



HAL
open science

Méthodes numériques avancées pour la simulation du procédé de trempe industrielle

Chahrazade Bahbah

► **To cite this version:**

Chahrazade Bahbah. Méthodes numériques avancées pour la simulation du procédé de trempe industrielle. Mechanics of materials [physics.class-ph]. Université Paris sciences et lettres, 2020. English. NNT : 2020UPSLM012 . tel-03157524

HAL Id: tel-03157524

<https://pastel.hal.science/tel-03157524>

Submitted on 3 Mar 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



THÈSE DE DOCTORAT

DE L'UNIVERSITÉ PSL

Préparée à MINES ParisTech

Advanced Numerical Methods for the simulation of the industrial quenching process

Méthodes numériques avancées pour la simulation du procédé de trempe industrielle

Soutenue par

Chahrazade BAHBAH

Le 28 Janvier 2020

Ecole doctorale n° 364

Sciences fondamentales et appliquées

Spécialité

**Mathématiques numériques,
Calcul intensif et Données**

Composition du jury :

Pr. Mederic, ARGENTINA Inst de Physique de Nice	<i>Président</i>
Pr. Ramon, CODINA Univ. Politecnica de Catalunya	<i>Rapporteur</i>
Dr. Luisa, SILVA Ecole Centrale de Nantes	<i>Rapporteur</i>
Pr. Julien, Bruchon Mines Saint Etienne	<i>Examineur</i>
Mr. Benoit, DRIEU Linamar Montupet	<i>Invité</i>
Dr. Youssef, MESRI Mines Paristech	<i>Co-directeur de thèse</i>
Dr. Elisabeth, MASSONI Mines Paristech	<i>Co-directrice de thèse</i>
Pr. Elie, HACHEM Mines Paristech	<i>Directeur de thèse</i>

الى أبي و أمي، أغلى ما عندي...

Contents

Contents	i
List of Figures	v
List of Tables	ix
1 General introduction	1
1.1 Introduction to the industrial quenching process	2
1.2 Physics involved in the quenching process	3
1.3 Brief literature on the quenching process	5
1.4 Role of computational modeling in the design of the quenching process	6
1.5 Objectives of the thesis	7
1.6 Work environment	10
1.7 Author's contribution during the PhD	10
1.8 Layout of the thesis	11
1.9 Résumé du chapitre en français	12
1.10 Bibliography	14
2 Eulerian conservative and adaptive framework	17
2.1 Introduction	18
2.2 Anisotropic mesh adaptation	19
2.2.1 Edge-based error estimation	19
2.2.2 Gradient recovery procedure	20
2.2.3 Metric construction	21
2.2.4 Mesh adaptation criterion	21
2.3 Conservative interpolation of fields between meshes	23
2.3.1 Linear interpolation	24
2.3.2 Conservation of physical quantities	26
2.4 Parallelization of the conservative adaptive interpolation algorithm	28
2.4.1 Parallel mesh adaptation algorithm	28
2.4.2 Parallel implementation the conservative interpolation algorithm	30
2.5 Numerical examples	32
2.5.1 2D Analytical functions	32
2.5.2 3D Analytical test case	35
2.5.3 2D Analytical test case with mesh adaptation	35
2.5.4 2D Lid driven cavity : interpolation and dynamic mesh adaptation	36
2.5.5 3D Lid driven cavity : interpolation and dynamic mesh adaptation	40
2.5.6 Unsteady flow past a 3D vehicle model	41
2.5.7 Scalability study on the 3D Lid driven cavity	43
2.6 Conclusion	44

2.7	Résumé du chapitre en français	45
2.8	Bibliography	46
3	Moving interface capturing	51
3.1	Introduction	52
3.2	Level set function	53
3.2.1	Basic definition of the level set function	53
3.2.2	Convective reactive level set method	55
3.3	Mixing laws	59
3.4	The incompressible Navier Stokes equation	60
3.4.1	Governing equations	60
3.4.2	Galerkin finite element formulation	61
3.4.3	Variational multiscale (VMS) approximation	62
3.5	Numerical examples	66
3.5.1	2D dam break	67
3.5.2	2D droplet splashing on thin liquid film at different Reynold numbers	69
3.5.3	2D rising bubble	72
3.5.4	3D simulations of the axisymmetric and non-axisymmetric merging of two bubbles	74
3.6	Conclusion	80
3.7	Résumé du chapitre en français	80
3.8	Bibliography	81
4	Boiling multiphase flows : liquid-vapor-solid interactions	87
4.1	Introduction	88
4.2	Phase Change	88
4.2.1	Derivation of the governing equation without the phase change model	90
4.2.2	Derivation of the governing equation with the phase change model	91
4.2.3	Definition of the mass transfer rate	92
4.3	Surface tension	93
4.3.1	Standard definition	93
4.3.2	Semi-implicit time integration	94
4.4	Convection Diffusion Reaction equation	95
4.4.1	Governing equation	96
4.4.2	Standard Galerkin formuation	96
4.4.3	Streamline Upwind Petrov-Galerkin (SUPG) method	97
4.5	Numerical Examples	98
4.5.1	3D simulations of horizontal film boiling	98
4.5.1.1	Single film boiling	98
4.5.1.2	Multi film boiling	103
4.5.2	2D Quenching	107
4.6	Conclusion	110
4.7	Résumé du chapitre en français	111
4.8	Bibliography	111
5	Coupling and industrial applications	115
5.1	Introduction	116
5.2	Coupling between Cimlib-CFD and Z-set	117
5.3	Industrial applications	120

