Modeling of the unsteady Stefan Problem using the Shifted Boundary Method : Application to de-icing systems

Tiffanie Carlier

THÈSE PRÉSENTÉE

POUR OBTENIR LE GRADE DE

DOCTEURE DE L'UNIVERSITÉ DE BORDEAUX

ÉCOLE DOCTORALE MATHÉMATIQUES ET INFORMATIQUE

Mathématiques appliquées et calcul scientifique

Par **Tiffanie CARLIER**

Modélisation du problème de Stefan instationnaire par la méthode des frontières décalées: application au dégivrage

Sous la direction de : **Mathieu COLIN**
Co-directrice : **Héloïse BEAUGENDRE**

Soutenue le 15/12/2023

Membres du jury :

Stéphane CLAIN	Full professor	Coimbra University	Rapporteur
Alexei LOZINSKI	Professeur	Université de Franche-Comté	Rapporteur
Mathieu COLIN	Professeur	Bordeaux INP, Université de Bordeaux	Directeur de thèse
Heloïse BEAUGENDRE	Professeure	Bordeaux INP, Université de Bordeaux	Co-Directrice de thèse
Clair POIGNARD	Directeur de recherche	INRIA Bordeaux Sud-Ouest	Président du jury
Brian T. HELENBROOK	Full professor	Clarkson University	Examinateur
Alireza MAZAHERI	Chargé de recherche	NASA Langley Research Center	Examinateur
Lisl WEYNANS	Maître de conférences	Université de Bordeaux	Examinatrice

Modélisation du problème de Stefan instationnaire par la méthode des frontières décalées: application au dégivrage

Résumé : Ce manuscrit présente une version enrichie de la méthode des frontières décalées (SBM) appliquée aux simulations par éléments finis sur des problèmes à frontières libres et mobiles. Le modèle étudié est le problème de Stefan, un modèle académique complexe pour la simulation de problèmes avec fronts de fusion. Plus précisément, le problème physique étudié dans ce manuscrit est le problème de fusion intervenant lors du dégivrage des avions. L'objectif est de développer des systèmes de dégivrage électrique plus efficaces, un défi majeur pour l'industrie aéronautique. Dans le cadre du développement de systèmes de dégivrage à résistance thermique, la modélisation par méthode à frontières immergées est une alternative qui apporte de nombreux avantages. Dans cette catégorie de méthodes, la méthode aux frontières décalées permet l'utilisation d'un seul maillage non conforme à la géomètrie où l'interface physique est remplacée par une interface numérique dont la définition dépend des éléments du maillage traversé par la véritable interface. La version d'ordre élevé proposée consiste à l'enrichissement de la formulation faible discrétisée du problème dans sa formulation mixte, permettant une précision d'ordre deux en espace sur la variable primale et son gradient, en temps et sur la position d'interface. Le lien entre les fonctions tests $\mathbb{P}^1$ et $\mathbb{P}^2$, ainsi que les développements de Taylor sur les conditions d'interface, du front physique à l'interface numérique, permettent d'obtenir une méthode précise sur un maillage non conforme. Sans précautions, l'utilisation de la méthode des frontières décalées avec fronts mobiles peut conduire à des instabilités qui peuvent se propager ou s'amplifier. Pour comprendre ces phénomènes, une étude de stabilité est effectuée sur une linéarisation du modèle autour d'un état stationnaire. Le modèle linéarisé est utilisé pour obtenir une relation de dispersion caractérisant les modes de dispersion du modèle en fonction du choix de conditions aux bords. Des tests numériques sont effectués pour démontrer la performance, la robustesse, la précision de la méthode ainsi que la stabilité du modèle par rapport aux perturbations de l'interface et sur le champ de température.

Mots-clés : Méthode des Frontières Décalées, Interface Mobile, Problème de Stefan, Enrichissement de Température, Analyse de Stabilité, Méthode des Éléments Finis

---

Unité de recherche
Équipe CARDAMOM, Centre INRIA de l'Université de Bordeaux, France.

Modeling of the unsteady Stefan Problem using the Shifted Boundary Method : Application to de-icing systems

Abstract: This manuscript introduces an expanded version of the Shifted Boundary Method (SBM) applied to a Finite Element Method designed to address problems with free and moving boundaries. The method is applied to the Stefan model, an academic challenge employed to simulate phase transitions. More precisely, the physical phenomena studied in this manuscript is the melting problem arising from the de-icing of aircraft. The objective is to get a better comprehension of these phenomena and to develop more efficient electric de-icing systems using thermal energy, a major challenge for the aeronautic industry. One of the main difficulties in the numerical simulation of the Stefan model is the treatment of the moving interface, the phase-change front between solid and liquid water. The moving boundary is defined using the Shifted Boundary Method (SBM), an embedded method that enables the use of non-body-fitted meshes, preventing the need for remeshing steps at each displacement of the interface and thus avoiding any problems that could arise from sharp geometries. The expanded method consists of improving the finite weak formulation to be at least second-order accurate by solving a mixed problem where both variables, the primal variable, and the flux, are solved. The link between the test functions $\mathbb{P}^1$ and $\mathbb{P}^2$ and the proper utilization of Taylor expansions allows us to achieve a fully second-order scheme in time, space, and for the interface position. Regarding the model study, the heat flux is responsible for moving the phase-change front, and the motion of the interface is derived from the Stefan condition. Since the interface is moving, some mesh nodes can switch from one part of the domain to another. The discontinuous aspect of the thermal flux at the interface makes the problem particularly delicate and can become a source of instabilities that can impact the solution on the entire domain. To understand this, a linear stability analysis of the numerical scheme and of the Stefan model is performed. The linear stability analysis provides an understanding of the behavior of the model depending on the choice made for its boundary conditions. The linearization is used to define a dispersion relation, which tells us about the stability or the dispersive nature of the wave components of the equations and characterizes the dynamics of spatially oscillating modes. Numerical tests are performed to demonstrate the performance, robustness, and accuracy of the method, as well as its stability regarding perturbations in both the primal variable and the interface location.

Keywords: Shifted Boundary Method, Moving Interface, Stefan Model, Temperature Enrichment, Stability Analysis, Finite Element Method

---

Research Laboratory
CARDAMOM Team, Centre INRIA de l'Université de Bordeaux, France.

Acknowledgements

This work has been made possible thanks to the people who supported me throughout my research journey. I want to begin by expressing my gratitude to the members of my committee who generously dedicated their time to allow me to present my work; without them, this would not have been possible. I would like to extend my thanks to Stéphane Clain, Alexei Lozinski, Brian Helenbrook, Alireza Mazaheri, Lisl Weynans, and Clair Poignard. I also want to give a special mention to my two rapporteurs who agreed to review this manuscript, into which I poured a lot of energy and effort. I will always be grateful to both of you for the time you devoted to me and my work.

I also want to thank my two PhD advisors, Heloïse Beaugendre and Mathieu Colin, who entrusted me from the start. I still remember myself at the beginning of these three years, and you helped me grow into a better scientist. My love for science is bigger than ever, thanks to both of you. You allowed me to create my own path with this work, and I am very grateful for the support you provided. I have learned a lot from you, and I will never forget that.

I would like to extend my gratitude to Mario Ricchiuto for entrusting me and introducing me to his friend and colleague, Brian Helenbrook, which made my research stay in the United States possible. Brian, you have been a real pleasure to work with, and I am keeping so many good memories from Clarkson University and the work we presented in Cannes. I will always hold you in high esteem and I am very grateful for all the knowledge you shared with me.

I would also like to mention my unofficial PhD advisor, Léo Nouveau, who was of great help at the beginning of this project when I felt lost in the work I was doing. You were always available to guide me through it. I wish you all the best in your next adventure.

I also want to mention the Cardamom team at INRIA Bordeaux and the people I worked with in that open space. While we may no longer share the same workspace, I wish all the best to each of you. Thank you to Francesco, Mirco, Sixtine, and Manon for all the times you listened to me complain, you are all dear friends to me now.

Finally, people in my life have been a huge source of support, especially my family. I would never have been able to get to where I am today without all of your encouragement. Thank you to all of you. Nevertheless, I want to give some special words for some of them. To my mother, who, as far as I can remember, has always motivated me to go in the direction I wanted to go, even when I thought that was beyond my reach. You inspire me a lot, and this work is also thanks to you. To my dad, who always believed that I would achieve great things, and who encouraged me to try and never let anyone tell me I could not do something. Thank you for your never-ending support. To my sister, who has been there from the beginning and who told me I could do anything if I really wanted to. I wish one day I can repay you with the same support. Finally, to my special person who saw me struggling more than anyone else. You allowed me to be me when I needed it the most, and all the support you gave me allowed me

to give my all when I thought I could not. Thank you for forever and always.

I met many people on my PhD journey, and this experience has definitely changed me for the better. However, without the financial support I received for my work, none of this would have been possible. I want to express my gratitude to the University of Bordeaux for providing me three years of research funding. I also thank Inria Bordeaux for their financial support for conferences and for the additional three months I needed to finalize my work.

Finally, I would like to extend my thanks to the Fulbright program, which chose me and my work and provided me the funding for the six months of research I conducted at Clarkson University. The foundation has my deepest gratitude for allowing me to have this amazing experience, and the entire commission was of tremendous help during my stay.

In conclusion, I will never forget any of you.

Résumé en français

1 Contexte général et application

L'une des menaces les plus importantes pour la sécurité d'un avion en vol est la **formation de givre** sur les parois de l'appareil ou sur ses composantes mécaniques. La formation de ce givre est due à la solidification de l'eau en surfusion présente dans l'atmosphère, une eau liquide à une température inférieure à la température de fusion. Cette eau est présente dans certaines zones de l'atmosphère et se solidifie sous conditions givrantes (ex: condition de température vérifiée sous le point de congélation) une fois en contact avec la surface de l'appareil [1]. Cet état intermédiaire particulier entre liquide et solide est dû à la pureté de l'eau présente dans l'atmosphère et à sa structure moléculaire, incapable d'achever le processus de nucléation pour devenir solide. Pour assurer la sécurité en vol, les avions sont munis de systèmes de protection contre le givre. Deux paradigmes existent:

1. Empêcher complétement la formation de givre en utilisant un système d'anti-givrage,
2. Autoriser la formation d'une fine couche de givre ensuite éliminée par un système de dégivrage.

Dans le contexte d'étude de système de dégivrage, la conception de systèmes électriques, basés sur l'utilisation de résistances chauffantes, permet un meilleur contrôle de la distribution des sources de chaleur et permet de réduire l'empreinte énergétique de ces dispositifs. L'optimisation de systèmes de dégivrage, pour laquelle le recours à la simulation numérique est essentielle, est un domaine de recherche actif pour les entreprises du secteur aéronautique. Il s'agit d'un problème multi-physique instationnaire très complexe qui fait intervenir la présence de fronts mobiles, comme le front de fusion par exemple, ce qui pose des difficultés majeures pour la résolution numérique.

Dans ce manuscrit de thèse, les systèmes thermiques sont étudiés d'un point de vue académique en explorant une méthode de **frontières immergées récente** : la **méthode des frontières décalées** introduite par Main et Scovazzi dans [2, 3]. Le modèle mathématique étudié est le **modèle de Stefan** [4], qui traite du changement de phase glace/eau liquide et du suivi du front de fusion mobile. Les objectifs de cette thèse sont d'étendre la méthode des frontières décalées à la simulation de problèmes avec interface mobile et d'étudier le problème de Stefan en formulation mixte de manière à proposer une simulation d'ordre élevé à la fois pour la température et son gradient. La méthode de Nitsche est utilisée pour permettre l'imposition des conditions aux limites dans la formulation faible. La formulation est également stabilisée par l'ajout de termes de stabilisation, en utilisant approche standard, voir par exemple [5, 6, 7, 8, 9]. Le travail réalisé sur le suivi explicite de l'interface, la précision de la méthode et la stabilité du front de fusion sont détaillés dans ce manuscrit de thèse.

2 Challenge numérique: une extension de la méthode des frontières décalées

Les simulations numériques permettent de prendre en compte un grand nombre de scénarios possibles de manière rapide et moins coûteuse que des tests en soufflerie [10] et sans les risques associés aux essais en vol. Néanmoins, l'une des principales difficultés de ces simulations numériques, en givrage/dégivrage, réside dans la conception du maillage de la géométrie qui va présenter des fronts mobiles soit par l'apparition de formes de glace irrégulières ou soit par l'apparition d'un front de fusion. Les méthodes dont le maillage épouse la géométrie nécessitent de l'adaptation de maillage et du remaillage (méthodes conformes) pour faire face aux déplacements de frontières mobiles et permettre une imposition correcte des conditions de bords. Ces méthodes ne sont pas complètement automatisées à ce jour. Dans la littérature, il existe également des méthodes aux frontières immergées pour lesquelles le maillage est fixe et ne suit pas la géométrie. Tout l'enjeu de ces méthodes est de proposer une prise en compte suffisamment précise des conditions aux bords. Les différentes méthodes de frontière immergée diffèrent par leur traitement des conditions de bord ainsi que par le traitement des éléments intersectés par la frontière physique.

Dans ce travail, la méthode de frontières immergées utilisée est **la méthode des frontières décalées** (Shifted Boundary Method - SBM) [2, 3]. L'idée de base est d'imposer faiblement les conditions aux limites en utilisant la méthode de Nitsche [11] via des développements de Taylor sur une frontière de substitution, définie par des arêtes/faces du maillage, issue d'éléments intersectés par la véritable interface. Le **modèle de Stefan**, qui regroupe les problèmes à frontière libre (où la position de la frontière est inconnue) et mobile, est le modèle de support de cette étude. Il a été nommé d'après le physicien Joseph Stefan (1835-1893) et son travail sur la formation de la glace dans les mers polaires [12]. L'équation de la chaleur est résolue dans chacune des phases de l'eau. La condition de Stefan [4] relie la vitesse locale de l'interface à la discontinuité du flux de chaleur. De plus, le modèle de Stefan est résolu dans sa forme mixte, c'est à dire que le gradient de température est aussi défini comme une inconnue du problème, et fait donc partie du système à résoudre. Cette formulation permet de définir une méthode d'ordre 2 sur la température et son gradient.

Plus précisément, une méthode d'enrichissement sur la température est proposée de manière à obtenir l'ordre 2 sur la variable primale et son gradient à partir d'une méthode éléments finis $\mathbb{P}^1$ tout en conservant un système linéaire de même taille. Sur un domaîne discrétisé les fonctions de bases $\mathbb{P}^2$ dénotées $\{\phi_i^{\mathbb{P}^2}(x)\}_{i\in N_1}$ sont définies en fonction des fonctions de bases $\mathbb{P}^1$ dénotées $\{\phi_i^{\mathbb{P}^1}(x)\}_{i\in N_2}$, et des développements de Taylor sont utilisés pour récupérer une approximation de la valeur milieux des nœuds originellement présent sur les éléments finis $\mathbb{P}^2$. Pour résumer, l'enrichissement de la température consiste à définir une approximation de la température T d'ordre 2 de la forme T_h:

$$T_h(x) = \sum_{i,sommet} T_i \phi_i^{\mathbb{P}^2}(x) \; + \; \sum_{k,milieu} T_k \phi_k^{\mathbb{P}^2}(x) \quad . \tag{1}$$

où k est un nœud milieu entre les nœuds de sommets i et j.

L'approximation (1) est remplacée par une définition où les fonctions de bases $\mathbb{P}^2 = \{\phi_i^{\mathbb{P}^2}(x)\}_{i\in N_2}$ sont remplacées par une combinaisons linéaires de fonctions de bases $\mathbb{P}^1 = \{\phi_i^{\mathbb{P}^1}(x)\}_{i\in N_1}$. Les valeurs milieux aux nœuds k sont remplacées par un développement de Taylor des nœuds aux extrémités d'indices i et j.

Le résultat est une approximation de la température qui dépend aussi de la définition du flux dénoté ici β. La définition suivante est ainsi obtenue:

$$T_h(x) = \sum_{i,sommet} T_i \phi_i^{\mathbb{P}^1}(x) + \sum_{k,milieu\ de\ [i,j]} 1/2\ (\lambda^{-1}\beta_j - \lambda^{-1}\beta_i) \cdot (x_j - x_i)\ \phi_i^{\mathbb{P}^1}(x)\phi_j^{\mathbb{P}^1}(x)\ , \qquad (2)$$

ou λ désigne la variable de conductivité thermique.
La définition (2) est rendue possible grâce au choix de résoudre le problème de Stefan dans sa formulation mixte, ce qui permet d'obtenir la valeur de β pour les degrés de liberté associés aux éléments de maillage.

Un des principaux problèmes avec l'interface mobile présente dans le modèle de Stefan est l'expansion d'une des phases du modèle, zone solide devenant liquide pour un modèle de dégivrage. Des nœuds présents dans la zone solide peuvent donc changer de zone à l'itération suivante, et les solutions des pas de temps précédents ne sont donc pas disponibles pour cette zone, bien que leurs valeurs soient nécessaires pour les schémas en temps. Dans ce manuscrit, ces valeurs sont extrapolées en utilisant une résolution des moindres carrés par factorisation QR. Cette méthode utilise un stencil de points défini en chaque nœud où une valeur est manquante, et est composé des nœuds voisins les plus proches du nœud ou la valeur est reconstruite. Les résultats numériques présentés dans le Chapitre 3 prouvent que cette méthode de reconstruction permet de conserver l'ordre 2 défini par le schéma. Dans un premier temps, le Chapitre 3 propose des résultats pour des interfaces mobiles non discrétisées, concernant une interface plane sur des domaines rectangulaires. Un cas test académique est également présenté, représentant un bloc de glace semi-infini avec un déplacement du front dans la direction horizontale. La méthode est également testée sur un domaine circulaire pour une interface définie par une valeur de rayon pour permettre l'étude d'un cas avec déplacement bidirectionnel. La méthode est étendue par la suite à des interfaces discrétisées dans le Chapitre 4.

Un second problème est engendré par le déplacement de l'interface: l'approximation de la vitesse de déplacement du front de fusion. La condition de fermeture du modèle, appelée condition de Stefan est discrétisée pour obtenir un schéma sur la position d'interface. Cependant, la vitesse calculée doit aussi être approximée sur l'interface physique et non sur l'interface numérique, l'interface numérique étant celle utilisée dans la résolution et où la solution est disponible. Une autre méthode de reconstruction est utilisée pour obtenir une approximation du saut du flux sur le véritable front de fusion, approximation qui elle aussi ne détériore pas l'ordre de convergence de la méthode établi par l'enrichissement de la température. Pour ce faire, une valeur de part et d'autre de l'interface est calculée dans chaque phase du modèle.

Le dernier point numérique concernant le modèle de Stefan est le type de conditions imposées à l'interface, une zone commune à deux sous-domaines représentant des éléments physiques différents (eau et glace). Numériquement, des conditions de sauts et de moyennes sont employées. Ces conditions sont imposées dans la matrice du système linéaire en dédoublant les nœuds de maillage présents le long de l'interface de substitution. Cette approche est réinitialisée lorsque le front de fusion se déplace et rencontre de nouveaux éléments du maillage. Il est important de prendre des précautions, car le déplacement du front de fusion peut modifier l'ordre d'identification des éléments traversés par l'interface physique. Lorsque de nouveaux éléments sont ajoutés à la zone définissant les éléments coupés par l'interface physique, ces nouveaux éléments peuvent apparaître n'importe où le long de l'interface.

La position des nœuds à l'interface lors de l'itération précédente doit être conservée pour permettre, si ces nœuds sont toujours présents, la redistribution de leurs valeurs sur la nouvelle position de l'interface numérique. En cas de nouveaux nœuds le long de l'interface de substitution, la méthode précédemment expliquée est utilisée pour reconstruire les valeurs manquantes.

3 Discontinuité à l'interface et stabilité

La formulation mixte est choisie pour fournir une meilleure précision sur le saut de flux utilisé pour déterminer la vitesse de déplacement de l'interface. Dans ce manuscrit de thèse, la stabilité des simulations représentant un changement de phase de solide à liquide et la configuration dans laquelle elles sont stables ou non est étudiée. Lorsque l'interface est discrétisée comme un ensemble de nœuds, le calcul de la vitesse sur ces nœuds peut engendrer des oscillations sur la position de l'interface et potentiellement impacter tous le domaine de résolution. D'un point de vue numérique, la stabilité de la méthode exprimée dans [13], article publié issu des travaux de cette thèse, est étudiée pour la méthode e-SBM appliquée aux fronts mobiles. Plus précisément, une étude sur le déplacement d'un front perturbé ainsi qu'une perturbation sur le champ de température est menée. L'analyse est effectuée sur les équations du problème de Stefan (approche continue) ainsi que sur la formulation variationnelle proposée (approche numérique). La linéarisation du modèle de Stefan est utilisée pour définir une relation de dispersion, qui caractérise la stabilité et la nature dispersive ou non des composantes d'onde du modèle autour d'un état stationnaire. L'idée principale de cette étude est d'examiner l'opérateur linéaire issu de l'équation autour d'un état stationnaire connu.

Une fois que l'EDP est linéarisée, il est alors possible d'obtenir **une relation de dispersion**. Plus précisément, cette relation de dispersion caractérise les modes d'oscillation spatiale du problème linéaire à partir d'un état stationnaire. L'analyse de stabilité est effectuée sur des configurations 2D, et sur le modèle de Stefan adimensionné pour permettre une meilleure interprétation des résultats. Le processus d'adimensionnement du modèle de Stefan ainsi que celui des conditions aux limites est développé comme étape préliminaire essentielle à l'étude de stabilité.

L'étude réalisée en Chapitre 4 a démontré que la méthode e-SBM développée et appliquée au modèle de Stefan dans un contexte de fusion de l'eau est **stable**, et qu'une perturbation à l'interface ou sur le champ de température **ne se propage pas** dans le domaine et **se dissipe au cours du temps**. La même conclusion a été faite sur les équations du modèle continu montrant que le schéma numérique utilisé conserve les propriétés du modèle continu. Plusieurs cas tests avec différentes géométries d'interface démontrent ces résultats dans le dernier chapitre de cette thèse. Un front perturbé avec une géométrie complexe et sur un domaine circulaire retourne à un état stationnaire sous la forme d'une interface circulaire après un certain nombre d'itérations. Ce cas permet de démontrer le caractére stable et robuste de la méthode pour une variété de géométries d'interface. Dans le cas de l'étude de stabilité sur la méthode numérique l'ordre de convergence du modèle sur la décroissance de la perturbation a aussi été étudié. Les résultats présentés en Chapitre 4 montre que l'ordre 2 est conservé.

Les travaux présentés dans ce manuscrit de thèse ont fait l'objet de plusieurs présentations lors de conférences internationales. Un article a été publié et un autre est en cours de production et sa finalisation et soumission sont prévues très prochainement. Ses différentes contributions sont présentées ci-dessous.

Conférences:

- "High Order Scheme for Mixed Formulation of Problems with Moving Internal Boundaries" T. Carlier, L. Nouveau, H. Beaugendre, M. Colin, M Ricchiuto, ECCOMAS 2022, European Congress on Computational Methods in Applied Sciences and Engineering, Juin 2022, Oslo, Norvège.

- "Shifted Boundary Method and Moving Front for Mixed Formulation of Phase Change Problems" T. Carlier, L. Nouveau, H. Beaugendre, M. Colin, M. Ricchiuto, ECCOMAS

2022, European Congress on Computational Methods in Applied Sciences and Engineering, Juin 2022, Oslo, Norvège.

- "An Enriched Shifted Boundary Method for Moving Boundaries" T. Carlier, B. T. Helenbrook, H. Beaugendre, M.Colin and L. Nouveau, Présentation sur invitation, NASA Langley research Center, Mars 2023, Virginie, USA.

- "Stability Analysis and Numerical Study of Stefan Problems for Embedded Computation of Moving Internal Boundaries" T. Carlier, B. T. Helenbrook, H. Beaugendre and M.Colin, CFC2023, 22nd Computational Fluids Conference, Avril 2023, Cannes, France.

- "Numerical Analysis of Stefan Problems for Embedded Computation of Moving Internal Boundaries" T. Carlier, B. T. Helenbrook, H. Beaugendre, M.Colin and L. Nouveau, ENUMATH 2023, Septembre 2023, Lisbonne Portugale.

Articles

- An Enriched Shifted Boundary Method to Account For Moving Fronts, T. Carlier, L. Nouveau, H. Beaugendre, M. Colin, M Ricchiuto, Juin 2023, Journal of Computational Physics, Publié

- Stability Analysis and Numerical Study of the Shifted Boundary Method and its application to the Stefan Model , T. Carlier, B. T. Helenbrook, H. Beaugendre and M. Colin, En Production, Soumission prévu avant soutenance

Normality is a paved road: It's comfortable to walk, but no flowers grow on it,

Vincent Van Gogh.

Contents

List of Abbreviations

AROD Adaptive Recontruction for Off-site Data

BVCM Boundary Value Correction Method

CSC Computer Science Coorporation

DOF Degree Of Freedom

EBM Embedded Boundary Method

ETIS Electro-Thermal Ice Protection Systems

e-SBM enriched Shifted Boundary Method

FE Finite Element

FEA Finite Element Analysis

FEM Finite Element Method

FIKI Flight Into Known Icing

FPD Freezing Point Depressant

FVM Finite Volume Method

IBM Immersed Boundary Method

MSC MacNeal-Schwendler Coorporation

NASA National Aeronautics and Space Administration

NASTRAN NAsa STructural ANalysis

nbDOF number of Degree Of Freedom

ODE Ordinary Differential Equation

PDE Partial Differential Equation

ROD Reconstruction for Off-Site Data

SBM Shifted Boundary Method

TIB-LBM Thermal Immersed Boundary-Lattice Boltzman Method

W-SBM Weighted Shifted Boundary Method

List of Figures

Introduction

List of Tables

Introduction

1 Context

1.1 Icing threats and ice formation

One of the most important threats to the safety of an aircraft is the **formation of ice**, a phenomenon called **icing** or **icing threat**. The formation of ice is due to the solidification of supercooled water on the surface of the aircraft [1]. Supercooled water is water present in the atmosphere at a physical state between liquid and solid. The temperature of the water is below the freezing point (0°C) but the water stays liquid due to the lack of formation of ice crystals. This particular state is due to the purity of the water present in the atmosphere and its lack of ability to complete the nucleation process to become solid. When a supercooled water droplet touches the aircraft's surface, and when the surface is below freezing temperature the molecules composing the droplet interact with each other (through the wave created at the impact with the surface) allowing the formation of ice crystals and the solidification of the water droplet on the aircraft. It is important to note that some droplets of water can get swept away along the aircraft surface and freeze further than their point of impact with the aircraft.

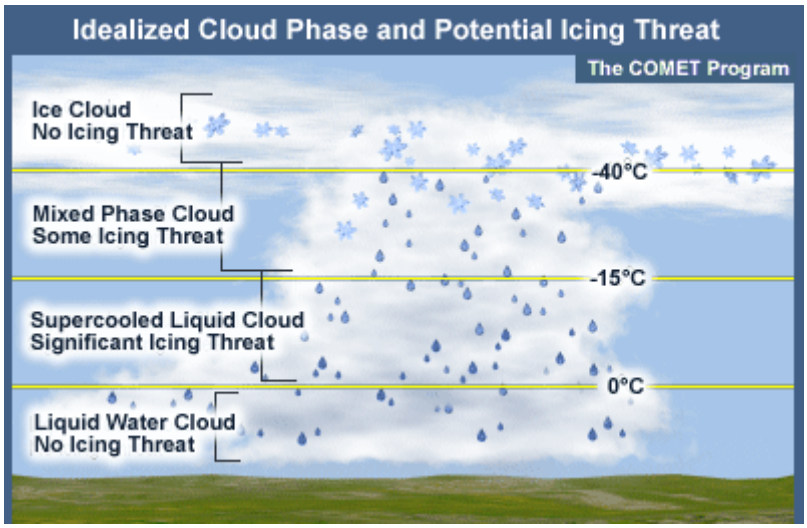


Figure 1: Ice hazard zones in the atmosphere[1]

Two conditions are necessary for the formation of ice on an aircraft, [1]. They are called **icing conditions** and are as stated as below:

[1] https://www.weather.gov/source/zhu/ZHU_Training_Page/icing_stuff/icing/icing.htm.

1. The **air temperature** needs to be at 0°C or colder. Moreover an aircraft which has passed in an area below freezing point can still have its surface below freezing point even after moving to a hotter area. The most dangerous zone for an aircraft to flight in, is a temperature zone between −15°C and 0°C, see Figure 1.

2. The **presence of supercooled water droplets** in the atmosphere, quantity and size being the main factors. The purer the droplets are the lower their freezing point is, making an area between −40°C and −15°C still a potential zone of risk, see Figure 1. Below −40°C even pure water would solidify, due to the phenomenon of homogeneous nucleation.

1.2 Adverse effects in flight

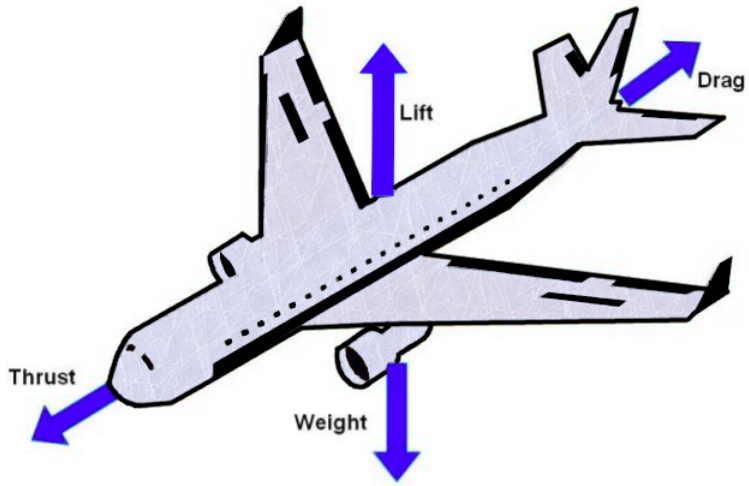


Figure 2: Adverse effects of icing on an aircraft[2]

Icing can cause multiple threats to an aircraft and they can be categorized into four main categories [14], visible in Figure 2:

1. Increased drag

   The **drag** is the **force opposite to the motion of the aircraft** that resists to the forces imposed by the surrounding fluid, here being the air. When it comes to aircraft, this type of drag is commonly referred to as **aerodynamic drag**. Aerodynamic drag is a result of the interaction between the aircraft and the air molecules through which it is moving. An increase in drag can impact fuel consumption and the overall performance of the aircraft.

2. Decreased lift

   Lift is the **upward force** that holds the aircraft in the air. It is mostly generated by the wings. When an aircraft's wing moves through the air, the specific shape of the wing and the angle at which it meets the incoming airflow (angle of attack) cause the air above the wing to move faster than the air below it. This difference in airspeed leads to lower pressure above the wing and higher pressure below the wing, resulting in an upward force

[2]https://www.grc.nasa.gov/www/k-12/VirtualAero/BottleRocket/airplane/lift1.html

called lift. A decrease in lift can result in a loss of altitude or the inability to maintain level flight.

3. Increased weight

   The accumulation of ice on the surface of the aircraft can increase the **weight of the aircraft** which can have multiple effects. The most dangerous ones are the ability to increase altitude or to maintain it. Depending on the distribution of ice on the aircraft surface it can also modify the center of gravity and impact the overall stability.

4. Reduced thrust

   Thrust is the **mechanical force** created by the engines of the aircraft. It is the necessary force used to move the aircraft forward, maintain its altitude, accelerate, change direction and to take off. In the case of reduced thrust, it can lead to a loss of engine efficiency, a reduction in the ability to climb to higher altitudes, such as the takeoff moment, and a decrease in speed. All of these factors can ultimately affect the aircraft's maneuverability.

Some of those adverse effects can cause minor problems such as an increase in fuel consumption or loss of performance but in the worse cases it could lead to a loss of control and cause an aircraft crash. One notorious tragic accident is the American Eagle Flight 4184, a domestic flight between Indiana to Chicago in the USA on October 31, 1994. The aircraft encountered severed icing conditions and due to the build up of ice, lost controlled and crashed into a field, which caused the death of all the people on board, [15].

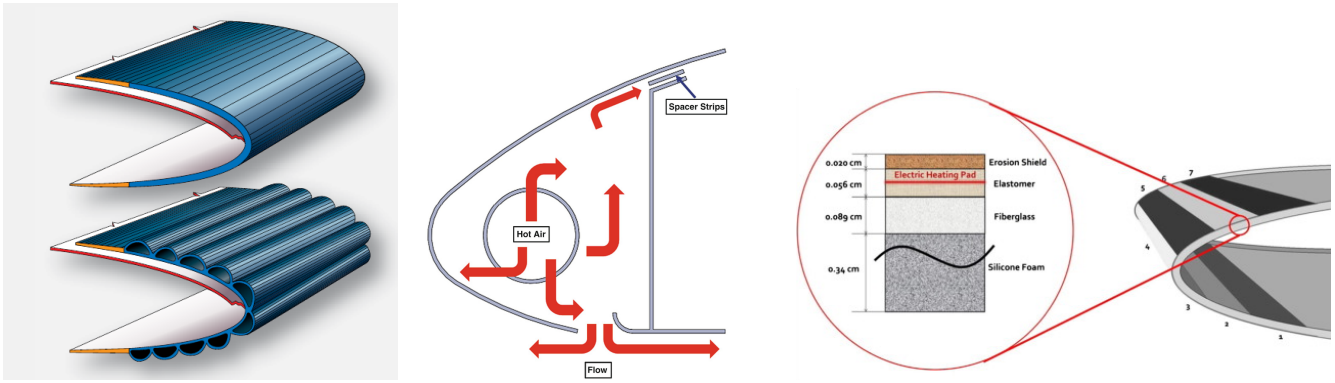
As referenced in the manual of aircraft ground de-icing/anti-icing operations of the International Civil Aviation Organization in 2018 "Wind tunnel and flight tests indicate that ice, frost or snow formations on the leading edge and upper surface of a wing, having a thickness and surface roughness similar to medium or coarse sandpaper, can reduce wing lift by as much as 30 per cent and increase drag by up to 40 per cent", meaning that icing threat is a high potential threat for the safety of an aircraft and its passengers, [16].

1.3 Ice Protection Systems

Aircraft are required to obtain certification confirming their suitability for safe flight in icing conditions. This certification depends on the airspace in which the flight would take place. For instance, in the United States and for aircraft operating in European airspace, being **FIKI** certified (Flight Into Known Icing) is a mandatory requirement for flights conducted in icing conditions, and is even considered only as a **minimum requirement**. There are two types of paradigms to be able to flight into icing conditions. One consists in preventing the formation of ice and the use of **anti-icing** systems. Those systems prevent the formation of ice by the generation of heat keeping the temperature of the surface above the freezing temperature. The heat can be originated from electrical heaters, the engines' bleed air or a chemical reaction through a fluid-based system. The second option is to let the accretion of ice happen and to remove it afterwards before it creates a problem for the aircraft (see Section 1.2) using **de-icing systems** such as thermal systems, pneumatic boots systems or liquid based systems, see Figure 3. Four main categories of anti-icing/de-icing systems can be defined:

- Pneumatic systems

  Pneumatic systems are de-icing systems which can be used only on the edge of the aircraft' wings to remove the forming ice. On the leading edge of the airfoil pneumatic rubber boots are inflates and deflates to break the ice, which then falls from the surface where it was attached, see Figure 3(a).



(a) Pneumatic de-icing systems (b) Bleed air anti-icing/de-icing systems (c) Thermal anti-icing/de-icing systems

Figure 3: Examples of anti-icing and de-icing systems[3,4,5]

- Bleed air systems

  Bleed air systems consist in the redirection of the air coming from the aircraft's hot engines to the critical areas where we want to prevent the accretion of ice. Those type of systems are used for the larger aircraft which possess turbine power. On smaller aircraft it could affect the temperature of the aircraft's engines (generally piston-powered), which make those systems suitable only for bigger aircraft, see Figure 3(b).

- Thermal systems

  Thermal systems use electrical heaters embedded in the plane's structure to generate heat. They can both be used as an anti-icing system and as a de-icing system and use electricity as a source of energy, see Figure 3(c). De-icing thermal systems cost less energy than anti-icing systems as they are not used constantly allowing the accumulation of a small amount of ice on the surface of the aircraft.

- Liquid based systems

  Fluid based systems use fluids called Freezing Point Depressant (FPD) fluids, [17]. In the case of an anti-icing system an antifreeze liquid solution is used to prevent the formation of ice. For a de-icing system the high pressure of the spray equipment (used for de-icing while the aircraft is on the ground) and the de-icing fluid allow to remove ice. De-icing fluids are based on ethylene glycol, diethylene glycol or propylene glycol which contain water, corrosion inhibitors, wetting agent and dye (the color is used to differentiate the fluid's types). Anti-icing fluids have a similar composition but with the addition of a thickener to stay longer on the surface and to have a longer period of time coverage. There are four types of fluids categorized as follow:

  - Type I : De-icing/Anti-icing fluids with a limited time of coverage
  - Type II : Anti-icing fluids designed to stay on the aircraft during ground operation
  - Type III : Anti-icing fluids similar to type II with shorter period of time coverage
  - Type IV : Anti-icing fluids similar to type II with longer period of time coverage.

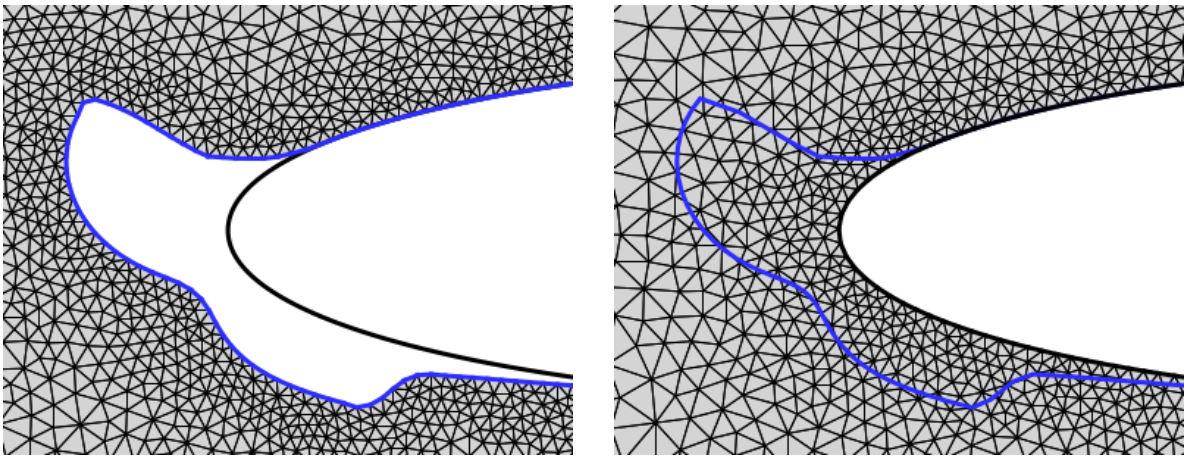
[3]https://www.aircraftsystemstech.com/2017/05/wing-and-stabilizer-deicing-systems.html
[4]Thomas Filburn, Anti-ice and Deice Systems for Wings, Nacelles, and Instruments, Springer, 2019.
[5]M. Pourbagian, W.G. Habashi, Aero-thermal optimization of in-flight electro-thermal ice protection systems in transient de-icing mode, International Journal of Heat and Fluid Flow, 2015.

Liquid based systems are a useful solution but do not guarantee full protection especially under bad weather conditions like heavy rain or active frost.

In the context of more electric aircraft, thermal ice protection systems or Electro-Thermal Ice Protection Systems (ETIPS) are a huge source of research and development. The use of self-regulating heater elements allows more control and can be tailored to specific areas of the aircraft.

In this thesis manuscript, thermal systems from a more academic perspective will be investigated, exploring recent methods for embedded simulations applied to the specific case of ice/water simulations and the explicit tracking of mobile interfaces, such as a phase-change front. The mathematical model studied is the **Stefan Problem**, describing phase-transition models for free and moving boundaries. The problem is solved with an embedded method: the **Shifted Boundary Method** introduced in [2, 3] and **expanded in this manuscript** to the simulation of **moving interfaces**. Results on the explicit tracking of the interface, the accuracy of the method, the stability of the model, and capacity of the scheme to handle discontinuities at the interface will be presented.

2 Methods and Numerical challenges



(a) Body-fitted approach (b) Embedded approach

Figure 4: Body-fitted approach and embedded approach on the ice accretion on an airfoil[6]

Numerical simulations allow for the consideration of a multitude of possible scenarios without the expenses associated with wind tunnel testing used to assess the effects of airflow on scale models [10], and without the risks associated with real-life flight testing. Nevertheless, one of the main difficulties of numerical simulations is the precision of the mesh on the sharp ice geometry and its adaptation to the moving boundary, the melting front between ice and water. Those method are referred to as **body-fitted approaches,** which require robustness and time in the generation of the mesh to fit to the geometry properly. In Figure 4(a) the mesh has been generated around the ice formed on an airfoil and edges of the mesh, which matches exactly the geometry of the ice accretion. There exists different approaches that allow the use of non body-fitted meshes, see Figure 4(b). Those methods are referred to as **Immersed Boundary Methods** (IBM) or **Embedded Boundary Methods** (EMB).

[6]Pierre LAVOIE, Méthode de frontières immergées pour la modélisation du givrage en vol des aéronefs, Thèse de l'Université de Toulouse, 2021

In this manuscript, to avoid confusions, Embedded Boundary Methods (EBM) refers to any methods associated with direct or continuous forcing techniques on non body fitted meshes. The different EBM techniques differ by their treatment of the boundary conditions. After reviewing some of the available methods on both body-fitted and embedded approaches used for icing and/or moving boundaries problems, a description of the method considered in this manuscript will be presented: the **Shifted Boundary Method**.

2.1 Body-fitted approach

Concerning the simulation of problems with moving boundaries some body fitted methods have proven their robustness in the Lagrangian framework [18, 19, 20, 21], or in the Arbitrary Lagrangian Eulerian formalism [22, 23, 24, 25, 26, 27]. These body-fitted methods use mesh deformations to match the interface displacement and require remeshing to handle large deformations. Another approach developed in [28, 29, 30] consists of an anisotropic mesh adaptions methods, which takes into account the anisotropic nature (non-uniform structural property) of the problem. In the same range of idea the overset grid method presented in [31, 32] can also be mentioned and has been applied to the unsteady Stefan problem in [33]. The method involves multiple grids overlapping each other, moving with time, and covering the entire domain of resolution to create a composite grid. resolution to create a composite grid.

In the specific area of icing simulations most of the numerical methods follow a sequential mode referred to as a **multi-step method:**

1. Mesh generation
2. Solvers (airflow, droplets, accretion,...)
3. Geometry update (moving boundaries)

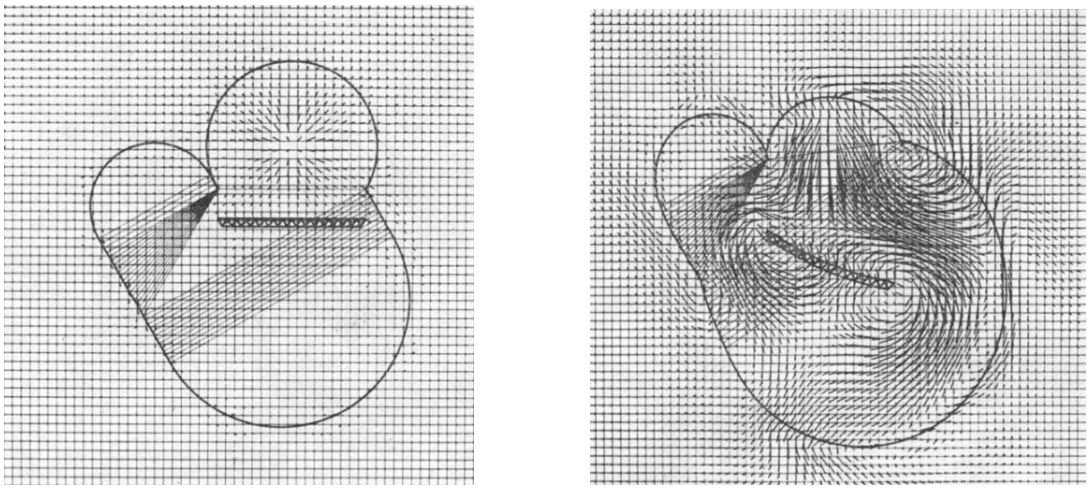
Step 1 and 3 require time to ensure a precise description of the new domain geometry and can not be fully automated when large deformations arise. This process of adapting/moving/refining the mesh can be time consuming, especially regarding highly parallel computations and complex geometries, such as those found in ice accretion.

In the next section, we present a category of methods that enable the use of non-conforming mesh discretization. This category provides an attractive approach for solving problems with moving boundaries.

2.2 Embedded approach

Performing numerical simulations on systems with boundaries requires a precise definition of the physical domain geometry and the tessellation of that domain can be a tedious task, especially when boundaries are moving. In this direction, Immersed and Embedded boundary methods became a popular and active field of research, since Charles S. Peskin's work in 1972 on simulating blood flow in the heart [34, 35]. The concept of immersed/embedded resolution developed by Peskin in [34] that he adapted to the muscular heart wall in [36], involves embedding the heart geometry within a cartesian meshed box, see Figure 5. The region outside the heart is considered as a fluid region, allowing for the deformation of the heart to incoming and outgoing blood flow. The boundary presented in his work is an internal boundary representing the valves and heart muscle in a Lagrangian framework. A stencil of points $\{x_k\}_{k\in\mathbb{N}}$ is used to

discretized the boundary and is updated using an equation of motion. The equation of motion is formulated to minimize the differentiation between fluid and non-fluid zones.



(a) Relaxation of the ventricle (b) Contraction of the atrium

Figure 5: Simulations of blood flow in the heart on an embedded cartesian grid from the work of Peskin in [36]

The main advantage of any immersed/embedded method is to avoid explicit meshing of the moving boundary or interface. They only require a fixed mesh on which the moving interface/boundary must be considered. The main challenge is then to correctly and accurately compute the moving interface or boundary conditions. Exhaustive reviews on EBM have been proposed by Mittal and Iaccarino [37], Sotiropoulos and Yang [38] and Griffith and Patankar [39]. EBM can be used on both structured or unstructured grids. Boundary conditions are treated using either a **continuous forcing technique** [40, 41, 42, 43, 44], a **discrete forcing technique** [25, 45, 46] or a **cut-cells technique** [47, 48, 49, 50, 51, 52].
These three techniques are detailed below :

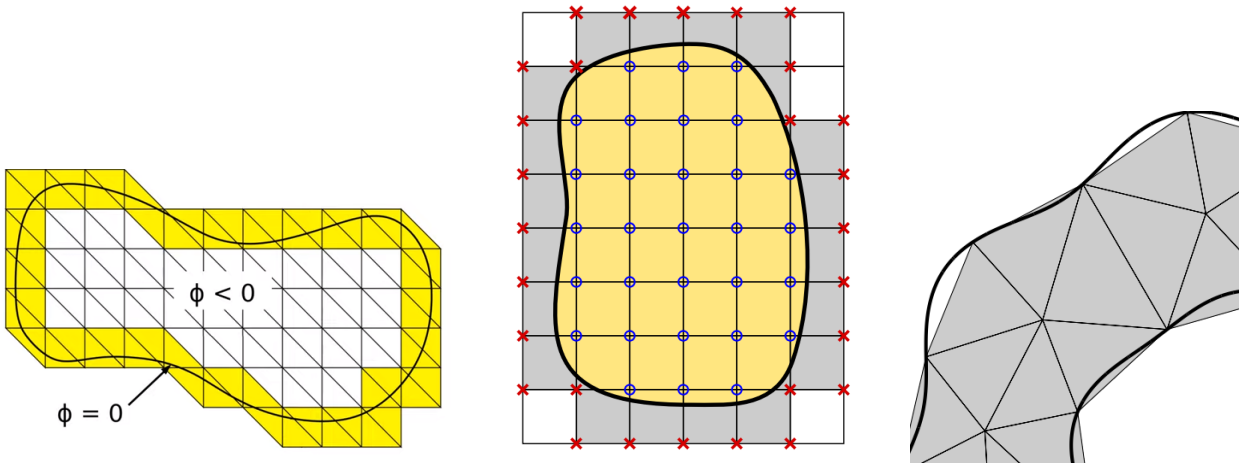
1. The **continuous forcing technique** is mainly used for fluid-structure interactions, for a body immersed in a fluid environment. The fluid equation is solved on a background mesh, not conform to the immersed domain. The forces (pressure, viscosity, gravity, thermal effects,...) exerted by the domain on the surrounding fluid are integrated into the continuous fluid equation (ex: Navier-Stokes equations). These forces are updated at every time step to incorporate the new position of the immersed boundary.

2. The **discrete forcing technique** has the same applications and concepts than the continuous forcing technique but differs from its imposition of boundary conditions on the immersed body. In that context, the boundary conditions are imposed locally on mesh nodes close to the immersed boundary using ghost cells. This technique is computationally more efficient, with a simpler implementation and a more straightforward approach. However, the accuracy of the method depends on the distribution of the nodes and stencil chosen to impose the boundary conditions.

3. The **cut-cell technique** consists in the treatment of the mesh elements cut by the boundary of the immersed body. The differentiation between the various cut-cell techniques lies in the resolution of the PDEs within the cut elements and the imposition of boundary conditions on them. A cut-cell method recovers the optimal accuracy of the underlying

numerical scheme but, in practice, it is complex to implement and suffers from the so-called "small cut-cell problems", particularly problematic in three dimensions. It provides bad-conditioning and requires specific stabilisation, see for instance [53, 54, 55]. Strategies have been proposed within the cutFEM context for embedded interface problems in [50] and immersed interface method for elliptic equations, see [56].

Innovative methods have been proposed as alternatives, to avoid the need of intersecting the embedded geometry and the mesh. Among those methods, one can find the ***ϕ*-FEM** based on **level-set** (see [57] for Dirichlet conditions and [58] for Neumann conditions), the **aggregated unfitted FEM** [59, 60], an **immersed method for curved domains** in [61] based on the work in [62, 63], and the **Shifted Boundary Method** [2, 64, 65, 66, 67]. The Shifted Boundary Method is expanded in this thesis and will be explained more in detailed in the next section of this Introduction and in Chapter 1. A description of the other methods is available below:



(a) ϕ-FEM approach[7] (b) Aggregated unfitted FEM[8] (c) ROD method (FVM approach)[9]

Figure 6: Visualization of various immersed methods and their approaches on handling the immersed boundary

- The ***ϕ*-FEM** method is a finite element method using a level set function ϕ to represent the boundary of the immersed body, see Figure 6(a). A good candidate is the signed distance function defined as null along the boundary of the immersed body and positive or negative inside the domain of resolution. The function ϕ is then integrated in the definition of the PDE and the weak formulation obtained is also depending on ϕ. The formulation to be solved is naturally ill-posed and need to be stabilized, with terms depending once again of ϕ (ex: ghost penalization).

- The **aggregated unfitted FEM** is an enhanced FEM based on a cell aggregation technique originally designed for elliptic problems. The idea is to eliminate problematic Degrees Of Freedom (DOF) by imposing constraints on the definition of the FE spaces. Two

[7]Seminario De EDP E Matematica aplicada, ϕ-FEM: A fictitious domain method for finite element metimmersed boundaryhods on domains defined by level-sets, M. Duprez, V. Lleras and A. Lozinski, 2023

[8]S. Badia, F. Verdugo, A. F. Martín, The aggregated unfitted finite element method for elliptic problems, Computer Methods in Applied Mechanics and Engineering, 2018, https://doi.org/10.1016/j.cma.2018.03.022.

[9]SHARK-FV 2018 Conference, Gaspar J. Machado Presentation slides, http://loubere.free.fr/SHARK_PRESENTATIONS/2018/Machado_SHARK18.pdf

distinct FE spaces are defined. The first consists of interior nodes (blue nodes in Figure 6(b)) where none of the nodes belong to mesh elements cut by the immersed boundary. The second one is constituted of outer nodes (red nodes in Figure 6(b)) where an extension operator is applied to incorporate the properties of the cut elements (aggregated cells) into the numerical scheme. These properties can include forces, constraints, velocity, pressure, density, and more. This method was developed to overcome conditioning problems in the matrix of the linear system associated with standard unfitted FEM.

- **Immersed method for curved domains** are a wide range of methods for curved domains. One can cite the **Reconstruction for Off-Site Data method** (ROD) in [68]. It consists in considering a computational polygonal domain different from the physical domain where the PDE takes place, see Figure 6(c). It is a Finite Volume Method (FVM) using a polynomial reconstructions in a Least Square context on collocation nodes located on the physical domain. The reconstructions are carried out on the mesh using data outside the computation domain, using information on the physical domain of resolution instead. The method has been expanded for moving boundaries with the AROD method in [63] (Adaptive Reconstruction for Off-site Data) by adding fitting conditions to the formulation. One can also find the AROD-xy method, which combines fitting conditions and least-square constraints of different types, as detailed in reference [63].

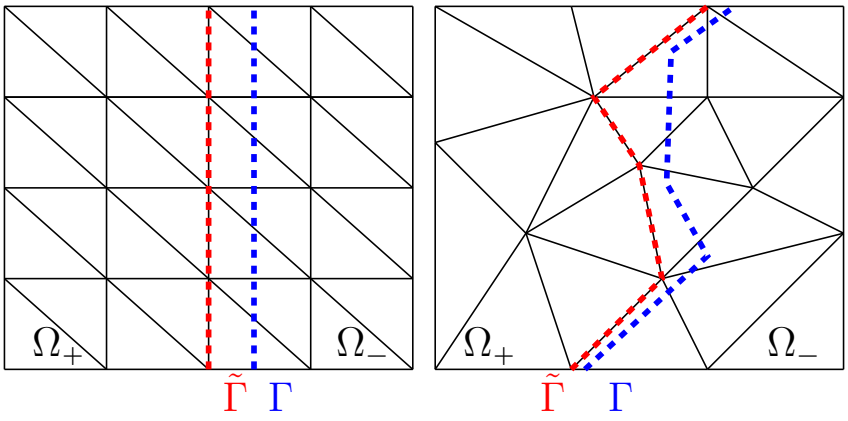
In the specific context of de-icing simulations, embedded methods have been proposed. A common method involves combining an embedded grid technique with the level-set approach. In reference [69], the level-set method is employed to model the evolving ice/air interface in a 3D ice accretion code. The no-slip boundary condition (fluid velocity null at the solid-fluid boundary) is imposed using a penalty term. The combination of the Level-Set method and penalization imposition has also been developed in [70] to study ice-wall interactions. This approach uses a linear spring collision model, and the velocity inside the solid is enforced through penalization. Another example can be found in [71], where an ice shedding trajectory model is discussed. In this case, a penalization term is once again employed to enforce the no-slip boundary condition, and the Level-Set method is used to capture the interface of the solid bodies.

For solid-liquid interactions, an embedded method applied to ice flurry flows has been developed by Kosuke et al. proposing a **Thermal Immersed Boundary-Lattice Boltzmann Method** (TIB-LBM) in [72]. The embedded method is used to satisfy the thermal boundary conditions. It was introduced by Feng and Michaelides for fluid-particles interactions in [73, 74]. The TIB-LBM has also showed its ability for the sedimentation of hot/cold particles in [75, 76]. Concerning the resolution of the Stefan problem itself some embedded methods have been tested, for instance with the immersed interface/boundary method in [44, 77, 78, 79].

In the next section, the Shifted Boundary Method is discussed. The method will be expanded in this manuscript to moving interfaces and will be referred as **e-SBM** (enriched Shifted Boundary Method).

2.3 The Shifted Boundary Method

In this manuscript the **embedded method** used to deal with the moving interface is the **Shifted Boundary Method** (SBM), introduced by Main and Scovazzi for Laplace and Stokes equations in [2]. The method is modified to account for the moving interface representing the phase-change front between liquid and solid water. The idea described in [2] is to weakly enforce the boundary conditions using the Nitsche's method [80] on a surrogate boundary $\tilde{\Gamma}$, defined



(a) Structured Mesh (b) Unstructured Mesh

Figure 7: Interface Γ and its surrogate interface $\tilde{\Gamma}$

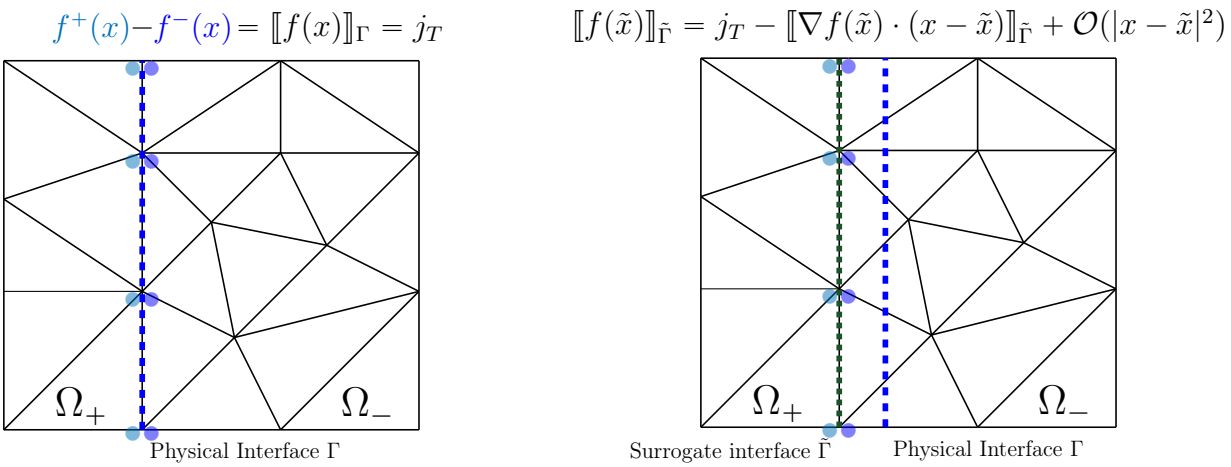
by edges/faces of the background mesh, see Figure 7 for an example in 2D.
The SBM is derived from the work of Bramble et al. on **the Boundary Value Correction Method** (BVCM) in [81]. The BVCM uses the projection method introduced by Nitsche in [11] to define a finite weak formulation with a penalty coefficient to impose boundary conditions, and verify the criteria of coercivity of the weak formulation. The domain of resolution Ω is approximated by $\Omega_h = \mathring{\Omega} \cup \partial\Omega_h$, where $\partial\Omega_h$ is a piece-wise linear approximation of $\partial\Omega$. A Taylor expansion is then used to approximate the boundary conditions on the new boundary $\partial\Omega_h$. The idea of the imposition of boundary conditions via Taylor expansions was taken over by Main and Scovazzi in [2] for the imposition of the boundary conditions on their surrogate boundary.
As a matter of example, an arbitrary function f with sufficient regularity on a domain Ω, and which possesses at least a derivative of order one, is chosen. If f satisfies the following conditions (see Figure 7) on Γ :

$$[\![f]\!]_\Gamma = f^+(x) - f^-(x) = j_T, \tag{1}$$

with j_T a given data and where $[\![\cdot]\!]$ is the jump operator on Γ ($f^+ \in \Omega_+$ and $f^- \in \Omega_-$, see Figure 8), then an approximation of that condition can be deduced on $\tilde{\Gamma}$ using a Taylor expansion from $x \in \Gamma$ to $\tilde{x} \in \tilde{\Gamma}$ as follows

$$\begin{aligned} [\![f(x)]\!]_\Gamma &= [\![f(\tilde{x}) + \nabla \cdot f(\tilde{x})(x - \tilde{x})]\!]_{\tilde{\Gamma}} + \mathcal{O}(|x - \tilde{x}|^2), \\ \implies [\![f(\tilde{x})]\!]_{\tilde{\Gamma}} &= j_T - [\![\nabla \cdot f(\tilde{x})(x - \tilde{x})]\!]_{\tilde{\Gamma}} - \mathcal{O}(|x - \tilde{x}|^2). \end{aligned} \tag{2}$$

The differences with the imposition of the two conditions are displayed below, conform with Equation (1) in Figure 8.a), and embedded with Equation (2) in Figure 8.b).



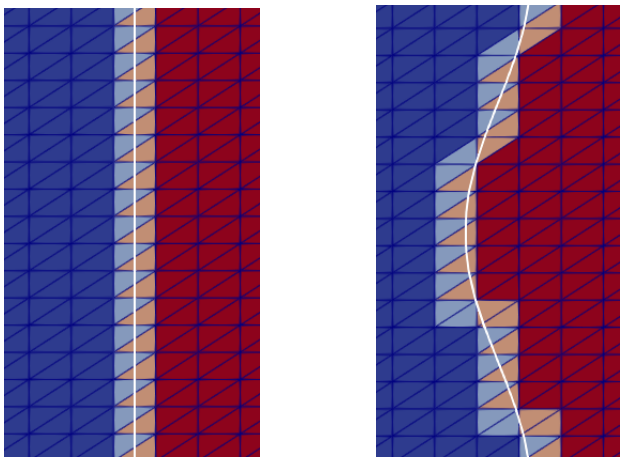
(a) Conform Imposition

(b) Embedded Imposition (SBM)

Figure 8: Differences between conform imposition and embedded imposition

An important difference between the SBM and the BVCM in [81] is the ability of the method to be used for any type of boundary geometry. Indeed, the BVCM is developed for domain with polygonal boundaries only. Nevertheless, one can cite the recent expansion of the method on unfitted meshes with the CutFEM method by Burman et al. in [82]. The boundary value correction method and the SBM are both falling into the category of **approximate domain methods**. As a non exhaustive list in the same category one can find the **ϕ-FEM method** [57, 58, 83], **the fictitious domain method** [54, 84, 85], **the Fat Boundary Method** [86, 87] and **methods with extensions of subdomains** [88, 89].
In this manuscript, the SBM was the choice made to deal with the moving front. This choice comes from the ability of the method to define numerical interfaces easily even for complicated geometry. The principle of the method, applied to a moving interface (Stefan model) is to recover an optimal accuracy of the interface conditions using Taylor expansions, where the accuracy depends on the order in the development of the Taylor expansion, see Equation (2).

Concerning the definition of the surrogate interface, it is defined as one of the boundary of the area created by the mesh elements cut by the physical interface. In Figure 9 an example in 2D for two different shapes of interface Γ is available. The surrogate is then a set of edges, where only two options are possible, the right and left side of the cut elements area. For 3D simulations the surrogate interface is a set of faces, and offer a wider range of choices for the definition of the surrogate interface.



(a) Flat interface

(b) cos/sin shape interface

Figure 9: Example in 2D of the identification of the elements cut by the physical interface

This strategy of shifting conditions has proven its robustness and accuracy for a variety of problems, including Navier-Stokes [3], free surface flows [90], and solid mechanics [91], with thorough analysis in [66] and [92] and mathematical analysis of the high-order version of the SBM in [67]. The method has not been tested on problems with phase-change transitions and is very recent in its development to moving boundaries, which is a computational challenge. Nevertheless, one can mention the recent work done on the incompressible Navier-Stokes flows with moving free-surfaces, where the nodes of the interface are moved using the flow velocity [90], the method is called **Weighted SBM** or **W-SBM**.

Considering the Stefan model the interface is defined by a discontinuity in the heat coefficients (specific heat, heat conductivity,...) and flux jump conditions. A formulation has been proposed in [80] for the steady heat equation whereas, in this work it is the unsteady problem which is considered, in which the front velocity is linked to the flux jump. The computation of this velocity requires accurate computation of the temperature gradient. To improve the accuracy, this gradient is considered as an unknown and the problem is proposed to be solved in its mixed form. This allows, as proposed in [93, 94], to perform an enrichment of the temperature leading to a quadratic approximation for both the temperature and the flux. In [95], the strategy has been employed with the SBM for the steady Darcy problem to recover at least a second-order accuracy for the primal quantity and its gradient.

Because of the discontinuity in flux at the interface (Stefan condition) and the moving front, instabilities could appear that could propagate in the whole domain. To study, the behavior of a perturbation at the interface (and on the temperature field), a linear stability analysis of the e-SBM and the analytical model is performed in the last Chapter of this thesis. Such study, has not been developed on the Stefan model yet for a melting configuration, but one can find material on the subject for faceted growth with application to horizontal ribbon growth in [96, 97, 98], in [99] for a three dimensional solitary waves model, or in [100] for a high order 3D method for the acoustic wave equation.

In this work, a new formulation for the unsteady Stefan model is provided that embeds the jump conditions defined at the interface. The method solves the temperature field and the gradient of temperature using a mixed formulation, and deals with the moving front and the imposition of the boundary conditions with the Shifted Boundary Method (SBM). The stability and robustness of the method are also detailed thoroughly. In the next section, the different chapters and sections of this manuscript are outlined.

3 Outline of the manuscript

In the first chapter of this manuscript all the elements necessary to implement the **Shifted Boundary Method** are introduced. In the first place, the development of the finite element method from both a mathematical and a numerical perspective is detailed. It enables the introduction of the **Galerkin method**, used to discretize the equations of the model into their finite weak formulation. In the following section, the distinction between a **conform** and an **embedded** resolution is introduced, along with all the required definitions to impose conditions using the Shifted Boundary Method. In the last section of this chapter, the **Nitsche's imposition** is defined, which allows to **impose conditions weakly** in the finite weak formulation.

In Chapter 2, a physical and mathematical description of the Stefan model is provided, which serves as the model used to test the **enriched SBM** (e-SBM). The conform and embedded schemes are compared in their construction to allow the reader a better understanding of the differences between the two approaches. Two types of interface conditions are compared to demonstrate their consequences in the resolution of the model and to determine the better-suited choice. In the following section, the method employed to achieve a global second-order accuracy in space, time, and for the interface location is introduced. The motion of the interface creates an area with missing values in the expanding phase (for melting: water) which needs to be handle with care. Handling this situation carefully is crucial to preserve the accuracy of the method built within the scheme and will be addressed in the last section of this chapter.

In Chapter 3, numerical results demonstrating the accuracy of the method described in Chapter 2 will be presented. Validation tests will emphasize the benefits gained through the enrichment of the primal variable and the enhancement of the accuracy of the associated flux. The method will also be tested for different types of interface shape to show the diversity of **application of the e-SBM**. To demonstrate the ability of the method to track the position of the front a simulation on a semi-infinite ice block is performed, that will show the preservation of the accuracy with the reconstruction techniques introduced in Chapter 2.

In **Chapter 4**, a **stability analysis** on the Stefan model is performed to express the stable modes of the model and the dispersive character of a perturbation at the interface and on the temperature field with the use of the e-SBM. The interface is now discretized and more complex geometries of interfaces can be handled. Only the case of a melting configuration will be discussed. The stability analysis will be performed with a fully unstructured 2D motion, on the **dimensionless form** of the model to allow a better interpretation of the results. It will also demonstrate that the shape of the interface does not impact the resolution, emphasizing the stability of the method.

Finally, a **Conclusion Chapter** will summarize the contributions of this work and its applications. Future work and perspective will also be discussed.

Chapter 1

The Shifted Boundary Method : A Finite Element Method

In this manuscript, numerical simulations are performed using the **F**inite **E**lement **M**ethod (**FEM**), a numerical method used for the resolution of systems of equations called **PDE** (**P**artial **D**ifferential **E**quation). This method is an essential tool for all the work presented here. In order to make the following content more accessible to a wider audience, the first section of Chapter 1 will cover fundamental concepts and key aspects of any FEM simulation. The development of the method from its beginning to its industrial use will be addressed, making of the **FEM** a huge source of research by its still recent development. The next section will be dedicated to the principle of the method itself and its mathematical formulation. Then, the **Shifted Boundary Method** (**SBM**) used to deal with non conforming mesh boundaries will be introduced. The Chapter will end on the definition of the **weak imposition** and the method used to impose conditions in the scheme : the **Nitsche's imposition**.

1.1 The Finite Element Method

The **FEM** is a numerical method designed to compute an approximation of the solution of a mathematical problem expressed by a system of **P**artial **D**ifferential **Equations** (PDEs). It converts the initial system of PDEs into a **finite weak form** (or finite weak formulation) where functions are defined in finite dimensional spaces.

1.1.1 Numerical development

The **FEM** was first developed for engineering purposes in the 1950's for the Aerospace industry by Boeing, following the development of new computers capable of solving more complex structural problems. The company has played an important role in the development of numerical applications of the **FEA** (**Finite Element Analysis**), the mathematical aspect behind the FEM. This advancement started by modeling airplane wings using triangular elements. However, the true numerical establishment of the method was done later on in the 1960's, when

analog computers were replaced by digital computers, which had the capacity to make thousands of operations simultaneously compared to their analog counterpart, only able to store a limited amount of data and which required specific knowledge to use them. At their time, analog computers were not suitable to a large scale development to the industry, see Figure 1.1.

Figure 1.1: Vizualization analog computer VS digital computer[10]

When digital computers made their appearance allowing the visualization of results, more complex problems were able to be solved, and the FEM became an accessible method to researchers and engineers. The enthusiasm behind FEM simulations were so important that **NASA** under United States government funding established a cooperative project to develop a unified structural analysis program based on the **FEA**. Other companies were involved such as the MacNeal-Schwendler Corporation (MSC), the Computer Science Corporation (CSC), the Martin Company (Martin Baltimore) and Bell Aerosystems in developing the **FEA** software known today as **NASTRAN** (**NA**sa **ST**ructural **AN**alysis), see [101]. This software is considered as the first full software dedicated to **FEM** and its applications.

It has later on been redeveloped by **MSC** for public usage, under the name **MSC**/**NASTRAN**, to cover the lack of user support and error bug corrections initially present within the original version of **NASTRAN** maintained by **NASA**. This version written in Fortran is the one used nowadays, and is one of the reasons why most of **FEM** codes are still largely developed in **FORTRAN** (**FOR**mula **TRAN**slation) as software translation is a very long and difficult task.

The general development of the method to the industry is finally out in the 1980's following the development of graphic processing which enabled engineers to study colored stress contours instead of looking at tabular outputs. Thanks to this advancement, the **FEM** became a tool used for general product design, allowing engineers to visualize physical prototypes and simulate their structural behavior subject to outside physical events. Two main advantages are seen by industries to choose **FEM** over other numerical techniques.

1. Being able to reduce physical prototypes;
2. Being able to consider multiple scenarios for the validity of a product or a system.

[10]https://guidancecorner.com/difference-between-analog-and-digital-computer/

These two advantages are mainly coming from the visualization properties of the **FEM**, which localizes element per element the behavior of the solution of the equations solved over a physical domain.

1.1.2 Mathematical development

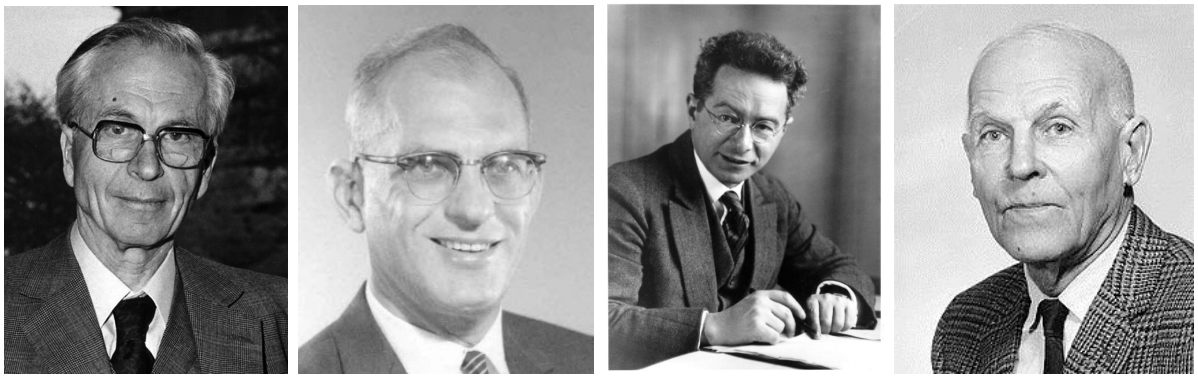


Figure 1.2: From left to right : John Argyris (1913 - 2004, Greece) - Ray W. Clough (1920 - 2016, USA) - Richard Courant (1888 - 1972, Germany/USA) - Alexander Hrennikoff (1896-1984, Russia/Canada)

The development of the **FEM** is largely linked to the emergence of digital computations. Back in 1950, only the biggest companies were able to afford computers robust enough to perform these simulations, explaining why the **FEM** has such a big history in its development within spatial and aerospace engineering, see [102].

The origin of the modern **FEM** is attributed to numerous researchers back in the 1950's. Professor **John Argyris** from both Imperial College, London, and the University of Stuttgart formulated the **classical force method** for stress analysis, using matrix format notations. The classical force method was developed to analyze the behavior of structures and can be summarized as follow:

1. Discretization of the domain of resolution into smaller elements;
2. Definition of an equilibrium equation for each node of the discretization based on the forces acting on the nodes;
3. Definition of compatibility equations to ensure that the displacement of each node is compatible with the deformation of the elements of the mesh discretization;
4. Resolution of the system of equations (equilibrium and compatibility). The unknowns of the system are the forces acting on the nodes and their displacement.

Argyris used matrices to explain the relationship between the forces and the deformation of the material for selected nodes in the structure, see [103]. Subsequently, the resulting system described the displacement as unknown values at designated nodes. This principle was named matrix displacement method. John Argyris's work is regarded as a foundational step in the development of the finite element method, see [104].

The idea of Argyris was presented by engineer **Jon Turner** [105], for a conference meeting at Boeing, where the idea was to generalize Argyris work to a two dimensional problem with elements being characterized by three nodes (current $\mathbb{P}_1$ FEM with linear triangular elements in $2D$).

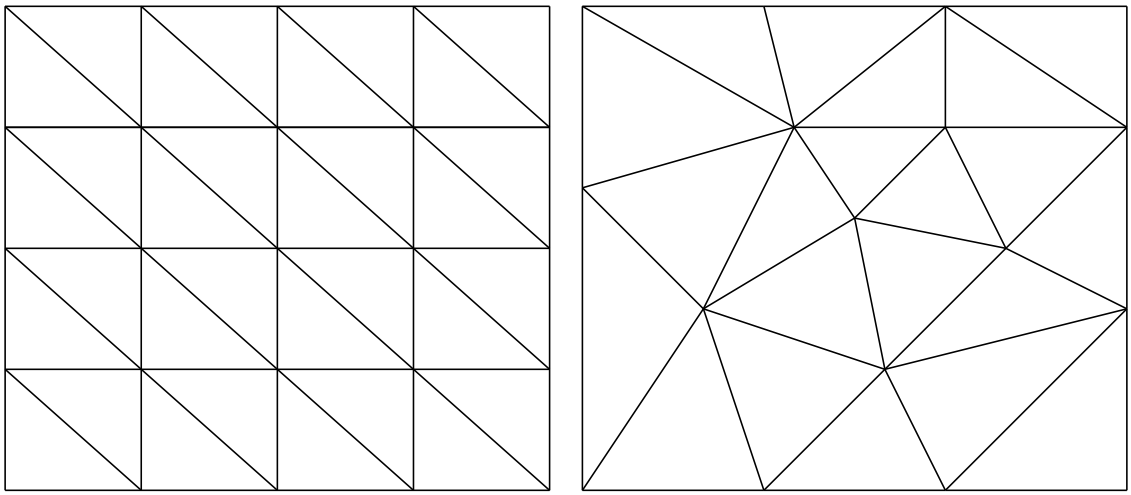
Considering the development of the theoretical aspect, it is under the guidance of Professor **Ray W.Clough** from the University of Berkeley that the idea was developed with a first paper published in 1956 and entitled "Stiffness and Deflection Analysis of Complex Structures", see [105]. Nevertheless, the term of "Finite Element Method" is only used four years later in his next paper entitled "The Finite Element Method in Plane Stress Analysis", see [106], which quickly became a reference. The concept was to allow the subdivision of a 2D structure into individual elements where the solution is solved independently on each of them, to be assembled later on, step by step into an overall structure preserving the initial conditions of the system.

