H. Beaugendre[a], M. Colin[a], M. Ricchiuto[c]

[a] *Univ. Bordeaux, INRIA, CNRS, Bordeaux INP, IMB, UMR 5251, F-33400, Talence, France, tiffanie.carlier@inria.fr, heloise.beaugendre@inria.fr, mathieu.colin@inria.fr,*

[b] *Univ Rennes, INSA Rennes, CNRS, IRMAR - UMR 6625, 35000 Rennes, France, leo.nouveau@insa-rennes.fr,*

[c] *INRIA, Univ. Bordeaux, CNRS, Bordeaux INP, IMB, UMR 5251, 200 Avenue de la Vieille Tour, 33405 Talence cedex, France, mario.ricchiuto@inria.fr,*

Abstract

We present an extension of the shifted boundary method to simulate partial differential equations with moving internal interfaces. The objective is to apply the method to phase change problems modelled by the Stefan equation : the parabolic heat equation with a discontinuity separating the two phases, moving at a speed given by the normal flux jump. To obtain an accurate prediction of the temperature field on both sides of the discontinuity, and of the position of the discontinuity itself, we propose an enhanced variant of the shifted boundary method. This method is based on an enriched mixed form proposed by some of the present authors, which allows a uniform second order accuracy in space and time. Stabilization terms are added on the whole domain to ensure convergence. The specificities generated by the interface displacement are handled by a particular construction, with a double nodal points structure at the surrogate interface. Several recovering techniques based on Taylor developments from the interface to its surrogate, and a least square minimization method for node initialization are performed to recover the missing values generated by the moving interface and the phase structure of the model.

Keywords: Enriched shifted boundary method, Temperature enrichment, Stefan mixed formulation, Moving front, Jump condition, Finite element method

1. Introduction

Since the pioneer work of C. Peskin in 1972 [1], immersed and embedded boundary methods have been an active field of research. The key idea behind such strategies is to avoid to discretize explicitly the geometry at stake in the mesh, considerably simplifying the mesh generation process, known to be one of the most time consuming step on a whole simulation for complex and/or moving boundaries. For complex geometries, one just needs to generate a mesh of the domain, regardless of the embedded interface, providing more flexibilities. Initially designed for cartesian grids, the process allows the simulation of geometries that could not be treated conformally [1, 2, 3, 4, 5, 6]. When dealing with moving interfaces or shocks, several body-fitted strategies have proven their robustness, such as the Lagrangian framework (mainly for solid mechanics or shocks) [7, 8, 9, 10], or Arbitrary Lagrangian Eulerian formalism (especially for fluid-structure interactions) [11, 12, 13, 14, 15, 16]. Such methods are characterized by a mesh deformation, that can match the interface displacement. However, they are still required for large deformations, remeshing and interpolation steps, which are time consuming, especially

regarding highly parallel computations. On the opposite, immersed techniques only require the advection of the interface defined implicitly throughout the simulation. The counterpart of those methods is the quick loss of accuracy in the imposition of the corresponding boundary conditions, which has been the main focus in their developments. Immersed Boundary (IB) strategies solve the problem on the entire domain and enforce the boundary condition through a diffuse forcing term, and can be assimilated to penalty methods. Among them, one can think of the original IB proposed by Peskin [1], and its developments (see for instance [17, 18, 19, 20] among others), or penalization to account for rigid bodies [3, 21, 22, 23]. Exhaustive reviews have been proposed by Mittal and Iaccarino [24], Sotiropoulos and Yang [25], and Griffith and Patankar [26]. On the other hand, embedded methods will tend to solve the problem only in the region of interest, and handle the boundary conditions with some specific treatments at a discrete level. Among those methods, for structured grids with Finite Volume or Finite Difference schemes, a common approach is the ghost penalty method, where the region of interest is extended on the other side of the boundary, see e.g. [6, 27, 28, 29]. For unstructured meshes, one can look at the work of Farhat and co-authors in the FIVER code, where the boundary conditions are imposed accurately using expansions from the position of the interface to the boundaries of the finite volume cells, see for instance [14, 30, 31]. Within the Finite Element framework, most well-known and famous strategies rely on the cut-cell technique. The geometry is recovered through intersections with the background mesh [32, 33, 34, 35, 36, 37], and boundary conditions are imposed weakly with for instance Nitsche's method or Lagrange multipliers. Nevertheless, such strategies, although allowing to recover the optimal accuracy of the underlying numerical scheme, can be complex to implement and suffer from the "small cut-cell" problem. The possible definition of new cells with very small measure can considerably impact the conditioning of the associated algebraic system if no specific stabilization is applied [38, 39, 40]. Innovative methods have been proposed as alternatives, to avoid the need of intersecting the embedded geometry and the embedding mesh, and the small cut-cells conditioning problem. Among those methods, one can find the ϕ-FEM based on level-set (see [41] for Dirichlet condition and [42] for Neumann one), the aggregated finite element method [43, 44], and the Shifted Boundary Method [45] that proposes, in a similar way than the FIVER code, to impose on a surrogate boundary an accurately modified boundary condition.

For those reasons, IBM are particularly well-suited for the Stefan problem, which aims at modeling time-dependent phase change, such as a liquid-solid system, ruled by the heat equation. Here, the interest is to follow the front of the phase-change area where the material coexists in a liquid and a solid state. Its thickness is usually between a few Angstroms and a few centimeters, but we can make the assumption that it has a zero thickness [46]. The other particularity of this model is the position of the interface which is also an unknown, as ruled by the heat flux. In a more general way, we are interesting in solving the heat equation, with an embedded interface defining a discontinuity in the heat coefficient, with temperature and heat flux jump conditions. Strategies have been proposed within the cutFEM context [34], and immersed interface method [20]. With the aim to provide an accurate method, preserving the underlying scheme accuracy, without resorting to cut-cell strategy, we propose here to employ the SBM, introduced by Main and Scovazzi for Laplace and Stokes problems in [45]. The idea is to weakly enforce the boundary conditions using Nitsche's method [47] on a surrogate boundary, defined by edges/faces of the background mesh. This surrogate interface being defined by elements of the underlying grid, no intersections with the physical interface are required, which avoids the definition of new small cells, and is pretty straightforward to settle. To account for the discrepancy between physical and surrogate interfaces, and recover optimal accuracy, the boundary conditions are modified, "shifted" from the physical boundary to the surrogate one, using Taylor expansions.
This strategy has proven its robustness and accuracy for a variety of problem, including Navier-

Stokes [48], free surface flows [49], solid mechanics [50], with thorough analysis in [51] and [52]. Regarding the problem at stake, a formulation has been proposed in [47] for the steady heat equations with discontinuous heat coefficients and jump conditions at the interface. However, in the present work, the quantities of interest are the temperature, but also its gradient (heat flux), responsible for the front motion. In order to provide a more accurate computation of this gradient, this quantity is considered as an unknown, and the problem is solved in its mixed form. This allows, as proposed in [53, 54], to perform an enrichment of the temperature, leading to a quadratic approximation on $\mathbb{P}^1$ finite elements and an improvement of the overall accuracy of the schemes. In [55], this strategy permits to recover for the steady Darcy problem with the SBM, embedded Dirichlet and Neumann conditions at least second order accuracy on both the primal quantity and its gradient. We propose to employ a similar strategy to provide a new formulation, embedding the jump conditions defined at our front, providing a second order approximation of the temperature and heat flux. We first define the weak stabilized finite element formulation employed for the problem with a static interface. Degrees of freedom at the surrogate interface being duplicated, when switching to moving fronts, the initialization of the variables at those added degrees of freedom needs to be performed properly, to account for the polynomial degree of approximation of the considered variable (quadratic for the temperature and linear for the heat flux) and not degrade the overall accuracy. We show here that a least square reconstruction enables to preserve the willing precision. The Backward Difference Formula (BDF2) time discretization is presented, to keep an homogeneous error. In this article, we provide the general process with second order time discretization for the motion of the interface, preserving the accuracy in the resolution of the Stefan problem. Numerical validations are performed to assess the polynomial degrees of approximation of the variables, convergence tests are run to prove the predicted convergence rates, and a final study on a classical melting/ice propagation is provided to validate the proposed scheme and methodology.

The paper is organized as follows: In Section 2, we give a description of the physical problem stated both in its primal form and in its mixed formulation by considering the flux as a variable. Then, in Section 3, a quick description of the Shifted Boundary Method is done, and all the necessary elements for the implementation are described. We then present a situation where the front is conformed to the geometry before going to the embedded formulation. We also introduce an enrichment process, in order to increase the accuracy of the scheme. Section 4 is devoted to the temporal discretization of our problem. Finally, in Section 5, we give several numerical computations which validate our approach.

2. Problem statement

Consider a domain Ω, bounded and connected open set in $\mathbb{R}^d$ ($d \geq 2$), with Lipschitz boundary $\partial\Omega$. The boundary $\partial\Omega$ is decomposed in several parts Γ_D and Γ_N such that $\Gamma_D \cup \Gamma_N = \partial\Omega$ and $\Gamma_D \cap \Gamma_N = \emptyset$. At time $t > 0$ the domain Ω is composed of two subdomains $\Omega_+(t)$ and $\Omega_-(t)$ separated by an interface $\Gamma(t) = \overline{\Omega_+}(t) \cap \overline{\Omega_-}(t)$ (see Figure 1). We denote by $\Gamma_+(t) = \overline{\Omega_+}(t) \cap \Gamma(t)$ and $\Gamma_-(t) = \overline{\Omega_-}(t) \cap \Gamma(t)$. Using the fact that at the interface Γ the outward normal vectors $\boldsymbol{n}_+$ and $\boldsymbol{n}_-$ are equals up to a sign we introduce

$$\boldsymbol{n} = \boldsymbol{n}_+ = -\boldsymbol{n}_- \tag{1}$$

We consider the classical Stefan problem, a time dependent model of phase change for transitions in a liquid-solid system. The model characterizes a variety of phenomena and can be used to describe the solidification or melting processes of pure materials. In such cases, the heat flux is responsible for the front displacement. In the sequel, the phase-change area is referred as the interface.

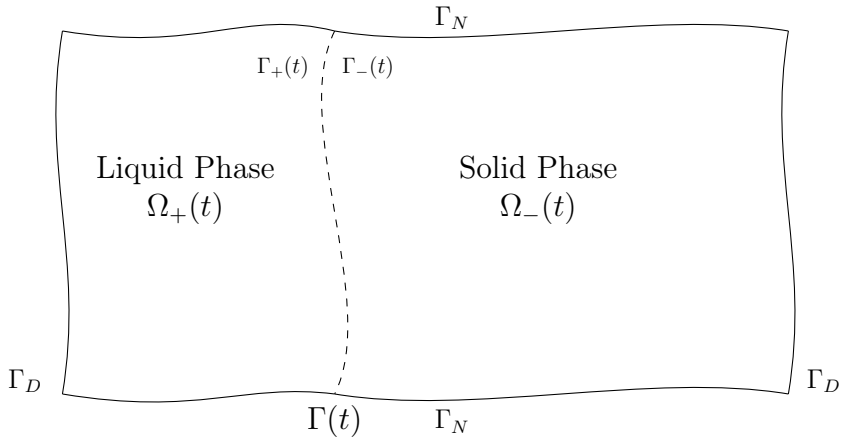


Figure 1: Visualization of the domain

Thus, we study the distribution of the temperature $T(\boldsymbol{x}, t)$ in Ω, a given material of density ρ. We denote by λ the thermal conductivity of the material, *i.e.* its ability to conduct heat. In general, the thermal conductivity is a tensor, but we consider here the isotropic case (conduction independent of the directions). A multiphase situation being considered, the conductivity can be discontinuous: λ_{solid} for the value inside the solid phase Ω_-, usually higher than λ_{liquid}, value in the liquid phase Ω_+. The specific heat of the material is denoted by c. It is also phase dependent and characterizes the necessary amount of heat to increase one unit of mass by one unit of temperature.
Those discontinuities in conductivity and specific heat, located at the interface Γ can lead to discontinuities. To keep all the mathematical generality, we consider at this stage, discontinuities on T and its gradients. Later one, for the physical test case, the temperature will be continuous and equal to the melting temperature at the moving interface. We will then denote by T^+ and T^- the values of T on respectively Γ^+ and Γ^-:

$$\begin{aligned} T^+ &= \lim_{x \in \Omega^+ \to x \in \Gamma} T(\boldsymbol{x}, t), \\ T^- &= \lim_{x \in \Omega^- \to x \in \Gamma} T(\boldsymbol{x}, t). \end{aligned} \tag{2}$$

For convenience, we introduce the jump operator $[\![\cdot]\!]_\Gamma$ and average operator $\{\cdot\}_\Gamma$, defined for a any scalar variable f or vector quantity $\boldsymbol{g}$ as

$$\begin{aligned} [\![f]\!]_\Gamma &= f^+ - f^-, \qquad & [\![\boldsymbol{g}]\!]_\Gamma &= \boldsymbol{g}^+ - \boldsymbol{g}^-, \\ \{f\}_\Gamma &= \frac{1}{2}(f^+ + f^-), & \{\boldsymbol{g}\}_\Gamma &= \frac{1}{2}(\boldsymbol{g}^+ + \boldsymbol{g}^-). \end{aligned} \tag{3}$$

Let us remark that for any functions ψ and ϕ, one has

$$[\![\psi]\!]_\Gamma = [\![\phi]\!]_\Gamma \{\psi\}_\Gamma + \{\phi\}_\Gamma [\![\psi]\!]_\Gamma. \tag{4}$$

Finally, since the interface Γ is moving, we denote by $\boldsymbol{\chi}(t)$ its position.

2.1. Primal form

In this section, we present the heat equation. Given an initial temperature T_0 and a position $\boldsymbol{\chi}(0)$ of the interface, the Stefan problem is, in its primal form

$$\begin{aligned}
&5.a) \quad \rho c \partial_t T(\boldsymbol{x},t) - \nabla \cdot (\lambda \nabla T(\boldsymbol{x},t)) = f(\boldsymbol{x},t) && \text{in } \Omega \times (0,t),\\
&5.b) \quad T(\boldsymbol{x},t) = T_D && \text{on } \Gamma_D \times (0,t),\\
&5.c) \quad \lambda \nabla T(\boldsymbol{x},t) \cdot \boldsymbol{n} = -h_N && \text{on } \Gamma_N \times (0,t),\\
&5.d) \quad T(\boldsymbol{x},0) = T_0 && \text{in } \Omega.
\end{aligned} \tag{5}$$

In order to close System (5.a)-(5.d), since Γ is a moving interface, we need boundary conditions on Γ. In the sequel, we propose two sets of boundary conditions. The first one consists in imposing the jump of temperature and the jump of normal heat flux across the interface:

$$\begin{cases}
[\![T(\boldsymbol{x},t)]\!]_\Gamma = j_T(\boldsymbol{x},t),\\
[\![\lambda \nabla T(\boldsymbol{x},t)]\!]_\Gamma \cdot \boldsymbol{n} = -\sigma(\boldsymbol{x},t).
\end{cases} \tag{6}$$

The second one prescribes the value of the temperature at the interface, as a Dirichlet condition:

$$\begin{cases}
T^+ = T_L,\\
T^- = T_R,
\end{cases} \tag{7}$$

where T_L and T_R are given data. In the latter case, the normal interface velocity can be derived from the recovered flux jump at the interface, through the Stefan condition:

$$\rho L_m \boldsymbol{\chi}'(t) \cdot \boldsymbol{n} = -[\![\lambda \nabla T]\!]_\Gamma \cdot \boldsymbol{n} \tag{8}$$

where L_m is the latent heat. In such cases, the interface position $\boldsymbol{\chi}(t)$ becomes an unknown of the problem. Note that the interface condition (7) can also be expressed as a condition on the temperature jump and temperature average:

$$\begin{cases}
[\![T]\!]_\Gamma = T_L - T_R = j_T,\\
\{T\}_\Gamma = \dfrac{T_L + T_R}{2} = a_T.
\end{cases} \tag{9}$$

The primal form can bring some advantages depending on the context of application. Nevertheless, in this work, when considering the Stefan condition (8), since the front motion is driven by the heat flux, it is necessary to accurately describe the gradient of the temperature. This can be achieved by using the mixed formulation of the problem, which combined with a temperature enrichment, can provide a better precision of the front velocity and then its position. In addition, in [56] it has been shown that the primal form can expose to a loss of accuracy and that mass conservation is not guaranteed either.

2.2. Mixed form

The mixed formulation of the Stefan problem is obtained by decoupling the original System (5). We introduce the local heat flux $\boldsymbol{\beta}$, corresponding to the amount of energy that flows through one unit of area per time unit and defined as $\boldsymbol{\beta} := -\lambda \nabla T$. The expression of $\boldsymbol{\beta}$ can be interpreted as the Fourier's law, which is a classical relation in the theory of heat conduction.

The mixed formulation of problem (5) then can be written into the form

$$\begin{aligned}
&10.a) \quad \rho c \partial_t T(\boldsymbol{x},t) + \nabla \cdot \boldsymbol{\beta}(\boldsymbol{x},t) = f(\boldsymbol{x},t) && \text{in } \Omega \times (0,t),\\
&10.b) \quad \boldsymbol{\beta}(\boldsymbol{x},t) = -\lambda \nabla T(\boldsymbol{x},t) && \text{in } \Omega \times (0,t),\\
&10.c) \quad T(\boldsymbol{x},t) = T_D && \text{on } \Gamma_D \times (0,t),\\
&10.d) \quad \boldsymbol{\beta}(\boldsymbol{x},t) \cdot \boldsymbol{n} = h_N && \text{on } \Gamma_N \times (0.t),\\
&10.e) \quad T(\boldsymbol{x},0) = T_0 && \text{in } \Omega.
\end{aligned} \tag{10}$$

where the unknowns are now the temperature distribution $T(\boldsymbol{x},t)$, the heat flux $\boldsymbol{\beta}(\boldsymbol{x},t)$ and possibly the interface position $\Gamma(t)$.
Taking into account the introduction of $\boldsymbol{\beta}$, one can rewrite Equation (6) as follows

$$\begin{cases} [\![T(\boldsymbol{x},t)]\!]_\Gamma = j_T(\boldsymbol{x},t), \\ [\![\boldsymbol{\beta} \cdot \boldsymbol{n}]\!]_\Gamma = \sigma(\boldsymbol{x},t), \end{cases} \tag{11}$$

while the Stefan condition (8) is

$$\rho L_m \boldsymbol{\chi}'(t) \cdot \boldsymbol{n} = [\![\boldsymbol{\beta}(\boldsymbol{x},t)]\!]_\Gamma \cdot \boldsymbol{n}. \tag{12}$$

3. The Shifted Boundary Method

The **S**hifted **B**oundary **M**ethod (**SBM**) is an embedded method which enables the use of non-body-fitted meshes. It is a valuable asset for solving time dependent problem with a moving front, which prevents to perform remeshing at each displacement of the interface.
The SBM can be summarized in two steps explained in details in [45]. The first one consists in defining an approximated interface, conformal to the grid, called the surrogate interface, on which the interface condition is imposed. In the second step we weakly impose the interface conditions using Nitsche's method. Roughly speaking, we impose this conditions through the weak formulation rather than in the finite element space [45]. A mapping is used to link the physical interface to its surrogate and is employed to properly modify the imposed conditions accounting for the discrepancy between the two interfaces. We now detail this procedure.

Let us consider an embedded grid that does not conform to the physical interface Γ. The surrogate, denoted by $\tilde{\Gamma}$, is composed by the edges (or faces if $d > 2$) that are the closest to the physical interface. The choice has been made to identify all the elements cut by the physical interface and to identify $\tilde{\Gamma}$ has one of the possible combinations of the outer edges of these elements (only two combinations possible in 2D, see Figure 2).

The symbol $\tilde{\ }$ over a notation always refers to a quantity defined on the surrogate. In order to connect the surrogate interface to the physical interface, we need to introduce a mapping $\mathbf{M}$ as follows:

$$\begin{aligned} \mathbf{M} \ : \ & \tilde{\Gamma} \to \Gamma \\ & \tilde{\boldsymbol{x}} \mapsto \boldsymbol{x} \quad . \end{aligned} \tag{13}$$

This mapping $\mathbf{M}$ is used to deduce boundary conditions on $\tilde{\Gamma}$ from the boundary conditions on Γ, via Taylor expansions. In this paper, we choose the orthogonal projection operator on Γ (note that Γ is a closed convex set, see Figure (3)).