5.3.1	Quenching of a solid and comparison to experimental data	120
5.3.1.1	Set up	121
5.3.1.2	Results and discussions	122
5.3.2	Quenching of Cylinder head	127
5.3.2.1	Set up	128
5.3.2.2	Results and discussions	129
5.3.3	Challenging geometry : crossmember of a car	132
5.3.3.1	Set up	133
5.3.3.2	Results and discussions	134
5.4	Conclusion	140
5.5	Résumé du chapitre en français	140
5.6	Bibliography	141
6	Conclusion and perspectives	143
6.1	Conclusion	144
6.2	Perspectives	145
6.3	Résumé du chapitre en français	146

List of Figures

1.1	Industrial parts and quenching in a liquid medium. Pictures taken from montupet.fr and conmecheng.com	2
1.2	Nukiyama curve: evolution of the surface heat flux q as a function of Δt , [6]	4
1.3	Schematic representation of the quenching process	7
1.4	General flowchart for a quenching process	8
2.1	Patch $\Gamma(i)$ associated with node x^i	20
2.2	Mesh adaptation algorithm applied to two immersed bodies	22
2.3	Final mesh adapted to the different interfaces	23
2.4	Mesh adaptation and fields' interpolation between meshes	24
2.5	Left: P_1 nodal fields - Right: P_0 fields defined at Gauss points	25
2.6	P_1 interpolation from an old mesh to a new one, adopted from [38]	25
2.7	Conservative mesh adaptation procedure in parallel	28
2.8	Interface between two subdomains, and respective connected cells	30
2.9	Cell nomenclature in the subdomain processor 1	31
2.10	Assembly of the system in parallel	31
2.11	2D representation of the functions	33
2.12	Mass variation vs. number of nodes	34
2.13	Left: 3D Anisotropic mesh - Right: 3D Isotropic mesh	35
2.14	Meshes and 2D representations of the function	37
2.15	Set up of the 2D lid-driven cavity	38
2.16	Initial mesh for the 2D driven cavity flow	38
2.17	Anisotropic meshes at Reynolds 1000 and 5000.	39
2.18	Comparison of velocity profiles in the mid-planes for $Re = 1000$ (top), for $Re = 5000$ (bottom). Left: Velocity profiles for V_x along $y = 0.5$. Right: Velocity profiles for V_y along $x = 0.5$	39
2.19	Set up of the 3D lid-driven cavity	40
2.20	3D Anisotropic meshes at Reynolds 1000	41
2.21	Immersion of the vehicule geometry in a 3D domain	42
2.22	Set up of the 3D vehicle model	42
2.23	3D Anisotropic meshes for the unsteady flow past a vehicule	43
2.24	Scalability study on the conservative interpolation algorithm	44
3.1	Right: schematic representation of the level set function for multi-domain problems. Left: definition of the level set function.	54
3.2	Right: basic level set function. Left: truncated level set function.	56
3.3	Filtered level set function and interface refinement using anisotropic mesh adaptation.	58
3.4	Characteristic length for isotropic and anisotropic element based on a classical formula	66

3.5	Set up of the 2D dam break	67
3.6	Column fall evolution and refined meshes	68
3.7	Non-dimensional front position evolution	68
3.8	Non-dimensional column height evolution	69
3.9	Set up of the 2D droplet splashing on thin liquid film	70
3.10	Meshes evolution for the droplet splashing on a thin film at $Re = 20$, $We = 2000$	70
3.11	Evolution of the zero-iso-value of the level set for the droplet splashing on a thin film at $Re = 20$, $We = 2000$	70
3.12	Meshes evolution for the droplet splashing on a thin film at $Re = 100$, $We = 2000$	71
3.13	Evolution of the zero-iso-value of the level set for the droplet splashing on a thin film at $Re = 100$, $We = 2000$	71
3.14	Meshes evolution for the droplet splashing on a thin film at $Re = 1000$, $We = 2000$	72
3.15	Evolution of the zero-iso-value of the level set for the droplet splashing on a thin film at $Re = 1000$, $We = 2000$	72
3.16	Evolution of the instantaneous interface for the droplet splashing on a thin film for the different Reynolds numbers, adopted from the work of [62]	73
3.17	Set up of the 2D rising bubble	73
3.18	2D rising bubble evolution	74
3.19	Refined meshes for the 2D rising bubble	75
3.20	Set up of the 3D simulations of the axisymmetric and non-axisymmetric merging of two bubbles	75
3.21	Comparison between the experimental measurement of [66] and the simulation for the co-axial coalescence of two bubbles	76
3.22	Comparison between the experimental measurement of [66] and the simulation for the oblique coalescence of two bubbles	77
3.23	Evolution of the refined meshes the 3D bubble shape for the co-axial coalescence	78
3.24	Evolution of the refined meshes the 3D bubble shape for the oblique coalescence	79
4.1	Heat fluxes from liquid and vapor phases to interface	92
4.2	Interface position for a volume of liquid that has vaporized in the time interval $[t, t + \Delta t]$	93
4.3	Representation of the surface tension force	93
4.4	Set up of the 3D single film boiling	98
4.5	Evolution of the liquid/vapor interface and meshes during 3D film boiling	100
4.6	Evolution of the liquid/vapor interface and velocity during 3D film boiling	101
4.7	Evolution of the temperature field and the interface location for single film boiling	102
4.8	Evolution of Nu for the single film boiling in 3D	103
4.9	Set up of the 3D multi film boiling	103
4.10	Evolution of the liquid/vapor interface and meshes during 3D multi film boiling	104
4.11	Evolution of the liquid/vapor interface and velocity during 3D multi film boiling	105
4.12	Evolution of the temperature field and the interface location for multi film boiling	106