Even if the overall method has been recently developed (compared to the History of Mathematics), it is important to note that some of the concepts present in the **FEM** can be associated to researchers that have taken place long before the 1950's. The **FEM** is considered a special formulation of the mathematics-based Ritz method (1909), which has been later on studied by **Richard Courant** (1943), but where his work could not expand further due to the lack of methods to solve large systems of equations, see [107]. The Russian-Canadian engineer, **Alexander Hrennikoff** (1941) is often associated alongside Courant as a pionner in the development of the FEM for his work on the lattice analogy (theory of elasticity) to model membranes and the plate bending of structures, see [108] (see Figure 1.2 for portraits, no picture of Jon Turner is available to the public). Important names such as **Galerkin**, **Rayleigh**, **Trefftz** and **Liebniz** can be added to the list of people who worked on such variational methods, and which has their names in some of the principle itself of the **FEM**. Even further back in time, the idea of solving problems on a split surface was already present such as in ancient Greece. **Archimedes of Syracuse** (287-2212 B.C) calculated the ratio of the circumference of a circle to its diameter by approximating the shape of a circle using straight lines. The more straight lines were used, the better the approximation was. This simple example shows that the development of computers helped mathematicians to contextualize concepts already known before their time.

Then, it is important to acknowledge that the development of the FEM was possible, only because its development happened alongside the development of computers and their capacity to solve bigger systems of equations, and to store more data [101]. These problems of power and storage are still a big issue for complex simulations, leading nowadays to the use of parallel calculus and the emergence of quantum computers. In the next section, the principle of the FEM is introduced and its concepts of local description, core idea of the method is described.

1.1.3 Principles

PDEs can be solved using numerical methods or analytical methods. Analytical methods are only relevant for academic purposes and mathematical analysis, while numerical methods are considered for engineering applications. **FEM** falls into the category of numerical methods. For any PDE where the solution is wanted on a bounded domain Ω the principle is to approximate the solution on a subdivision of that domain called **mesh**. The components of the mesh are referred as **elements**, while the subdivision is called **tessellation**. The tessellation does not have to be homogeneous i.e., that all of the elements of the mesh do not have necessarily the same characteristic length. For example in Figure 1.3(a) using triangles for a $2D$ domain, one can see that all the elements composing the mesh are of the same shape and size with a well organized structures, while in Figure 1.3(b) all the elements are of different sizes and

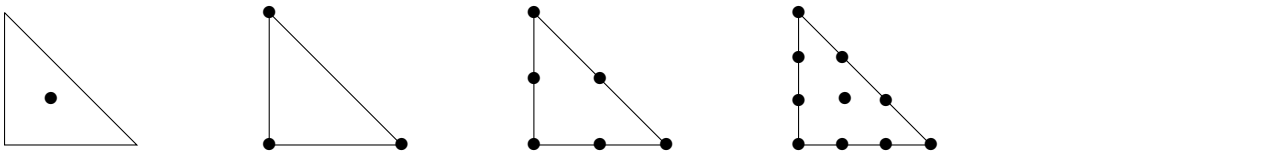


(a) Structured Mesh (b) Unstructured Mesh

Figure 1.3: Mesh Visualization - Comparison between an unstructured and structured mesh - 2D Simulation

with no specific organization in the structure. The solution of the considered problem is then approximated on points defined independently on each of the elements composing the mesh. The shape of the elements characterizes the simulation by the choice made for the position of the nodes where the approximated solution will be calculated. Moreover, the number of points defined the accuracy of the method.

The main idea of the FEM is to start from a **local description**, element per element with local matrices which are then assembled into the global matrix describing the system of equations. The first step consists of transforming the problem into its **weak form** also called **weak formulation** and to associate a finite weak form defined on a space of approximation. To this end, the finite dimensional approximated space needs to be determined first. The functions spanning this space are called **basis functions** or **shape functions**. For a 2D triangular resolution a classical choice is polynomial functions. The degree of the polynomial will depend on the desired accuracy.

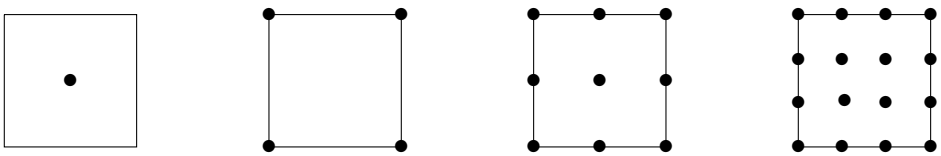


$\mathbb{P}^0 := \{1\}$ $\mathbb{P}^1 := \{1, X, Y\}$ $\mathbb{P}^2 := \{1, X, Y, XY, X^2, Y^2\}$ $\mathbb{P}^3 := \{1, X, Y, XY, X^2, Y^2, XY^2, X^2Y, X^3, Y^3\}$

Figure 1.4: Basis functions and degrees of freedom for triangular elements - 2D simulation

As presented in Figure 1.4, triangular elements are a classical choice for a $2D$ simulation. The number of nodes on each element is the same as the dimension of the space of approximation $\mathbb{P}^k$. Another frequent option for 2D simulations is the family of cubic elements, where their version is presented in Figure 1.5 for $\mathbb{Q}^k$ approximation spaces.

For $3D$ simulations, the most popular elements are **prismatic, tetrahedral, pyramidal** and **hexahedral** elements. In Figure 1.6, the second-order Lagrange elements are presented. Black nodes are degree of freedom naturally present within the mesh definition, white nodes are degree



$\mathbb{Q}^0 := \{1\}$ $\mathbb{Q}^1 := \{1, X, Y, XY\}$ $\mathbb{Q}^2 := \{1, X, Y, XY, X^2, X^2Y, Y^2, XY^2, X^2Y^2\}$ $\mathbb{Q}^3 := \{1, X, Y, XY, X^2, Y^2, XY^2, X^2Y, X^3, Y^3, X^2Y^2, X^3Y, X^3Y^2, X^3Y^3, Y^3X, Y^3X^2\}$

Figure 1.5: Basis functions and degrees of freedom for cubic elements - 2D simulation

of freedom associated to mid-point edges, and blue nodes are nodes located in the middle of the cubic surfaces. When the grey nodes are not taken into account the elements are referred to as serendipity elements, and only the nodes associated to the edge inter-connectivity between the elements of the mesh are considered, see [109].

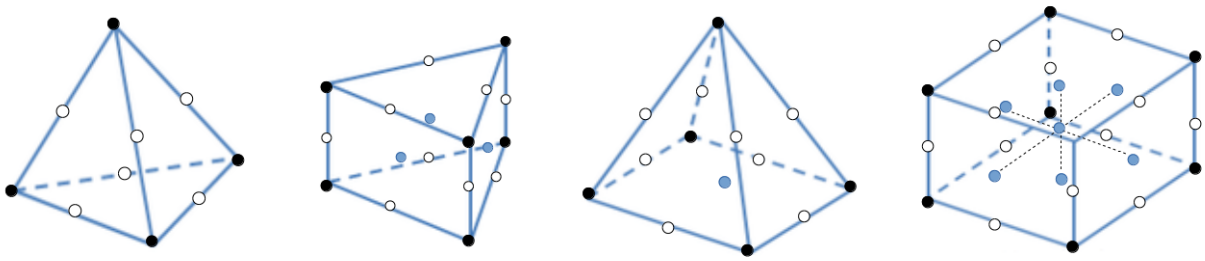


Figure 1.6: 3D Elements Lagrange, from left to right : Tetrahedral, Pyramidal, Hexahedral and Prismatic element and their associated degree of freedom

Elements presented in Figures 1.4, 1.5 and 1.6, with test functions from a $\mathbb{P}^k$ space or $\mathbb{Q}^k$ space ($k \in \mathbb{N}$), are referred to as **Lagrange elements** i.e., only the properties of the basis functions are considered. Other type of finite elements exist, **Hermite elements** are one of them. In contrast to Lagrange elements the basis functions of Hermite elements also satisfy properties for different orders of their derivatives, with the highest order of derivatives characterizing the type of Hermite element being considered.

The characterization of an element besides the choice of the space defining the basis functions is the position of the nodes within the element. Two elements with the same geometry will not correspond to the same **FEM** if the position of the nodes is not the same, even with an identical number of nodes. This **number of Degree of Freedom** (nbDoF) is characterized by the size of the finite element space, see Definition (1.1) for an example of $\mathbb{P}^k$ and Definition (1.2) for an example of $\mathbb{Q}^k$ polynomials.

`Lagrange polynomial elements,` $\mathbb{P}^k$ `:`

$$\begin{array}{lll} \underline{\text{Line}} & \text{nbDoF} = k+1 & 1D \\ \underline{\text{Triangle}} & \text{nbDoF} = \dfrac{(k+1)(k+2)}{2!} & 2D \\ \underline{\text{Tetrahedral}} & \text{nbDoF} = \dfrac{(k+1)(k+2)(k+3)}{3!} & 3D \end{array} \tag{1.1}$$

where $n! = \prod_{i=1}^{n} i$.

`Lagrange polynomial elements,` $\mathbb{Q}^k$ `:`

$$\begin{array}{lll} \underline{\text{Line}} & \text{nbDoF} = (k+1) & 1D \\ \underline{\text{Square}} & \text{nbDoF} = (k+1)^2 & 2D \\ \underline{\text{Hexahedral}} & \text{nbDoF} = (k+1)^3 & 3D \end{array} \tag{1.2}$$

The basis functions of a **Lagrange element** are designed to satisfy the following properties. Their numbers need to match the dimension of the considered space, see Equations 1.1 and 1.2 for an example. Then, each Lagrange basis function is associated to one of the degrees of freedom of the element such that the following property takes place.

Let N be the dimension of the space of approximation considered, x_i, $i \in \{1, N\}$, the set of degrees of freedom of an element of the mesh and ϕ_i, $i \in \{1, N\}$, the set of functions generating the space of approximation on that element. Then the functions ϕ_i, $i \in \{1, N\}$ are called basis functions for that element if they satisfy

$$\begin{cases} \phi_i(x_k) = 1 & \text{if } k = i \\ \phi_i(x_k) = 0 & \text{if } k \neq i \end{cases}. \tag{1.3}$$

Computationally, the basis Lagrange functions are defined in two stages. First they are defined on a reference element. This step is only performed once as the reference element is the same for all elements of the standard mesh (mesh composed of only one type of element's shape.). For a 2D simulation and on a triangular mesh the reference element is the unitary triangle $\{(0,0),(1,0),(0,1)\}$. To obtain the basis functions on any element of the mesh, the basis functions defined on the reference element are modified by affine transformation to correspond to the position of any other element of the mesh, see Figure 1.7.

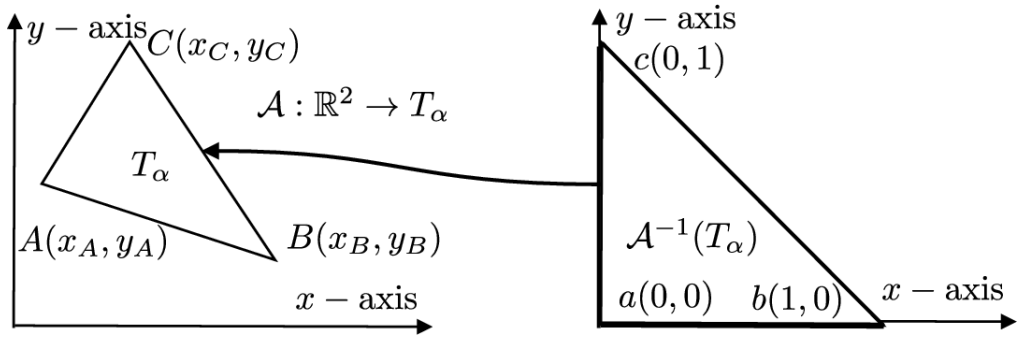


Figure 1.7: Affine transformation from the reference triangle to a mesh element, [110]

1.1.4 Variational formulation

Different methods exist to transform a continuous model into a discrete one. In this manuscript, the most classical option and the one used in the following chapters is explained : **the Galerkin method**. Instead of looking for an exact solution in an infinite dimensional space, an approximated solution is defined in a finite dimensional space. For the sake of clarity, the method is applied on an example, with Problem (1.4). For the formalism and the definition of the inner-product between functions see Section 1.3.2.

The domain of resolution Ω is an open, bounded set of $\mathbb{R}^d$ ($d \in \mathbb{N}^*$), and the right-hand side is a function g of $\mathbb{L}^2(\Omega)$. The problem to be solved consists of looking for a solution

$u \in H_0^1(\Omega) := \{u \in \mathbb{L}^2(\Omega), \nabla u \in (\mathbb{L}^2(\Omega))^d |\ u = 0 \text{ on } \partial\Omega\}$ such that

$$\begin{aligned} -\Delta u(x) &= g(x) \text{ in } \Omega, \\ u(x) &= 0 \quad \text{on } \partial\Omega. \end{aligned} \tag{1.4}$$

To obtain the variational problem associated to Problem (1.4), v a test function is chosen in $H_0^1(\Omega)$. The Problem (1.4) is then multiplied by the function v on both side of the equal sign and integrated,

$$-\int_\Omega v\Delta u(x)\ dx = \int_\Omega g(x)v(x)\ dx\ . \tag{1.5}$$

The second step to obtain a valid variational problem consists of imposing the remaining boundary conditions and in reducing the order of the differential operators of the integrated problem (1.5), see [111]. Boundary conditions can be applied in two different ways. A **strong imposition** is referred to a method where the test function v is chosen to satisfy the boundary conditions of the system. On the opposite, the **weak imposition** is referred to conditions being imposed through the **variational form**. This can refer to terms coming naturally from the integration by parts (Green's theorem), the Nitsche's imposition (see Section 1.3), the addition of penalty terms, and other techniques.

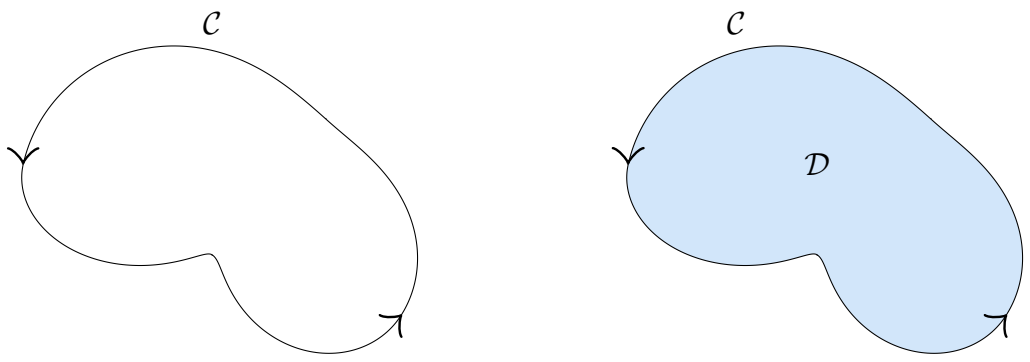
To properly modify Equation (1.5), the **Green's theorem** is introduced. Different variations of its definition are possible depending on the regularity of the functions and the domain of resolution at stake. In the context of application of this thesis, Theorem 1 describes the relationship between a line integral (integral on a boundary) and a surface integral. The Green's theorem in (1.6) is defined for the case where the dimension $d \geqslant 2$, and where one of the functions (u or v) is taken with a compact support in $\bar{\Omega}$. This criterion ensures that the integrals are defined even when Ω is not a bounded domain, see [112].

Theorem 1 (Green's Theorem)**.** *Let Ω be an open set of class $\mathcal{C}^1$ of dimension d, and $u, v \in \mathcal{C}^1(\bar{\Omega})$ with u or v a function compactly supported in $\bar{\Omega}$. For $j \in [\![1, d]\!]$ one has*

$$\int_\Omega \partial_j u\, v\ dx = -\int_\Omega u\, \partial_j v\ dx + \int_{\partial\Omega} u\, v\, n_j\ d\sigma. \tag{1.6}$$

The parameter $n_j = n \cdot e_j$ is the j-th coordinate of n, the outward unit normal vector on $\partial\Omega$ in the canonical basis of $\mathbb{R}^d$. The designation $d\sigma$ is the measure used for the integration on the boundary $\partial\Omega$.

Remark 1. The idea behind the Green's theorem is to observe than if a curve $\mathcal{C}$ represents an oriented closed curve, then the integral of a function F on $\mathcal{C}$ represents the circulation of F around $\mathcal{C}$, see Figure 1.8(a).



(a) Visualization of the closed curve $\mathcal{C}$ (b) Visualization of the surface area $\mathcal{D}$

Figure 1.8: Domain of resolution

Theorem 1 allows to transform the line integral into a surface integral over the region inside $\mathcal{C}$, the surface $\mathcal{D}$ in Figure 1.8(b). On a physical approach, the Green's theorem states that the macroscopic circulation around the curve $\mathcal{C}$ is equal to the sum of the microscopic circulation that is inside $\mathcal{C}$ i.e., the surface $\mathcal{D}$ in Figure 1.8, see [111] for more information.

The variational problem associated with Problem (1.4) is obtained using Theorem 1 on the integrated problem (1.5) and is stated as follows :

Find $u \in H_0^1(\Omega)$ such that

$$\forall v \in H_0^1(\Omega), \quad \int_\Omega \nabla u(x) \cdot \nabla v(x)\ dx = \int_\Omega g(x)v(x)\ dx, \tag{1.7}$$

where the boundary term vanished as $v \in H_0^1(\Omega)$.
A variational problem is often denoted through an abstract form.

Let

- $a(u, v)$ be a continuous bilinear form on $H_0^1(\Omega) \times H_0^1(\Omega)$ such that

$$a(u, v) = \int_\Omega \nabla u(x) \cdot \nabla v(x)\ dx,$$

- $f(v)$ be a continuous linear form on $H_0^1(\Omega)$ such that

$$f(v) = \int_\Omega g(x)v(x)\ dx.$$

Then, Problem (1.7) can be rewritten as follows

Find $u \in H_0^1(\Omega)$ such that

$$a(u, v) = f(v), \quad \forall v \in H_0^1(\Omega). \tag{1.8}$$

Now that a valid variational problem has been defined, the Galerkin method which consists of the approximation of Problem (1.8). The space of infinite dimension $H_0^1(\Omega)$ denoted V is approximated by a space a finite dimension V_h. More precisely, the **standard Galerkin method** also called **continuous Galerkin method** consists of an approximation where the space of approximation V_h satisfies $V_h \subset V$.

Find $u_h \in V_h$ such that

$$a(u_h, v_h) = f(v_h), \quad \forall v \in V_h. \tag{1.9}$$

Let $N = dim\ V_h$ be the dimension of the space of finite dimension V_h and $\{\zeta_i\}_{1 \leqslant i \leqslant N}$ a base of functions such that $V_h = \{\zeta_1, \zeta_2, ..., \zeta_N\}$. Then, Formulation (1.9) corresponds to a linear system of equations of the form $AU = F$. The matrix A, and second member F of the linear system associated to (1.9) are defined by

$$\begin{array}{ll} A_{i,j} = a(\zeta_i, \zeta_j) & 1 \leqslant i \leqslant N,\ 1 \leqslant j \leqslant N, \\ F_i = f(\zeta_i) & 1 \leqslant i \leqslant N. \end{array} \tag{1.10}$$

The approximation u_h of the solution u to Problem (1.4) can be decomposed on the basis $\{\zeta_i\}_{1 \leqslant i \leqslant N}$ as follow:

$$u_h = \sum_{i=1}^{N} u_i \zeta_i \tag{1.11}$$

where the vector $U = (u_i)_{1 \leqslant i \leqslant N}$ is composed of the value of u_h on the DOF of each element of the mesh discretization. For more details on the Galerkin method and the theory of application of the FEM see [113].

So far, the concept of the Finite Element Method and how it allows to obtain an approximated solution to a system of equations has been discussed. The order of the method has also been introduced and Chapter 2 will expand this aspect by proposing a method allowing a second order accuracy for both the primal variable and its gradient.

The FEM is a versatile tool, applicable to a lot of physical situations and which allows simulations on complex geometries. An important aspect of the method is the choice made to impose boundary conditions. In the next section, the method used to impose boundary conditions in the context of non-conforming mesh boundaries is expressed. The proposed method falls into the category of embedded approaches: **the Shifted Boundary Method**.

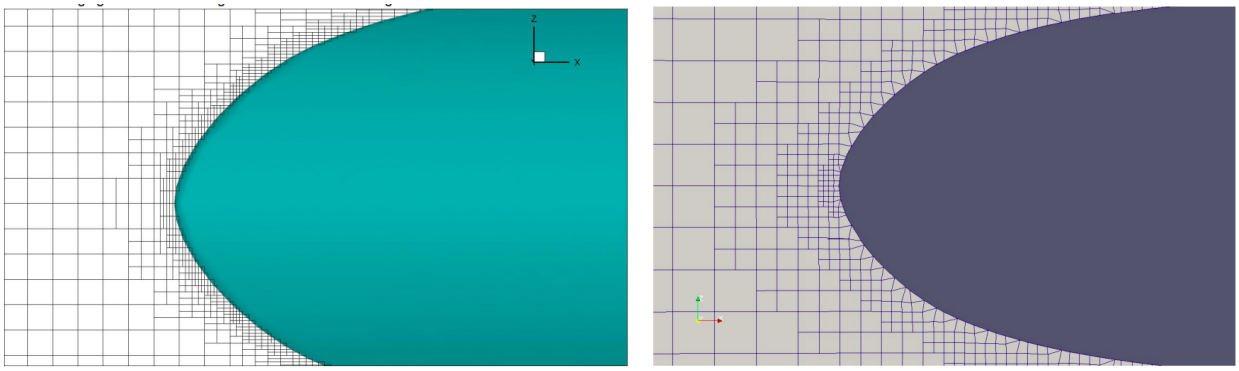
1.2 The Shifted Boundary Method

Performing numerical simulations on systems with boundaries requires a precise definition of the physical domain's geometry and the tessellation of that domain can be a tedious task, especially when boundaries are moving. In this direction, Immersed and Embedded boundary methods are a popular and active field of research, since Charles S. Peskin's work in 1972 [34]. In this work, to avoid confusions, **Embedded Boundary Methods (EBM)** is referred to any methods associated with **direct** or **continuous forcing techniques on non body fitted meshes**. The main advantage of these methods is to avoid explicit meshing of the moving boundaries or interfaces [34, 114, 115, 116, 117, 40]. Some body fitted methods have proven their robustness in the Lagrangian framework [18, 19, 20, 21], or in the Arbitrary Lagrangian Eulerian formalism [22, 23, 24, 25, 26, 27]. These methods use mesh deformations to match the interface displacement and require remeshing procedures to handle large deformations. This process can be time consuming, especially regarding highly parallel computations. In contrast, EBM only requires a fixed mesh on which the moving boundary must be considered. The main challenge of EBM is then to correctly and accurately compute the moving boundary position and/or the boundary conditions. In Section 1.2 of this chapter, the **Shifted Boundary Method** introduced by Main and Scovazzi for Laplace and Stokes equations [2] is described. The method will be adapted in Chapter 2 to account for the moving interface present in the Stefan model. The Shifted boundary Method will also be compared to the classical cut-cell method, one of the most popular EBM used.

1.2.1 Principles

In the field of FEM, the mesh description appears as an essential part of any model. The choice of the elements making the mesh will be used to define the type of basis/test functions employed, which will also determine the accuracy of the method. One possibility for complex geometries is to perform mesh adaptations around the areas where it is most needed. But, performing mesh adaptations on a complex geometry can lead to the deformation of the elements in sharp areas, as it can appear with the use of the cut cell method in Figure 1.9.

[11] https://www.karalit.com/docs/brochures/IBCFD_230915.pdf



(a) Before Adaptation (b) After Adaptation

Figure 1.9: Cut-cell method applied around the geometry of an airplane wing[11]

The cut-cell method consists of solving the problem on a Cartesian grid for the majority of the domain, and in the application of a specific treatment for the cells cut by the boundary of the immersed body (interface, subdomain, phase-change front, shock, solid body,...). Those cells are deformed to take into account the geometry of the boundary, see Figure 1.9. It is precisely the adaptation process which can lead to issues in the resolution, especially in $3D$. The elements cut are changed into new ones, where only the part of the element outside of the body is kept, meaning that the geometry of the considered body will defined the geometry of the mesh elements in this area. It is often necessary to refine these elements after adaptation to the boundary, which is referred to as the "cut-cell issue". The cut-cell method is a representative example of the different problems that can occur when adapting a mesh to a specific geometry. These problems include loss of accuracy, deformation of elements, time-consuming post-processing methods, remeshing for time-dependent problems, and a loss of accuracy in the geometry description. This problem becomes even more challenging when the considered boundary is moving, especially in higher-dimensional problems. Even if countermeasures have been found to solve those issues new areas of research are focusing on a different approach : **immersed/embedded resolution**.

The choice of doing an **embedded** resolution appears as an interesting option to prevent those types of problems in the first place. An embedded resolution consists of adapting the definition of the boundary conditions instead of adapting the mesh to the geometry of the boundary. In this range of methods, and for the work presented in this manuscript the **Shifted Boundary Method** also called **SB method** or **SBM** is considered. The **SBM** consists of imposing boundary conditions not as they are defined in the initial system on the physical boundary, but by shifting the values to a **surrogate boundary** defined as edges in $2D$ or faces in $3D$ of the considered mesh (see Figure 1.10 for an example in $2D$). The method is a valuable asset for solving time dependent problems with moving boundaries, which prevents to perform any remeshing step at each displacement of the physical boundary.
The main concern to apply correctly the method is to prevent any reduction of the convergence rate by doing a modification of the boundary conditions, which without adjustment are only first order accurate. The other preoccupation is to define the **surrogate boundary**, a numerical boundary that is conformed to the mesh discretization. There are different ways to define the surrogate especially in 3D, and one of them is to use the closest boundary to the physical boundary, using a closest-point projection algorithm, or a level set description of the boundary, see [80].

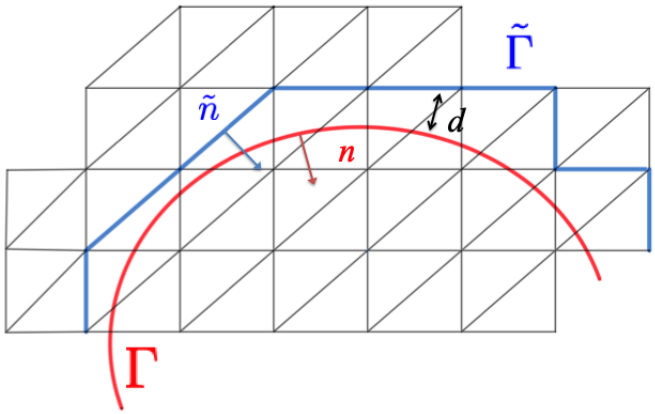


Figure 1.10: Visualization of a true/physical boundary Γ (red) and its associated surrogate $\tilde{\Gamma}$ (blue) conform to the mesh discretization[12]

1.2.2 Definition

The **SBM** can be summarized into **two steps** explained in details in [2]. In this section, the method is explained in a context of problems with interfaces which are boundaries embedded inside the domain of resolution, see Figure 1.11.

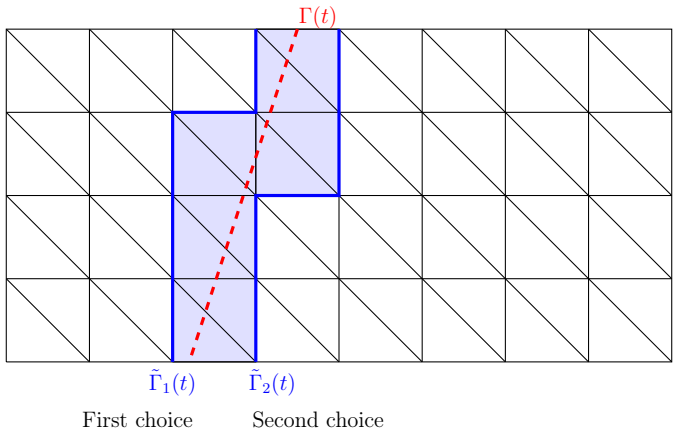


Figure 1.11: Surrogate definition associated to an interface embedded inside the domain of resolution

- The first step in the SBM consists of the definition of an approximated interface, conformed to the grid, called the **surrogate interface** (usually referred to as $\tilde{\Gamma}$), on which the interface conditions will be imposed. The surrogate interface can be seen as a numerical interface replacing the physical interface in the method of resolution. A mapping is employed to link the physical interface to its surrogate and is used to modify the imposed conditions to account for the discrepancy between the two interfaces.
- The second step is to enforce weakly, by adding terms in the weak formulation, the interface conditions using the **Nitsche's method**, see Section 1.3.

[12]workshop IMB, 27/01-28/01 2020, High order immersed boundary method: Shifted Boundary Method (SBM), H. Beaugendre, M. Colin, L. Nouveau, M. Ricchiuto

To visualize the method the focus is on the specific case of a $2D$ domain, but the **SBM** remains suitable for any higher dimensional problems.

Let the physical interface Γ be an embedded grid that does not conform to the mesh discretization. Its surrogate $\tilde{\Gamma}$, is composed of the edges (or faces if the dimension of the problem is greater than 2) that are the closest to the physical interface. The choice has been made to identify all the elements intersected by the physical interface (blue area in Figure 1.11) and to regard $\tilde{\Gamma}$ as one of the possible combinations of the outer edges of these elements, with only two possible combinations in 2D, as shown in Figure 1.11.

Remark 2. The position of the physical interface in the domain needs to be taken into account to define its surrogate, especially if the method is applied for any outer boundaries. Through this work the main focus is to model problems with interfaces. If the **SBM** is applied for the outer boundary of a domain, then the outside geometry will not be respected. When the boundary get deformed through time, a larger meshed area should be considered. An option would be to mesh a box surrounding the initial domain, a box that could also move with time, if the considered boundary is moving. A similar method was employed by Peskin in [36], in which the heart geometry is embedded in a meshed box and where the outside of the domain of resolution is considered as a fluid to accommodate the displacement of the boundary, see Figure 5.

Numerically, to identify the blue area in Figure 1.11 and define a corresponding surrogate interface a signed distance function is used (see Figure 1.12) and defined on every vertex of each triangle in the mesh.

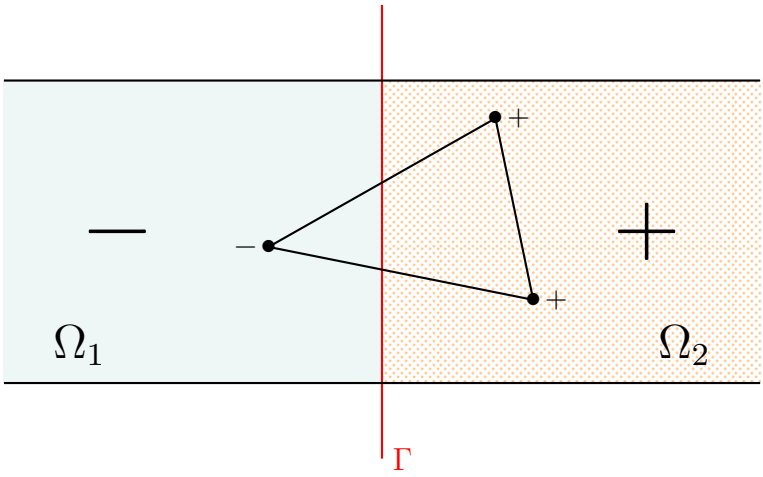


Figure 1.12: Convention of the signed distance function for a $2D$ case

The distance function is used to define a specific status for every triangle in the mesh, as indicated in Table 1.1. A negative node belongs to Ω_1, while a positive node belongs to Ω_2, see Figure 1.12.

A status is associated to each element by counting the number of vertices with a positive distance to the physical interface, see Table 1.1. A point that falls exactly on the physical interface is associated with the same case as a node with a negative distance, belonging to Ω_1, as depicted in Figure 1.12. This approach is going to be detailed in this section for 2D simulations but the distance sign function exists for higher dimension problems as well.

In Figure 1.12 a status of -1 defines an element that belongs to Ω_1, while a status of 2 means that the considered element belongs to Ω_2, see Figure 1.12. Elements of status 0 and 1 are the ones intersected by the physical interface.

Number of Positive node	Status of the element
0	-1
1	0
2	1
3	2

Table 1.1: Identification of the position of a triangular element in regards to the physical interface Γ

Two types of interface shapes were examined and implemented for $2D$ simulations in this work:

1. flat interfaces or circular interfaces defined by a radius value;
2. interfaces discretized by a set of nodes.

In the first case, the position of a node is easier to obtain as the projection is made onto a straight line or a circle. On the contrary, for the second case it is more difficult, as the identification of the area in which the node is present required more work. Multiple techniques have been tested throughout this work. The method of projection stays similar for all of them but they differ in their procedure for defining the sign associated to the distance function. The method presented in this Chapter 1 has been used for the results presented in Chapter 3. The stability analysis presented in Chapter 4 required a more flexible approach especially on circular domains. The method used for discretized interfaces is presented in Section 4.7.4.a of Chapter 4 with corresponding results. To represent the versatility and the various options available in this part of the SBM definition process, more than one method is detailed in this work. These methods are covered both in this current section and in Section 4.7.4.a. This choice is purely a numerical choice and does not impact the definition of the method.

In the following a method applied to the first type of interface geometry is presented. A flat interface is considered on a rectangular domain. In the case of a circular interface the method of identification of the position of an element is trivial; the coordinates of a node are compared to the radius that defines the physical interface.

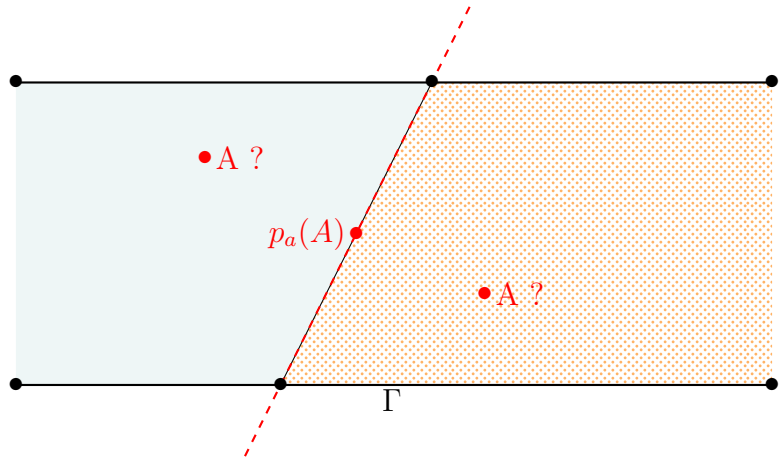
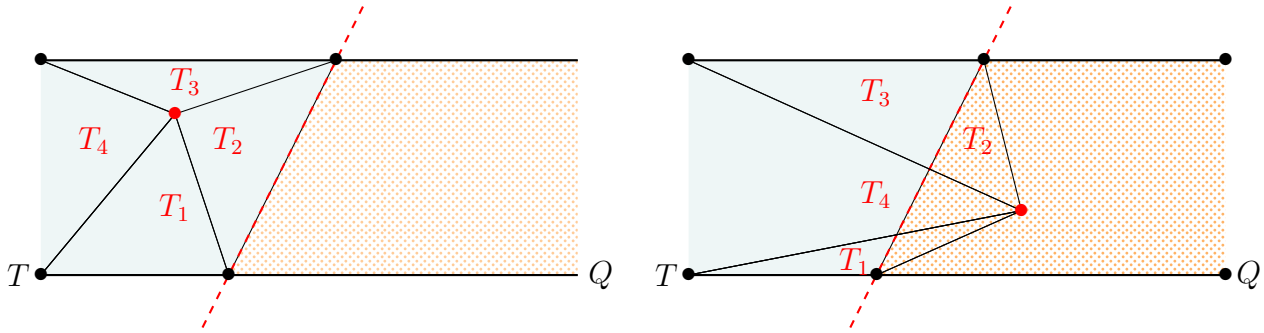


Figure 1.13: Visualisation of the problem of identification of domain

Let p_a be **the projection** of a mesh node A onto the physical interface Γ. The coordinates of the node A are known as well as the coordinates of $p_a(A)$ which is its projection.



(a) Node belonging to domain T

(b) Node belonging to domain Q

Figure 1.14: Identification of the position of a node compared to its projection

The projection $p_a(A)$ is defined on the physical interface. The straight line associated the interface cuts the domain of resolution into two subdomains. To know if a node A is defined in one domain or the other one can look at the sum of the areas between A and the boundaries of one of the subdomain (T or Q in 1.14). For example in Figure 1.14.a the sum of the areas of the triangles T_1,T_2,T_3 and T_4 is equal to the area of domain T (green domain in Figure 1.14), meaning that the considered point belongs to the domain T. In Figure 1.14.b the sum of the areas of the triangles T_1,T_2,T_3 and T_4 is not equal to the area of T and the considered node belongs to the domain Q. This method allows to associate a sign to the distance function, it is an easy method to implement.

Remark 3. The method presented uses area-based arguments detailed for 2D simulations. However, while no 3D simulations are available at this stage for the presented work in this thesis there is potential for the development of this procedure using volume-based arguments.

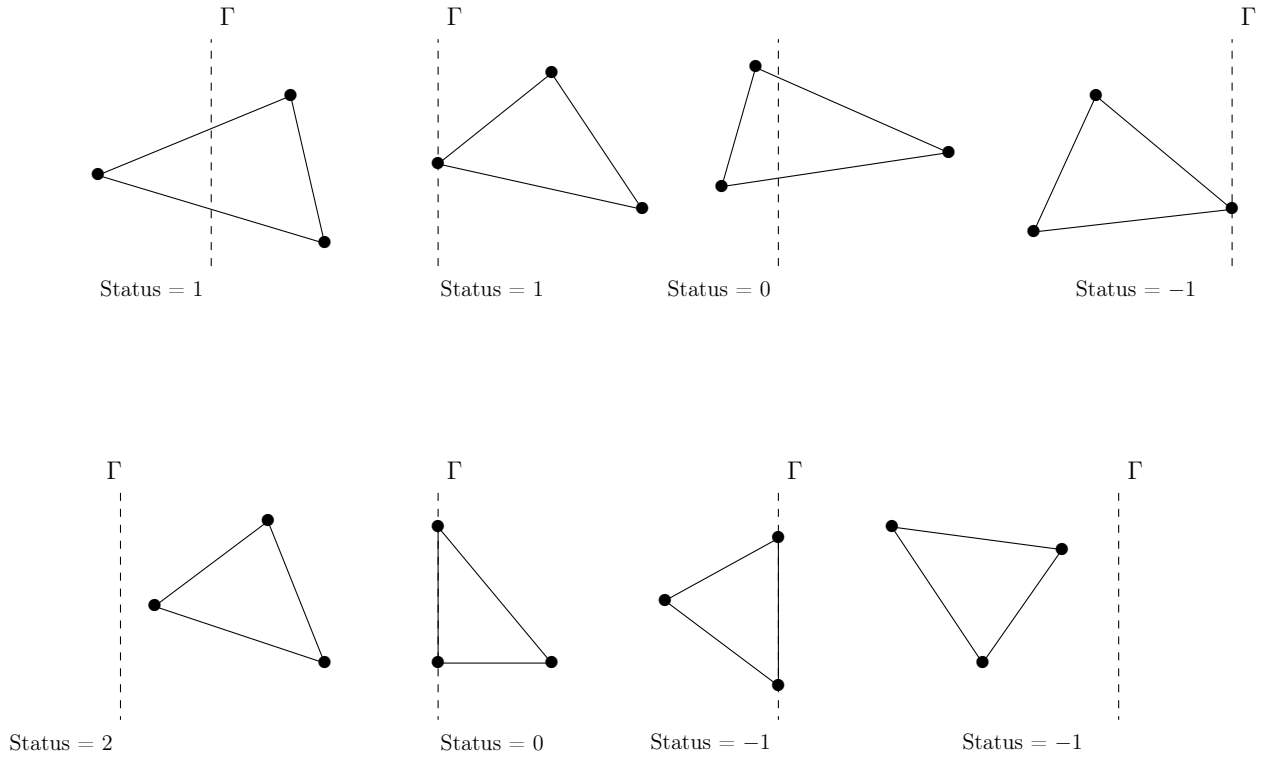


Figure 1.15: Possible configuration of a mesh element in regard to the physical interface position

Once a sign has been assigned to the distance function, the next step in any method of definition of the surrogate interface is to update the reference of the elements in terms of their status to the physical interface. This procedure allows to distinguish the two areas present in the domain. All the elements with status 1 and 0 are considered as being intersected by the physical interface, making elements with status 2 and -1 associated to two different sides of the boundary (blue area in Figure 1.11). Then, the definition of the surrogate consists of the treatment of the cut elements (see Figure 1.11) with status 1 or 0 (see Figure 1.15), and in which area Ω_1 or Ω_2 they are put into. For a $2D$ simulation, placing the cut elements with elements of status 2 defines $\tilde{\Gamma}_1(t)$ as the surrogate interface, while elements with the status of -1 make $\tilde{\Gamma}_2(t)$ the surrogate interface, see Figure 1.11. Taking $\tilde{\Gamma}_1(t)$ or $\tilde{\Gamma}_2(t)$ as the surrogate interface does not impact the resolution. Nevertheless, a convention should be established at the beginning of the simulation to ensure that the same definition is used for every new definition of the surrogate boundary when it is moving.

The following convention is now defined. Γ will be the standard notation to refer to the physical interface, while $\tilde{\Gamma}$ will always be its associated surrogate, see Figure 1.11. Moreover, the use of the symbol $\tilde{}$ over a notation will always refer to a quantity defined on the surrogate $\tilde{\Gamma}$, while the same notation without $\tilde{}$ will refer to the same quantity on the physical interface Γ. In order to connect the two interfaces, a mapping $\mathbf{M}$ is introduced to link the surrogate interface $\tilde{\Gamma}$ to its physical interface Γ

$$\begin{aligned} \mathbf{M} \ : \ \tilde{\Gamma} &\to \Gamma \\ \tilde{\boldsymbol{x}} &\mapsto \boldsymbol{x} \quad . \end{aligned} \tag{1.12}$$

The mapping $\mathbf{M}$ is used to deduce boundary conditions on $\tilde{\Gamma}$ from the boundary conditions on Γ, via Taylor expansions. In this work, the orthogonal projection operator is chosen. Even if the uniqueness of the projection is not ensured in a non-convex space, it should be noted that from a numerical perspective, one can introduce various criteria that allow the use of the orthogonal projection. The numerical precision (double precision) is often sufficient to prevent the definition of multiple projections for the same node. In cases where the problem arises, remeshing around the interface area is usually enough to ensure the uniqueness of the projection.

To maintain generality, a specific situation may arise where a node has no projection at all, as depicted in Figure 1.16. This situation occurs when the interface discretization includes sharp areas (see Perturbed interfaces in Chapter 4). In this situation the projection is defined using vector relationships to identify the two edges where the problem has appeared. Then, the projection is defined as the common node between these two edges. In that case, the projection becomes a node of the physical interface discretization.

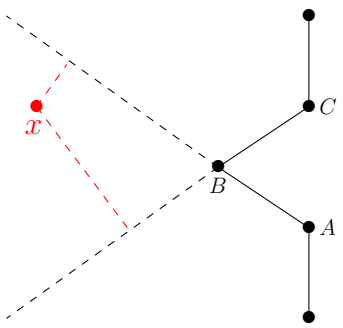


Figure 1.16: Node without orthogonal projection on the boundary

Numerically, it consists of looking at the projection on every edges that define the physical interface. If the projection is on one of the edges no problem arises. If the projection is on none

of them, then for every edge the information about the position of the projection in relation to the two nodes defining the edge is retained. If the situation arises where on the previous edge (from the loop on the edges of the interface discretization) the projection was defined ahead of the point B of an edge $\overrightarrow{AB}$, and on the next edge the projection is defined before the point B of the edge $\overrightarrow{BC}$, it indicates that the node is between the two edges. This is exactly the situation described in Figure 1.16. In this particular case the point B defines the projection of x onto the physical interface Γ.

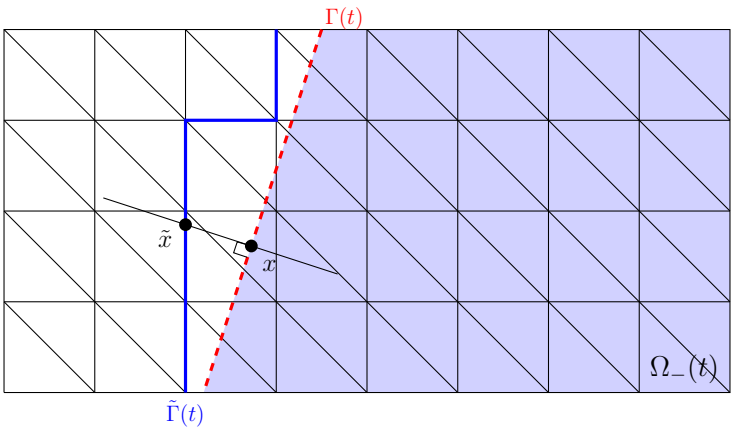


Figure 1.17: Definition of the mapping by orthogonal projection onto Ω_-

With the definition of the projection it is now possible to introduce the distance vector function (see Figure 1.17), measuring the distance between Γ and $\tilde{\Gamma}$

$$\boldsymbol{d_M}(\tilde{\boldsymbol{x}}) = \boldsymbol{x} - \tilde{\boldsymbol{x}} = [\mathbf{M} - \mathbf{I}](\tilde{\boldsymbol{x}}) \quad . \tag{1.13}$$

As a consequence of Definition (1.13), the distance vector $\boldsymbol{d}$ is aligned with the normal vector on Γ such that

$$\boldsymbol{d} = ||\boldsymbol{d}||\ \boldsymbol{n}, \tag{1.14}$$

where $||d||$ denotes the euclidean norm in $\mathbb{R}^d$. Owning these definitions, it is possible to define any functions on Γ as a function of $\tilde{\Gamma}$ by doing an extension of the original function. Indeed, if ψ is a function initially defined on Γ, then its extension $\tilde{\psi}$ on $\tilde{\Gamma}$ is given by

$$\tilde{\psi}(\tilde{\boldsymbol{x}}) = \psi(\mathbf{M}(\tilde{\boldsymbol{x}})) \quad . \tag{1.15}$$

In the next Section, the last step concerning the definition of the SBM is introduced. More precisely, the imposition of the interface conditions on the surrogate interface $\tilde{\Gamma}$ will be explained in detail.

1.2.3 Treatment of the Boundary Conditions

In this section, the method of imposition of the interface conditions on the surrogate interface is discussed. Initially, the interface conditions are only valid on Γ and not on the surrogate $\tilde{\Gamma}$. The idea of the Shifted Boundary Method is to determine the interface conditions on $\tilde{\Gamma}$ which are compatible with the interface conditions of the original PDE on Γ. For that purpose, Taylor expansions between Γ and $\tilde{\Gamma}$ are used, the order of the development in the Taylor expansion characterizes the accuracy of the imposed conditions on the surrogate interface $\tilde{\Gamma}$.

If one assumes that on the physical interface Γ the unknown $u(\boldsymbol{x})$ has to satisfy the Dirichlet

condition $u(\boldsymbol{x}) = g(\boldsymbol{x})$, where g is a given function with sufficient regularity. Then, using Taylor expansion from a point $\boldsymbol{x} \in \Gamma$ to a point $\tilde{\boldsymbol{x}} \in \tilde{\Gamma}$ one gets

$$g(\boldsymbol{x}) = u(\tilde{\boldsymbol{x}}) + \boldsymbol{\nabla} \boldsymbol{u}(\tilde{\boldsymbol{x}}) \cdot (\boldsymbol{x} - \tilde{\boldsymbol{x}}) + O(||\boldsymbol{x} - \tilde{\boldsymbol{x}}||^2), \tag{1.16}$$

which in the case of the development (1.16) characterizes a second order accuracy in the definition of the interface conditions on $\tilde{\Gamma}$. Using the characterization of map $\mathbf{M}$ defined by Equation (1.12) in Condition (1.16) it leads to

$$g(\boldsymbol{M}(\tilde{\boldsymbol{x}})) = u(\tilde{\boldsymbol{x}}) + \boldsymbol{\nabla} \boldsymbol{u}(\tilde{\boldsymbol{x}}) \cdot (\boldsymbol{M}(\tilde{\boldsymbol{x}}) - \tilde{\boldsymbol{x}}) + O(||\boldsymbol{M}(\tilde{\boldsymbol{x}}) - \tilde{\boldsymbol{x}}||^2) \quad .$$

By introducing the distance vector $\mathbf{d}$ defined by Equation (1.13) one now obtains

$$g(\boldsymbol{M}(\tilde{\boldsymbol{x}})) = u(\tilde{\boldsymbol{x}}) + \boldsymbol{\nabla} \boldsymbol{u}(\tilde{\boldsymbol{x}}) \cdot \boldsymbol{d}(\tilde{\boldsymbol{x}}) + O(||\boldsymbol{d}(\tilde{\boldsymbol{x}})||^2),$$

which with Equation (1.15) finally gives

$$\begin{aligned} g(\tilde{\boldsymbol{x}}) &= u(\tilde{\boldsymbol{x}}) + \boldsymbol{\nabla} \boldsymbol{u}(\tilde{\boldsymbol{x}}) \cdot \boldsymbol{d}(\tilde{\boldsymbol{x}}) + O(||\boldsymbol{d}(\tilde{\boldsymbol{x}})||^2) \\ u(\tilde{\boldsymbol{x}}) &\approx g(\tilde{\boldsymbol{x}}) - \boldsymbol{\nabla} \boldsymbol{u}(\tilde{\boldsymbol{x}}) \cdot \boldsymbol{d}(\tilde{\boldsymbol{x}}) \end{aligned} \tag{1.17}$$

The expression (1.17) is used to impose interface conditions in the context of an embedded resolution. In this context, the process of imposing boundary conditions has been introduced for Dirichlet conditions, but the same method is applicable to any type of conditions, a description for a Neumann condition is available in Appendix [A].

1.3 Weak imposition : The Nitsche's method

1.3.1 Principles

In this section, the Nitsche's method which allows to impose weakly boundary conditions for a system of PDEs is described. It consists of including the boundary conditions in the weak formulation of the system of PDEs rather than in the finite functional element space. It is useful to shift the value of boundary conditions of a domain to a surrogate boundary, see [2]. The Nitsche's imposition coupled with the **SBM** prevents problems such as "hanging nodes" which often appear during mesh adaptation [118], as mentioned in Section 1.2 for the cut-cell method.

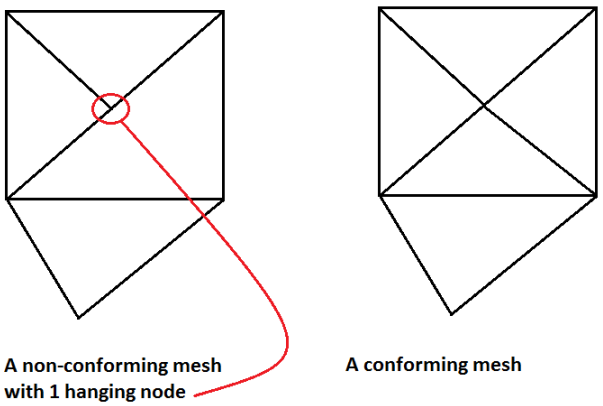


Figure 1.18: Illustration of an hanging node in 2D[13]

A hanging node is a node which is only connected to some of the other nodes of its direct

neighbours, see Figure 1.18, making the mesh unsuited for a conform simulation. It means that after the adaptation of the mesh to the new domain it is necessary to perform once again a new mesh adaption to cut the elements where hanging nodes have emerged. Thus, it is easily understandable that the task can be long and not straightforward.

The SBM is formulated on a Finite Element Method (FEM) and some boundary conditions can disappear during the process of definition of the weak problem. Some of the conditions can not be defined in the space of definition of the functions or imposed on the mesh through the scheme, after the different stages of the integration by parts. The SBM and the Nitsche's method are intrinsically linked together and the missing conditions are imposed with the Nitsche's method. In fact, the imposition of conditions can only be done if the boundary is a part of the mesh, which is not the case since no adaptation is performed on the mesh regarding the embedded boundary. To develop a robust method it is essential to simultaneously use the **SBM** with the **Nitsche's imposition**. But, one can note that the Nitsche's method can be used for a variety of problems and does not require the SBM.

The Nitsche's imposition is a method which consists of modifying the standard weak formulation by adding bilinear forms used to ensure the missing conditions. Some examples are available in the followings sections 1.3.3 and 1.3.4. Before looking at these examples, Section 1.3.2 introduces all the necessary notations and formalism to correctly establish any weak formulation presented in this work.

1.3.2 Formalism and 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. $D^{'}(\Omega)^d$) be the dual space associated. As usual, $L^2(\Omega)$ denotes 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 $(L^2(\Omega))^d$ denotes 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, the space $L^2(\Gamma)$ and $(L^2(\Gamma))^d$ are introduced for functions defined on Γ and are associated with the following inner-products

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

The classical Sobolev spaces $H^1(\Omega)$ and $H(\mathrm{div}, \Omega)$ are also introduced such that

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

associated to the norm

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

[13] https://www.researchgate.net/post/Hanging-nodes-in-finite-element

and

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

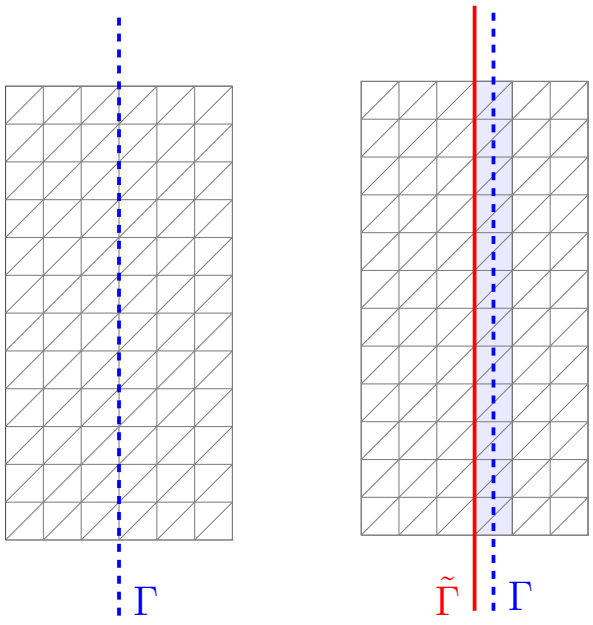
associated to the norm

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

Similar spaces are defined if one replaces Ω by Ω_+ or Ω_-, when Ω is composed of the two subdomains Ω_+ and Ω_- ($\Omega_+ \cup \Omega_- = \Omega$).

1.3.3 Dirichlet conditions

In this manuscript, a **conform** or **conformal case** refers to a simulation where the physical boundary matches the mesh discretization, and where the SBM is not required to impose boundary conditions (the distance d in Equation (1.13) being null). In Figure 1.19 one can see the differences between a conform resolution where the physical interface matches exactly the edges of the grid in Figure 1.19(a), while in Figure 1.19(b) the physical interface is intersected the elements of the background mesh and the surrogate interface is required, see Figure 1.19.



(a) Interface Γ conform to the mesh

(b) Interface Γ non conform to the mesh and its surrogate interface $\tilde{\Gamma}$

Figure 1.19: Comparison between conform resolution on Γ and an embedded resolution on $\tilde{\Gamma}$

1.3.3.a Conformal Resolution

To explain the Nitsche's imposition, the Poisson problem is chosen as example in its primal form with a Dirichlet boundary condition imposed on $\Gamma = \partial\Omega$. For the moment, the physical boundary is conform to the mesh discretization. All the different steps in the Nitsche's method will be explained in detail in this section and can be used as a reference on how to apply the technique to any type of problems.

The strong primal form of the Poisson problem with non-homogeneous Dirichlet boundary

conditions is characterized by

Find a solution $u \in V(\Omega) := \{u \in \mathbb{L}^2(\Omega), \nabla u \in (\mathbb{L}^2(\Omega))^d,\ u(x) = g(x) \text{ on } \Gamma\}$ such that

$$\begin{aligned} -\Delta u(\boldsymbol{x}) &= f(\boldsymbol{x}) \quad \text{on} \quad \Omega, \\ u(\boldsymbol{x}) &= g(\boldsymbol{x}) \quad \text{on} \quad \Gamma, \end{aligned} \tag{1.18}$$

where Ω is the domain of resolution, Γ the outer boundary i.e., $\Gamma = \partial\Omega$, and with f and g as given functions on Ω. An arbitrary initial condition is considered.

Let $w(\boldsymbol{x})$ be the test function associated to $u(\boldsymbol{x})$. Then, the conformal weak formulation for the Poisson problem can be written as

Step 1 : Weak Formulation on the physical boundary

Find $u \in V(\Omega)$ such that $\forall w \in V(\Omega)$ one has

$$\int_\Omega \boldsymbol{\nabla u}(\boldsymbol{x}) \cdot \boldsymbol{\nabla w}(\boldsymbol{x})\ dx - \int_\Gamma \boldsymbol{\nabla u}(\boldsymbol{x}) \cdot \boldsymbol{n}(\boldsymbol{x})\ w(\boldsymbol{x})\ d\gamma = \int_\Omega f(\boldsymbol{x}) w(\boldsymbol{x})\ dx.$$

The next step is to use the Nitsche's method to impose the condition $u(\boldsymbol{x}) = g(\boldsymbol{x})$ on Γ, which is cut into two stages. The first stage is called **symmetrization** and the second one is called **stabilization** [2]. The symmetrization gives the possibility to add the boundary condition in the formulation while the stabilization ensures the coercivity, which is essential for the uniqueness of the solution. It is a mandatory condition to satisfy the Lax-Milgram theorem which states the existence and the uniqueness of a solution to variational problems, see Theorem 5.

Step 2 : Symmetrization

Find $u \in V(\Omega)$ such that $\forall w \in V(\Omega)$ one has

$$\int_\Omega \boldsymbol{\nabla u} \cdot \boldsymbol{\nabla w}\ dx - \int_\Gamma \boldsymbol{\nabla u} \cdot \boldsymbol{n}\ w\ d\gamma - \int_\Gamma \boldsymbol{\nabla w} \cdot \boldsymbol{n}\ (u - g)\ d\gamma = \int_\Omega f w\ dx.$$

Step 3 : Stabilization

Find $u \in V(\Omega)$ such that $\forall w \in V(\Omega)$ one has

$$\int_\Omega \boldsymbol{\nabla u} \cdot \boldsymbol{\nabla w}\ dx - \int_\Gamma \boldsymbol{\nabla u}.\boldsymbol{n}\ w\ d\gamma - \int_\Gamma \boldsymbol{\nabla w}.\boldsymbol{n}\ (u-g)\ d\gamma + \int_\Gamma \frac{\alpha}{h} w(u-g) d\gamma = \int_\Omega f w\ dx \tag{1.19}$$

The coefficient $\frac{\alpha}{h}$ is considered as a penalty coefficient used to satisfy the coercivity condition. The parameter α has to be chosen following the problem, while h is the characteristic length of the considered mesh element. The parameter α can be determined by doing a dimensional analysis of the equations of the problem. Note that steps 2 and 3 preserve the consistency of the weak formulation.

1.3.3.b Embedded Resolution

In this section, the process of how to transform the weak formulation (1.19) in order to impose boundary conditions in the context of an embedded resolution is presented. A term to enforce a matching condition between tangential derivatives on $\tilde{\Gamma}$ (blue term in (1.20)) is also added to the weak formulation following reference [2]. The domain $\tilde{\Omega}$ will denote the surrogate domain where the definition of the surrogate boundary $\tilde{\Gamma} = \partial\tilde{\Omega}$ modifies the geometry of the considered domain.

Step 4 : Weak Formulation on the surrogate domain

Find $u \in V(\tilde{\Omega}) := \{u \in \mathbb{L}^2(\tilde{\Omega}), \nabla u \in (\mathbb{L}^2(\tilde{\Omega}))^d,\ u(x) = g(x) \text{ on } \tilde{\Gamma}\}$ such that $\forall w \in V(\tilde{\Omega})$ one has

$$\begin{aligned} &\int_{\tilde{\Omega}} \boldsymbol{\nabla u} \cdot \boldsymbol{\nabla w}\ dx - \int_{\tilde{\Gamma}} \boldsymbol{\nabla u} \cdot \tilde{\boldsymbol{n}}\ w\ d\gamma - \int_{\tilde{\Gamma}} \boldsymbol{\nabla w} \cdot \tilde{\boldsymbol{n}}\ \ (u + \boldsymbol{\nabla u} \cdot \boldsymbol{d} - g)\ d\gamma \\ &+ \int_{\tilde{\Gamma}} \frac{\alpha}{h}\ \ (w + \boldsymbol{\nabla w} \cdot \boldsymbol{d})\ \ (u + \boldsymbol{\nabla u} \cdot \boldsymbol{d} - g) d\gamma + \ \delta\ h \int_{\tilde{\Gamma}}\ w(u - g_\tau)\ d\gamma = \int_{\tilde{\Omega}} fw\ dx, \end{aligned} \tag{1.20}$$

where δ is a positive scalar, and $u = g_\tau$ is the value of u in the tangential direction τ on $\tilde{\Gamma}$. The blue term is often omitted if the geometry is not complex and is not considered in this work, see [2]. The green terms in (1.20) correspond to the Taylor expansion of the boundary conditions, where the development is also done on the test function to keep the property of symmetry in the weak formulation. After these different steps, the formulation is now taking place within the surrogate domain. Nevertheless, it is usually easier to work with a bilinear symmetric formulation. For that purpose, the term with the δ parameter is removed in (1.20), as it is considered as non essential for non complex geometries. A term in the variable w (the symmetric of $< \boldsymbol{\nabla w}.\tilde{\boldsymbol{n}}, \boldsymbol{\nabla u}.\boldsymbol{d} >_{\tilde{\Gamma}}$) which conserves the property of consistency is added in the formulation in order to bring symmetry in Formulation (1.20). Then, the embedded problem can be rewritten as

Find $u \in V(\tilde{\Omega})$ such that

$$a_u(u, w) = l_u(w) \quad \forall w \in V(\tilde{\Omega}), \tag{1.21}$$

where

$$\begin{aligned} a_u(u, w) = (\boldsymbol{\nabla u}, \boldsymbol{\nabla w})_{\tilde{\Omega}}\ - < w\ \ + \boldsymbol{\nabla w} \cdot \boldsymbol{d}\ , \boldsymbol{\nabla u} \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} - < \boldsymbol{\nabla w} \cdot \tilde{\boldsymbol{n}}, u + \boldsymbol{\nabla u} \cdot \boldsymbol{d} >_{\tilde{\Gamma}} + \\ < \boldsymbol{\nabla w} \cdot \boldsymbol{d}, \boldsymbol{\nabla u} \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} + \frac{\alpha}{h} < (w + \boldsymbol{\nabla w} \cdot \boldsymbol{d}), u + \boldsymbol{\nabla u} \cdot \boldsymbol{d} >_{\tilde{\Gamma}}, \end{aligned}$$

and

$$l_u(w) = (w, f)_{\tilde{\Omega}} - < \boldsymbol{\nabla w} \cdot \tilde{\boldsymbol{n}}, g >_{\tilde{\Gamma}} + \frac{\alpha}{h} < (w + \boldsymbol{\nabla w} \cdot \boldsymbol{d}), g >_{\tilde{\Gamma}}\ .$$

A symmetric formulation is almost obtained with formulation (1.21). To this end, a last transformation is done which consists of a decomposition for $\boldsymbol{\nabla u} \cdot \tilde{\boldsymbol{n}}$ and $\boldsymbol{\nabla u} \cdot \boldsymbol{d}$ introduced in (1.22) as follows

$$\begin{aligned} &a)\ \tilde{\boldsymbol{n}} = (\tilde{\boldsymbol{n}} \cdot \boldsymbol{n})\boldsymbol{n} + (\tilde{\boldsymbol{n}} \cdot \boldsymbol{\tau})\boldsymbol{\tau}, \\ &b)\ \boldsymbol{\nabla u} \cdot \tilde{\boldsymbol{n}} = ((\boldsymbol{\nabla u} \cdot \boldsymbol{n})\boldsymbol{n} + (\boldsymbol{\nabla u} \cdot \boldsymbol{\tau})\boldsymbol{\tau}) \cdot \tilde{\boldsymbol{n}}, \\ &c)\ \boldsymbol{\nabla u} \cdot \boldsymbol{d} = \boldsymbol{\nabla u} \cdot \boldsymbol{n}||\boldsymbol{d}||. \end{aligned} \tag{1.22}$$

Plugging transformation (1.22) into the term $< \boldsymbol{\nabla w} \cdot \boldsymbol{d}, \boldsymbol{\nabla u} \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}}$ in formulation (1.21), one obtains

$$\boldsymbol{\nabla u} \cdot \tilde{\boldsymbol{n}} = \frac{(\boldsymbol{n} \cdot \tilde{\boldsymbol{n}})}{||\boldsymbol{d}||} \boldsymbol{\nabla u} \cdot \boldsymbol{d} + (\boldsymbol{\nabla u} \cdot \boldsymbol{\tau})\boldsymbol{\tau} \cdot \tilde{\boldsymbol{n}}\ , \tag{1.23}$$

which brings the expected symmetry in the weak formulation (1.21) i.e., that $a_u(u, w) = a_u(w, u)$. Finally the Nitsche's method with Dirichlet conditions applied to Problem (1.18) and in the context of an embedded simulation can be summarized by

Find $u \in V(\tilde{\Omega})$ s.t.

$$a_u(u, w) = l_u(w) \quad \forall w \in V(\tilde{\Omega}),$$

with

$$a_u(u,w) = (\boldsymbol{\nabla u}, \boldsymbol{\nabla w})_{\tilde{\Omega}} - < w + \boldsymbol{\nabla w}\cdot\boldsymbol{d}, \boldsymbol{\nabla u}\cdot\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} - < \boldsymbol{\nabla w}\cdot\tilde{\boldsymbol{n}}, u + \boldsymbol{\nabla u}\cdot\boldsymbol{d} >_{\tilde{\Gamma}}$$
$$+ < \boldsymbol{\nabla w}\cdot\boldsymbol{d}, \frac{(\boldsymbol{n}\cdot\tilde{\boldsymbol{n}})}{||\boldsymbol{d}||}\boldsymbol{\nabla u}\cdot\boldsymbol{d} >_{\tilde{\Gamma}} + \frac{\alpha}{h} < (w + \boldsymbol{\nabla w}\cdot\boldsymbol{d}), u + \boldsymbol{\nabla u}\cdot\boldsymbol{d} >_{\tilde{\Gamma}},$$

and

$$l_u(w) = (w, f)_{\tilde{\Omega}} - < \boldsymbol{\nabla w}\cdot\tilde{\boldsymbol{n}}, g >_{\tilde{\Gamma}} + < \frac{\alpha}{h}(w + \boldsymbol{\nabla w}\cdot\boldsymbol{d}), g >_{\tilde{\Gamma}} - < \boldsymbol{\nabla w}\cdot\boldsymbol{d}, (\boldsymbol{\nabla g}\cdot\boldsymbol{\tau})\boldsymbol{\tau}\cdot\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} .$$

Remark 4. Working with a symmetric weak formulation is not required but ensures more stability and is in fact a stabilisation procedure by itself. In this case, the symmetric formulation is easier to implement. For moving fronts the tangential components have to be updated at each time step, see [80].

Remark 5. It is not always possible, nor is it easy to define a weak formulation that is completely symmetric, but it is always possible to make some of the terms in the weak formulation symmetric.

The Lax-Milgram theorem has briefly been introduced to justify in Step 3 the presence of the coefficient $\frac{\alpha}{h}$. Here, the general statement of this theorem is recalled and the definitions of the properties which need to be satisfied are expressed, see [119].

Theorem 2 (Bilinearity)**.** *A **bilinear form** on a space $\mathcal{V}$ is a function $b : \mathcal{V}\times\mathcal{V}\to\mathbb{R}$ which assigns a number to each pair of elements of $\mathcal{V}$ in such a way that b is linear in each variable.*

$$\begin{array}{lcll} b(v_1+v_2,w) & = & b(v_1,w)+b(v_2,w), & \forall \quad v_1,v_2,w\in\mathcal{V} \\ b(v,w_1+w_2) & = & b(v,w_1)+b(v,w_2), & \forall \quad v,w_1,w_2\in\mathcal{V} \\ b(av,w) & = & a\ b(v,w), & \forall \quad v,w\in\mathcal{V}, a\in\mathbb{R} \\ b(v,aw) & = & a\ b(v,w), & \forall \quad v,w\in\mathcal{V}, a\in\mathbb{R} \end{array}$$

Theorem 3 (Coercivity)**.** *Let $b(u,v)$ be a bilinear form of a Hilbert space $(\mathcal{V},||\cdot||_{\mathcal{V}})$, then the bilinear form $b(\cdot,\cdot)$ is said to be **coercive** if there exists $\gamma>0$ such that*

$$|b(u,u)| \geqslant \gamma||u||^2_{\mathcal{V}} \quad \forall u\in\mathcal{V}.$$

Theorem 4 (Continuity)**.** *Let $b(u,v)$ be a bilinear form of a Hilbert space $(\mathcal{V},||\cdot||_{\mathcal{V}})$, then the bilinear form $b(\cdot,\cdot)$ is said to be **continuous** if $\exists\ \alpha\geqslant 0$ such that $\forall u,v\in\mathcal{V}$ the form b satisfies*

$$|b(u,v)| \leqslant \alpha||u||_{\mathcal{V}}||v||_{\mathcal{V}}$$

Theorem 5 (Lax-Milgram)**.** *Let $b(u,v)$ be a **bilinear** form of a Hilbert space $(\mathcal{V},||\cdot||_{\mathcal{V}})$, and F be a **linear** form on $\mathcal{V}$. Consider the variational problem :*
Find $u\in\mathcal{V}$ such that

$$b(u,v) = F(v),\ \forall v\in\mathcal{V}\ .$$

*If b is a **continuous** and **coercive bilinear** form of $(\mathcal{V},||\cdot||_{\mathcal{V}})$, and F is a **continuous linear** form of $\mathcal{V}$, then there exists a unique solution $u\in\mathcal{V}$ to the linear variational problem with*

$$||u||_{\mathcal{V}} \leqslant \frac{1}{\gamma}||F||_{\mathcal{V}'}\ .$$

where γ is the coefficient satisfying the coercivity argument.

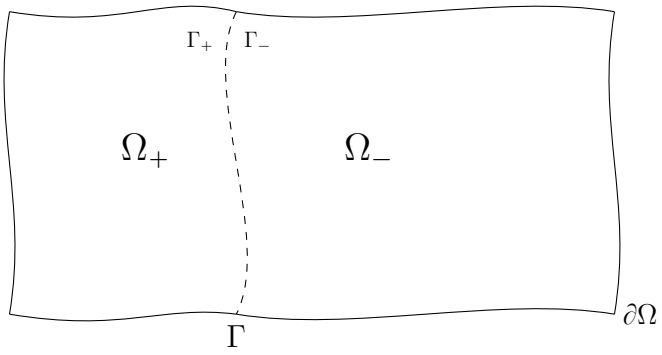


Figure 1.20: Visualization of a domain Ω for the definition of the interface conditions

1.3.4 Jump conditions and surrogate interface

For convenience, the jump operator $[\![\cdot]\!]_\Gamma$ and average operator $\{\cdot\}_\Gamma$ are now introduced. For any scalar f or vector quantity $\boldsymbol{g}$ on the interface Γ one has

$$\begin{aligned} [\![f]\!]_\Gamma &= f^+ - f^-, \qquad [\![\boldsymbol{g}]\!]_\Gamma = \boldsymbol{g}^+ - \boldsymbol{g}^-, \\ \{f\}_\Gamma &= \frac{1}{2}(f^+ + f^-), \quad \{\boldsymbol{g}\}_\Gamma = \frac{1}{2}(\boldsymbol{g}^+ + \boldsymbol{g}^-). \end{aligned} \tag{1.24}$$

where the values of f^+ (respectively g^+) and f^- (respectively g^-) are defined by (1.25), see Figure 1.20.

$$\begin{aligned} f^+ &= \lim_{x\in\Omega^+\to x\in\Gamma} f(t,\boldsymbol{x}), \\ f^- &= \lim_{x\in\Omega^-\to x\in\Gamma} f(t,\boldsymbol{x}). \end{aligned} \tag{1.25}$$

Moreover, for any functions f and h one has

$$[\![fh]\!]_\Gamma = [\![f]\!]_\Gamma \{h\}_\Gamma + \{f\}_\Gamma [\![h]\!]_\Gamma. \tag{1.26}$$

In this section, the mixed form of the Poisson equation is considered, where the flux is also defined as an unknown. In this example the surrogate boundary is a surrogate interface **embedded inside the domain** Ω. Let Ω be a domain composed of two subdomains Ω_+ and Ω_-, which are separated by an interface denoted $\Gamma = \bar{\Omega}_+ \cap \bar{\Omega}_-$, and with an outside boundary denoted $\partial\Omega$, see Figure 1.20. Then, the mixed form of the Poisson problem with jump conditions states as below

$$\begin{aligned} &a.1.27)\ \nabla\cdot\boldsymbol{\beta}(\boldsymbol{x}) &&= f(\boldsymbol{x}) && \text{on } \Omega, \\ &b.1.27)\ \boldsymbol{\beta}(\boldsymbol{x}) &&= -k\boldsymbol{\nabla} u(\boldsymbol{x}) && \text{on } \Omega, \\ &c.1.27)\ u(\boldsymbol{x}) &&= u_D && \text{on } \partial\Omega, \\ &d.1.27)\ [\![u(\boldsymbol{x})]\!]_\Gamma &&= J_1 && \text{on } \Gamma, \\ &e.1.27)\ [\![\boldsymbol{\beta}(\boldsymbol{x})]\!]_\Gamma\cdot n &&= -J_2 && \text{on } \Gamma. \end{aligned} \tag{1.27}$$

which is equivalent to its primal form

$$\begin{aligned} \nabla.(-k\boldsymbol{\nabla} u(\boldsymbol{x})) &= f && \text{on } \Omega, \\ u(\boldsymbol{x}) &= u_D && \text{on } \partial\Omega, \\ [\![u(\boldsymbol{x})]\!]_\Gamma &= J_1 && \text{on } \Gamma, \\ [\![k\boldsymbol{\nabla} u(\boldsymbol{x})]\!]_\Gamma\cdot n &= -J_2 && \text{on } \Gamma. \end{aligned} \tag{1.28}$$

The normal $\boldsymbol{n}$ corresponds to the normal at the interface Γ, oriented from Ω_+ towards Ω_-, see Figure 1.20. The mention of Γ in the jump operator specifies which interface the operator refers to, and simplifies the differentiation between the jump on the physical interface Γ and its surrogate $\tilde{\Gamma}$.
For this example, the subject of interest is the treatment made on the jump conditions. To achieve this, a strong enforcement of the Dirichlet conditions defined on $\partial\Omega$, the external boundary is considered. Let $(\phi, \boldsymbol{\psi})$ be the test functions associated to the unknowns $(u, \boldsymbol{\beta})$. Then the problem (1.27) can be written into its weak formulation as

Find $(u, \boldsymbol{\beta}) \in \mathcal{S}_u(\Omega) \times \mathcal{S}_{\boldsymbol{\beta}}(\Omega)$ such that $\forall\ (\phi, \boldsymbol{\psi}) \in \mathcal{S}_\phi(\Omega) \times \mathcal{S}_{\boldsymbol{\psi}}(\Omega)$ one has

$$\begin{array}{lll} 1.29.a) & (\phi, \nabla.\boldsymbol{\beta})_\Omega & = (f, \phi)_\Omega \\ 1.29.b) & (\boldsymbol{\psi}, k^{-1}\boldsymbol{\beta} + \boldsymbol{\nabla u})_\Omega & = 0 \end{array} \tag{1.29}$$

where

$$\begin{aligned} \mathcal{S}_u(\Omega) &= \{u \in H^2(\Omega_+) \cup H^2(\Omega_-) |\ u = u_D \ \text{ on } \ \partial\Omega\}, \\ \mathcal{S}_{\boldsymbol{\psi}}(\Omega) &= \mathcal{S}_{\boldsymbol{\beta}}(\Omega) = \{\boldsymbol{\psi} \in (H^2(\Omega_+))^d \cup (H^2(\Omega_-))^d\}, \\ \mathcal{S}_\phi(\Omega) &= \{\phi \in H^2(\Omega_+) \cup H^2(\Omega_-) \mid \phi = 0 \ \text{ on } \partial\Omega\}. \end{aligned}$$

The next step consists of incorporating the jump conditions into the weak formulation. The domain Ω is decomposed into its two subdomains, and an integration by parts is performed on Equation (1.29.a)
$\forall \phi \in \mathcal{S}_\phi(\Omega)$,

$$\begin{aligned} (\phi, \nabla \cdot \boldsymbol{\beta})_\Omega = \quad & (\phi, \nabla \cdot \boldsymbol{\beta})_{\Omega_-} + (\phi, \nabla \cdot \boldsymbol{\beta})_{\Omega_+}, \\ = & -(\boldsymbol{\nabla\phi}, \boldsymbol{\beta})_{\Omega_+} + < \boldsymbol{\beta}^+\phi^+, \boldsymbol{n}_+ >_{\Gamma^+} + < \boldsymbol{\beta}\phi, \boldsymbol{n} >_{\partial\Omega\cap\Omega_+} -(\boldsymbol{\nabla\phi}, \boldsymbol{\beta})_{\Omega_-} \\ & + < \boldsymbol{\beta}^-\phi^-, \boldsymbol{n}_- >_{\Gamma^-} + < \boldsymbol{\beta}\phi, \boldsymbol{n} >_{\partial\Omega\cap\Omega_-}, \\ = & -(\boldsymbol{\nabla\phi}, \boldsymbol{\beta})_\Omega + < \boldsymbol{\beta}^+\phi^+ - \boldsymbol{\beta}^-\phi^-, \boldsymbol{n} >_\Gamma, \\ = & -(\boldsymbol{\nabla\phi}, \boldsymbol{\beta})_\Omega + < [\![\boldsymbol{\beta}\phi]\!], \boldsymbol{n} >_\Gamma = (\phi, f)_\Omega, \end{aligned} \tag{1.30}$$

since ϕ is equal to 0 on $\partial\Omega$.
Equation (1.30) needs to be modified in order to take into account the Condition (1.27.e) and impose the condition in the definition of the finite element scheme. Equation (1.30) can be rewritten using the decomposition (1.26) which leads to
$\forall \phi \in \mathcal{S}_\phi(\Omega)$,

$$-(\boldsymbol{\nabla\phi}, \boldsymbol{\beta})_\Omega + < [\![\phi]\!], \{\boldsymbol{\beta}\}.\boldsymbol{n} >_\Gamma + < \{\phi\}, [\![\boldsymbol{\beta}]\!] \cdot \boldsymbol{n} >_\Gamma = (\phi, f)_\Omega\ . \tag{1.31}$$

Then the following problem is obtained

Find $(u, \boldsymbol{\beta}) \in \mathcal{S}_u(\Omega) \times \mathcal{S}_{\boldsymbol{\beta}}(\Omega)$ such that $\forall\ (\phi, \boldsymbol{\psi}) \in \mathcal{S}_\phi(\Omega) \times \mathcal{S}_{\boldsymbol{\psi}}(\Omega)$ one has

$$\begin{aligned} 1.32.a) & -(\boldsymbol{\nabla\phi}, \boldsymbol{\beta})_\Omega + < [\![\phi]\!], \{\boldsymbol{\beta}\}.\boldsymbol{n} >_\Gamma + \alpha\ < [\![\phi]\!], [\![u]\!] >_\Gamma = (\phi, f)_\Omega \\ & \qquad\qquad + < \{\phi\}, J_2 >_\Gamma + \alpha\ < [\![\phi]\!], J_1 >_\Gamma, \\ 1.32.b) & \ (\boldsymbol{\psi}, k^{-1}\boldsymbol{\beta} + \boldsymbol{\nabla u})_\Omega - < \{\boldsymbol{\psi}\}, [\![u]\!].\boldsymbol{n} >_\Gamma = - < \{\boldsymbol{\psi}\}, J_1 \cdot \boldsymbol{n} >_\Gamma, \end{aligned} \tag{1.32}$$

where α is a coefficient to be determined and which depends on the characteristic length of the elements of the mesh.