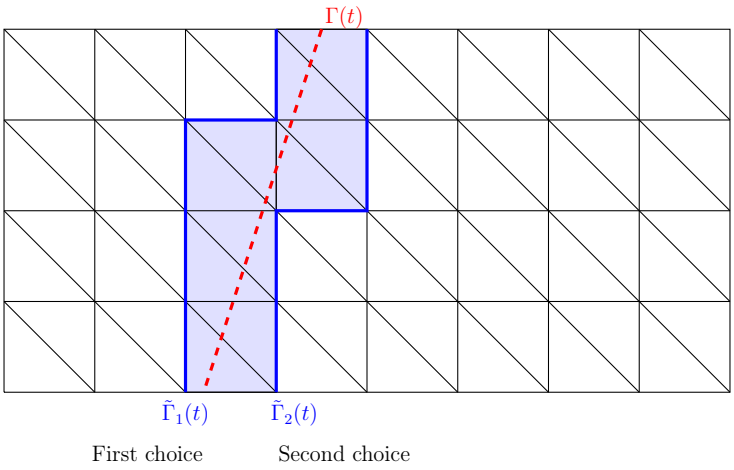


Figure 2: Surrogate definition

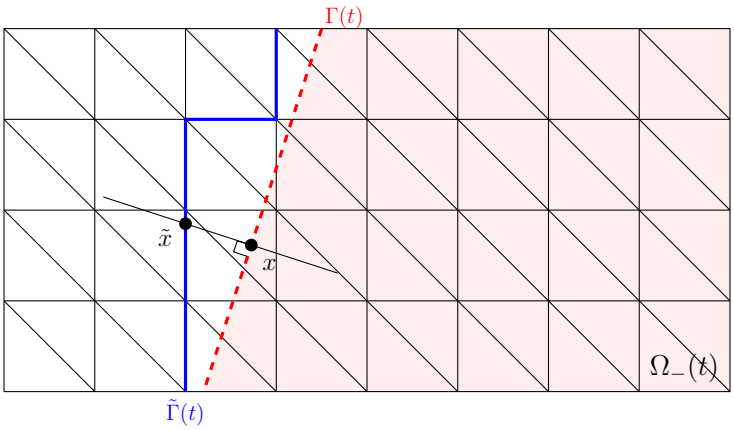


Figure 3: Definition of the mapping by orthogonal projection on Ω_-

We introduce the distance vector function, which measures the distance between Γ and $\tilde{\Gamma}$, as follows

$$\boldsymbol{d}(\tilde{\boldsymbol{x}}) = \boldsymbol{x} - \tilde{\boldsymbol{x}} = [\mathbf{M} - \mathbf{I}](\tilde{\boldsymbol{x}}). \tag{14}$$

Note that, by construction, $\boldsymbol{d}$ is aligned with the normal vector on Γ

$$\boldsymbol{d} = ||\boldsymbol{d}||\ \boldsymbol{n}, \tag{15}$$

where $||\boldsymbol{d}||$ denotes the euclidean norm in $\mathbb{R}^d$. Moreover, we assume that the normal $\boldsymbol{n}$ to Γ and the normal $\tilde{\boldsymbol{n}}$ to $\tilde{\Gamma}$ satisfy $\boldsymbol{n} \cdot \tilde{\boldsymbol{n}} > 0$. This condition describes that $\tilde{\boldsymbol{n}}$ is on the half plane identified by $\boldsymbol{n}$. Following [47], it can be interpreted as a geometric resolution condition. Note that if the details of the physical geometry have scales much finer than the computational grid then this condition can easily be not satisfied.

Owning these definitions, we are able to define any functions on Γ as a function on $\tilde{\Gamma}$ by doing an extension of the original function. Indeed, if ψ is a function initially define on Γ, then its extension $\tilde{\psi}$ on $\tilde{\Gamma}$ is given by

$$\tilde{\psi}(\tilde{\boldsymbol{x}}) = \psi(\mathbf{M}(\tilde{\boldsymbol{x}})). \tag{16}$$

3.1. Conformal case

In order to decouple the complexity of the problem addressed in this paper, we first consider the case where the mesh is grid conforming to the geometry. It is a good starting point

to understand the construction of the weak formulation (WF) and its properties, especially with the particularities brought by the presence of the moving interface. The construction and analysis of the proposed stabilized Finite Element scheme is based on the stabilized Continuous Galerkin (CG) and Discontinuous Galerkin (DG) schemes introduced by Hughes, Masud and co-authors (see e.g. [56, 57, 58]), and extended to the problem at stake in some recent articles, see for instance [59, 60, 61].

3.1.1. Notations

Let $\mathcal{D}(\Omega)$ (resp. $\mathcal{D}(\Omega)^d$) be the space of C^∞ (resp. $(C^\infty)^d$) functions with compact support in Ω and $D^{'}(\Omega)$ (resp. $D^{'}(\Omega)^d$) the dual spaces associated. As usual, we denote by $L^2(\Omega)$ the space of measurable and square-integrable functions from $\mathbb{R}^d$ into $\mathbb{R}$ endowed with the inner-product

$$(T, q)_\Omega = \int_\Omega Tq \;\; dx,$$

and by $(L^2(\Omega))^d$ the space of measurable and square-integrable function from $\mathbb{R}^d$ into $\mathbb{R}^d$ endowed with the inner product

$$(\boldsymbol{\beta}, \boldsymbol{w})_\Omega = \int_\Omega \boldsymbol{\beta} \cdot \boldsymbol{w} \;\; dx.$$

In a similar way, we introduce the space $L^2(\Gamma)$ and $(L^2(\Gamma))^d$ for functions defined on Γ, associated with the inner-products

$$< T, q >_\Gamma = \int_\Gamma Tq \;\; d\gamma \quad \text{and} \quad < \boldsymbol{\beta}, \boldsymbol{w} >_\Gamma = \int_\Gamma \boldsymbol{\beta} \cdot \boldsymbol{w} \;\; d\gamma.$$

We also introduce the classical Sobolev spaces $H^1(\Omega)$ and $H(\mathrm{div}, \Omega)$ such that

$$H^1(\Omega) = \left\{ T \in L^2(\Omega) \mid \nabla T \in (L^2(\Omega))^d \right\},$$

associated with the norm

$$\|T\|^2_{H^1(\Omega)} = \int_\Omega (|T|^2 + |\nabla T|^2) dx,$$

and

$$H(\mathrm{div}, \Omega) = \left\{ \boldsymbol{\beta} \in (L^2(\Omega))^d \mid \mathrm{div}(\boldsymbol{\beta}) \in L^2(\Omega) \right\},$$

associated with the norm

$$\|\boldsymbol{\beta}\|^2_{H(\mathrm{div},(\Omega))} = \int_\Omega (|\boldsymbol{\beta}|^2 + |\nabla \cdot \boldsymbol{\beta}|^2) dx.$$

We define similar spaces if we replace Ω by Ω_+ or Ω_-.
In order to formulate properly the variational form of System (10), we define

$$\begin{aligned} \mathcal{W}_T(\Omega) &= \{T \in H^1(\Omega_+) \cup H^1(\Omega_-)\}, \\ \mathcal{W}_\beta(\Omega) &= \{\boldsymbol{\beta} \in (H(\mathrm{div}, \Omega_+) \cup H(\mathrm{div}, \Omega_-))\}, \end{aligned} \tag{17}$$

endowed with the norms

$$\|T\|^2_{\mathcal{W}_T} = \|T\|^2_{H^1(\Omega_+)} + \|T\|^2_{H^1(\Omega_-)},$$

and

$$\|\boldsymbol{\beta}\|^2_{\mathcal{W}_\beta} = \|\boldsymbol{\beta}\|^2_{H(\mathrm{div},\Omega_+)} + \|\boldsymbol{\beta}\|^2_{H(\mathrm{div},\Omega_-)}.$$

For any time $t > 0$, $\mathbb{L}^2(]0, t[, \mathcal{W}_T(\Omega))$ is the set of measurable functions $T \;\; :]0, t[\times\Omega \longrightarrow \mathbb{R}$

such that

$$\int_0^t \|T(s)\|^2_{\mathcal{W}_T} ds < +\infty,$$

and $\mathbb{L}^2(\,]0,t[,\mathcal{W}_\beta)$ is the set of measurable functions $\boldsymbol{\beta}\,:]0,t[\times\Omega \longrightarrow \mathbb{R}$ such that

$$\int_0^t \|\boldsymbol{\beta}(s)\|^2_{\mathcal{W}_\beta} dt < +\infty.$$

Finally, we define

$$\boldsymbol{W}(]0,t[,\Omega) = \mathbb{L}^2(]0,t[,\mathcal{W}_T(\Omega)) \times \mathbb{L}^2(]0,t[,\mathcal{W}_B(\Omega)), \tag{18}$$

$$\boldsymbol{W}(\Omega) = \mathcal{W}_T(\Omega) \times \mathcal{W}_B(\Omega). \tag{19}$$

3.1.2. The weak formulation

The classical weak formulation is obtained by taking the L^2 scalar product of the first-two equations of System (10) with test functions. It can be formulated in the following way :

Find $(T,\boldsymbol{\beta}) \in \boldsymbol{W}(]0,t[,\Omega)$ s.t. $\forall\ (q,\boldsymbol{w}) \in \boldsymbol{W}(\Omega)$ one has

$$\begin{aligned} &20.a)\ (\lambda^{-1}\boldsymbol{\beta},\boldsymbol{w})_\Omega + (\nabla T,\boldsymbol{w})_\Omega = 0,\\ &20.b)\ (\rho c\partial_t T,q)_\Omega + (\nabla\cdot\boldsymbol{\beta},q)_\Omega = (f,q)_\Omega. \end{aligned} \tag{20}$$

Note that, for the sake of clarity, we do not use at this step time discretization on the variables. As a consequence, the formulation (20) is sometimes called a quasi-weak formulation. The latter formulation (20) being for infinite-dimensional spaces, we consider from now on its discrete spatial version.

Let $\mathcal{T}_h$ be a tesselation of the domain Ω, in which elements are denoted by K and the boundaries by ∂K. We denote by h the measure of an element, and N_v and N_e correspond respectively to the number of nodes and of elements of the mesh $\mathcal{T}_h$. We can then define the finite-dimensional subspaces of $\mathcal{W}_T(\Omega)$ and $\mathcal{W}_\beta(\Omega)$ by:

$$\begin{aligned} \mathcal{W}_T^l(\Omega) &= \{T\in\mathcal{W}_T(\Omega)\mid T|_K\in\mathbb{P}_l(K),\forall K\in\mathcal{T}_h\}\cap(C^0(\Omega_+)\cup C^0(\Omega_-)),\\ \mathcal{W}_\beta^p(\Omega) &= \{\boldsymbol{\beta}\in\mathcal{W}_\beta(\Omega)\mid \boldsymbol{\beta}|_K\in\mathbb{P}_p(K),\forall K\in\mathcal{T}_h\}\cap((C^0(\Omega_+)^d\cup(C^0(\Omega_-))^d), \end{aligned} \tag{21}$$

which corresponds to the sets of piece-wise polynomials of order l for the temperature, and of order p for the flux, continuous on each subdomain Ω_+ and Ω_-. In the following, as for the infinite dimensional spaces, we use the notation

$$\boldsymbol{W}^{l,p}(]0,t[,\Omega) = \mathbb{L}^2(]0,t[,\mathcal{W}_T^l(\Omega)) \times \mathbb{L}^2(]0,t[,\mathcal{W}_B^p(\Omega)), \tag{22}$$

$$\boldsymbol{W}^{l,p}(\Omega) = \mathcal{W}_T^l(\Omega) \times \mathcal{W}_B^p(\Omega). \tag{23}$$

Note that in this present work, we are interested in piece-wise linear approximations of the variables, *i.e.* $l=p=1$. The Galerkin weak formulation can be written as:

Find $(T,\boldsymbol{\beta}) \in \boldsymbol{W}^{l,p}(]0,t[,\Omega)$ s.t. $\forall\ (q,\boldsymbol{w}) \in \boldsymbol{W}^{l,p}(\Omega)$ one has

$$\begin{aligned} &24.a)\ (\lambda^{-1}\boldsymbol{\beta},\boldsymbol{w})_\Omega + (\nabla T,\boldsymbol{w})_\Omega = 0\ ,\\ &24.b)\ (\rho c\partial_t T,q)_\Omega + (\nabla\cdot\boldsymbol{\beta},q)_\Omega = (f,q)_\Omega\ . \end{aligned} \tag{24}$$

As example in Section 2.1, we consider two sets of interface conditions.
case 1: Jumps in flux and temperature are imposed at the interface, through (11). Using the

decomposition $\Omega = \Omega_+ \cup \Omega_-$ and Green's formula separately on Ω_+ and Ω_- on Equations (24), we obtain

$$\begin{aligned} &25.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T, \nabla.\boldsymbol{w})_\Omega + < T\ ,\ \boldsymbol{w}\cdot\boldsymbol{n} >_{\partial\Omega} + < [\![\boldsymbol{w}T]\!]_\Gamma, \boldsymbol{n} >_\Gamma = 0, \\ &25.b)\ (\rho c\partial_t T, q)_\Omega - (\boldsymbol{\beta}, \nabla q)_\Omega + < \boldsymbol{\beta}\cdot\boldsymbol{n}, q >_{\partial\Omega} + < [\![q\boldsymbol{\beta}]\!]_\Gamma, \boldsymbol{n} >_\Gamma = (f, q)_\Omega. \end{aligned} \tag{25}$$

The outer boundary $\partial\Omega$ is decomposed into Γ_D and Γ_N on which are respectively imposed Dirichlet and Neumann conditions, leading to

$$\begin{aligned} &26.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T, \nabla\cdot\boldsymbol{w})_\Omega + < T, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_N} + < T_D, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_D} + < [\![\boldsymbol{w}T]\!]_\Gamma, \boldsymbol{n} >_\Gamma = 0, \\ &26.b)\ (\rho c\partial_t T, q)_\Omega - (\boldsymbol{\beta}, \nabla q)_\Omega + < \boldsymbol{\beta}\cdot\boldsymbol{n}, q >_{\Gamma_D} + < h_N, q >_{\Gamma_N} + < [\![q\boldsymbol{\beta}]\!]_\Gamma, \boldsymbol{n} >_\Gamma = (f, q)_\Omega. \end{aligned} \tag{26}$$

Using the jump decomposition (4) for the term on Γ and using the interface conditions (11), we get

$$\begin{aligned} &27.a) < [\![\boldsymbol{w}T]\!]_\Gamma, \boldsymbol{n} >_\Gamma = < \{T\}_\Gamma, [\![\boldsymbol{w}]\!]_\Gamma\cdot\boldsymbol{n} >_\Gamma + < j_T(\boldsymbol{x}, t), \{\boldsymbol{w}\}_\Gamma\cdot\boldsymbol{n} >_\Gamma, \\ &27.b) < [\![q\boldsymbol{\beta}]\!]_\Gamma, \boldsymbol{n} >_\Gamma = < \{\boldsymbol{\beta}\}_\Gamma\cdot\boldsymbol{n}, [\![q]\!]_\Gamma >_\Gamma + < \sigma(\boldsymbol{x}, t), \{q\}_\Gamma >_\Gamma . \end{aligned} \tag{27}$$

Plugging (27) into (26), and performing again in (26.b) a Green's formula on Ω_+ and Ω_-, one can derive the final weak formulation.

Find $(T, \boldsymbol{\beta}) \in \boldsymbol{W}^{l,p}(]0, t[, \Omega)$ s.t. $\forall\ (q, \boldsymbol{w}) \in \boldsymbol{W}^{l,p}(\Omega)$ one has

$$\begin{aligned} &28.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T, \nabla\cdot\boldsymbol{w})_\Omega + < T, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_N} + < T_D, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_D} \\ &\qquad\qquad + < \{T\}_\Gamma, [\![\boldsymbol{w}]\!]_\Gamma\cdot\boldsymbol{n} >_\Gamma + < j_T, \{\boldsymbol{w}\}_\Gamma\cdot\boldsymbol{n} >_\Gamma = 0, \\ &28.b)\ (\rho c\partial_t T, q)_\Omega + (\nabla\cdot\boldsymbol{\beta}, q)_\Omega - < \boldsymbol{\beta}\cdot\boldsymbol{n} - h_N, q >_{\Gamma_N} - < [\![\boldsymbol{\beta}]\!]_\Gamma\cdot\boldsymbol{n} - \sigma, \{q\}_\Gamma >_\Gamma = (f, q)_\Omega. \end{aligned} \tag{28}$$

<u>*case 2:*</u> Dirichlet type conditions are imposed at the interface according to (9). In a similar way, performing again Green's formula separately on Ω_+ and Ω_- for (24.a) and on Ω for (24.b), we get

$$\begin{aligned} &29.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T, \nabla\cdot\boldsymbol{w})_\Omega + < T, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_N} + < T_D, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_D} + < [\![\boldsymbol{w}T]\!]_\Gamma, \boldsymbol{n} >_\Gamma = 0, \\ &29.b)\ (\rho c\partial_t T, q)_\Omega - (\boldsymbol{\beta}, \nabla q)_\Omega + < \boldsymbol{\beta}\cdot\boldsymbol{n}, q >_{\Gamma_D} + < h_N, q >_{\Gamma_N} = (f, q)_\Omega. \end{aligned} \tag{29}$$

Using again relation (4) with interface conditions (9), we obtain

$$< [\![\boldsymbol{w}T]\!]_\Gamma, \boldsymbol{n} >_\Gamma = < j_T, \{\boldsymbol{w}\}_\Gamma\cdot\boldsymbol{n} >_\Gamma + < a_T, [\![\boldsymbol{w}]\!]_\Gamma\cdot\boldsymbol{n} >_\Gamma . \tag{30}$$

Plugging (30) into (29.a) and performing again Green's formula on Ω in (29.b), we get the following weak formulation: Find $(T, \boldsymbol{\beta}) \in \boldsymbol{W}^{l,p}(]0, t[, \Omega)$ s.t. $\forall\ (q, \boldsymbol{w}) \in \boldsymbol{W}^{l,p}(\Omega)$ one has

$$\begin{aligned} &31.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T, \nabla\cdot\boldsymbol{w})_\Omega + < T, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_N} + < T_D, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_D} \\ &\qquad\qquad + < a_T, [\![\boldsymbol{w}]\!]_\Gamma\cdot\boldsymbol{n} >_\Gamma + < j_T, \{\boldsymbol{w}\}_\Gamma\cdot\boldsymbol{n} >_\Gamma = 0, \\ &31.b)\ (\rho c\partial_t T, q)_\Omega + (\nabla\cdot\boldsymbol{\beta}, q)_\Omega - < \boldsymbol{\beta}\cdot\boldsymbol{n} - h_N, q >_{\Gamma_N} = (f, q)_\Omega. \end{aligned} \tag{31}$$

Any solution $(T,\boldsymbol{\beta})$ of (28) or (31) with $T \in \mathbb{L}^2(]0, t[, \mathcal{W}_T(\Omega))$, $\boldsymbol{\beta} \in \mathbb{L}^2(]0, t[, \mathcal{W}_\beta(\Omega))$, $\partial_t T \in \mathbb{L}^2(]0, t[, \mathbb{H}^{-1}(\Omega))$ is called a weak solution of System (10,11) or (10,9).