4.13 Set up of the 2D quenching	107
4.14 Meshes and liquid/vapor interface location at times $t = 0, 1, 6, 25, 40, 60, 70,$ 90, 120 s	108
4.15 Temperature evolution and liquid/vapor interface location at times $t = 0, 1,$ 6, 25, 40, 60, 70, 90, 120 s	109
4.16 Sensors positions inside the solid part	110
4.17 Evolution of the temperature inside the solid for different positions	110
5.1 General diagram for a quenching process simulation	117
5.2 Division into two domains : Fluid-Solid and Solid	118
5.3 Coupling scheme between Cimlib-CFD and Z-set	119
5.4 Ring geometry	120
5.5 Immersion of the solid geometry in the fluid domain	121
5.6 Set up for the simulation of the quenching of a Ring geometry	122
5.7 Evolution of the liquid/vapor interface during the boiling at times $t = 0, 1, 5,$ 7, 8 s	123
5.8 Simulation results. Temperature distribution along the sectional plane x at times $t = 0, 0.15, 1, 2, 5, 7.5, 10, 12, 15$ s	124
5.9 Simulation results. Contour plot of temperature distribution on the solid domain at different times $t = 0, 0.15, 1, 2, 5, 7.5, 10, 12, 15$ s	125
5.10 Position of the thermocouples for the experimental analysis	126
5.11 Comparison between experimental data and numerical results	126
5.12 Position of sensors on the Ring geometry	127
5.13 Cylinder head geometry	128
5.14 Immersion of the solid geometry in the fluid domain	128
5.15 Set up for the simulation of the quenching of a cylinder head geometry	129
5.16 Simulation results. Evolution of meshes during the boiling at times $t = 0,$ 0.005, 0.15, 0.3, 0.5, 2 s	130
5.17 Simulation results. Evolution of the liquid/vapor interface during the boil- ing at times $t = 0, 0.005, 0.15, 0.3, 0.5, 2$ s	131
5.18 Simulation results. Temperature distribution along the sectional plane x at different times $t = 0, 0.2, 0.5, 0.7, 1, 1.2$ s	132
5.19 Simulation results. Temperature distribution on the surface of the geomtry $t = 0, 0.2, 0.5, 0.7, 1, 1.2$ s	133
5.20 Crossmember geometry	134
5.21 Set up for the simulation of the quenching of the crossmember geometry. Horizontal and vertical quenching	134
5.22 Evolution of the liquid/vapor interface for the horizontal quenching	135
5.23 Evolution of the liquid/vapor interface for the vertical quenching	136
5.24 Snapshot of the vapor field inside the solid geometry for the horizontal (left) and vertical (right) quenching taken at the same time	137
5.25 Set up for the simulation of the quenching of the crossmember geometry. Horizontal quenching	137
5.26 Simulation results. Temperature distribution along the sectional plane x at different times.	138
5.27 Simulation results. Temperature distribution along the sectional plane x at different times.	139

List of Tables

2.1	2D meshes used for the different transfers	34
2.2	Error for the transfer from anisotropic to isotropic	36
2.3	Comparison of the mass variation between both methods	36
2.4	Final momentum loss in direction x	39
2.5	Final momentum loss in direction y	40
2.6	Final momentum loss in each direction for the 3D lid driven cavity	40
2.7	Final momentum loss in each direction for the unsteady flow past a vehicle	42
2.8	Computational time speed up on the Cluster Intel	43
3.1	Pourcentage of error evolution for both level set methods	57
3.2	Pourcentage of error evolution for both level set methods	57
3.3	Pourcentage of error evolution for both level set methods	57
3.4	Physical parametres used for the different simulations of the 2D droplet splashing on thin liquid film	69
3.5	Mass variation for the 2D rising bubble with 5000 elements	74
3.6	Mass variation for the 2D rising bubble with 10 000 elements	74
4.1	Physical parameters for the 3D simulation of film boiling	99
4.2	Physical parameters for the 2D quenching process simulation	107
5.1	Physical parametres used for first simulation of the quenching of a Ring	122
5.2	Physical parametres of the immersed solid while considering the phase trans- formation	123

Chapter 1

General introduction

Contents

1.1 Introduction to the industrial quenching process	2
1.2 Physics involved in the quenching process	3
1.3 Brief literature on the quenching process	5
1.4 Role of computational modeling in the design of the quenching process	6
1.5 Objectives of the thesis	7
1.6 Work environment	10
1.7 Author's contribution during the PhD	10
1.8 Layout of the thesis	11
1.9 Résumé du chapitre en français	12
1.10 Bibliography	14

1.1 Introduction to the industrial quenching process

Thermal Treatment describes the multifaceted operations in heating furnaces and quenching tanks, performed on a material in the solid state, for the purpose of altering its microstructure and properties. The output of this step is the input of all the following manufacturing steps such as forging, rolling processes and even the prediction of microstructure evolution. Therefore, any lack of control in this upstream operation will affect the global manufacturing chain, and the consequences are then immediate such as prohibiting better quality, higher availability and adaptability of products. This leads to a need of improving our understanding of the involved physical phenomena, particularly for quenching processes. The modeling of the quenching process has started drawing attention of many more investigators as a result of the demand of many industrials, especially the automotive, nuclear and aerospace industry. Research in both experimental and numerical areas and through mathematical models has proven to be effective in accelerating the understanding of complex problems as well as helping decrease the development costs for new processes, [1–4]. In the past, the optimization and savings in large productions were made only by large companies that could support and afford the cost of sophisticated modeling tools, specialized engineers and computer software. Nowadays, modeling has become an essential element of research and development for many industrials; and realistic models of complex three dimensional quenching processes can be feasible on a personal computer. Quenching is a process that belongs to the fam-