The weak imposition is done following the same steps that described in Section 1.3.3, the only difference is the use of the jump operator, which is a linear operator and does not bring any complexity. As with the Dirichlet condition in Section 1.3.3, the formulation (1.32) is

not symmetric, but it is possible to transform it into one for more stability. Moreover, the formulation has been rewritten for a conformal resolution as a preliminary step. In order to use the **SBM** for a non conformal interface the formulation needs to be adapted. First, the interface conditions are introduced on the surrogate boundary $\tilde{\Gamma}$. For the flux jump 1.27.e) a Taylor expansion is performed

$$\begin{aligned} & [\![\boldsymbol{\beta} + \nabla \cdot \boldsymbol{\beta} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}}.\tilde{\boldsymbol{n}} && = [\![\boldsymbol{\beta}]\!]_{\Gamma} \cdot \boldsymbol{n} \\ \Longrightarrow\ & [\![\boldsymbol{\beta}]\!]_{\tilde{\Gamma}} \cdot \tilde{\boldsymbol{n}} && = [\![\boldsymbol{\beta}]\!]_{\Gamma} \cdot \boldsymbol{n} - [\![\nabla \cdot \boldsymbol{\beta} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}} \cdot \tilde{\boldsymbol{n}} \\ \Longrightarrow\ & [\![\boldsymbol{\beta}]\!]_{\tilde{\Gamma}} \cdot \tilde{\boldsymbol{n}} && = -J_2 - [\![\nabla \cdot \boldsymbol{\beta} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}} \cdot \tilde{\boldsymbol{n}} \end{aligned} \tag{1.33}$$

The same work is done for the primal variable u

$$\begin{aligned} & [\![u + \boldsymbol{\nabla u} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}} && = [\![u]\!]_{\Gamma} \\ \Longrightarrow\ & [\![u]\!]_{\tilde{\Gamma}} && = [\![u]\!]_{\Gamma} - [\![\boldsymbol{\nabla u} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}} \\ \Longrightarrow\ & [\![u]\!]_{\tilde{\Gamma}} && = J_1 - [\![\boldsymbol{\nabla u} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}} \ . \end{aligned} \tag{1.34}$$

Conditions (1.33) and (1.34) are then plugged into Formulation (1.32). To obtain the desired formulation the operators and the normal vector are considered on $\tilde{\Gamma}$ instead of Γ, where $\Omega = \tilde{\Omega}_+ \cup \tilde{\Omega}_-$.

Find $(u, \boldsymbol{\beta}) \in \mathcal{S}_u(\Omega) \times \mathcal{S}_{\boldsymbol{\beta}}(\Omega)$ such that $\forall\ (\phi, \boldsymbol{\psi}) \in \mathcal{S}_\phi(\Omega) \times \mathcal{S}_{\boldsymbol{\psi}}(\Omega)$ one has

$$\begin{aligned} &1.35.a) - (\boldsymbol{\nabla \phi}, \boldsymbol{\beta})_\Omega + < [\![\phi]\!]_{\tilde{\Gamma}}, \{\boldsymbol{\beta}\}_{\tilde{\Gamma}}.\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} + \alpha\ < [\![\phi]\!]_{\tilde{\Gamma}}, [\![u]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}} = (\phi, f)_\Omega \\ &\qquad + < \{\phi\}_{\tilde{\Gamma}}, J_2 - [\![\nabla.\boldsymbol{\beta} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}} \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} + \alpha\ < [\![\phi]\!]_{\tilde{\Gamma}}, J_1 - [\![\boldsymbol{\nabla u} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}}, \\ &1.35.b)\ (\boldsymbol{\psi}, k^{-1}\boldsymbol{\beta} + \boldsymbol{\nabla u})_\Omega - < \{\boldsymbol{\psi}\}_{\tilde{\Gamma}}, [\![u]\!]_{\tilde{\Gamma}}.\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} = - < \{\boldsymbol{\psi}\}_{\tilde{\Gamma}}, (J_1 - [\![\boldsymbol{\nabla u} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}}) \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} \quad . \end{aligned} \tag{1.35}$$

The next step consists of bringing more symmetry in Formulation (1.35) by developing the test functions to the same order as the Taylor expansions of the interface conditions, see Conditions (1.33) and (1.34).

Find $(u, \boldsymbol{\beta}) \in \mathcal{S}_u(\Omega) \times \mathcal{S}_{\boldsymbol{\beta}}(\Omega)$ such that $\forall\ (\phi, \boldsymbol{\psi}) \in \mathcal{S}_\phi(\Omega) \times \mathcal{S}_{\boldsymbol{\psi}}(\Omega)$ one has

$$\begin{aligned} &1.36.a) - (\boldsymbol{\nabla \phi}, \boldsymbol{\beta})_\Omega + < [\![\phi + \boldsymbol{\nabla \phi} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}}, \{\boldsymbol{\beta}\}_{\tilde{\Gamma}}.\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} + \alpha\ < [\![\phi + \boldsymbol{\nabla \phi} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}}, [\![u]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}} = (\phi, f)_\Omega \\ &+ < \{\phi + \boldsymbol{\nabla \phi} \cdot \boldsymbol{d}\}_{\tilde{\Gamma}}, J_2 - [\![\nabla.\boldsymbol{\beta} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}} \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} + \alpha\ < [\![\phi + \boldsymbol{\nabla \phi} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}}, J_1 - [\![\boldsymbol{\nabla u} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}}, \\ &1.36.b)\ (\boldsymbol{\psi}, \frac{\boldsymbol{\beta}}{k} + \boldsymbol{\nabla u})_\Omega - < \{\boldsymbol{\psi} + \nabla.\boldsymbol{\psi} \cdot \boldsymbol{d}\}_{\tilde{\Gamma}}, [\![u]\!]_{\tilde{\Gamma}}.\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} = - < \{\boldsymbol{\psi} + \nabla.\boldsymbol{\psi} \cdot \boldsymbol{d}\}_{\tilde{\Gamma}}, (J_1 - [\![\boldsymbol{\nabla u} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}}) \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} . \end{aligned} \tag{1.36}$$

The last transformation is to rearrange the terms concerning the bilinear part to the left hand side. This step is not necessary but brings more clarity in the definition of the weak formulation and will be a convention in the rest of the following Chapters.

Find $(u, \boldsymbol{\beta}) \in \mathcal{S}_u(\Omega) \times \mathcal{S}_{\boldsymbol{\beta}}(\Omega)$ such as $\forall\ (\phi, \boldsymbol{\psi}) \in \mathcal{S}_\phi(\Omega) \times \mathcal{S}_{\boldsymbol{\psi}}(\Omega)$ one that

$$\begin{aligned} &1.37.a) - (\boldsymbol{\nabla \phi}, \boldsymbol{\beta})_\Omega + < [\![\phi + \boldsymbol{\nabla \phi} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}}, \{\boldsymbol{\beta}\}_{\tilde{\Gamma}}.\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} + \alpha\ < [\![\phi + \boldsymbol{\nabla \phi} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}}, [\![u + \boldsymbol{\nabla u} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}} \\ &- < \{\phi + \boldsymbol{\nabla \phi} \cdot \boldsymbol{d}\}_{\tilde{\Gamma}}, [\![\nabla.\boldsymbol{\beta} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}} \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} = (\phi, f)_\Omega + < \{\phi + \boldsymbol{\nabla \phi} \cdot \boldsymbol{d}\}_{\tilde{\Gamma}}, J_2 >_{\tilde{\Gamma}} \\ &+ \alpha\ < [\![\phi + \boldsymbol{\nabla \phi} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}}, J_1 >_{\tilde{\Gamma}}, \\ &1.37.b)\ (\boldsymbol{\psi}, k^{-1}\boldsymbol{\beta} + \boldsymbol{\nabla u})_\Omega - < \{\boldsymbol{\psi} + \nabla \cdot \boldsymbol{\psi} \cdot \boldsymbol{d}\}_{\tilde{\Gamma}}, [\![u + \boldsymbol{\nabla u} \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}}.\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} = - < \{\boldsymbol{\psi} + \nabla.\boldsymbol{\psi} \cdot \boldsymbol{d}\}_{\tilde{\Gamma}}, J_1 \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} . \end{aligned} \tag{1.37}$$

1.4 Conclusion on Chapter 1

In this first Chapter, the necessary elements to understand the principle of the **Finite Element Method** (FEM) for the resolution of a system of equations, referred to as PDEs has been introduced. The focus was placed on the **Galerkin approach**, and the weak imposition of missing boundary conditions through the **Nitsche's method** was discussed in detailed. This chapter has also presented the choice made to handle moving boundaries without a need for mesh adaptation with the use of the **Shifted Boundary Method** (SBM), an embedded method for non conforming mesh boundaries. In the next chapter, the application of the SBM to the **Stefan model** will be discussed and adapted to handle moving interfaces.

Chapter 2

The Stefan Model

2.1 Description

The Stefan model is a category of problems which regroups free and moving boundary problems. They are named after Joseph Stefan (1835-1893) and his work on the formation of ice in the polar seas. The physicist's most important work concerns his discovery on heat radiation leading to the well-known Stefan-Boltzmann law of radiation, stating that heat radiation (energy travelling in all direction at the speed of light from its point of absorption) is proportional to the fourth power of the absolute temperature (measured in Kelvin where 0 is the absolute zero, i.e. with no motion of particles and no heat) [12].

In this Chapter, mathematical models describing phase change transitions are studied. A phase is a region in which the solution is continuous and differentiable up to the order of the PDE [4]. The considered model is composed of two sub-domains ($\Omega = \Omega_- \cup \Omega_+$), one filled with ice and the other one filled with water. The zone filled with water is called the liquid phase while the zone filled with ice is referred to as the solid phase. The system is a solid to liquid phase-change transition problem where the boundary moves from the liquid phase to the solid phase. At the interface the material coexists in both state; liquid and solid. Its thickness is usually between a few Angstroms (10^{-10}m) and a few centimeters (10^{-2}m), but the assumptions is made that the interface has zero thickness [4]. The initial position of the interface is part of the initial conditions of the considered problem. On a mathematical perspective the formulation relates the relationships between the variables of the melting process, defined in their primal form by the temperature field, the interface position, the thermo-physical properties of the material (heat conductivity, specific heat, density) and the interface conditions. Those relationships can be described independently on each of the phase composing the model.

Since phase transition models are free boundary problems, where the boundary position is also unknown, a closure condition is required to obtain a valid model. This condition is called the Stefan condition [4]. In heat transfer problems with phase changes, the physical constraint and closure condition states that the conservation of energy and the local velocity at the interface depend on the heat flux discontinuity at the interface. At the phase change area latent heat is released or absorbed, with latent heat characterizing the release or absorption of energy by the thermodynamic system.

In summary, for a classical Stefan formulation there are two unknowns

1. the temperature;
2. the position of the interface $\Gamma(t)$.

The temperature diffuses inside the domain, while the flux at the interface moves the boundary. The position of the moving boundary is a function of time and space and needs to be determined as a part of the solution. Then, resolving the partial differential equation (PDE) necessitates solving the heat equation in all of the phases, with the moving boundary determined by the Stefan condition.

Analytically, it is difficult to find a solution to a Stefan problem. Cases where it is possible are characterized a high degree of symmetry in the geometry of the domain and by a judicious choice of boundary conditions, see [4].

2.1.1 Mathematical Formulation

Consider a domain Ω, a **bounded** and **connected open set** in $\mathbb{R}^d$ ($d \geqslant 2$), 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 $\Gamma_D \cap \Gamma_N = \varnothing$. 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 2.1). Let $\Gamma_+(t)$ be the designation of the interface from the liquid phase $\Omega_+(t)$, defined as $\overline{\Omega_+}(t) \cap \Gamma(t)$. Similarly, let $\Gamma_-(t)$ be the designation of the interface from the solid phase of the problem $\Omega_-(t)$, defined as $\overline{\Omega_-}(t) \cap \Gamma(t)$. Using the fact that at the interface Γ the outward normal vectors $\boldsymbol{n}_+$ and $\boldsymbol{n}_-$ are equals up to a sign the following convention is introduced

$$\boldsymbol{n} = \boldsymbol{n}_+ = -\boldsymbol{n}_- \tag{2.1}$$

where n_+ is the normal vector oriented from Ω_+ to Ω_-.

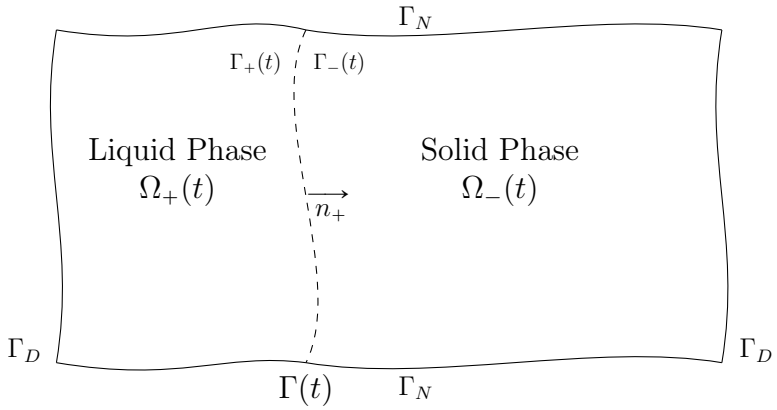


Figure 2.1: Configuration of the domain Ω and its different phases

The classical Stefan problem for a two-phase melting process is considered with one phase-change front, where the domain Ω_+ represents a liquid phase while the domain Ω_- is the solid phase, see Figure 2.1. The distribution of the temperature $T(\boldsymbol{x}, t)$ in Ω is studied, corresponding to a material of **density** ρ, taken constant in each of the domains. The variable λ denotes its **thermal conductivity**, *i.e.*, the ability of the material to conduct heat. In general, the thermal conductivity is a tensor, but here the isotropic case is being considered where the conduction is independent of the direction. In a multi-phase situation, the conductivity can be

discontinuous: λ_- represents the value inside the solid phase Ω_-, which is usually higher than λ_+, the value in the liquid phase Ω_+. The **specific heat** of the material is denoted by c, a phase dependent coefficient (c_+ and c_-) which characterizes the necessary amount of heat to increase one unit of mass of the material by one unit of temperature.
Those discontinuities in conductivity and specific heat, located at the interface Γ, can lead to discontinuities in the solution. To maintain mathematical generality, at this stage, discontinuities in T and its gradients are considered. Later on, for physical tests, the temperature will be taken continuous and equal to the melting temperature denoted T_m at the moving interface. The variables T^+ and T^- denote the values of T respectively on Γ^+ and Γ^-:

$$T^+ = \lim_{x\in\Omega^+\to x\in\Gamma} T(\boldsymbol{x},t) \quad ; \quad T^- = \lim_{x\in\Omega^-\to x\in\Gamma} T(\boldsymbol{x},t). \tag{2.2}$$

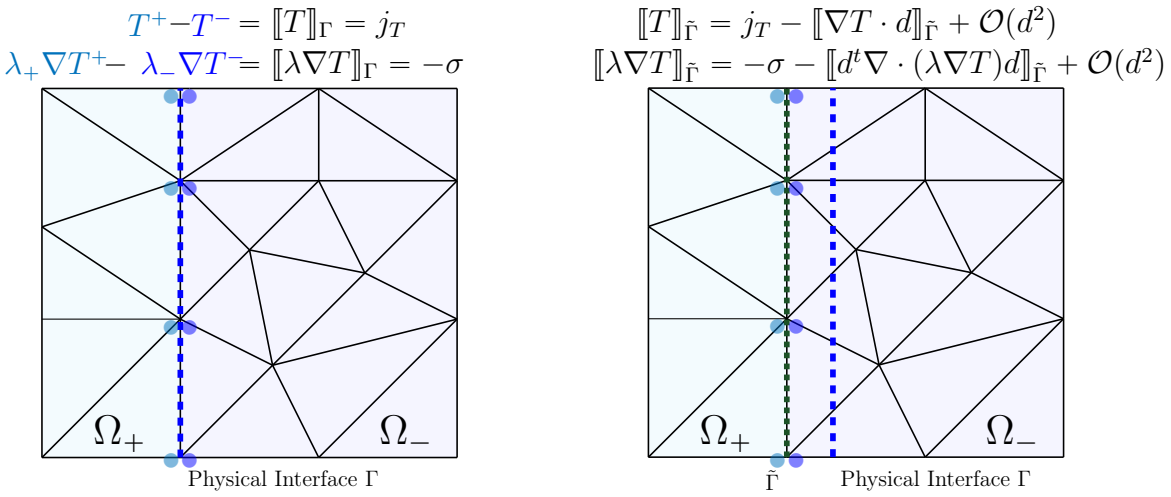
In the next section, the system of equations characterizing the Stefan model is presented. It will present the primal form of the model, where the temperature field T and the interface position Γ are the parameters to be determined.

2.1.2 Primal form

In this section, the classical formulation of the Stefan model referred to as the **primal form** is introduced. Given an initial temperature T_0, an initial interface position $\boldsymbol{\Gamma}(0)$ and a final time t_f, the Stefan problem consists of solving the following set of PDEs

$$\begin{aligned}
&2.3.a) \quad \rho c \partial_t T(t,\boldsymbol{x}) - \nabla\cdot(\lambda\nabla T(t,\boldsymbol{x})) = f(t,\boldsymbol{x}) && \text{in }]0,t_f[\times\Omega,\\
&2.3.b) \quad T(t,\boldsymbol{x}) = T_D && \text{on }]0,t_f[\times\Gamma_D,\\
&2.3.c) \quad \lambda\nabla T(t,\boldsymbol{x})\cdot\boldsymbol{n} = -h_N && \text{on }]0,t_f[\times\Gamma_N,\\
&2.3.d) \quad T(0,\boldsymbol{x}) = T_0 && \text{in } \Omega,
\end{aligned} \tag{2.3}$$

where T_D and h_N are given data. In order to close the System (2.3), a boundary conditions on Γ is required because Γ is a moving interface. **Two sets of boundary conditions** are proposed.



(a) Conform Resolution

(b) Embedded Resolution

Figure 2.2: Set 1 : Visualization of the imposition of the boundary conditions

The first set of interface conditions consists of imposing the **jump of temperature** and the

jump of normal heat flux across the interface:

$$\begin{aligned} [\![T(t,\boldsymbol{x})]\!]_\Gamma &= j_T, \\ [\![\lambda \nabla T(t,\boldsymbol{x})]\!]_\Gamma \cdot \boldsymbol{n} &= -\sigma, \end{aligned} \tag{2.4}$$

where j_T and σ are given data.

In Figure (2.2.a) the imposition of the conditions (2.4) for a conform resolution is presented. The conditions are imposed on the physical interface Γ. For an embedded resolution in Figure (2.2.b) the conditions (2.4) are extrapolated with Taylor expansions to be imposed on $\tilde{\Gamma}$.

The second set of interface conditions prescribes the value of the temperature at the interface, as a Dirichlet conditions:

$$\begin{cases} T^+ = T_L, \\ T^- = T_R, \end{cases} \tag{2.5}$$

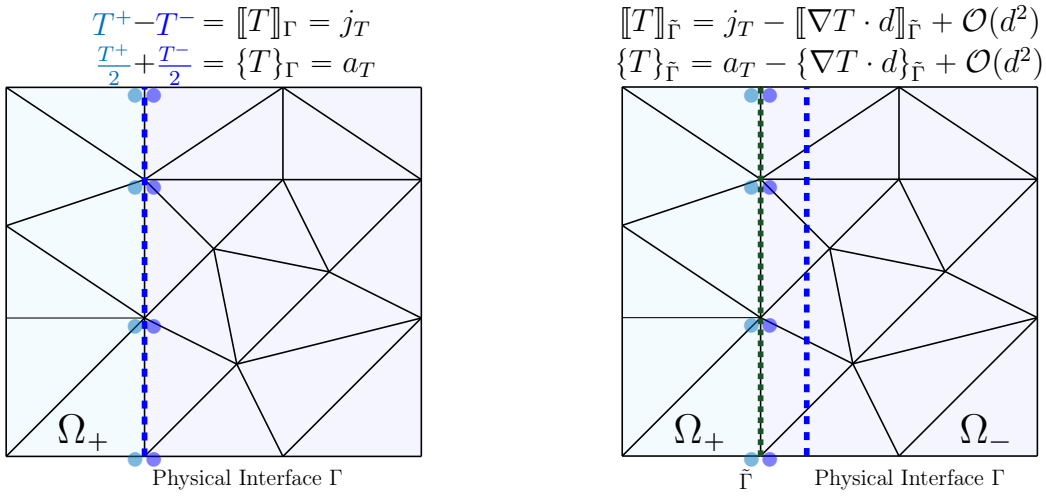
where T_L and T_R are given data. In the latter case, the normal interface velocity can be derived from the recovered flux jump at the interface, through the Stefan condition:

$$\rho L_m \boldsymbol{\Gamma}'(t) \cdot \boldsymbol{n} = -[\![\lambda \nabla T]\!]_\Gamma \cdot \boldsymbol{n}, \tag{2.6}$$

with $\Gamma'(t)$ the velocity associated to the interface position Γ and L_m the **latent heat**.
The interface condition (2.5) can also be expressed as a condition on the **temperature jump** and **temperature average**. Indeed, Conditions (2.5) are applied on Γ from both subdomains of Ω ($\Omega = \Omega_+ \cup \Omega_-$). Another option is to combine them and apply directly the conditions on the interface Γ without considering the subdomains. Moreover, Definition (2.7) allows for simulations satisfying the physics of the Stefan model, to impose the continuity of the temperature field T ($[\![T]\!] = 0$) and the melting temperature T_m ($\{T\} = T_m$) on Γ. Keeping once again all generality the Conditions (2.5) can be rewritten as follow

$$\boxed{\begin{aligned} [\![T]\!]_\Gamma &= T_L - T_R = j_T, \\ \{T\}_\Gamma &= \frac{T_L + T_R}{2} = a_T, \end{aligned}} \tag{2.7}$$

where j_T and a_T are given data.



(a) Conform Resolution (b) Embedded Resolution

Figure 2.3: Set 2 : Visualization of the imposition of the boundary conditions

In Figure (2.3.a) the imposition of the conditions (2.7) for a conform resolution is presented. They are imposed on the physical interface Γ. For an embedded resolution in Figure (2.3.b) the conditions (2.7) are extrapolated with Taylor expansions to be imposed on $\tilde{\Gamma}$.

The primal form can bring advantages depending on the context of application. Nevertheless, in this work, considering the Stefan condition (2.6) where the front motion is driven by the heat flux, it is necessary to accurately describe the gradient of the temperature to update the interface location. This can be achieved by using the mixed formulation of the problem, which, when combined with a temperature enrichment (see Section 2.4), can provide a better accuracy of the front velocity and then its position. Furthermore, in [5] the disadvantages of the primal formulation are mentioned where additional references are available. The primal form can expose to a loss of accuracy, and mass conservation is not guaranteed even when the flux is reconstructed afterwards.

2.1.3 Mixed form

The mixed formulation of the Stefan problem is obtained by decoupling the original System (2.3). The local heat flux $\boldsymbol{\beta}$ is introduced, corresponding to the amount of energy that flows through one unit of area per time unit. The expression of $\boldsymbol{\beta}$ can be interpreted as the Fourier's law, which is a classical relation in the theory of heat conduction [120].

Theorem 6 (Fourier's Law)**.** *The Fourier's law of thermal conductivity expresses that the rate of heat transfer through a medium is proportional to the negative gradient of temperature and the surface area through which the heat flows, see Figure 2.4. Then, the flux is linked to the gradient of temperature and the thermal conductivity of the surface as follows*

$$\beta := -\lambda \nabla T,$$

where

- β *is the vector of the local heat flux* $[W.m^{-2}]$
- λ *is the thermal conductivity of the surface* $[W.m^{-1}.K^{-1}]$
- ∇T *is the gradient of temperature* $[K.m^{-1}]$

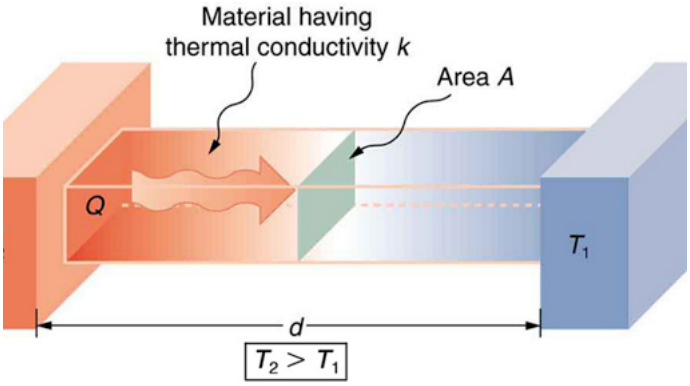


Figure 2.4: Transfers from hot to cold surface by thermal conduction[14]

[14]https://rogerscorp.com/blog/2021/thermal-management-for-laminated-busbars

The mixed formulation of problem (2.3) can be written into the form

$$
\begin{aligned}
&2.8.a) \quad \rho c \partial_t T(t,\boldsymbol{x}) + \nabla \cdot \boldsymbol{\beta}(t,\boldsymbol{x}) = f(t,\boldsymbol{x}) && \text{in }]0,t_f[\times\Omega,\\
&2.8.b) \quad \boldsymbol{\beta}(t,\boldsymbol{x}) = -\lambda \nabla T(t,\boldsymbol{x}) && \text{in }]0,t_f[\times\Omega,\\
&2.8.c) \quad T(t,\boldsymbol{x}) = T_D && \text{on }]0,t_f[\times\Gamma_D,\\
&2.8.d) \quad \boldsymbol{\beta}(t,\boldsymbol{x}) \cdot \boldsymbol{n} = h_N && \text{on }]0,t_f[\times\Gamma_N,\\
&2.8.e) \quad T(0,\boldsymbol{x}) = T_0 && \text{in } \Omega,
\end{aligned}
\tag{2.8}
$$

where the unknowns are the temperature field $T(\boldsymbol{x},t)$, the heat flux $\boldsymbol{\beta}(\boldsymbol{x},t)$ and the interface position $\Gamma(t)$.

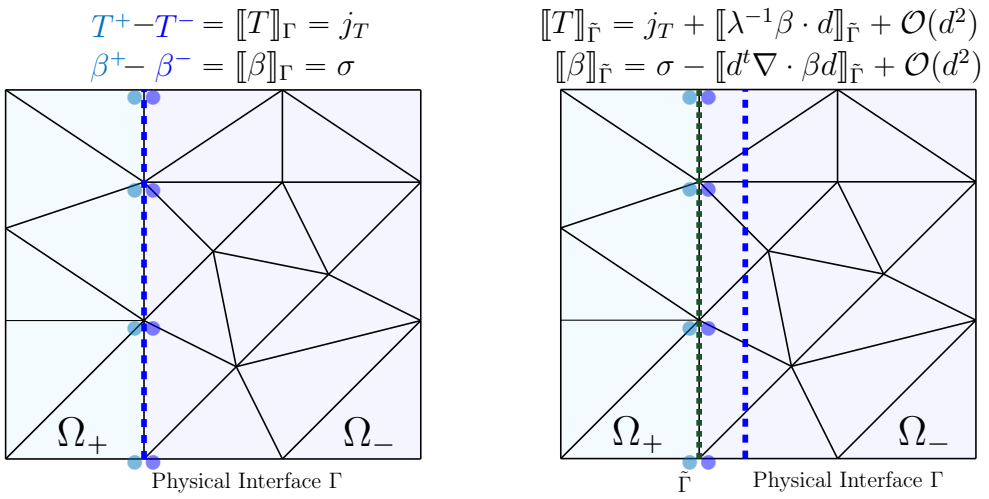
For the first set of interface conditions (2.4) the following conditions are imposed

$$
\begin{cases}
[\![T(t,\boldsymbol{x})]\!]_\Gamma &= j_T(t,\boldsymbol{x}),\\
[\![\boldsymbol{\beta}\cdot\boldsymbol{n}]\!]_\Gamma &= \sigma(t,\boldsymbol{x}),
\end{cases}
\tag{2.9}
$$

with the Stefan condition defined by

$$
\rho L_m \boldsymbol{\Gamma}'(t)\cdot\boldsymbol{n} = [\![\boldsymbol{\beta}(t,\boldsymbol{x})]\!]_\Gamma \cdot \boldsymbol{n}. \tag{2.10}
$$

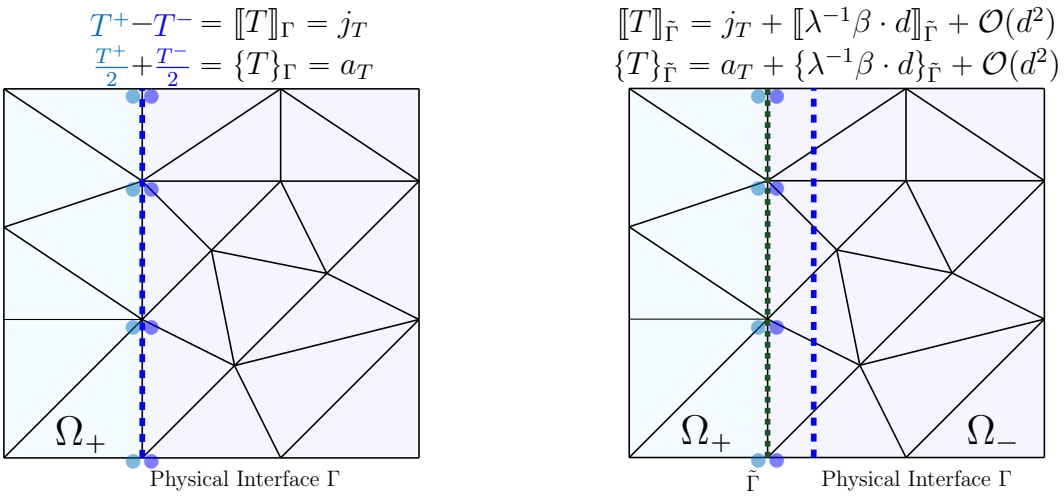
In Figure (2.5), the imposition of the conditions for both conform and embedded resolutions are displayed taking into account the definition of the flux β.



(a) Conform Resolution (b) Embedded Resolution

Figure 2.5: Set 1 : Visualization of the imposition of the boundary conditions considering $\beta = -\lambda\nabla T$

where d^t is the notation used to refer to the transpose of the vector d.
For the second set of interface conditions (2.7) the differences with the primal form of the Stefan model are for embedded resolutions only (see Figure (2.6).b). The definition of the conditions (2.7) is still valid for the mixed Problem (2.8) The difference is that β is now used in the Taylor developments of the condition on Γ for an imposition on $\tilde{\Gamma}$. Definition (2.10) is also used for the treatment of the Stefan condition for the second set of interface conditions (2.7).



(a) Conform Resolution

(b) Embedded Resolution

Figure 2.6: Set 2 : Visualization of the imposition of the boundary conditions considering $\beta = -\lambda \nabla T$

In the proceeding sections, only the mixed formulation is considered.

2.1.4 Matrix resolution approach

The finite weak formulation is the weak form associated to the space of approximation used to approximate the space of infinite dimension of the initial system of PDEs. The weak formulation can be associated to a matrix resolution approach where all the terms in the model are associated to a specific matrix or vector. Let the vector $U = (\beta_x, \beta_y, T)'$ be the vector of unknowns, such that $\dot{U} = (\partial_t \beta_x, \partial_t \beta_y, \partial_t T)'$, then the quasi-weak formulation, i.e., the weak formulation without the application of any time discretization schemes, consists of solving the following matrix system

$$M\dot{U} \; + \; AU \; = \; F \; + \; S \tag{2.11}$$

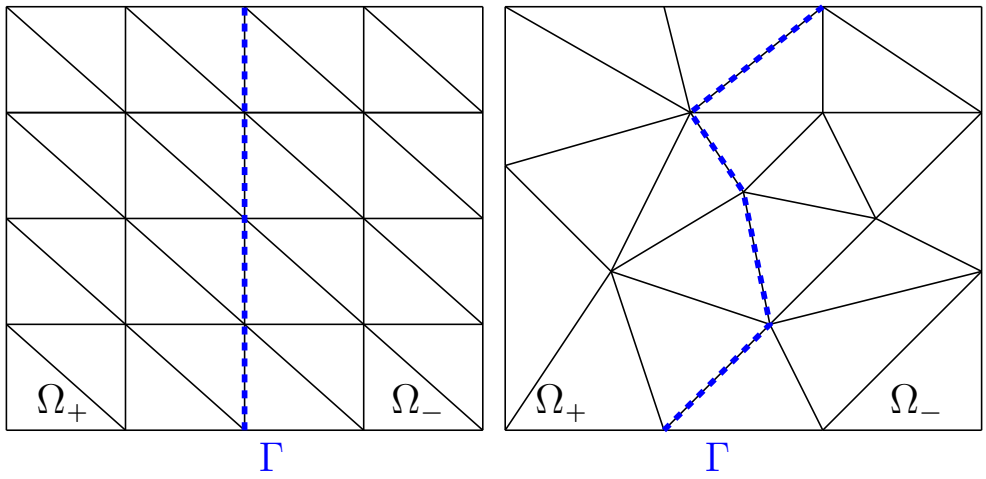
where

- U is the the vector of unknowns such that U=(β_x, β_y, T),
- $\dot{U}$ is the derivative of U with respect to time t,
- A is the matrix of the steady problem,
- M is the matrix of the time dependent terms,
- F is the vector taking into account the source term $f(x,t)$,
- S is the vector taking into account the boundary conditions.

There are specific methods called "time-space" FEM, see [121], which use test functions depending explicitly on the time variable. In this work, the proposed method is not one of them. The choice made for the treatment of the time dependent terms will be addressed in Section 2.4.4 of this Chapter.

2.2 Conformal case

In order to decouple the complexity of the problem, the case where the mesh is grid conformed to the geometry is first considered, i.e., where the use of the Shifted Boundary Method is not required, see Figure 2.7. It is a good starting point to understand the construction of the weak formulation 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. [5, 6, 7]), and extended in some recent articles, see for instance [8, 9, 122].



(a) Structured Mesh (b) Unstructured Mesh

Figure 2.7: Conform resolution - Interface Γ conform to the mesh geometry

2.2.1 Notations

In order to derive properly the variational formulation of System (2.8), one can define

$$\begin{aligned} \mathcal{W}_T(\Omega) &= \{T \in H^1(\Omega_+) \cup H^1(\Omega_-)\}, \\ \mathcal{W}_\beta(\Omega) &= \{\boldsymbol{\beta} \in (H(\mathrm{div}, \Omega_+) \cup H(\mathrm{div}, \Omega_-))\}, \end{aligned} \tag{2.12}$$

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

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

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

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

Finally, one can define

$$\boldsymbol{W}(]0,t_f[,\Omega) = \mathbb{L}^2(]0,t_f[,\mathcal{W}_T(\Omega)) \times \mathbb{L}^2(]0,t_f[,\mathcal{W}_B(\Omega)), \tag{2.13}$$

and

$$\boldsymbol{W}(\Omega) = \mathcal{W}_T(\Omega) \times \mathcal{W}_B(\Omega). \tag{2.14}$$

The classical weak formulation is obtained by taking the L^2 scalar product of the first-two equations of System (2.8) with test functions. The function q will denote the test function associated to the temperature field T, and $\boldsymbol{w}$ the test function associated to the flux $\boldsymbol{\beta}$. The two test functions are not of the same dimension, q being of dimension 1, while $\boldsymbol{w}$ is of dimension d (d being the number of components of the flux $\boldsymbol{\beta}$).
It can be formulated in the following way :

Find $(T,\boldsymbol{\beta}) \in \boldsymbol{W}(]0,t_f[,\Omega)$ such that $\forall\ (q,\boldsymbol{w}) \in \boldsymbol{W}(\Omega)$ one has

$$\begin{aligned} &2.15.a)\ (\lambda^{-1}\boldsymbol{\beta},\boldsymbol{w})_\Omega + (\nabla T,\boldsymbol{w})_\Omega = 0, \\ &2.15.b)\ (\rho c\partial_t T,q)_\Omega + (\nabla\cdot\boldsymbol{\beta},q)_\Omega = (f,q)_\Omega. \end{aligned} \tag{2.15}$$

Note that, for the sake of clarity, at this step no time discretization is used on the variables. The latter formulation (2.15), which is for infinite-dimensional spaces, is now considered in its discrete spatial version.

Let $\mathcal{T}_h$ be a tessellation of the domain Ω, in which elements are denoted by K and the boundaries by ∂K. Let h be the measure of an element, and N_v and N_e the number of nodes and elements of the mesh $\mathcal{T}_h$.

which correspond to sets of piece-wise polynomials of order l for the temperature, and of order p for the flux, which are continuous on each subdomain Ω_+ and Ω_-. Note that in this present work, the focus is on piece-wise linear approximations of the variables, i.e., where l = p = 1.

2.2.2 First Set of Interface Conditions

In this section, the imposition of the flux jump and temperature jump into the weak formulation are discussed according to Conditions (2.9).

2.2.2.a Weak formulation

The Galerkin weak formulation of Problem (2.15) consists of looking for a solution $(T,\boldsymbol{\beta})$ belonging to $\boldsymbol{W}^{l,p}(]0,t_f[,\Omega)$ such that $\forall\ (q,\boldsymbol{w}) \in \boldsymbol{W}^{l,p}(\Omega)$ one has

$$\begin{aligned} &2.16.a)\ (\lambda^{-1}\boldsymbol{\beta},\boldsymbol{w})_\Omega + (\nabla T,\boldsymbol{w})_\Omega = 0\ , \\ &2.16.b)\ (\rho c\partial_t T,q)_\Omega + (\nabla\cdot\boldsymbol{\beta},q)_\Omega = (f,q)_\Omega\ . \end{aligned} \tag{2.16}$$

Using the decomposition $\Omega = \Omega_+ \cup \Omega_-$ and Green's formula (1.6) separately on Ω_+ and Ω_- one obtains

$$\begin{aligned} &2.17.a)\ (\lambda^{-1}\boldsymbol{\beta},\boldsymbol{w})_\Omega - (T,\nabla.\boldsymbol{w})_\Omega + <T\ ,\ \boldsymbol{w}\cdot\boldsymbol{n}>_{\partial\Omega} + <[\![\boldsymbol{w}T]\!]_\Gamma,\boldsymbol{n}>_\Gamma = 0, \\ &2.17.b)\ (\rho c\partial_t T,q)_\Omega - (\boldsymbol{\beta},\nabla q)_\Omega + <\boldsymbol{\beta}\cdot\boldsymbol{n},q>_{\partial\Omega} + <[\![q\boldsymbol{\beta}]\!]_\Gamma,\boldsymbol{n}>_\Gamma = (f,q)_\Omega. \end{aligned} \tag{2.17}$$

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

$$\begin{aligned} &2.18.a)\ (\lambda^{-1}\boldsymbol{\beta},\boldsymbol{w})_\Omega-(T,\nabla\cdot\boldsymbol{w})_\Omega+<T,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_N}+<T_D,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_D}+<[\![\boldsymbol{w}T]\!]_\Gamma,\boldsymbol{n}>_\Gamma=0,\\ &2.18.b)\ (\rho c\partial_t T,q)_\Omega-(\boldsymbol{\beta},\nabla q)_\Omega+<\boldsymbol{\beta}\cdot\boldsymbol{n},q>_{\Gamma_D}+<h_N,q>_{\Gamma_N}+<[\![q\boldsymbol{\beta}]\!]_\Gamma,\boldsymbol{n}>_\Gamma=(f,q)_\Omega. \end{aligned} \tag{2.18}$$

Using the jump decomposition (1.26) for the terms depending on Γ and using the interface conditions (2.9), one gets

$$\begin{aligned} &2.19.a)<[\![\boldsymbol{w}T]\!]_\Gamma,\boldsymbol{n}>_\Gamma=<\{T\}_\Gamma,[\![\boldsymbol{w}]\!]_\Gamma\cdot\boldsymbol{n}>_\Gamma+<j_T(\boldsymbol{x},t),\{\boldsymbol{w}\}_\Gamma\cdot\boldsymbol{n}>_\Gamma,\\ &2.19.b)<[\![q\boldsymbol{\beta}]\!]_\Gamma,\boldsymbol{n}>_\Gamma=<\{\boldsymbol{\beta}\}_\Gamma\cdot\boldsymbol{n},[\![q]\!]_\Gamma>_\Gamma+<\sigma(\boldsymbol{x},t),\{q\}_\Gamma>_\Gamma. \end{aligned} \tag{2.19}$$

Plugging (2.19) into (2.18), and performing again in (2.18.b) a Green's formula on Ω_+ and Ω_-, one can derive the final weak formulation.

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

$$\begin{aligned} &2.20.a)\ (\lambda^{-1}\boldsymbol{\beta},\boldsymbol{w})_\Omega-(T,\nabla\cdot\boldsymbol{w})_\Omega+<T,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_N}+<T_D,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_D}\\ &\qquad\qquad+<\{T\}_\Gamma,[\![\boldsymbol{w}]\!]_\Gamma\cdot\boldsymbol{n}>_\Gamma+<j_T,\{\boldsymbol{w}\}_\Gamma\cdot\boldsymbol{n}>_\Gamma=0,\\ &2.20.b)\ (\rho c\partial_t T,q)_\Omega+(\nabla\cdot\boldsymbol{\beta},q)_\Omega-<\boldsymbol{\beta}\cdot\boldsymbol{n}-h_N,q>_{\Gamma_N}-<[\![\boldsymbol{\beta}]\!]_\Gamma\cdot\boldsymbol{n}-\sigma,\{q\}_\Gamma>_\Gamma=(f,q)_\Omega. \end{aligned} \tag{2.20}$$

Any solution $(T,\boldsymbol{\beta})$ of (2.20) with $T\in\mathbb{L}^2(]0,t_f[,\mathcal{W}_T^l(\Omega))$, $\boldsymbol{\beta}\in\mathbb{L}^2(]0,t_f[,\mathcal{W}_\beta^p(\Omega))$, $\partial_t T\in\mathbb{L}^2(]0,t_f[,\mathbb{H}^{-1}(\Omega))$ is called a **weak solution** of System (2.8,2.9) or (2.8,2.7).

2.2.2.b Stabilization

The previous formulation (2.20) is known to be stable for only certain choices of polynomial order of approximation. Following [5, 6, 7, 8, 9] two different kinds of stabilization terms are added to the proposed method. The first one is called the **div-div stabilization** and is equal to

$$\frac{\zeta_{div}h^2}{2\lambda}(\rho c\partial_t T+\nabla\cdot\boldsymbol{\beta}-f\ ,\ \nabla\cdot\boldsymbol{w})_\Omega\ , \tag{2.21}$$

where ζ_{div} is a coefficient satisfying $\zeta_{div}=\mathcal{O}(1)$. This div-div stabilization term (2.21) counterbalances the error induced by the discretization of the divergence operator. The other stabilization used is the so-called **momentum stabilization** and is equal to

$$-1/2(\boldsymbol{\beta}+\lambda\nabla T\ ,\ \lambda^{-1}\boldsymbol{w}-\nabla q)_\Omega. \tag{2.22}$$

It can be seen as a correction term using the residual of the equation $\boldsymbol{\beta}+\lambda\nabla T=0$, called the momentum equation. For convenience, the following terms A_{stab}, B_{stab} and L_{stab} are introduced

$$\begin{aligned} A_{stab}((T,\boldsymbol{\beta}),q) &:= \frac{1}{2}\ (\boldsymbol{\beta}+\lambda\nabla T\ ,\ \nabla q)_\Omega,\\ B_{stab}((T,\boldsymbol{\beta}),\boldsymbol{w}) &:= \frac{\zeta_{div}h^2}{2\lambda}(\rho c\partial_t T+\nabla\cdot\boldsymbol{\beta}\ ,\ \nabla\cdot\boldsymbol{w})_\Omega-1/2\ (\boldsymbol{\beta}+\lambda\nabla T\ ,\ \lambda^{-1}\boldsymbol{w})_\Omega,\\ L_{stab}(\boldsymbol{w}) &:= \frac{\zeta_{div}h^2}{2\lambda}(f\ ,\ \nabla\cdot\boldsymbol{w})_\Omega. \end{aligned} \tag{2.23}$$

Plugging the different stabilization terms into (2.20), one obtains the following new formulation:

Find $(T, \boldsymbol{\beta}) \in \boldsymbol{W}^{l,p}(]0, t_f[, \Omega)$ such that $\forall\ (q, \boldsymbol{w}) \in \boldsymbol{W}^{l,p}(\Omega)$

$$
\begin{aligned}
&2.24.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T, \nabla\cdot\boldsymbol{w})_\Omega + <\{T\}_\Gamma, [\![\boldsymbol{w}]\!]_\Gamma\cdot\boldsymbol{n}>_\Gamma + <T, \boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_N} \\
&\qquad + \ B_{stab}((T,\boldsymbol{\beta}),\boldsymbol{w}) = \ L_{stab}(\boldsymbol{w}) - <T_D, \boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_D} - <j_T, \{\boldsymbol{w}\}_\Gamma\cdot\boldsymbol{n}>_\Gamma, \\
&2.24.b)\ (\rho c\partial_t T, q)_\Omega + (\nabla\cdot\boldsymbol{\beta}, q)_\Omega - <\boldsymbol{\beta}\cdot\boldsymbol{n}, q>_{\Gamma_N} - <[\![\boldsymbol{\beta}]\!]_\Gamma\cdot\boldsymbol{n}, \{q\}>_\Gamma \\
&\qquad + \ A_{stab}((T,\boldsymbol{\beta}),q) = (f,q)_\Omega - <h_N, q>_{\Gamma_N} - <\sigma, \{q\}>_\Gamma,
\end{aligned}
\tag{2.24}
$$

Remark 6. The stabilization terms are independent of the use of the SBM.
Moreover, these stabilization terms are local, with h defined as the square root of the element's area. The temperature enrichment, explained later in Section 2.4, guarantees second-order accuracy in the flux, regardless of the choice of stabilization weight. Rather than proposing a new stabilization, the decision has been made to use one that has already been established and proven effective. For further information on the stabilization, please refer to references [5, 6].

2.2.2.c Nitsche penalty terms

In order to enforce the Dirichlet condition on Γ_D (see [123]) and the temperature jump on Γ (see [50]), Nitsche penalty terms which are the consistent terms $<\frac{\alpha}{h}(T - T_D), q>_{\Gamma_D}$ and $<\frac{\alpha}{h}([\![T]\!]_\Gamma - j_T), [\![q]\!]_\Gamma>_\Gamma$ are added to Equations (2.24). In addition, it has been shown in [6, 8, 9] that the jump penalty terms provide the theoretical stability of the weak formulation, using coercivity arguments. To lighten the definition of the weak formulation the following terms are introduced

$$
\begin{aligned}
N_{A_1}(T,q) &:= \frac{\alpha}{h} <[\![T]\!]_\Gamma, [\![q]\!]_\Gamma>_\Gamma + \frac{\alpha}{h} <T, q>_{\Gamma_D}, \\
N_{L_1}(q) &:= \frac{\alpha}{h} <j_T, [\![q]\!]_\Gamma>_\Gamma + \frac{\alpha}{h} <T_D, q>_{\Gamma_D},
\end{aligned}
\tag{2.25}
$$

The terms in (2.25) are added to formulation (2.20). The Nitsche terms are separated into two distinct terms, one part is added to the bilinear formulation (N_{A_1}), while the other is added to the linear form of the weak formulation (N_{L_1}), terms added in the right-hand side of the system to be solved.

$$
\begin{aligned}
&2.26.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T, \nabla\cdot\boldsymbol{w})_\Omega + <\{T\}_\Gamma, [\![\boldsymbol{w}]\!]_\Gamma\cdot\boldsymbol{n}>_\Gamma + <T, \boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_N} \\
&\qquad + \ B_{stab}((T,\boldsymbol{\beta}),\boldsymbol{w}) = \ L_{stab}(\boldsymbol{w}) - <T_D, \boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_D} - <j_T, \{\boldsymbol{w}\}_\Gamma\cdot\boldsymbol{n}>_\Gamma, \\
&2.26.b)\ (\rho c\partial_t T, q)_\Omega + (\nabla\cdot\boldsymbol{\beta}, q)_\Omega - <\boldsymbol{\beta}\cdot\boldsymbol{n}, q>_{\Gamma_N} - <[\![\boldsymbol{\beta}]\!]_\Gamma\cdot\boldsymbol{n}, \{q\}>_\Gamma \\
&\qquad + \ A_{stab}((T,\boldsymbol{\beta}),q) + N_{A_1}(T,q) = (f,q)_\Omega - <h_N, q>_{\Gamma_N} - <\sigma, \{q\}>_\Gamma + N_{L_1}(q),
\end{aligned}
\tag{2.26}
$$

Remark 7. In [6, 7, 8, 9], the questions of stability and convergence for the steady version of (2.8) are addressed. If $(T, \boldsymbol{\beta}) \in \boldsymbol{W}^{l,p}(]0, t_f[, \Omega)$, 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 provides respectively a first order accuracy for the flux and a second order accuracy for the temperature. In Section 2.4 a method to increase the order of accuracy of the flux while keeping piecewise linear test functions will be presented.

2.2.3 Second Set of interface conditions

The aim of this section is to present the finite weak formulation derived from (2.8) using the second set of interfaces conditions according to (2.7).

2.2.3.a Weak formulation

The Galerkin weak formulation of Problem (2.15) can be written as:

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

$$\begin{aligned} &2.27.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega + (\nabla T, \boldsymbol{w})_\Omega = 0\ , \\ &2.27.b)\ (\rho c \partial_t T, q)_\Omega + (\nabla \cdot \boldsymbol{\beta}, q)_\Omega = (f, q)_\Omega\ . \end{aligned} \tag{2.27}$$

In a similar way that the work performed for the first set of interface conditions (2.9) (see Section 2.2.2.a), the Green's formula (1) is used separately on Ω_+ and Ω_- for (2.27.a) and on Ω for (2.27.b). This provides

$$\begin{aligned} &2.28.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T, \nabla \cdot \boldsymbol{w})_\Omega + < T, \boldsymbol{w} \cdot \boldsymbol{n} >_{\Gamma_N} + < T_D, \boldsymbol{w} \cdot \boldsymbol{n} >_{\Gamma_D} + < [\![\boldsymbol{w}T]\!]_\Gamma, \boldsymbol{n} >_\Gamma = 0, \\ &2.28.b)\ (\rho c \partial_t T, q)_\Omega - (\boldsymbol{\beta}, \nabla q)_\Omega + < \boldsymbol{\beta} \cdot \boldsymbol{n}, q >_{\Gamma_D} + < h_N, q >_{\Gamma_N} = (f, q)_\Omega. \end{aligned} \tag{2.28}$$

Using the relation (1.26) with the interface conditions (2.7), one obtains

$$< [\![\boldsymbol{w}T]\!]_\Gamma, \boldsymbol{n} >_\Gamma = < j_T, \{\boldsymbol{w}\}_\Gamma \cdot \boldsymbol{n} >_\Gamma + < a_T, [\![\boldsymbol{w}]\!]_\Gamma \cdot \boldsymbol{n} >_\Gamma\ . \tag{2.29}$$

Plugging (2.29) into (2.28.a) and performing again Green's formula on Ω in (2.28.b), the following weak formulation is obtained:

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

$$\begin{aligned} &2.30.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T, \nabla \cdot \boldsymbol{w})_\Omega + < T, \boldsymbol{w} \cdot \boldsymbol{n} >_{\Gamma_N} + < T_D, \boldsymbol{w} \cdot \boldsymbol{n} >_{\Gamma_D} \\ &\qquad\qquad + < a_T, [\![\boldsymbol{w}]\!]_\Gamma \cdot \boldsymbol{n} >_\Gamma + < j_T, \{\boldsymbol{w}\}_\Gamma \cdot \boldsymbol{n} >_\Gamma = 0, \\ &2.30.b)\ (\rho c \partial_t T, q)_\Omega + (\nabla \cdot \boldsymbol{\beta}, q)_\Omega - < \boldsymbol{\beta} \cdot \boldsymbol{n} - h_N, q >_{\Gamma_N} = (f, q)_\Omega. \end{aligned} \tag{2.30}$$

Any solution $(T,\boldsymbol{\beta})$ of (2.30) with $T \in \mathbb{L}^2(]0, t_f[, \mathcal{W}_T^l(\Omega))$, $\boldsymbol{\beta} \in \mathbb{L}^2(]0, t_f[, \mathcal{W}_\beta^p(\Omega))$, $\partial_t T \in \mathbb{L}^2(]0, t_f[, \mathbb{H}^{-1}(\Omega))$ is called a **weak solution** of System (2.8-2.9) or (2.8-2.7).

2.2.3.b Stabilization

The formulation (2.30) is stabilized using the div-div and the momentum stabilizations introduced for the first set of interface condition in (2.23). Plugging (2.23) into (2.30) the following stabilized formulation can be defined:

Find $(T, \boldsymbol{\beta}) \in \boldsymbol{W}^{l,p}(]0, t_f[, \Omega)$ such that $\forall\ (q, \boldsymbol{w}) \in \boldsymbol{W}^{l,p}(\Omega)$

$$\begin{aligned} &2.31.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T, \nabla \cdot \boldsymbol{w})_\Omega + < T, \boldsymbol{w} \cdot \boldsymbol{n} >_{\Gamma_N} + < T_D, \boldsymbol{w} \cdot \boldsymbol{n} >_{\Gamma_D} + B_{stab}((T, \boldsymbol{\beta}), \boldsymbol{w}) \\ &\qquad\qquad + < a_T, [\![\boldsymbol{w}]\!]_\Gamma \cdot \boldsymbol{n} >_\Gamma + < j_T, \{\boldsymbol{w}\}_\Gamma \cdot \boldsymbol{n} >_\Gamma = L_{stab}(\boldsymbol{w}), \\ &2.31.b)\ (\rho c \partial_t T, q)_\Omega + (\nabla \cdot \boldsymbol{\beta}, q)_\Omega - < \boldsymbol{\beta} \cdot \boldsymbol{n} - h_N, q >_{\Gamma_N} + A_{stab}((T, \boldsymbol{\beta}), q) = (f, q)_\Omega. \end{aligned} \tag{2.31}$$

2.2.3.c The Nitche's penalty terms

In order to enforce the Dirichlet condition on Γ_D (see [123]) and the temperature jump on Γ (see [50]), the penalization of Nitsche is used by adding the consistent terms $< \frac{\alpha}{h}(T - T_D), q >_{\Gamma_D}$, $< \frac{\alpha}{h}([\![T]\!]_\Gamma - j_T), [\![q]\!]_\Gamma >_\Gamma$ and $< \frac{\alpha}{h}(\{T\}_\Gamma - a_T), \{q\}_\Gamma >_\Gamma$ to Equation (2.31). To lighten the

definition of the weak formulation the following terms are introduced

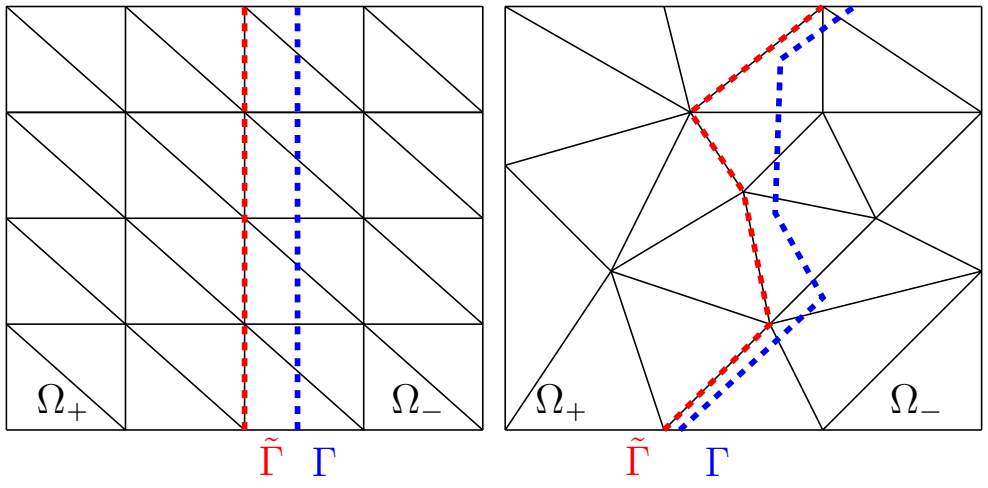
$$
\begin{aligned}
N_{A_2}(T,q) &:= \frac{\alpha}{h} < [\![T]\!]_\Gamma, [\![q]\!]_\Gamma >_\Gamma + \frac{\alpha}{h} < \{T\}_\Gamma, \{q\}_\Gamma >_\Gamma + \frac{\alpha}{h} < T, q >_{\Gamma_D}, \\
N_{L_2}(q) &:= \frac{\alpha}{h} < j_T, [\![q]\!]_\Gamma >_\Gamma + \frac{\alpha}{h} < a_T, \{q\}_\Gamma >_\Gamma + \frac{\alpha}{h} < T_D, q >_{\Gamma_D} .
\end{aligned}
\tag{2.32}
$$

The Nitsche terms are separated into two distinct terms, one part is added to the bilinear formulation (N_{A_2}), while the other part is added to the linear form of the weak formulation (N_{L_2}).

$$
\begin{aligned}
2.33.a)\ & (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T, \nabla\cdot\boldsymbol{w})_\Omega + < T, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_N} + < T_D, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_D} + B_{stab}((T,\boldsymbol{\beta}),\boldsymbol{w}) \\
& \qquad + < a_T, [\![\boldsymbol{w}]\!]_\Gamma \cdot \boldsymbol{n} >_\Gamma + < j_T, \{\boldsymbol{w}\}_\Gamma \cdot \boldsymbol{n} >_\Gamma = L_{stab}(\boldsymbol{w}), \\
2.33.b)\ & (\rho c \partial_t T, q)_\Omega + (\nabla\cdot\boldsymbol{\beta}, q)_\Omega - < \boldsymbol{\beta}\cdot\boldsymbol{n} - h_N, q >_{\Gamma_N} + N_{A_2}(T,q) + A_{stab}((T,\boldsymbol{\beta}),q) \\
& \qquad = (f,q)_\Omega + N_{L_2}(q).
\end{aligned}
\tag{2.33}
$$

In this section, the results of a conform resolution have been presented to facilitate the initial understanding of the proposed method. In the next section, results for an embedded resolution will be introduced which will be discussed later on in Chapter 3 with a chapter results. The embedded approach allows to handle the moving interface without the need to adapt the mesh.

2.3 Embedded case



(a) Structured Mesh (b) Unstructured Mesh

Figure 2.8: Embedded resolution - Interface Γ non conform to the mesh geometry and its surrogate interface $\tilde{\Gamma}$

In this section, the situation described in Section 2.2 is considered, and the Shifted Boundary Method is included for the treatment of the boundary conditions, see Figure 2.8. A configuration where the interface is non-fitted to the mesh of the domain is considered. The first step consists in rewriting Equations (2.4) and (2.7) using Taylor expansions between Γ and $\tilde{\Gamma}$

$$
\begin{aligned}
4.121.a)\ & [\![T]\!]_\Gamma = j_T = [\![T + \nabla T\cdot\boldsymbol{d}]\!]_{\tilde{\Gamma}} + \mathcal{O}(||\boldsymbol{d}(\tilde{\boldsymbol{x}})||^2), \\
4.121.b)\ & [\![\boldsymbol{\beta}]\!]_\Gamma \cdot \boldsymbol{n} = \sigma = [\![\boldsymbol{\beta} + \nabla\boldsymbol{\beta}\ \boldsymbol{d}]\!]_{\tilde{\Gamma}} \cdot \boldsymbol{n} + \mathcal{O}(||\boldsymbol{d}(\tilde{\boldsymbol{x}})||^2), \\
4.121.c)\ & \{T\}_\Gamma = a_T = \{T + \nabla T\ \cdot\boldsymbol{d}\}_{\tilde{\Gamma}} + \mathcal{O}(||\boldsymbol{d}(\tilde{\boldsymbol{x}})||^2),
\end{aligned}
\tag{2.34}
$$

where $\boldsymbol{d}$ is the distance vector, see Equation (1.13).
The normal $\tilde{\boldsymbol{n}}$ to $\tilde{\Gamma}$ oriented from Ω_+ to Ω_- (see Figure 2.1) is decomposed on the basis $(\boldsymbol{\tau}, \boldsymbol{n})$, where $\boldsymbol{n}$ is the outward normal on Γ and $\boldsymbol{\tau}$ its associated tangential vector.

$$\tilde{\boldsymbol{n}} = (\tilde{\boldsymbol{n}} \cdot \boldsymbol{n})\boldsymbol{n} + (\tilde{\boldsymbol{n}} \cdot \boldsymbol{\tau})\boldsymbol{\tau}. \tag{2.35}$$

Considering $\tilde{\Gamma}(t)$ the mutual boundary between $\tilde{\Omega}_+$ and $\tilde{\Omega}_-$ the following spaces are introduced

$$\begin{aligned} \mathcal{Q}_T(\Omega) &= \{T \in H^1(\tilde{\Omega}_+) \cup H^1(\tilde{\Omega}_-)\}, \\ \mathcal{Q}_\beta(\Omega) &= \{\boldsymbol{\beta} \in (H(\mathrm{div}, \tilde{\Omega}_+) \cup H(\mathrm{div}, \tilde{\Omega}_-))\}, \end{aligned} \tag{2.36}$$

and their finite-dimensional versions

$$\begin{aligned} \mathcal{Q}_T^l(\Omega) &= \{T \in \mathcal{Q}_T(\Omega) \,|\, T|_K \in \mathbb{P}^l(K),\ \forall K \in \mathcal{T}_h\} \cap (C^0(\tilde{\Omega}^+) \cup C^0(\tilde{\Omega}^-)) \ , \\ \mathcal{Q}_\beta^p(\Omega) &= \{\boldsymbol{\beta} \in \mathcal{Q}_\beta(\Omega) \,|\, \boldsymbol{\beta}|_K \in (\mathbb{P}^p(K))^d,\ \forall K \in \mathcal{T}_h\} \cap ((C^0(\tilde{\Omega}^+))^d \cup (C^0(\tilde{\Omega}^-))^d) \ . \end{aligned} \tag{2.37}$$

Finally one can consider

$$\boldsymbol{Q}^{l,p}(]0, t_f[, \Omega) = \mathbb{L}^2(]0, t_f[, \mathcal{Q}_T^l(\Omega)) \times \mathbb{L}^2(]0, t_f[, \mathcal{Q}_B^p(\Omega)) \ , \tag{2.38}$$

$$\boldsymbol{Q}^{l,p}(\Omega) = \mathcal{Q}_T^l(\Omega) \times \mathcal{Q}_B^p(\Omega) \ . \tag{2.39}$$

All the information necessary to derive the weak formulations corresponding to an embedded situation has been introduced.

2.3.1 First set of interface conditions

In this section, the imposition of the first set of interface conditions (2.9) and its imposition in the context of an embedded resolution is discussed. The comparison with the conform approach is displayed in Figure 2.5.

Repeating the arguments of Section (2.2.1), that is applying Green's formula (1) on (2.27.a-2.27.b) and recalling that $\Omega = \tilde{\Omega}_+ \cup \tilde{\Omega}_-$, one obtains

$$\begin{aligned} &2.40.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T, \nabla \cdot \boldsymbol{w})_\Omega + < T \ , \ \boldsymbol{w} \cdot \boldsymbol{n} >_{\partial\Omega} + < [\![\boldsymbol{w}T]\!]_{\tilde{\Gamma}}, \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} = 0, \\ &2.40.b)\ (\rho c \partial_t T, q)_\Omega - (\boldsymbol{\beta}, \nabla q)_\Omega + < \boldsymbol{\beta} \cdot \boldsymbol{n}, q >_{\partial\Omega} + < [\![q\boldsymbol{\beta}]\!]_{\tilde{\Gamma}}, \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} = (f, q)_\Omega. \end{aligned} \tag{2.40}$$

The jump term in (2.40.a) is handled combining (1.26) and (4.121.a), where the terms of order 2 with respect to $\boldsymbol{d}$ are neglected.

$$< [\![\boldsymbol{w}T]\!]_{\tilde{\Gamma}}, \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} = < \{T\}_{\tilde{\Gamma}}, [\![\boldsymbol{w}]\!]_{\tilde{\Gamma}} \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} + < j_T - [\![\nabla T \cdot \boldsymbol{d}]\!]_{\tilde{\Gamma}}, \{\boldsymbol{w}\}_{\tilde{\Gamma}} \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} \ . \tag{2.41}$$

The treatment of the jump term in (2.40.b) is more delicate and requires the decomposition of the interface normal $\tilde{\boldsymbol{n}}$ (2.35). Indeed, the extrapolation of the flux is known in the normal direction of the physical interface $\boldsymbol{n}$. Thus, when applying the jump decomposition (1.26), one needs to project the normal $\tilde{\boldsymbol{n}}$ to the surrogate interface onto $\boldsymbol{n}$ and $\boldsymbol{\tau}$, before using the truncated extrapolation (4.121.b), providing:

$$\begin{aligned} < [\![q\boldsymbol{\beta}]\!]_\Gamma, \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} = &< \{\boldsymbol{\beta}\}_{\tilde{\Gamma}} \cdot \tilde{\boldsymbol{n}}, [\![q]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}} + < [\![\boldsymbol{\beta}]\!]_{\tilde{\Gamma}} \cdot \boldsymbol{\tau}(\boldsymbol{\tau} \cdot \tilde{\boldsymbol{n}}), \{q\}_{\tilde{\Gamma}} >_{\tilde{\Gamma}} \\ &+ < (\sigma - [\![\nabla \boldsymbol{\beta}\ \boldsymbol{d}]\!]_{\tilde{\Gamma}} \cdot \boldsymbol{n})(\boldsymbol{n} \cdot \tilde{\boldsymbol{n}}), \{q\}_{\tilde{\Gamma}} >_{\tilde{\Gamma}} \ . \end{aligned} \tag{2.42}$$

Plugging (2.41-2.42) into (2.40) the formulation becomes

$$\begin{aligned}
&2.43.a)\ (\lambda^{-1}\boldsymbol{\beta},\boldsymbol{w})_\Omega-(T,\nabla\cdot\boldsymbol{w})_\Omega+<T\ ,\ \boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_N}+<T_D,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_D}\\
&\qquad+<\{T\}_{\tilde\Gamma},[\![\boldsymbol{w}]\!]_{\tilde\Gamma}\cdot\tilde{\boldsymbol{n}}>_{\tilde\Gamma}+<j_T-[\![\nabla T\cdot\boldsymbol{d}]\!]_{\tilde\Gamma},\{\boldsymbol{w}\}\cdot\tilde{\boldsymbol{n}}>_{\tilde\Gamma}=0,\\
&2.43.b)\ (\rho c\partial_t T,q)_\Omega-(\boldsymbol{\beta},\nabla q)_\Omega+<\boldsymbol{\beta}\cdot\boldsymbol{n},q>_{\Gamma_D}+<h_N,q>_{\Gamma_N}+<\{\boldsymbol{\beta}\}_{\tilde\Gamma}\cdot\tilde{\boldsymbol{n}},[\![q]\!]_{\tilde\Gamma}>_{\tilde\Gamma}\\
&\qquad+<[\![\boldsymbol{\beta}]\!]_{\tilde\Gamma}\cdot\boldsymbol{\tau}(\boldsymbol{\tau}\cdot\tilde{\boldsymbol{n}}),\{q\}_{\tilde\Gamma}>_{\tilde\Gamma}+<(\sigma-[\![\nabla\boldsymbol{\beta}\ \boldsymbol{d}]\!]_{\tilde\Gamma}\cdot\boldsymbol{n})(\boldsymbol{n}\cdot\tilde{\boldsymbol{n}}),\{q\}_{\tilde\Gamma}>_{\tilde\Gamma}=(f,q)_\Omega.
\end{aligned}\tag{2.43}$$

Now, performing again a Green's formula in (2.43.b), using the decomposition $\Omega=\tilde\Omega_+\cup\tilde\Omega_-$, one obtains, after a straightforward calculation, the following weak formulation:

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

$$\begin{aligned}
&2.44.a)\ (\lambda^{-1}\boldsymbol{\beta},\boldsymbol{w})_\Omega-(T,\nabla\cdot\boldsymbol{w})_\Omega+<T\ ,\ \boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_N}+<T_D,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_D}\\
&\qquad+<\{T\}_{\tilde\Gamma},[\![\boldsymbol{w}]\!]_{\tilde\Gamma}\cdot\tilde{\boldsymbol{n}}>_{\tilde\Gamma}+<j_T-[\![\nabla T\cdot\boldsymbol{d}]\!]_{\tilde\Gamma},\{\boldsymbol{w}\}\cdot\tilde{\boldsymbol{n}}>_{\tilde\Gamma}=0,\\
&2.44.b)\ (\rho c\partial_t T,q)_\Omega+(\nabla\cdot\boldsymbol{\beta},q)_\Omega-<\boldsymbol{\beta}\cdot\boldsymbol{n}-h_N,q>_{\Gamma_N}+<\{\boldsymbol{\beta}\}_{\tilde\Gamma}\cdot\tilde{\boldsymbol{n}},[\![q]\!]_{\tilde\Gamma}>_{\tilde\Gamma}\\
&\qquad+<(\sigma-[\![\boldsymbol{\beta}+\nabla\boldsymbol{\beta}\ \boldsymbol{d}]\!]_{\tilde\Gamma}\cdot\boldsymbol{n})(\boldsymbol{n}\cdot\tilde{\boldsymbol{n}}),\{q\}_{\tilde\Gamma}>_{\tilde\Gamma}=(f,q)_\Omega.
\end{aligned}\tag{2.44}$$

Before adding the Nitsche's penalty terms in (2.44), Taylor expansions are performed. As presented in [2, 95], the extrapolation can also be applied on the test functions to provide symmetry in the weak formulation. This gives

$$\begin{aligned}
\bar N_{A_1}(T,q)&:=\frac{\alpha}{h}<[\![T+\nabla T\cdot\boldsymbol{d}]\!]_{\tilde\Gamma},[\![q+\nabla q\cdot\boldsymbol{d}]\!]_{\tilde\Gamma}>_{\tilde\Gamma}+\frac{a}{h}<T,q>_{\Gamma_D},\\
\bar N_{L_1}(q)&:=\frac{\alpha}{h}<j_T,[\![q+\nabla q\cdot\boldsymbol{d}]\!]_{\tilde\Gamma}>_{\tilde\Gamma}+\frac{a}{h}<T_D,q>_{\Gamma_D}.
\end{aligned}\tag{2.45}$$

Then the stabilized formulation can be written as below :

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

$$\begin{aligned}
&2.46.a)\ (\lambda^{-1}\boldsymbol{\beta},\boldsymbol{w})_\Omega-(T,\nabla\cdot\boldsymbol{w})_\Omega+<T\ ,\ \boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_N}+<T_D,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_D}+B_{stab}((T,\boldsymbol{\beta},\boldsymbol{w})\\
&\qquad+<\{T\}_{\tilde\Gamma},[\![\boldsymbol{w}]\!]_{\tilde\Gamma}\cdot\tilde{\boldsymbol{n}}>_{\tilde\Gamma}+<j_T-[\![\nabla T\cdot\boldsymbol{d}]\!]_{\tilde\Gamma},\{\boldsymbol{w}\}\cdot\tilde{\boldsymbol{n}}>_{\tilde\Gamma}=L_{stab}(\boldsymbol{w}),\\
&2.46.b)\ (\rho c\partial_t T,q)_\Omega+(\nabla\cdot\boldsymbol{\beta},q)_\Omega-<\boldsymbol{\beta}\cdot\boldsymbol{n}-h_N,q>_{\Gamma_N}+<\{\boldsymbol{\beta}\}_{\tilde\Gamma}\cdot\tilde{\boldsymbol{n}},[\![q]\!]_{\tilde\Gamma}>_{\tilde\Gamma}\\
&+<(\sigma-[\![\boldsymbol{\beta}+\nabla\boldsymbol{\beta}\ \boldsymbol{d}]\!]_{\tilde\Gamma}\cdot\boldsymbol{n})(\boldsymbol{n}\cdot\tilde{\boldsymbol{n}}),\{q\}_{\tilde\Gamma}>_{\tilde\Gamma}+\bar N_{A_1}(T,q)+A_{stab}((T,\boldsymbol{\beta}),q)=(f,q)_\Omega+\bar N_{L_1}(q).
\end{aligned}\tag{2.46}$$

2.3.2 Second set of interface conditions

For the second set of interface condition (2.7), where the temperature is imposed on each side of the interface, only the momentum equation (2.27.a) needs to be modified. Using Green's formula (1), Equation (2.27.a) becomes

$$(\lambda^{-1}\boldsymbol{\beta},\boldsymbol{w})_\Omega-(T,\nabla\cdot\boldsymbol{w})_\Omega+<T_D,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_D}+<T,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_N}+<[\![\boldsymbol{w}T]\!]_{\tilde\Gamma},\tilde{\boldsymbol{n}}>_{\tilde\Gamma}=0.\tag{2.47}$$

Combining (1.26), (4.121.a) and (4.121.c), one obtains, at leading order,

$$<[\![T\boldsymbol{w}]\!]_{\tilde\Gamma},\tilde{\boldsymbol{n}}>_{\tilde\Gamma}=<a_T-\{\nabla T\cdot\boldsymbol{d}\}_{\tilde\Gamma},[\![\boldsymbol{w}]\!]_{\tilde\Gamma}\cdot\tilde{\boldsymbol{n}}>_{\tilde\Gamma}+<j_T-[\![\nabla T\cdot\boldsymbol{d}]\!]_{\tilde\Gamma},\{\boldsymbol{w}\}_{\tilde\Gamma}\cdot\tilde{\boldsymbol{n}}>_\Gamma.\tag{2.48}$$

As mentioned above, nothing has to be done on (2.27.b) regardless to the moving interface $\tilde\Gamma$. Thus, by plugging (2.48) into (2.47), the following weak formulation is directly obtained:

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

$$\begin{aligned}
&2.74.a)\ (\lambda^{-1}\boldsymbol{\beta},\boldsymbol{w})_\Omega - (T,\nabla\cdot\boldsymbol{w})_\Omega + <T_D,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_D} + <T,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_N} \\
&\qquad + <a_T - \{\nabla T\cdot\boldsymbol{d}\}_{\tilde{\Gamma}}, [\![\boldsymbol{w}]\!]_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}}>_{\tilde{\Gamma}} + <j_T - [\![\nabla T\cdot\boldsymbol{d}]\!]_{\tilde{\Gamma}}, \{\boldsymbol{w}\}_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}}>_{\tilde{\Gamma}} = 0, \\
&2.74.b)\ (\rho c\partial_t T, q)_\Omega + (\nabla\cdot\boldsymbol{\beta}, q)_\Omega - <\boldsymbol{\beta}\cdot\boldsymbol{n} - h_N, q>_{\Gamma_N} = (f,q)_\Omega.
\end{aligned} \tag{2.49}$$

Similar to the procedure explained for the first set of interface conditions, Taylor expansions need to be performed before being able to add the Nitsche's terms in the formulation to ensure that the accuracy of the method is not reduced. These Nitsche's terms are denoted by $\bar{N}_{A_2}(T,q)$ and $\bar{N}_{L_2}(q)$:

$$\begin{aligned}
\bar{N}_{A_2}(T,q) &:= \frac{\alpha}{h} <[\![T+\nabla T\cdot\boldsymbol{d}]\!]_{\tilde{\Gamma}}, [\![q+\nabla q\cdot\boldsymbol{d}]\!]_{\tilde{\Gamma}}>_{\tilde{\Gamma}} + \frac{\alpha}{h} <\{T+\nabla T\cdot\boldsymbol{d}\}_{\tilde{\Gamma}}, \{q+\nabla q\cdot\boldsymbol{d}\}_{\tilde{\Gamma}}>_{\tilde{\Gamma}} \\
&\qquad + \frac{a}{h} <T,q>_{\Gamma_D}, \\
\bar{N}_{L_2}(q) &:= \frac{\alpha}{h} <j_T, [\![q+\nabla q\cdot\boldsymbol{d}]\!]_{\tilde{\Gamma}}>_{\tilde{\Gamma}} + \frac{\alpha}{h} <a_T, \{q+\nabla q\cdot\boldsymbol{d}\}_{\tilde{\Gamma}}>_{\tilde{\Gamma}} + \frac{a}{h} <T_D,q>_{\Gamma_D}.
\end{aligned} \tag{2.50}$$

This leads to the following stabilized formulation

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

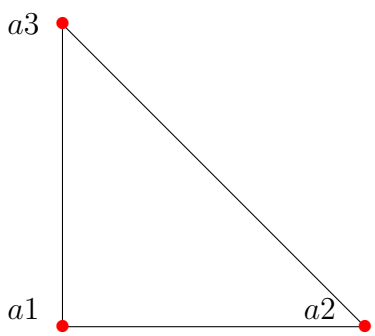
$$\begin{aligned}
&2.74.a)\ (\lambda^{-1}\boldsymbol{\beta},\boldsymbol{w})_\Omega - (T,\nabla\cdot\boldsymbol{w})_\Omega + <T_D,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_D} + <T,\boldsymbol{w}\cdot\boldsymbol{n}>_{\Gamma_N} + B_{stab}((T,\boldsymbol{\beta}),\boldsymbol{w}) \\
&\qquad + <a_T - \{\nabla T\cdot\boldsymbol{d}\}_{\tilde{\Gamma}}, [\![\boldsymbol{w}]\!]_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}}>_{\tilde{\Gamma}} + <j_T - [\![\nabla T\cdot\boldsymbol{d}]\!]_{\tilde{\Gamma}}, \{\boldsymbol{w}\}_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}}>_{\tilde{\Gamma}} = L_{stab}(\boldsymbol{w}), \\
&2.74.b)\ (\rho c\partial_t T, q)_\Omega + (\nabla\cdot\boldsymbol{\beta}, q)_\Omega - <\boldsymbol{\beta}\cdot\boldsymbol{n} - h_N, q>_{\Gamma_N} + A_{stab}((T,\boldsymbol{\beta}),q) + \bar{N}_{A_2}(T,q) \\
&\qquad = (f,q)_\Omega + \bar{N}_{L_2}(q).
\end{aligned} \tag{2.51}$$

So far, for both set of interface conditions a stabilized finite weak formulation ((2.46) and (2.74)) has been developed. In the next section of this chapter, a method to obtain a second order accuracy in space for both variables is detailed. The method only requires $\mathbb{P}^1$ basis/test functions, but allows a second order accuracy on both the primal variable and its gradient. This method will be referred as **temperature enrichment**.