3.1.3. Stabilization

The previous formulations (28) and (31) are known to be stable for only certain choices of polynomial order of approximation. Following [56, 57, 58, 59, 60] we add two different kinds of stabilization terms in the previous formulations. The first one is called the **div-div stabilization** and is equal to

$$\frac{\zeta_{div}h^2}{2\lambda}(\rho c\partial_t T + \nabla\cdot\boldsymbol{\beta} - f\ ,\ \nabla\cdot\boldsymbol{w})_\Omega,$$

where $\zeta_{div} = \mathcal{O}(1)$. This term counterbalances the error induced by the discretization of the divergence operator. The other stabilization term is the so-called **momentum stabilization** and is equal to

$$-1/2(\boldsymbol{\beta} + \lambda\nabla T\ ,\ \lambda^{-1}\boldsymbol{w} - \nabla q)_\Omega.$$

For convenience, we introduce the following terms

$$\begin{aligned}
A_{stab}((T,\boldsymbol{\beta}),q) &:= \frac{1}{2}\ (\boldsymbol{\beta} + \lambda\nabla T\ ,\ \nabla q)_\Omega,\\
B_{stab}((T,\boldsymbol{\beta}),\boldsymbol{w}) &:= \frac{\zeta_{div}h^2}{2\lambda}(\rho c\partial_t T + \nabla\cdot\boldsymbol{\beta}\ ,\ \nabla\cdot\boldsymbol{w})_\Omega - 1/2\ (\boldsymbol{\beta} + \lambda\nabla T\ ,\ \lambda^{-1}\boldsymbol{w})_\Omega,\\
L_{stab}(\boldsymbol{w}) &:= \frac{\zeta_{div}h^2}{2\lambda}(f\ ,\ \nabla\cdot\boldsymbol{w})_\Omega.
\end{aligned} \tag{32}$$

Plugging the different stabilization terms into (28) and (31), one obtains the following new formulations: Find $(T,\boldsymbol{\beta}) \in \boldsymbol{W}^{l,p}(]0,t[,\Omega)$ s.t. $\forall\ (q,\boldsymbol{w}) \in \boldsymbol{W}^{l,p}(\Omega)$

$$\begin{aligned}
&33.a)\ (\lambda^{-1}\boldsymbol{\beta},\boldsymbol{w})_\Omega - (T,\nabla\cdot\boldsymbol{w})_\Omega + <\{T\}_\Gamma, [\![\boldsymbol{w}]\!]_\Gamma\cdot\boldsymbol{n}>_\Gamma + <T,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_N}\\
&\qquad +\ B_{stab}((T,\boldsymbol{\beta}),\boldsymbol{w}) =\ L_{stab}(\boldsymbol{w}) - <T_D,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_D} - <j_T,\{\boldsymbol{w}\}_\Gamma\cdot\boldsymbol{n}>_\Gamma,\\
&33.b)\ (\rho c\partial_t T,q)_\Omega + (\nabla\cdot\boldsymbol{\beta},q)_\Omega - <\boldsymbol{\beta}\cdot\boldsymbol{n},q>_{\Gamma_N} - <[\![\boldsymbol{\beta}]\!]_\Gamma\cdot\boldsymbol{n},\{q\}>_\Gamma\\
&\qquad +\ A_{stab}((T,\boldsymbol{\beta}),q) = (f,q)_\Omega - <h_N,q>_{\Gamma_N} - <\sigma,\{q\}>_\Gamma,
\end{aligned} \tag{33}$$

and

Find $(T,\boldsymbol{\beta}) \in \boldsymbol{W}^{l,p}(]0,t[,\Omega)$ s.t. $\forall\ (q,\boldsymbol{w}) \in \boldsymbol{W}^{l,p}(\Omega)$

$$\begin{aligned}
&34.a)\ (\lambda^{-1}\boldsymbol{\beta},\boldsymbol{w})_\Omega - (T,\nabla\cdot\boldsymbol{w})_\Omega + <T,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_N} + <T_D,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_D} + B_{stab}((T,\boldsymbol{\beta}),\boldsymbol{w})\\
&\qquad + <a_T, [\![\boldsymbol{w}]\!]_\Gamma\cdot\boldsymbol{n}>_\Gamma + <j_T,\{\boldsymbol{w}\}_\Gamma\cdot\boldsymbol{n}>_\Gamma = L_{stab}(\boldsymbol{w}),\\
&34.b)\ (\rho c\partial_t T,q)_\Omega + (\nabla\cdot\boldsymbol{\beta},q)_\Omega - <\boldsymbol{\beta}\cdot\boldsymbol{n} - h_N,q>_{\Gamma_N} + A_{stab}((T,\boldsymbol{\beta}),q) = (f,q)_\Omega.
\end{aligned} \tag{34}$$

Remark 1. The stabilization terms are independent of the use of the SBM.

3.1.4. Nitsche penalty term

In order to enforce the Dirichlet condition on Γ_D (see [62]) and the temperature jump on Γ (see [34]), we add Nitsche penalty terms which are the consistent terms $<\frac{\alpha}{h}(T - T_D), q>_{\Gamma_D}$ and $<\frac{\alpha}{h}([\![T]\!]_\Gamma - j_T), [\![q]\!]_\Gamma>_\Gamma$ to Equations (33) and (34). In addition, it has been showed in [57, 59, 60] that the jump penalty terms provide the theoretical stability of the WF, using coercivity arguments. For the second set of conditions $<\frac{\alpha}{h}(\{T\}_\Gamma - a_T), \{q\}_\Gamma>_\Gamma$ is also added to impose the temperature at the interface as a Dirichlet condition.

To lighten the definition of the weak formulation we introduce the following terms

$$\begin{aligned} N_{A_1}(T,q) &:= \frac{\alpha}{h} < [\![T]\!]_\Gamma, [\![q]\!]_\Gamma >_\Gamma, \\ N_{L_1}(q) &:= \frac{\alpha}{h} < j_T, [\![q]\!]_\Gamma >_\Gamma, \end{aligned} \tag{35}$$

and

$$\begin{aligned} N_{A_2}(T,q) &:= \frac{\alpha}{h} < [\![T]\!]_\Gamma, [\![q]\!]_\Gamma >_\Gamma + \frac{\alpha}{h} < \{T\}_\Gamma, \{q\}_\Gamma >_\Gamma, \\ N_{L_2}(q) &:= \frac{\alpha}{h} < j_T, [\![q]\!]_\Gamma >_\Gamma + \frac{\alpha}{h} < a_T, \{q\}_\Gamma >_\Gamma . \end{aligned} \tag{36}$$

The terms in (35) are added for case 1, while those in (36) are added for case 2. The Nitsche terms are separated in two distinct terms, one part is added to the bilinear formulation (N_{A_1} or N_{A_2}), while the other is added to the linear form of the weak formulation (N_{L_1} or N_{L_2}).

Remark 2. In [57, 58, 59, 60], the questions of stability and convergence for the steady version of (10) are addressed. If $(T, \boldsymbol{\beta}) \in \boldsymbol{W}^{l,p}(]0,t[,\Omega)$, the scheme provides approximations of order $r := \min(p+1, l)$ for the flux and of order $r+1$ for the temperature. As a consequence, dealing with piecewise linear approximations furnishes respectively a first order accuracy for the flux and a second order accuracy for the temperature.

3.2. Embedded case

In this section, we consider the situation described in Section 3 which includes the Shifted Boundary Method. Our aim is to provide the different steps leading to the embedded version, proposed in [47] and referred as the Shifted Interface Method. We first rewrite Equations (6) and (9) using Taylor expansions between Γ and $\tilde{\Gamma}$

$$\begin{aligned} &37.a)\ [\![T]\!]_\Gamma &&= j_T &&= [\![T + \nabla T \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}} &&+ \mathcal{O}(||\boldsymbol{d}(\tilde{\boldsymbol{x}})||^2), \\ &37.b)\ [\![\boldsymbol{\beta}]\!]_\Gamma \cdot \boldsymbol{n} &&= \sigma &&= [\![\boldsymbol{\beta} + \nabla \boldsymbol{\beta}\ \boldsymbol{d}]\!]_{\tilde{\Gamma}} \cdot \boldsymbol{n} &&+ \mathcal{O}(||\boldsymbol{d}(\tilde{\boldsymbol{x}})||^2), \\ &37.c)\ \{T\}_\Gamma &&= a_T &&= \{T + \nabla T\ \cdot \boldsymbol{d}\}_{\tilde{\Gamma}} &&+ \mathcal{O}(||\boldsymbol{d}(\tilde{\boldsymbol{x}})||^2), \end{aligned} \tag{37}$$

where $\boldsymbol{d}$ is the distance vector (see 14).
We also decompose the normal $\tilde{\boldsymbol{n}}$ on the basis $(\boldsymbol{\tau}, \boldsymbol{n})$

$$\tilde{\boldsymbol{n}} = (\tilde{\boldsymbol{n}} \cdot \boldsymbol{n})\boldsymbol{n} + (\tilde{\boldsymbol{n}} \cdot \boldsymbol{\tau})\boldsymbol{\tau} \tag{38}$$

Since $\tilde{\Gamma}(t)$ is moving, the domains $\tilde{\Omega}_+$ and $\tilde{\Omega}_-$ separated by the interface $\tilde{\Gamma}$ depend now on time t. Introduce the space

$$\begin{aligned} \mathcal{Q}_T(\Omega) &= \{T \in H^1(\tilde{\Omega}_+) \cup H^1(\tilde{\Omega}_-)\}, \\ \mathcal{Q}_\beta(\Omega) &= \{\boldsymbol{\beta} \in (H(\mathrm{div}, \tilde{\Omega}_+) \cup H(\mathrm{div}, \tilde{\Omega}_-))\}, \end{aligned} \tag{39}$$

and their finite-dimensional versions:

$$\begin{aligned} \mathcal{Q}_T^l(\Omega) &= \{T \in \mathcal{Q}_T(\Omega)\,|\,T|_K \in \mathbb{P}_l(K),\ \forall K \in \mathcal{T}_h\} \cap (C^0(\tilde{\Omega}^+) \cup C^0(\tilde{\Omega}^-)) \\ \mathcal{Q}_\beta^p(\Omega) &= \{\boldsymbol{\beta} \in \mathcal{Q}_\beta(\Omega)\,|\,\boldsymbol{\beta}|_K \in (\mathbb{P}_p(K))^d,\ \forall K \in \mathcal{T}_h\} \cap ((C^0(\tilde{\Omega}^+))^d \cup (C^0(\tilde{\Omega}^-))^d) \end{aligned} \tag{40}$$

Finally, we consider

$$\boldsymbol{Q}^{l,p}(]0,t[,\Omega) = \mathbb{L}^2(]0,t[, \mathcal{Q}_T^l(\Omega)) \times \mathbb{L}^2(]0,t[, \mathcal{Q}_B^p(\Omega)) \tag{41}$$

$$\boldsymbol{Q}^{l,p}(\Omega) = \mathcal{Q}_T^l(\Omega) \times \mathcal{Q}_B^p(\Omega) \tag{42}$$

We now derive the weak formulations corresponding to the embedded case in both situations described in Section (3.1.2).

Case 1. Repeating the arguments of Section (3.1.2), that is applying Green's formula on (24.a, 24.b), recalling that $\Omega = \tilde{\Omega}_+ \cup \tilde{\Omega}_-$, one obtains

$$\begin{aligned} &43.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T, \nabla\cdot\boldsymbol{w})_\Omega + < T\ ,\ \boldsymbol{w}\cdot\boldsymbol{n} >_{\partial\Omega} + < [\![\boldsymbol{w}T]\!]_{\tilde{\Gamma}}, \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} = 0, \\ &43.b)\ (\rho c\partial_t T, q)_\Omega - (\boldsymbol{\beta}, \nabla q)_\Omega + < \boldsymbol{\beta}\cdot\boldsymbol{n}, q >_{\partial\Omega} + < [\![q\boldsymbol{\beta}]\!]_{\tilde{\Gamma}}, \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} = (f, q)_\Omega. \end{aligned} \tag{43}$$

The jump term in (43.a) is handled combining (4) and (37.a), neglecting the terms of order 2 with respect to $\boldsymbol{d}$

$$< [\![\boldsymbol{w}T]\!]_{\tilde{\Gamma}}, \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} = < \{T\}_{\tilde{\Gamma}}, [\![\boldsymbol{w}]\!]_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} + < j_T - [\![\nabla T\cdot\boldsymbol{d}]\!]_{\tilde{\Gamma}}, \{\boldsymbol{w}\}_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}}\ . \tag{44}$$

The treatment of the jump term in (43.b) is more delicate and requires the normal decomposition (38). Indeed, the extrapolation of the flux is known in the normal direction of the physical interface $\boldsymbol{n}$. Thus, when applying the jump decomposition (4), one needs to project the surrogate normal $\tilde{\boldsymbol{n}}$ onto $\boldsymbol{n}$ and $\boldsymbol{\tau}$ before using the truncated extrapolation (37.b), providing:

$$\begin{aligned} < [\![q\boldsymbol{\beta}]\!]_\Gamma, \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} = &< \{\boldsymbol{\beta}\}_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}}, [\![q]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}} + < [\![\boldsymbol{\beta}]\!]_{\tilde{\Gamma}}\cdot\boldsymbol{\tau}(\boldsymbol{\tau}\cdot\tilde{\boldsymbol{n}}), \{q\}_{\tilde{\Gamma}} >_{\tilde{\Gamma}} \\ &+ < (\sigma - [\![\nabla\boldsymbol{\beta}\ \boldsymbol{d}]\!]_{\tilde{\Gamma}}\cdot\boldsymbol{n})(\boldsymbol{n}\cdot\tilde{\boldsymbol{n}}), \{q\}_{\tilde{\Gamma}} >_{\tilde{\Gamma}}\ . \end{aligned} \tag{45}$$

Plugging (44,45) into (43), we thus obtain

$$\begin{aligned} &46.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T, \nabla\cdot\boldsymbol{w})_\Omega + < T\ ,\ \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_N} + < T_D, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_D} \\ &\qquad + < \{T\}_{\tilde{\Gamma}}, [\![\boldsymbol{w}]\!]_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} + < j_T - [\![\nabla T\cdot\boldsymbol{d}]\!]_{\tilde{\Gamma}}, \{\boldsymbol{w}\}\cdot\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} = 0, \\ &46.b)\ (\rho c\partial_t T, q)_\Omega - (\boldsymbol{\beta}, \nabla q)_\Omega + < \boldsymbol{\beta}\cdot\boldsymbol{n}, q >_{\Gamma_D} + < h_N, q >_{\Gamma_N} + < \{\boldsymbol{\beta}\}_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}}, [\![q]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}} \\ &\quad + < [\![\boldsymbol{\beta}]\!]_{\tilde{\Gamma}}\cdot\boldsymbol{\tau}(\boldsymbol{\tau}\cdot\tilde{\boldsymbol{n}}), \{q\}_{\tilde{\Gamma}} >_{\tilde{\Gamma}} + < (\sigma - [\![\nabla\boldsymbol{\beta}\ \boldsymbol{d}]\!]_{\tilde{\Gamma}}\cdot\boldsymbol{n})(\boldsymbol{n}\cdot\tilde{\boldsymbol{n}}), \{q\}_{\tilde{\Gamma}} >_{\tilde{\Gamma}} = (f, q)_\Omega. \end{aligned} \tag{46}$$

Now, performing again a Green's formula in (46.b), using the decomposition $\Omega = \tilde{\Omega}_+ \cup \tilde{\Omega}_-$, we obtain, after a straightforward calculation, the following weak formulation:

Find $(T, \boldsymbol{\beta}) \in \boldsymbol{Q}^{1,1}(]0,t[, \Omega)$ s.t. $\forall (q, \boldsymbol{w}) \in \boldsymbol{Q}^{1,1}(\Omega)$ one has

$$\begin{aligned} &47.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T, \nabla\cdot\boldsymbol{w})_\Omega + < T\ ,\ \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_N} + < T_D, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_D} \\ &\qquad + < \{T\}_{\tilde{\Gamma}}, [\![\boldsymbol{w}]\!]_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} + < j_T - [\![\nabla T\cdot\boldsymbol{d}]\!]_{\tilde{\Gamma}}, \{\boldsymbol{w}\}\cdot\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} = 0, \\ &47.b)\ (\rho c\partial_t T, q)_\Omega + (\nabla\cdot\boldsymbol{\beta}, q)_\Omega - < \boldsymbol{\beta}\cdot\boldsymbol{n} - h_N, q >_{\Gamma_N} + < \{\boldsymbol{\beta}\}_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}}, [\![q]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}} \\ &\qquad + < (\sigma - [\![\boldsymbol{\beta} + \nabla\boldsymbol{\beta}\ \boldsymbol{d}]\!]_{\tilde{\Gamma}}\cdot\boldsymbol{n})(\boldsymbol{n}\cdot\tilde{\boldsymbol{n}}), \{q\}_{\tilde{\Gamma}} >_{\tilde{\Gamma}} = (f, q)_\Omega. \end{aligned} \tag{47}$$

Notice that, for the sake of readability, we do not add the stabilization terms $A_{stab}((T, \boldsymbol{\beta}), q)$, $B_{stab}((T, \boldsymbol{\beta}), \boldsymbol{w})$ and $L_{stab}(\boldsymbol{w})$, neither the Nitsche penalty terms in (47). In addition, the Taylor expansion needs to be performed in the Nitsche penalty terms. And, as presented in [45, 55], the extrapolation can also be applied on the test function to provide symmetric terms:

$$\begin{aligned} \bar{N}_{A_1}(T, q) &:= \frac{\alpha}{h} < [\![T + \nabla T\cdot\boldsymbol{d}]\!]_{\tilde{\Gamma}}, [\![q + \nabla q\cdot\boldsymbol{d}]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}}, \\ \bar{N}_{L_1}(q) &:= \frac{\alpha}{h} < j_T, [\![q + \nabla q\cdot\boldsymbol{d}]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}}\ . \end{aligned} \tag{48}$$

Case 2.

For the second set of interface condition (9), where the temperature is imposed on each side of the interface, we only need to modify the momentum equation (24.a). Using Green's formula,

Equation (24.a) becomes

$$(\lambda^{-1}\boldsymbol{\beta},\boldsymbol{w})_\Omega-(T,\nabla\cdot\boldsymbol{w})_\Omega+<T_D,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_D}+<T,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_N}+<[\![\boldsymbol{w}T]\!]_{\tilde{\Gamma}},\tilde{\boldsymbol{n}}>_{\tilde{\Gamma}}=0. \quad (49)$$

Combine (4), (37.a) and (37.c), one obtains, at leading order,

$$<[\![T\boldsymbol{w}]\!]_{\tilde{\Gamma}},\tilde{\boldsymbol{n}}>_{\tilde{\Gamma}}=<a_T-\{\nabla T\cdot\boldsymbol{d}\}_{\tilde{\Gamma}},[\![\boldsymbol{w}]\!]_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}}>_{\tilde{\Gamma}}+<j_T-[\![\nabla T\cdot\boldsymbol{d}]\!]_{\tilde{\Gamma}},\{\boldsymbol{w}\}_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}}>_{\Gamma}. \quad (50)$$

As mentionned above, nothing has to be done on (24.b) regardless to the interface $\tilde{\Gamma}$. Thus, by plugging (50) into (49), we directly get the following weak formulation:
Find $(T,\boldsymbol{\beta})\in\boldsymbol{Q}^{1,1}(]0,t[,\Omega)$ s.t. $\forall(q,\boldsymbol{w})\in\boldsymbol{Q}^{1,1}(\Omega)$ we have

$$\begin{aligned}
&A.1.a)\ (\lambda^{-1}\boldsymbol{\beta},\boldsymbol{w})_\Omega-(T,\nabla\cdot\boldsymbol{w})_\Omega+<T_D,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_D}+<T,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_N}\\
&\qquad+<a_T-\{\nabla T\cdot\boldsymbol{d}\}_{\tilde{\Gamma}},[\![\boldsymbol{w}]\!]_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}}>_{\tilde{\Gamma}}+<j_T-[\![\nabla T\cdot\boldsymbol{d}]\!]_{\tilde{\Gamma}},\{\boldsymbol{w}\}_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}}>_{\tilde{\Gamma}}=0, \quad (51)\\
&A.1.b)\ (\rho c\partial_t T,q)_\Omega+(\nabla\cdot\boldsymbol{\beta},q)_\Omega-<\boldsymbol{\beta}\cdot\boldsymbol{n}-h_N,q>_{\Gamma_N}=(f,q)_\Omega.
\end{aligned}$$

Again, this last formulation has to be stabilized with the terms $A_{stab}((T,\boldsymbol{\beta}),q)$, $B_{stab}((T,\boldsymbol{\beta}),\boldsymbol{w})$, $L_{stab}(\boldsymbol{w})$, and the extrapolated Nitsche penalty terms $\bar{N}_{A_2}(T,q)$ and $\bar{N}_{L_2}(q)$:

$$\begin{aligned}
\bar{N}_{A_2}(T,q)&:=\frac{\alpha}{h}<[\![T+\nabla T\cdot\boldsymbol{d}]\!]_{\tilde{\Gamma}},[\![q+\nabla q\cdot\boldsymbol{d}]\!]_{\tilde{\Gamma}}>_{\tilde{\Gamma}}+\frac{\alpha}{h}<\{T+\nabla T\cdot\boldsymbol{d}\}_{\tilde{\Gamma}},\{q+\nabla q\cdot\boldsymbol{d}\}_{\tilde{\Gamma}}>_{\tilde{\Gamma}},\\
\bar{N}_{L_2}(q)&:=\frac{\alpha}{h}<j_T,[\![q+\nabla q\cdot\boldsymbol{d}]\!]_{\tilde{\Gamma}}>_{\tilde{\Gamma}}+\frac{\alpha}{h}<a_T,\{q+\nabla q\cdot\boldsymbol{d}\}_{\tilde{\Gamma}}>_{\tilde{\Gamma}}.
\end{aligned} \quad (52)$$

3.3. High order gradients - An enriched embedded scheme

The use of the flux as an additional variable, at least locally, is mandatory to preserve high order accuracy on the temperature, when gradient conditions (or flux conditions) are shifted, such as Neumann BC. Indeed, even if some gradient reconstruction could be considered, as originally reported in [45], the dependency on the reconstruction tends to provide a non robust scheme. In our situation, to preserve a second order approximation on the temperature as in the conformal case, it is necessary to obtain a high order flux. When dealing with Neumann conditions, as numerically studied in [55], the temperature will be at most of the same order than the flux. In [50], the authors genuinely employed a mixed formulation for solid mechanics only close to the interface, to get a local resolution of the gradient and recover the willing accuracy. In the context of the ϕ-FEM method [41, 42], sharing similarities with the SBM in the scheme conception, a similar requirement arises when dealing with Neumann condition [42], where an auxiliary variable mimicking the gradient is also locally employed.