Figure 1.1: Industrial parts and quenching in a liquid medium. Pictures taken from montupet.fr and conmecheng.com

ily of thermal treatments, the process is illustrated in Figure (1.1). The purpose of the quenching process is to give a certain micro-structure to the metal in order to achieve the required mechanical performance. This process has direct impacts on changing mechanical properties, controlling micro-structure and releasing residual stresses. Quenching is

a heat treatment method where a hot metal part is cooled down rapidly with the help of a quenchant such as air, water, oil, other liquids, or combinations of them. Good control of quenching is essential for correctly controlling the phase changes that take place within the alloy, and obtain the micro-structure exhibiting the desired thermomechanical properties. Different parameters are controlled by the manufacturers, such as the tempering time, the chemical and thermal properties of the quenchant, or the number and the solid parts in the quenching tank. Moreover, the coolant flow structure is very important to predict heat exchanges between the solid and the hot fluid.

This Phd is done in collaboration with the company Linamar Montupet specialized in the manufacture of complex cast aluminum components for the automotive industry. They are interested in the quenching of metallic parts in liquid quenchant that can vaporize. The vaporization is generally the leading phenomenon that drives the system. Indeed, the cooling of the part is strongly conditioned by the behavior of the surrounding fluid that extracts the heat therein. The liquid determines the heat transfer at the surface of the part. Thus the study of two-phase flows with phase change, thermal transfers and fluid-solid interactions is a first step to a better understanding of quenching processes.

1.2 Physics involved in the quenching process

As explained previously, during the quenching process, an evaporation occurs at the interface between the hot solid part and the surrounding liquid, we call it phase change. This phenomenon is present in various industrial processes but also in natural phenomena such as boiling of water, condensation that creates clouds, icing of aircraft wings, etc. Each kind of phase change is characterized by triggering mechanisms induced by temperature or pressure variations. In the quenching process, the phase change is boiling and it is due to the metallic part that provides the required heat. Indeed, during a liquid quenching, the orders of magnitude between the temperatures involved are so high that the boiling phenomena occur during most of the process. Thus, understanding these phenomena is crucial to properly control the quenching. Therefore, in order to understand the physics behind the quenching process, we must understand the different boiling modes involved. To do so, the classic approach is to consider a relative difference temperature ΔT between the hot surface and the saturation temperature of the fluid and then analyze the heat flux q at the interface between the solid and the fluid. It is common to draw the Nukiyama curve (see Figure 1.2), proposed for the first time by [5], showing the heat flux as a function of the difference in temperature. It highlights the different boiling regimes of water in contact with a heated metal. As one progresses from the saturation temperature to higher temperatures, more and more bubbles appear and then merge into a vapor film, which act as a thermal insulation. As see in Figure (1.2), we can usually distinguish five boiling modes :

- The **natural convection** that starts before the saturation of the water, usually caused by the variation of the density of the fluid. Indeed, when the temperature gets higher, the density value decreases and on the opposite, with a lower temperature, we have a higher density. In this mode, no bubbles are yet created. It is easily modeled numerically, and many correlations are available in the literature to estimate the natural convection of a liquid around a hot solid.
- The **partial nucleate boiling** is the first step of the boiling. Small bubbles of vapor start to form on preferential sites called nucleation sites. In the case of no forced