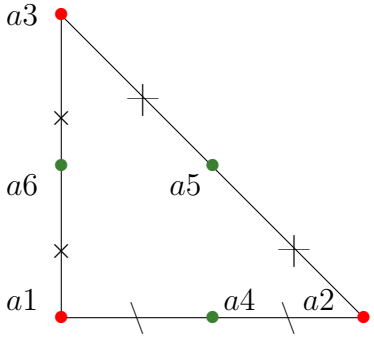
2.4 Enrichment procedure

2.4.1 Explanations

In this work, to provide a fully second order scheme in space for the temperature, the heat flux and the interface location, one needs to ensure a second order flux. To this end, the strategy proposed in [95] is followed and an enrichment procedure for the primal variable is considered. It consists of improving the weak formulation to be at least second order accurate for both variables using the link between the basis 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 using values associated to new degrees of freedom at each edge midpoint as it is shown in Figure 2.9. Only the definition of the $\mathbb{P}^1$ basis functions and their associated nodes are used, with the values of the edge midpoints being extrapolated.



(a) $\mathbb{P}^1$ element, 3 degrees of freedom



(b) $\mathbb{P}^2$ element, 6 degrees of freedom

Figure 2.9: Extrapolation of midPoints values - Comparison of a $\mathbb{P}^1$ element with a $\mathbb{P}^2$ element

Let T_h denotes the approximation of the temperature decomposed with respect to the basis $\{\phi_i^{\mathbb{P}^2}\}_{1\leqslant i\leqslant N}$, a basis of $\mathbb{P}^2(\Omega)$ of dimension N. The parameter h in the definition T_h represents the discretization in space of the domain Ω, which intends to vanish at convergence.
Then, the approximation T_h on the basis $\{\phi_i^{\mathbb{P}^2}\}_{1\leqslant i\leqslant N}$ is defined by

$$T_h(x) = \sum_{i,node} T_i \phi_i^{\mathbb{P}^2}(x) \; + \; \sum_{j,edge} T_j \phi_j^{\mathbb{P}^2}(x) \quad . \tag{2.52}$$

The approximation (2.52) is the one expected but without having to built the $\mathbb{P}^2$ basis functions or adding new points to the resolution (those necessary to transform a $\mathbb{P}^1$ element into a $\mathbb{P}^2$ element, see Figure 2.9). Let $\{\phi_i^{\mathbb{P}^1}\}_{1\leqslant i\leqslant K}$ be a basis of $\mathbb{P}^1(\Omega)$ of dimension $N = 2K$. To define the basis functions in the approximation T_h in (2.52), the relations (2.53) are introduced, where for both two and three dimensions, the basis functions of a $\mathbb{P}^2$ element ($\phi_i^{\mathbb{P}^2}$) and the basis functions of a $\mathbb{P}^1$ element ($\phi_i^{\mathbb{P}^1}$) satisfy

$$\begin{aligned} \phi_i^{\mathbb{P}^2}(x) &= \phi_i^{\mathbb{P}^1}(x)(2\phi_i^{\mathbb{P}^1}(x) - 1) \quad \text{if i a vertex,} \\ \phi_k^{\mathbb{P}^2}(x) &= 4\phi_i^{\mathbb{P}^1}(x)\phi_j^{\mathbb{P}^1}(x) \qquad \text{if k is the midpoint of the edge [i,j].} \end{aligned} \tag{2.53}$$

In Figure 2.10 the $2D$ version of Definition (2.53) is presented, where they can be expressed on the reference triangle $T_K = \{(0,0),(1,0),(0,1)\}$ by combination of linear equations. The red nodes correspond to the nodes of a $\mathbb{P}^1$ finite element while the green nodes are those added to transform a $\mathbb{P}^1$ finite element into a $\mathbb{P}^2$ finite element, see Figure 2.10.

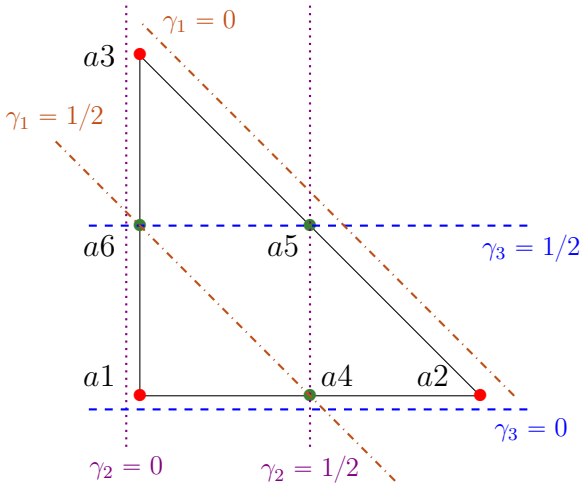


Figure 2.10: Link between the $\mathbb{P}^1$ and $\mathbb{P}^2$ basis functions on the reference triangle

Let $\{\gamma_i\}_{1\leqslant i\leqslant 3}$ be a basis of $\mathbb{P}^1(T_K)$ on the reference element T_K. Then the elements of $\{\gamma_i\}_{1\leqslant i\leqslant 3}$ are defined by

$$\begin{aligned} \gamma_1(x,y) &= 1-x-y \\ \gamma_2(x,y) &= x \qquad\qquad ;\ 0\leqslant x\leqslant 1 \\ \gamma_3(x,y) &= y \qquad\qquad ;\ 0\leqslant y\leqslant 1-x \end{aligned} \tag{2.54}$$

while the elements of $\{\phi_i\}_{1\leqslant i\leqslant 6}$ basis of $\mathbb{P}^2(T_K)$ are defined by

$$\begin{aligned} \phi_1(x,y) &= 1-3x-3y+4xy+4x^2+4y^2 \\ \phi_2(x,y) &= -x+2x^2 \\ \phi_3(x,y) &= -y+2y^2 \\ \phi_4(x,y) &= 4x-4xy-4x^2 \\ \phi_5(x,y) &= 4xy \qquad\qquad ;\ 0\leqslant x\leqslant 1 \\ \phi_6(x,y) &= 4y-4xy-4y^2 \qquad\qquad ;\ 0\leqslant y\leqslant 1-x \end{aligned} \tag{2.55}$$

By representing some of the linear equations on the reference triangle (see Figure 2.10), associated to the basis functions (2.54) which are passing by the nodes of a $\mathbb{P}^2$ element (denoted $\{a_i\}_{1\leqslant i\leqslant 6}$), one can see that the basis functions $\{\phi_i\}_{1\leqslant i\leqslant 6}$ can be defined by linear combinations of $\mathbb{P}^1$ basis functions. Let a_1 be a node of a $\mathbb{P}^2$ element. In order to define ϕ_1 element of the base $\{\phi_i\}_{1\leqslant i\leqslant 6}$ as a basis function ϕ_1 needs to satisfy

$$\begin{aligned} &1.\ \phi_1(a_1) = 1 \\ &2.\ \phi_1(a_i) = 0, \quad 2\leqslant i\leqslant 6 \end{aligned} \tag{2.56}$$

By definition of $\gamma_1(a_1) = 1$ the basis function ϕ_1 can be defined by

$$\phi_1(x,y) = k(x,y)\gamma_1(x,y), \tag{2.57}$$

such that $k(x,y)$ is a real function that needs to be determined. The next step is to determine $k(x,y)$ such that $\phi_1(a_i) = 0,\ \forall i\in[[2,6]]$. If one looks at Figure 2.10 then $\gamma_1(x,y) = 0$ for the nodes a_2, a_5 and a_3. Moreover, $\gamma_1(x,y) = \frac{1}{2}$ for the nodes a_4 and a_6. Then $k(x,y)$ is defined by $k(x,y) = \delta(\gamma_1(x,y) - \frac{1}{2})$, where $\delta\in\mathbb{R}$ is used to satisfy $k(a_1) = 1$. Then $\phi_1(x,y)$ is defined by

$$\boxed{\phi_1(x,y) = 2(\gamma_1(x,y) - \tfrac{1}{2})\gamma_1(x,y)}.$$

The same work is done for every element of the basis $\{\phi_i\}_{1\leqslant i\leqslant N}$. In Figure 2.10 one can see that each $\mathbb{P}^2$ basis function is a combination of $\mathbb{P}^1$ basis functions. Then, the elements of the basis $\{\phi_i\}_{1\leqslant i\leqslant N}$ are of the form

$$\begin{aligned}
\phi_1(x,y) &= 2(\gamma_1(x,y) - \frac{1}{2})\gamma_1(x,y),\\
\phi_2(x,y) &= 2(\gamma_2(x,y) - \frac{1}{2})\gamma_2(x,y),\\
\phi_3(x,y) &= 2(\gamma_3(x,y) - \frac{1}{2})\gamma_3(x,y),\\
\phi_4(x,y) &= 4\gamma_1(x,y)\gamma_2(x,y),\\
\phi_5(x,y) &= 4\gamma_2(x,y)\gamma_3(x,y),\\
\phi_6(x,y) &= 4\gamma_1(x,y)\gamma_3(x,y).
\end{aligned} \tag{2.58}$$

The definition (2.53) has been verified in the case of a $2D$ simulations.

Remark 8. We recall that the basis functions are defined on the reference element before being adapted to any element of the mesh by affine transformation, see Section 1.1.3.

The next step to define the temperature enrichment is to exclude the addition of the edge midpoints, which correspond to the green nodes in Figure 2.10, in order to avoid creating a larger system to solve. The main idea is to replace those values using only the information known for the nodes of a $\mathbb{P}^1$ element, the red nodes in Figure 2.10. To achieve this, a third-order truncated Taylor expansion of the temperature along the edge is considered, along with a second order Taylor expansion of the flux; i.e.,

$$\begin{aligned}
&2.59.a)\ T(x+\delta x) = T(x) + \nabla T(x)\cdot\boldsymbol{\delta x} + 1/2\ \boldsymbol{\delta x}^t\mathcal{H}(T)\boldsymbol{\delta x} + \mathcal{O}(||\boldsymbol{\delta x}||^3)\\
&2.59.b)\ \boldsymbol{\beta}(x+\delta x) = \boldsymbol{\beta}(x) + \nabla.\ \boldsymbol{\beta}(x)\boldsymbol{\delta} x + \mathcal{O}(||\boldsymbol{\delta x}||^2)
\end{aligned} \tag{2.59}$$

Then Conditions (2.59) are modified to use the particularity of the mixed formulation where the expression of the flux is linked to the temperature. In order to replace $\mathcal{H}(T)$ in (2.59.a) the definition $\nabla T = -\frac{1}{\lambda}\beta$ is used (by definition of the problem in (2.8)) which leads to $\mathcal{H}(T) = -\nabla\cdot(\lambda^{-1}\beta)$. Then the Taylor expansions (2.59) is defined by the following approximation

$$\begin{aligned}
&2.60.a)\ T(x+\delta x) \approx T(x) - \lambda^{-1}\beta\cdot\delta x - \frac{1}{2}\delta x^t\nabla(\lambda^{-1}\beta)\delta x,\\
&2.60.b)\ (\lambda^{-1}\beta)(x+\delta x) \approx (\lambda^{-1}\beta)(x) + \nabla(\lambda^{-1}\beta)(x)\delta x\ \ .
\end{aligned} \tag{2.60}$$

Remark 9. The operator $\mathcal{H}(T)$ is **the Hessian of** T, while $\nabla\cdot(-\lambda^{-1}\beta)$ is the **Hessian tensor of** $(-\lambda^{-1}\beta)$ i.e.

$$\mathcal{H}(T) = \begin{pmatrix} \partial_{x^2}T & \partial_x\partial_y T\\ \partial_y\partial_x T & \partial_{y^2}T\end{pmatrix} = \nabla\cdot(-\lambda^{-1}\beta) = -\lambda^{-1}\begin{pmatrix}\partial_x\beta_x & \partial_x\beta_y\\ \partial_y\beta_x & \partial_y\beta_y\end{pmatrix}\ \ . \tag{2.61}$$

Remark 10. In this work, only simulations conducted on triangular meshes using $\mathbb{P}^1$ basis functions are presented. It means that the Hessian associated to the temperature is not defined if T is used as the only variable to be solved, thus equation (2.60) is always preferred.

Let $e_{i,j} = x_j - x_i$ be the edge vector, with $x_i, x_j \in \mathbb{R}^d$, and x_k the edge midpoint of the edge $[x_i, x_j]$, see Figure 2.11. As x_k is the midpoint of the edge $[x_i, x_j]$ one has

$$e_{i,k} = -e_{j,k} = \frac{1}{2}e_{i,j}. \tag{2.62}$$

The Taylor expansion of $\lambda^{-1}\boldsymbol{\beta}(x_j) = \lambda^{-1}\boldsymbol{\beta}_j$ and $\lambda^{-1}\boldsymbol{\beta}(x_i) = \lambda^{-1}\boldsymbol{\beta}_i$ in the direction of the point x_k leads to the following approximation of $\nabla.(\lambda^{-1}\boldsymbol{\beta})(x)\boldsymbol{e}_{i,j}$ in (2.63).

$$\begin{aligned} &1.\ \lambda^{-1}\beta(x_j) = \lambda^{-1}\beta(x_k) + \nabla\cdot(\lambda^{-1}\beta(x_k))\cdot(x_j - x_k) + \mathcal{O}(||x_j - x_k||^2) \\ &2.\ \lambda^{-1}\beta(x_i) = \lambda^{-1}\beta(x_k) + \nabla\cdot(\lambda^{-1}\beta(x_k))\cdot(x_i - x_k) + \mathcal{O}(||x_i - x_k||^2) \\ &\implies \boxed{\nabla\cdot(\lambda^{-1}\beta(x_k))(x_j - x_i) = \nabla\cdot(\lambda^{-1}\beta_k)e_{i,j} \approx \lambda^{-1}\beta_j - \lambda^{-1}\beta_i} \end{aligned} \tag{2.63}$$

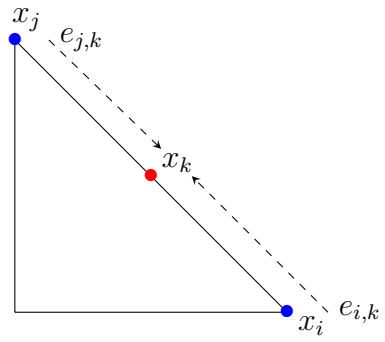


Figure 2.11: Approximation at a point x_k

Using Equation (2.63) and by performing Taylor expansions of T_k in the directions of the nodes x_i and x_j, two approximations of $T_k := T(x_k)$ are obtained.

First in the direction of the node x_i the Equation (2.64) is obtained

$$\begin{aligned} T_k &= T_i - \lambda^{-1}\beta_i\cdot e_{i,k} - \frac{1}{2}\nabla\cdot(\lambda^{-1}\beta_i)e_{i,k} + \mathcal{O}(||x_k - x_i||^3), \\ &\approx T_i - \lambda^{-1}\beta_i\cdot e_{i,k} - \frac{1}{2}(\lambda^{-1}\beta_k - \lambda^{-1}\beta_i)\cdot e_{i,k}\ , \end{aligned} \tag{2.64}$$

while in the direction of the node x_j the Equation (2.65) is defined

$$\begin{aligned} T_k &= T_j - \lambda^{-1}\beta_j\cdot e_{j,k} - \frac{1}{2}\nabla\cdot(\lambda^{-1}\beta_j)e_{j,k} + \mathcal{O}(||x_k - x_j||^3), \\ &\approx T_j - \lambda^{-1}\beta_j\cdot e_{j,k} - \frac{1}{2}(\lambda^{-1}\beta_k - \lambda^{-1}\beta_j)\cdot e_{j,k}. \end{aligned} \tag{2.65}$$

By adding the Formula (2.64) and (2.65) and by taking the average value, an approximation of T_k denoted T_k^h is given by

$$T_k^h = \frac{1}{2}(T_i + T_j - \lambda^{-1}\beta_i\cdot e_{i,k} - \lambda^{-1}\beta_j\cdot e_{j,k} - \frac{1}{2}(\lambda^{-1}\beta_k - \lambda^{-1}\beta_i)\cdot e_{i,k} - \frac{1}{2}(\lambda^{-1}\beta_k - \lambda^{-1}\beta_j)\cdot e_{j,k})\ . \tag{2.66}$$

Using Definition (2.62) an approximation of T_k is obtained independent of the node x_k.

$$\begin{aligned} T_k^h &= 1/2\ (T_i + T_j - \lambda^{-1}\boldsymbol{\beta}_i\cdot\frac{e_{i,j}}{2} + \lambda^{-1}\boldsymbol{\beta}_j\cdot\frac{e_{i,j}}{2} + 1/2(\lambda^{-1}\boldsymbol{\beta}_k - \lambda^{-1}\boldsymbol{\beta}_j)\cdot\frac{e_{i,j}}{2} \\ &\quad - 1/2\ (\lambda^{-1}\boldsymbol{\beta}_k - \lambda^{-1}\boldsymbol{\beta}_i)\cdot\frac{e_{i,j}}{2}) \\ &= 1/2\ (T_i + T_j) + 1/8\ (\lambda^{-1}\boldsymbol{\beta}_j - \lambda^{-1}\boldsymbol{\beta}_i)\cdot e_{i,j}\ . \end{aligned} \tag{2.67}$$

In the following $\Delta(\lambda^{-1}\beta)_e$ denotes $\Delta(\lambda^{-1}\beta)_e = (\lambda^{-1}\boldsymbol{\beta}_j - \lambda^{-1}\boldsymbol{\beta}_i)$ and $\Delta x_e = e_{i,j}$.

The last step consists in plugging Expression (2.67) into (2.52), and by changing the definition

of the $\mathbb{P}^2$ basis functions by their relations with the $\mathbb{P}^1$ defined by Equation (2.53).

$$
\begin{aligned}
T_h(x) &= \sum_{i,node} T_i \phi_i^{\mathbb{P}^2}(x) \; + \; \sum_{k,edge} T_k \phi_k^{\mathbb{P}^2}(x) \\
&\approx \sum_{i,node} T_i \phi_i^{\mathbb{P}^1}(x)(2\phi_i^{\mathbb{P}^1}(x) - 1) \; + \; \sum_{k,edge} (\frac{1}{2}(T_i + T_j) + \frac{1}{8}(\lambda^{-1}\beta_j - \lambda^{-1}\beta_i) \cdot e_{i,j}) 4 \phi_i^{\mathbb{P}^1}(x) \phi_j^{\mathbb{P}^1}(x) \\
&\approx \sum_{i,node} T_i \phi_i^{\mathbb{P}^1}(x)(2\phi_i^{\mathbb{P}^1}(x) - 1) \; + \; \sum_{k,edge} 2(T_i + T_j) \phi_i^{\mathbb{P}^1}(x) \phi_j^{\mathbb{P}^1}(x) + \sum_{k,edge} \frac{1}{2} \Delta(\lambda^{-1}\beta)_e \cdot \Delta x_e \phi_i^{\mathbb{P}^1}(x) \phi_j^{\mathbb{P}^1}(x) \\
&\approx \sum_{i,node} 2 T_i \phi_i^{\mathbb{P}^1}(x)(\sum_{j,node} \phi_j^{\mathbb{P}^1}(x) \; - 1) + \sum_{i,node} T_i \phi_i^{\mathbb{P}^1}(x) + \sum_{k,edge} \frac{1}{2} \Delta(\lambda^{-1}\beta)_e \cdot \Delta x_e \phi_i^{\mathbb{P}^1}(x) \phi_j^{\mathbb{P}^1}(x) \\
&\approx \boxed{\sum_{i,node} T_i \phi_i^{\mathbb{P}^1}(x) + \sum_{k,edge} \tfrac{1}{2} \Delta(\lambda^{-1}\beta)_e \cdot \Delta x_e \phi_i^{\mathbb{P}^1}(x) \phi_j^{\mathbb{P}^1}(x)}
\end{aligned}
\tag{2.68}
$$

since $\sum_{j,node} \phi_j^{\mathbb{P}^1}(x) = 1$. Thus, the enriched temperature is build using only the $\mathbb{P}^1$ basis functions and without having to calculate the midpoint values of each edge. The terminology enrichment comes from its definition (2.68) where only new terms have been added to the usual $\mathbb{P}^1$ approximation.

Remark 11. The enrichment method allows to improve the space accuracy on both variables (T,$\boldsymbol{\beta}$) without increasing the size of the linear system associated to the finite weak formulation. For the example of a 2D-simulation the mixed form requires the treatment of two additional unknowns β_x and β_y compared to its primal form, where only T would be considered. An important step in the FEM is to solve a linear system, where the matrix depends on several parameters such as the number of unknowns, the mesh discretization, and the choice of the space of approximation. This step is time-consuming and one always wants to decrease the size of the matrix.

Let N_v be the number of vertices, associated to a standard triangular mesh discretization $\mathcal{T}_h$, then the size of the final system for a 2D simulation (2 components for $\boldsymbol{\beta}$) is equal to

- $(3(3N_v))^2$, for a standard P^1 simulation solved in mixed form,
- $(3(3N_v))^2$, for an enriched P^1 simulation solved in mixed form,
- $(3(6N_v))^2$, for a standard P^2 simulation solved in mixed form,
- $(6N_v)^2$, for a standard P^2 simulation solved in primal form.

The standard $\mathbb{P}^2$ simulation in primal form requires to solve a bigger system than the enriched $\mathbb{P}^1$ simulation in its mixed form, which gives the same accuracy than a P^2 simulation, as demonstrated in this section. For time-dependent problems, the mixed form imposes fewer constraints on the time step (for explicit schemes), and as previously mentioned in Section 2.1.2, it is more stable than the primal form.

For the Stefan problem, the flux needs to be known at the interface. If the primal form was considered, a flux reconstruction method would still be required. Reconstruction methods, such as the Green-Gauss method, are largely sensitive to numerical quadrature, discontinuities in the primal variable, changes in flow direction and the mesh quality. For example, the accuracy at the boundary can be impacted by the shape of the elements, especially for complex boundary shapes and can lead to singularities in the gradient. For this reason, the use of the mixed form provides an interesting alternative to handle two-phases situations.

In the following, the usual SBM is extended to increase the order of accuracy (see [95]), using Definition (2.68). Results for both set of interface conditions (2.9) and (2.7) are presented. To

this end, the spaces of resolution associated to the enrichment procedure defined in (2.68) are first introduced.

2.4.2 Enrichment of the primal variable

Considering the mutual boundary $\tilde{\Gamma}(t)$ between $\tilde{\Omega}_+$ and $\tilde{\Omega}_-$ and for the sake of clarity the following spaces are once again introduced

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

which with Definition (2.68) allows to introduce the finite-dimensional spaces (2.69) associated with an enriched temperature version.

$$\begin{aligned}\mathcal{Q}_T^{2*}(\Omega) = \{T \in \mathcal{Q}_T(\Omega)\,|\,T|_K = &\sum_{i,node} T_i \phi_i^{\mathbb{P}^1(K)}(x) + \sum_{k,edge} \frac{1}{2}\Delta(\lambda^{-1}\beta)_e \Delta x_e \phi_i^{\mathbb{P}^1(K)}(x)\phi_j^{\mathbb{P}^1(K)}(x),\\ &\text{such that } \{\phi_i^{\mathbb{P}^1(K)}\}_{1\leqslant i\leqslant 3} \text{ basis of } \mathbb{P}^1(K),\ \forall K \in \mathcal{T}_h\} \cap (C^0(\tilde{\Omega}^+) \cup C^0(\tilde{\Omega}^-))\ ,\end{aligned}$$

$$\mathcal{Q}_\beta^1(\Omega) = \{\boldsymbol{\beta} \in \mathcal{Q}_\beta(\Omega)\,|\,\boldsymbol{\beta}|_K \in (\mathbb{P}^1(K))^d,\ \forall K \in \mathcal{T}_h\} \cap ((C^0(\tilde{\Omega}^+))^d \cup (C^0(\tilde{\Omega}^-))^d)\ . \tag{2.69}$$

Finally the following spaces are considered

$$\begin{aligned}\boldsymbol{Q}^{2*,1}(]0,t_f[,\Omega) &= \mathbb{L}^2(]0,t_f[,\mathcal{Q}_T^{2*}(\Omega)) \times \mathbb{L}^2(]0,t_f[,\mathcal{Q}_B^1(\Omega))\ ,\\ \boldsymbol{Q}^{2*,1}(\Omega) &= \mathcal{Q}_T^{2*}(\Omega) \times \mathcal{Q}_B^1(\Omega)\ .\end{aligned} \tag{2.70}$$

The method to transform an embedded scheme into an enriched embedded scheme is as follow.

1. The definition (2.68) defined for the temperature field T is also used for its test function q to define more symmetry in the definition of the finite weak formulation.
2. Higher order Taylor approximations for the impositions of the interface conditions on the surrogate $\tilde{\Gamma}$ are necessary to keep the second order accuracy obtained by the enrichment (2.68). This means that, at the very least, a second-order expansion for the temperature field and its test function are employed to maintain symmetry, (see Equation (2.64) and Equation (2.65)).
3. To differentiate an enriched variable from the other variables in the scheme, the notation $*$ is used to denote an enriched variable. For example T^* refers to the enriched temperature field and q^* refers to the associated enriched test function.

It is also possible to look at a non-symmetric enrichment by taking a non enriched test function. However, a symmetric formulation is also a form of stabilization by bringing symmetry to the matrix of the linear system. It is then advised to use both the enrichment on the temperature T^* and on its test function q^*. In the next section, the enriched scheme for both set of interface conditions (2.9) and (2.7) is introduced.

2.4.3 First set of interface conditions

The enriched version consists in replacing T by T^* and q by q^* in (2.46), where T^* and q^* are given by Definition (2.68).

Finding $(T^*, \boldsymbol{\beta}) \in \boldsymbol{Q}^{2*,1}(\,]0, t_f[\,, \Omega)$, such that $\forall (q^*, \boldsymbol{w}) \in \boldsymbol{Q}^{2*,1}(\Omega)$ one has

$$
\begin{aligned}
&2.71.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T^*, \nabla \cdot \boldsymbol{w})_\Omega + < [\![\lambda^{-1}\boldsymbol{\beta} \cdot \boldsymbol{d} + 1/2 \boldsymbol{d}^t \nabla(\lambda^{-1}\boldsymbol{\beta})\boldsymbol{d}]\!]_{\tilde{\Gamma}}\ , \{\boldsymbol{w}\} \cdot \tilde{\boldsymbol{n}}\ >_{\tilde{\Gamma}} \\
&+ < \{T^*\}, [\![\boldsymbol{w}]\!]_{\tilde{\Gamma}} \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} + B_{stab}((T^*, \boldsymbol{\beta}), \boldsymbol{w}) + < T^*, \boldsymbol{w} \cdot n >_{\Gamma_N} = \ - < j_T, \{\boldsymbol{w}\} \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} \\
&- < T_D, \boldsymbol{w} \cdot \boldsymbol{n} >_{\Gamma_D} + L_{stab}(\boldsymbol{w}); \\
&2.71.b)\ (\rho c \partial_t T^*, q)_\Omega - < [\![\boldsymbol{\beta} + \nabla \cdot \boldsymbol{\beta}\ \boldsymbol{d}]\!]_{\tilde{\Gamma}} \cdot \boldsymbol{n}(\boldsymbol{n} \cdot \tilde{\boldsymbol{n}}), \{q^*\} >_{\tilde{\Gamma}} + (\nabla \cdot \boldsymbol{\beta}, q^*)_\Omega\ + A_{stab}((T^*, \boldsymbol{\beta}), q^*) \\
&- < \boldsymbol{\beta} \cdot \boldsymbol{n} - h_N, q^* >_{\Gamma_N} = \ (f, q^*)_\Omega - < \sigma(\boldsymbol{n} \cdot \tilde{\boldsymbol{n}}), \{q^*\} >_{\tilde{\Gamma}}\ .
\end{aligned}
\tag{2.71}
$$

Accordingly to formulation (2.71) the Nitsche penalty terms defined in (2.45) need to be modified to match the Taylor developments of the enrichment method, see Section 2.4.2. The Taylor developments are done at a higher order and are modified on both the temperature and its test function:

$$
\begin{aligned}
\tilde{N}_{A_1}(T^*, q^*) &:= \frac{\alpha}{h} < [\![T^* + \nabla T^* \cdot \boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t \mathcal{H}(T^*)\boldsymbol{d}]\!]_{\tilde{\Gamma}}, [\![q^* + \nabla q^* \cdot \boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t \mathcal{H}(q^*)\boldsymbol{d}]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}} \\
\tilde{N}_{L_1}(q^*) &:= \frac{\alpha}{h} < j_T, [\![q^* + \nabla q^* \cdot \boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t \mathcal{H}(q^*)\boldsymbol{d}]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}}
\end{aligned}
\tag{2.72}
$$

Using the definition of the flux $\beta = -\lambda \nabla T$, the Nitsche's penalty terms (2.72) can be rewritten as follow

$$
\begin{aligned}
\tilde{N}_{A_1}(T^*, q^*) &:= \frac{\alpha}{h} < [\![T^* - \lambda^{-1}\beta \cdot \boldsymbol{d} - \frac{1}{2}\boldsymbol{d}^t(\lambda^{-1}\nabla\boldsymbol{\beta})\boldsymbol{d}]\!]_{\tilde{\Gamma}}, [\![q^* - \lambda^{-1}\boldsymbol{w} \cdot \boldsymbol{d} - \frac{1}{2}\boldsymbol{d}^t(\lambda^{-1}\nabla\boldsymbol{w})\boldsymbol{d}]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}} \\
\tilde{N}_{L_1}(q^*) &:= \frac{\alpha}{h} < j_T, [\![q^* - \lambda^{-1}\boldsymbol{w} \cdot \boldsymbol{d} - \frac{1}{2}\boldsymbol{d}^t(-\lambda^{-1}\nabla\boldsymbol{w}))\boldsymbol{d}]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}}\ .
\end{aligned}
\tag{2.73}
$$

The Nitsche penalty terms (2.73) are then added to formulation (2.71) to obtain the scheme associated with an enriched embedded simulation, corresponding to the first set of interface condition (2.9). The same procedure is done for the second set of interface conditions (2.7) in the next section.

2.4.4 Second set of interface conditions

In this section, the method explained in Section 2.4.2 is applied to the second set of interface conditions (2.7). Then, from formulation (2.74) one is able to obtain the following enriched scheme.

Finding $(T^*, \boldsymbol{\beta}) \in \boldsymbol{Q}^{2*,1}(\,]0, t_f[\,, \Omega)$, such that $\forall (q^*, \boldsymbol{w}) \in \boldsymbol{Q}^{2*,1}(\Omega)$ one has

$$
\begin{aligned}
&2.74.a)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T^*, \nabla \cdot \boldsymbol{w})_\Omega + < T_D, \boldsymbol{w} \cdot \boldsymbol{n} >_{\Gamma_D} + < T^*, \boldsymbol{w} \cdot \boldsymbol{n} >_{\Gamma_N} + B_{stab}((T^*, \boldsymbol{\beta}), \boldsymbol{w}) \\
&- < \{\nabla T^* \cdot \boldsymbol{d} + 1/2 \boldsymbol{d}^t \mathcal{H}(T^*)\boldsymbol{d}\}_{\tilde{\Gamma}}, [\![\boldsymbol{w}]\!]_{\tilde{\Gamma}} \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} - < [\![\nabla T \cdot \boldsymbol{d} + 1/2 \boldsymbol{d}^t \mathcal{H}(T^*)\boldsymbol{d}]\!]_{\tilde{\Gamma}}, \{\boldsymbol{w}\}_{\tilde{\Gamma}} \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} \\
&= - < a_T, [\![\boldsymbol{w}]\!]_{\tilde{\Gamma}} \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} - < j_T, \{\boldsymbol{w}\}_{\tilde{\Gamma}} \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} + L_{stab}(\boldsymbol{w}), \\
&2.74.b)\ (\rho c \partial_t T^*, q^*)_\Omega + (\nabla \cdot \boldsymbol{\beta}, q)_\Omega - < \boldsymbol{\beta} \cdot \boldsymbol{n} - h_N, q >_{\Gamma_N} + A_{stab}((T^*, \boldsymbol{\beta}), q^*) = (f, q^*)_\Omega.
\end{aligned}
\tag{2.74}
$$

To the enriched weak formulation (2.74) the Nitsche penalty terms (2.75) are also added in the scheme :

$$\begin{aligned}
&\tilde{N}_{A_2}(T^*,q^*) := \frac{\alpha}{h} < [\![T^* + \nabla T^* \cdot \boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t\mathcal{H}(T^*)\boldsymbol{d}]\!]_{\tilde{\Gamma}}, [\![q^* + \nabla q^* \cdot \boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t\mathcal{H}(q^*)\boldsymbol{d}]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}} \\
&+ \frac{\alpha}{h} < \{T^* + \nabla T^* \cdot \boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t\mathcal{H}(T^*)\boldsymbol{d}\}_{\tilde{\Gamma}}, \{q^* + \nabla q^* \cdot \boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t\mathcal{H}(q^*)\boldsymbol{d}\}_{\tilde{\Gamma}} >_{\tilde{\Gamma}}, \\
&\tilde{N}_{L_2}(q^*) := \frac{\alpha}{h} < j_T, [\![q^* + \nabla q^* \cdot \boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t\mathcal{H}(q^*)\boldsymbol{d}]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}} + \frac{\alpha}{h} < a_T, \{q^* + \nabla q^* \cdot \boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t\mathcal{H}(q^*)\boldsymbol{d}\}_{\tilde{\Gamma}} >_{\tilde{\Gamma}} .
\end{aligned} \tag{2.75}$$

Following Remark 9, the Nitsche penalty terms (2.75) and the weak formulation (2.74) can be rewritten using the information known from the mixed form. To reduce the computational cost ∇T^* is replaced by $-\lambda^{-1}\boldsymbol{\beta}$ and $\mathcal{H}(T^*)$ by $-\nabla\cdot(\lambda^{-1}\boldsymbol{\beta})$ in Definition (2.75). The same work is done on the test function q^*. This allows to reduce the number of additional terms which leads to the following finite weak formulation

Finding $(T^*,\boldsymbol{\beta}) \in \boldsymbol{Q}^{2*,1}(\,]0,t_f[\,,\Omega)$, such that $\forall (q^*,\boldsymbol{w}) \in \boldsymbol{Q}^{2*,1}(\Omega)$ the following equations hold

$$\begin{aligned}
&2.74.a)\ (\lambda^{-1}\boldsymbol{\beta},\boldsymbol{w})_\Omega - (T^*,\nabla\cdot\boldsymbol{w})_\Omega + < T_D, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_D} + < T^*, \boldsymbol{w}\cdot\boldsymbol{n} >_{\Gamma_N} + B_{stab}((T^*,\boldsymbol{\beta}),\boldsymbol{w}) \\
&+ < \{\lambda^{-1}\boldsymbol{\beta}\cdot\boldsymbol{d} + 1/2\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{\beta})\boldsymbol{d}\}_{\tilde{\Gamma}}, [\![\boldsymbol{w}]\!]_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} + < [\![\lambda^{-1}\boldsymbol{\beta}\cdot\boldsymbol{d} + 1/2\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{\beta})\boldsymbol{d}]\!]_{\tilde{\Gamma}}, \{\boldsymbol{w}\}_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} \\
&= - < a_T, [\![\boldsymbol{w}]\!]_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} - < j_T, \{\boldsymbol{w}\}_{\tilde{\Gamma}}\cdot\tilde{\boldsymbol{n}} >_{\tilde{\Gamma}} + L_{stab}(\boldsymbol{w}), \\
&2.74.b)\ (\rho c\partial_t T^*, q^*)_\Omega + (\nabla\cdot\boldsymbol{\beta}, q)_\Omega - < \boldsymbol{\beta}\cdot\boldsymbol{n} - h_N, q >_{\Gamma_N} + A_{stab}((T^*,\boldsymbol{\beta}),q^*) = (f,q^*)_\Omega,
\end{aligned} \tag{2.76}$$

and where the Nitsche's penalty terms (4.123) are added to Formulation (2.76).

$$\begin{aligned}
&\tilde{N}_{A_2}(T^*,q^*) := \frac{\alpha}{h} < [\![T^* - \lambda^{-1}\boldsymbol{\beta}\cdot\boldsymbol{d} - 1/2\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{\beta})\boldsymbol{d}]\!]_{\tilde{\Gamma}}, [\![q^* - \lambda^{-1}\boldsymbol{w}\cdot\boldsymbol{d} - 1/2\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{w})\boldsymbol{d}]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}} \\
&+ \frac{\alpha}{h} < \{T^* - \lambda^{-1}\boldsymbol{\beta}\cdot\boldsymbol{d} - 1/2\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{\beta})\boldsymbol{d}\}_{\tilde{\Gamma}}, \{q^* - \lambda^{-1}\boldsymbol{w}\cdot\boldsymbol{d} - \frac{1}{2}\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{w})\boldsymbol{d}\}_{\tilde{\Gamma}} >_{\tilde{\Gamma}} \\
&\tilde{N}_{L_2}(q^*) := \frac{\alpha}{h} < j_T, [\![q^* - \lambda^{-1}\boldsymbol{w}\cdot\boldsymbol{d} - 1/2\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{w})\boldsymbol{d}]\!]_{\tilde{\Gamma}} >_{\tilde{\Gamma}} \\
&+ \frac{\alpha}{h} < a_T, \{q^* - \lambda^{-1}\boldsymbol{w}\cdot\boldsymbol{d} - \frac{1}{2}\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{w})\boldsymbol{d}\}_{\tilde{\Gamma}} >_{\tilde{\Gamma}} .
\end{aligned} \tag{2.77}$$

In the next section, the treatment applied to the moving interface is introduced. The section will detail the procedures used to maintain the spatial second-order accuracy brought by the enrichment of the temperature field as well as the necessary reconstruction procedures to facilitate the expansion of the liquid phase in the melting configuration.

2.5 Moving the interface

In this Section, the focus is put on the challenges brought by the moving interface and the use simultaneously of the **SBM**. Dealing with these two particular aspects is a delicate part of the problem. Since the interface is moving, some mesh nodes can switch from one part of the domain to the other one, separated by the surrogate interface $\tilde{\Gamma}$. This problem is particularly emphasized by the discontinuous aspect of the variables at the surrogate interface.

Some techniques described here are related to the particular study of the Stefan problem explained in this Chapter 2 (Stefan Condition), while some methods can be applicable to any type of systems (reconstruction procedures). We will look more precisely at techniques used to get at least a second order accuracy in time and space for all variables of the model. The

method will be expressed with examples for $2D$ simulations but the principles presented here can be generalized to $3D$ simulations as well.

Three different aspects will be studied

1. Moving the surrogate interface,
2. Dealing with time dependent terms,
3. Solving the Stefan condition for the interface motion.

In the proceeding section, a time interval $[0, t_f]$ is considered. Let N_t be the number of points of the time discretization. For any $1 \leqslant k \leqslant N_t$, we set $\Delta t^k = t^{k+1} - t^k$, a non constant time step such that $\Delta t^0 < \Delta t^1 < \Delta t^2 < ... < \Delta t^{N_t}$. For any generic quantity f, the notation f^k denotes the approximation of f at time t^k.

2.5.1 Variables initialization

As illustrated in Figure 2.12, each node located on the surrogate interface is duplicated. This construction allows to implement more easily the jump conditions at the interface. This enables to define a variable independently on each phase for nodes located on the surrogate interface.

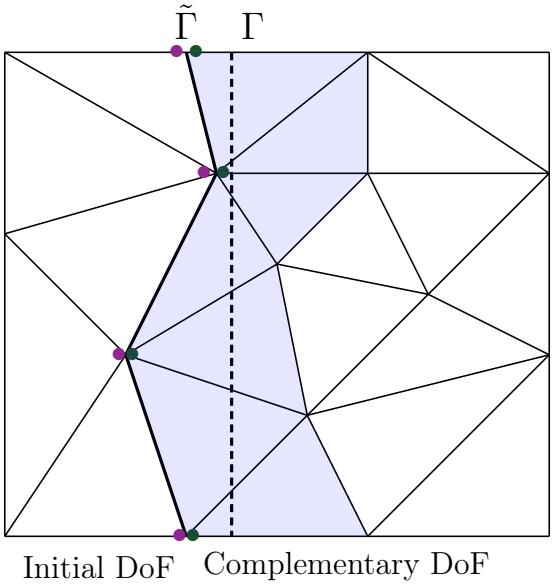


Figure 2.12: Interface structure

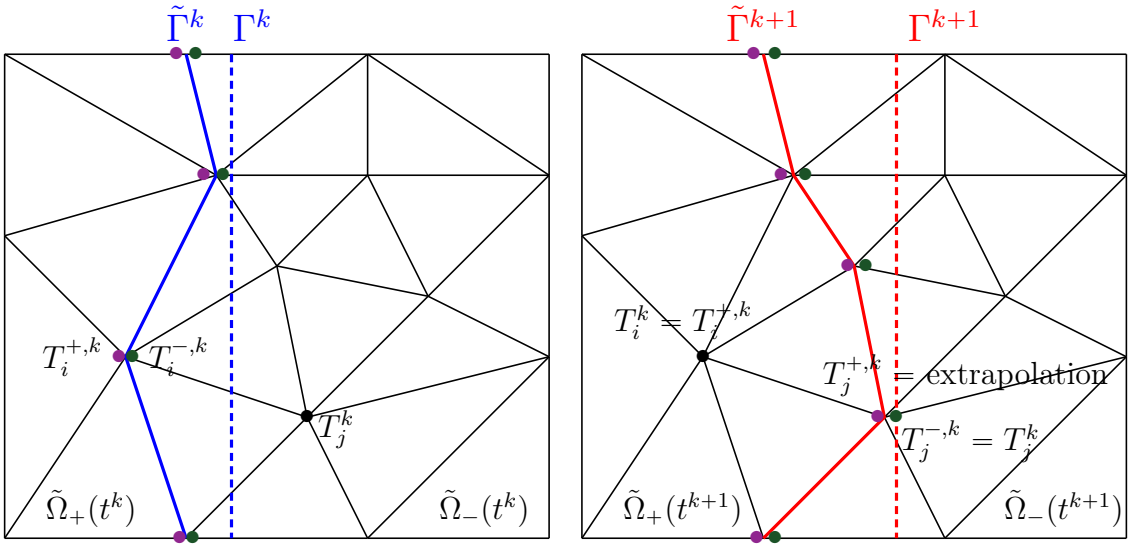
The left DoF (Degree Of Freedom) in Figure 2.12 is associated with the initial mesh numbering and is called **Initial DoF** (purple nodes in Figure 2.12), while the right DoF is called **Complementary DoF** (green nodes in Figure 2.12). If from the original mesh generation (only the vertices are taking into account as $\mathbb{P}^1$ simulations are the ones being considered) there are N_v nodes, then to prevent the modification of the initial numbering, the Complementary DoF gets a reference number starting from the number $N_v + 1$. Computationally it prevents the reallocation of the matrix and vectors structures by initializing them with a maximum size of complementary nodes. Since the interface is moving, the duplication of the DoF has to be performed at each time step when the surrogate changes. Indeed, if no new elements are crossed by the physical interface, then the definition of the surrogate remains the same. If the surrogate interface $\tilde{\Gamma}$ moves alongside the physical interface, then for a same node, its numbering

can change from one time iteration to the next. When the surrogate interface $\tilde{\Gamma}$ moves, two configurations can appear.

1. A node located on the surrogate interface at the previous iteration can now be in one phase only $\tilde{\Omega}_+$ or $\tilde{\Omega}_-$.

2. A node located in one phase only ($\tilde{\Omega}_+$ or $\tilde{\Omega}_-$) at the previous iteration can now be at the phase-change area on $\tilde{\Gamma}$.

More precisely, some points of $\tilde{\Gamma}^k$ at time t^k do not belong to $\tilde{\Gamma}^{k+1}$ and lies in $\tilde{\Omega}_+^k$ or $\tilde{\Omega}_-^k$, while conversely some points of $\tilde{\Omega}_+^k$ or $\tilde{\Omega}_-^k$ can move to $\tilde{\Gamma}^{k+1}$. The points leaving the surrogate at time k are treated easily. Indeed, the values of $(T^{+,k}, \boldsymbol{\beta}^{+,k})$ and $(T^{-,k}, \boldsymbol{\beta}^{-,k})$ on both side of the interface are known, see Figure 2.12. Then if the node now belongs to $\tilde{\Omega}_+^k$, $(T^k, \boldsymbol{\beta}^k) = (T^{+,k}, \boldsymbol{\beta}^{+,k})$ while if it belongs to $\tilde{\Omega}_-^k$, $(T^k, \boldsymbol{\beta}^k) = (T^{-,k}, \boldsymbol{\beta}^{-,k})$.



(a) Interface at time t^k

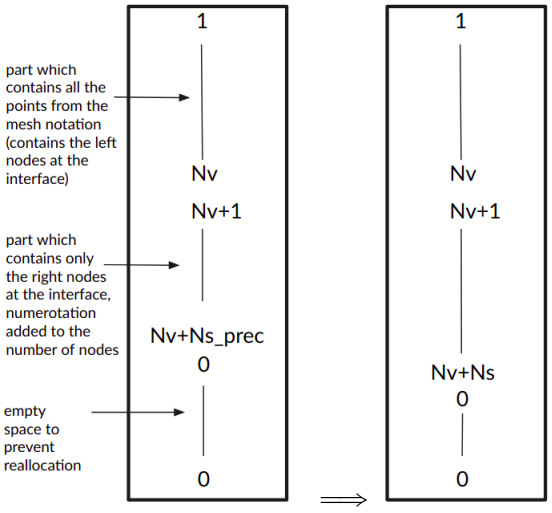
(b) Interface at time t^{k+1}

Figure 2.13: Adaptation of the solution vector at time t^k for the new surrogate interface structure at time t^{k+1}

For example, in Figure 2.13.b, the interface is moving to the right, and the left value is conserved. The treatment of the entering nodes in the surrogate $\tilde{\Gamma}^{k+1}$ is more delicate, since it requires a duplicated DoF, which was not present at the previous iteration and now needs to be initialized. If the node belongs to $\tilde{\Omega}_+$, then we set $(T^{+,k}, \boldsymbol{\beta}^{+,k}) = (T^k, \boldsymbol{\beta}^k)$ whereas if the node belongs to $\tilde{\Omega}_-$, then $(T^{-,k}, \boldsymbol{\beta}^{-,k}) = (T^k, \boldsymbol{\beta}^k)$. For example, in Figure 2.13.b the right value, i.e., the green DoF, is initialized. However, the other side of the interface needs to be defined for time t^k, and the value of $T_j^{+,k}$ is missing as in Figure 2.13. The missing value comes from the expansion of the liquid phase $\tilde{\Omega}_+$ between time t^k and time t^{k+1}. Missing values are reconstructed using polynomial extrapolation. This part must be done carefully in order to preserve the accuracy of the approximation and the improvement brought by the enrichment definition, see next Section with Figure 2.14.

Let us summarize by visualizing the structure of the vector of the right-hand side of the linear system. Let N_s be the number of nodes at the interface at time $k+1$ and N_s_prec the number of nodes at the interface at time k.

For a **displacement from the liquid phase to the solid phase** we have the following steps:



- If a **point is not on the surrogate interface anymore** only the value in the liquid phase is conserved,
- If a **new point is detected** on the surrogate interface, the solid value is known and the liquid value is reconstructed. The liquid value is a part of the mesh construction $\{1, N_v\}$, while the solid value, which was previously not on the surrogate is stored in area $\{N_v + 1, N_v + N_s\}$,
- **Modification of the numbering :** The points still present at the surrogate interface are not always associated with the same numbering and need to be re-localized.

For the missing values the temperature is being approximated **quadratically** and the flux **linearly**, i.e., a **second order polynomial extrapolation for** T and a **first order polynomial extrapolation for** $\boldsymbol{\beta}$ are employed. A quadratic approximation for T is chosen to keep the accuracy brought by the temperature enrichment defined in Section 2.4.

$$A_{2D} = \begin{pmatrix} 1 & x_1 & y_1 & x_1y_1 & x_1^2 & y_1^2 \\ 1 & x_2 & y_2 & x_2y_2 & x_2^2 & y_2^2 \\ 1 & x_3 & y_3 & x_3y_3 & x_3^2 & y_3^2 \\ 1 & x_4 & y_4 & x_4y_4 & x_4^2 & y_4^2 \\ 1 & x_5 & y_5 & x_5y_5 & x_5^2 & y_5^2 \\ 1 & x_6 & y_6 & x_6y_6 & x_6^2 & y_6^2 \end{pmatrix} \quad ; \quad A_{3D} = \begin{pmatrix} 1 & x_1 & y_1 & z_1 & x_1y_1 & x_1z_1 & z_1y_1 & x_1^2 & y_1^2 & z_1^2 \\ 1 & x_2 & y_2 & z_2 & x_2y_2 & x_2z_2 & z_2y_2 & x_2^2 & y_2^2 & z_2^2 \\ 1 & x_3 & y_3 & z_3 & x_3y_3 & x_3z_3 & z_3y_3 & x_3^2 & y_3^2 & z_3^2 \\ 1 & x_4 & y_4 & z_4 & x_4y_4 & x_4z_4 & z_4y_4 & x_4^2 & y_4^2 & z_4^2 \\ 1 & x_5 & y_5 & z_5 & x_5y_5 & x_5z_5 & z_5y_5 & x_5^2 & y_5^2 & z_5^2 \\ 1 & x_6 & y_6 & z_6 & x_6y_6 & x_6z_6 & z_6y_6 & x_6^2 & y_6^2 & z_6^2 \end{pmatrix} \tag{2.78}$$

Let $\boldsymbol{A}$ and $\boldsymbol{b}$ be respectively the matrix and right-hand-side vector of the least square reconstruction, where A is defined in 2D by A_{2D} in (2.78) and in $3D$ by A_{3D} in (2.78). Two strategies are investigated to solve the problem.

1. The first strategy is to solve the least square problem which requires to solve the system $A^tAx = A^tb$ (normal equations). The matrix A^tA has more regularity than its counterpart A and is easier to solve. We can see with the Definition of A in (2.78) that if the coordinates of the stencil are too similar, some of the columns would be a linear combination of each other. For example in $2D$ solving the problem by normal equation means

that for a stencil $\{(x_i, y_i)\}_{i\leqslant s}$, the system studied consists in solving

$$\begin{pmatrix} \sum_{i=1}^{s} 1 & \sum_{i=1}^{s} x_i & \sum_{i=1}^{s} y_i & \sum_{i=1}^{s} x_i y_i & \sum_{i=1}^{s} x_i^2 & \sum_{i=1}^{s} y_i^2 \\ \sum_{i=1}^{s} x_i & \sum_{i=1}^{s} x_i^2 & \sum_{i=1}^{s} x_i y_i & \sum_{i=1}^{s} x_i^2 y_i & \sum_{i=1}^{s} x_i^3 & \sum_{i=1}^{s} x_i y_i^2 \\ \sum_{i=1}^{s} y_i & \sum_{i=1}^{s} x_i y_i & \sum_{i=1}^{s} y_i^2 & \sum_{i=1}^{s} x_i y_i^2 & \sum_{i=1}^{s} x_i^2 y_i & \sum_{i=1}^{s} y_i^3 \\ \sum_{i=1}^{s} x_i y_i & \sum_{i=1}^{s} x_i^2 y_i & \sum_{i=1}^{s} x_i y_i^2 & \sum_{i=1}^{s} x_i^2 y_i^2 & \sum_{i=1}^{s} x_i^3 y_i & \sum_{i=1}^{s} x_i y_i^3 \\ \sum_{i=1}^{s} x_i^2 & \sum_{i=1}^{s} x_i^3 & \sum_{i=1}^{s} x_i^2 y_i & \sum_{i=1}^{s} x_i^3 y_i & \sum_{i=1}^{s} x_i^4 & \sum_{i=1}^{s} x_i^2 y_i^2 \\ \sum_{i=1}^{s} y_i^2 & \sum_{i=1}^{s} x_i y_i^2 & \sum_{i=1}^{s} y_i^3 & \sum_{i=1}^{s} x_i y_i^3 & \sum_{i=1}^{s} x_i^2 y_i^2 & \sum_{i=1}^{s} y_i^4 \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{s} b_i \\ \sum_{i=1}^{s} b_i x_i \\ \sum_{i=1}^{s} b_i y_i \\ \sum_{i=1}^{s} b_i x_i y_i \\ \sum_{i=1}^{s} b_i x_i^2 \\ \sum_{i=1}^{s} b_i y_i^2 \end{pmatrix} \tag{2.79}$$

where a_1, a_2, a_3, a_4, a_5 and a_6 are the coefficient of the polynomial reconstruction ($\mathbb{P}^2$ reconstruction).

2. The second strategy is to use the QR decomposition of A. It means that minimizing $||Ax - b||_2^2$ is equivalent to the minimization of $||Rx - Q^t b||_2^2$, where no inversion or convergence method is needed by definition of R, which is an upper triangular matrix.

The test cases in Section 3.3 of Chapter 3, will demonstrate that the QR decomposition, using the Householder transformations, is more stable and accurate. Comparing two methods for the node reconstruction is essential to show how sensitive the method is to the choice made to initialize values at the surrogate interface.

2.5.2 QR factorization and Householder transformations

In this Section, the QR factorization of the matrix A defined in (2.78) is presented. The QR factorization is a tool used for a variety of applications and one of them is to solve linear systems, using the principle of **back substitution** (the initial linear system is replaced by an equivalent system). One of the most reliable method to compute the matrices Q and R is through the Householder matrices. The matrix A of the original system is multiplied by orthogonal matrices until A becomes a superior triangular matrix (matrix R of the QR factorization). The orthogonal matrices used in the process are called the Householder matrices which are often used in matrix reduction method. In the proposed algorithm, the Householder matrices are defined through the columns of the matrix A.

Let v be a non null vector of $\mathbb{R}^d$, then the Householder matrix of v, $H(v)$ is defined as follows

$$H(v) = I - 2\frac{vv^T}{||v||_2^2} \ , \tag{2.80}$$

where $H(v)_{i,j} = \delta_{i,j} - 2\frac{v_i v_j}{\sum_{k=1}^{d} |v_k|^2}$ and $H(0) = I$.

Algorithm of the QR factorization

Let $A \in \mathcal{M}_{n,p}(\mathbb{R})$, such that $n \geqslant p$, and let the matrices H^k be defined by $H^k \in \mathcal{M}_n(\mathbb{R})$ (see Definition (2.82)) and $A^{k+1} \in \mathcal{M}_{n,p}(\mathbb{R})$ such that $1 \leqslant k \leqslant p$, where $A^1 = A$, $A^{k+1} = H^k A^k$ and $A^{p+1} = R$ the matrix of the QR factorization.

The QR factorization algorithm is as follows:

- Step 1 : Initialization

  Let us define $A^1 = A$ and a^1 the vector defined through the first column of A^1. Then, if a^1 is of the form

$$\begin{pmatrix} a^1_{1,1} \\ 0 \\ 0 \\ 0 \\ . \\ . \\ 0 \end{pmatrix}$$

  no other action is required, meaning than $H^1 = I$. In the other case $A^2 = H^1 A^1$, with $H^1 = H(a^1 - ||a^1||e_1)$, and where e_1 is the first vector of the canonical basis of $\mathbb{R}^n$.

- Step k+1 :

  The $(k-1)$ first columns of A^k have now 0 below their diagonal. Let a^k be the vector of size $(n+1-k)$, composed of the coefficients of the column k of A^k such as $a^k = (a^k_{k,k}, a^k_{k+1,k}, ..., a^k_{n,k})$. Note that they are the last $(n+1-k)$ components of the column k of A^k. If a^k is of the form

$$\begin{pmatrix} a^k_{k,k} \\ 0 \\ 0 \\ . \\ . \\ 0 \end{pmatrix} \tag{2.81}$$

  then $H^k = I$. In the other case A^{k+1} becomes $A^{k+1} = H^k A^k$ such that

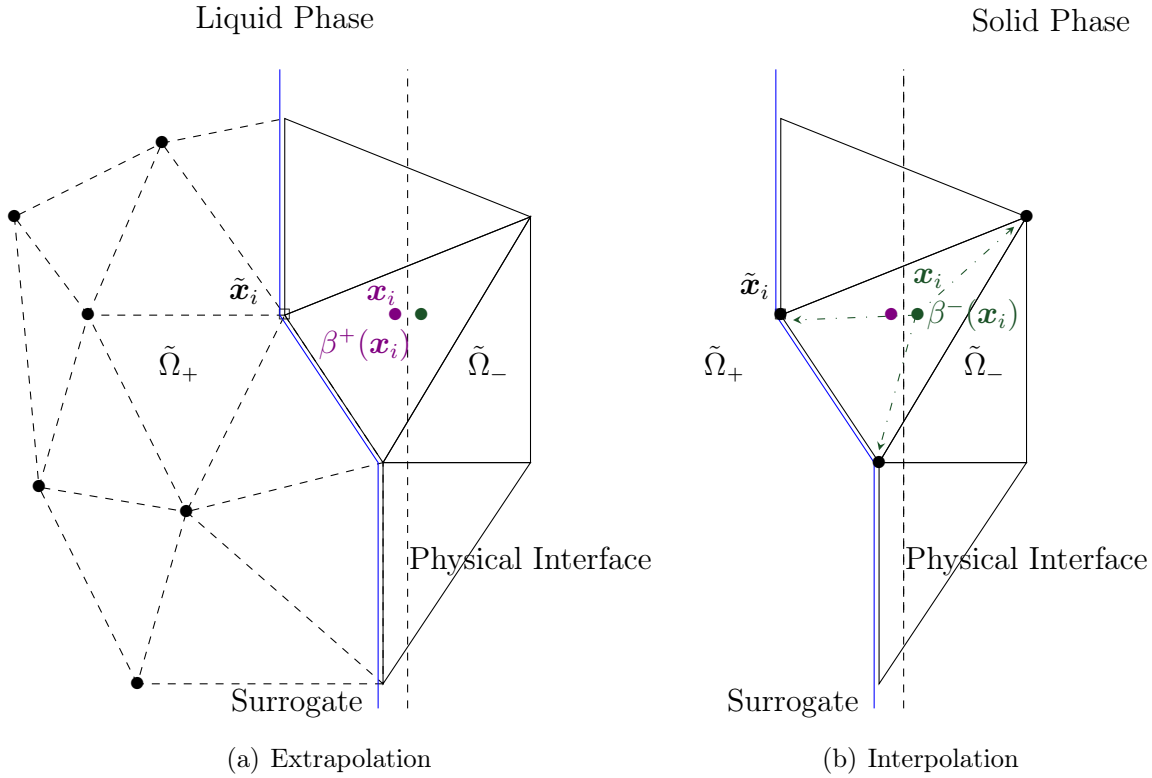
$$H^k = \begin{pmatrix} I_{k-1} & 0 \\ 0 & H(a^k - ||a^k||e_1) \ , \end{pmatrix} \tag{2.82}$$

  where this time e_1 is the first vector of the canonical basis of $\mathbb{R}^{n+1-k}$

- Step $p+1$: $A = QR$ such that $R = A^{p+1} = H^p A^p$ and $Q = H1 * H2 * ... * Hp$

An important notion of the QR factorization is that the decomposition is unique and numerically stable, meaning that small perturbations in the original matrix will not impact the behavior of the factorization. This holds a crucial significance within numerical analysis, where small errors have the potential to accumulate and give rise to substantial inaccuracies in the final solution.

2.5.3 Approximation of the flux jump



(a) Extrapolation (b) Interpolation

Figure 2.14: Visualization of the jump reconstruction method

Since the interface is moving, the duplication of the DoF has to be performed each time the surrogate interface moves. 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 this configuration, one can compute this jump on the surrogate interface. As a consequence, the flux jump needs to be recovered on the physical interface.

For the numerical tests presented in this thesis, the following method is used:

For each node $\tilde{\boldsymbol{x}}_i$ on the surrogate interface, liquid and solid values of each variables are known. Using the mapping M, to every node $\tilde{\boldsymbol{x}}_i$ a corresponding node $\boldsymbol{x}_i$ is associated on the physical interface Γ. Then any value on the physical interface at a 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 2.14(b)).

In order to obtain an approximated value of $\boldsymbol{\beta}^-(x_i)$ (value in the solid phase) a linear interpolation is performed using the values in $\tilde{\Omega}_-$ in the element in which the node x_i is located. For $\boldsymbol{\beta}^+(\boldsymbol{x}_i)$, the same strategy as the one used for variable initialization is employed, see Section 2.5.1. An extrapolation using the solution in Ω_+, at specific points of a stencil (black dots in figure 2.14(a)), allows to recover the value of $\boldsymbol{\beta}^+(\boldsymbol{x}_i)$. Since the flux is represented by a piecewise linear polynomial, a linear interpolation/extrapolation is sufficient. Then, we are able to define, for each node and for both side of the surrogate, a corresponding value of the flux on the physical interface.

In the next section, the choices made on the discretization of the time dependent terms will be introduced for the formulation proposed in Section 2.4. The discretization of the Stefan condition will also be discussed.

2.6 Time discretization

For the time dependent terms present in our weak formulation a BDF2 scheme is proposed. For the condition expressing the interface motion (Stefan condition) a second order scheme has been tested, referred to as an extrapolated BDF2 scheme (exBDF2) .

2.6.1 Second order time scheme for the interface motion

In this section, we discuss the resolution of the Stefan condition (2.10). This part has to be handled carefully if one wants to avoid extra computational costs. Indeed, even though (2.10) is an ordinary differential equation (ODE), the right hand side depends on the solution of System (2.8). Note that an implicit scheme requires to know $[\![\boldsymbol{\beta}^{k+1}]\!]_{\tilde{\Gamma}^{k+1}}$ and the surrogate interface position $\tilde{\Gamma}^{k+1}$ at time t^{k+1} and thus necessitating the imposition of a Newton procedure. Moreover, the use of a first order scheme will degrade the temperature/flux accuracy. In order to solve the Stefan condition (2.10) a second order explicit scheme is proposed. In this Section, the extrapolated BDF2 strategy is chosen. Considering a generic equation $y'(t) = f(t, y(t))$, the BDF2 scheme can be written under the form

$$\gamma_{k+1}u^{k+1} + \gamma_k u^k + \gamma_{k-1}u^{k-1} = f(t^{k+1}, y^{k+1}), \tag{2.83}$$

where

$$\gamma_{k+1} = \frac{2\Delta t^k + \Delta t^{k-1}}{\Delta t^k(\Delta t^k + \Delta t^{k-1})}, \quad \gamma_k = -\frac{\Delta t^k + \Delta t^{k-1}}{\Delta t^k \Delta t^{k-1}}, \quad \gamma_{k-1} = \frac{\Delta t^k}{\Delta t^{k-1}(\Delta t^k + \Delta t^{k-1})}. \tag{2.84}$$

Equation (2.84) defines the coefficient γ_{k+1}, γ_k and γ_{k-1} for a **non constant time step**, Δt^k. Here, since an explicit scheme is used, the right-hand-side of Equation (2.83) is extrapolated with a second order accuracy using times t^k and t^{k-1}. This leads to the **second order extrapolated BDF2 scheme (BDF2exp)**:

$$\gamma_{k+1}u^{k+1} + \gamma_k u^k + \gamma_{k-1}u^{k-1} = \frac{\Delta t^k + \Delta t^{k-1}}{\Delta t^{k-1}} f(t^k, y^k) - \frac{\Delta t^k}{\Delta t^{k-1}} f(t^{k-1}, y^{k-1}). \tag{2.85}$$

For the interface position, one has to solve

$$\Gamma'(t) = v(t) \ , \tag{2.86}$$

where $\Gamma(t) = \boldsymbol{\Gamma}(t)\cdot\boldsymbol{n}$ 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^{k-1} and t^k (see Section 2.5.3), the position at time t^{k+1} can be evaluated without the use of a Newton process

$$\Gamma^{k+1} = -\frac{\gamma_k}{\gamma_{k+1}}\Gamma^k - \frac{\gamma_{k-1}}{\gamma_{k+1}}\Gamma^{k-1} + \frac{1}{\gamma_{k+1}}\left(\frac{\Delta t^k + \Delta t^{k-1}}{\Delta t^{k-1}} v^k - \frac{\Delta t^k}{\Delta t^{k-1}} v^{k-1}\right), \tag{2.87}$$

$$v^k = \frac{1}{\rho L m}[\![\boldsymbol{\beta}^k]\!]_{\Gamma^k}\cdot\boldsymbol{n}^k, \quad v^{k-1} = \frac{1}{\rho L m}[\![\boldsymbol{\beta}^{k-1}]\!]_{\Gamma^{k-1}}\cdot\boldsymbol{n}^{k-1}. \tag{2.88}$$

Note that as v depends directly on the flux jump $[\![\boldsymbol{\beta}]\!]_\Gamma$, a high order resolution of the flux will preserve the accuracy of the exBDF2 scheme.

2.6.2 Second order time scheme

This section is devoted to the time discretization of the variational problem, where a term concerning the derivation of T with respect to time takes place in the original system of equation of the Stefan model and its associated div-div stabilization term, see Section 2.2.3.b. Once again a BDF2 scheme is proposed. The main advantage is that the finite weak formulation and the computation of the location of the interface can be written at time t^{k+1}. As the procedure is the same for both sets of interface conditions in this section the procedure is presented with the second set of interface conditions (2.7).

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

$$
\begin{aligned}
&2.74.a)\ (\lambda^{-1}\boldsymbol{\beta}^{k+1}, \boldsymbol{w})_{\Omega^{k+1}} - (T^{*,k+1}, \nabla \cdot \boldsymbol{w})_{\Omega^{k+1}} + < T_D, \boldsymbol{w} \cdot \boldsymbol{n} >_{\Gamma_D} + < T^{*,k+1}, \boldsymbol{w} \cdot \boldsymbol{n} >_{\Gamma_N} + \\
&B_{stab}((T^{*,k+1}, \boldsymbol{\beta}^{k+1}), \boldsymbol{w}) + < \{\lambda^{-1}\boldsymbol{\beta}^{k+1} \cdot \boldsymbol{d}^{k+1} + 1/2 \boldsymbol{d}^{t,k+1} \nabla \cdot (\lambda^{-1}\boldsymbol{\beta}^{k+1}) \boldsymbol{d}^{k+1}\}_{\tilde{\Gamma}^{k+1}}, [\![\boldsymbol{w}]\!]_{\tilde{\Gamma}^{k+1}} \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}^{k+1}} \\
&+ < [\![\lambda^{-1}\boldsymbol{\beta}^{k+1} \cdot \boldsymbol{d}^{k+1} + 1/2 \boldsymbol{d}^{t,k+1} \nabla \cdot (\lambda^{-1}\boldsymbol{\beta}^{k+1}) \boldsymbol{d}^{k+1}]\!]_{\tilde{\Gamma}^{k+1}}, \{\boldsymbol{w}\}_{\tilde{\Gamma}^{k+1}} \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}^{k+1}} + \tilde{N}_{A_2}(T^*, q^*) \\
&= - < a_T, [\![\boldsymbol{w}]\!]_{\tilde{\Gamma}^{k+1}} \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}^{k+1}} - < j_T, \{\boldsymbol{w}\}_{\tilde{\Gamma}^{k+1}} \cdot \tilde{\boldsymbol{n}} >_{\tilde{\Gamma}^{k+1}} + L_{stab}(\boldsymbol{w}), \\
&2.74.b)\ (\rho c(\gamma^{k+1} T^{*,k+1} + \gamma^k T^{*,k} + \gamma^{k-1} T^{*,k-1}), q^*)_{\Omega^{k+1}} + (\nabla \cdot \boldsymbol{\beta}^{k+1}, q)_{\Omega^{k+1}} - < \boldsymbol{\beta}^{k+1} \cdot \boldsymbol{n} - h_N, q >_{\Gamma_N} \\
&+ A_{stab}((T^{*,k+1}, \boldsymbol{\beta}^{k+1}), q^*) = (f^{k+1}, q^*)_{\Omega^{k+1}} + \tilde{N}_{L_2}(q^*),
\end{aligned}
\tag{2.89}
$$

As a reminder, for a melting problem the values of a_T and j_T are constant with respect to the time variable t. Indeed, a_T is the melting temperature and depends on the material property which does not change through the simulation, while j_T characterizes the continuity of the primal variable. The boundaries Γ_D and Γ_N and the conditions imposed on them are also independent of time. The only issue in (2.89) is the definition of $T^{*,k}$ and $T^{*,k-1}$ for the nodes that change domains at time t^{k+1}. In such situations, according to their previous location, extrapolations or interpolations as presented in Section 2.5.1 need to be performed to obtain a consistent value for the previous time steps.

2.7 Conclusion on Chapter 2

In this chapter, an extension of the **Shifted Boundary Method** has been presented to handle **moving interfaces** in the context of a **melting front**. A **second-order accurate method** has been proposed, which effectively achieves second-order accuracy on the primal variable and its gradient. Judicious choices of time discretization schemes allow for the same level of accuracy in both the time variable and interface location. The different procedures proposed in this chapter to update the surrogate interface are tested in the following chapter of this manuscript, and the question of accuracy will be discussed and demonstrated.

Chapter 3

Numerical tests on the Mixed Stefan Model

In this Chapter, some results for polynomial analytical functions are presented to highlight the second order accuracy obtained by the enrichment of the primal variable and the efficiency of the constructions used to move the interface using the methods presented in Chapter 2. For the tests presented in Section 3.1, all generality are considered, and a discontinuity at the interface for the primal variable is also investigated. The examples will show the behavior of the method when the variables are continuous or not between the two phases. In Section 3.2 a self-made solution on a circular domain and for a circular interface will show the result on the accuracy of the method for a full 2D motion. Finally in Section 3.3 results for a physical example modeling the melting of a semi infinite ice block is presented, a test defined in [4] which possesses an analytical solution for the temperature field and the interface motion. Concerning that particular test, numerical results concerning the front motion, which can be compared to an analytical solution will be investigated more closely, to express the accuracy of the method on the tracking of the interface position.