For the problem at stake here, as the interface condition (11) imposes a flux jump that needs to be shifted, a similar issue arises. In the context of the Shifted Interface Method [47], the author obtained a scheme of order 1.5 for the flux and of order 2 for the temperature. However, when dealing with the Stefan condition (12), if the interface position wants to be numerically solved with high order accuracy, as will be described later in Section 4, the flux needs to be of high order accurate. In our case, to provide a fully second order scheme for the temperature, the heat flux and the interface location, we thus need to ensure a second order flux in the embedded case. To this purpose, we follow the strategy proposed in [55], and employ an enrichment of the temperature.

The next step in our process is to define a method with a higher order of accuracy by extending the usual SBM (see [55]). It consists in improving the weak formulation to be at least second order accurate for both variables using the link between the test functions $\mathbb{P}^1$ and $\mathbb{P}^2$, and by

a proper utilization of Taylor expansions for both variables.
More precisely the key idea is to define a quadratic polynomial temperature in each element of the mesh by adding new degrees of freedom (DoF) associated to the value of each edge midpoint.

Let us denote T_h the approximation of the temperature in the finite elements $\mathbb{P}_2$ basis. In a classical way, we would decompose T_h as follows

$$T_h(x) = \sum_{i,node} T_i \phi_i^{\mathbb{P}^2}(x) \; + \; \sum_{j,edge} T_j \phi_j^{\mathbb{P}^2}(x) \quad . \tag{53}$$

Our point here is to avoid the construction of $\mathbb{P}_2$ elements by taking advantage of the following relations between test functions of a $\mathbb{P}^2$ element ($\phi_i^{\mathbb{P}^2}$) and test functions of a $\mathbb{P}^1$ element ($\phi_i^{\mathbb{P}^1}$).

$$\begin{aligned} \phi_i^{\mathbb{P}^2}(x) &= \phi_i^{\mathbb{P}^1}(x)(2\phi_i^{\mathbb{P}^1}(x) - 1) \quad \text{, if } \boldsymbol{i} \text{ a vertex} \\ \phi_k^{\mathbb{P}^2}(x) &= 4\phi_i^{\mathbb{P}^1}(x)\phi_j^{\mathbb{P}^1}(x) \quad \text{, if } \boldsymbol{k} \text{ is the midpoint of the edge [i,j]} \quad . \end{aligned} \tag{54}$$

In Figure 4 the relations (54) can be expressed by looking at the reference triangle. The red DoF correspond to the nodes of a $\mathbb{P}^1$ finite element while the green DoF are those added to transform a $\mathbb{P}^1$ finite element into a $\mathbb{P}^2$ one.

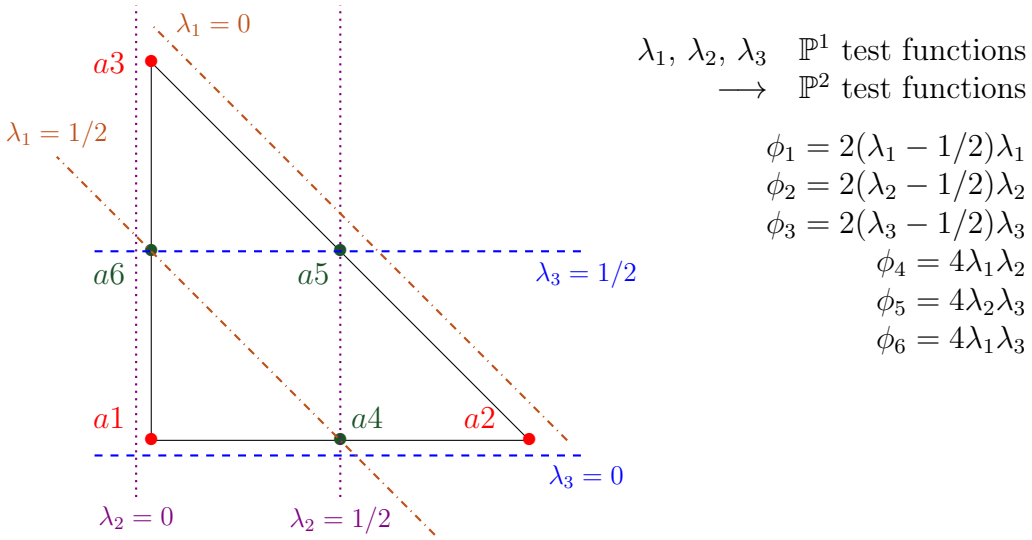


Figure 4: Visualisation of the link between the $\mathbb{P}^1$ and $\mathbb{P}^2$ test functions on the reference triangle.

Naturally, the $\mathbb{P}_2$ basis can be expressed in terms of the $\mathbb{P}_1$ basis. However, since we do not want to increase the complexity of our scheme by adding new unknowns (green DoF), we propose to replace these new values using only the information known at the initial nodes (red DoF in Figure 4). To this end, we consider a third-order truncated Taylor expansion for the temperature along the edge, and a second order Taylor expansion for the flux i.e.

$$\begin{aligned} &55.a)\ T(\boldsymbol{x}+\boldsymbol{\delta x}) = T(\boldsymbol{x}) + \nabla T(\boldsymbol{x})\cdot\boldsymbol{\delta x} + 1/2\ \boldsymbol{\delta x}^t \mathcal{H}(T)\boldsymbol{\delta x} + \mathcal{O}(||\boldsymbol{\delta x}||^3), \\ &55.b)\ \boldsymbol{\beta}(\boldsymbol{x}+\boldsymbol{\delta x}) = \boldsymbol{\beta}(\boldsymbol{x}) + \nabla\boldsymbol{\beta}(\boldsymbol{x})\boldsymbol{\delta x} + \mathcal{O}(||\boldsymbol{\delta x}||^2). \end{aligned} \tag{55}$$

The idea is to modify Expansions (55) to use the particularity of the mixed formulation where the expression of the flux is linked to the temperature. Taking advantage of dealing with the mixed formulation, that is the flux is linked to the temperature, one can rewrite (55) in the

following way

$$\begin{aligned} &56.a)\ T(\boldsymbol{x}+\boldsymbol{\delta x}) = T(\boldsymbol{x}) - \lambda^{-1}\boldsymbol{\beta}\cdot\boldsymbol{\delta x} - 1/2\ \boldsymbol{\delta x}^t\nabla(\lambda^{-1}\boldsymbol{\beta})\boldsymbol{\delta x} + \mathcal{O}(||\boldsymbol{\delta x}||^3), \\ &56.b)\ (\lambda^{-1}\boldsymbol{\beta})(\boldsymbol{x}+\boldsymbol{\delta x}) = (\lambda^{-1}\boldsymbol{\beta})(\boldsymbol{x}) + \nabla(\lambda^{-1}\boldsymbol{\beta})(\boldsymbol{x})\boldsymbol{\delta x} + \mathcal{O}(||\boldsymbol{\delta x}||^2). \end{aligned} \tag{56}$$

Let us denote $e_{i,j} = x_j - x_i$ the edge vector, then the Taylor expansion of $\lambda^{-1}\boldsymbol{\beta}$ leads to

$$\lambda_j^{-1}\boldsymbol{\beta}_j - \lambda_i^{-1}\boldsymbol{\beta}_i = \nabla(\lambda^{-1}\boldsymbol{\beta})(\boldsymbol{x})\boldsymbol{e}_{i,j} + \mathcal{O}(||\boldsymbol{\delta x}||^2). \tag{57}$$

Let i and j be two nodes of a same edge of the original mesh and denote by k the middle of the edge (see Figure 5). The temperature T_K is obtained by taking the mean value of the expressions (56), written successively between i and k and j and k. Using (57), we obtain, at leading order,

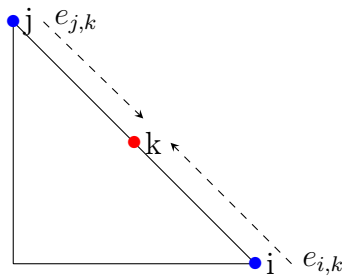


Figure 5: Approximation at a point k

$$\begin{aligned} T_k &= \frac{1}{2}\ (T_i + T_j - \lambda_i^{-1}\boldsymbol{\beta}_i\cdot\boldsymbol{e}_{i,k} - \lambda_j^{-1}\boldsymbol{\beta}_j\cdot\boldsymbol{e}_{j,k} - \frac{1}{2}(\lambda_k^{-1}\boldsymbol{\beta}_k - \lambda_j^{-1}\boldsymbol{\beta}_j)\cdot\boldsymbol{e}_{j,k} \\ &\quad - \frac{1}{2}\ (\lambda_k^{-1}\boldsymbol{\beta}_k - \lambda_i^{-1}\boldsymbol{\beta}_i)\cdot\boldsymbol{e}_{i,k}), \\ &= \frac{1}{2}\ (T_i + T_j) - \frac{1}{4}(\lambda_i^{-1}\boldsymbol{\beta}_i\cdot\boldsymbol{e}_{i,k} + \lambda_j^{-1}\boldsymbol{\beta}_j\cdot\boldsymbol{e}_{j,k}) \\ &= \frac{1}{2}\ (T_i + T_j) - \frac{1}{8}\ (\lambda_i^{-1}\boldsymbol{\beta}_i - \lambda_j^{-1}\boldsymbol{\beta}_j)\cdot e_{i,j}, \end{aligned} \tag{58}$$

since, by convention, one has

$$e_{i,j} = 2e_{i,k} = -2e_{j,k}.$$

Note that the last expression in (58) is independent of k. The last step consists in combining (53), (54) and (58) to obtain

$$\begin{aligned} T_h(x) &= \sum_{i,node} T_i\phi_i^{\mathbb{P}^2}(x) + \sum_{j,edge} T_j\phi_j^{\mathbb{P}^2}(x) \\ &= \sum_{i,node} T_i\phi_i^{\mathbb{P}^1}(x)(2\phi_i^{\mathbb{P}^1}(x) - 1) + \sum_{k,edge}(1/2\ (T_i + T_j) + 1/8\ (\lambda_j^{-1}\boldsymbol{\beta}_j - \lambda_i^{-1}\boldsymbol{\beta}_i)\cdot\boldsymbol{e}_{i,j})4\phi_i^{\mathbb{P}^1}(x)\phi_j^{\mathbb{P}^1}(x) \\ &= \sum_{i,node} T_i\phi_i^{\mathbb{P}^1}(x) + \sum_{k,edge} 1/2\ \Delta(\lambda^{-1}\boldsymbol{\beta})_e\cdot\Delta x_e\phi_i^{\mathbb{P}^1}(x)\phi_j^{\mathbb{P}^1}(x)\quad . \end{aligned} \tag{59}$$

Thus the enriched temperature is built using $\mathbb{P}^1$ test functions and without having to calculate the midpoint values. The terminology enrichment comes from definition (59) where we only have to add new terms to the usual $\mathbb{P}^1$ approximation.

To conclude this part, the enriched version consists in replacing T by T^* and the test function q by q^* in (47), where T^* and q^* are given by (59). This provides the following enriched weak version of Formulation (47).

Finding $(T^*, \boldsymbol{\beta}) \in \boldsymbol{Q}^{2,1}(\,]0,t[\,,\Omega)$, s.t. $\forall (q^*, \boldsymbol{w}) \in \boldsymbol{Q}^{2,1}(\Omega)$ we have

$$
\begin{aligned}
&60.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T^*, \nabla\cdot\boldsymbol{w})_\Omega + < [\![\lambda^{-1}\boldsymbol{\beta}\cdot\boldsymbol{d} + 1/2\boldsymbol{d}^t\nabla(\lambda^{-1}\boldsymbol{\beta})\boldsymbol{d}]\!]_{\tilde{\Gamma}}\ , \{\boldsymbol{w}\}\cdot\tilde{\boldsymbol{n}}\ >_{\tilde{\Gamma}} \\
&+ < \{T^*\}, [\![\boldsymbol{w}]\!]_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} + B_{stab}((T^*,\boldsymbol{\beta}),\boldsymbol{w}) + < T^*, \boldsymbol{w}\cdot n >_{\Gamma_N} = \ - < j_T, \{\boldsymbol{w}\}\cdot\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} \\
&- < T_D, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_D} + L_{stab}(\boldsymbol{w}); \\
&60.b)\ (\rho c\partial_t T^*, q)_\Omega - < [\![\boldsymbol{\beta} + \nabla\cdot\boldsymbol{\beta}\ \boldsymbol{d}]\!]_{\tilde{\Gamma}}\cdot\boldsymbol{n}(\boldsymbol{n}\cdot\tilde{\boldsymbol{n}}), \{q^*\} >_{\tilde{\Gamma}} + (\nabla\cdot\boldsymbol{\beta}, q^*)_\Omega\ + A_{stab}((T^*,\boldsymbol{\beta}),q^*) \\
&- < \boldsymbol{\beta}\cdot\boldsymbol{n} - h_N, q^* >_{\Gamma_N} = \ (f, q^*)_\Omega - < \sigma(\boldsymbol{n}\cdot\tilde{\boldsymbol{n}}), \{q^*\} >_{\tilde{\Gamma}}\ .
\end{aligned}
\tag{60}
$$

We also need to modify the Nitsche penalty terms defined in (48). The Taylor developments are done at a higher order and are modified on both the temperature and its test function.

$$
\begin{aligned}
\tilde{N}_{A_1}(T^*, q^*) &:= \frac{\alpha}{h} < [\![T^* + \nabla T^*\cdot\boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t\mathcal{H}(T^*)\boldsymbol{d}]\!]_{\tilde{\Gamma}}, [\![q^* + \nabla q^*\cdot\boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t\mathcal{H}(q^*)\boldsymbol{d}]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}} \\
\tilde{N}_{L_1}(q^*) &:= \frac{\alpha}{h} < j_T, [\![q^* + \nabla q^*\cdot\boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t\mathcal{H}(q^*)\boldsymbol{d}]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}}
\end{aligned}
\tag{61}
$$

The enriched weak formulation for case 2 and the associated Nitsche penalty terms are addressed in Appendix A.

Remark 3. To reduce the computational cost of (61) the information known from the mixed form is used to replace both ∇T^* by $-\lambda^{-1}\boldsymbol{\beta}$ and $\mathcal{H}(T^*)$ by $-\nabla(\lambda^{-1}\boldsymbol{\beta})$ in Definition (61) (respectively on (A.2)). The same work is done on the test functions.

4. Globally second order space and time discretization

This section is devoted to the time discretization of the enriched formulation (60). It is a delicate part of our problem. Since the interface is moving, some mesh nodes can switch from one part of the domain to another one. This problem is particularly emphasized by the discontinuous aspect of the variables at the interface. A specific treatment has to be done to properly define the values at such critical points. In addition, the time discretization of the interface motion needs to be performed, and as it is dependent of the solution of the mixed heat equation, their resolution and accuracy impact each other. In the sequel, we consider a time interval $[0, t]$. Let N_t be the number of point discretization, for any $1 \leq n \leq N_t$, we set $\Delta t^n = t^{n+1} - t^n$. For any generic quantity f, we denote by f^n the approximation of f at time t^n.

4.1. Variable initialization

As illustrated in Figure 6, each node located on the surrogate is duplicated, giving two degrees of freedom for those nodes. The left DoF, corresponding to the initial mesh numbering, will be referred as **Initial DoF** while the right DoF will be called **Complementary DoF**. In order to validate our scheme, we assume that the physical interface is governed by a 1D motion. Then, the interface location can be simply described by one of the two coordinates x or y. Depending on the test we want to perform, the velocity of the interface could be either analytical or computed using the jump condition on $\boldsymbol{\beta}$, as in a classical Stefan problem. This simplified motion has no impact on the update in time of the surrogate neither on the proposed reconstruction, but allows a proper validation of the method. To adapt the method to a more complex motion, a level set description of the interface or a front tracking technique has to be used.

At each time step, the surrogate interface $\tilde{\Gamma}^n$ has to be updated. Note that $\tilde{\Gamma}^n$ and $\tilde{\Gamma}^{n+1}$ are generally different. Some points of $\tilde{\Gamma}^n$ at time t^n do not belong to $\tilde{\Gamma}^{n+1}$ and lies in $\tilde{\Omega}^n_+$ or $\tilde{\Omega}^n_-$ while conversely points of $\tilde{\Omega}^n_+$ or $\tilde{\Omega}^n_-$ can move to $\tilde{\Gamma}^{n+1}$. The points leaving the surrogate $\tilde{\Gamma}^n$ can be treated easily. Indeed, at time t^n, we know the values of $(T^{+,n}, \boldsymbol{\beta}^{+,n})$ and $(T^{-,n}, \boldsymbol{\beta}^{-,n})$

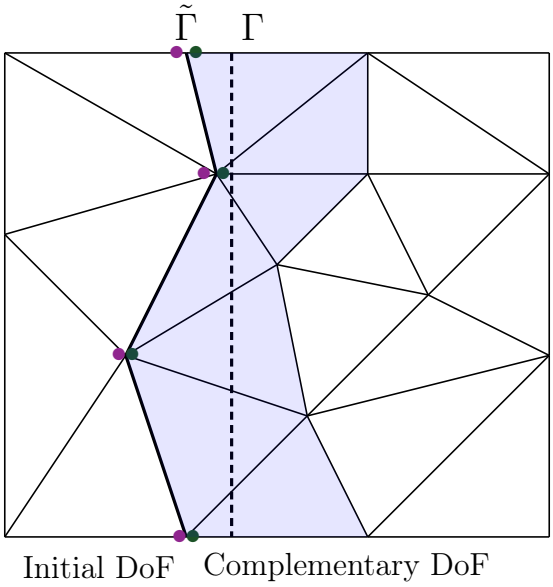
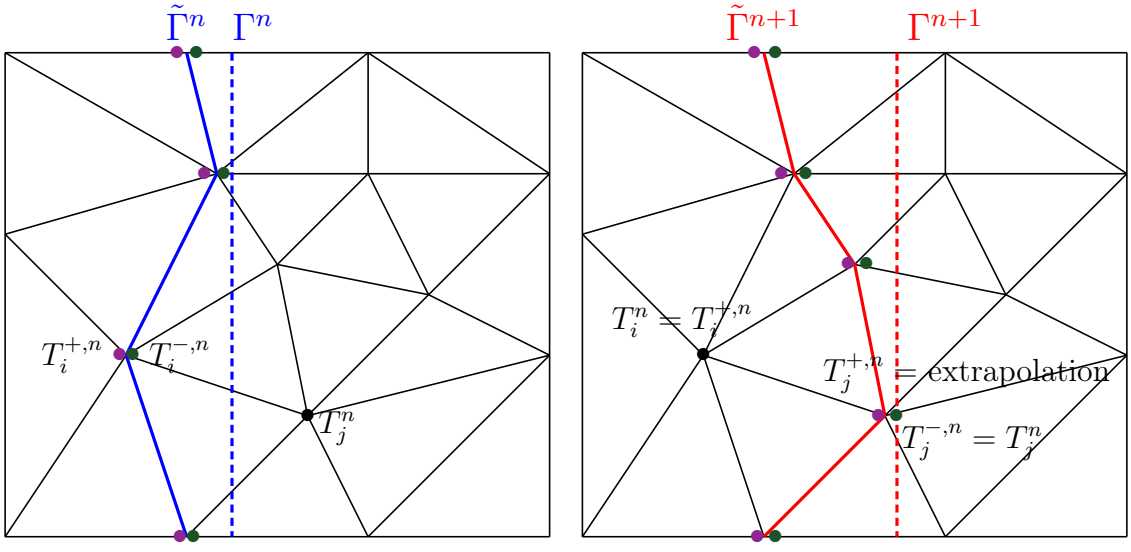


Figure 6: Interface structure

on both side of the interface. Then if the node belongs to Ω_+^n, we set $(T^n, \boldsymbol{\beta}^n) = (T^{+,n}, \boldsymbol{\beta}^{+,n})$ while if it belongs to Ω_-^n, then $(T^n, \boldsymbol{\beta}^n) = (T^{-,n}, \boldsymbol{\beta}^{-,n})$. For example, in Figure 7, the interface is moving to the right, and the left value is conserved.