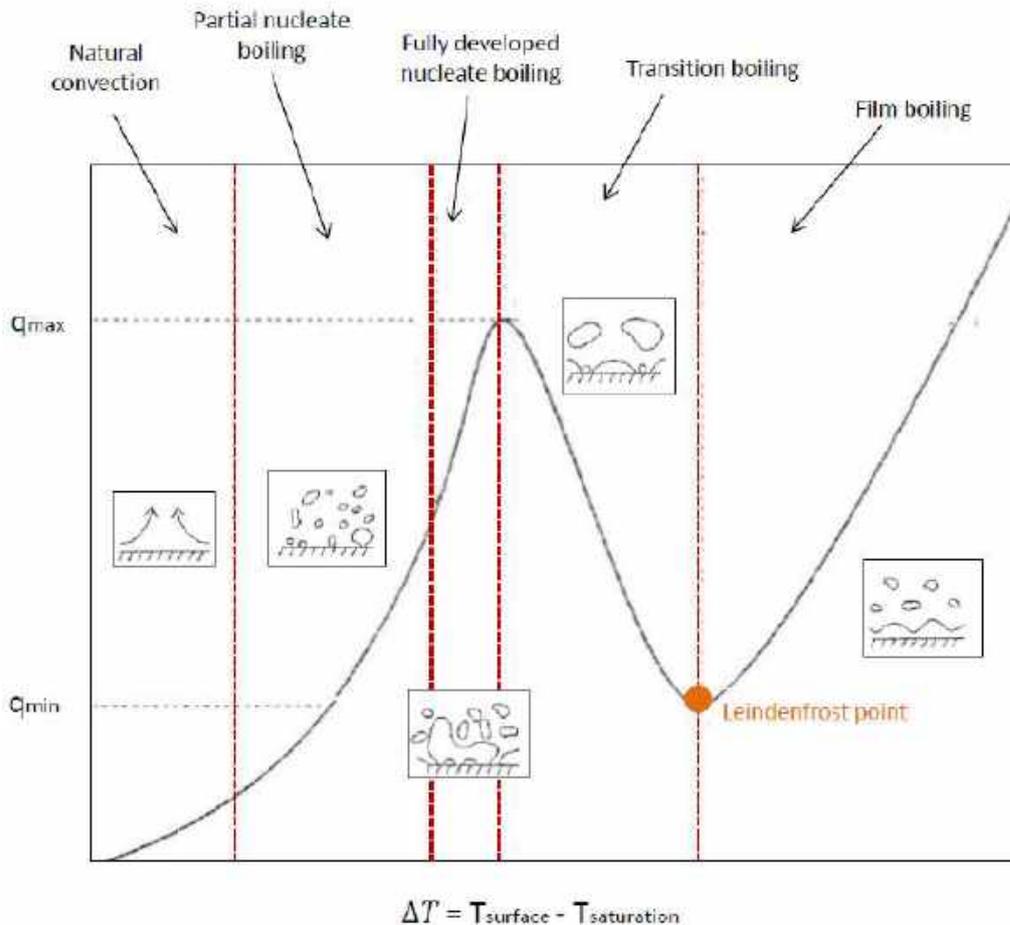


Figure 1.2: Nukiyama curve: evolution of the surface heat flux q as a function of Δt , [6]

convection, this is the most ordered boiling mode: bubbles form at nucleation sites and are released once critical size is exceeded. These sites are sufficiently spaced and the frequency of creation of bubbles is sufficiently small, and thus no bubble coalescence occurs.

- The **fully developed nucleate boiling** starts when the heat transfer is important enough so that the boiling occurs in the whole surface of the solid. In this mode, the evaporation is so important that the vapor bubbles start moving, coalesce and create unstable columns of vapor that leave the surface in a chaotic way. It is the most efficient boiling mode in terms of heat transfer. Indeed, most of the heat exchanges happen near these column of bubbles which combine evaporation and convection.
- The **transition boiling** is a mode where boiling is so important that in some regions, a partial vapor film start to create, blocking the contact between the liquid and the solid part. This degrades the heat transfer, which results in a negative slope on the curve from Nukiyama. This mode generally does not last long during the quenching process.
- The **film boiling** is a mode where a continuous vapor film covers the entire hot surface of the solid part. The relative difference temperature is so high that the vaporization takes place directly at the interface between the film and the liquid. This film has an isolating effect, thus greatly limiting the heat transfer between the liquid and the solid part. This mode starts at the minimal temperature $T_{min} = 100^{\circ}\text{C}$

also called Leidenfrost temperature, which refers to the Leidenfrost effect. It is the phenomenon of calefaction that appears when a drop is placed on a hot plate: the vapor film thus created makes it "levitate" above the plate for a time that can be quite long (more than a minute depending on conditions).

In the case of industrial applications, nucleate boiling represents the most efficient boiling heat transfer regime. In quenching processes, the temperature is considerably larger than the Leidenfrost temperature. Thus, a vapor film surrounds instantaneously the part and prevents it from cooling.

1.3 Brief literature on the quenching process

In order to give an idea of the evolution of the numerical modeling of the quenching process, a brief literature review will be presented here. Major influencing factors such as the design of the quenching tank, the complexity of the solid geometry, the location of the solid part in the tank and the surface state of the part where pointed out in several studies, [4; 7]. In the article of [8], it was discussed that the kind of quenching medium, the selection of quenching medium temperature and the selection of the medium state (unagitated, agitated) are determining factors. Indeed, the effect of agitation on the performance of various quenchant has been analyzed in [3; 9; 10], and it was shown that the agitation clearly affects the depth of the hardness of the solid part, since it is partly responsible of the mechanical rupture of the vapor film. On one hand, [11] pointed out that the thermal characteristics of the liquid are very important. For instance, the vaporization temperature has a direct impact on the different boiling modes. Indeed, the fluid might in some cases store more energy per unit volume, and this will increase the cooling rate since the temperature of the fluid is increased less fast; and it can be the other way around. On the other hand, in [12], it was pointed out that the thermal properties of the solid have an important influence on the boiling during the quenching process. The surface characteristics of the hot part such as roughness, oxidation and wettability are among those parameters. Another study in [13] showed that the orientation and the position of the hot part in the quenching tank, as well as the shape of the geometry play an important role in the quenching process. In fact, considerable differences were observed between a horizontal and a vertical surface in terms of behavior such as the bubble generation, growth and detachment.

In view of this brief literature review, it is clear that heat transfer during the film boiling and nucleate boiling processes during heat transfer in vaporizable liquids such as water must be as uniform as possible throughout the quenching process. This heat treatment process must take into account optimal combinations of parameters with their complexity in order to obtain the desired metallurgical properties such as hardness and yield strength. To this day, the physical modeling of the problem and the numerical simulation have proven to be a very powerful tool for predicting all the physical phenomena and controlling them thanks to the increasing performance of computational resources. For instance, the use of inverse analysis to deduce heat transfer coefficients was proposed by [11; 14]. This method is called the Heat Transfer Coefficient but has its limits. This allow to simply and quickly deduce the cooling of a solid, however, without any knowledge of the fluid behavior. Moreover, it requires experiments and is not very accurate and gives no information on the underlying physics. As it was pointed out in [11; 15], this method is satisfying in case of simple geometries but in an industrial context, if we consider an