3.1 Numerical results on polynomial analytical functions

In this section, validations are done using continuous and discontinuous variables. The two sets of interface conditions presented in (2.9) and (2.7) are being applied with the BDF2 scheme which will prove its robustness for the physical tests presented later on in Section 3.3. For simplification in the caption of the figures presented in this section, we will refer to **Set** 1 to results associated with **the first set of interface conditions** defined in (2.9) and to **Set** 2 to results associated with **the second set of interface conditions** defined in (2.7).

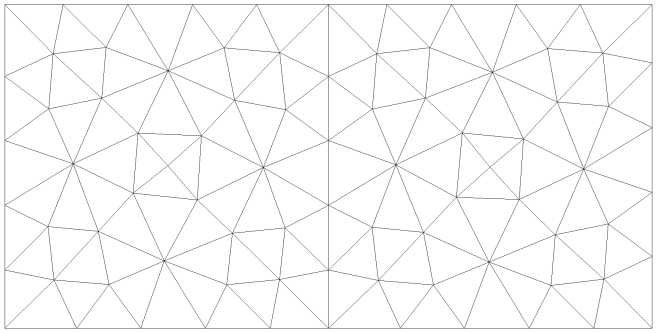
For the different test cases in this section, the time step Δt is defined by $\Delta t = \frac{h\ CFL}{2v}$, where v is a constant velocity used to move the interface, and CFL a constant number used to control the size of the time step. This definition allows to define a time step which ensures that not more than one element is crossed at a time by the physical interface. This condition will always be mandatory in all the numerical tests presented in this thesis manuscript. Indeed, in Chapter

2 we have introduced in Section 2.5 that some nodes need to be initialized when the physical interface moves. If more than one row of elements is crossed at a time then it is not possible to define a stencil of nodes to reconstruct a value for these nodes as all the values around them also need to be initialized, see Section 2.5.

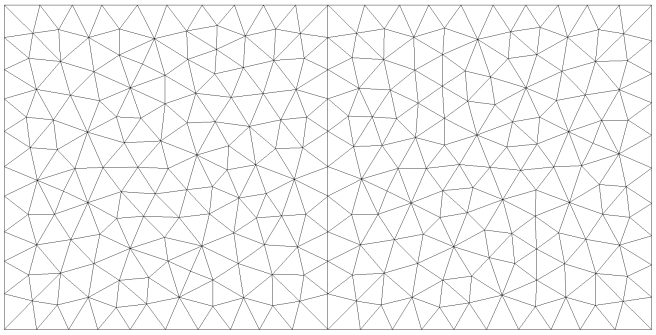
For the following sub-sections the accuracy of the method is tested with a uniform velocity, and the model will be extended, later on, in Chapter 4 to any types of interface structures and interface motions. In Section 3.1, the physical interface is modeled by a set of nodes defined by the same x coordinate for resolution on rectangular domains (flat interface). A similar configuration will be used in Section 3.2 where the points at the interface are defined by the same radius value. Moreover, the initial interface position is used to initialize the structure of the two domains Ω_+ and Ω_- which, with the use of the exact solutions is made possible. The initial interface position x_{init} is chosen for all the meshes and test cases such that $x_{init} = 0.5$. The $\mathbb{L}^2$ error is displayed for the final time t_f, identical for all test cases such that $t_f = 2$. The exact functions defining the different tests will be used to imposed Dirichlet boundary conditions on the left and right sides of the domain of resolution, Neumann conditions on the top and bottom boundaries of the domain (see Figure 3.0) and to define the initial conditions.

3.1.1 Meshes

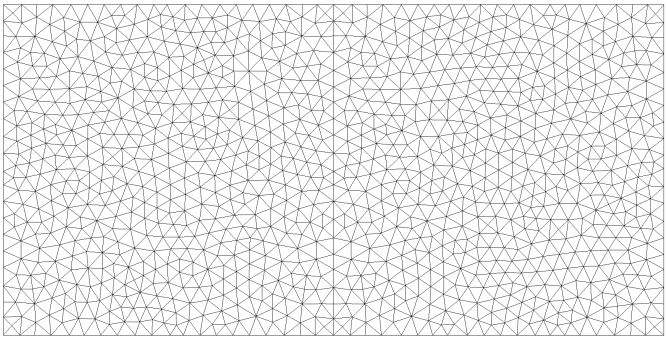
The different meshes used in Section 3.1 for the various simulations are presented. The variable h in Figure 3.0 represents the characteristic length of the elements of the considered mesh, Ne the number of elements and Nv the number of vertices, which is the same as the number of DoF for a $\mathbb{P}^1$ simulation. The variable h is the value used to define the CFL variable in the definition of the time step Δt, such that $\Delta t \times v < h$, which satisfies that not more than one row of elements is crossed at a time by the moving interface. All meshes are unstructured triangular meshes.



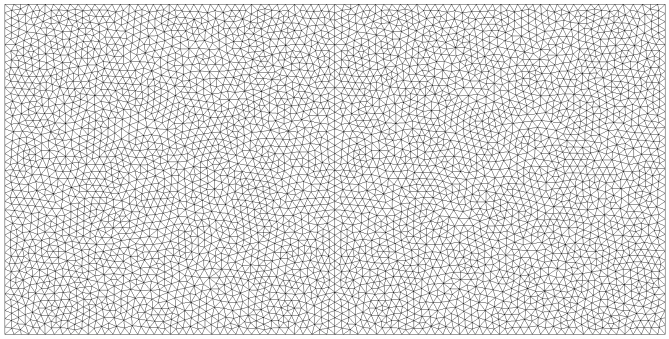
(a) Mesh 1 / $h = 0.2$ / $Ne = 136$ / $Nv = 84$



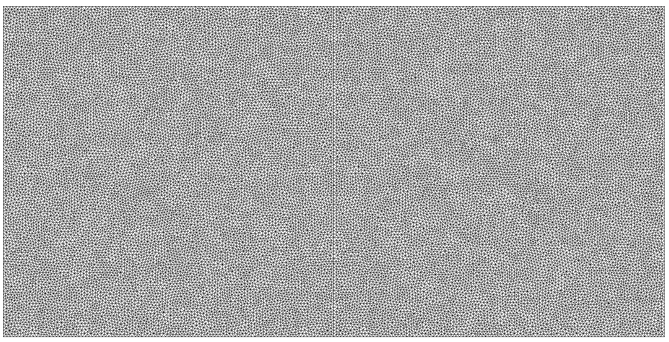
(b) Mesh 2 / $h = 0.1$ / $Ne = 492$ / $Nv = 277$



(c) Mesh 3 / $h = 0.05$ / $Ne = 2090$ / $Nv = 1106$



(d) Mesh 4 / $h = 0.025$/ $Ne = 8504$ / $Nv = 4373$



(e) Mesh 5 / $h = 0.0125$ / $Ne = 52966$ / $Nv = 26784$

Figure 3.0: Meshes For the Second Order accuracy verification

3.1.2 Machine precision

Machine precision is obtained for self-made tests defined by polynomial analytical solutions of degree less or equal to two, i.e., when $T \in \mathbb{P}^2(\Omega)$. The results is obtained for both sets of interface conditions (2.9) and (2.7), using the BDF2 scheme. $\mathbb{L}^2$ **Errors on** T and $\boldsymbol{\beta}$ are available in Tables 3.1 and 3.2 for the first set of interface conditions defined in (2.9) (Set 1), while for the second set of interface conditions defined in (2.7) (Set 2) results are displayed in Tables 3.3 and 3.4.

In order to compare different types of simulations the following abbreviations are defined:

- **sym enriched** : resolution with a symmetric enrichment, both the primal variable T and its associated test function are enriched using Definition (2.68),
- **non-sym enriched** : resolution with an enrichment of the primal variable T **only** using Definition (2.68).

The domain is rectangular $\Omega = [0, 2] \times [0, 1]$. Triangular elements of characteristic length $h = 0.05$ are used to mesh the domain, see Figure 3.0.c. $CFL = 1$, $\lambda_+ = 0.6$, $\lambda_- = 2.1$, $v = 0.5$, $\zeta_{div} = 0.5$, $t > 0$. Three tests are performed, where the exact functions for T are chosen such that

$$\text{Test 01} \rightarrow T(t,x,y) := \begin{cases} 7t+1 & \text{if } (x,y) \in \Omega_+ \\ 3t+3 & \text{if } (x,y) \in \Omega_- \end{cases} ; \text{ Test 02} \rightarrow T(t,x,y) := \begin{cases} x+t & \text{if } (x,y) \in \Omega_+ \\ xyt & \text{if } (x,y) \in \Omega_- \end{cases} \tag{3.1}$$

$$\text{Test 03} \rightarrow T(t,x,y) := \begin{cases} 4+33xy+21y^2+11t+6t^2 & \text{if } (x,y) \in \Omega_+ \\ 77+x^2+2t-8t^2 & \text{if } (x,y) \in \Omega_- \end{cases} . \tag{3.2}$$

T Polynomial	eT sym enriched	eT non-sym enriched	eT non enriched
Test 01	8.69037675694078E-013	1.07246296107150E-012	5.77492039596584E-013
Test 02	1.98905839613197E-012	3.29073939077805E-013	8.17679869657093E-005
Test 03	9.87454293814777E-013	3.88923825456129E-012	9.24537618134559E-003

Table 3.1: $\mathbb{L}^2$ Errors on T for polynomials of different degrees - Set 1

T Polynomial	e$\boldsymbol{\beta}$ sym enriched	e$\boldsymbol{\beta}$ non-sym enriched	e$\boldsymbol{\beta}$ non enriched
Test 01	5.87760006006547E-012	1.17309095334435E-011	6.34466424474283E-012
Test 02	7.34523157516188E-012	5.49049321486107E-012	7.10842777209640E-003
Test 03	4.70098442382738E-012	1.24055486419729E-011	0.182914290803606

Table 3.2: $\mathbb{L}^2$ Errors on $\boldsymbol{\beta}$ for polynomials of different degrees - Set 1

As can be seen in Table 3.1 and Table 3.2, doing a symmetric enrichment (sym enriched) or a non symmetric enrichment (non-sym enriched) 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 accuracy, as expected with the standard $\mathbb{P}^1$ FEM which is only first order accurate for β. The machine precision is not obtained for Test 02 and Test 03 functions in Equation (3.1) and (3.2), and the error on the flux $\boldsymbol{\beta}$ is higher.

T Polynomial	eT sym enriched	eT non-sym enriched	eT non enriched
Test 01	6.45889120732631E-013	5.56607024070457E-013	1.28224367945556E-012
Test 02	3.41553121024897E-013	4.34799948808748E-013	4.00045558192290E-005
Test 03	5.28262199105204E-013	5.12808329455716E-013	5.63157166009521E-003

Table 3.3: $\mathbb{L}^2$ Errors on T for polynomials of different degrees - Set 2

T Polynomial	e$\boldsymbol{\beta}$ sym enriched	e$\boldsymbol{\beta}$ non-sym enriched	e$\boldsymbol{\beta}$ non enriched
Test 01	3.29044974771348E-012	3.71568549940826E-012	4.97683492500090E-012
Test 02	4.54975964210432E-012	6.40200226600249E-012	5.49640705908567E-003
Test 03	6.35716917005916E-012	4.66651583565093E-012	4.10827076797471E-002

Table 3.4: $\mathbb{L}^2$ Errors $\boldsymbol{\beta}$ for polynomials of different degrees - Set 2

Similar results are obtained with the second set of interface conditions (Set 2) in Table 3.3 and 3.4. By comparing Tables 3.1 and 3.2 with Tables 3.3 and 3.4 we globally notice a better precision with the second set of interface conditions, see Equation (2.7). Moreover, depending on the choice made for the imposition of conditions at the interface, both options are still suitable. In the following, the results for both sets of interface conditions are investigated, where the front displacement is not related to the flux jump, but to a known velocity value. More precisely, the impact of the continuity or not of the functions at the interface on the accuracy of the method is studied.

3.1.3 Accuracy verification and Discontinuities

The previous section has demonstrated than the symmetry in the enrichment does not impact the accuracy of the method. For the following tests only a symmetric enrichment is considered, as symmetry in the weak formulation also brings stability in the scheme. From now on, both the primal variable T and its test function are enriched using Definition (2.68), when the enriched approach is considered. The interface is moving at a constant speed $v = 0.5$.

3.1.3.a Influence of the time reconstruction

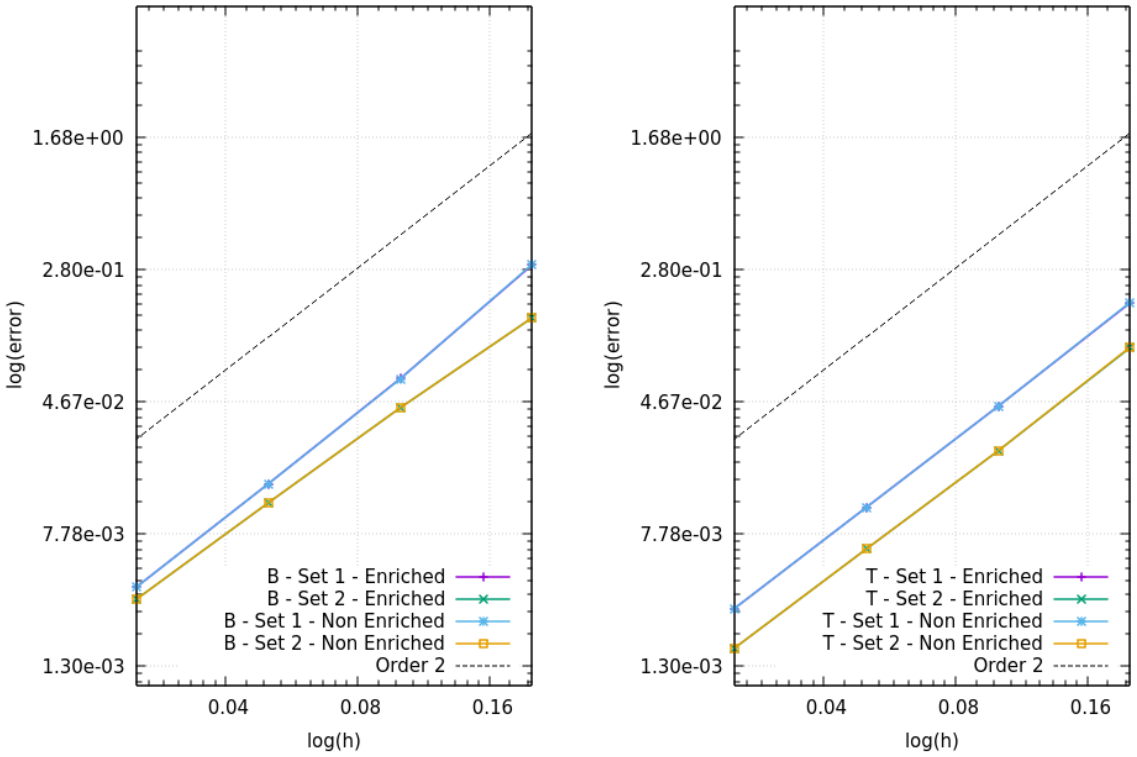
On the same computational domain given in Section 3.1.2, a discontinuous solution in the primal variable T and independent of the space variables (x, y) is investigated. This test case

is referred to as Test 1. The particular situation where T **depends only** on t and $\boldsymbol{\beta} = 0$ is studied to access **the temporal discretization** independently of the enrichment approach, and see if it has an impact on the accuracy of the method.

For Test 1, the function T is defined by

$$\text{Test } 1 \rightarrow T(t,x,y) = \begin{cases} 3t^2 + 7t + 1 + 11t^3 & \text{if } (x,y) \in \Omega_+, \\ 3t + 3 + 11t^2 + t^3/2 & \text{if } (x,y) \in \Omega_- \ . \end{cases} \tag{3.3}$$

As can be seen in Figure 3.1, the proposed reconstruction at the interface allows to achieve a second order accuracy on both T and $\boldsymbol{\beta}$. It shows that the reconstruction is indeed appropriate to conserve the second order accuracy brought by the enrichment. Moreover, the two sets of interface conditions give similar results.



(f) $\boldsymbol{\beta}$ continuous and null - Error in log scale

(g) T discontinuous - Error in log scale

Figure 3.1: Test 1 : Comparison between the enriched and non enriched method for the study of the time parameter influence - $CFL = 1.$, $v = 0.5$, $\zeta_{div} = 0.5$

3.1.3.b Flux discontinuity

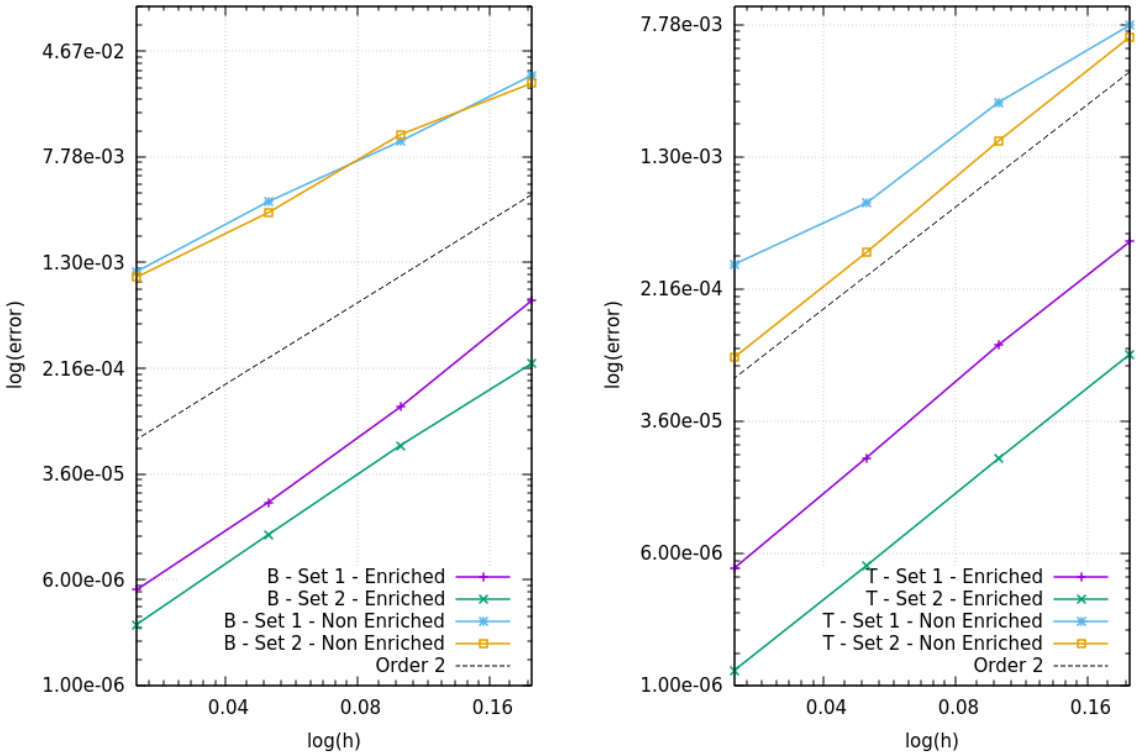
For Test 2 defined in (3.4) a continuous solution in the primal variable T and a **discontinuous solution in flux** $\boldsymbol{\beta}$ is studied. For this purpose, λ is chosen discontinuous between the two phases such that, $\lambda_+ = 0.6$ and $\lambda_- = 2.1$.

T is then defined by

$$\text{Test } 2 \rightarrow T(t,x,y) = log(t+1)/2 \ + y^2 \text{ if } (x,y) \in \Omega_+, \ \Omega_-. \tag{3.4}$$

A grid convergence study is performed. Figure 3.2 compares $\mathbb{L}^2$ errors on $\boldsymbol{\beta}$ and T to the theoretical second order (dotted line). As expected, Figure 3.2.b shows a second order accuracy

for T with the enriched approach. For the first set of interface conditions (2.9) the order is decreasing at convergence and is not of second order anymore. Moreover, both figures in Figure 3.2 show that the errors are significantly lower using the enrichment of the primal variable (enriched approach). For the flux $\boldsymbol{\beta}$, without enrichment, second order is not achieved for both sets of interface conditions (Set 1 in (2.9) and Set 2 in (2.7)), while it is obtained with enrichment in Figure 3.2(a) for both sets of interface conditions.



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

(b) T continuous - Error in log scale

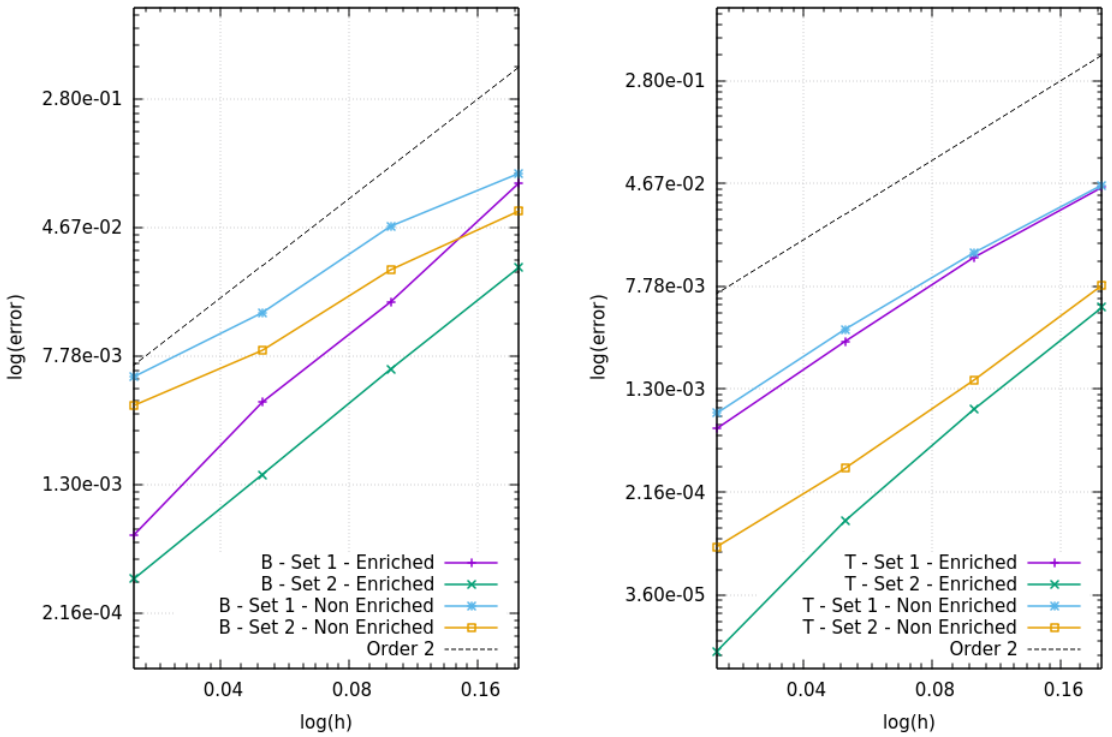
Figure 3.2: Test 2 : Comparison between the enriched and non enriched method for a discontinuity in $\boldsymbol{\beta}$ - $CFL = 1.$, $v = 0.5$, $\zeta_{div} = 0.5$

In Test 3, the situation where T and $\boldsymbol{\beta}$ are **both discontinuous** is investigated. The permeability variable λ is defined such that $\lambda_+ = 0.6$, $\lambda_- = 2.1$ and T is defined by

$$\text{Test 3} \rightarrow T(t,x,y) = \begin{cases} x^4 + 3xyt + 7 & \text{if } (x,y) \in \Omega_+, \\ log(x+1) + t^3 + y & \text{if } (x,y) \in \Omega_- \ . \end{cases} \tag{3.5}$$

Second order accuracy is obtained for T and for both sets of interface conditions (Set 1 in (2.9) and Set 2 in (2.7)), see Figure 3.3(b). Moreover, on all meshes, the error is always smaller using the enrichment of the temperature. On the flux, in Figure 3.3(a), the enrichment allows to achieve second order accuracy on $\boldsymbol{\beta}$, which was not the case without enrichment for Set 1 and Set 2, where the differences are largely visible. The errors are also globally reduced on all meshes with the enrichment of the primal variable using Definition (2.68).

Like the results observed in Figure 3.2.b , in Figure 3.3.b a second order accuracy is observed and verified with both sets of interface conditions on T, where for Test 3 the precision on T is approximately the same with or without enrichment.



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

(b) T discontinuous - Error in log scale

Figure 3.3: Test 3 : Comparison between the enriched and non enriched method for a discontinuity in T and $\boldsymbol{\beta}$ - $CFL = 1.$, $v = 0.5$, $\zeta_{div} = 0.5$

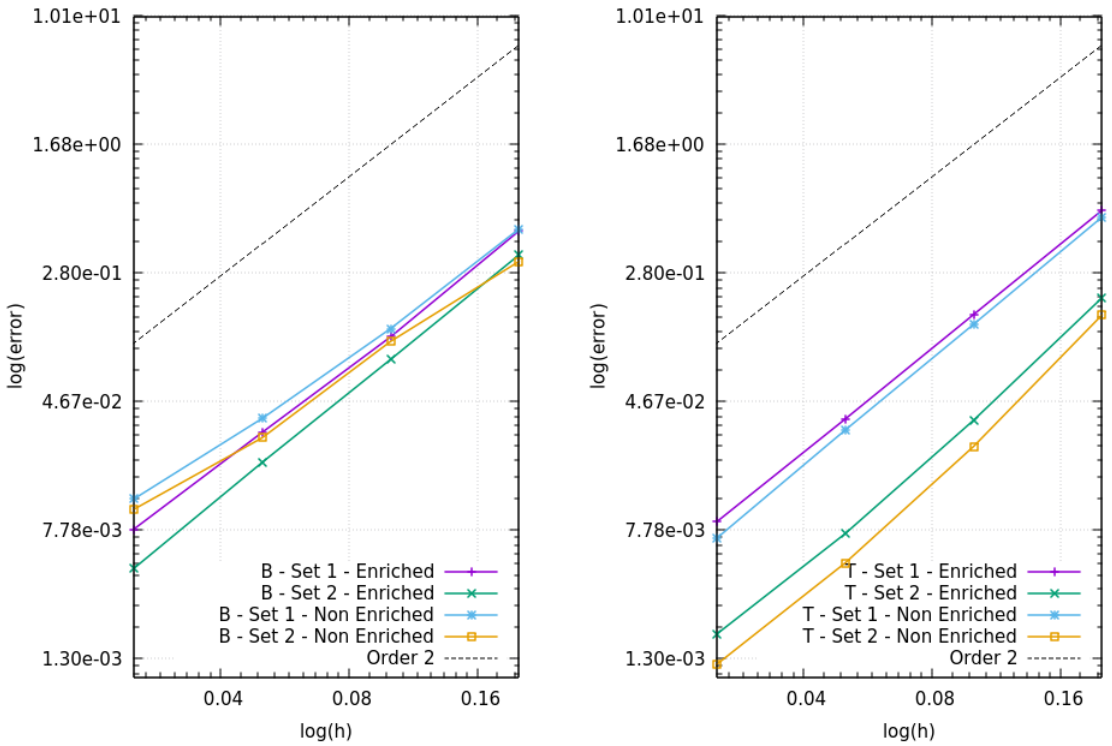
In Figure 3.3.a the non enriched method is not second order accurate and the difference in order of accuracy is even more observable in this example compared to Test 1 (see Figure 3.2.a), and only a first order accuracy is observed on β. Moreover, a better precision for the flux with the enriched method is obtained for both sets of interface conditions.

3.1.3.c Identical permeability values

Test 4 is devoted to the **continuity at the interface of the two functions** T and $\boldsymbol{\beta}$. The permeability variable is chosen identical on both phases such that $\lambda_+ = \lambda_- = 0.6$ and T is defined by

$$\text{Test } 4 \rightarrow T(t,x,y) = 7 + 7xy + 7y^3 + 7t - 3t^2 + 12t^3 \qquad \text{if } (x,y) \in \Omega_+,\ \Omega_- \ . \tag{3.6}$$

Using the enrichment process defined in Equation (2.68), a second order accuracy is achieved on the flux $\boldsymbol{\beta}$, see Figure 3.4(a). On the variable T, second order of accuracy is obtained by both approaches (the two sets of interface conditions) with or without enrichment of the primal variable, see Figure 3.4(b). For the two sets of interface conditions, the behavior of the enriched or non enriched simulation is sensibly the same. Moreover, **Set 2**, as for **all the previous test cases** always provides **errors smaller** than that for Set 1. For this test, in Figure 3.4.a it is observable that the enriched method has stabilized the error curve on $\boldsymbol{\beta}$, and that the method is indeed second order accurate.



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

(b) T continuous - Error in log scale

Figure 3.4: Test 4 : Comparison between the enriched and non enriched method for continuity in T and $\boldsymbol{\beta}$ - $CFL = 1.$, $v = 0.5$, $\zeta_{div} = 0.5$

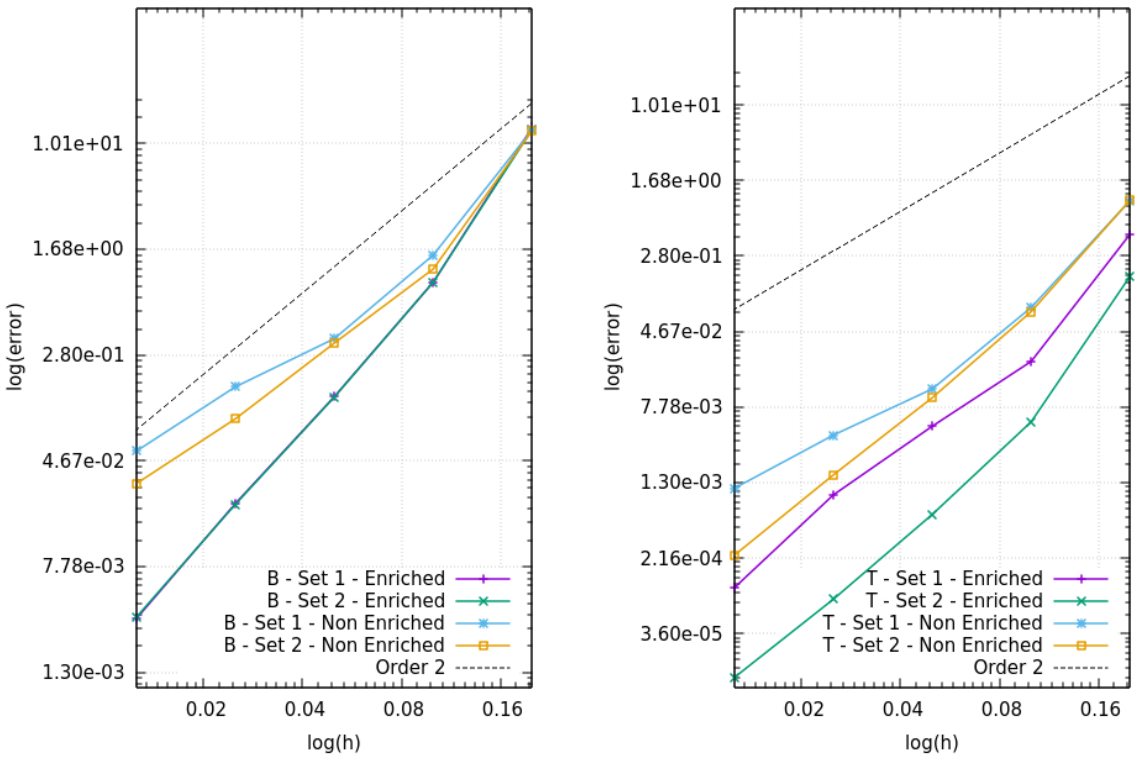
For 3.4.b its effect is even more noticeable, the curve representing the enrichment has a slope higher than 2, which is visible by comparing with the reference curve (dotted line) in Figure 3.4. The improvement brought by the enrichment is **significant** even for T, which is by definition already second order accurate.

3.1.3.d Discontinuity of the primal variable

The last test case, **Test 5** deals with a **continuous flux** $\boldsymbol{\beta}$ and a **discontinuous** variable T, λ is chosen such that $\lambda_+ = \lambda_- = 2.1$, and T is defined by

$$\text{Test } 5 \rightarrow T(t,x,y) = \begin{cases} exp(3xy) + log(t+1)/2 \ + t^3 & \text{if } (x,y) \in \Omega_+, \\ exp(3xy) + 2\sqrt{t} & \text{if } (x,y) \in \Omega_- \ . \end{cases} \tag{3.7}$$

Without enrichment, the approximation on $\boldsymbol{\beta}$ is of first order (see Figure 3.5(a)), while using the enrichment provides second order accuracy. Note also that the two enriched curves for Set 1 and 2 are mingled. As for T, the second order accuracy is achieved using both approaches (enriched and non enriched) for Set 2. However, for Set 1, the second order is not preserved on all meshes on the curve representing the non enriched simulation, see blue curve Figure 3.5(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 3.5: Test 5 : Comparison between the enriched and non enriched method for a discontinuity in T - $CFL = 1.$, $v = 0.5$, $\zeta_{div} = 0.5$

Whatever the combination **continuous/discontinuous** T and/or $\boldsymbol{\beta}$ at the interface, the enrichment and proposed reconstruction at the interface ensure 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 $\boldsymbol{\beta}$.

3.1.4 Conclusion

Both sets of interface conditions give the expected accuracy (second order). However, the second set of interface conditions gives a better approximation for every tests performed so far. The throwback of the first set of boundary conditions comes from the coupling of the Stefan conditions in the definition of the weak formulation and its use to move the physical interface. In the case of physical applications where the interface motion is also an unknown a Newton procedure is mandatory to recover the flux jump at the physical interface. The results of the current convergence tests performed in this section showed us that the precision is not better for the first set of interface conditions, and that additional work with this set of interface conditions should not be considered, and does not seem relevant to be considered an option as it would add more complexity in the resolution. In what follows, the second set of interface conditions is considered as the most efficient option and is used for all the other tests. The comparison made in this section has demonstrated that the imposition of conditions at the interface has an impact on the precision of the model. It has also indicated the best choice of interface conditions for the considered problem at stake. For the results presented in Chapter 4, only the second set of interface conditions (jump and average conditions on T at the interface) will be investigated in the study of stability.

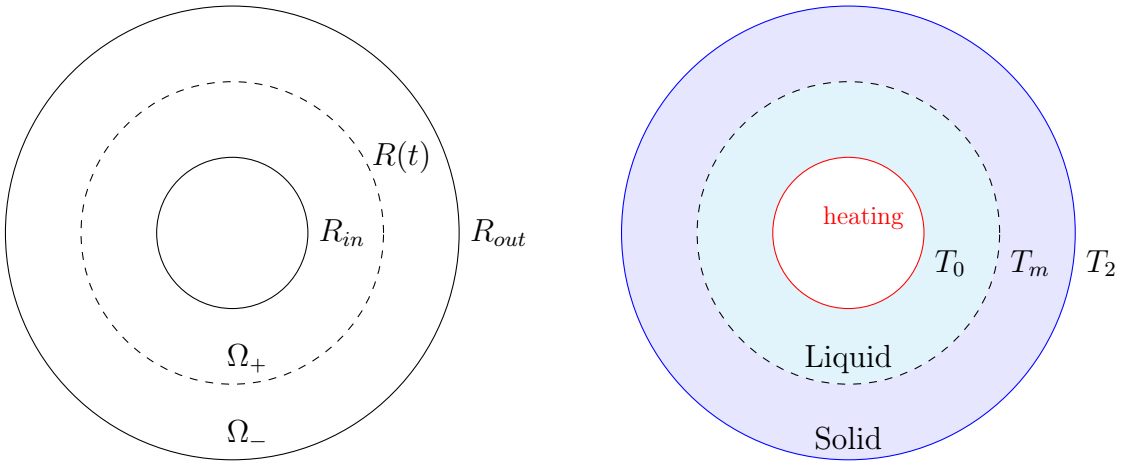
3.2 Tests on a circular domain

In this Section, a simulation on a circular domain is performed, where the physical interface is a circle represented by a radius value. In this situation, shifting the values of the conditions at the physical interface to a surrogate interface is an appealing idea to prevent the definition of small elements around the interface that would have to capture the interface geometry and adapt to its displacement. To present the improvements defined by the enriched Shifted Boundary Method (e-SBM), a convergence study is performed on a self-made test where the primal variable is defined continuous on the physical interface and where the moving interface operates in a full 2D mode. After the results presented in Section 3.1, only the second set of interface conditions (2.7) is now being considered.

3.2.1 Moving interface

In this Section, a simulation over an increasing circular interface in a full embedded situation is presented. The test is used to demonstrate the ability of the method when the moving interface expands in both directions, in the context of a $2D$ simulation.

The considered domain is a hollow cylinder in 2D, with inner radius R_{in} and outer radius R_{out}. Uniform heating of the inner surface R_{in} will result in a symmetric front $r = R(t)$ propagating from R_{in} to R_{out}, and separating the two phases, see Figure 4.22.



(a) Domain Definition (b) Temperature Data

Figure 3.6: Illustration of the test case for an inner melting configuration

To show the accuracy obtained with the enrichment method on a cylinder domain, a self-made function which allows to verify the order of convergence of the method is defined. For this test, the definition of the functions is given in polar coordinates. The exact solution for the temperature field T is defined by

$$T(t,r,\theta) = \begin{cases} T_m + (T_0 - T_m)\dfrac{ln(\frac{r}{R(t)})}{ln(\frac{R_{in}}{R(t)})} & \text{if } r \leqslant R(t), t > 0, \\ T_m + (T_2 - T_m)\dfrac{ln(\frac{r}{R(t)})}{ln(\frac{R_{out}}{R(t)})} & \text{if } r > R(t), t > 0\ , \end{cases} \tag{3.8}$$

where $R(t)$, the equation of the interface position, is chosen as follow

$$R(t) = R_{init} + (t - t_{init}) \times v_I, \tag{3.9}$$

with v_I a constant velocity. The definition (3.9) allows to define a uniform velocity for the interface displacement and allows the study of the behavior of the surrogate interface during the evolution of the physical front.

Definition (3.8) on the temperature field defines the following exact solution for the flux β:

$$\beta(t,r,\theta) = \begin{cases} -\lambda_1(\dfrac{(T_0 - T_m)}{ln(\frac{R_{in}}{R(t)})} cos(\theta) \times \dfrac{1}{r}) \\ -\lambda_1(\dfrac{(T_0 - T_m)}{ln(\frac{R_{in}}{R(t)})} sin(\theta) \times \dfrac{1}{r}) \end{cases} \quad \text{if } r \leqslant R(t),$$

and

$$\beta(t,r,\theta) = \begin{cases} -\lambda_2(\dfrac{(T_2 - T_m)}{ln(\frac{R_{out}}{R(t)})} cos(\theta) \times \dfrac{1}{r}) \\ -\lambda_2(\dfrac{(T_2 - T_m)}{ln(\frac{R_{out}}{R(t)})} sin(\theta) \times \dfrac{1}{r}) \end{cases} \quad \text{if } r > R(t), \tag{3.10}$$

and the source term of Equation (2.8) is defined by

$$f(t,r,\theta) = \begin{cases} \dfrac{vI}{R(t)}(T_0 - T_m)\dfrac{ln(\frac{r}{R_{in}})}{ln(\frac{R_{in}}{R(t)})^2} & \text{if } r \leqslant R(t), \\ \dfrac{vI}{R(t)}(T_2 - T_m)\dfrac{ln(\frac{r}{R_{out}})}{ln(\frac{R_{out}}{R(t)})^2} & \text{if } r > R(t). \end{cases}$$

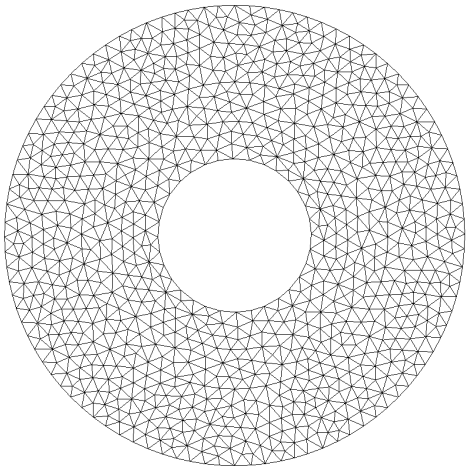
The analytical solutions defined in (3.8) and (3.10) are used to defined initial conditions.
In Table 3.5, the parameters used to specify the circular test case are defined. Any other variables used for this test are taken from Table 3.8, used for the definition of the last test presented in this Chapter 3. The velocity v_I is chosen to satisfy $\Delta t \times v_I < h$, meaning that only one row of mesh elements at a time can be crossed per iteration.

Symbol	Value	Physical Variable
T_0	$263.15(K)$	Inner temperature on R_{in}
T_2	$283.15(K)$	Outer temperature on R_{out}
Δ_t	$h^2/min(\lambda_{Water}, \lambda_{Ice})$	Time step
v_I	$0.1(m.s^{-1})$	Interface velocity
R_{in}	$0.01(m)$	Interior radius
R_{out}	$0.03(m)$	Outer radius
R_{init}	$0.015(m)$	Initial front position

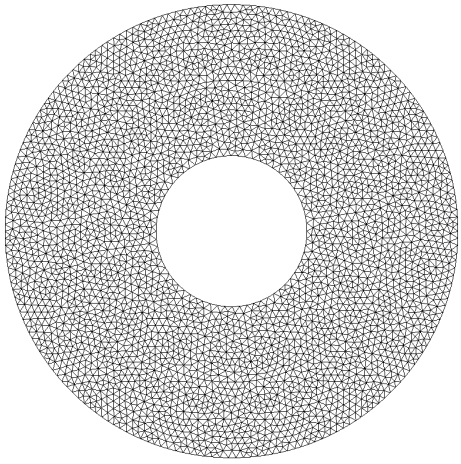
Table 3.5: Physical parameters for the circular test

3.2.2 Meshes and configuration of the domain of resolution

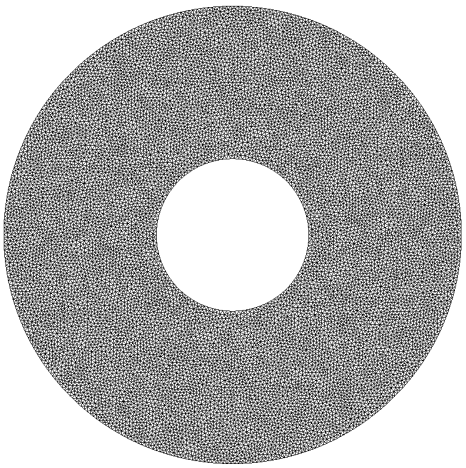
In Figure 3.7, the different meshes used for the convergence test defined in Figure 3.9 and Tables 3.6 and 3.7 are presented. All the meshes are unstructured and defined with triangular elements. The variable h defines the characteristic length of the elements of the considered mesh, Ne the number of elements and Nv the number of vertices, equals to the number of DoF.



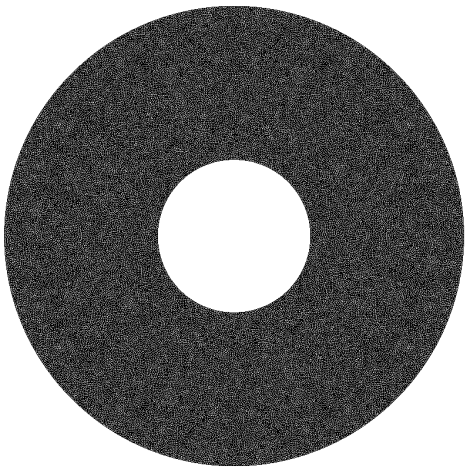
(a) $h = 0.2$ / $Ne = 1790$ / $Nv = 944$



(b) $h = 0.1$ / $Ne = 3480$ / $Nv = 6706$



(c) $h = 0.05$ / $Ne = 26334$ / $Nv = 13419$



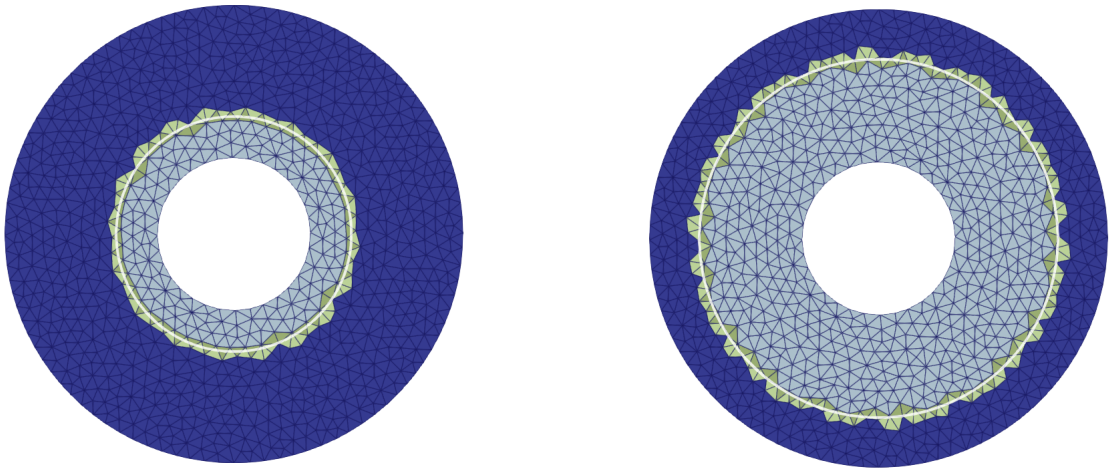
(d) $h = 0.025$ / $Ne = 107730$ / $Nv = 54368$

Figure 3.7: Meshes for the convergence study of the circular test

3.2.3 Numerical results

Results associated with the simulation of the test defined in Equation (3.8) are now presented. In Figure 3.8, the expansion of the inner phase and the modification of the surrogate interface at two instances of time are shown. The white circle represents the physical interface, the green

elements are the elements intersected by the physical interface and used to define the surrogate interface. In this situation, the surrogate interface is chosen as the inward boundary of the area defined by the cut elements in Figure 3.8. In Figure 3.9, the second order of accuracy is achieved for both variables with the temperature enrichment defined in Section 2.4 by Definition (2.68). Third order accuracy is even achieved (grey dotted line as a comparison) on the temperature, as shown in Figure 3.9.b The values used in Figure 3.9 are available in Tables 3.6 and 3.7, where the improvement on the flux by the temperature enrichment is more visible. The third accuracy achieved on T can be explained by the regularity of the analytical solutions defining the presented test. The temperature enrichment mimics the results of a standard $\mathbb{P}^2$ simulation, which would provide a third-order accuracy on T. In the code implemented for this thesis work, the Taylor expansion on T is conducted using the Hessian of T, allowing tests with a high degree of regularity to achieve third-order accuracy on the temperature field.



(a) Time t = 1(s)

(b) Time t = 8.5(s)

Figure 3.8: Visualization of the surrogate and interface displacement

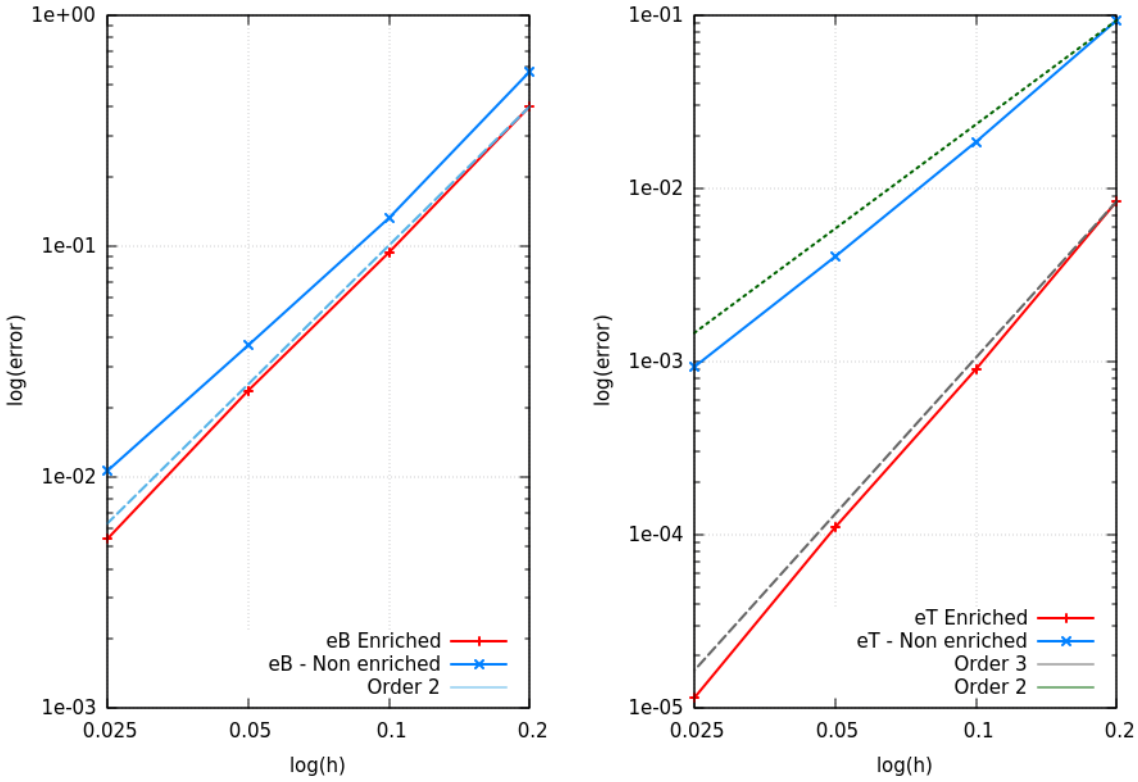
h	eT Enriched	Slope Enriched	eT Non Enriched	Slope Non-Enriched
0.2	8.41E-03		9.33E-02	
0.1	9.02E-04	3.21	1.85E-02	2.33
0.05	1.10E-04	3.04	4.02E-03	2.20
0.025	1.14E-05	3.26	9.29E-04	2.11

Table 3.6: $\mathbb{L}^2$ Errors on T for enriched and non enriched simulation with associated slope

h	eβ Enriched	Slope Enriched	eβ Non Enriched	Slope Non-Enriched
0.2	4.02E-01		5.69E-01	
0.1	9.37E-02	2.11	1.32E-01	2.08
0.05	2.36E-02	1.99	3.70E-02	1.83
0.025	5.38E-03	2.14	1.05E-02	1.81

Table 3.7: $\mathbb{L}^2$ Errors on β for enriched and non enriched simulation with associated slope

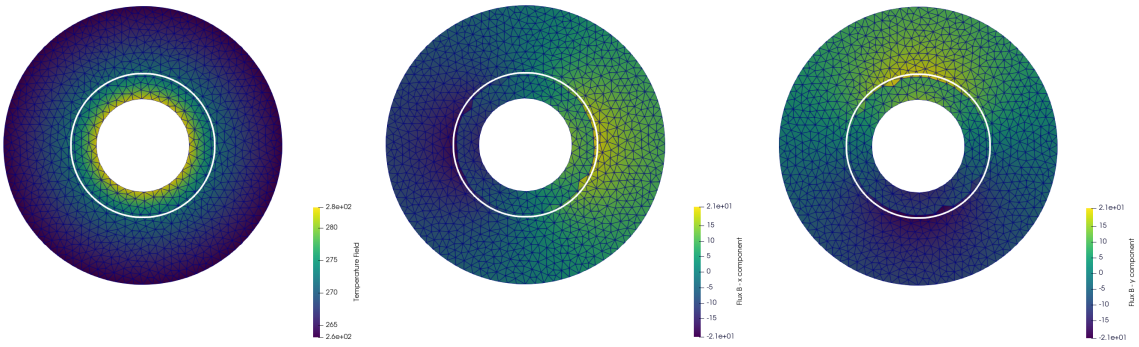
As a last result for this test, Figure 3.10 and Figure 3.11 illustrate snapshots of the solution at time $t = 1(s)$ and $t = 8.5(s)$, where the surrogate interface configuration is shown in Figure 3.8. The temperature is increasing from the heat source located at $r = R_{in}$, the inner radius and the the behavior of the flux follows the position of the front properly. No discontinuities are visible on the calculated temperature. This test has demonstrated the accuracy of the method on an interface moving with a 2D motion, and the ability of the e-SBM to impose properly interfaces conditions in the context of moving boundaries.



(a) β - Error in log scale

(b) T - Error in log scale

Figure 3.9: Comparison between the enriched and non enriched method - Circular Test

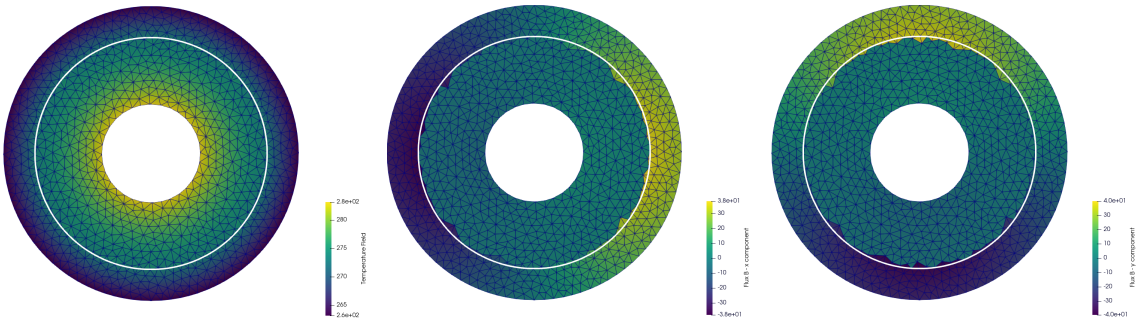


(a) Temperature

(b) β_x

(c) β_y

Figure 3.10: Snapshots of the temperature field T and flux β at Time t = 1(s)



(a) Temperature (b) β_x (c) β_y

Figure 3.11: Snapshots of the temperature field T and flux β at Time t = 8.5(s)

3.3 Physical test : Ice block Melting

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 reference [4].

3.3.1 Test case definition

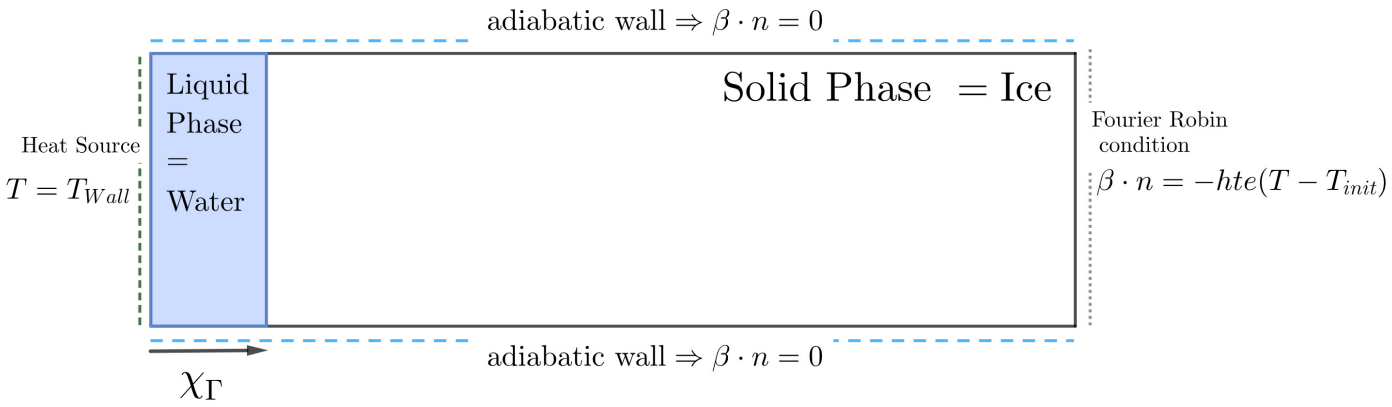


Figure 3.12: Illustration of the test case

The computational domain in Figure 3.12 is a rectangular box $\Omega = [0\,m, 0.5\,m] \times [0\,m, 0.025\,m]$, for the simulation of a semi-infinite ice block. 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 moves in the direction from left (water) to right (ice). 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 chosen sufficiently large that the condition does not affect the position of the front. The physical parameters of the problem are summarized in Table 3.8.
In this work, the density is chosen constant, s.t. $\rho_{ice} = \rho_{water} = 1000(Kg.m^{-3})$. The assumption is made in [4], for the definition of the analytical test case too. In order to define the analytical position of the interface, the Stefan numbers for liquid water and ice are introduced in (3.11). The Stefan number characterizes the rate of phase change, and is characterized by the ratio

Symbol	Value	Physical Variable
T_{Wall}	$283.15(K)$	Heat source temperature
T_r	$263.15(K)$	Exterior temperature
T_{init}	$263.15(K)$	Initial temperature of the ice block
Tm	$273.15(K)$	Phase change temperature
hte	$300(w.m^{-2}.K^{-1})$	Heat transfer coefficient
ρ	$1000(Kg.m^{-3})$	Material density
Lm	$333000(J.Kg^{-1})$	Latent heat
λ_{Ice}	$2.1(W.m^{-1}.K^{-1})$	Thermal conductivity / Ice phase
λ_{Water}	$0.6(W.m^{-1}.K^{-1})$	Thermal conductivity / Water phase
c_{Ice}	$2060(J.Kg^{-1}.K^{-1})$	Specific heat / Ice phase
c_{Water}	$4185(J.Kg^{-1}.K^{-1})$	Specific heat / Water phase

Table 3.8: Physical parameters

between the sensible heat (heat exchanged) and the latent heat (energy released or absorbed)

$$St_W := \frac{c_{Water}(T_{Wall} - T_m)}{Lm} \quad , \quad St_I := \frac{c_{Ice}(T_m - T_r)}{Lm}. \tag{3.11}$$

In this study, the creation of an interface addressing the process of ice melting starting from a unique solid/ice configuration, has not been explored due to time constraints. This remains a potential area for future research and extension of the method. For this reason, the initial position of the interface depends on the initial time of the simulation and is given by the initial conditions of the Problem (2.8). 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, which is refined around the initial interface position (see Figure 3.13 and Table 3.9). This allows for a quicker simulation than an homogeneous space discretization and provides the needed precision to move the interface, see Section 2.5.

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 3.9: Mesh characteristic following h for results in Figure 3.15

For the 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 $\chi(t)$ in (3.12) is the analytical position of the interface and is used to validate the results obtained by the simulations.

$$\chi(t) := 2\chi\sqrt{\alpha_L t} \quad , \quad \nu := \sqrt{\frac{\alpha_L}{\alpha_S}} \quad , \quad \alpha_L := \frac{\lambda_{Water}}{\rho c_{Water}} \quad , \quad \alpha_S := \frac{\lambda_{Ice}}{\rho c_{Ice}} \ , \tag{3.12}$$

with χ as a solution of (3.13)

$$\frac{St_L}{exp(\chi^2)erf(\chi)} - \frac{St_S}{\nu exp(\nu^2\chi^2)erfc(\nu\chi)} - \chi\sqrt{\pi} = 0 \quad . \tag{3.13}$$

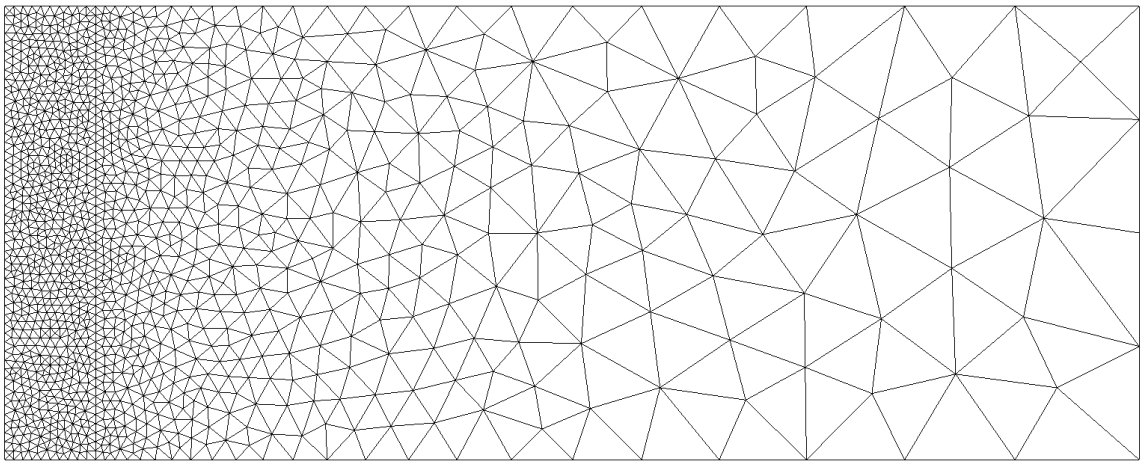


Figure 3.13: Discretization of the domain of resolution - ice block test

Equation (3.13) is solved using a Newton algorithm. Here, the value of χ found is $\chi \approx 0.2018$, for the variables defined as in Table 3.8. For the convergence tests in Section 3.3.3 the value of the calculated temperature field and the associated flux will be compared to the analytical solutions (3.14) and (3.15) for T and (3.16) and (3.17) for β, defined for their respective phase. The source term f in Problem (2.8) is null for both phases such that $f(t,x,y) = 0,\ \forall (x,y) \in \Omega_+, \Omega_-$.

Analytical **temperature** in the **liquid region** (water) :

$$T(t,x,y) = T_{Wall} - (T_{Wall} - T_m)\frac{erf(\frac{x}{2\sqrt{\alpha_l t}})}{erf(\chi)}. \tag{3.14}$$

Analytical **temperature** in the **solid region** (ice) :

$$T(t,x,y) = T_{init} + (T_m - T_{init})\frac{erfc(\frac{x}{2\sqrt{\alpha_s t}})}{erfc(\chi\sqrt{\frac{\alpha_l}{\alpha_t}})}. \tag{3.15}$$

Definitions (3.14) and (3.15) are used to defined the analytical flux as follow:

Analytical **flux** in the **liquid region** (water) :

$$\beta(t,x,y) = \begin{cases} \beta_x(t,x,y) = \lambda_{Water}\ \dfrac{(T_{Wall} - T_m)}{erf(\chi)\sqrt{\pi\alpha_l t}}\ exp(\dfrac{-x^2}{4\alpha_l t}) \\ \beta_y(t,x,y) = 0 \end{cases} . \tag{3.16}$$

Analytical **flux** in the **solid region** (ice) :

$$\beta(t,x,y) = \begin{cases} \beta_x(t,x,y) = \lambda_{Ice}\ \dfrac{(T_m - T_{init})}{erfc(\chi\sqrt{\frac{\alpha_l}{\alpha_s}})\sqrt{\pi\alpha_s t}}\ exp(\dfrac{-x^2}{4\alpha_s t}) \\ \beta_y(t,x,y) = 0 \end{cases} . \tag{3.17}$$

Remark 12. The function erf is the error function, while $erfc$ is the complementary error function. There are defined in Equation (3.18) using the leading factor $\frac{2}{\sqrt{\pi}}$ which is sometimes omitted by some authors.

$$\begin{aligned} erf(x) &= \frac{2}{\sqrt{\pi}} \int_0^x exp(-t^2)\ dt\ , \\ erfc(x) &= \frac{2}{\sqrt{\pi}} \int_x^\infty exp(-t^2)\ dt\ . \end{aligned} \tag{3.18}$$

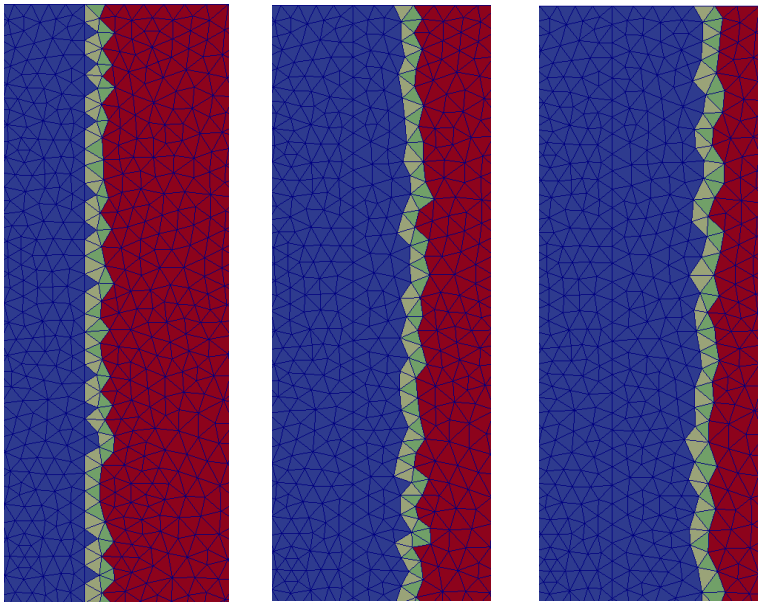
The two functions satisfy

$$erf(x) + erfc(x) = 1\ .$$

Remark 13. The test presented in this section for the simulation of a semi-infinite ice-block is performed over a 2D domain (see Figure 3.12), but with a uniform front motion in one direction only, see Equation (3.13). Then, a single velocity value is used to update the front position, and all the points at the physical interface are moved at the same speed (flat interface). For the computation of this test, the nodes at the surrogate interface from the mesh discretization are used to retrieve positions of nodes on the physical interface where the flux jump is calculated, using the mapping function (1.12) that links the two interfaces together.

The methods described in Section 2.5 are used to approximate values of $[\![\beta]\!]_\Gamma$, which through the Stefan condition (2.10) gives an approximation of the front velocity. The average of those values is then used as a single velocity value to update the front position. Taking the average value can be seen as a smoothing technique on the calculation of the speed of the physical interface. The stability analysis in Chapter 4 will develop the discretization of the interface to a full 2D motion (both in the x and y direction) and every point at the physical interface will be moved independently of each other.

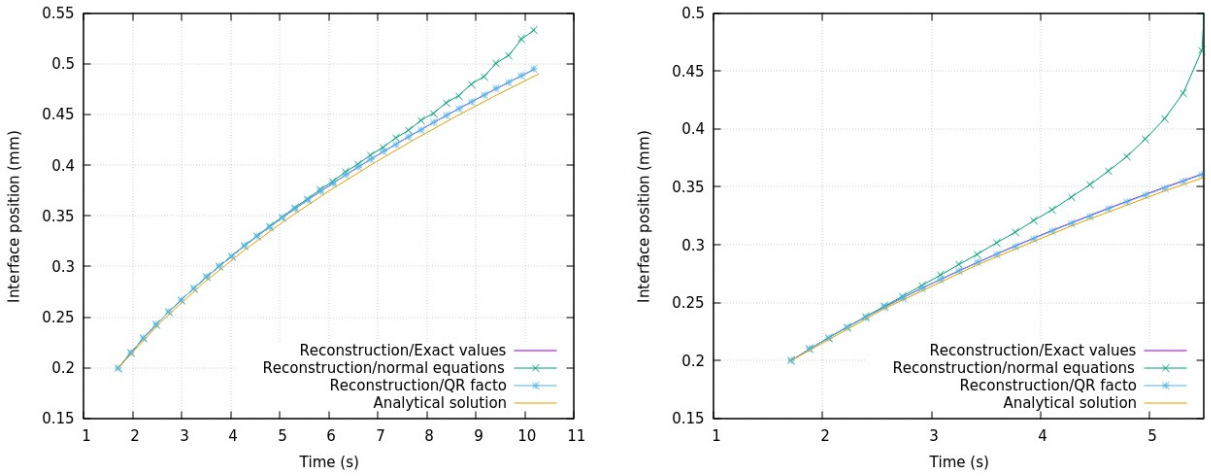
3.3.2 Observations



Iteration 1 Iteration 6 Iteration 12

Figure 3.14: 2D View from the top of the domain (h=0.00008) and the displacement of the surrogate at different time step

In Figure 3.14, the results on the surrogate interface identification is shown for three different times. In blue, in Figure 3.14, the liquid phase is expanding while in the red area, the solid phase is getting smaller. The elements in light and dark green are those that are intersected by the physical interface on both Ω_+ and Ω_-.
It is important to examine the impact of node reconstruction, which can potentially deteriorate the exact front position. Following the two approaches described in Section 2.5.1, for the reconstruction, the position of the interface for different element sizes in Figure 3.15 is studied. The problem solved is a least square minimization problem and three methods and their



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

Figure 3.15: Comparison between the exact interface position and the numerical positions

results on the front position are compared to the analytical front position (orange curve). The exact values method (purple curve) involves initializing the missing values with the analytical solutions defined in (3.14),(3.15),(3.16) and (3.17).

The curve is mingled with the analytical solution curve (see Equation (3.12)) in Figure 3.15. The reconstruction with normal equations consists in solving the equivalent system $A^tAX = A^tB$ of the initial system $AX = B$, green curve in Figure 3.15. The QR factorization method (blue curve) shows the interface position when the missing values are reconstructed by solving system $RX = Q^tB$, equivalent to $AX = B$ by property of the QR factorization, see Section 2.5.2. One 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, see Figure 3.15(b). We can see that the QR factorization is giving better results than the normal equations method. The main advantage of the QR factorization is that no inversion is required and that the QR factorization is known to be stable under small perturbations. The R matrix of the QR factorization is a triangular matrix, while the matrix A^tA need to be inverted using a direct or iterative solver. The Figure 3.15(b) shows that the condition number of the matrix A^tA is bigger for refined mesh, which makes the QR procedure a more pertinent method. Another approach could have considered a bigger stencil of points for the resolution of $A^tAX = A^tB$, but a detailed analysis would have been required to determine a rule allowing an efficient use of this method on any types of mesh. The QR factorization has been preferred instead.

3.3.3 Numerical results

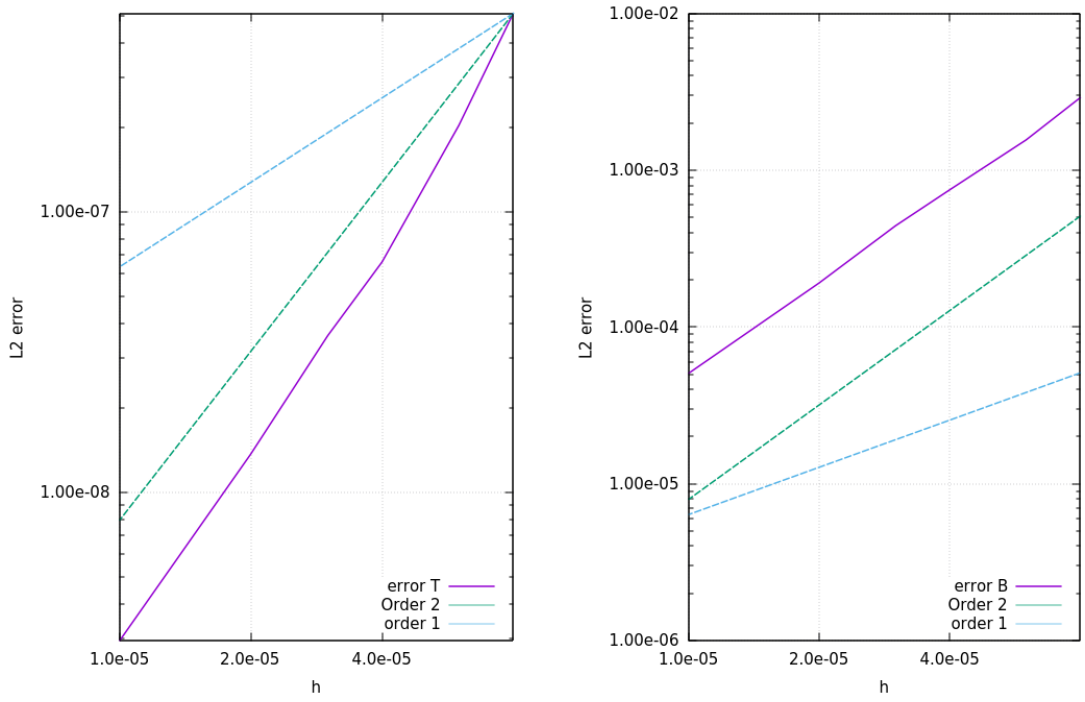
In order to investigate properly the accuracy of the e-SBM for this particular test the simulations are restrained to meshes with a constant characteristic length.

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

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

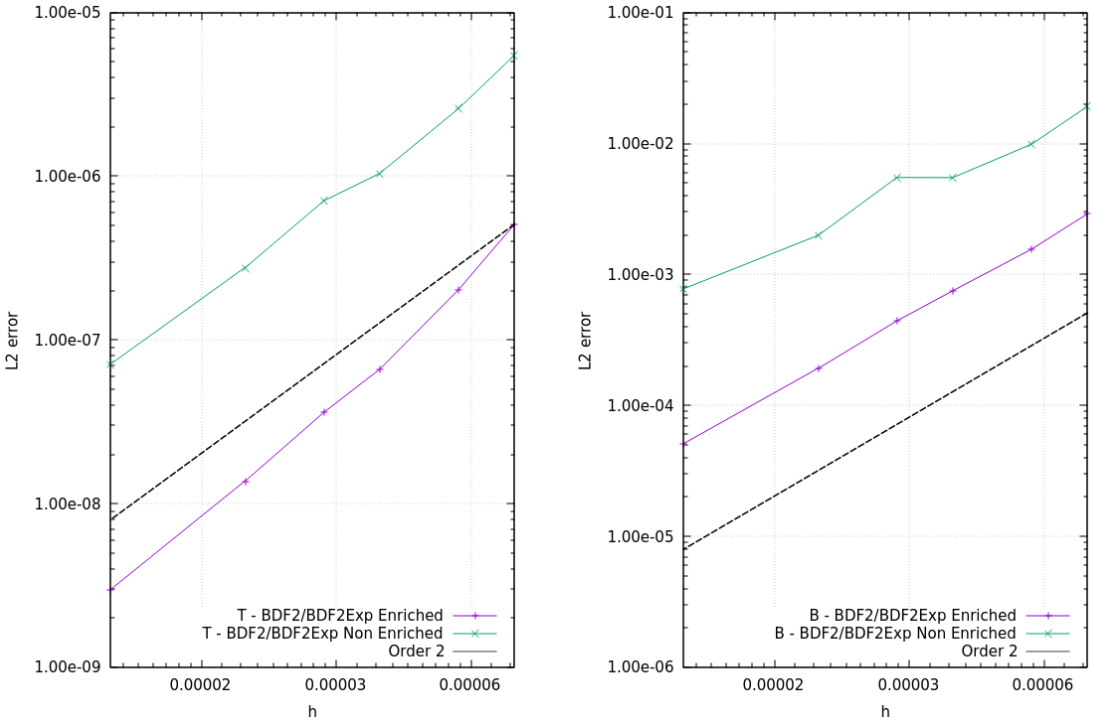


(a) Error on T

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

Figure 3.16: Convergence study - Error at final time Tf = 5 seconds

In Table 3.10, the errors obtained from the simulations performed over different meshes are displayed. The order of accuracy obtained is indeed 2 for both variables T and β, using the enrichment of temperature defined in (2.68). The values in Table 3.10 are used in Figure 3.16 and compared to two reference curves, one for the first order accuracy (blue curve in Figure 3.16) and one for the second order accuracy (green curve in Figure 3.16).



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

Figure 3.17: Convergence study on the accuracy of an enriched and non enriched simulation

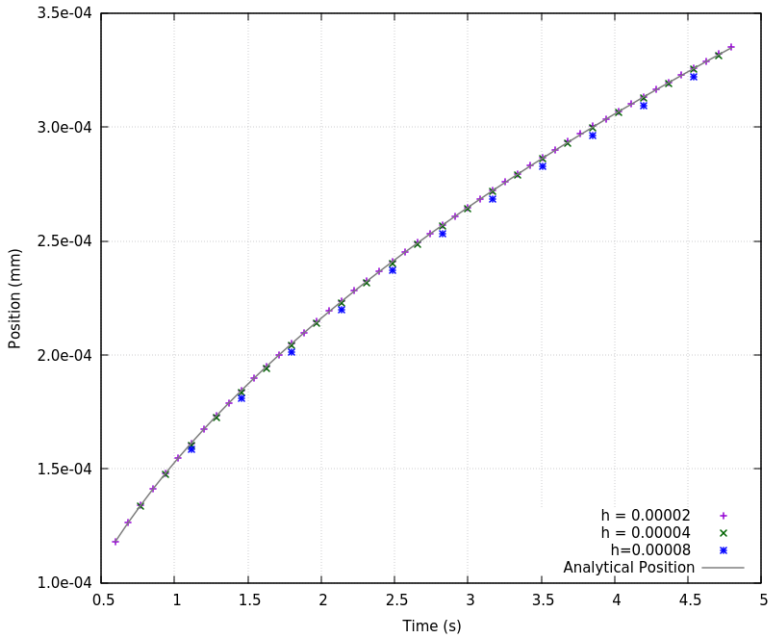


Figure 3.18: Comparison between the numerical and analytical front position

In Figure 3.17, results presented in Tables 3.10 and 3.11 are used to compare the differences

between the enriched (symmetric enrichment) and non enriched simulation, where all schemes used are second order accurate. The BDF2 scheme is used on the time discretization and the extrapolated BDF2 (BDF2exp) is used on the discretization of the Stefan condition for the update of the interface position, see Section 2.6.1. A degradation of the order of convergence is observable when the enrichment of the temperature is deactivated, see Figure 3.17 and Table 3.11.