The treatment of the entering nodes in the surrogate $\tilde{\Gamma}^{n+1}$ is more delicate, since it requires a duplicated DoF. If the node belongs to Ω_+, then we set $(T^{+,n}, \boldsymbol{\beta}^{+,n}) = (T^n, \boldsymbol{\beta}^n)$ whereas if the node belongs to Ω_-, then we define $(T^{-,n}, \boldsymbol{\beta}^{-,n}) = (T^n, \boldsymbol{\beta}^n)$. For example, in Figure 7 the right value, i.e. the green DoF, is initialized. However, the other side of the interface needs to be defined for time t^n somehow. Missing values are constructed using polynomial extrapolation.



(a) Interface at time t^n

(b) Interface at time t^{n+1}

Figure 7: Adaptation of the solution vector at time t^n for the new structure at time t^{n+1}

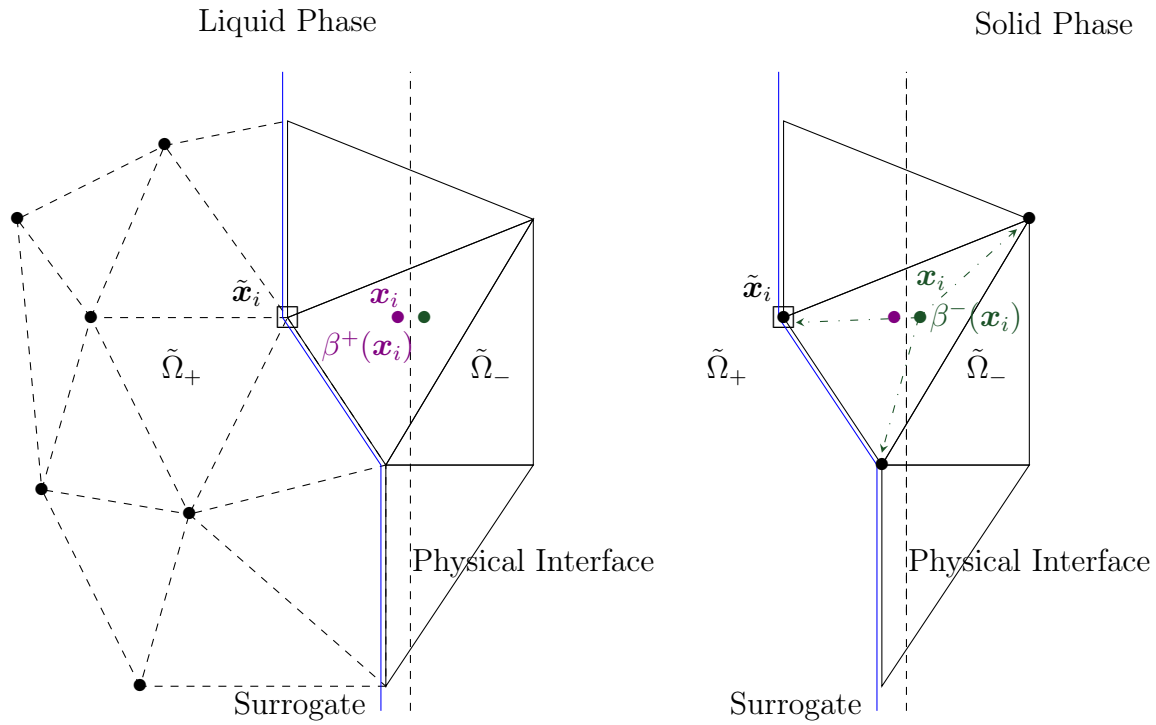
This part must be done carefully in order to preserve the accuracy of the FE approximation (see Figure 8). The temperature being approximated quadratically and the flux linearly, a second order polynomial extrapolation for T and a first order polynomial extrapolation for $\boldsymbol{\beta}$

are employed. The stencil for each node is chosen in the proper sub-domain depending on the interface motion (see Figure 8.b).

A least square minimization is performed using at least a six nodes stencil. If $\boldsymbol{A}$ and $\boldsymbol{b}$ denotes the matrix and right-hand-side vector of the least square reconstruction, two strategies are tested :

- The first strategy consists in solving a least square problem $||\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}||_2^2$, using the dual formulation $\boldsymbol{A}^t\boldsymbol{A}\boldsymbol{x} = \boldsymbol{A}^t\boldsymbol{b}$.
- The second strategy consists in using the QR decomposition of $\boldsymbol{A}$.

The test case in Section 5.2 demonstrates that the QR decomposition, using Householder transformations, is more stable and more accurate.



(a) Extrapolation (b) Interpolation

Figure 8: Visualization of the jump reconstruction method

4.2. Approximation of the flux jump

Since the interface is moving, the duplication of the DoF has to be performed at each time step. The motion of the physical interface Γ is driven by the flux jump $[\![\boldsymbol{\beta}]\!]_\Gamma \cdot \boldsymbol{n}$. However, it is clear that in our configuration, one can compute this jump on the surrogate one. As a consequence, we need to recover the flux jump on the physical interface.

For each node on the surrogate $\tilde{\boldsymbol{x}}_i$, left and right values of each variables are known. Using the mapping M, we associate to $\tilde{\boldsymbol{x}}_i$ the corresponding node $\boldsymbol{x}_i$ on the physical interface. Then any value on the physical interface at the node $\boldsymbol{x}_i$ can be recovered by extrapolation or interpolation, depending on its location inside the domain $\tilde{\Omega}_+$ or $\tilde{\Omega}_-$ (here $\tilde{\Omega}_-$ in figure 8(b)).

More precisely, to obtain an approximated value of $\boldsymbol{\beta}^-(x_i)$ of the duplicated DoF, $\boldsymbol{x}_i$ is located on the mesh, see figure 8(b) and a linear interpolation is performed using the values in $\tilde{\Omega}_-$. For

$\boldsymbol{\beta}^+(\boldsymbol{x}_i)$, the same strategy as the one used for variables initialization is employed, see Section 4.1. An extrapolation using the solution in Ω_+, at specific points of the stencil (black dots in figure 8(a)), allows to recover the value $\boldsymbol{\beta}^+(\boldsymbol{x}_i)$.

Since the flux is represented as a piecewise linear polynomial, a linear interpolation/extrapolation is then sufficient. To recover the left and right values of the temperature, a second order interpolation/extrapolation needs to be set up, since T is a piecewise quadratic polynomial thanks to the enrichment.

Then, we are able to define for each node, and for both side of the surrogate, a corresponding value of our unknowns on the physical interface. Note that, in the sequel, we need explicitly the flux jump at the physical interface, to update at each time step the position front.

4.3. Second order extrapolated BDF2 scheme for the interface motion

In this section, we discuss the resolution of the Stefan condition (12). This part of our resolution scheme has to be handled carefully if one wants to avoid extra computational costs. Indeed, even though (12) is an ordinary differential equation (ODE), the right hand side depends on the solution of System (10). Note that an implicit scheme requires to know $[\![\boldsymbol{\beta}^{n+1}]\!]_{\tilde{\Gamma}^{n+1}}$ and $\tilde{\Gamma}^{n+1}$ at time t^{n+1} and thus imposes a Newton procedure. Moreover, the use of a first order scheme will degrade the temperature/flux accuracy. This is why we want to solve the Stefan condition (12) with a second order explicit scheme. In this work, we choose the extrapolated BDF2 strategy. Considering a generic equation $y'(t) = f(t, y(t))$, the BDF2 scheme can be written under the form

$$\gamma_{n+1}u^{n+1} + \gamma_n u^n + \gamma_{n-1}u^{n-1} = f(t^{n+1}, y^{n+1}), \tag{62}$$

where

$$\gamma_{n+1} = \frac{2\Delta t^n - \Delta t^{n-1}}{\Delta t^n(\Delta t^n + \Delta t^{n-1})},\ \gamma_n = -\frac{\Delta t^n + \Delta t^{n-1}}{\Delta t^n \Delta t^{n-1}},\ \gamma_{n-1} = \frac{\Delta t^n}{\Delta t^{n-1}(\Delta t^n + \Delta t^{n-1})}. \tag{63}$$

Here, since we want to use an explicit scheme, we propose to extrapolate the right-hand-side of (62) with a second order accuracy using times t^n and t^{n-1}. This leads to the second order extrapolated BDF (exBDF2) scheme:

$$\gamma_{n+1}u^{n+1} + \gamma_n u^n + \gamma_{n-1}u^{n-1} = \frac{\Delta t^n + \Delta t^{n-1}}{\Delta t^{n-1}} f(t^n, y^n) - \frac{\Delta t^n}{\Delta t^{n-1}} f(t^{n-1}, y^{n-1}). \tag{64}$$

For the problem at stake, we have to solve

$$\chi'(t) = v(t) \tag{65}$$

where $\chi(t) = \boldsymbol{\chi}(t)\cdot\boldsymbol{n}$ is the coordinate of the interface in the normal direction and $v(t) = \boldsymbol{v}(t)\cdot\boldsymbol{n}$ is the normal velocity, defined as $\boldsymbol{v}(t)\cdot\boldsymbol{n} = \dfrac{1}{\rho L m}[\![\boldsymbol{\beta}]\!]_\Gamma \cdot \boldsymbol{n}$. Knowing the normal flux and the interface positions at times t^{n-1} and t^n (see Section 4.2), we can then evaluate the position at time t^{n+1} using the following scheme

$$\chi^{n+1} = -\frac{\gamma_n}{\gamma_{n+1}}\chi^n - \frac{\gamma_{n-1}}{\gamma_{n+1}}\chi^{n-1} + \frac{1}{\gamma_{n+1}}\left(\frac{\Delta t^n + \Delta t^{n-1}}{\Delta t^{n-1}}v^n - \frac{\Delta t^n}{\Delta t^{n-1}}v^{n-1}\right), \tag{66}$$

$$v^n = \frac{1}{\rho L m}[\![\boldsymbol{\beta}^n]\!]_{\Gamma^n}\cdot\boldsymbol{n}^n,\quad v^{n-1} = \frac{1}{\rho L m}[\![\boldsymbol{\beta}^{n-1}]\!]_{\Gamma^{n-1}}\cdot\boldsymbol{n}^{n-1}. \tag{67}$$

Note that as v depends directly on the flux jump $[\![\boldsymbol{\beta}]\!]$, a high order resolution of the flux will preserve the accuracy of the exBDF2 scheme. As explained in the previous section, the use of

the enrichment is necessary to provide the willing accuracy.

Remark 4. Other one stage second order explicit scheme can be considered here, as for instance the Adam-Bashforth scheme.

4.4. Second order time scheme

This section is devoted to the time discretization of the variational problem. We propose to use a BDF2 scheme.

The main advantage is that the weak formulation and the computation of the location of the interface can be written at time t^{n+1} only. Focusing on case 1 and its enriched weak formulation (the methodology being exactly the same for case 2, see (A.1)), the scheme is:
Finding $(T^*, \boldsymbol{\beta}) \in \mathcal{Q}^{2,1}(]0,t[,\Omega)$ s.t. $\forall (q^*, \boldsymbol{w}) \in Q^{2,1}(\Omega)$ we have

$$
\begin{aligned}
&68.a)\ (\lambda^{-1}\boldsymbol{\beta}^{n+1}, \boldsymbol{w})_{\Omega^{n+1}} - (T^{*,n+1}, \nabla\cdot\boldsymbol{w})_{\Omega^{n+1}} + < T_D, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_D^{n+1}} + < T^{*,n+1}, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_N^{n+1}} \\
&+ < [\![\lambda^{-1}\boldsymbol{\beta}^{n+1}\cdot\boldsymbol{d} + 1/2\boldsymbol{d}^t\nabla(\lambda^{-1}\boldsymbol{\beta}^{n+1})\boldsymbol{d}]\!]_{\tilde{\Gamma}}, \{\boldsymbol{w}\}_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}^{n+1}} + < \{T^{*,n+1}\}_{\tilde{\Gamma}}, [\![\boldsymbol{w}]\!]_{\tilde{\Gamma}}\cdot\tilde{n} >_{\tilde{\Gamma}^{n+1}} \\
&= - < j_T^{n+1}, \{\boldsymbol{w}\}_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}^{n+1}} \\
&68.b)\ (\rho c(\gamma_{n+1}T^{*,n+1} + \gamma_n T^{*,n} + \gamma_{n-1}T^{*,n-1}), q^*)_{\Omega^{n+1}} + (\nabla\cdot\boldsymbol{\beta}^{n+1}, q^*)_{\Omega^{n+1}} - < \boldsymbol{\beta}^{n+1}\cdot\boldsymbol{n} - h_N^{n+1}, q^* >_{\Gamma_N^{n+1}} \\
&- < [\![\boldsymbol{\beta}^{n+1} + \nabla\cdot\boldsymbol{\beta}^{n+1}\boldsymbol{d}]\!]_{\tilde{\Gamma}}\cdot\boldsymbol{n}(\boldsymbol{n}\cdot\tilde{\boldsymbol{n}}), \{q^*\}_{\tilde{\Gamma}} >_{\tilde{\Gamma}^{n+1}} = (f^{n+1}, q^*)_{\Omega^{n+1}} - < \sigma(\boldsymbol{n}\cdot\tilde{\boldsymbol{n}}), \{q^*\} >_{\tilde{\Gamma}} .
\end{aligned}
\tag{68}
$$

The stabilization terms $A_{stab}((T^{*,n+1}, \boldsymbol{\beta}^{n+1}), q^*)$, $B_{stab}((T^{*,n+1}, \boldsymbol{\beta}^{n+1}), \boldsymbol{w})$, $L_{stab}(\boldsymbol{w})$, and the Nitsche penalty terms (61) are also added to Scheme (68). In this latter Scheme (68), some "$n+1$" have been omitted to lighten the notations, but jump, average, distance function and normals are for the configuration at time t^{n+1}. The only issue in (68) is the definition of T^n and T^{n-1} for the nodes that change domain for time t^{n+1}. In such cases, according to their previous location, extrapolations or interpolations as presented in the previous sections need to be performed to have a consistent value for the previous time steps.

5. Numerical Results

In Section 5.1, results for polynomial analytical functions are analyzed to highlight the second order accuracy obtained by the enrichment and verify the efficiency of the reconstruction used for the interface motion. Validations are done using continuous and discontinuous variables.
Then, results for a physical example modelling the melting of a semi infinite ice block is presented in Section 5.2. To reduce the complexity generated by the moving interface the second case of interface conditions will be used. It enables the use of the Stefan condition, for the front motion, and imposed the melting temperature as a Dirichlet condition connecting the two phases.

5.1. Second order accuracy verification

For the different test cases of Section 5.1, Δt is defined by $\Delta t = \dfrac{h\ CFL}{2v}$, where v is a constant velocity used to move the interface. The coordinate defining the interface position is then updated by $\chi_{inter}^{n+1} = \chi_{inter}^{n} + v \times \Delta t$. The initial interface position along with the analytical solution is used to initialize the two domains $\tilde{\Omega}_+$ and $\tilde{\Omega}_-$. In the following, the initial interface position $\chi_{inter}^{0} = 0.5$ is chosen identical for all meshes and test cases, and determines the initial time of the simulation. The $\mathbb{L}^2$ error is displayed for the final time tf, identical for all test cases s.t. $tf = 2$.

5.1.1. Machine precision

First, machine precision is obtained for polynomial analytical solutions of degree less or equal to two for both set of interface conditions. $\mathbb{L}^2$ Errors on T and $\boldsymbol{\beta}$ are given for different tests. Case 1 is available in Tables 1 and 2, while case 2 is displayed in Tables 3 and 4.
The domain is rectangular $\Omega = [0,2] \times [0,1]$. Triangular elements of characteristic length $h = 0.05$ are used to mesh the domain. $CFL = 1$, $\lambda_+ = 0.6$, $\lambda_- = 2.1$, $v = 0.5$, $\zeta_{div} = 0.5$ and analytical functions for T are chosen such as

$$\mathbb{P}^0 \rightarrow T := \begin{cases} 7t+1 & \text{if } (x,y) \in \tilde{\Omega}_+ \\ 3t+3 & \text{if } (x,y) \in \tilde{\Omega}_- \end{cases} ; \ \mathbb{Q}^1 \rightarrow T := \begin{cases} x+t & \text{if } (x,y) \in \tilde{\Omega}_+ \\ xyt & \text{if } (x,y) \in \tilde{\Omega}_- \end{cases}$$

$$\mathbb{P}^2 \rightarrow T := \begin{cases} 4+33xy+21y^2+11t+6t^2 & \text{if } (x,y) \in \tilde{\Omega}_+ \\ 77+x^2+2t-8t^2 & \text{if } (x,y) \in \tilde{\Omega}_- \end{cases} .$$

T Polynomial	eT enriched sym	eT enriched non-sym	eT non enriched
$\mathbb{P}^0$	8.69037675694078E-013	1.07246296107150E-012	5.77492039596584E-013
$\mathbb{Q}^1$	1.98905839613197E-012	3.29073939077805E-013	8.17679869657093E-005
$\mathbb{P}^2$	9.87454293814777E-013	3.88923825456129E-012	9.24537618134559E-003

Table 1: $\mathbb{L}^2$ Errors on T for polynomials of different degrees - Case 1

T Polynomial	e$\boldsymbol{\beta}$ enriched sym	e$\boldsymbol{\beta}$ enriched non-sym	e$\boldsymbol{\beta}$ non enriched
$\mathbb{P}^0$	5.87760006006547E-012	1.17309095334435E-011	6.34466424474283E-012
$\mathbb{Q}^1$	7.34523157516188E-012	5.49049321486107E-012	7.10842777209640E-003
$\mathbb{P}^2$	4.70098442382738E-012	1.24055486419729E-011	0.182914290803606

Table 2: $\mathbb{L}^2$ Errors on $\boldsymbol{\beta}$ for polynomials of different degrees - Case 1

As can be seen in Tables 1 and 2, doing a symmetric enrichment (enriched sym) or a non symmetric enrichment (enriched non-sym) does not change the accuracy of the method and both approaches give the expected machine precision. The non-enriched approach (non enriched) does not allow to achieve second order, as expected. Indeed, the machine precision is not obtained for $\mathbb{Q}^1$ and $\mathbb{P}^2$ functions, and the error on the flux $\boldsymbol{\beta}$ is bigger. It comes from the definition (59), where the new terms added to the usual $\mathbb{P}^1$ formulation concerned the flux variable.