experimental investigation, it remains time consuming and difficult to achieve and not so reliable. A full experimental optimization of this process is not a viable strategy due to the cost of the processes involved. Thus, it has become very important to simulate and visualize the complexity of the flows (liquid-vapor phase transition, agitation,...) and to deal with fluid-solid heat coupling. In the literature, there have been number of recent publications illustrating the enormous potential in the understanding of the quenching process through CFD (Computational Fluid Dynamics) analysis. It can be used to design new and innovative quench systems with the objective of optimizing the quenching performance for distortion control and reduction of cracking problems. The author in [16] proposed a historical review of the numerical methods that have been developed to simulate the quenching process since the 80s. The work of Garwood and al. [1] was one of the first attempts to characterize a quench tank using computational fluid dynamics; they were able to predict qualitatively the main flow features in the quenchant. Others proposed to combine the flow motion with interface tracking methods to follow the evolution of the liquid-vapor-solid interface. For instance, [17] presented the numerical simulations of a real cylinder head quench cooling process employing a new boiling phase change model; separated computational domains are constructed for the solid and liquid regions and are numerically coupled at the different interfaces. However, the modeling of the nucleation and pure convection mode is not carried out. In [18], the forced convection quenching process of hot pieces in the subcooled liquid is numerically investigated. Their method is validated and they show that there is a good agreement between the developed code and existing analytical solution and also experimental data. However, the modeling of the partial nucleate boiling is not performed and the surface tension model surface is not robust. Today, there is still room for progress in terms of modeling the quenching process.

1.4 Role of computational modeling in the design of the quenching process

As explained previously, quenching is a complex thermomechanical process involving the coupling of multiple physical processes. These processes are generally multi-parameters controlled. A deviation during the quenching process has immediate consequences on the quality of the parts. Moreover, because the quality of the quench also depends on the internal shape of the parts, it is not possible to draw a general conclusion on how to perfectly quench a complex geometry. In other words, the quenching of a complex part is always a matter of compromise. This is where numerical modeling can be a great help, particularly regarding the prediction of residual stresses. A way to circumvent the need of tremendous computational resources is to identify the phenomena to simulate, with satisfactory precision and fidelity with respect to the real process. To enable the numerical tool to be predictive, only phenomena with no significant impact on the results must be neglected. Thus, the major factor to be considered in the modeling of a quenching process is the phase change and the heat transfer at the interface between the solid and the fluid, see Figure 1.3. To study and optimize a quenching process, the heat transfer in the quenching tank has to be modeled in the same way of a real situation as closely as possible. Given the geometry of the solid part, different boundary conditions, fluid composition and properties and other complexities, an analytical solution is not feasible and computational modeling has to be resorted to. Over the last 20 years, CFD has gained its reputation of being an efficient tool in identifying and solving such problems. The tools

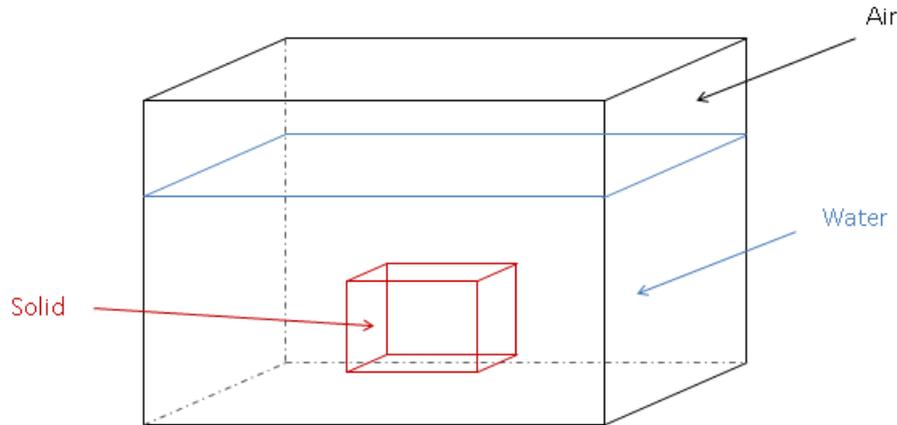


Figure 1.3: Schematic representation of the quenching process

used in this thesis are the Finite Element Method (FEM) and Computational Fluid Dynamics (CFD). This method is shown as an attractive way to solve turbulent flows and heat transfers and it can be applied for a variety of solid geometry and boundary conditions. The main process is detailed in the flowchart presented in Figure (2.4).

1.5 Objectives of the thesis

As explained previously, although computational fluid dynamics are being used increasingly in quenching tank design, there is still considerable imprecision due to assumptions that must be made in particular the use of simple geometries and approximated quenching environment. Today, there is a strong demand from many industrial companies to control this heat treatment process taking into account optimal combinations of parameters with their complexity in order to obtain the desired metallurgical properties such as hardness and yield strength. Thus, the objective of this thesis is to set a numerical framework able to simulate the quenching process at an industrial scale. In this thesis, two main aspects will be studied, analyze and simulate the liquid-vapor-solid interactions and the implementation of the coupling between two codes (Cimlib-CFD and Z-set). The results coming from these numerical development will be validated by confrontations with the experiments proposed in agreement with the company Linamar Montupet.