In Figure 3.18, the e-SBM is able to correctly track the position of the front, which is also an unknown of the problem, the more refined the mesh is, the more the numerical front position matches the analytical position. These results show the ability of the method on the tracking of the front position. For this test, once again the e-SBM is an effective procedure to obtain and maintain an overall second order accuracy for problems with a moving interface.

3.4 Conclusion on Chapter 3

In this Chapter 3, we have demonstrated that the e-SBM defined in Chapter 2 and the schemes developed in (2.71) and (2.76) are efficient for the modeling of the unsteady Stefan problem. The expected accuracy has been verified and the enrichment of the method has proven its benefits in all the tests presented in this Chapter. The second set of interface conditions (2.7) exhibited better results and precision than the first set of interface conditions (2.9), making it the preferred choice for the resolution of the Stefan model. Therefore, it will be the only set of interface conditions used in Chapter 4 for the stability analysis.

In the next chapter, which is the last one of this manuscript, the stability of the method will be explored. More complicated interface geometry will be used, and perturbations on the front position and the temperature field will be investigated in a melting configuration to study the behavior of the method in regards to perturbations

Chapter 4

Stability Analysis of the unsteady Stefan model

Classical finite element methods used to model problems with internal boundaries rely on body fitted computational grids. However, those methods encounter computational challenges when the boundaries are deformed or moved substantially through time. In this direction, embedded methods do not require the use of boundary-fitted grids; instead, they involve immersing the boundary within a pre-existing fixed grid. This ability justifies the choice made in this thesis to operate within the embedded framework and be able to deal with the moving front of the Stefan model. In the first two chapters an extension of the Shifted Boundary Method (e-SBM) to moving internal boundaries has been presented. The method has been applied in Chapter 3 for non discretized interface on the Stefan model and its phase-change front, showing the accuracy of the method and its ability to track moving interfaces. In the context of modeling more complex structures the interface is discretized as a set of nodes. The recovery of the velocity of those nodes allows for the movement of the physical interface position within the mesh discretization. Without a proper analysis, this displacement of the interface and the stability of the model remain unknown, especially within the embedded framework. The question of stability of the e-SBM and the Stefan model is the subject of discussion of this chapter. More precisely, a linear stability analysis is performed on the analytical Stefan model for a melting configuration and on the numerical problem solved by the e-SBM. Such studies have been applied to different physical problems such as in [96, 97, 98, 99, 100] but to the best of the author's knowledge, no study of this nature has been conducted on the Stefan model within the framework of embedded simulations. On a numerical side, the stability of the method expressed in Chapter 2 and the scheme developed in Equation (2.89) will be studied. The subject of discussion will be the displacement of a perturbed front where the velocity is recovered through the Stefan condition and the behavior of this perturbation through the simulation. The temperature field is also studied to look in a similar way into the growth or decay in time of a perturbation on the primal variable and its impact on the definition of the flux, present explicitly in the numerical scheme by the use of the mixed formulation, and used to move the interface through the Stefan condition.

4.1 Stability Analysis

In order to introduce the stability analysis of the unsteady Stefan model, the stability analysis procedure is initially expressed for simple problems. A preliminary example is detailed with the Fisher equation which possesses both a stable and an unstable mode.

4.1.1 Principle & Concept

Suppose that $u(x,t)$ is a function defined on a domain $\{-\infty < x < +\infty,\, t > 0\}$, which satisfies a **linear** PDE with **constant** parameters such as the wave or diffusion equation.

Let $k \in \mathbb{R}$ be a **wavenumber**, $s \in \mathbb{C}$ the associated **frequency**, $x \in \mathbb{R}$ the space variable, $t \in \mathbb{R}_+$ the time variable and i the standard notation for the imaginary number $i^2 = -1$. In this chapter, the variable i as a number will always refer to the complex number. Then, Equation (4.1) characterizes a solution to a linear PDE with a **growth** or **decay** in time behavior, see [124].

$$\boxed{u(x,t) = exp(ikx + st)} \tag{4.1}$$

Plugging a solution of the form (4.1) into a linear PDE results in a relationship of the form $s := s(k)$ called the **dispersion relation**. The dispersion relation characterizes the dynamics of spatially oscillating modes relating the **wavelength** or **wavenumber** k of a wave to its **frequency** s. In the cases where s is independent of k, the system is characterized by no dispersion.

The coefficient s being a complex number, and the sign of the real part of s, denoted $Re(s)$ ($Im(s)$ denotes its imaginary part) characterizes whether a solution of the form (4.1) has a growth or decay in time behavior. Indeed, the value of x in the solution (4.1) depends on the complex number i, defining the exponential part on x as a sum of trigonometric functions. Using Euler's formula (4.4) on Equation (4.1) the following development exists

$$exp(ikx + st) = exp(ikx)exp(st) = (cos(kx) + i\ sin(kx))exp(st). \tag{4.2}$$

For the exponential part depending on the variable s in Equation (4.2), once again as $s \in \mathbb{C}$ Euler's formula (4.4) can be used. It expresses that the behavior of the solution (4.1) depends only on $Re(s)$.

$$\begin{aligned} exp(st) &= exp((Re(s) + i\ Im(s))t) \\ &= (cos(Im(s)t) + i\ sin(Im(s)t))\ exp(Re(s)t) \end{aligned} \tag{4.3}$$

In Equation (4.3) if $Re(s) \leqslant 0$, the term $exp(Re(s)t)$ is a bounded function, otherwise its limit tends to infinity and Equation 4.1 characterizes a growth in time behavior.

Remark 14. The **wavelength** (denoted λ in Figure 4.1), is a parameter describing the spatial period of a periodic wave. It represents the distance over which the wave's shape repeats itself. The wavenumber k in Equation (4.1) is the spatial frequency of a wave, it represents the number of wavelengths per unit distance. Longer wavelengths characterize low frequencies while smaller wavelengths characterize high frequencies, see Figure 4.1.

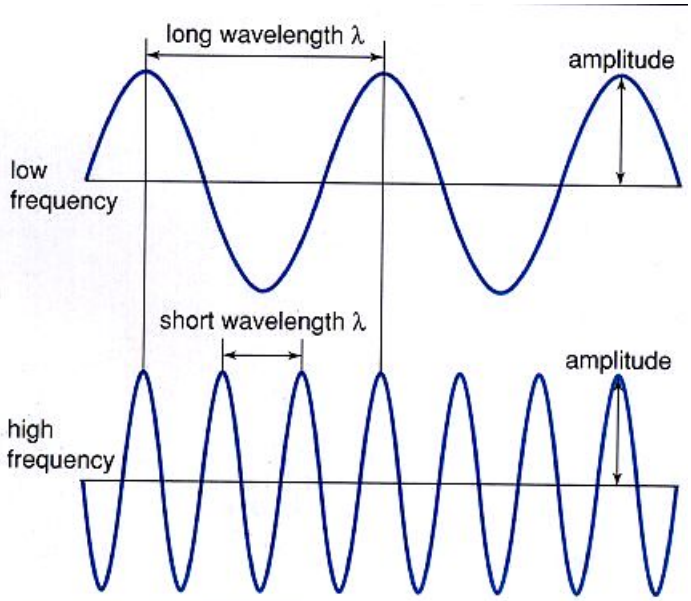


Figure 4.1: Example of wavelengths for a high and low frequency profile λ[15]

Remark 15. **Euler's formula** is a relationship describing the link between trigonometric functions and a complex exponential function. Let $x \in \mathbb{R}$ and i be the complex number. The Euler's formula states as below:

$$exp(ix) = cos(x) + i\ sin(x). \tag{4.4}$$

4.1.2 Linearization

To study the properties of a non-linear PDE, a linear approximation is a useful way to get information about the original problem. It is the first step in the study of stability presented in this chapter. The main idea of such approximation is to look at the first order Taylor expansion of the function around a chosen point of interest. Geometrically, a local linear approximation is often referred to as a tangent line approximation. If one zooms in on a point on a graph, the curve will look similar to a straight line. The tangent line which is the closest line to that point can then be used to approximate other values along the curve. The tangent line approximation of a function f at a point of interest a can be expressed as a function $L(x)$ defined by

$$L(x) = f(a) + \frac{\partial f}{\partial x}(a)(x - a)\ . \tag{4.5}$$

For non-linear PDEs, the first step involves finding a steady state $\boldsymbol{u_0}(\boldsymbol{x})$, called the **leading order solution** or **equilibrium state**. Let u be the solution of a PDE of the form

$$\frac{\partial u}{\partial t} = R(u, \frac{\partial u}{\partial x}, \frac{\partial^2 u}{\partial x^2}, ...)\ . \tag{4.6}$$

A steady state of (4.6) is a function $u_0(x)$, independent of the time variable t which satisfies

$$R(u_0, \frac{\partial u_0}{\partial x}, \frac{\partial^2 u_0}{\partial x^2}, ...) = 0\ . \tag{4.7}$$

[15]https://bramblechemistry.weebly.com/4b1-electromagnetic-radiation.html

The **linearization process** involves examining a solution $u(x,t)$ of the form

$$\boxed{u(x,t) = u_0(x) + \epsilon w(x,t)}, \tag{4.8}$$

where ϵ is taken as a **small parameter**. It can usually be expressed in terms of physical variables associated with the considered PDE.

Plugging a solution of the form (4.8) in Problem (4.6) gives the linearized version of the PDE, referred to as the **linearization** of (4.6) around u_0, where only the terms of order ϵ are kept from the development of (4.8) in (4.6). Non-linear PDEs often possess more than one steady state. It is important to denote that the linearization process described by (4.8) is not unique, and depends on the choice of the steady state u_0. The next section 4.1.3 presents an example of such PDE: the Fisher equation.

4.1.3 Application to an example: the Fisher Equation

The linearization process explained in Section 4.1.2 and the definition of the dispersion relation detailed in Section 4.1.1 are applied in this Section to a steady state of the Fisher Equation. The criteria of stability associated to the study of the dispersion relation will be expressed in the next section of this chapter.

Let u be the unknown of the single equation (4.9) defining the Fisher Equation

$$\frac{\partial u}{\partial t}(x,t) = \frac{\partial^2 u}{\partial x^2}(x,t) + u(x,t)(1-u(x,t)) \ ; \ -\infty < x < +\infty \ , \ t > 0, \tag{4.9}$$

and which satisfies unspecified initial and boundary conditions.

A linear solution of (4.9) of the form (4.10) is considered

$$u(x,t) = u_0(x) + \epsilon w(x,t). \tag{4.10}$$

A steady state solution of (4.9) denoted $u_0(x)$ is used to construct the linear solution (4.10). By plugging (4.10) into (4.9) one gets

$$\begin{aligned}
\epsilon\frac{\partial w}{\partial t}(x,t) &= \frac{\partial^2 u_0}{\partial x^2}(x,t) + \epsilon\frac{\partial^2 w}{\partial x^2}(x,t) + (u_0(x)+\epsilon w(x,t))(1-u_0(x)-\epsilon w(x,t)),\\
\epsilon\frac{\partial w}{\partial t}(x,t) &= \epsilon\frac{\partial^2 w}{\partial x^2}(x,t) + \epsilon w(x,t)(1-\epsilon w(x,t)) - 2u_0(x)\epsilon w(x,t),\\
\frac{\partial w}{\partial t}(x,t) &= \frac{\partial^2 w}{\partial x^2}(x,t) + w(x,t)(1-\epsilon w(x,t)) - 2u_0(x)w(x,t),\\
\implies & \boxed{\tfrac{\partial w}{\partial t}(x,t) = \tfrac{\partial^2 w}{\partial x^2}(x,t) + w(x,t)(1-2u_0(x))}.
\end{aligned} \tag{4.11}$$

Equation (4.11) is the linearization of (4.9) around $u_0(x)$.
From the linearized Fisher equation (4.11) a **dispersion relation** can be defined as expressed in Section 4.1.1. Let $w(x,t) = exp(ikx+st)$, where k is a constant coefficient. Then, plugging the definition of $w(x,t)$ into Equation (4.11) one obtains

$$\begin{aligned}
& s\ exp(ikx+st) = exp(ikx+st)(1-2u_0(x)) - k^2 exp(ikx+st),\\
\implies & s = 1-k^2-2u_0(x).
\end{aligned} \tag{4.12}$$

Equation (4.12) leads to the following dispersion relation

$$\boxed{s(k) = 1 - k^2 - 2u_0}\,, \tag{4.13}$$

where u_0 is a steady state of Equation (4.9). The dispersion relation is then a relationship that links s and k together through the choice of a steady state u_0.

4.1.4 Stability

The study of the sign of the real part of $s(k)$ (see reference [124]), denoted $Re(s(k))$ gives the necessary information to characterize the stability of the associated system of equations, as expressed in Section 4.1.1.

Four situations are possible:

1. $Re(s(k)) < 0\ \forall k$, then the system is called **stable**;
2. There exists some values of k for which $Re(s(k)) > 0$, then the system is called **unstable**;
3. If $Re(s(k)) < 0$ and $Re(s(k)) = 0$ for some k, which means that the maximum of $Re(s(k))$ is 0, the system is called **marginally stable**. This state is often referred as an intermediate state;
4. If $Re(s(k)) = 0\ \forall k$, i.e., that $s(k)$ is purely imaginary or is constant and equals to zero, **no conclusion** can be drawn regarding the stability of the considered steady state.

As previously mentioned, the result of the stability is dependent on the choice made for the steady state used during the linearization process. If an equation possesses more than one steady state, it is important to look at the results for all possible choices of u_0. To illustrate this argument, once again the Fisher equation is considered.

The Fisher equation (4.9) has two steady states denoted here u_0 and $\bar{u}_0$ such that $u_0 = 0$ and $\bar{u}_0 = 1$. For the study of the first steady state $u_0 = 0$ the linearization process in Section 4.1.2 with Equation (4.13) gives a dispersion relation of the form

$$s(k) = 1 - k^2. \tag{4.14}$$

The sign of Equation (4.14) depends on the choice made for k. It results that if $|k| < 1$ then $s(k) > 0,\ \forall k \in \mathbb{R}$, and the Fisher equation (4.9) is unstable around the steady state u_0 following criteria 2 in Section 4.1.4.

For the study of the second steady state $\bar{u}_0 = 1$, the results of the linearization process is different and following Definition (4.13) the dispersion relation around $\bar{u}_0$ is of the form

$$s(k) = -1 - k^2. \tag{4.15}$$

The dispersion relation (4.15) is negative for all values of k, and the Fisher equation is linearly stable around $\bar{u}_0$ following criteria 1 in Section 4.1.4.

In the next section, the Stefan model and its configuration for which the stability analysis is performed are described.

4.2 The Stefan model

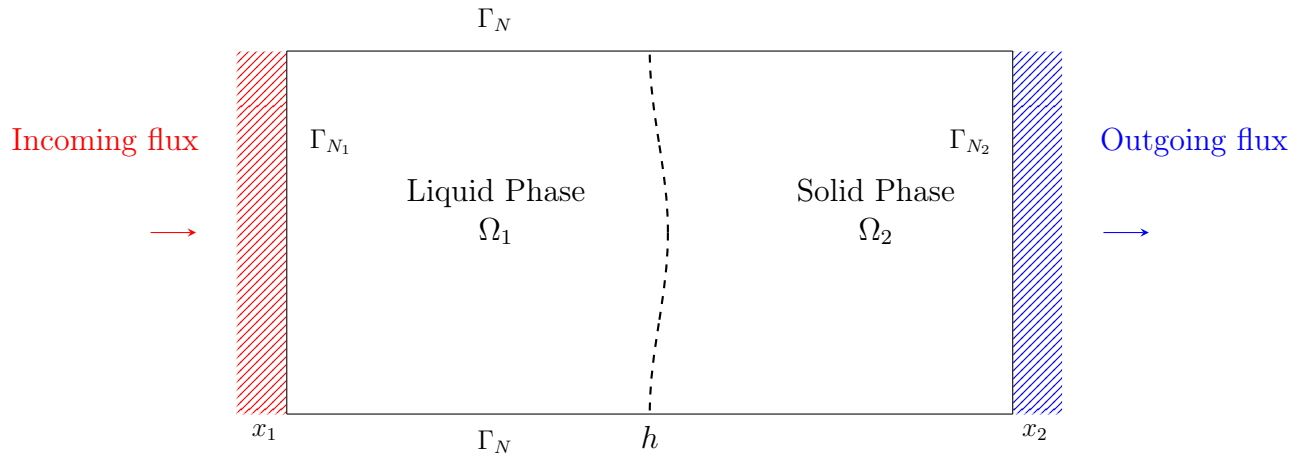


Figure 4.2: Domain of resolution for the stability analysis

The Stefan model regroups free and moving boundaries problems. In this Chapter, once again the mathematical model describing a phase change transition from solid water (ice) to liquid water is investigated. The model is composed of two sub-domains ($\Omega = \Omega_1 \cup \Omega_2$), one filled with water (Ω_1) and the other one filled with ice (Ω_2), see Figure 4.2. The material coexists at the interface region ($\Omega_1 \cap \Omega_2$) in both states, liquid and solid, see [4].

Since phase transition models are free boundary problems where the boundary position is also an unknown, to obtain a valid model a closure condition is required: the **Stefan condition**. In heat transfer problems with phase-change, the physical constraint and closure condition state that

1. Energy conservation at the interfaces and a jump in flux $\implies$ latent heat release,
2. The temperature is known at the interface: the melting temperature T_m.

These physical data will be used to define the mode in which the Stefan model is either stable or unstable. Numerically, the Stefan condition is discretized to obtain an approximation of the velocity of the nodes defining the physical interface. In this Chapter, the physical interface is defined as a set of points to be moved. In order to match the numerical model defined in Chapter 2 and implemented for the work presented in this manuscript in a $2D$ code, the stability analysis on the continuous model is presented in two dimensions (x, y) as well. Moreover, the stable motion of the interface is considered uni-dimensional. This configuration allows the study of a perturbation at the interface in a reference frame where any instability is easily identifiable.

In the next section, the model used for the analysis is described. Some information on the Stefan model are also recalled, for more details on the Stefan model see Chapter 2.

A rectangular domain of length $[-L, L]$ is considered (see Figure 4.2), denoted as Ω, where $x_1 = -L$ and $x_2 = L$. The initial interface position is denoted by h_0. For the stability analysis the classical formulation of the Stefan model is considered, i.e., its **primal formulation**. The mixed form and primal form are equivalent on a continuous level for the study of stability of the PDEs associated to the Stefan model. On the other hand the mixed formulation will be studied for the numerical analysis, see Section 4.7.

Find a **temperature field** $T(x, y, t)$ and an **interface location** $h(y, t)$ satisfying :

$$\rho c_j \frac{\partial T(x,y,t)}{\partial t} + \frac{\partial}{\partial x}\left(-\lambda_j \frac{\partial T(x,y,t)}{\partial x}\right) + \frac{\partial}{\partial y}\left(-\lambda_j \frac{\partial T(x,y,t)}{\partial y}\right) = 0. \tag{4.16}$$

In the following, the index j will always refer to a component phase-depend, $j = 1$ for the liquid phase, $j = 2$ for the solid phase, such that λ_j represents the thermal conductivity of the material and c_j the heat capacity for their respective phase. For simplicity the **thermal diffusivity** $\alpha_j = \frac{\lambda_j}{\rho c_j}$ is also defined. The parameter ρ in Equation (4.16) is the density of the material, and is considered equal in both the liquid and solid water phases. This is a common assumption for ice melting situations, see [4].

Equation (4.16) states that the temperature diffuses inside the domain, while the flux jump at the interface drives the boundary motion. The position of the moving boundary is a time dependent function and needs to be determined as a part of the solution. Then, the resolution of the PDE requires to solve the heat equation in all of the phases, while the moving boundary is determined by the Stefan condition, through the determination of the flux jump at the interface. For the analysis of the Stefan model, the Stefan condition is also considered in its primal form and the following definition is used

$$[\![-\lambda \nabla T]\!]_h \cdot \boldsymbol{n} = \rho L_m \frac{\partial h(y,t)}{\partial t}. \tag{4.17}$$

The definition of the interface in this analysis is defined such that any point (x, y) at the physical interface corresponds to coordinates of the form $(x, y) = (h(y, t), y)$. It leads that any point at the interface has a velocity v defined by:

$$v = \frac{dx}{dt}\vec{i} + 0\vec{j} = \frac{dh(y,t)}{dt}\vec{i} + 0\vec{j}. \tag{4.18}$$

The unit vectors $(\vec{i}, \vec{j})$ characterizes the cartesian coordinate system along the x-axis and the y-axis. Definition (4.18) defines a configuration in which the interface motion is uni-dimensional which simplifies the problem for the stability analysis.

Equation (4.16) is the standard primal formulation of the Stefan model defined by the heat equation in both phases of the domain, see Figure 4.2. In Equation (4.16), only a **heat diffusion term** (diffusive flux) is present for the moment, i.e. with the presence of

$$\nabla \cdot (-\lambda_j \nabla T) = \frac{\partial}{\partial x}\left(-\lambda_j \frac{\partial T(x,y,t)}{\partial x}\right) + \frac{\partial}{\partial y}\left(-\lambda_j \frac{\partial T(x,y,t)}{\partial y}\right).$$

Nevertheless, the configuration in which the stability analysis is performed requires imposing the movement in which the physical interface is moving. It can be interpreted as a change in reference frame of the problem in which the problem is situated.

1. Over time the interface is moving in such a way that the solid (ice) is receding from one of the boundaries of the domain in the x-direction only, see Figure 4.2. This specific configuration allows to study the behavior of a perturbation at the interface in a configuration where its displacement should be uniform and in one direction. More precisely, its behavior is compared to a reference displacement, i.e., whether the perturbation flattens or not through the simulation.

2. The initial interface position at time $t = 0$ is defined by h_0 (non perturbed interface), where $\forall y \in \Omega$ the coordinate of a point at the interface h_0 is (x_0, y), i.e., that h_0 is a flat

interface, and with x_0 a chosen position.

3. A convective flux term in the x direction is added to formulation (4.16). This term comes from a change in the reference frame where the solution is studied. It allows the introduction of the velocity of the solid u that is receding from the domain at the boundary $x = x_2$, see Figure 4.2.

   Remark 16. To illustrate the introduction of the convective flux term in Equation (4.16) a simplified problem (some coefficients are taken equal to 1) is chosen to detail the procedure. First, the heat equation is recalled for a two dimensional problem of unknown T

$$\frac{\partial T(x,y,t)}{\partial t} = \alpha \frac{\partial^2 T(x,y,t)}{\partial x^2} + \alpha \frac{\partial^2 T(x,y,t)}{\partial y^2}. \tag{4.19}$$

   In order to introduce a convective term in the x direction in Equation (4.19), a change in coordinate is performed. The time variable t remains invariant in the new coordinate system.
   Let

$$\begin{aligned} s_x &= \sqrt{2}x - 2ut, \\ s_y &= \sqrt{2}y, \end{aligned} \tag{4.20}$$

   where u is the velocity of the solid pulled away. The transformation (4.20) represents a characteristic curve in the x direction along which the temperature remains constant. Transformation (4.20) allows to modify the terms of Equation (4.19) as follow

$$\begin{aligned} &1.\ \frac{\partial T}{\partial t} = \frac{\partial T}{\partial s_x}\frac{\partial s_x}{\partial t} + \frac{\partial T}{\partial s_y}\frac{\partial s_y}{\partial t} = -2u\frac{\partial T}{\partial s_x}, \\ &2.\ \alpha\frac{\partial^2 T}{\partial x^2} = \alpha\frac{\partial}{\partial x}\left(\frac{\partial T}{\partial s_x}\frac{\partial s_x}{\partial x}\right) + \alpha\frac{\partial}{\partial x}\left(\frac{\partial T}{\partial s_y}\frac{\partial s_y}{\partial x}\right) = \alpha\frac{\partial}{\partial s_x}\frac{\partial s_x}{\partial x}\left(\frac{\partial T}{\partial s_x}\frac{\partial s_x}{\partial x}\right) = 2\alpha\frac{\partial^2 T}{\partial s_x^2}, \\ &3.\ \alpha\frac{\partial^2 T}{\partial y^2} = \alpha\frac{\partial}{\partial y}\left(\frac{\partial T}{\partial s_x}\frac{\partial s_x}{\partial y}\right) + \alpha\frac{\partial}{\partial y}\left(\frac{\partial T}{\partial s_y}\frac{\partial s_y}{\partial y}\right) = \alpha\frac{\partial}{\partial s_y}\frac{\partial s_y}{\partial y}\left(\frac{\partial T}{\partial s_y}\frac{\partial s_y}{\partial y}\right) = 2\alpha\frac{\partial^2 T}{\partial s_y^2}. \end{aligned} \tag{4.21}$$

   Using the results from the change of variables (4.21) the following equations hold

$$\begin{aligned} &1.\ \frac{\partial T}{\partial t} = -2u\frac{\partial T}{\partial s_x}, \\ &2.\ \frac{\partial T}{\partial t} = 2\alpha\frac{\partial^2 T}{\partial s_x^2} + 2\alpha\frac{\partial^2 T}{\partial s_y^2}, \end{aligned} \tag{4.22}$$

   and the heat equation (4.19) can be rewritten by

$$\begin{aligned} 2\frac{\partial T}{\partial t} &= -2u\frac{\partial T}{\partial s_x} + 2\alpha\frac{\partial^2 T}{\partial s_x^2} + 2\alpha\frac{\partial^2 T}{\partial s_y^2}. \\ \Longrightarrow \frac{\partial T}{\partial t} &= -u\frac{\partial T}{\partial s_x} + \alpha\frac{\partial^2 T}{\partial s_x^2} + \alpha\frac{\partial^2 T}{\partial s_y^2}. \end{aligned} \tag{4.23}$$

   Equation (4.23) can be written using different notations where (s_x, s_y) and (x, y) can be used instead, and the convective flux term has been added to the initial formulation (4.19).

Remark 17. In the case of a stable model a perturbation at the interface should flatten and go back to its initial interface shape.

The **problem at stake** for the **stability analysis** is the following :

Find a **temperature field** $T(x,y,t)$ and an **interface location** $h(y,t)$ satisfying :

$$\rho c_j \frac{\partial T(x,y,t)}{\partial t} + \frac{\partial}{\partial x}\left(\rho c_j u T(x,y,t) - \lambda_j \frac{\partial T(x,y,t)}{\partial x} \right) + \frac{\partial}{\partial y}\left(-\lambda_j \frac{\partial T(x,y,t)}{\partial y} \right) = 0. \quad (4.24)$$

Equation (4.24) is associated with the following boundary conditions:

$$\begin{aligned}
& f_1 && = -\lambda_1 \nabla T(x_1,y,t)\cdot n + \rho c_1 u T(x_1,y,t) && \text{on } \Gamma_{N_1}, \\
& f_2 && = -\lambda_2 \nabla T(x_2,y,t)\cdot n + \rho c_2 u T(x_2,y,t) && \text{on } \Gamma_{N_2}, \\
& -\lambda_j \nabla T(x,y,t)\cdot n && = 0 && \text{on } \Gamma_N\ , \\
& T(h(y,t),y,t) && = T_m && \text{on } h(y,t)\ , \\
& [\![-\lambda_j \nabla T(h(y,t),y,t)]\!]_h \cdot n && = \rho L_m u_g && \text{on } h(y,t)\ ,
\end{aligned} \quad (4.25)$$

with $u_g = ((\frac{\partial h(y,t)}{\partial t} - u)\vec{i} + 0\ \vec{j})\cdot n$ within the cartesian coordinate system, see Figure 4.3 for a visualization of the boundary conditions on the domain of resolution Ω.

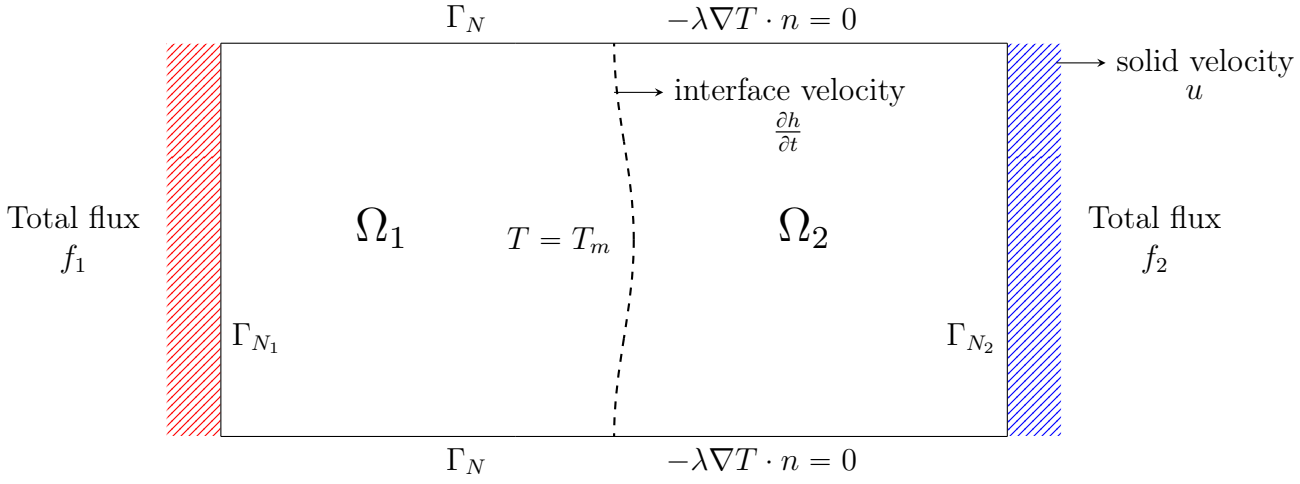


Figure 4.3: Visualization of the boundary conditions (4.25) for the stability analysis

The constant variable u in Equation (4.24) and Conditions (4.25) represents the **velocity of the solid pulled away** (from the addition of the convective flux term in Equation (4.24)), while $\frac{\partial h(y,t)}{\partial t}$ is **the velocity associated to the displacement of the physical interface**.

The variables f_1 and f_2 in (4.25) represent the **total flux** for their respective phases on the boundaries Γ_{N_1} and Γ_{N_2}, see Figures 4.2 and 4.3. The term **total flux** characterizes the amount of heat passing through a unit surface area per unit time. It is defined as the sum of the **convective flux** (mass transfer due to the bulk motion of a fluid) and the **diffusive flux** (flow of matter from a region of high concentration to a region of low concentration).
In Equation (4.24) the **diffusive flux** is defined by the terms

$$\frac{\partial}{\partial x}(-\lambda_j \frac{\partial T(x,y,t)}{\partial x}) + \frac{\partial}{\partial y}(-\lambda_j \frac{\partial T(x,y,t)}{\partial y}),$$

while the **convective flux** is present only in the x direction

$$\rho c_j u \frac{\partial T(x,y,t)}{\partial x}.$$

Remark 18. The definition of total flux often refers to the sum of all the fluxes associated with the considered system. More generally, it represents the cumulative transfer or flow of any quantity through a region or boundary within the domain of resolution. It can account for various types of flux, depending on the nature of the problem at hand.

The solid and the liquid phases are assumed to be **incompressible**, with equal and constant density ρ, T_m being the melting temperature and L_m the latent heat.

For the stability analysis of Equation (4.24) the first step consists of defining the temperature field T and the interface position h as **linear solutions**, of a linear system of PDE, see Section 4.1.2. The linearization of the temperature field is defined around a steady state $T_{j,0}(x)$ that needs to be identified. The same procedure is applied for the interface position $h(y,t)$ for a steady state denoted h_0, chosen at the initial time $t = 0$, corresponding to the initial interface position.

In the next section, the identification of the steady state $T_{j,0}$ is the subject of discussion. As the temperature is phase dependent, the steady state will also be defined independently for each phase.

The **linear solutions** examined in the following sections are of the form

$$\begin{aligned} T_j(x,y,t) &= T_{j,0}(x) + \epsilon T_{j,1}(x,y,t), \\ x := h(y,t) &= h_0 + \epsilon h_1(y,t), \end{aligned} \tag{4.26}$$

where the index j refers to the corresponding phase of the solution and belongs to either Ω_1 or Ω_2, see Figure 4.2.

4.3 Leading Order Solution

$T_{j,0}(x)$ as defined in Equation (4.26) is a steady state solution of Equation (4.24) and satisfies

$$\frac{\partial T_{j,0}(x)}{\partial t} = 0. \tag{4.27}$$

The definition of $T_{j,0}(x)$ expressed in (4.27) and the definition of the problem at stake in Equation (4.24) lead to

$$u\frac{\partial T_{j,0}}{\partial x} - \alpha_j \frac{\partial^2 T_{j,0}}{\partial x^2} = 0 \ . \tag{4.28}$$

To define $T_{j,0}(x)$ the next step consists in the resolution of Problem (4.28). To generalize the approach expressed in this section the resolution of (4.28) is detailed thoughtfully.

To prevent the division by $\frac{\partial T_{j,0}}{\partial x}$ in the resolution of (4.28), which without further information could be null for some points of the domain, the ODE (4.28) is solved by looking at the associated second order equation (4.29) of unknown X

$$uX - \alpha_j X^2 = 0 \ , \tag{4.29}$$

which has the associated phase-dependent discriminant number Δ_j

$$\Delta_j = u^2 - 4 \times (-\alpha_j) \times 0 = u^2 \geqslant 0. \tag{4.30}$$

The sign of the discriminant number (4.30) gives the information on the number of square roots associated to the resolution of the ODE (4.29). Let $r_{j,1}$ and $r_{j,2}$ be the phase dependent

square-roots associated to Problem (4.29). Then they are defined by

$$\begin{aligned} r_{j,1} &= \frac{-u-\sqrt{\Delta_j}}{-2\alpha_j} = \frac{u}{\alpha_j}, \\ r_{j,2} &= \frac{-u+\sqrt{\Delta_j}}{-2\alpha_j} = 0. \end{aligned} \tag{4.31}$$

By assuming that the solution of (4.29) is of the form $A_j e^{r_{j,1}x} + B_j e^{r_{j,2}x}$, with $A_j, B_j \in \mathbb{R}$, phase dependent coefficients, one can write $T_{j,0}(x)$ the solution of (4.28) as follows

$$T_{j,0}(x) = A_j e^{\frac{ux}{\alpha_j}} + B_j, \quad A_j, B_j \in \mathbb{R}. \tag{4.32}$$

The four constants A_1, A_2, B_1, B_2 are specified using the boundary conditions (4.25). For the configuration in which the problem has been defined, see Figures 4.2 and 4.3, the heat flows from the boundary of the liquid phase in a single direction (x direction) and exits through the boundary of the solid phase in the same direction. It allows to consider that the total flux is constant along the y-axis in both phases. This information allows to look at the value of the total flux at the interface for both phases for the definition of the coefficients A_j. From Conditions (4.25) the total flux f_j can be expressed at the interface h_0 by

$$f_j = -\lambda_j \nabla T(h_0, y, t) \cdot n + \rho c_j u T(h_0, y, t). \tag{4.33}$$

Let $T_{j,0}(x)$ satisfies condition (4.33) then

$$\begin{aligned} &-\lambda_j \frac{\partial T_{j,0}(h_0)}{\partial x} + \rho c_j u T_{j,0}(h_0) = f_j, \\ &-\lambda_j \frac{u}{\alpha_j} A_j e^{\frac{uh_0}{\alpha_j}} + \rho c_j u T_m = f_j, \\ &A_j = -(f_j - \rho c_j u T_m) e^{-\frac{uh_0}{\alpha_j}} \times \frac{\alpha_j}{u\lambda_j}, \\ \Longrightarrow &\boxed{A_j = -q_j \tfrac{\alpha_j}{u\lambda_j} e^{\frac{-uh_0}{\alpha_j}}}, \end{aligned} \tag{4.34}$$

with $q_j = (f_j - \rho c_j u T_m)$. The definition of coefficient A_j in (4.34) leads for the moment to the following definition of $T_{j,0}(x)$

$$T_{j,0}(x) = -q_j \frac{\alpha_j}{u\lambda_j} e^{\frac{u(x-h_0)}{\alpha_j}} + B_j, \tag{4.35}$$

where B_j needs to be determined.
For the definition of the coefficient B_j in (4.35), $T_{j,0}(h_0)$ is defined such that the condition of melting at the interface is satisfied. Using Definition (4.35) one gets

$$T_{j,0}(h_0) = -q_j \frac{\alpha_j}{u\lambda_j} + B_j, \tag{4.36}$$

and from Condition (4.25)

$$T_{j,0}(h_0) = T_m. \tag{4.37}$$

Combining (4.35) and (4.36) together

$$B_j = T_m + q_j \frac{\alpha_j}{u\lambda_j}. \tag{4.38}$$

Finally the leading order solution is defined by

$$T_{j,0}(x) = \frac{\alpha_j}{u\lambda_j} q_j \left(1 - e^{u\frac{(x-h_0)}{\alpha_j}}\right) + T_m \ . \tag{4.39}$$

However, a last step is necessary to get the value of $T_{j,0}(x)$: the definition of the **solid velocity** $\boldsymbol{u}$. The last condition in (4.25) which has not been used yet is the Stefan condition, recalled below:

$$[\![-\lambda_j \nabla T(h(y,t),y,t)]\!]_h \cdot n = \rho L_m u_g \tag{4.40}$$

For the steady state solution, as the function $T_{j,0}(x)$ is independent of the time variable, the coefficient $u_g = (\frac{\partial h(y,t)}{\partial t} - u)\vec{i} + 0\ \vec{j}) \cdot n$ in Conditions (4.40) can be simplified into $u_g = -u$, and allows to define u as

$$\begin{aligned}
& -\lambda_1 \frac{\partial T_{1,0}}{\partial x}(h_0) - \left(-\lambda_2 \frac{\partial T_{2,0}}{\partial x}(h_0)\right) && = -\rho L_m u, \\
\iff & -\lambda_1 \frac{\alpha_1}{u\lambda_1} q_1 \left(-\frac{u}{\alpha_1}\right) + \lambda_2 \frac{\alpha_2}{u\lambda_2} q_2 \left(-\frac{u}{\alpha_2}\right) && = -\rho L_m u, \\
\iff & (f_1 - \rho c_1 u T_m) - (f_2 - \rho c_2 u T_m) && = -\rho L_m u, \\
\implies & \boxed{u = \tfrac{f_1 - f_2}{\rho(T_m(c_1 - c_2) - L_m)}}, &&
\end{aligned} \tag{4.41}$$

where $q_j = (f_j - \rho c_j u T_m)$.
In the case of a melting simulation, the latent heat Lm is positive, as **heat is added in the system** through the liquid phase. The specific heat satisfies $c_1 > c_2$, and the melting temperature are known values, and the total flux is defined such that $f_1 > f_2$. This specificity makes the solid velocity u being a negative variable. For a solidification problem the sign of the velocity is opposite.

4.4 First Order Solution

To obtain the first order solution in the development around the steady state $T_{j,0}(x)$, the solution $T_{j,1}(x,y,t)$ and the interface location $h(x,y,t)$ are assumed to be of the form

$$\begin{aligned}
T_{j,1}(x,y,t) &= T_{j,1}(x) exp(iky + st), \\
h_1(y,t) &= h_1 exp(iky + st),
\end{aligned} \tag{4.42}$$

where $\boldsymbol{k}$ defines a **perturbation wavenumber**, and $\boldsymbol{s}$ a prediction of the **temporal behavior of the system**, which is determined as a function of k, see Section 4.1.1.

The first step in the determination of the first order solution is to obtain an equation in terms of $T_{j,1}(x)$. To achieve this, Definition (4.26) is inserted in Equation (4.24) to obtain the linearized form of Equation (4.24), leading to

$$\rho c_j \frac{\partial T_j}{\partial t}(x,y,t) + \frac{\partial}{\partial x}\left(\rho c_j u T_j(x,y,t) - \lambda_j \frac{\partial T_j}{\partial x}(x,y,t)\right) + \frac{\partial}{\partial y}\left(-\lambda_j \frac{\partial T_j}{\partial y}(x,y,t)\right) = 0. \tag{4.43}$$

Then, the development of $T_j(x,y,t)$ defined in Equation (4.26) can be done to make $T_{j,0}(x)$, the steady state disappear in Equation (4.43). Then, one obtains the following Equation depending on $T_{j,1}(x,y,t)$ only

$$\rho c_j \frac{\partial T_{j,1}}{\partial t}(x,y,t) + \frac{\partial}{\partial x}\left(\rho c_j u T_{j,1}(x,y,t) - \lambda_j \frac{\partial T_{j,1}}{\partial x}(x,y,t)\right) + \frac{\partial}{\partial y}\left(-\lambda_j \frac{\partial T_{j,1}}{\partial y}(x,y,t)\right) = 0, \quad (4.44)$$

which can be developed using the definition of $T_{j,1}(x,y,t)$ in Equation (4.42)

$$\begin{aligned} &\rho c_j T_{j,1}(x) s e^{iky+st} + \rho c_j u \frac{\partial T_{j,1}}{\partial x}(x) e^{iky+st} - \lambda_j \frac{\partial^2 T_{j,1}}{\partial x^2}(x) e^{iky+st} - \lambda_j (ik)^2 T_{j,1}(x) e^{iky+st} = 0, \\ \Longrightarrow &\boxed{T_{j,1}(x)(s+\alpha_j k^2) + u\frac{\partial T_{j,1}(x)}{\partial x} - \alpha_j \frac{\partial^2 T_{j,1}(x)}{\partial x^2} = 0}. \end{aligned} \quad (4.45)$$

Before being able to express a definition for $T_{j,1}(x)$, the conditions satisfied by $T_{j,1}(x)$ need to be expressed. This step is referred to as the **linearization of the boundary conditions**. Some of the conditions of the problem at stake (4.24) are already satisfied by the leading order solution $T_{j,0}(x)$. The next part of this section addresses the definition of the conditions satisfied by $T_{j,1}(x,y,t)$, and the construction of a solution $T_j(x)$ (see Equation (4.26)) conformed with the initial system of equations (4.24) and its conditions (4.25).

First, the conditions that the leading order solution $T_{j,0}(x)$ satisfies are recalled with the following system of equations (4.46)

$$\begin{aligned} -\lambda_j \frac{\partial T_{j,0}(h_0)}{\partial x} \cdot n_x &= f_j - \rho c_j u T(h_0), \\ T_{j,0}(h_0) &= T_m, \\ [\![-\lambda_j \frac{\partial T_{j,0}(h_0)}{\partial x}]\!] &= -\rho L_m u. \end{aligned} \quad (4.46)$$

where h_0 is the initial interface position. The Neumann conditions on Γ_N are satisfied naturally as $T_{j,0}(x)$ is independent of the coordinate y.

From the original system of conditions (4.25) and the conditions satisfied by $T_{j,0}(x)$ in (4.46) one can easily define that $T_{j,1}(x)$ satisfies

$$\begin{aligned} \rho c_j u T_{j,1}(x_j) - \lambda_j \frac{\partial T_{j,1}}{\partial x}(x_j) &= 0, \\ T_{j,1}(h_0) &= \frac{h_1}{\lambda_j}(f_j - \rho c_j u T_m), \end{aligned} \quad (4.47)$$

where $x_1 = -L$ and $x_2 = L$, see Figure 4.2.
The first condition in (4.47) is trivial using that the boundary condition is already satisfied by $T_{j,0}$ and that the total flux f_j (sum of the convective and diffusive flux) is constant through each of its respective phases, see conditions (4.46).

For the second condition in (4.47), the approach consists of doing a Taylor development around the initial interface position h_0 and to use the information known about $T_{j,0}(h_0)$ to get a definition of $T_{j,1}(h_0)$. One can find that at the interface $h(y,t) = h_0 + \epsilon h_1(y,t)$ the function $T_j(x,y,t)$ satisfies

$$T_j(h(y,t),y,t) = T_j(h_0 + \epsilon h_1(y,t), y, t) \simeq T_j(h_0,y,t) + \epsilon h_1(y,t)\frac{\partial T_j(h_0,y,t)}{\partial x} = T_m, \quad (4.48)$$

which leads to

$$T_j(h_0+\epsilon h_1,y,t)\simeq T_{j,0}(h_0)+\epsilon T_{j,1}(h_0,y,t)+\epsilon h_1\frac{\partial T_{j,0}}{\partial x}(h_0)+\epsilon^2 h_1\frac{\partial T_{j,1}}{\partial x}(h_0,y,t)=T_m\ . \quad (4.49)$$

By conserving only the terms of order ϵ in Equation (4.49) one finally gets

$$\begin{aligned} &T_{j,1}(h_0,y,t)+h_1\frac{\partial T_{j,0}}{\partial x}(h_0)=T_m-T_{j,0}(h_0)=0,\\ \Longrightarrow\ &\boxed{T_{j,1}(h_0,y,t)=\tfrac{h_1}{\lambda_j}(f_j-\rho c_j u T_m)=\tfrac{h_1 q_j}{\lambda_j}}. \end{aligned}$$

The last condition to work with is the Stefan condition, which requires to modify the definition of the normal n, oriented from the liquid to the solid phase, see Figure 4.2. To achieve this, a change of variable is performed to get a decomposition of the normal at the interface depending on the interface velocity, where it is recalled that its motion is restricted to the horizontal direction only (convective term in the x-direction). The change of variable presented in this section will **only be used for the decomposition of the interface normal**. This procedure is used to separate the space variable (x,y) in the definition of the Stefan condition.

One can describe the interface position as a function depending on **curvilinear coordinates** (ζ,t), t being invariant in the new coordinates system.
Let x_s be defined by

$$x_s(\zeta,t)=(x_0+\epsilon x_1(\zeta,t),\zeta\Delta y), \quad (4.50)$$

where Δy represents the mesh-discretization in the y-direction, x_0 an initial position and $\epsilon x_1(\zeta,t)$ a small displacement from x_0.

One can define

$$r(\zeta)=(x_0+\epsilon x_1(\zeta,t))\vec{i}+\zeta\Delta y\vec{j}, \quad (4.51)$$

such that

$$r'(\zeta)=\epsilon\frac{\partial x_1}{\partial\zeta}\vec{i}+\Delta y\vec{j}, \quad (4.52)$$

with $||r'||=\sqrt{\Delta y^2+\epsilon^2\frac{\partial x_1}{\partial\zeta}^2}$.
Then, the tangent vector $\vec{\tau}$ associated to $r(\zeta)$ is defined by

$$\vec{\tau}=\frac{\epsilon\frac{\partial x_1}{\partial\zeta}\vec{i}+\Delta y\vec{j}}{||r'||}. \quad (4.53)$$

The definition of the tangential vector τ in (4.53) allows to deduct the normal vector

$$\vec{n}=\frac{\Delta y\vec{i}-\epsilon\frac{\partial x_1}{\partial\zeta}\vec{j}}{||r'||}\ , \quad (4.54)$$

where the sign of the normal verifies the convention defined previously from liquid to solid phase.

The interface velocity $\frac{\partial h(y,t)}{\partial t}$ can be defined using the temporal derivative of $x_s(\zeta,t)$ defined in (4.50) such that

$$\frac{\partial h}{\partial t}(y,t)=\frac{\partial x_s(\zeta,t)}{\partial t}=\epsilon\frac{\partial x_1(\zeta,t)}{\partial t}\vec{i}+0\vec{j}, \quad (4.55)$$

Definition (4.55) can be used to characterize the velocity u_g defined in conditions (4.25), which takes into account both the interface velocity $\frac{\partial h(y,t)}{\partial t}$ and the velocity of the ice pulled away u.

Then u_g is defined by

$$u_g = \left(\left(\frac{\partial x_s}{\partial t} - u \right) \vec{i} + 0\vec{j} \right) \cdot n \ . \tag{4.56}$$

The velocity of the solid u can be decomposed as $u = u_1 n_x + u_2 n_y$, where u_1 is the **horizontal velocity** and u_2 the **vertical velocity**.

Following the conditions of the original system (4.24), $T_j(x, y, t)$ (where $T_j(x, y, t) := T_{j,0}(x) + \epsilon T_{j,1}(x, y, t)$) needs to satisfy

$$\begin{aligned}
&[\![-\lambda_j \nabla T_j(h_0 + \epsilon h_1(y,t), y, t)]\!] \cdot n = \rho L_m u_g, \\
\implies &[\![-\lambda_j \nabla T_j(h_0, y, t)]\!] \cdot n + \epsilon h_1(y,t) \frac{\partial}{\partial x} \left([\![-\lambda_j \nabla T_j(h_0, y, t)]\!] \cdot n \right) = \rho L_m u_g, \\
\implies &\left[\!\!\left[-\lambda_j \frac{\partial T_{j,1}(h_0, y, t)}{\partial x} \right]\!\!\right] \frac{\Delta y}{||r'||} + h_1(y,t) [\![-\lambda_j \partial^2_{x^2} T_{j,0}(h_0)]\!] \frac{\Delta y}{||r'||} = \rho L_m \frac{\partial x_1}{\partial t} \frac{\Delta y}{||r'||}, \\
\implies &\left[\!\!\left[-\lambda_j \frac{\partial T_{j,1}}{\partial x}(h_0) \right]\!\!\right] = exp(-(iky + st)) \left(\rho L_m \frac{\partial x_1}{\partial t} - h_1(y,t) \left[\!\!\left[\frac{u}{\alpha_j} (f_j - \rho c_j u T_m) \right]\!\!\right] \right).
\end{aligned} \tag{4.57}$$

In (4.42) the interface location is defined by $h(y,t) = h_0 + \epsilon h_1 \ e^{iky+st}$. By identification with (4.57) the interface velocity $\frac{\partial x_1}{\partial t}$ is associated with the time derivative of $h(y,t)$, and the following equation holds

$$\frac{\partial x_1}{\partial t}(\zeta, t) = s h_1 e^{iky+st}. \tag{4.58}$$

Then, Equation (4.57) can be simplified into

$$\boxed{\left[\!\!\left[-\lambda_j \tfrac{\partial T_{j,1}}{\partial x}(h_0) \right]\!\!\right] = -h_1([\![\tfrac{u}{\alpha_j}(f_j - \rho c_j u T_m)]\!] - \rho L_m s)} \ . \tag{4.59}$$

Finally, the conditions that $T_{j,1}(x)$ needs to satisfy are the following

$$\begin{aligned}
&1.\ \rho c_j u T_{j,1}(x_j) - \lambda_j \frac{\partial T_{j,1}}{\partial x}(x_j) &&= 0, \\
&2.\ T_{j,1}(h_0) &&= \frac{h_1}{\lambda_j}(f_j - \rho c_j u T_m), \\
&3.\ \left[\!\!\left[-\lambda_j \frac{\partial T_{j,1}}{\partial x}(h_0) \right]\!\!\right] &&= -h_1([\![\frac{u}{\alpha_j}(f_j - \rho c_j u T_m)]\!] - \rho L_m s),
\end{aligned} \tag{4.60}$$

where h_1 is an unknown to be determined.

In the next section, using the ODE (4.45) and the associated boundary conditions (4.60) a definition of $T_{j,1}(x, y, t)$ is expressed. From the definition of $T_{j,1}(x)$ a **dispersion relation** is defined. Its definition is detailed in the next section.

4.5 Dispersion Relation

To simplify the definition of the dispersion relation of Problem (4.24) calculated in this section, the initial interface position is now defined at $h_0 = 0$. This procedure is always possible by the use of an appropriate change of variable, and will significantly simplify the definition of the dispersion relation presented in this section. This choice was not proposed earlier in the analysis to keep most of the generalities and provide an approach more easily adaptable to different applications and configurations.

From Conditions (4.60) and the ODE (4.45) it is assumed that $T_{j,1}(x)$ is of the form

$$T_{j,1}(x) = exp\left(\frac{ux}{2\alpha_j}\right)(A_j cosh(L_j x) + B_j sinh(L_j x)), \tag{4.61}$$

corresponding to a solution of an ODE using the hyperbolic trigonometric functions *cosh* defined in (4.62) and *sinh* defined in (4.63).

Remark 19. The hyperbolic cosine function $cosh(\cdot)$ is defined by

$$cosh(x) = \frac{e^x + e^{-x}}{2}. \tag{4.62}$$

The hyperbolic sine function $sinh(\cdot)$ is defined by

$$sinh(x) = \frac{e^x - e^{-x}}{2}. \tag{4.63}$$

The coefficients A_j and B_j in (4.61) are phase dependent constants which need to be determined. The phase-dependent variable L_j in (4.61) is defined by

$$L_j = \frac{\sqrt{\tilde{\Delta}_j}}{-2\alpha_j}, \tag{4.64}$$

and associated to the discriminant number $\tilde{\Delta}_j$ of the ODE (4.45), also phase-dependent

$$\tilde{\Delta}_j = u^2 + 4\alpha_j(s + \alpha_j k^2) \quad . \tag{4.65}$$

From the development of (4.61) in the first condition of (4.60) one gets

$$\begin{aligned} &\rho c_j u\ exp\left(\frac{ux_j}{2\alpha_j}\right)(A_j cosh(L_j x_j) + B_j sinh(L_j x_j)) - \lambda_j \frac{u}{2\alpha_j} exp\left(\frac{ux_j}{2\alpha_j}\right)(A_j cosh(L_j x_j) + B_j sinh(L_j x_j)) \\ &- \lambda_j exp\left(\frac{ux_j}{2\alpha_j}\right)(A_j L_j sinh(L_j x_j) + B_j L_j cosh(L_j x_j)) = 0, \\ &\implies B_j = A_j \frac{\alpha_j L_j sinh(L_j x_j) - \frac{u}{2} cosh(L_j x_j)}{\frac{u}{2} sinh(L_j x_j) - \alpha_j L_j cosh(L_j x_j)} = A_j c_{s,j}, \end{aligned} \tag{4.66}$$

where $c_{s,j}$ is defined by

$$c_{s,j} = \frac{\alpha_j L_j sinh(L_j x_j) - \frac{u}{2} cosh(L_j x_j)}{\frac{u}{2} sinh(L_j x_j) - \alpha_j L_j cosh(L_j x_j)}. \tag{4.67}$$

Knowing the value of the temperature at $h_0 = 0$ from the second condition of (4.60) recalled below

$$T_{j,1}(h_0) = \frac{h_1}{\lambda_j}(f_j - \rho c_j u T_m), \tag{4.68}$$

one gets

$$exp\left(\frac{0 \times u}{2\alpha_j}\right)(A_j cosh(L_j \times 0) + B_j sinh(L_j \times 0)) = \frac{h_1}{\lambda_j}(f_j - \rho c_j u T_m), \tag{4.69}$$

which leads to

$$A_j = \frac{h_1}{\lambda_j}(f_j - \rho c_j u T_m) = \frac{h_1}{\lambda_j} q_j. \tag{4.70}$$

Finally, from the Stefan condition in (4.60) (third condition) and the initial interface position at $h_0 = 0$, the following equation holds

$$\begin{aligned}
&- \lambda_1 \left(\frac{u}{2\alpha_1} \left(A_1 cosh(0) + B_1 sinh(0) \right) + A_1 L_1 sinh(0) + B_1 L_1 cosh(0) \right) \\
&+ \lambda_2 \left(\frac{u}{2\alpha_2} \left(A_2 cosh(0) + B_2 sinh(0) \right) + A_2 L_2 sinh(0) + B_2 L_2 cosh(0) \right) - \lambda_1 \left(A_1 \frac{u}{2\alpha_1} + B_1 L_1 \right) \\
&+ \lambda_2 \left(A_2 \frac{u}{2\alpha_2} + B_2 L_2 \right) = -h_1 \left(\left[\!\left[\frac{u}{\alpha_j} (f_j - \rho c_j u T_m) \right]\!\right] - \rho L_m s \right), \\
\iff &- \lambda_1 \left(A_1 \frac{u}{2\alpha_1} + B_1 L_1 \right) + \lambda_2 \left(A_2 \frac{u}{2\alpha_2} + B_2 L_2 \right) = -h_1 \left(\left[\!\left[\frac{u}{\alpha_j} (f_j - \rho c_j u T_m) \right]\!\right] - \rho L_m s \right),
\end{aligned} \tag{4.71}$$

using the definition of B_j from (4.66), one gets

$$\begin{aligned}
&-\frac{(f_1 - \rho c_1 u T_m)}{\alpha_1} \left(-\frac{u}{2} + L_1 \alpha_1 \frac{\alpha_1 L_1 - \frac{u}{2} \frac{cosh(L_1 x_1)}{sinh(L_1 x_1)}}{\frac{u}{2} - \alpha_1 L_1 \frac{cosh(L_1 x_1)}{sinh(L_1 x_1)}} \right) + \\
&\qquad \frac{(f_2 - \rho c_2 u T_m)}{\alpha_2} \left(-\frac{u}{2} + L_2 \alpha_2 \frac{\alpha_2 L_2 - \frac{u}{2} \frac{cosh(L_2 x_2)}{sinh(L_2 x_2)}}{\frac{u}{2} - \alpha_2 L_2 \frac{cosh(L_2 x_2)}{sinh(L_2 x_2)}} \right) - \rho L_m s = 0
\end{aligned} \tag{4.72}$$

Finally, the dispersion relation associated to Problem (4.24) consists in the resolution of Equation (4.72) which can be defined as the zero of the function $f(s)$ expressed below

$$f(s) = -\frac{q_1}{\alpha_1} \left(-\frac{u}{2} + L_1 \alpha_1 \frac{\alpha_1 L_1 - \frac{u}{2} \frac{cosh(L_1 x_1)}{sinh(L_1 x_1)}}{\frac{u}{2} - \alpha_1 L_1 \frac{cosh(L_1 x_1)}{sinh(L_1 x_1)}} \right) + \frac{q_2}{\alpha_2} \left(-\frac{u}{2} + L_2 \alpha_2 \frac{\alpha_2 L_2 - \frac{u}{2} \frac{cosh(L_2 x_2)}{sinh(L_2 x_2)}}{\frac{u}{2} - \alpha_2 L_2 \frac{cosh(L_2 x_2)}{sinh(L_2 x_2)}} \right) - \rho L_m s \tag{4.73}$$

where

$$q_j = f_j - \rho c_j u T_m \tag{4.74}$$

As both s and k are values to be determined, a value of k is chosen and s is determined in terms of k. The value of s is determined by finding the zero of the function $f(s)$ in Equation (4.73) using a Newton procedure. The dependency in s of the function f in (4.73) is nonlinear as the variable s is present in the definitions of the coefficients L_1 and L_2, see Definition (4.64).

In this section, the general process to obtain the dispersion relation associated with the Problem (4.24) is detailed. The analysis of the dispersion relation (4.73) defined by the function $f(s)$ characterizes the stability of the Stefan model in its primal form defined in Equation (4.24). However, the dimensions of the different physical components of the Stefan model, especially for a melting configuration make it hard to analyze the results associated to the study of the dispersion relation (4.73) (e.g., the latent heat is of the order of 10^5). Section 4.4 and 4.5 have detailed a procedure that can be applied to other physical problem easily. In the next section, the focus is on the dimensionless Stefan model and its conditions. The associated dispersion relation is presented and the result of stability will be discussed. The process of defining a dimensionless system allows a better interpretation of the results concerning the stability of the model, and will show its benefits in the numerical scheme and the study of its stability.

4.6 Dimensionless Stefan model and its stability analysis

4.6.1 Dimensionless System

To simplify the analysis of the dispersion relation of Problem (4.24), the dimensionless form of the Stefan model is presented in this section with its corresponding conditions. The dimensionless system will enable to get a diffusive flux of order 1. In this case, if $q_1 = 0.5$ (where $q_j = f_j - \rho c_j u T_m$ and f_j the total flux) it means that half of the heat present in the system is removed through the liquid phase, at contrary if $q_1 > 1$ then heat is added to the system through the liquid phase which corresponds physically to a melting configuration.

The dimensionless variables are defined as follows:

$$\hat{T} = \frac{c}{L_m} T \ ; \ \hat{x} = \frac{|u|}{\alpha_s} x \ \ ; \ \hat{y} = \frac{|u|}{\alpha_s} y \ \ ; \ \ \hat{t} = \frac{u^2}{\alpha_s} t \tag{4.75}$$

where $\hat{T}$, $\hat{x}$, $\hat{y}$, $\hat{t}$ are the dimensionless variables, and α_s can either be α_1 or α_2 (the thermal diffusivity). The symbol $\hat{}$ over a variable will be used to refer to the dimensionless variables. As a reminder it is recalled that for a melting simulation the velocity u is a negative number. The sign of the solid velocity needs to be handled carefully to get the correct dimensionless linear system and a correct definition of $T_j(x, y, t)$ and $h_1(y, t)$.

Remark 20. The change in variables (4.75) allows to get variables without dimensions, this property is detailed below using Table 4.1. The notation $[\cdot]$ will be used to define that the dimension of a variable is the value looked at.

Symbol	Definition
J	Joule
Kg	Kilogram
K	Kelvin
m	meter
W	Watt
s	second

Table 4.1: Symbols used for the verification of the dimension of the change of variables (4.75)

- Concerning the dimensionless temperature $\hat{T}$:

$$\begin{aligned} [\hat{T}] = \left[\frac{c}{L_m} T\right] &= [c][L_m^{-1}][T] \\ &= (J.Kg^{-1}.K^{-1})(J^{-1}.Kg)(K) \\ &= 1. \end{aligned} \tag{4.76}$$

- Concerning the dimensionless time variable $\hat{t}$:

$$\begin{aligned} [\hat{t}] = \left[\frac{u^2}{\alpha_s} t\right] &= [u^2][\rho][c][\lambda^{-1}][t] \\ &= (m^2.s^{-2})(Kg.m^{-3})(J.Kg^{-1}.K^{-1})(W^{-1}.m.K)(s) \\ &= s^{-1}.J.W^{-1} \\ &= s^{-1}.J.(J^{-1}.s) \\ &= 1. \end{aligned} \tag{4.77}$$

The equality $1.W = 1.J.s^{-1}$ has been used to simplify the calculation.

- Concerning the space variables $\hat{x}$ and $\hat{y}$:

$$\begin{aligned}
[\hat{x}] = \left[\frac{|u|}{\alpha_s}x\right] &= [u][\alpha_s^{-1}][x] \\
&= [u][\rho][c][\lambda^{-1}][x] \\
&= (m.s^{-1})(Kg.m^{-3})(J.Kg^{-1}.K^{-1})(W^{-1}.m.K)(m) \\
&= s^{-1}.J.W^{-1} \\
&= s^{-1}.J.(J^{-1}.s) \\
&= 1.
\end{aligned} \tag{4.78}$$

The same goes for the variable $\hat{y}$. Once again, the equality $1.W = 1.J.s^{-1}$ has been used to simplify the calculation.

The first step for the dimensionless stability analysis is to express the dimensionless system of equation associated to the Stefan model. The differential operators are modified according to the change of variables (4.75). According to (4.24) the problem at stake in its dimensional form is

$$\rho c\frac{\partial T}{\partial t} + \frac{\partial}{\partial x}\left(\rho cuT - \lambda_j\frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(-\lambda_j\frac{\partial T}{\partial y}\right) = 0, \tag{4.79}$$

using the change of variable (4.75), Equation (4.79) becomes

$$\begin{aligned}
&\rho c\frac{u^2}{\alpha_s}\frac{\partial \hat{T}}{\partial \hat{t}} + \frac{|u|}{\alpha_s}\frac{\partial}{\partial \hat{x}}\left(\rho cu\hat{T} - \frac{|u|}{\alpha_s}\lambda_j\frac{\partial \hat{T}}{\partial \hat{x}}\right) - \frac{|u|^2}{\alpha_s^2}\frac{\partial}{\partial \hat{y}}\left(\lambda_j\frac{\partial \hat{T}}{\partial \hat{y}}\right) &&= 0, \\
\iff &\rho c|u|\frac{\partial \hat{T}}{\partial \hat{t}} + |u|\frac{\partial}{\partial \hat{x}}\left(-\rho c\hat{T} - \frac{\lambda_j}{\lambda_s}\rho c\frac{\partial \hat{T}}{\partial \hat{x}}\right) - |u|\frac{\partial}{\partial \hat{y}}\left(\frac{\lambda_j}{\lambda_s}\rho c\frac{\partial \hat{T}}{\partial \hat{y}}\right) &&= 0, \\
\iff &\boxed{\tfrac{\partial \hat{T}}{\partial \hat{t}} - \tfrac{\partial \hat{T}}{\partial \hat{x}} + \tfrac{\partial}{\partial \hat{x}}(-\hat{\lambda}_j\tfrac{\partial \hat{T}}{\partial \hat{x}}) + \tfrac{\partial}{\partial \hat{y}}(-\hat{\lambda}_j\tfrac{\partial \hat{T}}{\partial \hat{y}}) = 0},
\end{aligned} \tag{4.80}$$

with $\hat{\lambda}_j = \frac{\lambda_j}{\lambda_s}$ and λ_s chosen as λ_1 or λ_2 depending on the choice made for α_s. All the parameters of the model a parameters constant by phase.

To the dimensionless equation (4.80), one needs to associate the corresponding boundary conditions. The first three conditions of the dimensional model defined in (4.25) lead to:

$$\begin{aligned}
&1.)\ -\hat{\lambda}_j\hat{\nabla}\hat{T} &&= \hat{f}_j + \hat{T}_m &&\text{on } \hat{h}(\hat{y},\hat{t}), \\
&2.)\ \hat{q}_1 - \hat{q}_2 &&= 1 &&\text{on } \hat{h}(\hat{y},\hat{t}), \\
&3.)\ -\hat{\lambda}_j\hat{\nabla}\hat{T}(\hat{x},\hat{y},\hat{t})\cdot\hat{n} &&= 0 &&\text{on } \Gamma_N\ .
\end{aligned} \tag{4.81}$$

where $\hat{f}_j = \frac{f_j}{\rho L_m|u|}$, $\hat{T}_m = \frac{cT_m}{L_m}$ and $\hat{q}_j = \hat{f}_j + \hat{T}_m$.

To remove additional difficulties which do not impact the study of stability of the considered model, the thermal diffusivity is considered equal on both phases, i.e., $c_1 = c_2 = c$. This simplification allows to treat the jump operator in the Stefan Condition with more ease. The dimensional form of the Stefan condition from Conditions (4.25) is recalled below:

$$[\![-\lambda_j\nabla T_j\cdot n]\!] = \rho L_m\left(\frac{\partial h}{\partial t} - u\right)\cdot n, \tag{4.82}$$

using the change of variables (4.75) one gets

$$\begin{aligned}
& -\lambda_1 \nabla T_1 \cdot n + \lambda_2 \nabla T_2 \cdot n && = \rho L_m \left(\frac{\partial h}{\partial t} - u \right) \cdot n, \\
\iff & -\lambda_1 \frac{|u|}{\alpha_s} \hat{\nabla} T_1 \cdot n + \lambda_2 \frac{|u|}{\alpha_s} \hat{\nabla} T_2 \cdot n && = \rho L_m \left(\frac{u^2}{\alpha_s} \frac{\partial h}{\partial \hat{t}} - u \right) \cdot n, \\
\iff & -\frac{\lambda_1}{\alpha_s} \hat{\nabla} \left(\frac{L_m}{c_s} \hat{T}_1 \right) \cdot n + \frac{\lambda_2}{\alpha_s} \hat{\nabla} \left(\frac{L_m}{c_s} \hat{T}_2 \right) \cdot n && = \rho L_m \left(\frac{|u|}{\alpha_s} \frac{\partial h}{\partial \hat{t}} + 1 \right) \cdot n, \\
\implies & \boxed{\left[\!\left[-\hat{\lambda}_j \hat{\nabla} \hat{T}_j \cdot n \right]\!\right] = \left(\tfrac{\partial \hat{h}}{\partial \hat{t}} + 1 \right) \cdot n}.
\end{aligned} \tag{4.83}$$

To summarize, the dimensionless problem consists in :
Find $\hat{T}(\hat{x}, \hat{y}, \hat{t})$ and an interface location $\hat{h}(\hat{y}, \hat{t})$ satisfying

$$\partial_{\hat{t}} \hat{T} + \frac{\partial}{\partial \hat{x}} \left(-\hat{T} - \hat{\lambda}_j \frac{\partial}{\partial \hat{x}} \hat{T} \right) + \frac{\partial}{\partial \hat{y}} \left(-\hat{\lambda}_j \frac{\partial}{\partial \hat{y}} \hat{T} \right) = 0, \tag{4.84}$$

where the following boundary conditions are satisfied

$$\begin{aligned}
& 1. \ -\hat{\lambda}_j \hat{\nabla} \hat{T}(\hat{x}_j) && = \hat{f}_j + \hat{T}(\hat{x}_j) && \text{on } \Gamma_{N_j} \\
& 2. \ \hat{T}(\hat{x}) && = \hat{T}_m && \text{on } \hat{h}(\hat{y}, \hat{t}), \\
& 3. \ [\![\hat{q}_j]\!] = \hat{q}_1 - \hat{q}_2 && = 1 && \text{on } \hat{h}(\hat{y}, \hat{t}), \\
& 4. \ -\hat{\lambda}_j \hat{\nabla} \hat{T}(\hat{x}, \hat{y}, \hat{t}) \cdot \hat{n} && = 0 && \text{on } \Gamma_N \ . \\
& 5. \ [\![-\hat{\lambda}_j \hat{\nabla} \hat{T}]\!] \cdot n && = (\partial_{\hat{t}} \hat{h} + 1) \cdot n && \text{on } \hat{h}(\hat{y}, \hat{t}).
\end{aligned} \tag{4.85}$$

4.6.2 Dimensionless Linearization Process

The process of linearization is the same as expressed in Section 4.3 and Section 4.4 for the dimensional Stefan model. For more details on the different calculation steps the reader is referred to these sections.

Definition (4.26) of $T_j(x)$ needs to be adjusted to correspond to the dimensionless system (4.84) and its conditions (4.85). The ODE associated with the calculation of the leading order solution $\hat{T}_{j,0}(\hat{x})$ is as follows:

$$-\frac{\partial \hat{T}_{j,0}}{\partial \hat{x}}(\hat{x}) - \hat{\lambda}_j \frac{\partial^2 \hat{T}_{j,0}}{\partial \hat{x}^2}(\hat{x}) = 0, \tag{4.86}$$

which possesses two square roots; 0 and $\frac{-1}{\hat{\lambda}_j}$. Using the same method as in Section 4.3 for the identification of the constants one gets

$$\hat{T}_{j,0}(\hat{x}) = \hat{T}_m + \hat{q}_j \left(e^{\frac{-(\hat{x} - \hat{h}_0)}{\hat{\lambda}_j}} - 1 \right), \tag{4.87}$$

where $\hat{h}_0$ is the initial interface position.

Now, the interest is reported on the initialization of the perturbed solution $\hat{T}_{j,1}(\hat{x}, \hat{y}, \hat{t})$. As a reminder, it is recalled that the approximation $\hat{T}_j(\hat{x}, \hat{y}, \hat{t})$ is defined by the leading order solution $\hat{T}_{j,0}(\hat{x})$ and the perturbed solution $\hat{T}_{j,1}(\hat{x}, \hat{y}, \hat{t})$ also called first order solution where

$$\hat{T}_j(\hat{x}, \hat{y}, \hat{t}) = \hat{T}_{j,0}(\hat{x}) + \epsilon \hat{T}_{j,1}(\hat{x}, \hat{y}, \hat{t}). \tag{4.88}$$

with ϵ a small perturbation.
To get the dimensionless form of the perturbed solution $\hat{T}_{j,1}(\hat{x},\hat{y},\hat{t})$, the first step consists in defining the dimensionless conditions that $\hat{T}_{j,1}(\hat{x})$ has to satisfy. For the dimensional model the conditions were

$$\begin{aligned} &1.\ -\lambda_j \nabla T_{j,1}(x_j) + \rho c u T_{j,1}(x_j) &&= 0, \\ &2.\ T_{j,1}(h_0) &&= \frac{h_1}{\lambda_j}(f_j - \rho c u T_m), \\ &3.\ [\![-\lambda_j \nabla T_{j,1}(h_0)]\!] &&= -h_1 \left(\left[\!\!\left[\frac{u}{\alpha_j}(f_j - \rho c u T_m) \right]\!\!\right] - \rho L_m s \right). \end{aligned} \tag{4.89}$$

From the first condition of (4.89) and the change of variables (4.75) one gets

$$\begin{aligned} &-\lambda_j \frac{\partial}{\frac{\alpha_s}{|u|}\partial\hat{x}} \frac{L_m}{c}\hat{T}_{j,1} + \rho c u \frac{L_m}{c}\hat{T}_{j,1} = 0, \\ \Longrightarrow\ &\boxed{-\hat{\lambda}_j \hat{\nabla}\hat{T}_{j,1} - \hat{T}_{j,1} = 0}\,. \end{aligned} \tag{4.90}$$

From the second condition of (4.89) and the change of variables (4.75) the condition becomes

$$\begin{aligned} &\frac{L_m}{c}\hat{T}_{j,1}(h_0) &&= \frac{h_1}{\lambda_j}(\rho L_m |u| \hat{f}_j - \rho u L_m \hat{T}_m), \\ \Longleftrightarrow\ &\hat{T}_{j,1}(h_0) &&= \frac{h_1}{\hat{\lambda}_j}\frac{|u|}{\alpha_s}\hat{q}_j, \\ \Longrightarrow\ &\boxed{\hat{T}_{j,1}(h_0) = \tfrac{\hat{h}_1}{\hat{\lambda}_j}\hat{q}_j}, \end{aligned} \tag{4.91}$$

where $\hat{h}_1 = \frac{|u|h_1}{\alpha_s}$.
Finally, for the last condition of (4.89) and the change of variables (4.75) the dimensionless Stefan condition is defined as follows

$$\begin{aligned} &\left[\!\!\left[-\hat{\lambda}_j \frac{\rho c |u|}{c} L_m \hat{\nabla}\hat{T} \right]\!\!\right] &&= -h_1 \left(\left[\!\!\left[\frac{u}{\alpha_j}\rho L_m |u| \hat{q}_j \right]\!\!\right] - \rho L_m s \right) \\ \Longleftrightarrow\ &\left[\!\!\left[-\hat{\lambda}_j \hat{\nabla}\hat{T} \right]\!\!\right] &&= \hat{h}_1 \frac{\alpha_s}{u} \left(\left[\!\!\left[\frac{u}{\alpha_j}\hat{q}_j \right]\!\!\right] - \frac{s}{|u|} \right) \\ \Longrightarrow\ &\boxed{[\![-\hat{\lambda}_j \hat{\nabla}\hat{T}]\!] = \hat{h}_1 \left(\left[\!\!\left[\tfrac{1}{\hat{\lambda}_j}\hat{q}_j \right]\!\!\right] + \hat{s} \right)} \end{aligned} \tag{4.92}$$

where $\hat{s} = s\frac{\alpha_s}{u^2}$.
A change of variables is also necessary for the value of s and k which are variables with dimension. The wavenumber k is replaced by its dimensionless form $\hat{k} = k\frac{\alpha_s}{|u|}$.