T Polynomial	eT enriched sym	eT enriched non-sym	eT non enriched
$\mathbb{P}^0$	6.45889120732631E-013	5.56607024070457E-013	1.28224367945556E-012
$\mathbb{Q}^1$	3.41553121024897E-013	4.34799948808748E-013	4.00045558192290E-005
$\mathbb{P}^2$	5.28262199105204E-013	5.12808329455716E-013	5.63157166009521E-003

Table 3: $\mathbb{L}^2$ Errors on T for polynomials of different degrees - Case 2

T Polynomial	e$\boldsymbol{\beta}$ enriched sym	e$\boldsymbol{\beta}$ enriched non-sym	e$\boldsymbol{\beta}$ non enriched
$\mathbb{P}^0$	3.29044974771348E-012	3.71568549940826E-012	4.97683492500090E-012
$\mathbb{Q}^1$	4.54975964210432E-012	6.40200226600249E-012	5.49640705908567E-003
$\mathbb{P}^2$	6.35716917005916E-012	4.66651583565093E-012	4.10827076797471E-002

Table 4: $\mathbb{L}^2$ Errors $\boldsymbol{\beta}$ for polynomials of different degrees - Case 2

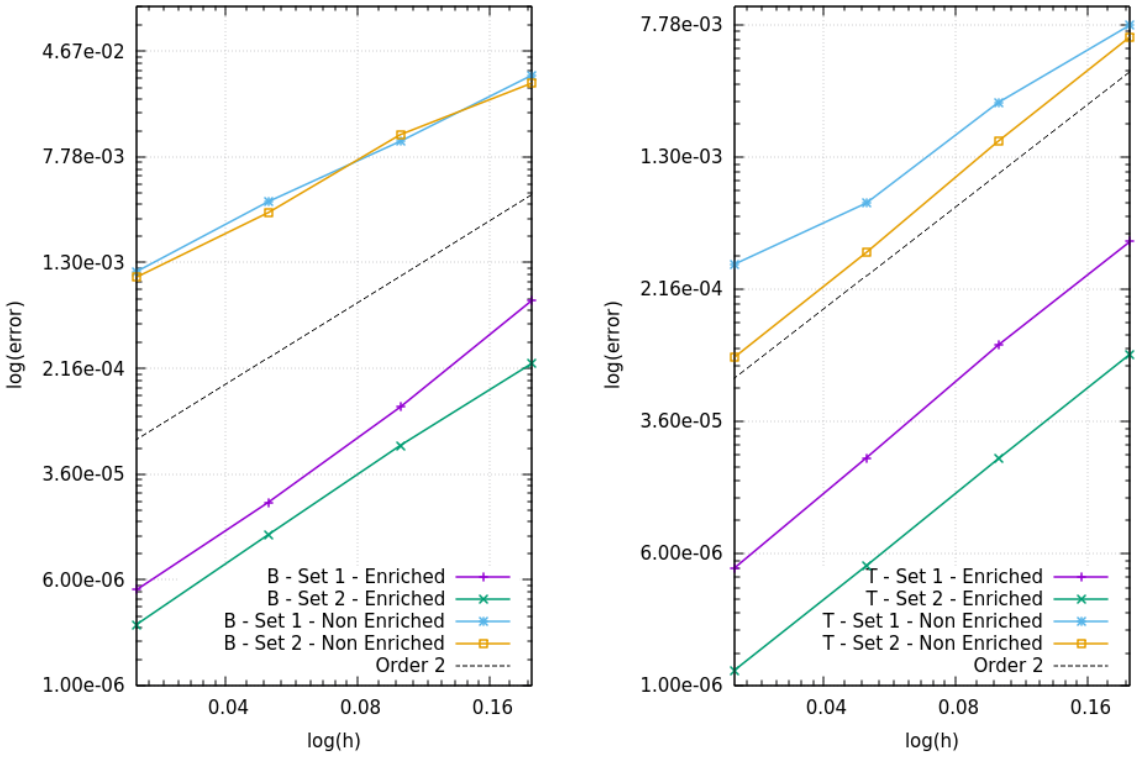
Similar results are obtained with the second set of interface conditions in Table 3 and 4. By comparing Tables 1 and 2 with Tables 3 and 4 we globally notice a better precision with the second set of interface conditions (case 2). Moreover, depending on the choice made for the imposition of conditions at the interface, both options are suitable. In the following, for the tests in Section 5.1 we will look at the results for both set of interface conditions, where the front displacement is not related to the flux jump. The particularity of the Stefan problem will be handle in Section 5.2 with the use of the second set of interface conditions.

5.1.2. Flux discontinuity

On the same computational domain, we are now looking at a continuous solution in temperature T and a discontinuous solution in flux $\boldsymbol{\beta}$, this test case is referred as Test 1. For that purpose, λ is chosen s.t., $\lambda_+ = 0.6$ and $\lambda_- = 2.1$. T is then defined by

$$T(x,y,t) = \begin{cases} log(t+1)/2 \ + y^2 \text{ if } (x,y) \in \tilde{\Omega}_+, \\ \tilde{\Omega}_-. \end{cases}$$

A grid convergence study is performed for Test 1. Figure 9 compares $\mathbb{L}^2$ errors on $\boldsymbol{\beta}$ and T to the theoretical second order (dotted line). As expected, Figure 9.b shows a second order accuracy for T with the enriched approach, but for the first set of interface conditions the order is decreasing at convergence and is not second order anymore. Moreover, the errors are significantly lower using the enriched approach. For the flux $\boldsymbol{\beta}$, without enrichment, second order is not achieved for both cases, while it is obtained with enrichment in Figure 9(a).



(a) $\boldsymbol{\beta}$ discontinuous - Error in log scale

(b) T continuous - Error in log scale

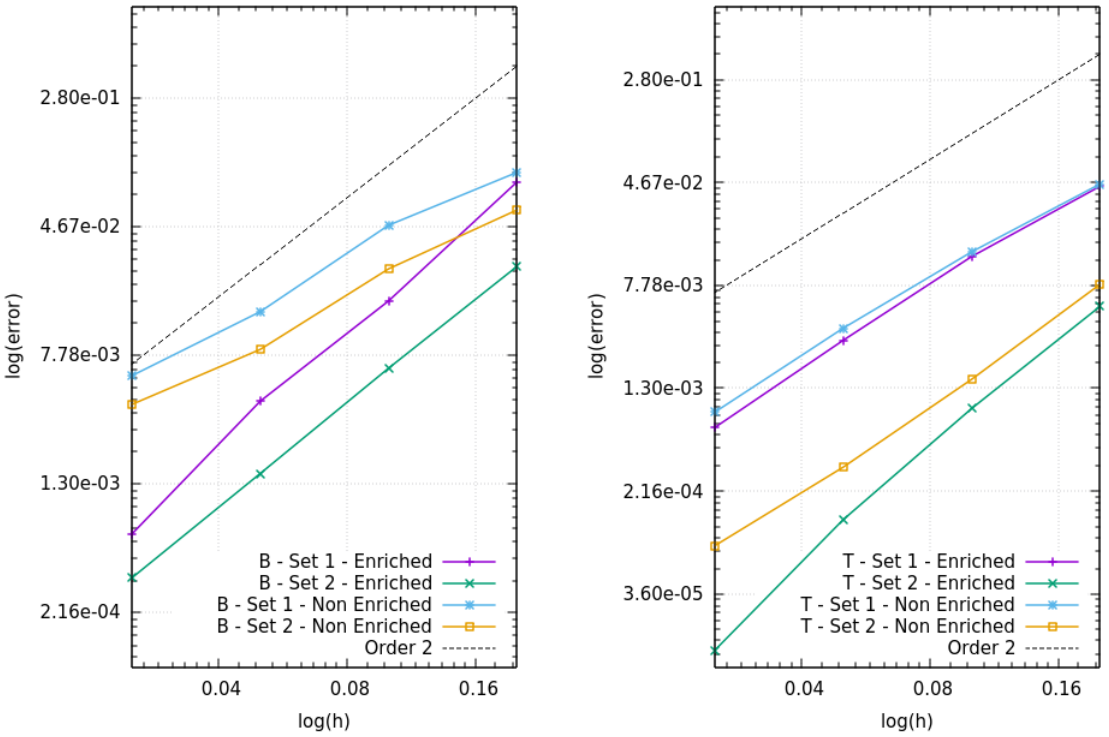
Figure 9: Test 1 : Comparison between the enriched and non enriched method - $CFL = 1.$, $v = 0.5$, $\zeta_{div} = 0.5$

In Test 2, we investigate the situation where T and $\boldsymbol{\beta}$ are both discontinuous. Let λ such that

$\lambda_+ = 0.6$, $\lambda_- = 2.1$ and T defined by

$$T(x,y,t) = \begin{cases} x^4 + 3xyt + 7 & \text{if } (x,y) \in \tilde{\Omega}_+, \\ log(x+1) + t^3 + y & \text{if } (x,y) \in \tilde{\Omega}_- \ . \end{cases}$$

Second order accuracy is obtained for T and for both approaches, see Figure 10(b). Moreover, on all meshes, the error is always smaller using the enrichment of the temperature. On the flux, Figure 10(a), the enrichment allows to achieve second order accuracy on $\boldsymbol{\beta}$, which was not the case without enrichment for case 1 and case 2, where the differences are largely visible. The errors are also globally reduced on all meshes when enrichment is used.



(a) $\boldsymbol{\beta}$ discontinuous - Error in log scale

(b) T discontinuous - Error in log scale

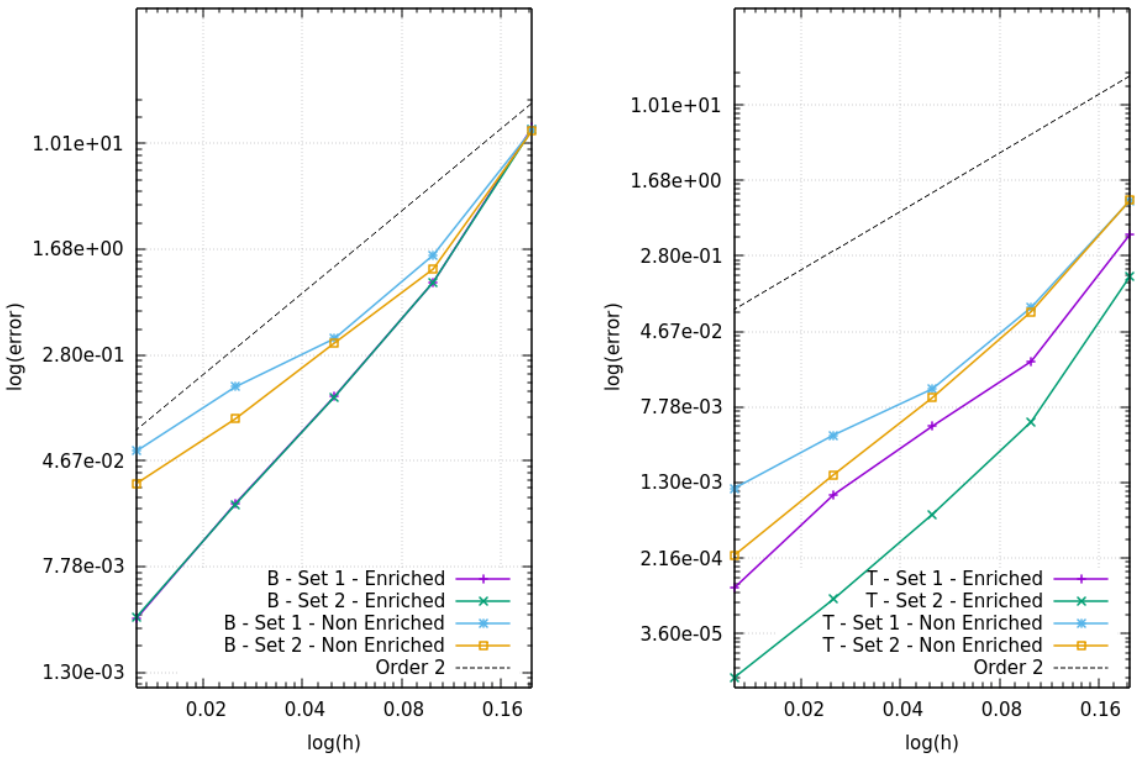
Figure 10: Test 2 : Comparison between the enriched and non enriched method - $CFL = 1.$, $v = 0.5$, $\zeta_{div} = 0.5$

5.1.3. Discontinuity of the primal variable

The last test case deals with a continuous flux $\boldsymbol{\beta}$ and a discontinuous variable T, λ is chosen s.t $\lambda_+ = \lambda_- = 2.1$, and T is defined by

$$T(x,y,t) = \begin{cases} exp(3xy) + log(t+1)/2 \ + t^3 & \text{if } (x,y) \in \tilde{\Omega}_+, \\ exp(3xy) + 2\sqrt{t} & \text{if } (x,y) \in \tilde{\Omega}_- \ . \end{cases}$$

Without enrichment, the approximation on $\boldsymbol{\beta}$ is of order 1 (see Figure 11(a)), while using the enrichment provides second order accuracy. Note also that the two enriched curves for case 1 and 2 are mingled. As for T, it is second order accurate using both approaches for case 2. For case 1, the second order is not preserved on all meshes on the curve representing the non enriched simulation, see Figure 11(b). For each variable, the enrichment decreases the error on all meshes.



(a) $\boldsymbol{\beta}$ continuous - Error in log scale

(b) T discontinuous - Error in log scale

Figure 11: Test 3 : Comparison between the enriched and non enriched method - $CFL = 1.$, $v = 0.5$, $\zeta_{div} = 0.5$

Whatever the combination continuous/discontinuous T and/or $\boldsymbol{\beta}$ at the interface, the enrichment and proposed reconstruction at the interface insure a second order accuracy for T and $\boldsymbol{\beta}$. For all cases and for all the meshes, the enrichment approach decreases the $\mathbb{L}^2$ errors on T and $\boldsymbol{\beta}$.

5.2. Ice block melting

5.2.1. Test case data

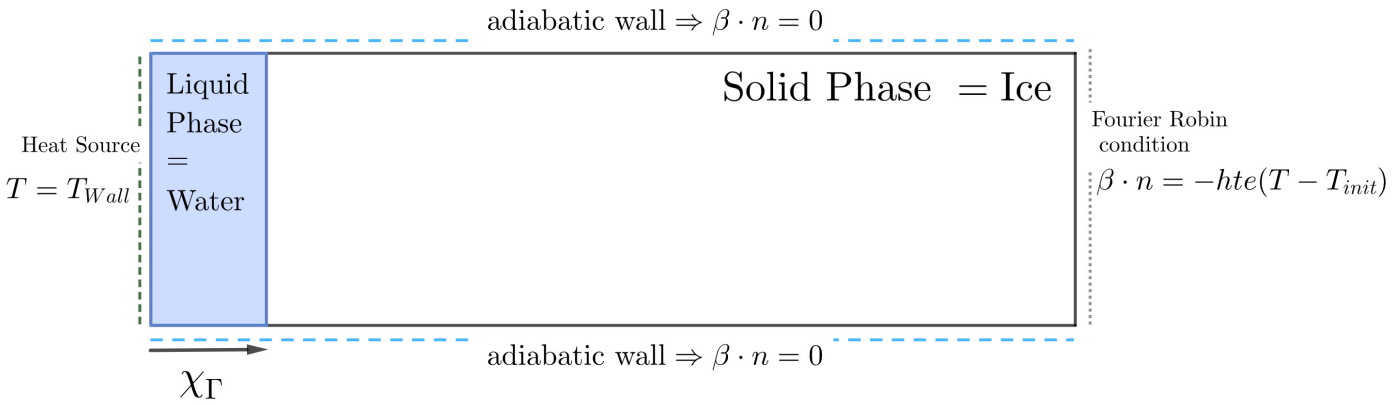


Figure 12: Illustration of the test case

This test simulates the melting of a semi-infinite ice block, for which analytical solutions for the temperature and the front position have been derived in [46]. The computational domain, Figure 12, is a rectangular box $\Omega = [0\,m, 0.5\,m] \times [0\,m, 0.025\,m]$. A Dirichlet boundary condition

is imposed on the left side of the domain with a temperature $T_{wall} > T_m$, where T_m is the ice melting temperature. The melting front will then move from left to right. Adiabatic wall boundary conditions are imposed on top and bottom of the domain. The right boundary condition models the convective heat transfer with ambient air and the length of the domain is sufficiently large for the condition to not affect the position of the front.

We denote

$$
\begin{aligned}
T_{Wall} &= 283.15\ (K) && \text{Temperature of the heat source}\\
T_r &= 263.15\ (K) && \text{Exterior temperature}\\
T_{init} &= 263.15\ (K) && \text{Initial temperature of the ice block}\\
Tm &= 273.15\ (K) && \text{Phase change temperature}\\
hte &= 300\ (w.m^{-2}.K^{-1}) && \text{Heat transfer coefficient}\\
\rho &= 1000\ (Kg.m^{-3}) && \text{Material density}\\
Lm &= 333000\ (J.Kg^{-1}) && \text{Latent heat}\\
\lambda_{Ice} &= 2.1\ (W.m^{-1}.K^{-1}),\ \lambda_{Water} = 0.6\ (W.m^{-1}.K^{-1}) && \text{Thermal conductivity}\\
c_{Ice} &= 2060\ (W.m^{-1}.K^{-1}), c_{Water} = 4185\ (W.m^{-1}.K^{-1}) && \text{Specific heat}
\end{aligned}
\tag{69}
$$

In this work, the density is chosen constant, s.t. $\rho_{ice} = \rho_{water} = 1000\, Kg.m^{-3}$. We also introduce other variables called Stefan numbers. The Stefan number characterizes the rate of phase change defined by the ratio between the sensible heat (heat exchanged) and the latent heat (energy released or absorbed).

$$
St_W := \frac{c_{Water}(T_{Wall} - T_m)}{Lm} \quad ; \quad St_I := \frac{c_{Ice}(T_m - T_r)}{Lm} \tag{70}
$$

We introduced two different sets of interface conditions. For this particular test, we propose the use of the second set of interface conditions to take care of the Stefan condition (12), defining the problem as a free interface problem. Then, the flux jump will not appear in the weak formulation, which prevent the calculation a posteriori of $[\![\beta]\!]$ necessary with the first set of interface conditions (case 1).

In this study the special case of the creation of an interface (ice melting process starting with a unique solid/ice domain) is not addressed and will be discussed in future work. For this reason, the initial position of the interface is fixed and depends on the initial time of the simulation. Multiple unstructured triangular meshes with a refined zone in the area of the front motion are generated, where h is the characteristic size of the smallest elements of each mesh, that is the refined part around the initial interface position (see Figure 13 and Table 5).

h	Number of Nodes	Number of elements
0.00008	15569	30463
0.00005	25576	50264
0.00004	32577	964117
0.00003	45515	89762
0.00002	73015	144314

Table 5: Mesh characteristic following h

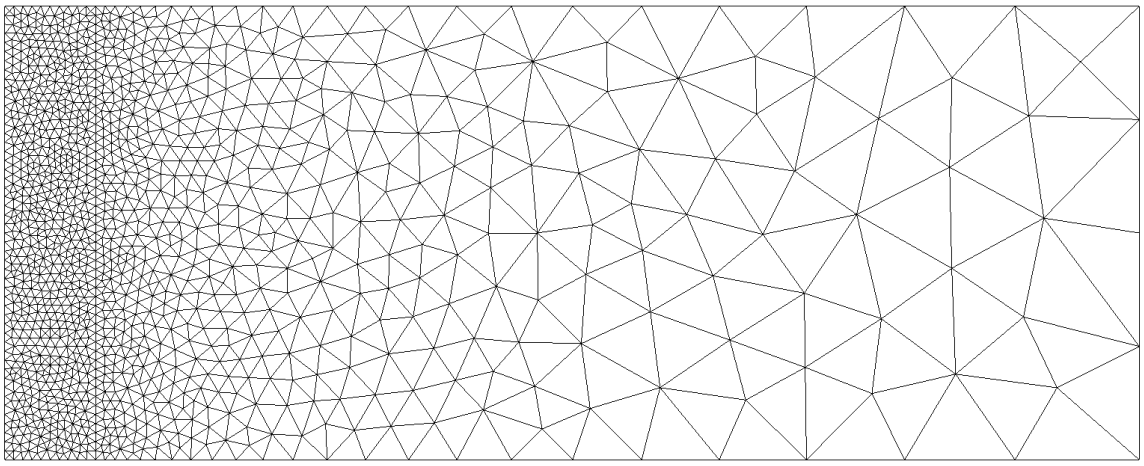


Figure 13: Discretization of the domain of resolution

For our different simulations the time step is given by $\Delta t := \dfrac{h}{2\chi'(t_{min})}$ and the final time $t_f = 5$. The function $\chi(t)$ is the analytical position of the interface and is used to validate our results.

$$\begin{aligned} &\chi(t) := 2\chi\sqrt{\frac{\lambda_{Water}}{\rho c_{Water}}t} \quad ; \quad \nu := \sqrt{\frac{\lambda_{Water}}{\rho c_{Water}}\frac{\rho c_{Ice}}{\lambda_{Ice}}} \\ &\text{where } \chi \text{ is the solution of :} \\ &\frac{St_L}{exp(\chi^2)erf(\chi)} - \frac{St_S}{\nu exp(\nu^2\chi^2)erfc(\nu\chi)} - \chi\sqrt{\pi} = 0 \quad . \end{aligned} \tag{71}$$

The value of χ is found by solving equation (71) with a Newton method. This gives here $\chi \approx 0.2018255$.

5.2.2. Preliminary Observations

In Figure 14 we show for three different iterations the update made on the surrogate position and the modifications brought by it on the definition of the domain. In blue, in Figure 14, the liquid phase is growing while in the red area, one can see that the solid phase is getting smaller. The elements in light and dark green are those cut by the physical interface on both Ω_+ and Ω_-.

An important part to look at is the impact of the node reconstruction which can possibly deteriorate the exact front position. Following the two approaches described in Section 4.1 we look at the position of the interface for different size of elements and for the two methods. Figure 15 shows the importance of the method employed for the reconstruction of the missing values independently of the update of the interface.