Due to the complexities of the physics that may occur for such application, the modeling of the fluid-solid interaction is not straightforward. The proposed numerical method for modeling such multiphase applications (gas/fluid/solid) will be referred as the immersed volume method (IVM). It allows an improved, simple and accurate resolution; in particular at the interface between the fluid and solid. This method simplifies considerably the geometric definition and the resolutions, and has been widely used for multiphase applications. The direct numerical simulation of multiphase flows with phase change requires the use of a method to locate the liquid/vapor interface. In fact, to locate each interface, a signed distance function called level set is used, this approach is defined as a monolithic approach since it consists in solving a single set of equations for the whole computation domain, and this will allow to take into all the different scales of the problem; from the millimeter of the vapor bubble to the meter of the complex geometries. The first step towards a more accurate framework is to focus on evaluating and then improv-

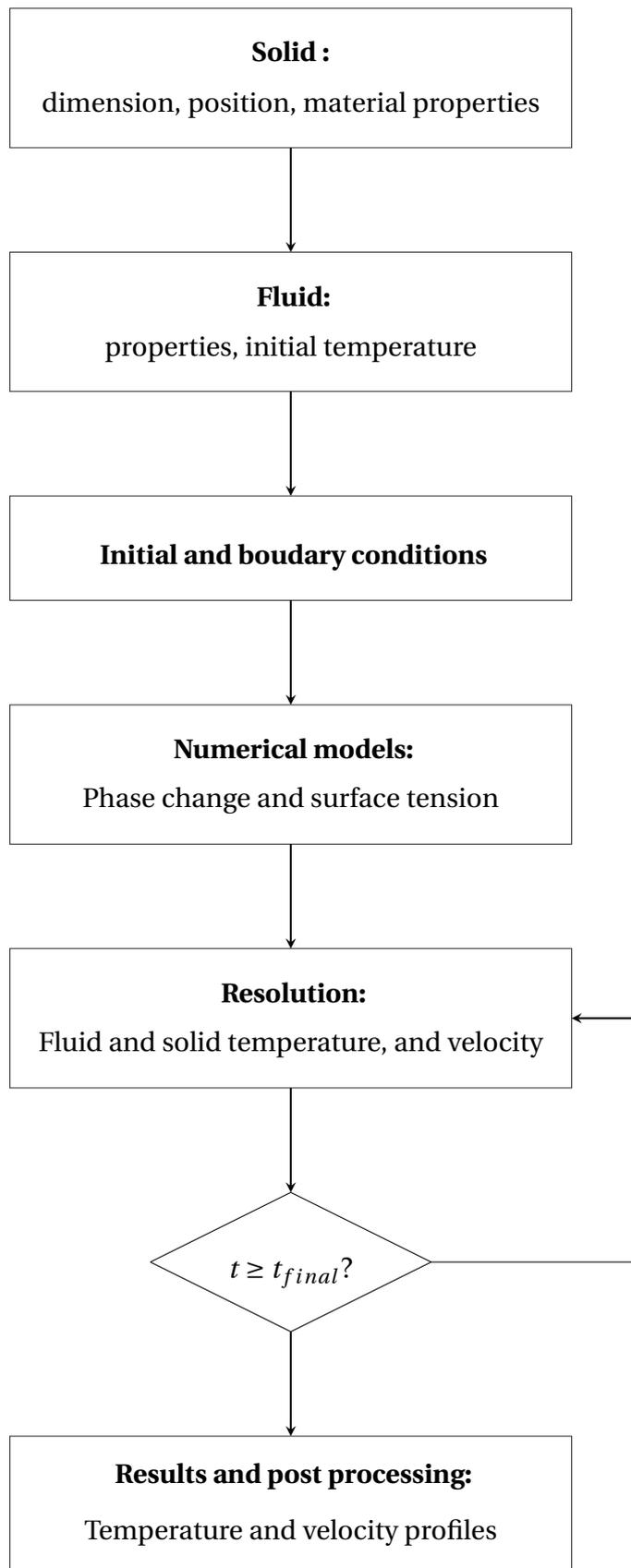


Figure 1.4: General flowchart for a quenching process

ing the existing level method : the presence of strong thermal gradients at the interface, these equations are not dubious of the resolution by stabilized element which ensures the good convergence of the solution. A full description and details about this method will be given. Furthermore, the immersed volume method is combined with an anisotropic mesh adaptation technique to ensure an accurate capture of the different interfaces. The use of a well adapted mesh along the interfaces ensures an accurate distribution of the thermo-mechanical parameters over the computational domain. The mesh is anisotropic and well adapted along the fluid/solid interfaces and isotropic with a relatively small background mesh size in the rest of the domain.

Despite the considerable advances in computational fluid dynamics (CFD) and the increasing computer power, different challenges still need to be addressed for providing accurate and efficient simulations. The problem can be cast as a thermal fluid-structure interaction one involving the simultaneous resolution of turbulent flows with phase change and conjugate heat transfers between the solid and the fluid subdomains. Indeed, this framework must take into account turbulent boiling (with or without agitators), phase change natural and forced convection and therefore the complexity of industrial cases with a high thermal gradient. The industrial objective will therefore be to characterize the various boiling regimes observed under transient conditions, to precisely measure the associated heat transfer, to couple the thermal field and finally to be able to control the cooling rate of the process. Indeed, it becomes necessary, for a complete quenching simulation to take into account more precisely the evolution of the liquid and vapor phases: heating regime, nucleate boiling regime and convection regime. Finally, it is important to mention that classical Finite Element methods to solve the unsteady Navier-Stokes and heat transfer equations suffer from lack of stability, in particular at high Reynolds and Peclet numbers. These sources of numerical difficulties have been treated using different approaches and these finite element solvers are already implemented in our library.