All the information necessary to characterize the perturbation on the temperature and the interface position is now available. In their dimensionless form $\hat{T}_{j,1}(\hat{x},\hat{y},\hat{t})$ and $\hat{h}(\hat{y},\hat{t})$ are defined by

$$\begin{aligned} \hat{T}_{j,1}(\hat{x},\hat{y},\hat{t}) &= \hat{T}_{j,1}(\hat{x}) exp(i\hat{k}\hat{y} + \hat{s}\hat{t}) \\ \hat{h}_1(\hat{y},\hat{t}) &= \hat{h}_1 exp(i\hat{k}\hat{y} + \hat{s}\hat{t}) \end{aligned} \tag{4.93}$$

Plugging Equation (4.93) for the definition of $\hat{T}_{j,1}(\hat{x},\hat{y},\hat{t})$ in Equation (4.80) one obtains the following ODE :

$$\hat{T}_{j,1}(\hat{x})(\hat{s}+\hat{\lambda}_j\hat{k}^2) - \frac{\partial \hat{T}_{j,1}}{\partial \hat{x}}(\hat{x}) - \hat{\lambda}_j \frac{\partial^2 \hat{T}_{j,1}}{\partial \hat{x}^2}(\hat{x}) = 0, \tag{4.94}$$

which has a solution of the form

$$\hat{T}_{j,1}(\hat{x}) = exp(\frac{-\hat{x}}{2\hat{\lambda}_j})(\hat{A}_j cosh(\hat{L}_j\hat{x}) + \hat{B}_j sinh(\hat{L}_j\hat{x})), \tag{4.95}$$

and where

$$\begin{aligned} \hat{\Delta}_j &= 1 + 4\hat{\lambda}_j(\hat{s}+\hat{\lambda}_j\hat{k}^2) \ , \\ \hat{L}_j &= \frac{\sqrt{\hat{\Delta}_j}}{-2\hat{\lambda}_j} \ . \end{aligned} \tag{4.96}$$

In order to express the parameters $\hat{A}_j$ and $\hat{B}_j$, the definition of $\hat{T}_{j,1}(\hat{x})$ is plugged in Condition (4.90). Then one gets

$$\hat{B}_j = \frac{\frac{1}{2}cosh(\hat{L}_j\hat{x}_j) + \hat{\lambda}_j\hat{L}_j sinh(\hat{L}_j\hat{x}_j)}{-\frac{1}{2}sinh(\hat{L}_j\hat{x}_j) - \hat{\lambda}_j\hat{L}_j cosh(\hat{L}_j\hat{x}_j)}\hat{A}_j = \hat{c}_{s,j}\hat{A}_j \ , \tag{4.97}$$

where

$$\hat{c}_{s,j} = \frac{\frac{1}{2}cosh(\hat{L}_j\hat{x}_j) + \hat{\lambda}_j\hat{L}_j sinh(\hat{L}_j\hat{x}_j)}{-\frac{1}{2}sinh(\hat{L}_j\hat{x}_j) - \hat{\lambda}_j\hat{L}_j cosh(\hat{L}_j\hat{x}_j)} \ . \tag{4.98}$$

For the definition of the phase-change parameters $\hat{A}_j$ one gets from condition (4.91)

$$\hat{A}_j = \hat{h}_1 \frac{\hat{q}_j}{\hat{\lambda}_j}. \tag{4.99}$$

The last step consists in defining the dispersion relation using the Stefan condition (4.92), defined at the interface position $\hat{h}_0 = 0$. In a similar way than in Section 4.5 the following equation holds

$$\implies \boxed{-\frac{\hat{q}_1}{\hat{\lambda}_1}\left(\frac{1}{2} + \hat{L}_1\hat{c}_{s,1}\hat{\lambda}_1\right) + \frac{\hat{q}_2}{\hat{\lambda}_2}\left(\frac{1}{2} + \hat{L}_2\hat{c}_{s,2}\hat{\lambda}_2\right) - \hat{s} = 0} \tag{4.100}$$

Then, the dispersion relation associated to the dimensionless Stefan model (4.84) is $\hat{f}(\hat{s})$ defined as follows

$$\hat{f}(\hat{s}) = -\frac{\hat{q}_1}{\hat{\lambda}_1}(\frac{1}{2} + \hat{\lambda}_1\hat{L}_1\frac{\hat{\lambda}_1\hat{L}_1 tanh(\hat{L}_1\hat{x}_1) + \frac{1}{2}}{\frac{-1}{2}tanh(\hat{L}_1\hat{x}_1) - \hat{\lambda}_1\hat{L}_1}) + \frac{\hat{q}_2}{\hat{\lambda}_2}(\frac{1}{2} + \hat{\lambda}_2\hat{L}_2\frac{\hat{\lambda}_2\hat{L}_2 tanh(\hat{L}_2\hat{x}_2) + \frac{1}{2}}{\frac{-1}{2}tanh(\hat{L}_2\hat{x}_2) - \hat{\lambda}_2\hat{L}_2}) - \hat{s} \ , \tag{4.101}$$

where

$$tanh(x) = \frac{sinh(x)}{cosh(x)}, \tag{4.102}$$

see Equation (4.62) and Equation (4.63) for the definition of *cosh* and *sinh*.

Remark 21. The dependency in $\hat{s}$ of the function $f(\hat{s})$ in Equation (4.101) is nonlinear. Indeed, the variable $\hat{s}$ is present in the definition of $\hat{L}_j$ defined in (4.96). Following a choice of value $\hat{k}$, a Newton procedure is used to get the corresponding $\hat{s}$ value.

The study of the dimensionless dispersion relation (4.101) of the dimensionless Stefan model (4.84) is presented in Figure 4.4. If $\hat{q}_1 < 0.5$ (the liquid phase) the system is unstable, as $\hat{s}$ is a positive real number for all values of $\hat{k}$. For the study of the Stefan model, the value of $\hat{s}$

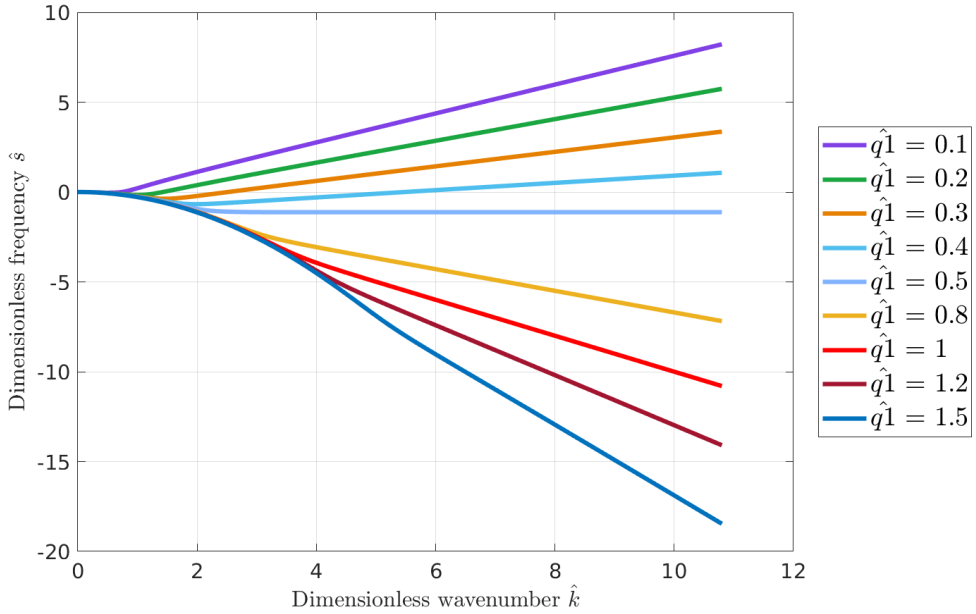


Figure 4.4: Analysis of the behavior of $\hat{s}$ in terms of $\hat{k}$ and flux $\hat{q}_1$ from the liquid phase

calculated from the choice of $\hat{k}$ in Equation (4.101) is always a real number, i.e., $Re(\hat{s}) = \hat{s}$. Nevertheless for a melting simulation, the physical properties of the problem gives us that $\hat{q}_1$ satisfies $\hat{q}_1 > 1.0$, as heat is added to the domain through the liquid phase. By analysis of Figure 4.4, the values of $\hat{s}$ in that scenario are always negative for all values of $\hat{k}$, and the conclusion can be made that the Stefan model presented in its dimensionless form in (4.84) is a stable model around the steady state $\hat{T}_{j,0}(\hat{x})$ defined in Equation (4.87). In this section the theoretical stability analysis of the Stefan model has shown a stable model under the correct choice of flux imposition, which in the case of a melting simulation is always stable ($\hat{q}_1 > 1$; $\hat{q}_2 = \hat{q}_1 - 1$).

4.6.3 Dissipation of the perturbation at the boundaries

An important part of the stability analysis is to verify that the perturbation added to the steady state $\hat{T}_{j,0}(\hat{x})$, i.e. the first order solution $\hat{T}_{j,1}(\hat{x}, \hat{y}, \hat{t})$ converges to 0 at the left and right boundary of the domain, see Figure 4.2. In this Section this criterion is verified. The behavior of $\hat{T}_{1,1}(\hat{x})$ is studied when $\hat{x}_1 \to -\infty$. The same procedure is done for $\hat{T}_{2,1}(\hat{x})$ when $\hat{x}_2 \to +\infty$, see Figure 4.2. The first step in the study of convergence consists in expressing the value of the coefficient $\hat{B}_j$ by developing the hyperbolic functions in its definition to obtain a definition where the calculation of the limit is easier to obtain, see Equation (4.97).

$$\begin{aligned} \hat{B}_j &= \frac{0.5(\frac{e^{\hat{L}_j\hat{x}_j}+e^{-\hat{L}_j\hat{x}_j}}{2}) + \hat{\lambda}_j\hat{L}_j(\frac{e^{\hat{L}_j\hat{x}_j}-e^{-\hat{L}_j\hat{x}_j}}{2})}{-0.5(\frac{e^{\hat{L}_j\hat{x}_j}-e^{-\hat{L}_j\hat{x}_j}}{2}) - \hat{\lambda}_j\hat{L}_j(\frac{e^{\hat{L}_j\hat{x}_j}+e^{-\hat{L}_j\hat{x}_j}}{2})}\hat{A}_j \\ &= \frac{e^{\hat{L}_j\hat{x}_j}(0.5 + \hat{\lambda}_j\hat{L}_j) + e^{-\hat{L}_j\hat{x}_j}(0.5 - \hat{\lambda}_j\hat{L}_j)}{e^{\hat{L}_j\hat{x}_j}(-0.5 - \hat{\lambda}_j\hat{L}_j) + e^{-\hat{L}_j\hat{x}_j}(0.5 - \hat{\lambda}_j\hat{L}_j)}\hat{A}_j \end{aligned} \tag{4.103}$$

For the coefficient $\hat{B}_1$ from the liquid phase, using that $\hat{L}_1$ is a negative number, see Equation (4.96), the limit of $\hat{B}_1(\hat{x}_1)$ when $\hat{x}_1 \to -\infty$ is defined by

$$\begin{aligned} \hat{B}_1(\hat{x}_1) &= \frac{(0.5+\hat{\lambda}_1\hat{L}_1) + e^{-2\hat{L}_1\hat{x}_1}(0.5-\hat{\lambda}_1\hat{L}_1)}{(-0.5-\hat{\lambda}_1\hat{L}_1) + e^{-2\hat{L}_1\hat{x}_1}(0.5-\hat{\lambda}_1\hat{L}_1)}\hat{A}_1, \\ \implies \hat{B}_1(\hat{x}_1 \to -\infty) &= \frac{(0.5+\hat{\lambda}_1\hat{L}_1)}{(-0.5-\hat{\lambda}_1\hat{L}_1)} = -\hat{A}_1. \end{aligned} \tag{4.104}$$

For the coefficient $\hat{B}_2$ for the solid phase, the limit of $\hat{B}_2(\hat{x}_2)$ when $\hat{x}_2 \to +\infty$ is equal to

$$\begin{aligned} \hat{B}_2(\hat{x}_2) &= \frac{e^{2\hat{L}_2\hat{x}_2}(0.5+\hat{\lambda}_2\hat{L}_2) + (0.5-\hat{\lambda}_2\hat{L}_2)}{e^{2\hat{L}_2\hat{x}_2}(-0.5-\hat{\lambda}_2\hat{L}_2) + (0.5-\hat{\lambda}_2\hat{L}_2)}\hat{A}_2, \\ \implies \hat{B}_2(\hat{x}_2 \to +\infty) &= \frac{(0.5-\hat{\lambda}_2\hat{L}_2)}{(0.5-\hat{\lambda}_2\hat{L}_2)} = \hat{A}_2. \end{aligned} \tag{4.105}$$

Using Equation (4.104) and Equation (4.105), at convergence of the boundary to infinity, i.e., considering a domain infinite in the x-direction the functions $\hat{T}_{1,1}(\hat{x})$ and $\hat{T}_{2,1}(\hat{x})$ can be rewritten as

$$\begin{aligned} \hat{T}_{1,1}(\hat{x}) &= exp\left(\frac{\hat{x}}{2\hat{\lambda}_1}\left(\sqrt{\hat{\Delta}_1}-1\right)\right)\hat{A}_1, \\ \hat{T}_{2,1}(\hat{x}) &= exp\left(\frac{-\hat{x}}{2\hat{\lambda}_2}\left(\sqrt{\hat{\Delta}_2}+1\right)\right)\hat{A}_2. \end{aligned} \tag{4.106}$$

Since $\sqrt{\hat{\Delta}_1}-1>0$ and $\sqrt{\hat{\Delta}_2}+1>0$, both functions are converging to 0 at their boundaries $\hat{x}_1$ and $\hat{x}_2$.

4.6.4 Examples of perturbation

In Figure 4.5 one can observe an example of a wavelength associated to a wavenumber $k=10$ on a domain of height $L_y = \frac{2\pi}{\hat{k}}$. This perturbation at the interface can be associated with a perturbed temperature shown in Figure 4.6. The perturbed temperature is continuous at the interface for $\hat{T}_{j,0}$, but the definition of $\hat{T}_{j,1}$ creates a discontinuity that disappears with time as the model is stable, as shown in Figure 4.4 for a melting configuration.
Figure 4.7 shows the perturbed interface for $\epsilon = 1e^{-3}$, $\hat{k}=20$, $\hat{q}_1 = 1.5$ and for a domain of height $\frac{2\pi}{10}$. Figure 4.8 shows the contour of the perturbed temperature $\hat{T}_1$ using the same parameters and with $\hat{\rho}=1$, $\hat{L}_m=1$, $\hat{T}_m = 1.6898$, $\hat{c}_1=\hat{c}_2=1$, $\hat{\lambda}_1 = \frac{0.6}{2.1}$ and $\hat{\lambda}_2=1$. Different types of perturbations can be chosen to assess numerically the stability of the complete model.

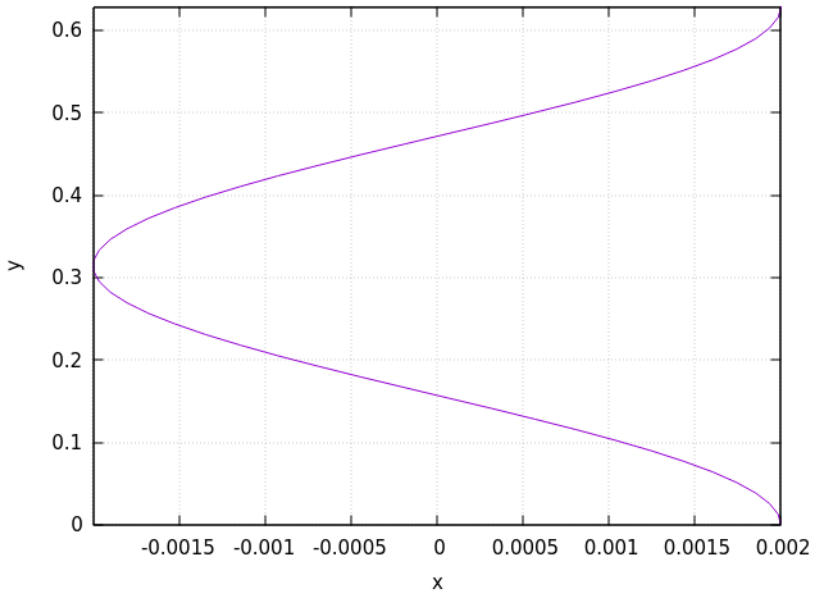


Figure 4.5: Example of Perturbed Interface - $\epsilon = 1e^{-3}$, $\hat{k} = 10$, $\hat{q}_1 = 1.5$

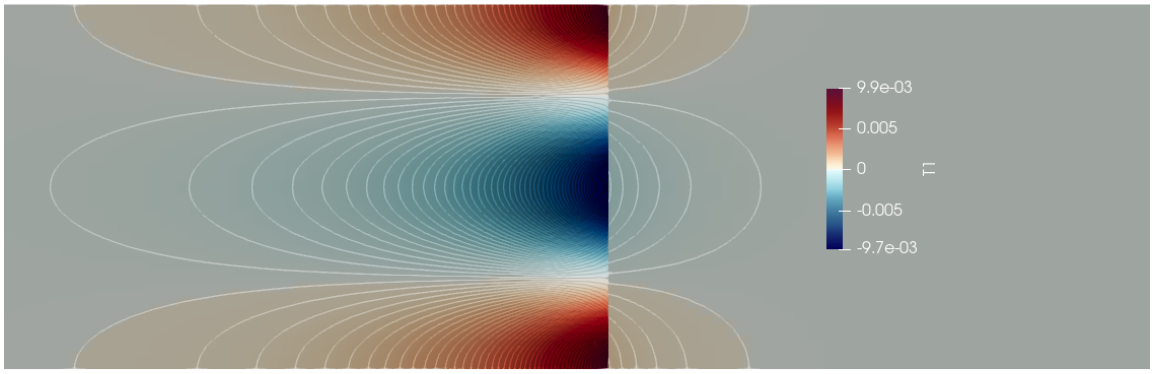


Figure 4.6: Visualization of the perturbed temperature $\hat{T}_1$ on both phases for Figure 4.5 with isolines, $\hat{\rho} = 1$, $\hat{L}_m = 1$,$\hat{T}m = 1.6898$, $\hat{c}_1 = \hat{c}_2 = 1$, $\hat{\lambda}_1 = 0.6/2.1$, $\hat{\lambda}_2 = 1$

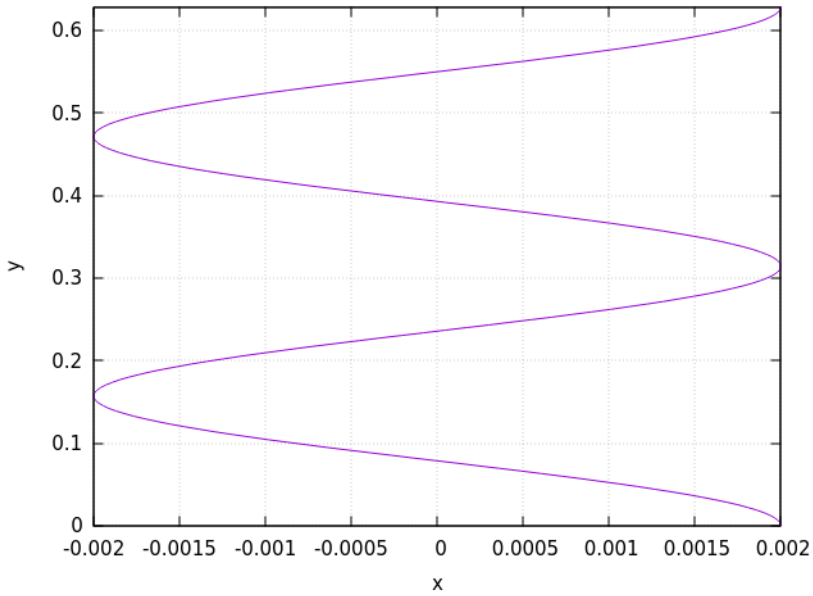


Figure 4.7: Example of Perturbed Interface - $\epsilon = 1e^{-3}$, $\hat{k} = 20$, $\hat{q}_1 = 1.5$

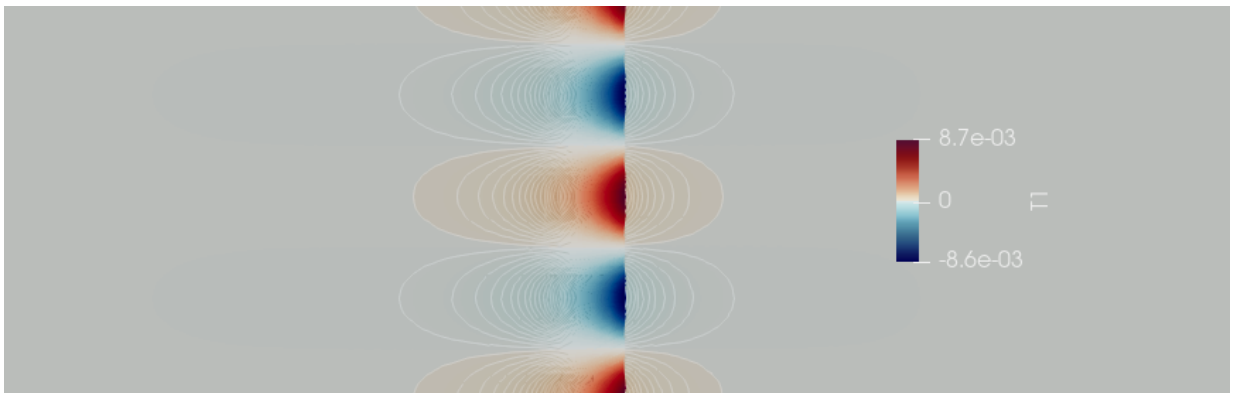


Figure 4.8: Visualization of perturbation temperature $\hat{T}_1$ on both phases for Figure 4.7 with isolines, $\hat{\rho} = 1$, $\hat{L}_m = 1$,$\hat{T}m = 1.6898$, $\hat{c}_1 = \hat{c}_2 = 1$, $\hat{\lambda}_1 = 0.6/2.1$, $\hat{\lambda}_2 = 1$

4.6.5 Definition of a real solution

By definition $\hat{T}_{j,1}(\hat{x}, \hat{y}, \hat{t})$ in (4.88) is a complex function. In order to deal with only real numbers in the numerical stability analysis the initial condition and perturbation are constructed using the solution associated with a pair of wavenumber values $(\hat{k}, -\hat{k})$. Indeed, conditions (4.89) define an eigen value problem of the form

$$MV = \gamma V, \tag{4.107}$$

where the vector of unknown is $\{\hat{A}_1, \hat{B}_1, \hat{A}_2, \hat{B}_2, \hat{h}_1\}$.
If System (4.107) is associated with the wavenumber $\hat{\boldsymbol{k}}$, then the wavenumber $-\hat{\boldsymbol{k}}$ is associated with the complex conjugate system

$$\bar{M}\bar{V} = \bar{\gamma}\bar{V}. \tag{4.108}$$

Then, using a pair of wavenumber $(\hat{k}, -\hat{k})$ it is possible to construct a solution $\hat{T}_{j,1}(\hat{x})$ such as

$$\hat{T}_{j,1}(\hat{x}, \hat{y}, \hat{t}) = \psi_1 \hat{T}_{j,1}(\hat{x}) exp(i\hat{k}\hat{y} + \hat{s}\hat{t}) + \psi_2 \hat{\bar{T}}_{j,1}(\hat{x}) exp(-i\hat{k}\hat{y} + \hat{s}\hat{t}), \tag{4.109}$$

where ψ_1 and ψ_2 are complex conjugate numbers

$$\begin{aligned} \psi_1 &= a + ib, \\ \psi_2 &= a - ib, \end{aligned} \tag{4.110}$$

where $a, b \in \mathbb{R}$.
At $\hat{t} = 0$, when Euler's formula (4.4) is employed and the following equality holds

$$\hat{T}_{j,1}(\hat{x}) = \psi_1 \hat{T}_{j,1}(\hat{x})(cos(\hat{k}\hat{y}) + isin(\hat{k}\hat{y})) + \psi_2 \hat{\bar{T}}_{j,1}(\hat{x})(cos(\hat{k}\hat{y}) - isin(\hat{k}\hat{y})). \tag{4.111}$$

Let T_R be the real part of $\hat{T}_{j,1}(\hat{x})$ and T_I be the imaginary part of $\hat{T}_{j,1}(\hat{x})$, Equation (4.111) becomes

$$\hat{T}_{j,1} = 2T_R(acos(\hat{k}\hat{y}) - bsin(\hat{k}\hat{y})) - 2T_I(asin(\hat{k}\hat{y}) + bcos(\hat{k}\hat{y})). \tag{4.112}$$

The choice of a and b is arbitrary. In the following $a = 1$ and $b = 0$, i.e. that $\hat{T}_{j,1}(\hat{x}, \hat{y}, \hat{t})$ is of the form

$$\hat{T}_{j,1} = 2T_R cos(\hat{k}\hat{y}) - 2T_I sin(\hat{k}\hat{y}), \tag{4.113}$$

and Equation (4.113) defines a real solution.
The same procedure is done on the interface perturbation $\hat{h}_1(\hat{y}, \hat{t})$ which leads to the real

solution

$$\hat{h}_1(\hat{y},\hat{t}) = 2h_{1,R}cos(\hat{k}\hat{y}) - 2h_{1,I}sin(\hat{k}\hat{y}). \tag{4.114}$$

In the next section, the stability of the numerical scheme presented in Chapter 2 and in Equation (2.89) will be discussed. Results on the convergence of the method will be presented and the method of reconstruction of the interface velocity will be adapted to the stability analysis on grids composed of triangular elements.

4.7 Numerical Model & Stability

In this Section, the stability analysis is studied for the e-SBM, numerical method implemented to solve the Stefan problem. Some details presented in Chapter 2 will be briefly recalled to allow a faster and easier comprehension of the work presented in this Section.
Contrary to the stability analysis conducted on the continuous Stefan model, the numerical study is performed using the mixed formulation. Results will be presented once again for dimensionless variables. The section will end by some results on the stability of the method and its convergence. To simplify the comprehension of this section the notation $\hat{}$ will be dropped but the dimensionless variables presented in Section 4.6 are considered.

4.7.1 Brief overview of the Shifted Boundary Method

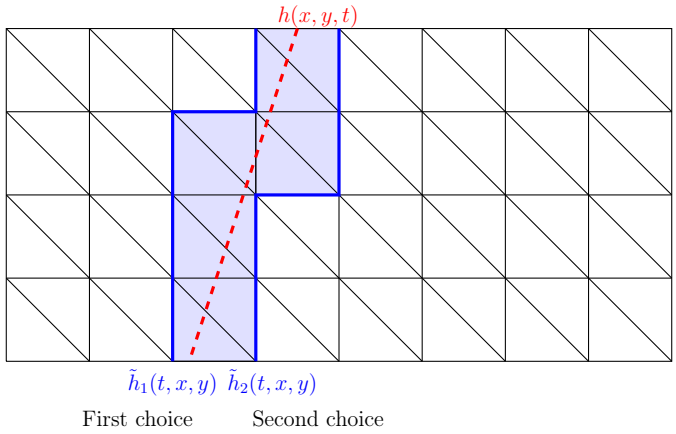
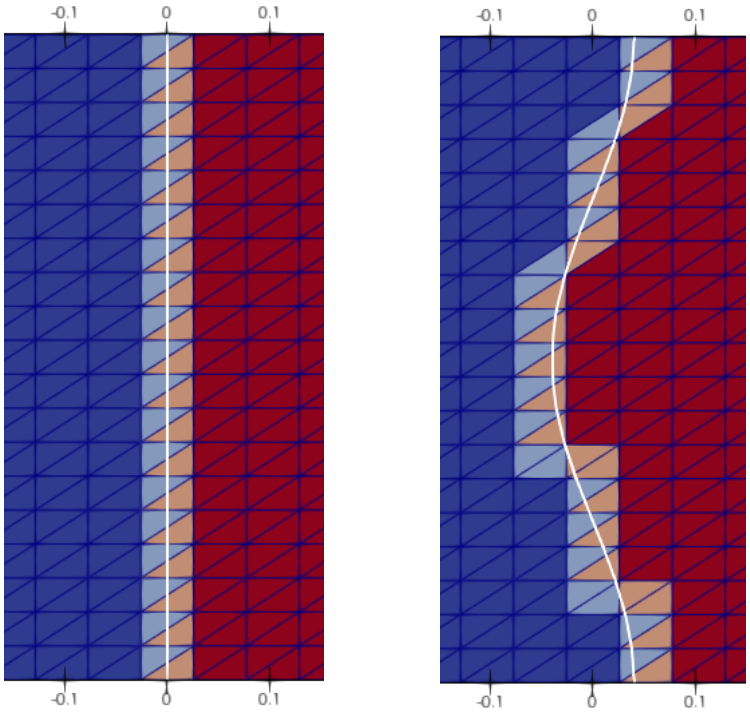


Figure 4.9: Definition of the surrogate interface associated to $h(x,y,t)$

The **Shifted Boundary Method** (SBM) is a method coupled with a finite element resolution that involves imposing boundary conditions differently from how they are defined in the initial system for the physical boundary. Instead, these conditions are shifted to a surrogate boundary, which is defined as edges in 2D or faces in 3D of the considered mesh. The first step in the method consists of defining the surrogate interface $\tilde{h}(x,y,t)$, conform to the grid, on which the interface conditions are imposed instead of on the physical interface $h(x,y,t)$. Numerically, the elements crossed by the physical interface create a domain of elements where the interface is not grid conformed (see blue area in Figure 4.9). The surrogate interface becomes one of the boundaries of this identified domain, $\tilde{h}_1$ and $\tilde{h}_2$ in Figure 4.9. In Figure 4.10 the identification of the surrogate interface is shown before and after a perturbation of the interface for a perturbation $\epsilon = 2e^{-2}$. The list of elements of the mesh cut by the interface is updated and the choice of the surrogate interface becomes different as the area identified by the cut elements has changed between Figure 4.10(a) and Figure 4.10(b). In this chapter, the physical interface



(a) Interface without perturbation

(b) Perturbed interface - $\epsilon = 2e^{-2}$

Figure 4.10: Identification of the Surrogate before and After Perturbation

is discretized and represented as a set of points defining a set of edges. The position of the points is moved independently to update the interface position.

In order to connect the physical and surrogate boundary to each other a mapping $\mathbf{M}$ is introduced to deduce boundary conditions on $\tilde{h}(x, y, t)$ from the boundary conditions on $h(x, y, t)$

$$\begin{aligned} \mathbf{M} \ : \ & \tilde{h}(x, y, t) \rightarrow h(x, y, t) \\ & (\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}) \mapsto (\boldsymbol{x}, \boldsymbol{y}) \quad . \end{aligned} \tag{4.115}$$

With the definition of the mapping $\mathbf{M}$ the distance vector function $\boldsymbol{d}$ can be defined and measures the distance between $h(x, y, t)$ and $\tilde{h}(x, y, t)$, as follows

$$\boldsymbol{d}(\tilde{\boldsymbol{x}}) = \boldsymbol{x} - \tilde{\boldsymbol{x}} = [\mathbf{M} - \mathbf{I}](\tilde{\boldsymbol{x}}) \quad . \tag{4.116}$$

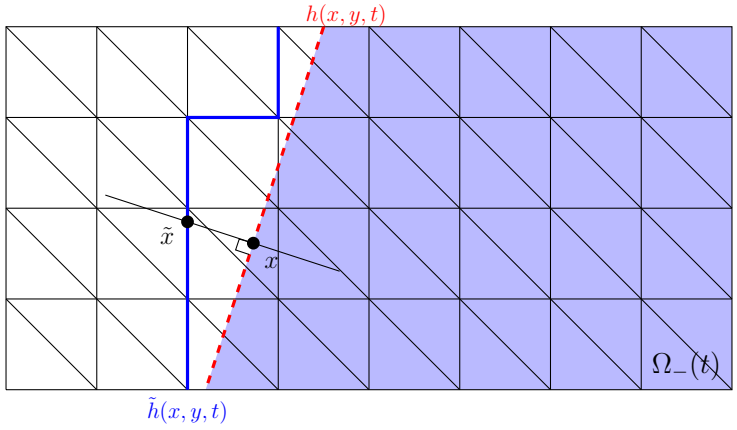


Figure 4.11: Numerical analysis: definition of the mapping $\mathbf{M}$

Even if the uniqueness of the projection defined by the map $\mathbf{M}$ in (4.115) is defined for closed convex spaces only, numerically different criteria are added to make the projection suitable to all domains of resolution, see Section 1.2 of Chapter 1 for more details.

The second step in the SBM is to enforce weakly the boundary conditions using the **Nitsche's method** (see [48]), by replacing the conditions taking place on the physical boundary by an extrapolation by **Taylor expansions** to the surrogate boundary $\tilde{h}(x, y, t)$. The order of accuracy of the method depends on the order of the development in the Taylor expansion

$$\begin{aligned} [\![T]\!]_h &= [\![T + \nabla T \cdot \boldsymbol{d}]\!]_{\tilde{h}} + \mathcal{O}(||\boldsymbol{d}(\tilde{\boldsymbol{x}})||^2), \\ \implies [\![T]\!]_{\tilde{h}} &= [\![T]\!]_h - [\![\nabla T \cdot \boldsymbol{d}]\!]_{\tilde{h}}, \end{aligned} \tag{4.117}$$

where d is defined in (4.116).
In Equation (4.117) the accuracy of $[\![T]\!]_{\tilde{h}}$ is of order 2 as the Taylor development is performed to the second order. For additional information on the SBM see Chapter 1 of this manuscript and references [2, 80, 3, 90, 92, 66, 67, 95].

4.7.2 Mixed Formulation

For the numerical stability analysis of the Stefan model the mixed formulation is the one considered. The mixed formulation refers to a formulation where both the temperature T and the gradient of temperature $-\lambda\nabla T$ are explicit unknowns to be solved. This choice was made to prevent any gradient reconstruction, where the value of the flux is necessary to update numerically the physical interface position. The problem presented in this section is also solved in its dimensionless form, see (4.75) for the definition of the change of variables.

4.7.2.a Problem definition

Given an initial temperature T_0, a flux β_0 and an initial interface position h_0, the mixed Stefan model in its dimensionless form consists in finding (T, β) and the interface position $h(x, y, t)$ such that

$$\begin{aligned} \frac{\partial T}{\partial t} + \nabla \cdot \beta &= 0 && \text{on } \Omega, \\ \beta &= -\lambda \nabla T && \text{on } \Omega, \end{aligned} \tag{4.118}$$

where (T, β) satisfy the boundary conditions (4.119).

$$\begin{aligned} &1.\ \beta(x_j) && = q_j && \text{on } \Gamma_{N_j}, \\ &2.\ T(x) && = T_m && \text{on } h(x, y, t), \\ &3.\ [\![q_j]\!] = q_1 - q_2 && = 1 && \text{on } h(x, y, t), \\ &4.\ \beta(x, y, t) \cdot n && = 0 && \text{on } \Gamma_N\ . \\ &5.\ [\![\beta]\!] \cdot n && = \left(\frac{\partial h}{\partial t} + 1\right) \cdot n && \text{on } h(x, y, t). \end{aligned} \tag{4.119}$$

The convective term presents in Equation (4.84) for the stability analysis of the continuous model, is not considered in the definition of Problem (4.118) used to study the numerical stability of the Stefan model and the e-SBM. This choice was made to study the same model as established in Chapter 2 and 3. We recall that the e-SBM presented in Chapter 2 is an enriched $\mathbb{P}^1$ FEM allowing a second order accuracy on both the temperature field and the flux for moving boundaries.

The convective term is only used in the definition of the linearized system to initialize the interface position a $t = 0$ as a flat interface, and defined a reference model where the solid is pulled away of the domain. The addition of that term does not impact the resolution and the same linearization than proposed in Section (4.6) can be used as initial condition. The choice of a mixed formulation in (4.118) leads to the following steady state couple $(T_{j,0}, \beta_{j,0})$ and interface position h_0 used at the initialization:

$$\begin{aligned} T_{j,0}(x) &= T_m + q_j(e^{\frac{-(x-h_0)}{\lambda_j}} - 1), \\ \beta_{j,0}|_x(x) &= q_j\ e^{-\frac{(x-h_0)}{\lambda_j}}, \\ \beta_{j,0}|_y(x) &= 0, \\ h_0 &= 0. \end{aligned} \tag{4.120}$$

In order to study the stability, the temperature needs to be perturbed $(T_{j,1})$ and Definition (4.114) is used to define a real perturbed solution.
In the next section the weak formulation used to solve system (4.118) and associated to conditions (4.119) is recalled. For details in the calculation of the weak formulation and its different steps, see Chapter 2 and Section (2.89).

4.7.2.b Weak Formulation

In order to apply the SBM to the weak formulation of Problem (4.118) one needs first to rewrite the Conditions (4.119) on the surrogate interface $\tilde{h}(x, y, t)$ using Taylor expansions. The conditions in (4.119) concerning the melting temperature at the interface is treated through the average operator (see Definition (1.24)) stating that the temperature at the interface has to be the melting temperature using information from both sides of the interface. In this case the melting temperature is the dimensionless melting temperature, see Equation (4.75).

$$\begin{aligned} &4.121.a)\ [\![T]\!]_h = 0 = \left[\!\!\left[T + \nabla T \cdot \boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t \mathcal{H}(T)\boldsymbol{d} \right]\!\!\right]_{\tilde{h}} + \mathcal{O}(||\boldsymbol{d}(\tilde{\boldsymbol{x}})||^3), \\ &4.121.b)\ [\![\boldsymbol{\beta}]\!]_h \cdot \boldsymbol{n} = \left(\frac{\partial h}{\partial t} + 1\right) \cdot n = [\![\boldsymbol{\beta} + \nabla \boldsymbol{\beta}\ \boldsymbol{d}]\!]_{\tilde{h}} \cdot \boldsymbol{n} + \mathcal{O}(||\boldsymbol{d}(\tilde{\boldsymbol{x}})||^2), \\ &4.121.c)\ \{T\}_h = T_m = \left\{ T + \nabla T\ \cdot \boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^t \mathcal{H}(T)\boldsymbol{d} \right\}_{\tilde{h}} + \mathcal{O}(||\boldsymbol{d}(\tilde{\boldsymbol{x}})||^3). \end{aligned} \tag{4.121}$$

The definition of the flux $\beta = -\lambda \nabla T$ allows to replace the value of ∇T and $\mathcal{H}(T)$ by a term depending on β in the Taylor developments (4.121).

The scheme studied for the numerical analysis is the following:

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

$$2.76.a)\ (\frac{\partial T}{\partial t}, q)_\Omega + (\nabla \cdot \boldsymbol{\beta}, q)_\Omega - < \boldsymbol{\beta} \cdot \boldsymbol{n}, q >_{\Gamma_N} + A_{stab}((T, \boldsymbol{\beta}), q) = 0,$$

$$\begin{aligned} &2.76.b)\ (\lambda^{-1}\boldsymbol{\beta}, \boldsymbol{w})_\Omega - (T, \nabla \cdot \boldsymbol{w})_\Omega + < q_j, \boldsymbol{w} \cdot \boldsymbol{n} >_{\Gamma_{N_j}} + < T, \boldsymbol{w} \cdot \boldsymbol{n} >_{\Gamma_N} + B_{stab}((T, \boldsymbol{\beta}), \boldsymbol{w}) \\ &+ < \{\lambda^{-1}\boldsymbol{\beta} \cdot \boldsymbol{d} + 1/2 \boldsymbol{d}^t \nabla \cdot (\lambda^{-1}\boldsymbol{\beta})\boldsymbol{d}\}_{\tilde{h}}, [\![\boldsymbol{w}]\!]_{\tilde{h}} \cdot \tilde{\boldsymbol{n}} >_{\tilde{h}} + < [\![\lambda^{-1}\boldsymbol{\beta} \cdot \boldsymbol{d} + 1/2 \boldsymbol{d}^t \nabla \cdot (\lambda^{-1}\boldsymbol{\beta})\boldsymbol{d}]\!]_{\tilde{h}}, \{\boldsymbol{w}\}_{\tilde{h}} \cdot \tilde{\boldsymbol{n}} >_{\tilde{h}} \\ &+ N_A(T, q) = - < T_m, [\![\boldsymbol{w}]\!]_{\tilde{h}} \cdot \tilde{\boldsymbol{n}} >_{\tilde{h}} + N_L(q) \end{aligned} \tag{4.122}$$

where $[\![T]\!]_h = 0$ and $\{T\}_h = T_m$. As part of the SBM, conditions are enforced weakly in the weak formulation using the Nitche's method. The terms $N_A(T,q)$ and $N_L(q)$ in (4.122) are defined as follows:

$$
\begin{aligned}
N_A(T,q) &:= \frac{\alpha}{h} < [\![T - \lambda^{-1}\boldsymbol{\beta}\cdot\boldsymbol{d} - 1/2\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{\beta})\boldsymbol{d}]\!]_{\tilde{h}}, [\![q - \lambda^{-1}\boldsymbol{w}\cdot\boldsymbol{d} - 1/2\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{w})\boldsymbol{d}]\!]_{\tilde{h}} >_{\tilde{h}} \\
&+ \frac{\alpha}{h} < \{T - \lambda^{-1}\boldsymbol{\beta}\cdot\boldsymbol{d} - 1/2\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{\beta})\boldsymbol{d}\}_{\tilde{h}}, \{q - \lambda^{-1}\boldsymbol{w}\cdot\boldsymbol{d} - 1/2\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{w})\boldsymbol{d}\}_{\tilde{h}} >_{\tilde{h}}, \\
N_L(q) &:= \frac{\alpha}{h} < T_m, \{q - \lambda^{-1}\boldsymbol{w}\cdot\boldsymbol{d} - 1/2\boldsymbol{d}^t\nabla\cdot(\lambda^{-1}\boldsymbol{w})\boldsymbol{d}\}_{\tilde{h}} >_{\tilde{h}} .
\end{aligned}
\tag{4.123}
$$

A_{stab} and B_{stab} in (4.122) are the stabilisation terms presented in Equation (2.23). Concerning the spaces of approximations in which the weak formulation (4.122) takes place they are the same as defined in Section 2.2.3.b of Chapter 2. They are recalled below

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

such that

$$
\begin{aligned}
\mathcal{Q}_T^2(\Omega) &= \{T \in \mathcal{Q}_T(\Omega) \,|\, T|_K \in \mathbb{P}^2(K),\ \forall K \in \mathcal{T}_h\} \cap (C^0(\tilde{\Omega}^+) \cup C^0(\tilde{\Omega}^-)) , \\
\mathcal{Q}_\beta^1(\Omega) &= \{\boldsymbol{\beta} \in \mathcal{Q}_\beta(\Omega) \,|\, \boldsymbol{\beta}|_K \in (\mathbb{P}^1(K))^d,\ \forall K \in \mathcal{T}_h\} \cap ((C^0(\tilde{\Omega}^+))^d \cup (C^0(\tilde{\Omega}^-))^d).
\end{aligned}
$$

Finally, the following spaces are considered

$$
\begin{aligned}
\boldsymbol{Q}^{2,1}(]0,t_f[,\Omega) &= \mathbb{L}^2(]0,t_f[,\mathcal{Q}_T^2(\Omega)) \times \mathbb{L}^2(]0,t_f[,\mathcal{Q}_B^1(\Omega)) , \\
\boldsymbol{Q}^{2,1}(\Omega) &= \mathcal{Q}_T^2(\Omega) \times \mathcal{Q}_B^1(\Omega) .
\end{aligned}
\tag{4.124}
$$

For more details on the process of definition of the weak formulation (4.122), see Chapter 2.

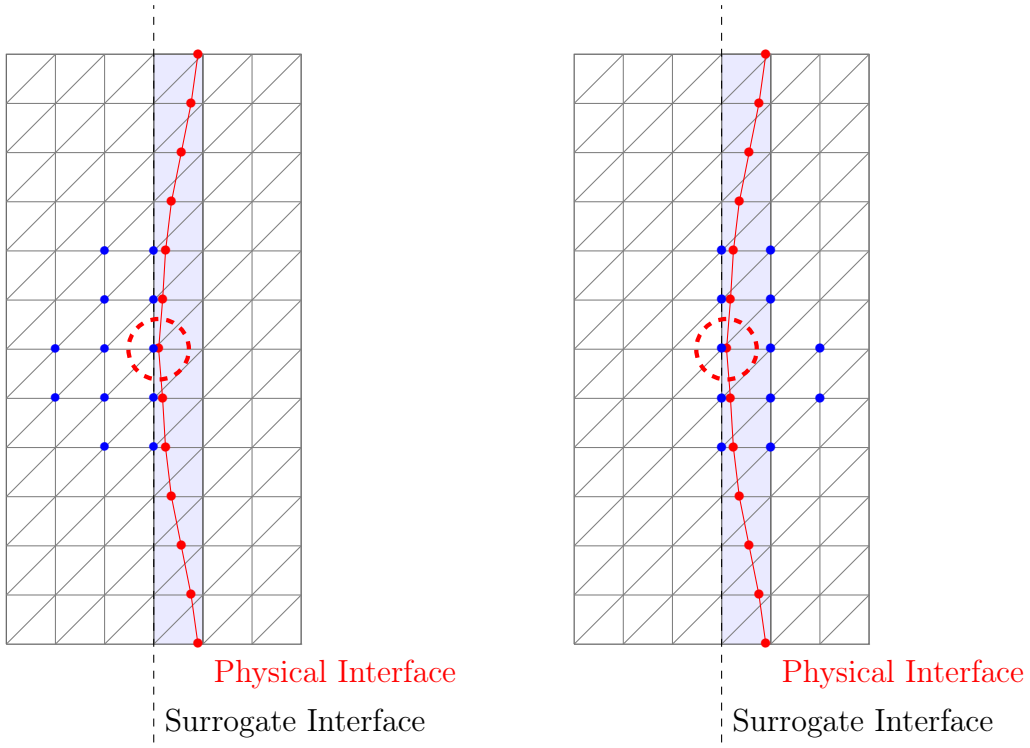
Remark 22. The numerical results presented in the following sections will be presented with the enrichment of the temperature. The global method is called e-SBM. The procedure to define an enriched scheme has been detailed in Section 2.4 of Chapter 2.

4.7.3 Calculation of the interface velocity

A numerical specificity of the Stefan problem is the necessity to express the interface velocity through the calculation of $[\![\beta]\!]_h$, which is also an unknown of the problem. The use of the mixed formulation allows to get an approximation of the flux on mesh nodes, which do not correspond necessarily to nodes on the physical interface, as the interface is not grid conformed to the geometry. To prevent any errors that could come from the mesh generation in the presented stability analysis the focus is made on uniform meshes. A symmetric stencil reconstruction method is proposed to recover β in each phase, see Figures 4.12, 4.13 and 4.14. The method is the same on both sides of the interface. A stencil of nodes is used to reconstruct a polynomial approximation of β_j by solving a system of the form $MX = v$. As an example, for a stencil of 6 nodes and for a $\mathbb{P}^2$ approximation the system $MX = v$ corresponds to :

$$
M = \begin{pmatrix}
1 & x_1 & y_1 & x_1y_1 & x_1^2 & y_1^2 \\
1 & x_2 & y_2 & x_2y_2 & x_2^2 & y_2^2 \\
1 & x_3 & y_3 & x_3y_3 & x_3^2 & y_3^2 \\
1 & x_4 & y_4 & x_4y_4 & x_4^2 & y_4^2 \\
1 & x_5 & y_5 & x_5y_5 & x_5^2 & y_5^2 \\
1 & x_6 & y_6 & x_6y_6 & x_6^2 & y_6^2
\end{pmatrix}
\quad ; \quad
v = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \beta_4 \\ \beta_5 \\ \beta_6 \end{pmatrix}
\quad ; \quad
X = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \end{pmatrix}.
\tag{4.125}
$$

A value is calculated independently for each direction of the flux (x direction and y direction in 2D). The physical interface is numerically discretized as a set of nodes, where the flux jump $[\![\beta]\!]_h$ is used to update the position of each node. The method to obtain a value for β_1 (value in the liquid phase) and β_2 (value in the solid phase) at any nodes of the physical interface is divided into four steps. The size of the stencil depends on the order of the polynomial approximation wanted (at least 6 for $\mathbb{P}^2$), Nevertheless an over-determined system is more stable as refined meshes tend to give matrices with a high condition number as nodes close to each other have similar coordinates. Another specificity of the method is the configuration of the mesh node at



(a) Reference stencil for β_1 (b) Reference stencil for β_2

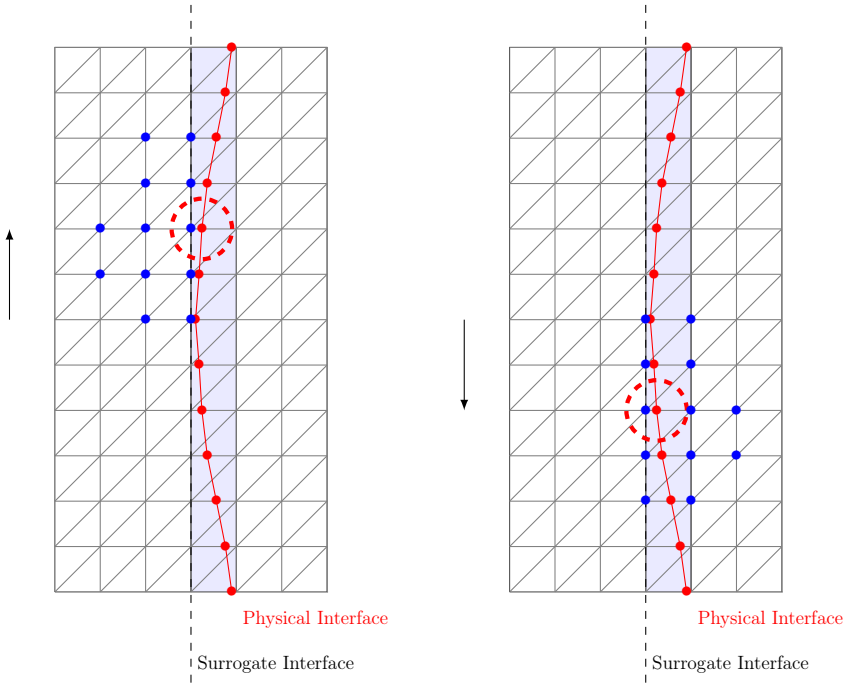
Figure 4.12: Definition of the reference stencil for the reconstruction of the flux at the interface the interface, as defined in Section 2.5 of Chapter 2. As the surrogate interface is on both part of the liquid and solid phase, the mesh nodes at the surrogate interface are associated with two values to allow the definition of $[\![\beta]\!]_{\tilde{h}}$, $[\![T]\!]_{\tilde{h}}$ or $\{T\}_{\tilde{h}}$. For more information on the procedure see Section 2.5.

The method of reconstruction for $[\![\beta]\!]_h$ is decomposed as follows

1. Selection of a **reference stencil** using a node located in the middle of the domain along the physical interface position, called reference node, see Figure 4.12

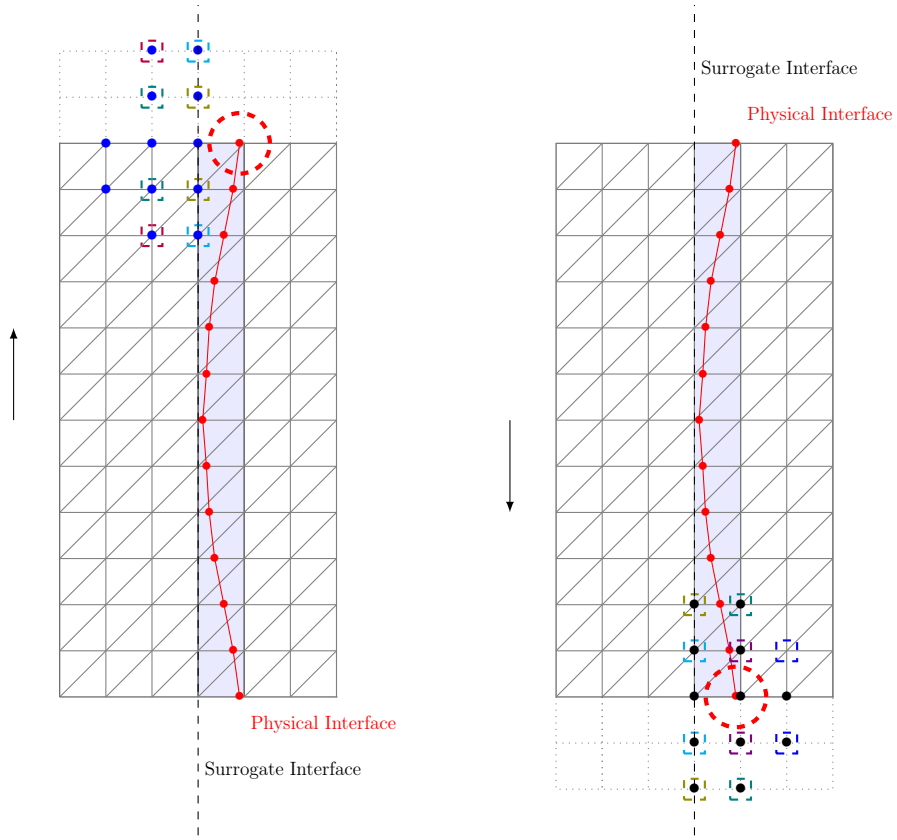
2. Displacement of the **reference stencil** along the surrogate interface in two stages. To the top of the domain for the nodes with a y coordinate higher than the reference node used in step 1, and then displacement to the bottom of the domain for the other nodes, see Figure 4.13

3. Near the top and bottom boundaries of the domain the stencil will have nodes outside of the domain of resolution, which is not meshed. The problematic nodes are treated using the coordinates outside of the domain (see matrix M in 4.125) but using the value of β associated to their mirror values inside the domain (see vector v in 4.125), see Figure 4.14.



(a) To the top of the domain (b) To the bottom of the domain

Figure 4.13: Visualization of the stencil displacement



(a) Symmetric stencil - Top of the domain

(b) Symmetric stencil - Bottom of the domain

Figure 4.14: Symmetric Stencil - Mirror image reconstruction

4. Calculation of a polynomial approximation solving system (4.125) for each node defining the physical interface, using the QR factorization. This configuration allows to mimic the definition of symmetric boundary conditions. In comparison with the stability of the continuous model, it is similar to considering a domain infinite in the y-direction.

In the next section, the numerical stability associated to the resolution of the mixed formulation of the Stefan model is presented.

4.7.4 Numerical Stability Analysis

Symbol	Value	Definition
Tm	$273.15(K)$	Melting temperature
ρ	$1000(Kg.m^{-3})$	Density of the material
Lm	$333000(J.Kg^{-1})$	Latent heat
λ_1	$0.6(W.m^{-1}.K^{-1})$	Thermal conductivity - liquid phase
λ_2	$2.1(W.m^{-1}.K^{-1})$	Thermal conductivity - solid phase
c	$2060(J.Kg^{-1}.K^{-1})$	Heat capacity
k	$10(m)$	Wavenumber
u	$1e^{-4}(m.s^{-1})$	Velocity of the solid
Δt	$\frac{\frac{\Delta_x \Delta_y}{2}}{4}$	Time step defined through the space discretization Δ_x,Δ_y

Table 4.2: Definition of the physical parameters for variables with dimensions

For the results presented in this section the variables of the model are defined in Table 4.2 and in their dimensionless form using the change of variables (4.75) in Table 4.3. The parameters are shown both in dimensional and dimensionless forms to facilitate the comprehension of the reader on the meaning of the different parameters. The parameters are chosen to represent to a configuration between liquid and solid water. However, all of the figures in this section are displayed with the dimensionless parameters from Table 4.3. The definition of the flux q_j is only expressed in its dimensionless form, as it is chosen to satisfy condition (4.119) where $q_1 - q_2 = 1$.

Symbol	Value	Definition
$\hat{Tm}$	1.6898	Dimensionless melting temperature
$\hat{\rho}$	1	Dimensionless density
$\hat{Lm}$	1	Dimensionless latent heat
$\hat{\lambda}_1$	0.2857	Dimensionless thermal conductivity - liquid phase
$\hat{\lambda}_2$	1	Dimensionless thermal conductivity - solid phase
$\hat{c}$	1	Dimensionless heat capacity
$\hat{k}$	0.1019	Dimensionless wavenumber
$\hat{u}$	1	Dimensionless velocity of the solid
$\hat{q}_1$	1.5	Incoming flux - dimensionless value
$\hat{q}_2$	0.5	Outgoing flux - dimensionless value

Table 4.3: Dimensionless parameters associated to Table 4.2

A flux q_1 greater than 1 in the liquid phase represents the addition of heat into the system through the liquid phase. This choice has demonstrated to be stable for the stability analysis of

the continuous model in Section 4.6 as, displayed in Figure 4.4. A non uniform discretization in x and y is considered, i.e., that $\Delta x \neq \Delta y$ such that the time step Δt is defined by $\Delta t = \frac{\Delta x \Delta y}{2}/4$. Δx is chosen larger than Δy to define more mesh nodes further away from the interface location used for the definition of the different stencils used for reconstruction procedures.

4.7.4.a Interface Perturbation

Some improvements in the interface displacement method are made compared to results presented in Chapter 3. Previously the method allowed the displacement of the interface only through a homogeneous motion using a smoothing method and the average speed velocity at the physical interface. Taking into account the shape of the perturbed interface for the presented stability analysis (combinations of cosine and sine waves), this method was not usable by itself and some improvements have been made in the identifications process of the elements cut by the physical interface, allowing to get a full unstructured motion.

The procedure now states as below :

- The physical interfaces are discretized as a set of nodes. On rectangular domains the number of nodes is at least equal to the number of elements in the y-direction.
- The projection is performed on all of the edges defining the physical interface, if the projection is not unique the one minimizing the distance is chosen; the numerical precision is often enough to define a unique minimizer. In the case of multiple minimizer the first one identified by the algorithm is chosen .
- In the case of no projection, the common node between the two edges where the projection was identified is chosen, see Section 1.2.2 and Figure 1.16 for more details.
- A mesh element is defined cut by the physical interface if the value of the distance between the vertices of the mesh element and their respective projection have a different signs. Here the distance function is a sign distance function. If the distance to the interface is a positive number then a node is located in the liquid phase while if the distance is a negative number the node is located in the solid phase .

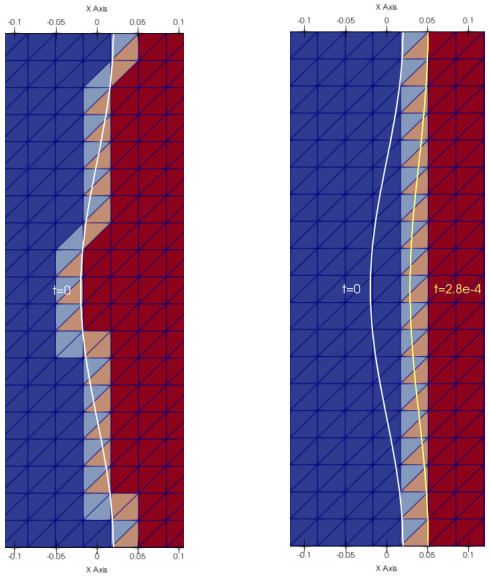
Remark 23. The projection of a point into the physical interface stays the same as defined in Chapter 1, see Section 1.2.2. The improvement of the method involves in a different procedure for the assignation of a sign to the distance between a node and its projection. The sign distance function is calculated using the cross product between the vector representing the edge of the interface where the projection has been found and the vector of the projection onto that edge. The cross product of these two vectors result in a scalar value.

If that value is

1. positive : the two vectors have a counter-clockwise relationship,
2. negative: the two vectors have a clockwise relationship,
3. zero : the two vectors are collinear.

This method allows to treat more complex interface geometries and will show its robustness in the results presented in this Chapter.

In Figure 4.15 the identification of the elements cut by the interface is presented for two different times t. Comparing Figure 4.15.a and 4.15.b one can see that the perturbation ϵ at the interface is decreasing in amplitude and that the geometry of the two surrogate interfaces is different.



(a) Initial Interface Perturbation, $t = 0$

(b) Position of the interface at time $t = 2.8e - 4$

Figure 4.15: Identification of the Surrogate through the simulation - $\epsilon = 1e - 2$

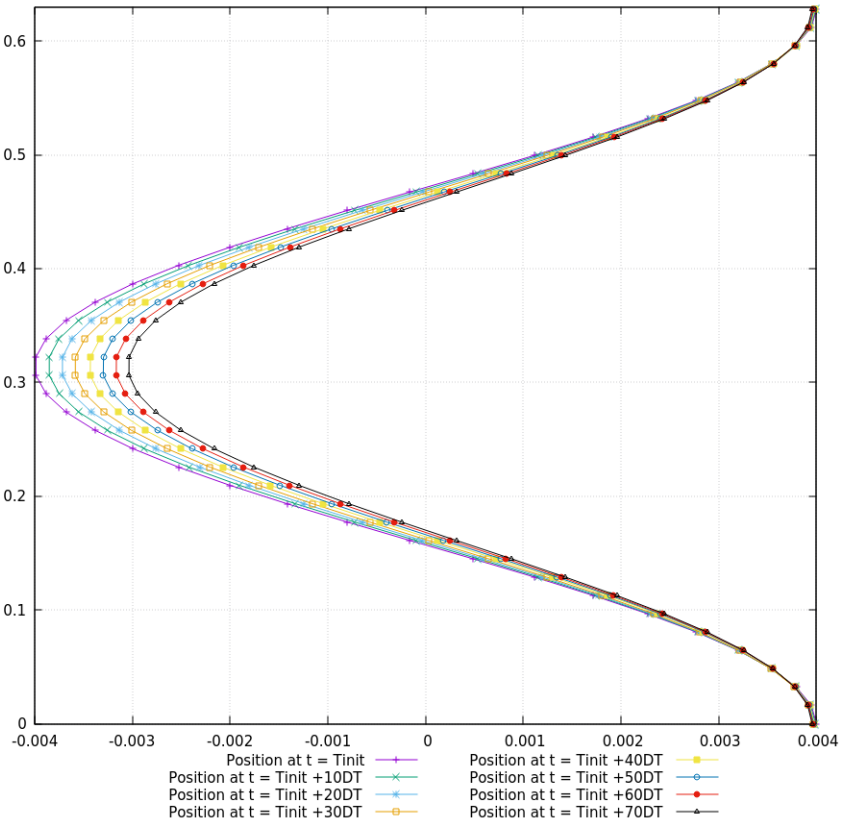
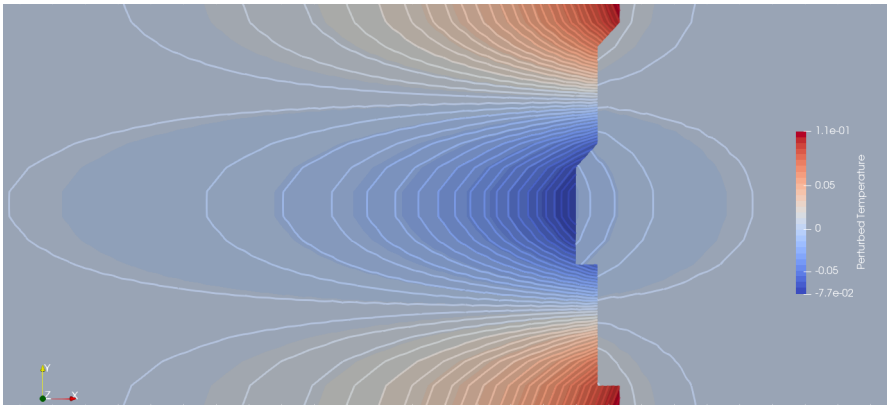


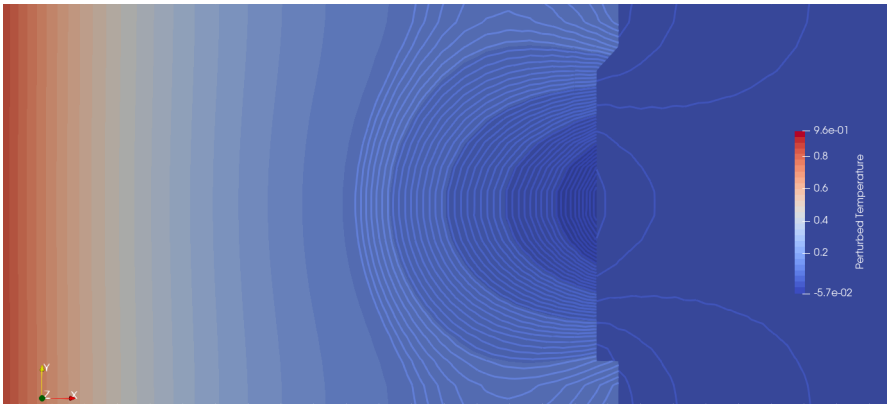
Figure 4.16: Comparison of the interface position at different iterations - $\epsilon = 2e{-}3$, $\Delta t = 1.7e{-}3$

In Figure 4.16 the position of the perturbation at the interface is compared for different times t. All the curves have been translated to match the x coordinate of the initial amplitude position (purple curve in Figure 4.16). This allows us to see more clearly the change in shape occurring at the interface. In Figure 4.16 the same results as in Figure 4.15 can be observed and a decrease in amplitude in the interface perturbation is visible, without any instability. This result shows the robustness of the method proposed in this manuscript for moving interfaces (e-SBM, see Chapter 2).

4.7.4.b Temperature Perturbation



(a) Initial perturbed temperature, $t = 0$



(b) Perturbed temperature after 10 time steps, $\Delta t = 1.7e-3$, $t = 0.0170$

Figure 4.17: Evolution of the profile of the perturbed temperature for different iterations Figure 1/2

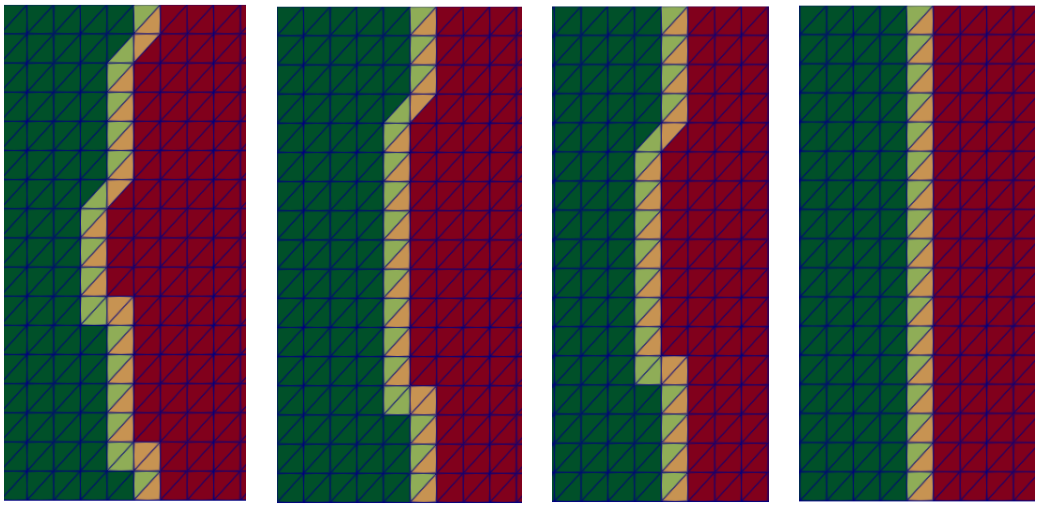
In Figures 4.17 and 4.18 the dissipation of the perturbation added to the temperature is presented for the interface defined in Figure 4.15. The perturbation of the temperature follows the behavior observed in Figure 4.15 and 4.16 concerning the interface position. Isolines are used to follow the behavior of the solution, where no growth in the perturbation is visible through the simulation. In Figure 4.18.b) the isolines are now parallel to each other and the perturbation added on the temperature field is not visible anymore. Figures 4.17 and 4.18 show the temperature calculated by the scheme (4.122) minus the initial temperature field at time $t = 0$, see Figure 4.17.a). On the left side of figures 4.17.b), 4.18.a) and 4.18.b) the error visible is coming from the position of the physical interface which is used in the condition imposed on the left boundary of the domain of resolution. The interface position being different from the

initialization at time $t = 0$. As the physical boundary is moving an error is created in the representation of the perturbed temperature. However, Figures 4.17 and 4.18 allow us to visualize properly the dissipation of the perturbation at the interface, addressing the concerning point and once again demonstrating the stability of the e-SBM.

(a) Perturbed temperature after 50 time steps, $\Delta t = 1.7e-3$, $t = 0.085$

(b) Perturbed temperature after 340 time steps, $\Delta t = 1.7e-3$, $t = 0.578$

Figure 4.18: Evolution of the profile of the perturbed temperature for different iterations Figure 2/2



(a) t_{init} (b) $t + 20\Delta t$ (c) $t + 30\Delta t$ (d) $t + 50\Delta t$

Figure 4.19: Identification of the mesh elements cut by the physical interface

In Figure 4.19 the different configurations of the surrogate interface for the simulation presented

in Figures 4.17 and 4.18 are displayed. It allows for visualizing the decrease in amplitude of the perturbation added to the interface position in the definition of the surrogate interface. It also shows that the surrogate definition goes back to a flat interface structure, see Figure 4.19.d.

4.7.4.c Convergence Study

The perturbed temperature $T_{j,1}(x, y, t)$ is defined in terms of an exponential function representing a decreasing perturbation to 0 at the left and right boundaries of the domain, see Section 4.6.3. The definition of $T_{j,1}(x, y, t)$ gives a criterion on the accuracy of the numerical method. The curve representing the amplitude of the interface position, defined by $|min(h(y,t)) - max(h(y,t))|$ decreases with a slope equals to the value of s, the frequency associated to the chosen wavenumber k and its complex conjugate $-k$ for the definition of a real function, see Section 4.6.5.

To look at the accuracy of the e-SBM applied to the Stefan model a convergence study on the slope associated to the amplitude of the interface position is performed.
A domain of height $\frac{2\pi}{k}$ and of length $8 \times \frac{2\pi}{k}$ is considered, where $k = 10$ and $q_1 = 1.5$. For the discretization of the domain nDx denotes the number of elements in the x-direction, nDy the number of elements in the y-direction, nbE the number of elements of the considered mesh, $sCal$ the calculated value of s and erS the relative error between the calculated value and the analytical value of s, here $\boldsymbol{s = -16.8858}$ (see Figure 4.4 for the corresponding value of s in term of k and q_1).

nDx	nDy	nbE	Δ_x	Δ_y	sCal	L^2 erS
100	12	1200	0.05026	0.05236	-12.126	0.28186
125	15	1875	0.04021	0.04188	-14.221	0.15784
150	18	2700	0.03351	0.03491	-14.947	0.11480
200	25	5000	0.02513	0.02513	-15.986	0.05328
250	30	7500	0.02011	0.02094	-16.314	0.03388
300	36	10080	0.01675	0.01745	-16.507	0.02243

Table 4.4: Convergence study on the frequency s

Figure 4.20 shows the relative error between the calculated value of s expressed in Table 4.4 and the analytical value of $s = -16.8858$ in term of Δ_x and Δ_y (non homogeneous discretization). The curves are defined using the values associated to the convergence study on the amplitude of the interface position. The values are displayed in Table 4.4 and showed in Figure 4.21 in standard and log scale. The slope on the error on s in Figure 4.20 is calculated through the polyfit function in Matlab and gives **a slope of 2.2989** for the blue curve and **2.2834** for the red curve, allowing the conclusion that once again the **e-SBM** is a second order accurate method.

In Figure 4.21 one can see the comparison between the evolution of the amplitude of the interface position through time for different mesh discretizations. Some oscillations are visible on the curves representing the case ($nDx = 100$,$nDy = 12$) and the case ($nDx = 125$,$nDy = 15$). Those small oscillations correspond to the displacement of the surrogate interface and the reconstructions used to update the previous solution vectors when new elements of the mesh are crossed. Nevertheless, this phenomenon disappears at convergence and the curves are converging towards the analytical solution of s (the red curve in Figure 4.20 corresponds to the function $f(x) = 0.008exp(sx)$). The standard scaling in Figure 4.21(a) is displayed to observe the exponential decay of the amplitude of the perturbed interface.

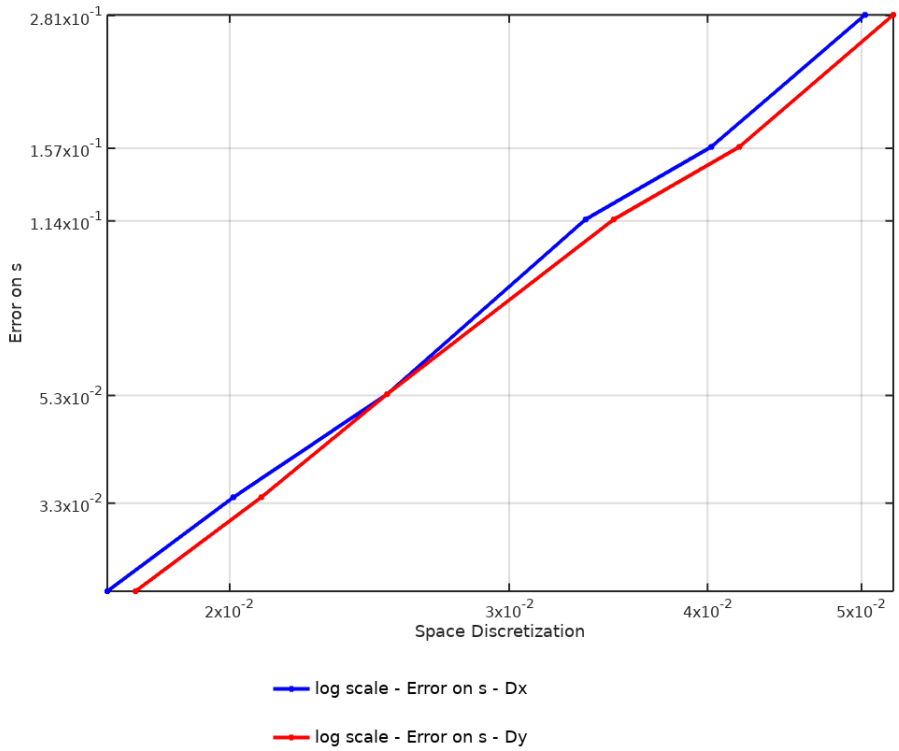
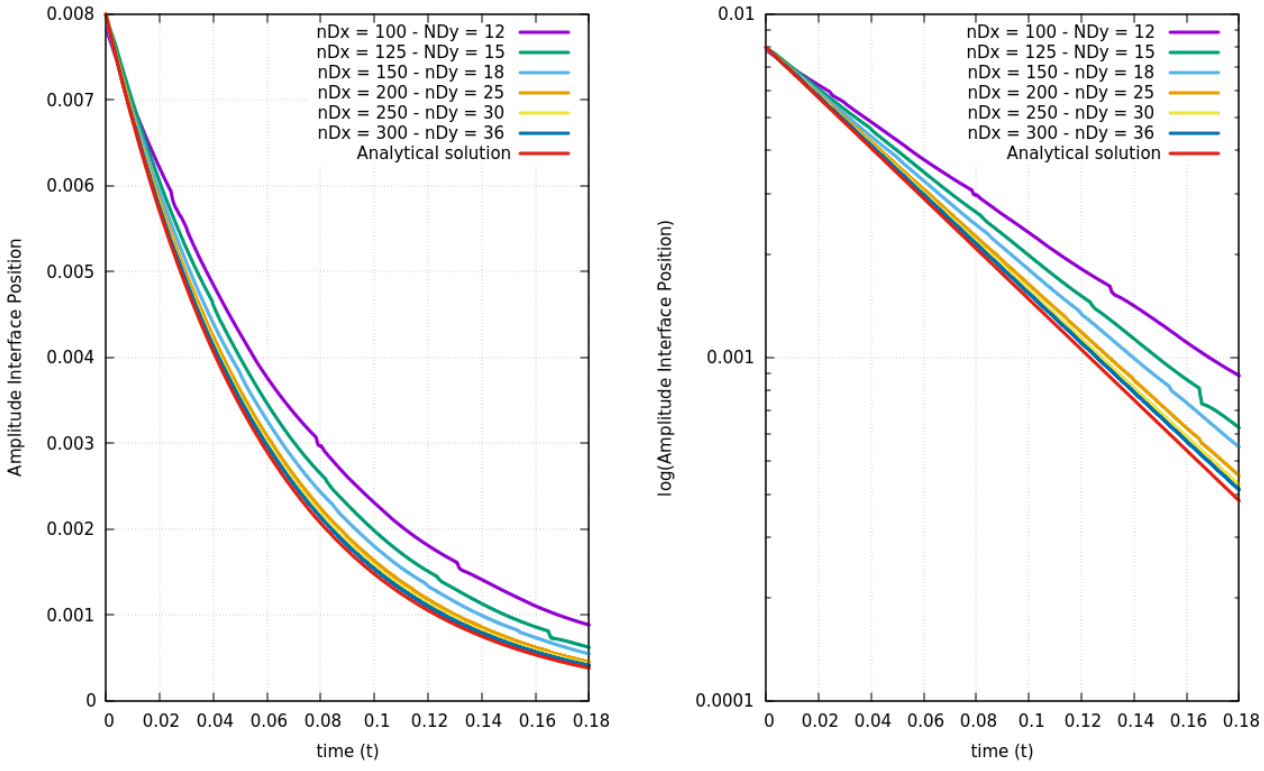


Figure 4.20: Error in log scale on the calculation s - for the discretization following the x-coordinate in blue and y-coordinate in red



(a) Convergence study on the amplitude of the interface position - Standard scaling

(b) Convergence study on the amplitude of the interface position - Log scale

Figure 4.21: Visualization of the convergence study for the calculation s for two different scaling

The results presented in this section have shown that the numerical model used to solve the Stefan model in the context of a melting simulation is stable and allows to deal with complex interface geometries. The stability analysis on the numerical model is in adequacy with the stability analysis performed in Section 4.6 for the continuous model. Moreover, the expected accuracy brought by the e-SBM is conserved, i.e., the second order accuracy.