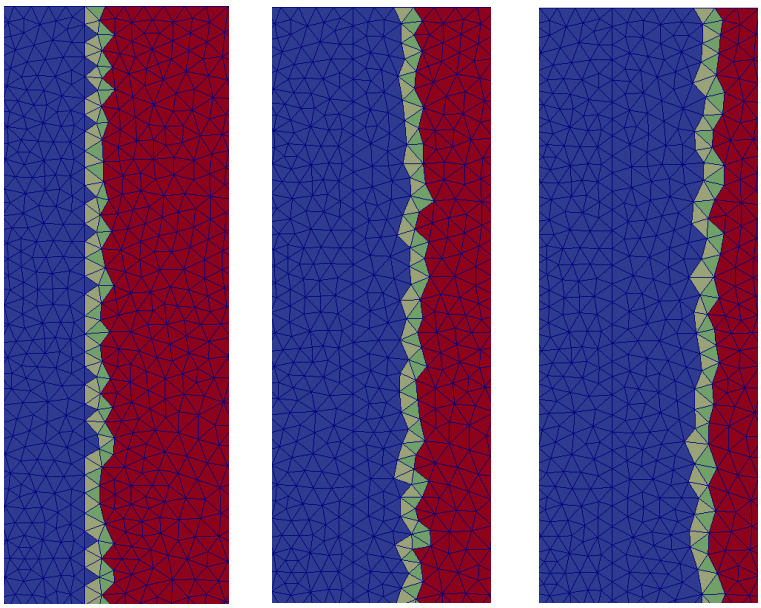
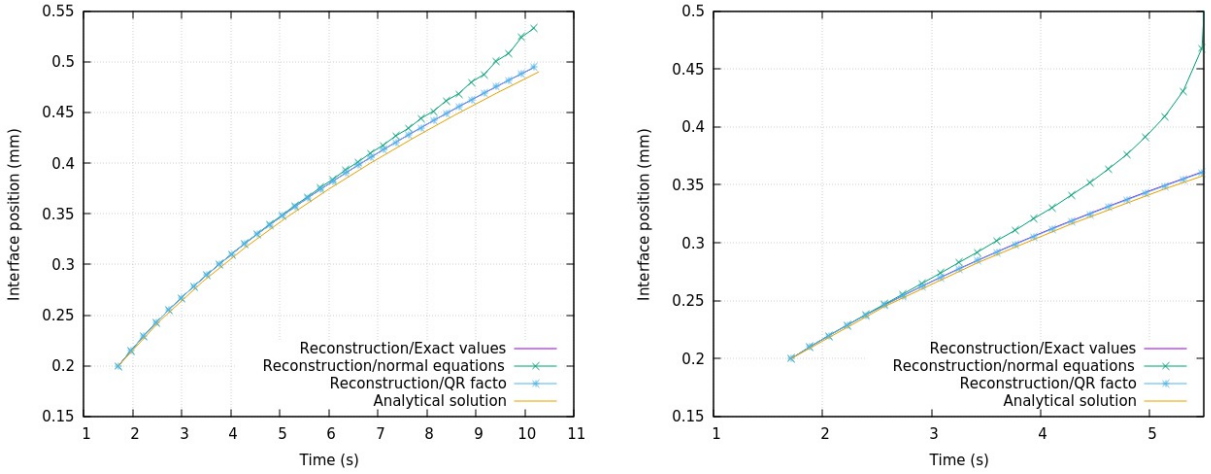


Figure 14: 2D View from the top of the domain (h=0.00008) and the displacement of the surrogate at different time step



(a) h = 0.00003 (b) h = 0.00002

Figure 15: Comparison between the exact interface position and the numerical positions

In Figure 15 we compare the reconstruction at the interface with the exact values from the analytical solutions, the values from the resolution of the normal equations and by the use of the QR factorization. We can observe that the type of reconstruction used to solve the least squares problem can make the numerical position of the front diverging from the analytical position. The phenomenon is even more amplified with smaller elements like in Figure 15(b) We can see that the QR factorization is the best fit, where it has been implemented via the Householder transformation known to have a greater numerical stability.

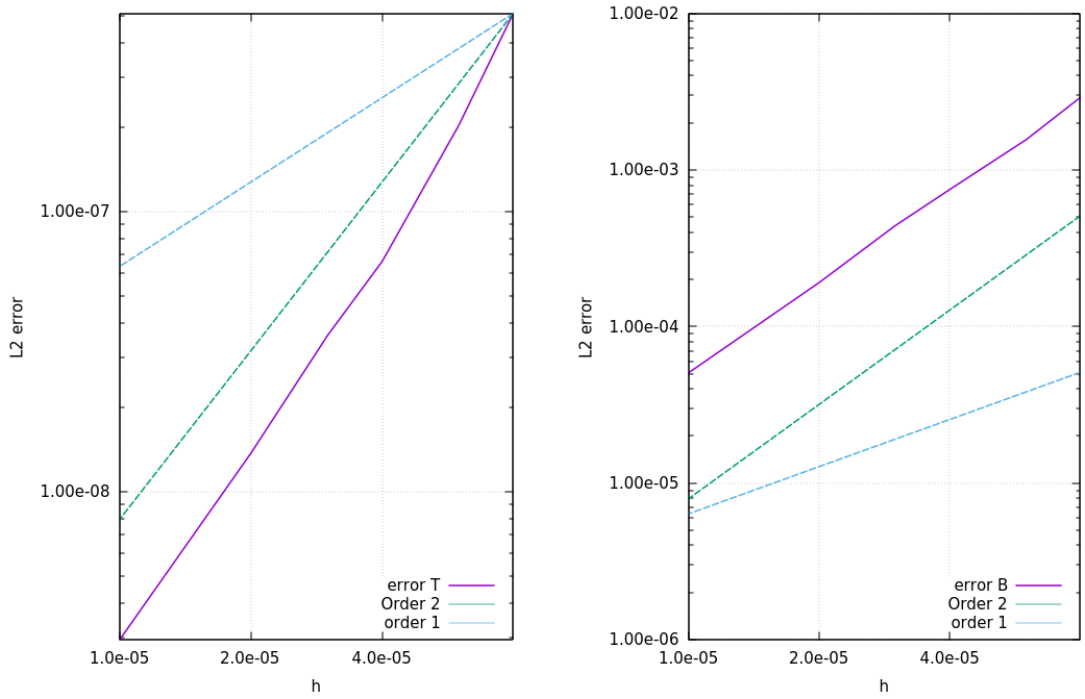
The results obtained with the resolution of the normal equations can be explained by the stencils of points used, with similar kind of points. The phenomenon is even more amplified in Figure 15(b) where the mesh elements are smaller. On the contrary, the QR method does not require the computation of an inverse matrix, which makes this procedure more pertinent.

5.2.3. Results

In order to show properly the accuracy of the method for this particular test we restrain our simulations to meshes with a constant characteristic length.

h	eT	e$\boldsymbol{\beta}$	Order on T	Order on $\boldsymbol{\beta}$
8E-05	5.09E-07	2.90E-03		
6E-05	2.03E-07	1.56E-03	3.195	2.159
4E-05	6.62E-08	7.50E-04	2.763	1.805
3E-05	3.61E-08	4.42E-04	2.113	1.837
2E-05	1.37E-08	1.91E-04	2.394	2.073
1E-05	2.96E-09	5.07E-05	2.206	1.912

Table 6: Enriched Simulation - $\mathbb{L}^2$ Errors obtained on variable $\boldsymbol{\beta}$ and T at final time $Tf = 5$ seconds



(a) Error on T (b) Error on $\boldsymbol{\beta}$

Figure 16: Convergence study - Error at final time Tf = 5 seconds

In Table 6, we show the error obtained for different mesh discretizations. We can observe that the order obtained is indeed 2 for both variables. The values in Table 6 are shown in Figure 16 compared with two reference curves, one for the first order accuracy (blue curve in Figure 16) and one for the second order accuracy (green curve in Figure 16).

We are also able to track properly the position of the front which is an unknown of the problem. In Figure 17 we can see that the more refined the mesh is, the more the numerical front fits the analytical position. Here, to solve the Stefan condition, the scheme detailed in Section 4.3 is employed and shows its ability to conserve the second order of accuracy.

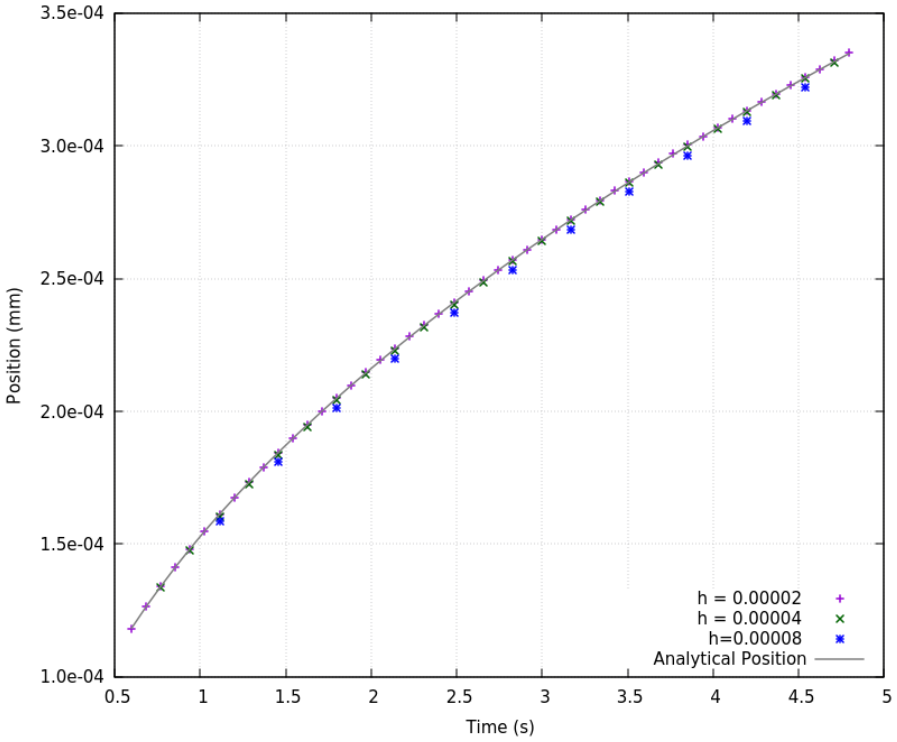
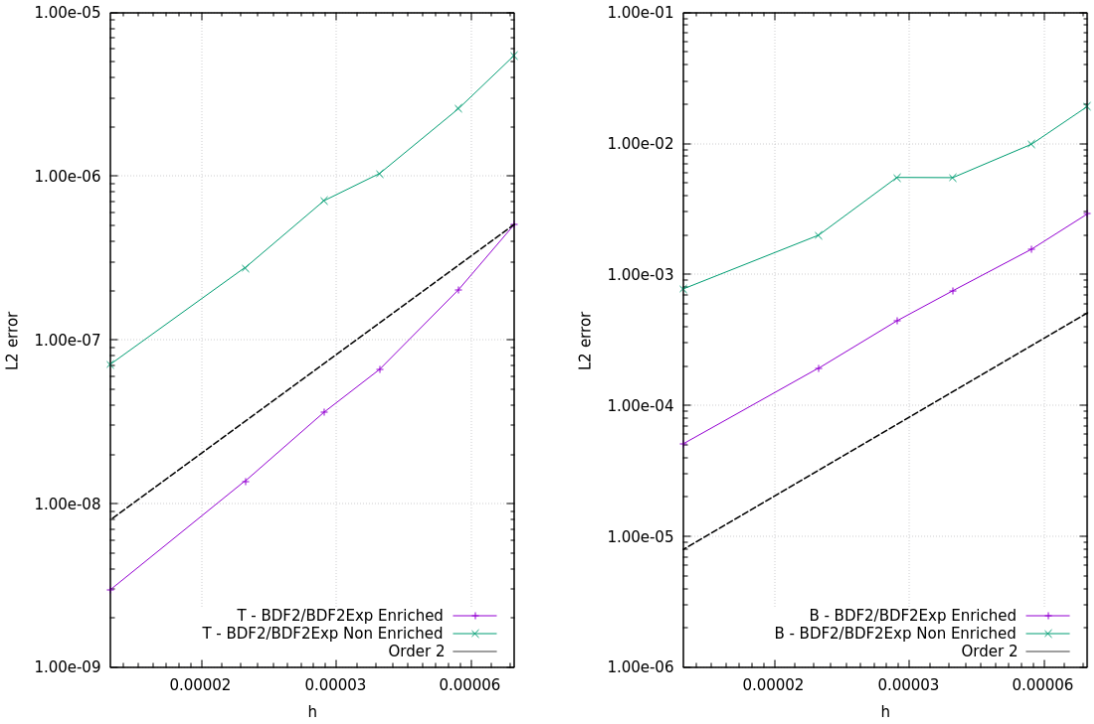


Figure 17: Comparison between the Numerical and Analytical Front Position

In Figure 18 and Tables 6 and 7, we look at the differences between the enriched and non enriched simulations, where all schemes used are second order accurate. We can observe a degradation of the order when the enrichment on the temperature is deactivated, see Figure 18. Then, the enrichment is an effective procedure to obtain and maintain the second order accuracy for the overall method.

h	eT	e$\boldsymbol{\beta}$	Order on T	Order on $\boldsymbol{\beta}$
8E-05	5.49E-06	1.92E-02		
6E-05	2.59E-06	9.87E-03	2.606	2.323
4E-05	1.04E-06	5.48E-03	2.251	1.451
3E-05	7.07E-07	5.50E-03	1.348	-0.012
2E-05	2.76E-07	1.98E-03	2.320	2.521
1E-05	7.06E-08	7.72E-04	1.965	1.357

Table 7: Non Enriched Simulation - Value from the Study Convergence of Figure 18



(a) Error on T (b) Error on $\boldsymbol{\beta}$

Figure 18: Convergence study - Enrichment Comparison

6. Conclusion

The shifted boundary method is a suitable method for simulating moving interfaces. In this work, we focus on the specific case of Stefan problems. We propose a mixed formulation, which is not classically used, but enables to introduce explicitly the gradient essential to the calculation of the speed of the interface. A second order accurate method has been validated. It is based on an enrichment procedure and correct Taylor approximations to account for the boundary conditions of the free interface. Regarding the melting of a semi-infinite ice block, the position of the front is correctly predicted by the proposed approach. Note that the enrichment technique is mandatory to obtain the willing second order accuracy, which justify the proposed mixed formulation. In a future work, we plan to focus on the representation of the moving interface either by a method of level-set or front tracking.

Appendices

Appendix A. Enriched Formulation for case 2

We are interested here in the enriched weak formulation associated to case 2. We have defined case 1 in Section 3.3 by (60) and we proceed the same way in this section to get case 2.
We recall that the enrichment is defined by (59) and is applied on both the temperature and the test function.

Finding $(T^*, \boldsymbol{\beta}) \in \boldsymbol{Q}^{2,1}(\,]0,t[\,,\Omega)$, s.t. $\forall (q^*, \boldsymbol{w}) \in \boldsymbol{Q}^{2,1}(\Omega)$ we have

$$\begin{aligned}
&A.1.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T^*, \nabla\cdot\boldsymbol{w})_\Omega + < T_D, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_D} + < T^*, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_N} + B_{stab}((T^*,\boldsymbol{\beta}),\boldsymbol{w}) \\
&- < \{\nabla T^*\cdot\boldsymbol{d} + 1/2\boldsymbol{d}^t\mathcal{H}(T^*)\boldsymbol{d}\}_{\tilde\Gamma}, [\![\boldsymbol{w}]\!]_{\tilde\Gamma}\cdot\tilde{\boldsymbol{n}} >_{\tilde\Gamma} - < [\![\nabla T\cdot\boldsymbol{d} + 1/2\boldsymbol{d}^t\mathcal{H}(T^*)\boldsymbol{d}]\!]_{\tilde\Gamma}, \{\boldsymbol{w}\}_{\tilde\Gamma}\cdot\tilde{\boldsymbol{n}} >_{\tilde\Gamma} \\
&= - < a_T, [\![\boldsymbol{w}]\!]_{\tilde\Gamma}\cdot\tilde{\boldsymbol{n}} >_{\tilde\Gamma} - < j_T, \{\boldsymbol{w}\}_{\tilde\Gamma}\cdot\tilde{\boldsymbol{n}} >_{\tilde\Gamma} + L_{stab}(\boldsymbol{w}), \\
&A.1.b)\ (\rho c\partial_t T^*, q^*)_\Omega + (\nabla\cdot\boldsymbol{\beta}, q)_\Omega - < \boldsymbol{\beta}\cdot\boldsymbol{n} - h_N, q >_{\Gamma_N} + A_{stab}((T^*,\boldsymbol{\beta}),q^*) = (f,q^*)_\Omega.
\end{aligned} \tag{A.1}$$

To the enriched weak formulation (A.1) we also add the following Nitsche penalty terms:

$$\begin{aligned}
&\tilde N_{A_2}(T^*,q^*) := \frac{\alpha}{h} < [\![T^* + \nabla T^*\cdot\boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t\mathcal{H}(T^*)\boldsymbol{d}]\!]_{\tilde\Gamma}, [\![q^* + \nabla q^*\cdot\boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t\mathcal{H}(q^*)\boldsymbol{d}]\!]_{\tilde\Gamma} >_{\tilde\Gamma} \\
&+ \frac{\alpha}{h} < \{T^* + \nabla T^*\cdot\boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t\mathcal{H}(T^*)\boldsymbol{d}\}_{\tilde\Gamma}, \{q^* + \nabla q^*\cdot\boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t\mathcal{H}(q^*)\boldsymbol{d}\}_{\tilde\Gamma} >_{\tilde\Gamma}, \\
&\tilde N_{L_2}(q^*) := \frac{\alpha}{h} < j_T, [\![q^* + \nabla q^*\cdot\boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t\mathcal{H}(q^*)\boldsymbol{d}]\!]_{\tilde\Gamma} >_{\tilde\Gamma} + \frac{\alpha}{h} < a_T, \{q^* + \nabla q^*\cdot\boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t\mathcal{H}(q^*)\boldsymbol{d}\}_{\tilde\Gamma} >_{\tilde\Gamma}.
\end{aligned} \tag{A.2}$$

Following remark 3 the Nitsche penalty terms A.2 and the weak formulation A.1 can be rewritten using the information known from the mixed form. The definition (59) adds new terms in the formulation and then requires to implement them. Nevertheless, we can reduce the number of additional terms by using the following form instead :

Finding $(T^*, \boldsymbol{\beta}) \in \boldsymbol{Q}^{2,1}(\,]0,t[\,,\Omega)$, s.t. $\forall (q^*, \boldsymbol{w}) \in \boldsymbol{Q}^{2,1}(\Omega)$ we have

$$\begin{aligned}
&A.1.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T^*, \nabla\cdot\boldsymbol{w})_\Omega + < T_D, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_D} + < T^*, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_N} + B_{stab}((T^*,\boldsymbol{\beta}),\boldsymbol{w}) \\
&+ < \{\lambda^{-1}\boldsymbol{\beta}\cdot\boldsymbol{d} + 1/2\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{\beta})\boldsymbol{d}\}_{\tilde\Gamma}, [\![\boldsymbol{w}]\!]_{\tilde\Gamma}\cdot\tilde{\boldsymbol{n}} >_{\tilde\Gamma} + < [\![\lambda^{-1}\boldsymbol{\beta}\cdot\boldsymbol{d} + 1/2\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{\beta})\boldsymbol{d}]\!]_{\tilde\Gamma}, \{\boldsymbol{w}\}_{\tilde\Gamma}\cdot\tilde{\boldsymbol{n}} >_{\tilde\Gamma} \\
&= - < a_T, [\![\boldsymbol{w}]\!]_{\tilde\Gamma}\cdot\tilde{\boldsymbol{n}} >_{\tilde\Gamma} - < j_T, \{\boldsymbol{w}\}_{\tilde\Gamma}\cdot\tilde{\boldsymbol{n}} >_{\tilde\Gamma} + L_{stab}(\boldsymbol{w}), \\
&A.1.b)\ (\rho c\partial_t T^*, q^*)_\Omega + (\nabla\cdot\boldsymbol{\beta}, q)_\Omega - < \boldsymbol{\beta}\cdot\boldsymbol{n} - h_N, q >_{\Gamma_N} + A_{stab}((T^*,\boldsymbol{\beta}),q^*) = (f,q^*)_\Omega,
\end{aligned} \tag{A.3}$$

and

$$\begin{aligned}
&\tilde N_{A_2}(T^*,q^*) := \frac{\alpha}{h} < [\![T^* - \lambda^{-1}\boldsymbol{\beta}\cdot\boldsymbol{d} - 1/2\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{\beta})\boldsymbol{d}]\!]_{\tilde\Gamma}, [\![q^* - \lambda^{-1}\boldsymbol{w}\cdot\boldsymbol{d} - 1/2\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{w})\boldsymbol{d}]\!]_{\tilde\Gamma} >_{\tilde\Gamma} \\
&+ \frac{\alpha}{h} < \{T^* - \lambda^{-1}\boldsymbol{\beta}\cdot\boldsymbol{d} - 1/2\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{\beta})\boldsymbol{d}\}_{\tilde\Gamma}, \{q^* - \lambda^{-1}\boldsymbol{w}\cdot\boldsymbol{d} - \frac{1}{2}\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{w})\boldsymbol{d}\}_{\tilde\Gamma} >_{\tilde\Gamma} \\
&\tilde N_{L_2}(q^*) := \frac{\alpha}{h} < j_T, [\![q^* - \lambda^{-1}\boldsymbol{w}\cdot\boldsymbol{d} - 1/2\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{w})\boldsymbol{d}]\!]_{\tilde\Gamma} >_{\tilde\Gamma} \\
&+ \frac{\alpha}{h} < a_T, \{q^* - \lambda^{-1}\boldsymbol{w}\cdot\boldsymbol{d} - \frac{1}{2}\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{w})\boldsymbol{d}\}_{\tilde\Gamma} >_{\tilde\Gamma}.
\end{aligned} \tag{A.4}$$