In the next section, a self-made steady solution is defined on a circular domain. The non perturbed interface is a single radius value. The interface position will then be perturbed and updated through the simulation using the flux jump $[\![\beta]\!]_h$. The test case will show the ability of the method to converge to a steady position and the dissipation of the perturbation introduced at the interface.

4.8 Application Test

In this section, results on a perturbed interface for a circular configuration are presented. A steady self-made solution in polar coordinates is defined and the velocity at the interface is recovered through the flux jump at the interface. The points from the discretization of the interface are moved independently using the numerical solution of the Stefan condition, see Section 2.6.1. The different results presented in this section will show that the perturbed front converge to a steady position where the interface is a circle.

4.8.1 Test Case Definition

For the definition of the test presented in this Section 4.8 polar coordinates are used, i.e.,

$$(x, y) = (rcos(\theta), rsin(\theta)), \tag{4.126}$$

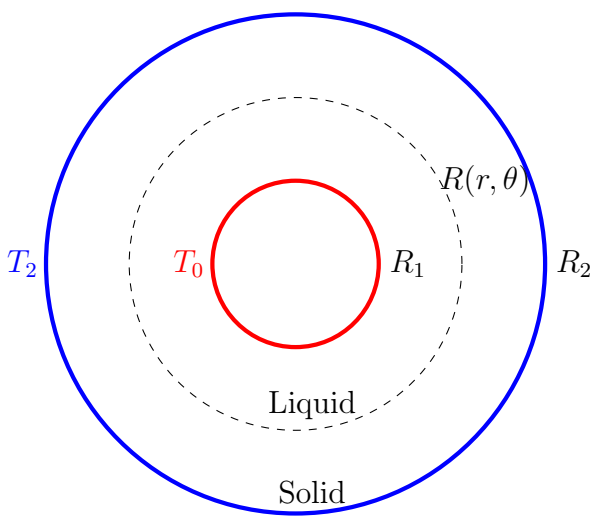
such that

$$r = \sqrt{x^2 + y^2}, \tag{4.127}$$

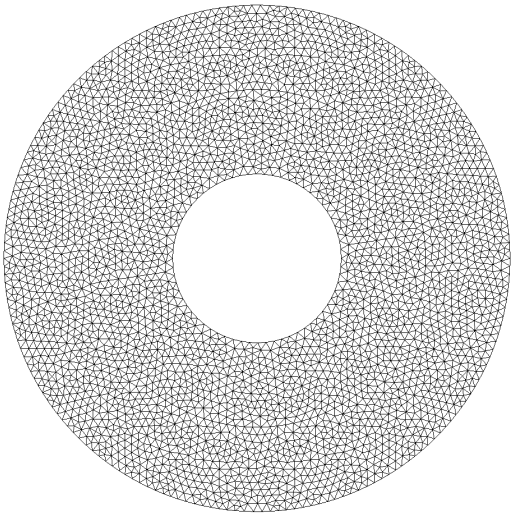
and

$$\theta = \begin{cases} arctan\left(\dfrac{y}{x}\right) & \text{if } x > 0 \ ; \ y \geqslant 0, \\ arctan\left(\dfrac{y}{x}\right) + 2\pi & \text{if } x > 0 \ ; \ y < 0, \\ arctan\left(\dfrac{y}{x}\right) + \pi & \text{if } x < 0, \\ \dfrac{\pi}{2} & \text{if } x = 0 \ ; \ y > 0, \\ \dfrac{3\pi}{2} & \text{if } x = 0 \ ; \ y < 0. \end{cases} \tag{4.128}$$

The domain of resolution is a circle (see Figure 4.22), the interface is denoted by $R(r, \theta)$, the inner circle by R_1 and the outer circle by R_2. The circular domain in Figure 4.22 is defined with a hole inside to allow the imposition of a Dirichlet condition for the liquid phase. The mesh used to show the evolution of the perturbed front is defined in Figure 4.22.b).



(a) Domain of resolution without perturbation at the interface and the imposition of the Dirichlet conditions



(b) Mesh Discretization / $Ne = 3480$ / $Nv = 6706$

Figure 4.22: Illustration of the test case

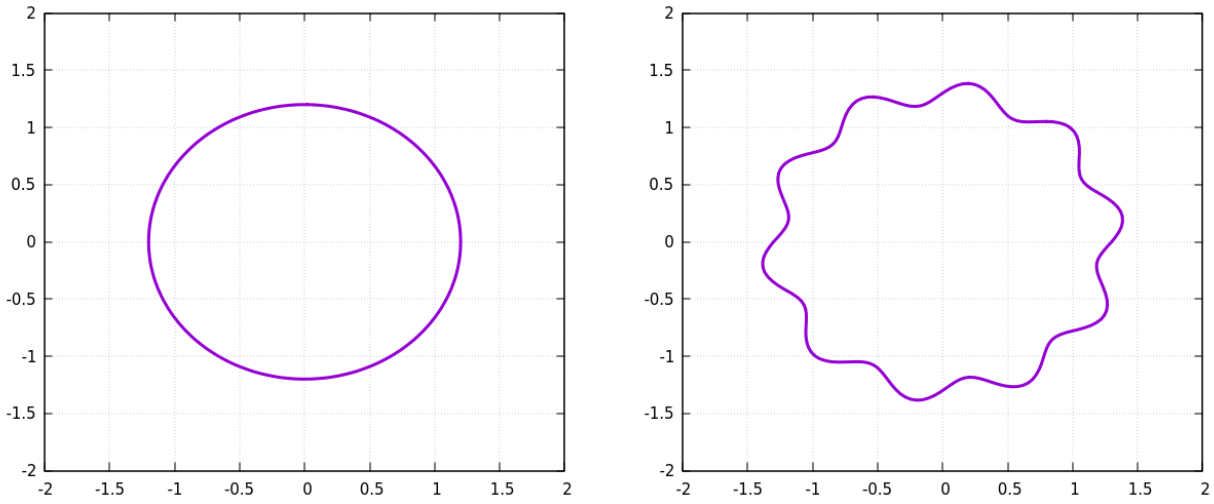
In Figure 4.22.b Ne represents the number of elements of the mesh and Nv the number of vertices. The initial non perturbed interface is defined as a circle of radius equals to 1.2 ($R(r,\theta)$ in Figure 4.22). The discretization of the physical interface leads to a set of nodes. Those nodes are associated with a position (r,θ). The initial interface is then perturbed using the following algorithm:

1. Initialization:

   $\boldsymbol{\theta} = 0$
   $\boldsymbol{n}$ = Number of nodes of the discretization of the physical interface
   $\boldsymbol{d\theta} = \frac{2\pi}{n}$
   $\boldsymbol{Interface\%Coor}(n, 1:2)$ = Structure containing the points of the discretization of the physical interface in cartesian coordinates

2. Perturbation of the Interface:

   DO $\boldsymbol{i} = 1, n$
   $\boldsymbol{r} = 0.1sin(10\theta) + 1.2$
   $\boldsymbol{Interface\%Coor}(i,1) = rcos(\theta)$
   $\boldsymbol{Interface\%Coor}(i,2) = rsin(\theta)$
   $\boldsymbol{\theta} = \theta + d\theta$
   END DO



(a) Interface Position before Perturbation

(b) Interface Position after Perturbation

Figure 4.23: Position of the interface before and after perturbation using the algorithm of Section 4.8.1

The self-made solution presented in Equation (4.129) is used to initialize the solution on the domain of resolution shown in Figure 4.22.a). The primal variable T is defined by:

$$T = \begin{cases} T_1(r,\theta) = T_m + (T_0 - T_m)\dfrac{ln\left(\frac{r}{0.1sin(10\theta)+1.2}\right)}{ln\left(\frac{R_1}{0.1sin(10\theta)+1.2}\right)}, & \text{Liquid Phase} \\ \\ T_2(r,\theta) = T_m + (T_2 - T_m)\dfrac{ln\left(\frac{r}{0.1sin(10\theta)+1.2}\right)}{ln\left(\frac{R_2}{0.1sin(10\theta)+1.2}\right)}, & \text{Solid Phase.} \end{cases} \tag{4.129}$$

For the flux, its definition depends on the definition used for θ (see Equation (4.128)) and is defined by calculating the gradient of T, the temperature field defined in Equation (4.129). We recall that β is defined by two components in the x and y directions. Both of them are modified using the change of variables (4.126).

If $\left(\theta = arctan\left(\frac{y}{x}\right)\right)$ or $\left(\theta = arctan(\frac{y}{x}) + 2\pi\right)$ or $\left(\theta = arctan\left(\frac{y}{x}\right) + \pi\right)$ then

$$\beta_1 = \begin{pmatrix} -\lambda_1(T_0 - T_m)\dfrac{(Du_x v - uDv_x)}{v^2} \\ -\lambda_1(T_0 - T_m)\dfrac{(Du_y v - uDv_y)}{v^2} \end{pmatrix} ; \quad \text{Liquid Phase}$$

$$\beta_2 = \begin{pmatrix} -\lambda_2(T_2 - T_m)\dfrac{(Du_x v - uDv_x)}{v^2} \\ -\lambda_2(T_2 - T_m)\dfrac{(Du_y v - uDv_y)}{v^2} \end{pmatrix} ; \quad \text{Solid Phase} \tag{4.130}$$

If $\left(\theta = \frac{\pi}{2}\right)$ or $\left(\theta = \frac{3\pi}{2}\right)$ then

$$\beta_1 = \begin{pmatrix} -\lambda_1 \left(\dfrac{(T_0 - T_m)}{ln\left(\frac{R_1}{0.1sin(10\theta)+1.2}\right)} \dfrac{cos(\theta)}{r} \right) \\ -\lambda_1 \left(\dfrac{(T_0 - T_m)}{ln\left(\frac{R_1}{0.1sin(10\theta)+1.2}\right)} \dfrac{sin(\theta)}{r} \right) \end{pmatrix} \; ; \quad \text{Liquid Phase}$$

$$\beta_2 = \begin{pmatrix} -\lambda_2 \left(\dfrac{(T_2 - T_m)}{ln\left(\frac{R_2}{0.1sin(10\theta)+1.2}\right)} \dfrac{cos(\theta)}{r} \right) \\ -\lambda_2 \left(\dfrac{(T_2 - T_m)}{ln\left(\frac{R_2}{0.1sin(10\theta)+1.2}\right)} \dfrac{sin(\theta)}{r} \right) \end{pmatrix} \; ; \quad \text{Solid Phase} \tag{4.131}$$

where

$$Du_x = \frac{cos(\theta)}{r} + sin(\theta)\frac{cos(10\theta)}{r(0.1sin(10\theta)+1.2)} \quad ; \quad Du_y = \frac{sin(\theta)}{r} - cos(\theta)\frac{cos(10\theta)}{r(0.1sin(10\theta)+1.2)},$$
$$Dv_x = cos(10\theta)\frac{sin(\theta)}{r(0.1sin(10\theta)+1.2)} \quad ; \quad Dv_y = -cos(10\theta)\frac{cos(\theta)}{r(0.1sin(10\theta)+1.2)}. \tag{4.132}$$

Before presenting some results associated to the analytical solutions (4.129), (4.130) and (4.131), the parameters of the simulation are defined in Table 4.5.

The time step in this simulation is the same as defined in Chapter 3, where $\Delta t = \frac{h^2}{min(\lambda_1,\lambda_2)}$, and where the value of h associated to the mesh discretization defined in Figure 4.22.b is $h = 0.1$.

Symbol	Value	Definition
R_1	1	Inner circle
R_2	2.1	Outer circle
L_1	4185	Heat capacity - Liquid phase
L_2	2060	Heat capacity - Solid phase
ρ	1	Density
L_m	3000	Latent Heat
T_0	268.15	Boundary condition - Temperature imposed on the inner circle R_1
T_2	278.15	Boundary condition - Temperature imposed on the outer circle R_2
T_m	273.15	Melting temperature
λ_1	0.6	Thermal conductivity - Liquid phase
λ_2	2.1	Thermal conductivity - Solid phase
Δt	0.67	Time Step
T_{init}	0	Initial time of the simulation

Table 4.5: Parameters for the simulation of a perturbed front on a circular domain

At the initialization the profile of the perturbed temperature $T_{j,0} + \epsilon T_{j,1}$ defined in Equation 4.129, is used as an initial condition and is displayed in Figure 4.24.

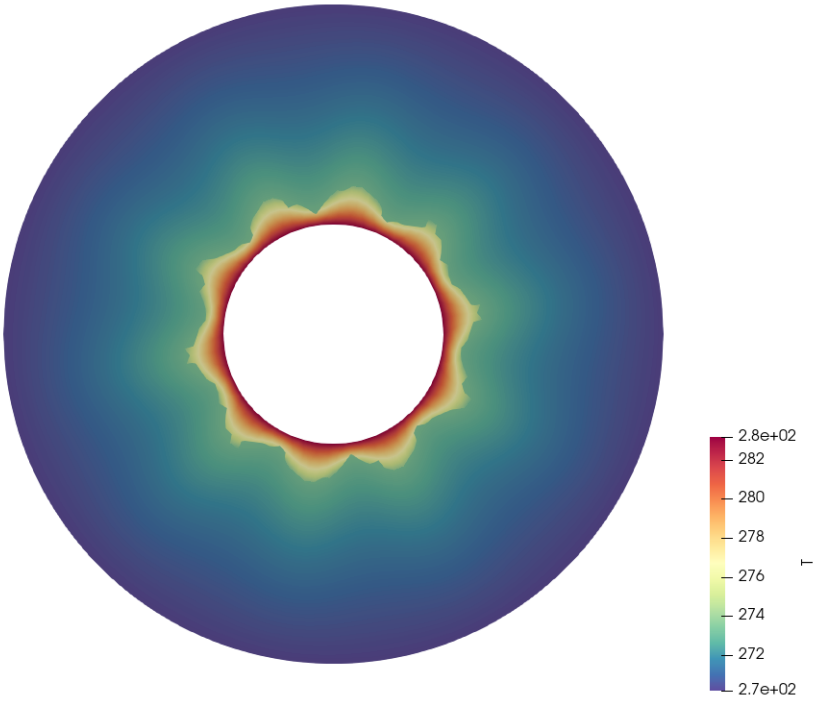
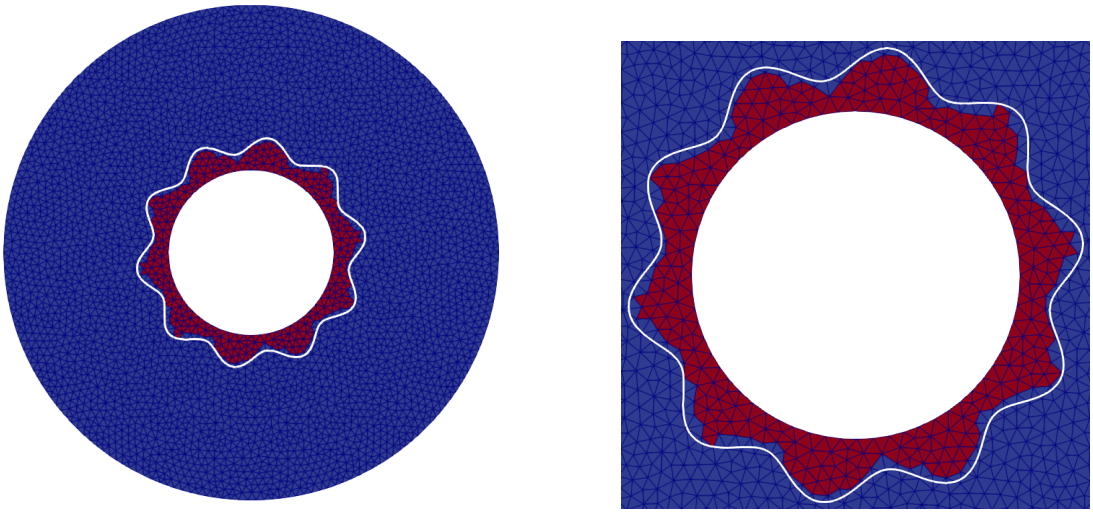


Figure 4.24: Profile of the temperature at the initialization, non dimensionless form

The results are presented for variables with dimensions, as the purpose of this section is not to analyse the numerical method but to show numerically the stability of the e-SBM.

In Figure 4.25(a) the red area represents the liquid phase while the blue area is the solid phase. The common interface between the two phases is the set of edges defining the surrogate interface.



(a) Identifications of the liquid and solid zone

(b) Identification of the surrogate interface

Figure 4.25: Visualization of the initial configuration of the domain and the definition of the surrogate interface within the mesh

In Figure 4.25(b), on can see a zoom around the inner circle of the domain presented in Figure 4.25(a). The surrogate interface is defined as the inner option of the area created by the elements cut by the physical interface (in 2D two options are possible for the surrogate definition, see

Section 1.2.2). In the next section, the behavior of the perturbed front (see Figure 4.23.b)) is studied, only the interface position is perturbed.

4.8.2 Results on the evolution of a perturbed front

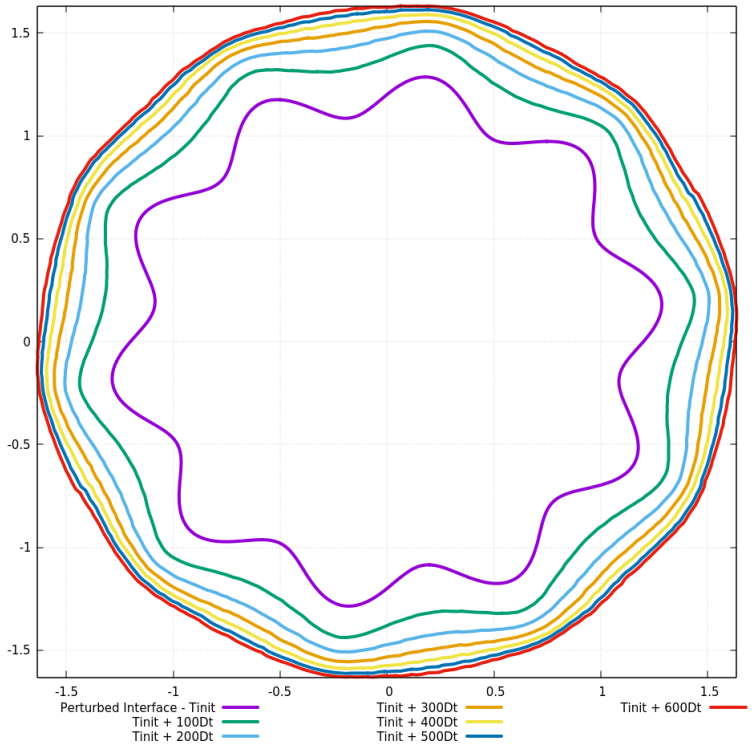


Figure 4.26: Evolution of the physical interface profile

In Figure 4.26 the position of the front is displayed at different times. The purple curve is the initial perturbed interface position introduced in Figure 4.22.b). The velocity at the interface is recovered from the Stefan condition and calculated for every points of the discretization. The interface is discretized as a set of nodes, here 400 nodes are used. Neither instabilities nor a growth of the initial perturbation is visible for the different iterations displayed in Figure 4.26. The last iteration ($t_{init} + 600\Delta t$, the red curve) is also displayed on the temperature field in Figure 4.27 and for the profile of the flux in x in 4.28.a) and in y in 4.28.b). Figures 4.27 and 4.28 show the robustness of the method in the calculation of the temperature and flux with a moving front, no instabilities are visible in the different figures. In this section the ability of the method to disperse perturbations, and more generally the stability regarding large perturbations have been demonstrated. The results presented have also shown the ability to move the interface with a complete unstructured motion compared to results presented in Chapter 3. The method of reconstruction used for the expansion of one of the phases (liquid phase for a melting configuration) showed once again that the method does not impact the resolution and has no impact on the stability of the method.

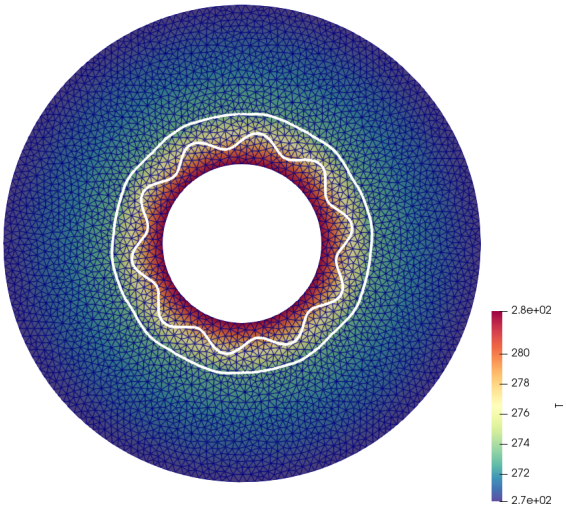
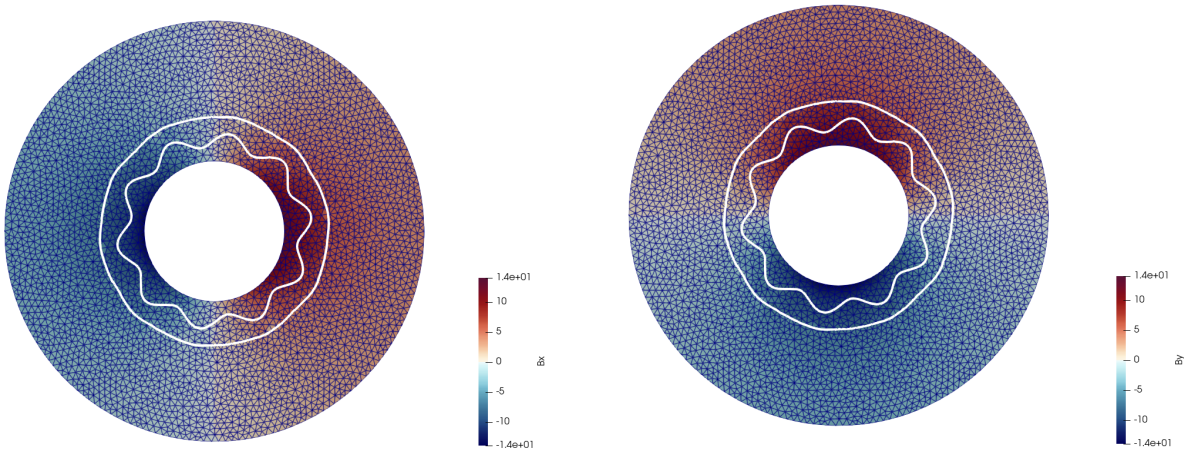


Figure 4.27: Profile of the temperature and the interface position at initialization and after 600 iterations



(a) x component of the flux

(b) y component of the flux

Figure 4.28: Profile of the flux and the interface position at initialization and after 600 iterations

4.9 Conclusion on Chapter 4

In this Chapter 4, a linear stability analysis has been performed on the Stefan model and the scheme associated with its resolution using the e-SBM. Both the continuous model and the scheme proposed for its resolution have shown to be stable for a melting configuration, i.e., with the addition of heat into the system. Complete unstructured motion was tested for different interface geometries, and stability was consistently demonstrated. This chapter has demonstrated the robustness and stability of the Shifted Boundary method in the context of its application to moving interfaces tested in this manuscript on the Stefan model.

Manuscript Conclusion

1 Contributions

In this manuscript, the **Shifted Boundary Method** (SBM), a Finite Element Method (FEM) used for the resolution of problems with embedded boundaries has been **extended** to simulations **involving moving boundaries**. The SBM is a method allowing the imposition of conditions on a numerical boundary composed of mesh edges (or faces) of mesh elements crossed by the physical boundary on a background fixed mesh. Taylor expansions are used to modify the initial boundary conditions according to the choice of the numerical boundary called surrogate boundary. The **Nitsche's method** is used within the SBM approach to **impose** the boundary conditions **weakly** into the FEM scheme.

The proposed method, referred to as e-SBM for **enriched Shifted Boundary Method**, has been applied in the context of the **Stefan model** for **two-phase situations** where the moving boundary is a moving interface which evolves according to the simulation and the computation of the Stefan condition. In this context, the interface represents a **melting front** moving with the expansion of the liquid phase. The **mixed formulation** of the Stefan problem has been proposed to **improve the accuracy of the gradients computation**.

The mixed formulation consists in the introduction of the gradient of temperature as an explicit unknown solved by the linear system associated with the resolution of the Finite Element Method. The enriched SBM (e-SBM) has demonstrated its ability to solve these types of problems with a **second-order accuracy** in space for both the temperature field and its flux, in time and for the interface motion. The mixed formulation has been chosen to enable the use of an enriched $\mathbb{P}^1$ Lagrange FEM allowing an enhancement of the accuracy without increasing the size of the linear system to be solved. It has consisted in an approximation of $\mathbb{P}^2$ Lagrange FEM using only $\mathbb{P}^1$ basis/test functions and $\mathbb{P}^1$ degrees of freedom. Two sets of interface conditions for the moving interface and their corresponding finite weak formulations have been described and verified through analytical test cases in this thesis manuscript. This work has helped to identify a better-suited choice in terms of precision and ease of implementation for the imposition of interface conditions in the scheme. The most favorable set of interface conditions involves imposing the melting temperature at the interface, with a temperature jump equal to zero and an average temperature equal to the melting temperature.

The enriched SBM (e-SBM) has demonstrated, on a realistic in-flight melting configuration, its ability to solve Stefan problems with a second-order accuracy in space, time and for the interface motion from the discretization of the Stefan condition. Procedures for reconstructing missing values due to the phase expansion and the moving front have been tested to ensure that they have no impact on the accuracy of the method. A stability analysis of the continuous Stefan model and of the numerical e-SBM have been studied in the context where the **front**

position and the **temperature field** are **both perturbed** following a **linearization** of the problem around a **steady state**. A change in the referential, in the primal form of the continuous Stefan model, has been made to introduce a term of advection. This term allows for the definition of the solid velocity pulled away from the domain in the system of equations. This particular form allowed to define a linear perturbed solution used as initial condition for the stability analysis of the numerical model. For a melting configuration, the continuous Stefan model has been shown to be stable.

Results presented in this manuscript have demonstrated that the e-SBM conserves the stability property of the continuous model, indicating the robustness of the method and its ability to handle perturbations. This property is highly valuable as it suggests that the model will naturally tend to stabilize spurious oscillations that might occur with the moving nodes discretizing the melting front.

2 Perspectives of evolution

In this work, the choice of **discretizing the interface position** as a set of nodes has allowed for the definition of an easy way to update the interface position. The use of a Level-Set method to identify the interface position was not investigated to avoid adding complexity to the method's development. However, developing the e-SBM with the Level-Set approach is a possible subject of development. It will be necessary to find a way to expand the definition of the velocity field outside the interface region, which, for the moment, has not been investigated. Many methods used for solving melting problems incorporate a Level-Set approach [125, 126, 127, 128, 129] and could potentially integrate the reconstructions and enrichment proposed in this manuscript into their own methods, which as demonstrated by the stability analysis conducted with this work, has shown great potential.

Even though the e-SBM has been tested with the Stefan model only, its application is quite versatile, and it can be applied to a variety of models with moving boundaries and/or for mixed formulations (thermo-mechanics, Navier Stokes, Free Surface Flow, acoustic, Shallow water, fracture mechanics,...). Concerning the subject of the order of accuracy, the e-SBM has demonstrated to be of second order for all of its variables. In order to increase the order of accuracy it would involve choosing a different polynomial approximation and parallelizing the code. In terms of order in the Taylor development, to avoid reducing the order of accuracy resulting from the imposition of boundary conditions, recent developments in the SBM have introduced new methods for calculating high-order derivatives. The method involves extrapolating the boundary conditions with a specific level of accuracy to match the overall accuracy of the method and compute the derivatives in the different Taylor developments [67].

The Shifted Boundary Method has already demonstrated its capability for 3D problems in various domains of applications [65, 90, 91, 130]. In this work, the focus was put on 2D simulations due to time constraints and the current absence of 3D resources in the code where the e-SBM has been implemented. Nevertheless, the work presented in this manuscript aims to generalize the implementation of the different procedures for 3D models as well. The main challenge when dealing with a moving interface in 3D is to determine the sign of the distance function associated with the definition of the surrogate interface. However, this issue is purely computational and does not affect the procedures proposed in the different chapters of this thesis. Another subject of expansion for the method is the parallelization of the code used for the results presented in this work. It would optimize the computation time and make the method even more suitable for larger systems of resolution due to the use of more refined meshes.

This work has been added to the DAFFEM code, a FORTRAN 90 code originally created by Léo Nouveau a former member of the CARDAMOM team at INRIA Bordeaux. The work presented with this manuscript has been **integrated to this code** initially designed for the resolution of the Darcy equations in their mixed form where the Shifted Method was applied in the context of a steady problem without a moving boundary. The objective with this code is to continue its evolution alongside other applications, with the intention of facilitating its adoption by other researchers to develop the proposed method and expand its domain of applications.

One aspect that has not been studied in this work is the creation of a phase starting from a model with only one phase, i.e., in the context of a melting simulation where initially, only a solid phase is present. The problem arising from this situation is the absence of nodes in the liquid phase, which does not exist at initialization, but are necessary for the reconstruction procedures used to calculate the velocity of the interface and update its position. Due to the computational complexity of this problem, further investigations are necessary to establish a solid baseline appropriate for this method.

Another source of development could involve incorporating external physical phenomena into the resolution of the Stefan model which would allow the method to move away from academic models. For in-flight aircraft and de-icing applications, it is necessary to set up more realistic test cases that take into account the presence of heating resistors. The difficulty with this type of setup is, as previously mentioned, the creation of a melting front, which is not present at the start of the simulation where only a solid phase is present. For more complex configurations, testing the e-SBM for multiple phase-change fronts would be an interesting subject of research, especially within the SBM approach. Fronts could merge or detach from each other, creating numerical challenges that would require time to be studied and a significant implementation effort.

3 Conclusion

In conclusion, the Shifted Boundary Method is a **promising approach** for **handling moving interfaces** and **boundaries**. This work aimed to demonstrate that it is an effective method for addressing the remeshing constraints associated with the movement of boundaries in the considered model. The method has proven to be stable even under significant perturbations, enabling its use in a variety of applications. However, the complexity of the SBM and its application to moving boundaries is associated with questions of accuracy, and one needs to be careful with the imposition of boundary conditions. Nevertheless, all methods dealing with moving boundaries have their own drawbacks, and the SBM has been a significant source of development for a few years now. This aspect has been studied by others [65], and theoretical analyses exist, showing the existence of convergence theorems concerning the method [67]. Therefore, one should continue to explore this method, which will likely expand its applications in future research in the computational field.

Appendix A

Shifted Boundary Method and Neumann conditions

In this Appendix, an example of imposition of a Neumann condition on a surrogate boundary is proposed. The Poisson problem with Neumann boundary conditions is considered as a model for the example. The method detailed in Chapter 1 is applied for this example (see Section 1.3.3).

Neumann conditions are naturally applied in the variational formulation, but the ones applied are the conditions from the physical boundary, which do not correspond to the mesh discretization in the context of an embedded resolution. The work presented in this Appendix consists in the modification of the value of the condition to adapt it to an imposition on a surrogate boundary using the Shifted Boundary Method.

The following Poisson problem of unknown u is considered as an example

Find $u \in H^1(\Omega)$ such that

$$\begin{aligned} -\Delta u(\boldsymbol{x}) &= f(\boldsymbol{x}) && \text{on} \quad \Omega, \\ \boldsymbol{\nabla} \boldsymbol{u}(\boldsymbol{x}) \cdot \boldsymbol{n} &= g_2(\boldsymbol{x}) && \text{on} \quad \Gamma := \partial\Omega, \end{aligned}$$

where Ω is the domain of resolution and $\Gamma := \partial\Omega$ the embedded boundary. An arbitrary initial condition is being considered. By denoting v as a test function one can write the weak formulation as follows

Find $u \in V(\Omega) = \{v \mid v \in H^1(\Omega)\}$ such that $\forall\ v \in V(\Omega)$ one has

$$(\boldsymbol{\nabla} \boldsymbol{u}, \boldsymbol{\nabla} \boldsymbol{v})_\Omega - < \boldsymbol{\nabla} \boldsymbol{u} \cdot \boldsymbol{n}, v >_\Gamma = (f, v)_\Omega \ ,$$

which leads on the surrogate domain $\tilde{\Omega}$ to

Find $u \in V(\tilde{\Omega}) = \{v \mid v \in H^1(\tilde{\Omega})\}$ such that $\forall\ v \in V(\tilde{\Omega})$ one has

$$< \boldsymbol{\nabla} \boldsymbol{u}, \boldsymbol{\nabla} \boldsymbol{v} >_{\tilde{\Omega}} - < \boldsymbol{\nabla} \boldsymbol{u} \cdot \tilde{\boldsymbol{n}}, v >_{\tilde{\Gamma}} = < f, v >_{\tilde{\Omega}} \ .$$

The modifications on the boundary condition are the same as the ones presented for the Dirichlet conditions in Section 1.3.3, but in this case a Taylor expansion of $\boldsymbol{\nabla u}$ is performed.
The term $\boldsymbol{\nabla u}$ can be decomposed using the normal n and the tangential vectors τ on $\partial\Omega$ (see Chapter 1 and Definition (1.22)) such that

$$\begin{aligned} \boldsymbol{\nabla u} &= (\boldsymbol{\nabla u}\cdot\boldsymbol{n})\boldsymbol{n} + (\boldsymbol{\nabla u}\cdot\boldsymbol{\tau})\boldsymbol{\tau}, \\ \Longrightarrow \quad \boldsymbol{\nabla u}\cdot\tilde{\boldsymbol{n}} &= (\boldsymbol{\nabla u}\cdot\boldsymbol{n})\boldsymbol{n}\cdot\tilde{\boldsymbol{n}} + (\boldsymbol{\nabla u}\cdot\boldsymbol{\tau})\boldsymbol{\tau}\cdot\tilde{\boldsymbol{n}}. \end{aligned} \tag{A.1}$$

Using Equation (A.1) the Taylor expansion of $\boldsymbol{\nabla u}$ at a point $\tilde{\boldsymbol{x}} \in \tilde{\Gamma}$ gives

$$\boldsymbol{\nabla u}(\boldsymbol{x}) = \boldsymbol{\nabla u}(\tilde{\boldsymbol{x}}) + \nabla\cdot(\boldsymbol{\nabla u}(\tilde{\boldsymbol{x}}))\cdot(\boldsymbol{x}-\tilde{\boldsymbol{x}}) + O(||\boldsymbol{x}-\tilde{\boldsymbol{x}}||^2),$$

which leads to the following approximation

$$\begin{aligned} \boldsymbol{\nabla u}(\boldsymbol{x})\cdot\boldsymbol{n} &\approx \boldsymbol{\nabla u}(\tilde{\boldsymbol{x}})\cdot\boldsymbol{n} + (\nabla\cdot(\boldsymbol{\nabla u}(\tilde{\boldsymbol{x}}))\cdot\boldsymbol{d})\cdot\boldsymbol{n} \\ \Longrightarrow \quad g_2 \quad &\approx \boldsymbol{\nabla u}(\tilde{\boldsymbol{x}})\cdot\boldsymbol{n} + (\nabla\cdot(\boldsymbol{\nabla u}(\tilde{\boldsymbol{x}}))\cdot\boldsymbol{d})\cdot\boldsymbol{n} \\ \Longrightarrow \boldsymbol{\nabla u}(\tilde{\boldsymbol{x}})\cdot\boldsymbol{n} &\approx g_2 - (\nabla\cdot(\boldsymbol{\nabla u}(\tilde{\boldsymbol{x}}))\cdot\boldsymbol{d})\cdot\boldsymbol{n}. \end{aligned} \tag{A.2}$$

Finally, using Equation (A.2) and Definition (A.1), the value of the Neumann condition on $\tilde{\Gamma}$ can be interpreted as

$$\boldsymbol{\nabla u}\cdot\tilde{\boldsymbol{n}} = (g_2 - (\nabla\cdot(\boldsymbol{\nabla u}(\tilde{\boldsymbol{x}}))\cdot\boldsymbol{d})\cdot\boldsymbol{n})\boldsymbol{n}\cdot\tilde{\boldsymbol{n}} + (\boldsymbol{\nabla u}\cdot\boldsymbol{\tau})\boldsymbol{\tau}\cdot\tilde{\boldsymbol{n}}$$

and the weak formulation verified on the surrogate domain is

Find $u \in V(\tilde{\Omega}) = \{v \mid v \in H^1(\tilde{\Omega})\}$ such that $\forall\ v \in V(\tilde{\Omega})$ one has

$$\begin{aligned} <\boldsymbol{\nabla u},\boldsymbol{\nabla v}>_{\tilde{\Omega}} + <((\nabla\cdot(\boldsymbol{\nabla u}(\tilde{\boldsymbol{x}}))\cdot\boldsymbol{d})\cdot\boldsymbol{n})\boldsymbol{n}\cdot\tilde{\boldsymbol{n}}, v>_{\tilde{\Gamma}} - <(\boldsymbol{\nabla u}\cdot\boldsymbol{\tau})\boldsymbol{\tau}\cdot\tilde{\boldsymbol{n}}, v>_{\tilde{\Gamma}} \\ =<f,v>_{\tilde{\Omega}} + <g_2(\boldsymbol{n}\cdot\tilde{\boldsymbol{n}}), v>_{\tilde{\Gamma}}. \end{aligned} \tag{A.3}$$

One of the difficulties of Formulation (A.3) and more generally of embedded Neumann conditions is to be able to compute the value of $\nabla\cdot(\boldsymbol{\nabla u})$. A possible approach is to consider a gradient recovering technique, for example a Green Gauss method.

Bibliography

[1] National Weather Service. *Icing*. https://www.weather.gov/source/zhu/ZHU_Training_Page/icing_stuff/icing/icing.htm. Official website of the US government.

[2] A. Main and G. Scovazzi. "The Shifted Boundary Method for Embedded Domain Computations. Part I : Poisson and Stokes Problems". In: *Journal of Computational Physics* 372 (Oct. 2017). DOI: 10.1016/j.jcp.2017.10.026. URL: https://www.sciencedirect.com/science/article/pii/S0021999117307799.

[3] A. Main and G. Scovazzi. "The shifted boundary method for embedded domain computations. Part II: Linear advection–diffusion and incompressible Navier–Stokes equations". In: *Journal of Computational Physics* 372 (2018), pp. 996–1026. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2018.01.023. URL: https://www.sciencedirect.com/science/article/pii/S0021999118300330.

[4] V. Alexiades and A. D. Solomon. *Mathematical Modeling of Melting and Freezing Processes*. May 2018. ISBN: 9780203749449. DOI: 10.1201/9780203749449.

[5] A. Masud and T. Hughes. "A stabilized mixed finite element method for Darcy flow". In: *Computer Methods in Applied Mechanics and Engineering* 191 (Aug. 2002), pp. 4341–4370. URL: https://www.sciencedirect.com/science/article/pii/S0045782502003717.

[6] T. Hughes, A. Masud, and J. Wan. "A stabilized mixed discontinuous Galerkin method for Darcy flow". In: *Computer Methods in Applied Mechanics and Engineering* 195.25 (2006). Discontinuous Galerkin Methods, pp. 3347–3381. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2005.06.018. URL: https://www.sciencedirect.com/science/article/pii/S0045782505002732.

[7] F. Brezzi et al. "Mixed Discontinuous Galerkin Methods for Darcy Flow". In: *Journal of Scientific Computing* 22-23 (June 2005), pp. 119–145. DOI: 10.1007/s10915-004-4150-8. URL: https://link.springer.com/article/10.1007/s10915-004-4150-8.

[8] S.H.S. Joodat, K.B. Nakshatrala, and R. Ballarini. "Modeling flow in porous media with double porosity/permeability: A stabilized mixed formulation, error analysis, and numerical solutions". In: *Computer Methods in Applied Mechanics and Engineering* 337 (2018), pp. 632–676. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2018.04.004. URL: https://www.sciencedirect.com/science/article/pii/S0045782518301749.

[9] M.S. Joshaghani, S.H.S. Joodat, and K.B. Nakshatrala. "A stabilized mixed discontinuous Galerkin formulation for double porosity/permeability model". In: *Computer Methods in Applied Mechanics and Engineering* 352 (2019), pp. 508–560. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2019.04.010. URL: https://www.sciencedirect.com/science/article/pii/S0045782519302075.

[10] D. Hitt. *What Are Wind Tunnels?* https://www.nasa.gov/audience/forstudents/k-4/stories/nasa-knows/what-are-wind-tunnels-k4.html. National Aeronautics and Space Administration (NASA) website. 2018.

[11] J. Nitsche. "Lineare Spline-Funktionen und die Methoden von Ritz für elliptische Randwertprobleme". In: *Arch. Rational Mech. Anal.* 36 (1970). published vy: American Mathematical Society, pp. 348–355.

[12] C. Vuik. "Some historical notes about the Stefan problem". In: 1993. URL: https://www.researchgate.net/publication/2764813_Some_historical_notes_about_the_Stefan_problem.

[13] T. Carlier et al. "An enriched shifted boundary method to account for moving fronts". In: *Journal of Computational Physics* 489 (2023), p. 112295. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2023.112295. URL: https://www.sciencedirect.com/science/article/pii/S002199912300390X.

[14] G. Mingione et al. *Flight in icing conditions.* https://www.ecologie.gouv.fr/sites/default/files/Icing_flight_manual.pdf. On behalf of French DGAC.

[15] Federal Aviation Administration. *Avions de Transport Regional 72-212.* https://www.faa.gov/lessons_learned/transport_airplane/accidents/N401AM. Official website of the US government. 2022.

[16] International Civil Aviation Organization. *Manual of Aircraft Ground De-icing/Anti-icing Operations.* https://skybrary.aero/sites/default/files/bookshelf/4400.pdf. 2018.

[17] Canadian Transport. *Deicing/Anti-icing Fluids.* https://tc.canada.ca/en/aviation/publications/when-doubtsmall-large-aircraft-aircraft-critical-surface-contamination-training-tp-10643/chapter-3-deicing-anti-icing-fluids-15. Official website of the Canadian government. 2010.

[18] G. Georges, J. Breil, and P. H. Maire. "A 3D GCL compatible cell-centered Lagrangian scheme for solving gas dynamics equations". In: *Journal of Computational Physics* 305 (2016), pp. 921–941. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2015.10.040. URL: https://www.sciencedirect.com/science/article/pii/S0021999115007172.

[19] E.J. Caramana et al. "The Construction of Compatible Hydrodynamics Algorithms Utilizing Conservation of Total Energy". In: *Journal of Computational Physics* 146.1 (1998), pp. 227–262. ISSN: 0021-9991. DOI: https://doi.org/10.1006/jcph.1998.6029. URL: https://www.sciencedirect.com/science/article/pii/S0021999198960296.

[20] R. Loubère, P. H. Maire, and P. Váchal. "3D staggered Lagrangian hydrodynamics scheme with cell-centered Riemann solver-based artificial viscosity". In: *International Journal for Numerical Methods in Fluids* 72.1 (2013), pp. 22–42. DOI: https://doi.org/10.1002/fld.3730. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.3730.

[21] P.M. Maire et al. "A Cell-Centered Lagrangian Scheme for Two-Dimensional Compressible Flow Problems". In: *SIAM Journal on Scientific Computing* 29.4 (2007), pp. 1781–1824. DOI: 10.1137/050633019. URL: https://doi.org/10.1137/050633019.

[22] C.W. Hirt, A. A. Amsden, and J.L Cook. "An arbitrary Lagrangian-Eulerian computing method for all flow speeds". In: *Journal of Computational Physics* 14.3 (1974), pp. 227–253. ISSN: 0021-9991. DOI: https://doi.org/10.1016/0021-9991(74)90051-5. URL: https://www.sciencedirect.com/science/article/pii/0021999174900515.

[23] A.A. Johnson and T.E. Tezduyar. "Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces". In: *Computer Methods in Applied Mechanics and Engineering* 119.1 (1994), pp. 73–94. ISSN: 0045-7825. DOI: https://doi.org/10.1016/0045-7825(94)00077-8. URL: https://www.sciencedirect.com/science/article/pii/0045782594000778.

[24] A.A. Johnson and T.E. Tezduyar. "Simulation of multiple spheres falling in a liquid-filled tube". In: *Computer Methods in Applied Mechanics and Engineering* 134.3 (1996), pp. 351–373. ISSN: 0045-7825. DOI: https://doi.org/10.1016/0045-7825(95)00988-4. URL: https://www.sciencedirect.com/science/article/pii/0045782595009884.

[25] C. Farhat, P. Geuzaine, and C. Grandmont. "The Discrete Geometric Conservation Law and the Nonlinear Stability of ALE Schemes for the Solution of Flow Problems on Moving Grids". In: *Journal of Computational Physics* 174.2 (2001), pp. 669–694. ISSN: 0021-9991. DOI: https://doi.org/10.1006/jcph.2001.6932. URL: https://www.sciencedirect.com/science/article/pii/S0021999101969323.

[26] A. Masud, M. Bhanabhagvanwala, and R. A. Khurram. "An adaptive mesh rezoning scheme for moving boundary flows and fluid–structure interaction". In: *Computers & Fluids* 36.1 (2007). Challenges and Advances in Flow Simulation and Modeling, pp. 77–91. ISSN: 0045-7930. DOI: https://doi.org/10.1016/j.compfluid.2005.07.013. URL: https://www.sciencedirect.com/science/article/pii/S0045793005001301.

[27] A. Jendoubi, J. Deteix, and A. Fortin. "A simple mesh-update procedure for fluid–structure interaction problems". In: *Computers & Structures* 169 (2016), pp. 13–23. ISSN: 0045-7949. DOI: https://doi.org/10.1016/j.compstruc.2016.02.015. URL: https://www.sciencedirect.com/science/article/pii/S0045794916300396.

[28] N. Barral, G. Olivier, and F. Alauzet. "Time-accurate anisotropic mesh adaptation for three-dimensional time-dependent problems with body-fitted moving geometries". In: *Journal of Computational Physics* 331 (2017), pp. 157–187. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2016.11.029. URL: https://www.sciencedirect.com/science/article/pii/S0021999116306155.

[29] F. Alauzet. "A changing-topology moving mesh technique for large displacements". In: *Engineering with Computers* 30.2 (2014), pp. 175–200. DOI: 10.1007/s00366-013-0340-z. URL: https://inria.hal.science/hal-01114995.

[30] R. Costa et al. "High-order accurate conjugate heat transfer solutions with a finite volume method in anisotropic meshes with application in polymer processing". In: *International Journal for Numerical Methods in Engineering* (2021). DOI: https://doi.org/10.1002/nme.6892. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.6892.

[31] W. M. Chan. "Overset grid technology development at NASA Ames Research Center". In: *Computers Fluids* 38.3 (2009), pp. 496–503. ISSN: 0045-7930. DOI: https://doi.org/10.1016/j.compfluid.2008.06.009. URL: https://www.sciencedirect.com/science/article/pii/S0045793008001953.

[32] M. Bergmann, M. G. Carlino, and A. Iollo. "Second Order ADER Scheme for Unsteady Advection-Diffusion on Moving Overset Grids with a Compact Transmission Condition". In: *SIAM Journal on Scientific Computing* 44.1 (2022), A524–A553. DOI: 10.1137/21M1393911. eprint: https://doi.org/10.1137/21M1393911. URL: https://doi.org/10.1137/21M1393911.

[33] T. Liu et al. "A three-dimensional aircraft ice accretion model based on the numerical solution of the unsteady Stefan problem". In: *Aerospace Science and Technology* 93 (2019), p. 105328. ISSN: 1270-9638. DOI: https://doi.org/10.1016/j.ast.2019.105328. URL: https://www.sciencedirect.com/science/article/pii/S1270963819310016.

[34] C. Peskin. "Flow patterns around heart valves: A numerical method". In: *Journal of Computational Physics* 10.2 (1972), pp. 252–271. ISSN: 0021-9991. DOI: https://doi.org/10.1016/0021-9991(72)90065-4. URL: https://www.sciencedirect.com/science/article/pii/0021999172900654.

[35] C. Peskin. "The immersed boundary method". In: *Acta Numerica* 11 (2002), pp. 479–517. DOI: 10.1017/S0962492902000077.

[36] C. Peskin. "Numerical analysis of blood flow in the heart". English (US). In: *Journal of Computational Physics* 25.3 (Nov. 1977), pp. 220–252. ISSN: 0021-9991. DOI: 10.1016/0021-9991(77)90100-0.

[37] R. Mittal and G. Iaccarino. "Imersed Boundary Methods". In: *Annual Review of Fluid Mechanics* 37.1 (2005), pp. 239–261. DOI: 10.1146/annurev.fluid.37.061903.175743. URL: https://doi.org/10.1146/annurev.fluid.37.061903.175743.

[38] F. Sotiropoulos and X. Yang. "Immersed boundary methods for simulating fluid–structure interaction". In: *Progress in Aerospace Sciences* 65 (2014), pp. 1–21. ISSN: 0376-0421. DOI: https://doi.org/10.1016/j.paerosci.2013.09.003. URL: https://www.sciencedirect.com/science/article/pii/S0376042113000870.

[39] B. E. Griffith and N. A. Patankar. "Immersed Methods for Fluid Structure Interaction". In: *Annual Review of Fluid Mechanics* 52.1 (2020), pp. 421–448. DOI: 10.1146/annurev-fluid-010719-060228. URL: https://doi.org/10.1146/annurev-fluid-010719-060228.

[40] R. Mittal et al. "A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries". In: *Journal of Computational Physics* 227.10 (2008), pp. 4825–4852. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2008.01.028. URL: https://www.sciencedirect.com/science/article/pii/S0021999108000235.

[41] R. P. Fedkiw et al. "A Non-oscillatory Eulerian Approach to Interfaces in Multimaterial Flows (the Ghost Fluid Method)". In: *Journal of Computational Physics* 152.2 (1999), pp. 457–492. ISSN: 0021-9991. DOI: https://doi.org/10.1006/jcph.1999.6236. URL: https://www.sciencedirect.com/science/article/pii/S0021999199962368.

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

[43] Y. Gorsse et al. "A simple second order cartesian scheme for compressible Euler flows". In: *Journal of Computational Physics* 231.23 (2012), pp. 7780–7794. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2012.07.014. URL: https://www.sciencedirect.com/science/article/pii/S0021999112003877.

[44] B. Constant et al. "An improved immersed boundary method for turbulent flow simulations on Cartesian grids". In: *Journal of Computational Physics* 435 (2021), p. 110240. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2021.110240. URL: https://www.sciencedirect.com/science/article/pii/S0021999121001352.

[45] A. Main and 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". In: *Journal of Computational Physics* 258 (2014), pp. 613–633. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2013.11.001. URL: https://www.sciencedirect.com/science/article/pii/S002199911300747X.

[46] D. Z. Huang, D. De Santis, and C. Farhat. "A family of position- and orientation-independent embedded boundary methods for viscous flow and fluid–structure interaction problems". In: *Journal of Computational Physics* 365 (2018), pp. 74–104. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2018.03.028. URL: https://www.sciencedirect.com/science/article/pii/S0021999118301888.

[47] M. Fournie and A.Lozinski. "Stability and Optimal Convergence of Unfitted Extended Finite Element Methods with Lagrange Multipliers for the Stokes Equations". In: (2017). Ed. by Stéphane P. A. Bordas et al., pp. 143–182.

[48] A. Hansbo and P. Hansbo. "An unfitted finite element method, based on Nitsche's method, for elliptic interface problems". In: *Computer Methods in Applied Mechanics and Engineering* 191.47 (2002), pp. 5537–5552. ISSN: 0045-7825. DOI: https://doi.org/10.1016/S0045-7825(02)00524-8. URL: https://www.sciencedirect.com/science/article/pii/S0045782502005248.

[49] H. M. Mourad, J. Dolbow, and I. Harari. "A bubble-stabilized finite element method for Dirichlet constraints on embedded interfaces". In: *International Journal for Numerical Methods in Engineering* 69.4 (2007), pp. 772–793. DOI: https://doi.org/10.1002/nme.1788. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.1788.

[50] J. Dolbow and I. Harari. "An efficient finite element method for embedded interface problems". In: *International Journal for Numerical Methods in Engineering* 78.2 (2009), pp. 229–252. DOI: https://doi.org/10.1002/nme.2486. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.2486.

[51] J. Parvizian, A. Düster, and E. Rank. "Finite cell method". In: *Computational Mechanics* 41.1 (2007), pp. 121–133. DOI: 10.1007/s00466-007-0173-y. URL: https://link.springer.com/article/10.1007/s00466-007-0173-y.

[52] M.P. Kirkpatrick, S.W. Armfield, and J.H. Kent. "A representation of curved boundaries for the solution of the Navier–Stokes equations on a staggered three-dimensional Cartesian grid". In: *Journal of Computational Physics* 184.1 (2003), pp. 1–36. ISSN: 0021-9991. DOI: https://doi.org/10.1016/S0021-9991(02)00013-X. URL: https://www.sciencedirect.com/science/article/pii/S002199910200013X.

[53] E. Burman. "Ghost penalty". In: *Comptes Rendus Mathematique* 348.21 (2010), pp. 1217–1220. ISSN: 1631-073X. DOI: https://doi.org/10.1016/j.crma.2010.10.006. URL: https://www.sciencedirect.com/science/article/pii/S1631073X10002827.

[54] E.Burman and P. Hansbo. "Fictitious domain methods using cut elements: III. A stabilized Nitsche method for Stokes´ problem". In: *ESAIM: M2AN* 48.3 (2014), pp. 859–874. DOI: 10.1051/m2an/2013123. URL: https://doi.org/10.1051/m2an/2013123.

[55] B. Schott et al. "A face-oriented stabilized Nitsche-type extended variational multiscale method for incompressible two-phase flow". In: *International Journal for Numerical Methods in Engineering* 104.7 (2015), pp. 721–748. DOI: https://doi.org/10.1002/nme.4789. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.4789.

[56] R. J. Leveque and Z. Li. "The Immersed Interface Method for Elliptic Equations with Discontinuous Coefficients and Singular Sources". In: *SIAM Journal on Numerical Analysis* 31.4 (1994), pp. 1019–1044. DOI: 10.1137/0731054. URL: https://www.jstor.org/stable/2158113.

[57] M. Duprez and A. Lozinski. "ϕ-FEM: A Finite Element Method on Domains Defined by Level-Sets". In: *SIAM Journal on Numerical Analysis* 58.2 (2020), pp. 1008–1028. DOI: 10.1137/19M1248947. URL: https://doi.org/10.1137/19M1248947.

[58] M. Duprez, V. Lleras, and A. Lozinski. "A new ϕ-FEM approach for problems with natural boundary conditions". In: *Numerical Methods for Partial Differential Equations* 39.1 (2022), pp. 281–303. DOI: 10.1002/num.22878. URL: https://hal.science/hal-02521042.

[59] S. Badia, F. Verdugo, and A. F. Martín. "The aggregated unfitted finite element method for elliptic problems". In: *Computer Methods in Applied Mechanics and Engineering* 336 (2018), pp. 533–553. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2018.03.022. URL: https://www.sciencedirect.com/science/article/pii/S0045782518301476.

[60] E. Neiva and S. Badia. "Robust and scalable h-adaptive aggregated unfitted finite elements for interface elliptic problems". In: *Computer Methods in Applied Mechanics and Engineering* 380 (2021), p. 113769. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2021.113769. URL: https://www.sciencedirect.com/science/article/pii/S0045782521001055.

[61] J. Fernández-Fidalgo et al. "Very high-order method on immersed curved domains for finite difference schemes with regular Cartesian grids". In: *Computer Methods in Applied Mechanics and Engineering* 360 (Mar. 2020). DOI: 10.1016/j.cma.2019.112782.

[62] R. Costa et al. "High-order accurate finite volume scheme on curved boundaries for the two-dimensional steady-state convection-diffusion equation with Dirichlet condition". In: *Applied Mathematical Modelling* 54 (Feb. 2018), pp. 752–767. DOI: 10.1016/j.apm.2017.10.016.

[63] R. Costa et al. "Very high-order accurate finite volume scheme for the convection-diffusion equation with general boundary conditions on arbitrary curved boundaries". In: *International Journal for Numerical Methods in Engineering* (Sept. 2018), pp. 1–33. DOI: 10.1002/nme.5953.

[64] T. Song et al. "The shifted boundary method for hyperbolic systems: Embedded domain computations of linear waves and shallow water flows". In: *Journal of Computational Physics* 369 (2018), pp. 45–79. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2018.04.052. URL: https://www.sciencedirect.com/science/article/pii/S0021999118302857.

[65] M. Ciallella et al. "Shifted boundary polynomial corrections for compressible flows: high order on curved domains using linear meshes". In: *Applied Mathematics and Computation* 441 (2023), p. 127698. ISSN: 0096-3003. DOI: https://doi.org/10.1016/j.amc.2022.127698. URL: https://www.sciencedirect.com/science/article/pii/S0096300322007664.

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

[67] N. M. Atallah, C. Canuto, and G. Scovazzi. "The high-order Shifted Boundary Method and its analysis". In: *Computer Methods in Applied Mechanics and Engineering* 394 (2022), p. 114885. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2022.114885. URL: https://www.sciencedirect.com/science/article/pii/S0045782522001797.

[68] S. Clain, D. Lopes, and R.M.S. Pereira. "Very high-order Cartesian-grid finite difference method on arbitrary geometries". In: *Journal of Computational Physics* 434 (2021), p. 110217. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2021.110217. URL: https://www.sciencedirect.com/science/article/pii/S0021999121001121.

[69] A. Al-Kebsi, R. Mose, and Y. Hoarau. "Multi-Step Ice Accretion Simulation Using the Level-Set Method". In: *SAE International Conference on Icing of Aircraft, Engines, and Structure*. Minneapolis, United States, June 2019. DOI: 10.4271/2019-01-1955. URL: https://hal.science/hal-02562783.

[70] F. Morency and H. Beaugendre. "Mathematical model for ice wall interactions within a level set method". In: *International Journal of Engineering Systems Modelling and Simulation* 8 (Jan. 2016), pp. 125–135. DOI: 10.1504/IJESMS.2016.075552.

[71] H. Beaugendre et al. "Computation of Ice Shedding Trajectories Using Cartesian Grids, Penalization, and Level Sets". In: *Modelling and Simulation in Engineering* 15 (Jan. 2011). DOI: 10.1155/2011/274947.

[72] K. Suzuki et al. "Numerical simulations of solid–liquid and solid–solid interactions in ice slurry flows by the thermal immersed boundary–lattice Boltzmann method". In: *International Journal of Heat and Mass Transfer* 157 (2020), p. 119944. ISSN: 0017-9310. DOI: https://doi.org/10.1016/j.ijheatmasstransfer.2020.119944. URL: https://www.sciencedirect.com/science/article/pii/S0017931019346903.

[73] Z. G. Feng and E. E. Michaelides. "The immersed boundary-lattice Boltzmann method for solving fluid–particles interaction problems". In: *Journal of Computational Physics* 195.2 (2004), pp. 602–628. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2003.10.013. URL: https://www.sciencedirect.com/science/article/pii/S0021999103005758.

[74] Z. G. Feng and E. E. Michaelides. "Proteus: a direct forcing method in the simulations of particulate flows". In: *Journal of Computational Physics* 202.1 (2005), pp. 20–51. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2004.06.020. URL: https://www.sciencedirect.com/science/article/pii/S0021999104002669.

[75] H. Zhang et al. "A combined TLBM–IBM–DEM scheme for simulating isothermal particulate flow in fluid". In: *International Journal of Heat and Mass Transfer* 91 (2015), pp. 178–189. ISSN: 0017-9310. DOI: https://doi.org/10.1016/j.ijheatmasstransfer.2015.07.119. URL: https://www.sciencedirect.com/science/article/pii/S0017931015008376.

[76] A. Eshghinejadfard and D. Thévenin. "Numerical simulation of heat transfer in particulate flows using a thermal immersed boundary lattice Boltzmann method". In: *International Journal of Heat and Fluid Flow* 60 (2016), pp. 31–46. ISSN: 0142-727X. DOI: https://doi.org/10.1016/j.ijheatfluidflow.2016.04.002. URL: https://www.sciencedirect.com/science/article/pii/S0142727X16301126.

[77] Jinzi Mac Huang, Michael J. Shelley, and David B. Stein. "A stable and accurate scheme for solving the Stefan problem coupled with natural convection using the Immersed Boundary Smooth Extension method". In: *Journal of Computational Physics* 432 (2021), p. 110162. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2021.110162. URL: https://www.sciencedirect.com/science/article/pii/S0021999121000541.

[78] R. Abgrall, H. Beaugendre, and C. Dobrzynski. "An immersed boundary method using unstructured anisotropic mesh adaptation combined with level-sets and penalization techniques". In: *Journal of Computational Physics* 257 (2014), pp. 83–101. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2013.08.052. URL: https://www.sciencedirect.com/science/article/pii/S0021999113005962.

[79] E. M. Kolahdouz et al. "An immersed interface method for discrete surfaces". In: *Journal of Computational Physics* 400 (2020), p. 108854. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2019.07.052. URL: https://www.sciencedirect.com/science/article/pii/S0021999119305388.

[80] K. Li N. Atallah G.A. Main and G. Scovazzi. "The Shifted Interface Method: A Flexible Approach to Embedded Interface Computations". In: *International Journal for Numerical Methods in Engineering* 121 (Sept. 2019). DOI: 10.1002/nme.6231. URL: https://onlinelibrary.wiley.com/doi/full/10.1002/nme.6231.

[81] J. H. Bramble, T. Dupont, and V. Thomée. "Methods for Dirichlet's Problem in Approximating Polygonal Domains with Boundary-Value Corrections". In: *Mathematics of Computation* (October 1972). published vy: American Mathematical Society.

[82] E. Burman, P. Hansbo, and M. G. Larson. "A Cut Finite Element Method with Boundary Value Correction". In: *Conference paper, ENUMATH 2017* (July 2015). Part of the Lecture Notes in Computational Science and Engineering book series (LNCSE,volume 126).

[83] M. Duprez, V. Lleras, and A. Lozinski. "ϕ-FEM: an optimally convergent and easily implementable immersed boundary method for particulate flows and Stokes equations". In: *ESAIM : Mathematical Modelling and Numerical Analysis* 57.1 (2023), pp. 1111–1142. DOI: https://doi.org/10.1051/m2an/2023010. URL: https://www.esaim-m2an.org/articles/m2an/abs/2023/03/m2an220037/m2an220037.html.

[84] E. Burman and P. Hansbo. "Fictitious domain finite element methods using cut elements: I. A stabilized Lagrange multiplier method". In: *Computer Methods in Applied Mechanics and Engineering* 199.41 (2010), pp. 2680–2686. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2010.05.011. URL: https://www.sciencedirect.com/science/article/pii/S004578251000160X.

[85] E. Burman and P. Hansbo. "Fictitious domain finite element methods using cut elements: II. A stabilized Nitsche method". In: *Applied Numerical Mathematics* 62.4 (2012). Third Chilean Workshop on Numerical Analysis of Partial Differential Equations (WONAPDE 2010), pp. 328–341. ISSN: 0168-9274. DOI: https://doi.org/10.1016/j.apnum.2011.01.008. URL: https://www.sciencedirect.com/science/article/pii/S0168927411000298.

[86] S. Bertoluzza, M. Ismail, and B. Maury. "The Fat Boundary Method: Semi-Discrete Scheme and Some Numerical Experiments". In: *Domain Decomposition Methods in science and engineering*. Lecture Notes in Computational Science and Engineering. springer, 2005, p. 513. URL: https://hal.science/hal-00666064.

[87] S. Bertoluzza, M. Ismail, and B. Maury. "Analysis of the fully discrete fat boundary method". In: *Numerische Mathematik* 118.1 (July 2010), pp. 49–77. DOI: 10.1007/s00211-010-0317-4. URL: https://hal.science/hal-00665644.

[88] B. Cockburn, W. Qiu, and M. E. Solano. "A priori error analysis for HDG methods using extensions from subdomains to achieve boundary conformity". In: *Mathematics of computation* 83.286 (March 2014,). Article electronically published on July 18, 2013, pp. 665–699. DOI: 10.1090/S0025-5718-2013-02747-0. URL: https://www.ams.org/journals/mcom/2014-83-286/S0025-5718-2013-02747-0/S0025-5718-2013-02747-0.pdf.

[89] B. Cockburn and M. Solano. "Solving Convection-Diffusion Problems on Curved Domains by Extensions from Subdomains". In: *Journal of Scientific Computing* 1 (2014). DOI: https://doi.org/10.1007/s10915-013-9776-y. URL: https://www.researchgate.net/publication/261566797_Solving_Convection-Diffusion_Problems_on_Curved_Domains_by_Extensions_from_Subdomains.

[90] O. Colomés et al. "A weighted Shifted Boundary Method for free surface flow problems". In: *Journal of Computational Physics* 424 (2021), p. 109837. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2020.109837. URL: https://www.sciencedirect.com/science/article/pii/S0021999120306112.

[91] N. M. Atallah, C. Canuto, and G. Scovazzi. "The shifted boundary method for solid mechanics". In: *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.

[92] N. M. Atallah, C. Canuto, and G. Scovazzi. "The second-generation Shifted Boundary Method and its numerical analysis". In: *Computer Methods in Applied Mechanics and Engineering* 372 (2020), p. 113341. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2020.113341. URL: https://www.sciencedirect.com/science/article/pii/S0045782520305260.

[93] H. Nishikawa. "Robust and accurate viscous discretization via upwind scheme – I: Basic principle". In: *Computers & Fluids* 49.1 (2011), pp. 62–86. ISSN: 0045-7930. DOI: https://doi.org/10.1016/j.compfluid.2011.04.014. URL: https://www.sciencedirect.com/science/article/pii/S0045793011001484.

[94] A. Mazaheri and H. Nishikawa. "Improved second-order hyperbolic residual-distribution scheme and its extension to third-order on arbitrary triangular grids". In: *Journal of Computational Physics* 300 (2015), pp. 455–491. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2015.07.054. URL: https://www.sciencedirect.com/science/article/pii/S0021999115005070.

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

[96] B. T. Helenbrook and N. S. Barlow. "Spatial–temporal stability analysis of faceted growth with application to horizontal ribbon growth". In: *Journal of Crystal Growth* 454 (2016), pp. 35–44. ISSN: 0022-0248. DOI: https://doi.org/10.1016/j.jcrysgro.2016.08.052. URL: https://www.sciencedirect.com/science/article/pii/S0022024816304663.

[97] P. Daggolu et al. "Thermal-capillary analysis of the horizontal ribbon growth of silicon crystals". In: *Journal of Crystal Growth* 355.1 (2012), pp. 129–139. ISSN: 0022-0248. DOI: https://doi.org/10.1016/j.jcrysgro.2012.06.055. URL: https://www.sciencedirect.com/science/article/pii/S0022024812004757.

[98] P. Daggolu et al. "Stability limits for the horizontal ribbon growth of silicon crystals". In: *Journal of Crystal Growth* 363 (2013), pp. 132–140. ISSN: 0022-0248. DOI: https://doi.org/10.1016/j.jcrysgro.2012.10.024. URL: https://www.sciencedirect.com/science/article/pii/S0022024812007294.

[99] A Muñoz Mateo and J Brand. "Stability and dispersion relations of three-dimensional solitary waves in trapped Bose–Einstein condensates". In: *New Journal of Physics* 17.12 (Dec. 2015), p. 125013. DOI: 10.1088/1367-2630/17/12/125013. URL: https://dx.doi.org/10.1088/1367-2630/17/12/125013.

[100] W. Liao. "On the dispersion, stability and accuracy of a compact higher-order finite difference scheme for 3D acoustic wave equation". In: *Journal of Computational and Applied Mathematics* 270 (2014). Fourth International Conference on Finite Element Methods in Engineering and Sciences (FEMTEC 2013), pp. 571–583. ISSN: 0377-0427. DOI: https://doi.org/10.1016/j.cam.2013.08.024. URL: https://www.sciencedirect.com/science/article/pii/S0377042713004299.

[101] *NASA general purpose structural analysis program*. https://ntrs.nasa.gov/citations/19680019432. 1968.

[102] Founder of CADFEM GmbH Dr.-Ing. Guenter Mueller and Managing Partner at CADFEM International GmbH. *The origins of the finite element method*. https://www.cadfem.net/sg/en/cadfem-informs/cadfem-newsroom/cadfem-journal/fem-history.html.

[103] J. H. Argyris and S. Kelsey. "The matrix force method of structural analysis and some new applications". In: *Aeronautical Research Council Reports Memoranda* (1956). DOI: 10.1016/j.jcp.2017.10.026. URL: https://reports.aerade.cranfield.ac.uk/handle/1826.2/3602.

[104] T. Hughes, J. Oden, and M. Papadrakakis. *John H. Argyris*. https://www.nae.edu/189542/JOHN-H-ARGYRIS-19132004. Submitted by Nae Home Secretary, originally published in 2004 in Computer Methods in Applied Mechanics and Engineering journal. 2011.

[105] M. J. Turner et al. "Stiffness and Deflection Analysis of Complex Structures". In: *Journal of the Aeronautical Sciences* 23 (1956), pp. 805–823.

[106] R.W. Clough. "The Finite Element Method in Plane Stress Analysis". In: *Proceedings of American Society of Civil Engineers* (Sept. 1960).

[107] R. Courant. "Variational methods for the solution of problems of equilibrium and vibrations". In: *Bulletin of the American Mathematical Society* (January 1943).

[108] A. Hrennikoff. "Solution of Problems of Elasticity by the Framework Method". In: *Journal of Applied Mechanisc* (Dec. 1941).

[109] COMSOL Multiphysics Official Website. *Understanding and Changing the Element Order*. https://www.comsol.com/support/learning-center/article/Understanding-and-Changing-the-Element-Order-47301.

[110] F. Camacho. "A Posteriori Error Estimates for Surface Finite Element Methods". In: 2014. URL: https://api.semanticscholar.org/CorpusID:52027795.

[111] *Math Insight : The idea behind Green's theorem*. https://mathinsight.org/greens_theorem_idea.

[112] J. Royer. *Green Formula*. https://www.math.univ-toulouse.fr/~jroyer/TD/2022-23-M1/M1-Ch3.pdf. Leture note. 2022-2023.

[113] A. Ern and J. L. Guermond. *Éléments finis: théorie, applications, mise en oeuvre*. Springer Berlin, Heidelberg, 2002. ISBN: 9783540426158. URL: https://api.semanticscholar.org/CorpusID:169303969.

[114] T. Ye et al. "An Accurate Cartesian Grid Method for Viscous Incompressible Flows with Complex Immersed Boundaries". In: *Journal of Computational Physics* 156.2 (1999), pp. 209–240. ISSN: 0021-9991. DOI: https://doi.org/10.1006/jcph.1999.6356. URL: https://www.sciencedirect.com/science/article/pii/S0021999199963568.

[115] F. Morency, H. Beaugendre, and F. Gallizio. "Aerodynamic force evaluation for ice shedding phenomenon using vortex in cell scheme, penalisation and level set approaches". In: *International Journal of Computational Fluid Dynamics* 26.9-10 (2012), pp. 435–450. DOI: 10.1080/10618562.2012.739683. URL: https://doi.org/10.1080/10618562.2012.739683.

[116] M. Bergmann and A. Iollo. "Modeling and simulation of fish-like swimming". In: *Journal of Computational Physics* 230.2 (2011), pp. 329–348. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2010.09.017. URL: https://www.sciencedirect.com/science/article/pii/S0021999110005115.

[117] A. Gilmanov, F. Sotiropoulos, and E. Balaras. "A general reconstruction algorithm for simulating flows with complex 3D immersed boundaries on Cartesian grids". In: *Journal of Computational Physics* 191.2 (2003), pp. 660–669. ISSN: 0021-9991. DOI: https://doi.org/10.1016/S0021-9991(03)00321-8. URL: https://www.sciencedirect.com/science/article/pii/S0021999103003218.

[118] P. Di Stolfoa et al. *An easy treatment of hanging nodes in hp-finite elements*.

[119] A. Ern and J. L. Guermond. *Theory and practice of finite elements*. Apr. 2004. ISBN: 9780387205748.

[120] F. Mohebbi, B. Evans, and T. Rabczuk. "Solving direct and inverse heat conduction problems in functionally graded materials using an accurate and robust numerical method". In: *International Journal of Thermal Sciences* 159 (2021). URL: https://doi.org/10.1016/j.ijthermalsci.2020.106629.

[121] A. Fortin and A. Garon. *Les éléments finis : de la théorie à la pratique*. 1997-2006.

[122] A. Costa-Solé, E. Ruiz-Gironés, and J. Sarrate. "High-Order Hybridizable Discontinuous Galerkin Formulation for One-Phase Flow Through Porous Media". In: *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.

[123] J. C. C. Nitsche. "Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind". In: *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg* 36 (1971), pp. 9–15.

[124] K. Glasner. "Math 456: Partial differential equations - Dispersion relations, stability and linearization". In: (2008). Leture note - The University of Arizona. URL: https://www.math.arizona.edu/~kglasner/math456/dispersionstability.pdf.

[125] W. Roh and N. Kikuchi. "Analysis of Stefan Problem with Level Set Method". In: *8th AIAA/ASME Joint Thermophysics and Heat Transfer Conference* (June 2002). DOI: 10.2514/6.2002-2874.

[126] M. Shkolnikov, H. Soner, and V. Tissot-Daguette. *Deep Level-Set Method for Stefan Problems*. June 2023. DOI: 10.2139/ssrn.4523252.

[127] S. Chen et al. "A Simple Level Set Method for Solving Stefan Problems". In: *Journal of Computational Physics* 135.1 (1997), pp. 8–29. ISSN: 0021-9991. DOI: https://doi.org/10.1006/jcph.1997.5721. URL: https://www.sciencedirect.com/science/article/pii/S0021999197957211.

[128] H. Li, A. Goharzadeh, and Y. F. Yap. "A combined enthalpy-level-set approach for melting/solidification within two immiscible fluids". In: *International Journal of Thermal Sciences* 177 (2022), p. 107551. ISSN: 1290-0729. DOI: https://doi.org/10.1016/j.ijthermalsci.2022.107551. URL: https://www.sciencedirect.com/science/article/pii/S1290072922000904.

[129] A. Limare et al. "A hybrid level-set / embedded boundary method applied to solidification-melt problems". In: *Journal of Computational Physics* (2022). DOI: 10.1016/j.jcp.2022.111829. URL: https://inserm.hal.science/IJLRDA-FCIH/hal-03889680v1.

[130] C.-H. Yanga et al. "Optimal Surrogate Boundary Selection and Scalability Studies for the Shifted Boundary Method on Octree Meshes". In: *Preprint submitted to Journal of Computational Physics* (2023). URL: https://arxiv.org/pdf/2307.01479.pdf.