References

[1] C. S. Peskin, Flow patterns around heart valves: A numerical method, Journal of Computational Physics 10 (2) (1972) 252–271. doi:https://doi.org/10.1016/0021-9991(72)90065-4.
URL https://www.sciencedirect.com/science/article/pii/0021999172900654

[2] T. Ye, R. Mittal, H. Udaykumar, W. Shyy, An accurate cartesian grid method for viscous incompressible flows with complex immersed boundaries, Journal of Computational Physics 156 (2) (1999) 209–240. doi:https://doi.org/10.1006/jcph.1999.6356.
URL https://www.sciencedirect.com/science/article/pii/S0021999199963568

[3] F. Morency, H. Beaugendre, F. Gallizio, Aerodynamic force evaluation for ice shedding phenomenon using vortex in cell scheme, penalisation and level set approaches, International Journal of Computational Fluid Dynamics 26 (9-10) (2012) 435–450. doi:10.1080/10618562.2012.739683.
URL https://doi.org/10.1080/10618562.2012.739683

[4] M. Bergmann, A. Iollo, Modeling and simulation of fish-like swimming, Journal of Computational Physics 230 (2) (2011) 329–348. doi:https://doi.org/10.1016/j.jcp.2010.09.017.
URL https://www.sciencedirect.com/science/article/pii/S0021999110005115

[5] A. Gilmanov, F. Sotiropoulos, E. Balaras, A general reconstruction algorithm for simulating flows with complex 3d immersed boundaries on cartesian grids, Journal of Computational Physics 191 (2) (2003) 660–669. doi:https://doi.org/10.1016/S0021-9991(03)00321-8.
URL https://www.sciencedirect.com/science/article/pii/S0021999103003218

[6] R. Mittal, H. Dong, M. Bozkurttas, F. Najjar, A. Vargas, A. von Loebbecke, A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries, Journal of Computational Physics 227 (10) (2008) 4825–4852. doi:https://doi.org/10.1016/j.jcp.2008.01.028.
URL https://www.sciencedirect.com/science/article/pii/S0021999108000235

[7] G. Georges, J. Breil, P.-H. Maire, A 3d gcl compatible cell-centered lagrangian scheme for solving gas dynamics equations, Journal of Computational Physics 305 (2016) 921–941. doi:https://doi.org/10.1016/j.jcp.2015.10.040.
URL https://www.sciencedirect.com/science/article/pii/S0021999115007172

[8] E. Caramana, D. Burton, M. Shashkov, P. Whalen, The construction of compatible hydrodynamics algorithms utilizing conservation of total energy, Journal of Computational Physics 146 (1) (1998) 227–262. doi:https://doi.org/10.1006/jcph.1998.6029.
URL https://www.sciencedirect.com/science/article/pii/S0021999198960296

[9] R. Loubère, P.-H. Maire, P. Váchal, 3d staggered lagrangian hydrodynamics scheme with cell-centered riemann solver-based artificial viscosity, International Journal for Numerical Methods in Fluids 72 (1) (2013) 22–42. doi:https://doi.org/10.1002/fld.3730.
URL https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.3730

[10] P.-H. Maire, R. Abgrall, J. Breil, J. Ovadia, A cell-centered lagrangian scheme for two-dimensional compressible flow problems, SIAM Journal on Scientific Computing 29 (4) (2007) 1781–1824. doi:10.1137/050633019.
URL https://doi.org/10.1137/050633019

[11] C. Hirt, A. Amsden, J. Cook, An arbitrary lagrangian-eulerian computing method for all flow speeds, Journal of Computational Physics 14 (3) (1974) 227–253. doi:https://doi.org/10.1016/0021-9991(74)90051-5.
URL https://www.sciencedirect.com/science/article/pii/0021999174900515

[12] A. Johnson, T. Tezduyar, Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces, Computer Methods in Applied Mechanics and Engineering 119 (1) (1994) 73–94. doi:https://doi.org/10.1016/0045-7825(94)00077-8.
URL https://www.sciencedirect.com/science/article/pii/0045782594000778

[13] A. Johnson, T. Tezduyar, Simulation of multiple spheres falling in a liquid-filled tube, Computer Methods in Applied Mechanics and Engineering 134 (3) (1996) 351–373. doi:https://doi.org/10.1016/0045-7825(95)00988-4.
URL https://www.sciencedirect.com/science/article/pii/0045782595009884

[14] C. Farhat, P. Geuzaine, C. Grandmont, The discrete geometric conservation law and the nonlinear stability of ale schemes for the solution of flow problems on moving grids, Journal of Computational Physics 174 (2) (2001) 669–694. doi:https://doi.org/10.1006/jcph.2001.6932.
URL https://www.sciencedirect.com/science/article/pii/S0021999101969323

[15] A. Masud, M. Bhanabhagvanwala, R. A. Khurram, An adaptive mesh rezoning scheme for moving boundary flows and fluid–structure interaction, Computers & Fluids 36 (1) (2007) 77–91, challenges and Advances in Flow Simulation and Modeling. doi:https://doi.org/10.1016/j.compfluid.2005.07.013.
URL https://www.sciencedirect.com/science/article/pii/S0045793005001301

[16] A. Jendoubi, J. Deteix, A. Fortin, A simple mesh-update procedure for fluid–structure interaction problems, Computers & Structures 169 (2016) 13–23. doi:https://doi.org/10.1016/j.compstruc.2016.02.015.
URL https://www.sciencedirect.com/science/article/pii/S0045794916300396

[17] A. Sarthou, S. Vincent, P. Angot, J.-P. Caltagirone, The algebraic immersed interface and boundary method for elliptic equations with jump conditions, Open Journal of fluid Dynamics 10 (3) (2020) 239 – 269. doi:10.4236/ojfd.2020.103015.
URL https://hal.archives-ouvertes.fr/hal-02921726

[18] C. Tu, C. S. Peskin, Stability and instability in the computation of flows with moving immersed boundaries: A comparison of three methods, SIAM Journal on Scientific and Statistical Computing 13 (6) (1992) 1361–1376. doi:10.1137/0913077.
URL https://doi.org/10.1137/0913077

[19] W.-P. Breugem, A second-order accurate immersed boundary method for fully resolved simulations of particle-laden flows, Journal of Computational Physics 231 (13) (2012) 4469–4498. doi:https://doi.org/10.1016/j.jcp.2012.02.026.
URL https://www.sciencedirect.com/science/article/pii/S0021999112001374

[20] R. J. Leveque, Z. Li, The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, SIAM Journal on Numerical Analysis 31 (4) (1994) 1019–1044. doi:10.1137/0731054.
URL https://www.jstor.org/stable/2158113

[21] E. Arquis, J. Caltagirone, Sur les conditions hydrodynamiques au voisinage d'une interface milieu fluide-milieux poreux: application la convectionnaturelle, C.R. Acad. Sci. Paris II 299 (1984) 1–4.

[22] P. Angot, C.-H. Bruneau, P. Fabrie, A penalization method to take into account obstacles in incompressible viscous flows, Numerische Mathematik 81 (4) (1999) 497–520. doi:10.1007/s002110050401.
URL https://doi.org/10.1007/s002110050401

[23] R. Abgrall, H. Beaugendre, C. Dobrzynski, An immersed boundary method using unstructured anisotropic mesh adaptation combined with level-sets and penalization techniques, Journal of Computational Physics 257 (2014) 83–101. doi:https://doi.org/10.1016/j.jcp.2013.08.052.
URL https://www.sciencedirect.com/science/article/pii/S0021999113005962

[24] R. Mittal, G. Iaccarino, Immersed boundary methods, Annual Review of Fluid Mechanics 37 (1) (2005) 239–261. doi:10.1146/annurev.fluid.37.061903.175743.
URL https://doi.org/10.1146/annurev.fluid.37.061903.175743

[25] F. Sotiropoulos, X. Yang, Immersed boundary methods for simulating fluid–structure interaction, Progress in Aerospace Sciences 65 (2014) 1–21. doi:https://doi.org/10.1016/j.paerosci.2013.09.003.
URL https://www.sciencedirect.com/science/article/pii/S0376042113000870

[26] B. E. Griffith, N. A. Patankar, Immersed methods for fluid–structure interaction, Annual Review of Fluid Mechanics 52 (1) (2020) 421–448. doi:10.1146/annurev-fluid-010719-060228.
URL https://doi.org/10.1146/annurev-fluid-010719-060228

[27] R. P. Fedkiw, T. Aslam, B. Merriman, S. Osher, A non-oscillatory eulerian approach to interfaces in multimaterial flows (the ghost fluid method), Journal of Computational Physics 152 (2) (1999) 457–492. doi:https://doi.org/10.1006/jcph.1999.6236.
URL https://www.sciencedirect.com/science/article/pii/S0021999199962368

[28] S. Majumdar, G. Iaccarion, P. Durbin, Rans solver with adaptive structured boundary non conforming grids (2001).
URL https://web.stanford.edu/group/ctr/ResBriefs01/sekhar.pdf

[29] Y. Gorsse, A. Iollo, H. Telib, L. Weynans, A simple second order cartesian scheme for compressible euler flows, Journal of Computational Physics 231 (23) (2012) 7780–7794. doi:https://doi.org/10.1016/j.jcp.2012.07.014.
URL https://www.sciencedirect.com/science/article/pii/S0021999112003877

[30] A. Main, C. Farhat, A second-order time-accurate implicit finite volume method with exact two-phase riemann problems for compressible multi-phase fluid and fluid–structure problems, Journal of Computational Physics 258 (2014) 613–633. doi:https://doi.org/10.1016/j.jcp.2013.11.001.
URL https://www.sciencedirect.com/science/article/pii/S002199911300747X

[31] D. Z. Huang, D. De Santis, C. Farhat, A family of position- and orientation-independent embedded boundary methods for viscous flow and fluid–structure interaction problems, Journal of Computational Physics 365 (2018) 74–104. doi:https://doi.org/10.1016/j.jcp.2018.03.028.
URL https://www.sciencedirect.com/science/article/pii/S0021999118301888

[32] A. Hansbo, 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) (2002) 5537–5552. doi:https://doi.org/10.1016/S0045-7825(02)00524-8.
URL https://www.sciencedirect.com/science/article/pii/S0045782502005248

[33] H. M. Mourad, J. Dolbow, I. Harari, A bubble-stabilized finite element method for dirichlet constraints on embedded interfaces, International Journal for Numerical Methods in Engineering 69 (4) (2007) 772–793. doi:https://doi.org/10.1002/nme.1788.
URL https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.1788

[34] J. Dolbow, I. Harari, An efficient finite element method for embedded interface problems, International Journal for Numerical Methods in Engineering 78 (2) (2009) 229–252. doi:https://doi.org/10.1002/nme.2486.
URL https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.2486

[35] M. Fournié, A. Lozinski, Stability and optimal convergence of unfitted extended finite element methods with lagrange multipliers for the stokes equations, in: S. P. A. Bordas, E. Burman, M. G. Larson, M. A. Olshanskii (Eds.), Geometrically Unfitted Finite Element Methods and Applications, Springer International Publishing, Cham, 2017, pp. 143–182.

[36] J. Parvizian, A. Düster, E. Rank, Finite cell method, Computational Mechanics 41 (1) (2007) 121–133. doi:10.1007/s00466-007-0173-y.
URL https://link.springer.com/article/10.1007/s00466-007-0173-y

[37] M. Kirkpatrick, S. Armfield, J. Kent, A representation of curved boundaries for the solution of the navier–stokes equations on a staggered three-dimensional cartesian grid, Journal of Computational Physics 184 (1) (2003) 1–36. doi:https://doi.org/10.1016/S0021-9991(02)00013-X.
URL https://www.sciencedirect.com/science/article/pii/S002199910200013X

[38] E. Burman, Ghost penalty, Comptes Rendus Mathematique 348 (21) (2010) 1217–1220. doi:https://doi.org/10.1016/j.crma.2010.10.006.
URL https://www.sciencedirect.com/science/article/pii/S1631073X10002827

[39] Burman, Erik, Hansbo, Peter, Fictitious domain methods using cut elements: Iii. a stabilized nitsche method for stokes´ problem, ESAIM: M2AN 48 (3) (2014) 859–874. doi:10.1051/m2an/2013123.
URL https://doi.org/10.1051/m2an/2013123

[40] B. Schott, U. Rasthofer, V. Gravemeier, W. A. Wall, A face-oriented stabilized nitsche-type extended variational multiscale method for incompressible two-phase flow, International Journal for Numerical Methods in Engineering 104 (7) (2015) 721–748. doi:https://doi.org/10.1002/nme.4789.
URL https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.4789

[41] M. Duprez, A. Lozinski, ϕ-fem: A finite element method on domains defined by level-sets, SIAM Journal on Numerical Analysis 58 (2) (2020) 1008–1028. doi:10.1137/19M1248947.
URL https://doi.org/10.1137/19M1248947

[42] M. Duprez, V. Lleras, A. Lozinski, A new ϕ-fem approach for problems with natural boundary conditions, preprint (2020).
URL https://hal.archives-ouvertes.fr/hal-02521042v2

[43] S. Badia, F. Verdugo, A. F. Martín, The aggregated unfitted finite element method for elliptic problems, Computer Methods in Applied Mechanics and Engineering 336 (2018) 533–553. doi:https://doi.org/10.1016/j.cma.2018.03.022.
URL https://www.sciencedirect.com/science/article/pii/S0045782518301476

[44] E. Neiva, S. Badia, Robust and scalable h-adaptive aggregated unfitted finite elements for interface elliptic problems, Computer Methods in Applied Mechanics and Engineering 380 (2021) 113769. doi:https://doi.org/10.1016/j.cma.2021.113769.
URL https://www.sciencedirect.com/science/article/pii/S0045782521001055

[45] A. Main, G. Scovazzi, The shifted boundary method for embedded domain computations. part i : Poisson and stokes problems, Journal of Computational Physics 372 (10 2017). doi:10.1016/j.jcp.2017.10.026.
URL https://www.sciencedirect.com/science/article/pii/S0021999117307799

[46] V. Alexiades, A. D. Solomon, Mathematical Modeling of Melting and Freezing Processes, 2018. doi:10.1201/9780203749449.

[47] K. Li, N. Atallah, G. A. Main, G. Scovazzi, The shifted interface method: A flexible approach to embedded interface computations, International Journal for Numerical Methods in Engineering 121 (09 2019). doi:10.1002/nme.6231.
URL https://onlinelibrary.wiley.com/doi/full/10.1002/nme.6231

[48] A. Main, G. Scovazzi, The shifted boundary method for embedded domain computations. part ii: Linear advection–diffusion and incompressible navier–stokes equations, Journal of Computational Physics 372 (2018) 996–1026. doi:https://doi.org/10.1016/j.jcp.2018.01.023.
URL https://www.sciencedirect.com/science/article/pii/S0021999118300330

[49] O. Colomés, A. Main, L. Nouveau, G. Scovazzi, A weighted shifted boundary method for free surface flow problems, Journal of Computational Physics 424 (2021) 109837. doi:https://doi.org/10.1016/j.jcp.2020.109837.
URL https://www.sciencedirect.com/science/article/pii/S0021999120306112

[50] N. M. Atallah, C. Canuto, G. Scovazzi, The shifted boundary method for solid mechanics, International Journal for Numerical Methods in Engineering (2021). doi:https://doi.org/10.1002/nme.6779.
URL https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.6779

[51] N. M. Atallah, C. Canuto, G. Scovazzi, The second-generation shifted boundary method and its numerical analysis, Computer Methods in Applied Mechanics and Engineering 372 (2020) 113341. doi:https://doi.org/10.1016/j.cma.2020.113341.
URL https://www.sciencedirect.com/science/article/pii/S0045782520305260

[52] N. M. Atallah, C. Canuto, G. Scovazzi, Analysis of the shifted boundary method for the poisson problem in domains with corners, Math. Comp. 90 (2021) 2031–2069. doi:https://doi.org/10.1090/mcom/3641.
URL https://arxiv.org/abs/2006.00872

[53] H. Nishikawa, Robust and accurate viscous discretization via upwind scheme – i: Basic principle, Computers & Fluids 49 (1) (2011) 62–86. doi:https://doi.org/10.1016/j.compfluid.2011.04.014.
URL https://www.sciencedirect.com/science/article/pii/S0045793011001484

[54] A. Mazaheri, H. Nishikawa, Improved second-order hyperbolic residual-distribution scheme and its extension to third-order on arbitrary triangular grids, Journal of Computational Physics 300 (2015) 455–491. doi:https://doi.org/10.1016/j.jcp.2015.07.054.
URL https://www.sciencedirect.com/science/article/pii/S0021999115005070

[55] L. Nouveau, M. Ricchiuto, G. Scovazzi, High-Order Gradients with the Shifted Boundary Method: An Embedded Enriched Mixed Formulation for Elliptic PDEs, Journal of Computational Physics 398 (08 2019). doi:10.1016/j.jcp.2019.108898.
URL https://www.sciencedirect.com/science/article/pii/S0021999119305960

[56] A. Masud, T. Hughes, A stabilized mixed finite element method for darcy flow, Computer Methods in Applied Mechanics and Engineering 191 (2002) 4341–4370. doi:10.1016/S0045-7825(02)00371-7.
URL https://www.sciencedirect.com/science/article/pii/S0045782502003717

[57] T. J. Hughes, A. Masud, J. Wan, A stabilized mixed discontinuous galerkin method for darcy flow, Computer Methods in Applied Mechanics and Engineering 195 (25) (2006) 3347–3381, discontinuous Galerkin Methods. doi:https://doi.org/10.1016/j.cma.2005.06.018.
URL https://www.sciencedirect.com/science/article/pii/S0045782505002732

[58] F. Brezzi, T. Hughes, L. Marini, A. Masud, Mixed discontinuous galerkin methods for darcy flow, Journal of Scientific Computing 22-23 (2005) 119–145. doi:10.1007/s10915-004-4150-8.
URL https://link.springer.com/article/10.1007/s10915-004-4150-8

[59] S. Joodat, K. Nakshatrala, R. Ballarini, Modeling flow in porous media with double porosity/permeability: A stabilized mixed formulation, error analysis, and numerical solutions, Computer Methods in Applied Mechanics and Engineering 337 (2018) 632–676. doi:https://doi.org/10.1016/j.cma.2018.04.004.
URL https://www.sciencedirect.com/science/article/pii/S0045782518301749

[60] M. Joshaghani, S. Joodat, K. Nakshatrala, A stabilized mixed discontinuous galerkin formulation for double porosity/permeability model, Computer Methods in Applied Mechanics and Engineering 352 (2019) 508–560. doi:https://doi.org/10.1016/j.cma.2019.04.010.
URL https://www.sciencedirect.com/science/article/pii/S0045782519302075

[61] A. Costa-Solé, E. Ruiz-Gironés, J. Sarrate, High-order hybridizable discontinuous galerkin formulation for one-phase flow through porous media, Journal of Scientific Computing 87 (1) (mar 2021). doi:10.1007/s10915-021-01436-9.
URL https://doi.org/10.1007%2Fs10915-021-01436-9

[62] J. C. C. 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 (1971) 9–15.