In the case of the quenching process, the phenomena studied concern both the phase change at small scales (germination and growth of a vapor bubble) and the boiling phenomena at large scales occurring during the quenching but also the phase transformation of the treated part and the residual stresses. Residual stresses developed after quenching of alloys cause distortion during subsequent machining. As a result, machined parts may be out of tolerance and have to be cold worked or re-machined. Experimental measurement of residual stresses is lengthy, tedious and very expensive. Thus, a simulation that takes into account the phase transformation inside the solid is mandatory; this is the second objective of the thesis. The objective of this task is to develop a generalist "gateway" between two codes capable of accurately transferring fields from a fluid-structure mesh to a structure mesh. This structure mesh will be used in the software Z-set to perform residual stress analysis. Several interpolation techniques of the fields (temperature, flux, etc ...) exist in the literature. In our library, we use only 2D triangular elements and tetrahedral elements in 3D. The implemented interpolation method must be robust and optimal in terms of computational time, works in parallel on multiple cores and most importantly, must be conservative. Note that the interpolation is not only used for the coupling, but most of all, during the anisotropic adaptation procedure. And recall that in multiphase applications, the mesh adaptation is user every time step. An improvement on the linear interpolation method will therefore be proposed. Finally, to validate and to assess the performance of the developed framework, experimental data will be provided by the industrial partner and the numerical validation of the new complete quenching

model will be done. The complexity of the geometry, the calculation time, the flexibility for the implementation of the simulations, taking into account the complete quenching tank with the entry and the exit of water to maintain the temperature of the quench tank, the test coupling with Z-set, taking into account the phase transformation and the residual stresses, will form all points of interest in this task.

To sum up, all these following tasks have been achieved:

- Develop advanced numerical methods to describe the interfaces liquid/vapor/solid
- Couple the multiphase flows with the convection-diffusion-reaction equations to model the phase change at the liquid-vapor interface
- Use the latest development on the mesh adaptation procedure and parallel computation to handle large complex industrial simulations
- Conservative interpolation between unrelated unstructured meshes and coupling between two codes
- Validate the numerical framework by comparison with the experimental data provided by real quenching simulations provided by the industrial partner Linamar Montupet

To summarize, the originality of this work lies in the simulation of realistic industrial conditions and quenching devices, and robustness by limiting the assumptions made and restraining non-physical use of quenching parameters. All those elements represent the features dedicated to industrial abilities of the framework.

1.6 Work environment

In this thesis, all the numerical implementations of the developed methods are carried out using the finite element library Cimlib-CFD; which stands for CIM as Advanced Computing in Material forming research group and LIB for library and CFD, computational fluid dynamics. It is developed by the CFL team which stands for Computers and Fluids. It is the base for different numerical applications developed at CEMEF (www.cemef.mines-paristech.fr), in collaboration with other research teams and industrial partners. This scientific library represents an Object Oriented Program and a fully parallel code, written in C++, gathers the numerical development of the group (Ph.D. students, researcher, associate professor...). Cimlib-CFD aims at providing a set of components that can be organized to build numerical simulation of different industrial processes and the present one is the simulation of the quenching process for the company Linamar Montupet.

1.7 Author's contribution during the PhD

The contributions of the author in terms of publications, oral communications and prizes are presented below.

Journal Articles

1. Bahbah, C., Mesri, Y., & Hachem, E. *Interpolation with restrictions in an anisotropic adaptive finite element framework*. Finite Elements in Analysis and Design, 2018, 142, 30-41.
2. Bahbah, C., Khalloufi, M., Larcher, A., Mesri, Y., Coupez, T., Valette, R., & Hachem, E. *Conservative and adaptive level-set method for the simulation of two-fluid flows*. Computers & Fluids, 2019, 191.
3. Bahbah, C. & Hachem, E. *Fluid-solid coupling for simulating the temperature evolution during the quenching process*. In progress

Communications

1. Bahbah, C., Mesri, Y., & Hachem, E. *Éléments finis adaptatifs pour la simulation des phénomènes interfaciaux avec changement de phase*. 13ème Colloque National en Calcul des Structures, France 2017.
2. Bahbah, C., Mesri, Y., & Hachem, E. *Interpolation with restrictions in an anisotropic adaptive finite element framework for CFD applications*. International Workshop on Complex Turbulent Flows, Morocco 2017.
3. Bahbah, C. & Hachem, E. *Accurate Adaptive Eulerian Framework for Liquid-gas-solid interactions*. Fluids and Complexity, France 2018.
4. Bahbah, C., Khalloufi, M., Mesri, Y., & Hachem, E. *Accurate Adaptive Eulerian Framework for Liquid-gas-solid interactions*. 13th World Congress on Computational Mechanics (WCCM XIII), USA 2018.
5. Bahbah, C. & Hachem, E. *Accurate Adaptive Eulerian Framework for Liquid-gas-solid interactions*. UCA Complex days, France 2019.

Awards

1. Price for the best poster in the session numerical modeling for multiphysics, Conference CSMA 2017.
2. Finalist of the Pierre Laffitte medal 2018. The medal was created in the honor of the Professor Pierre Laffitte who was the former director of MINES ParisTech (1972-1984) and senator of the region Alpes Maritimes in France (1995 – 2008).

1.8 Layout of the thesis

The thesis is divided into six chapters. Chapter 1 is an introduction to the topic considered in this thesis. Chapter 2 is dedicated to the Eulerian conservative framework; the Immersed Volume Method (IVM) coupled to the mesh adaptation technique and a newly developed parallel conservative interpolation method are presented. In Chapter 3, we present the moving interface capturing method to follow the evolution of the different interfaces in multiphase flows. A new definition of the level set function is presented and special attention is given to stabilization methods for solving the Navier-Stokes equations. The mathematical modeling of the phase change at the interface as well as the stabilized finite element used for solving the conjugate heat transfer are detailed in Chapter 4. The coupling between our code Cimlib-CFD and the software Z-set is explained in Chapter

6. The validation of the framework on some industrial applications is also detailed in the second part of this chapter. The computation of different benchmarks tests has been also carried out and several comparisons with experimental works will be also presented along this manuscript. Finally, in the last chapter, conclusion and the possible extension of the present work to include more features is explored.

1.9 Résumé du chapitre en français

Ce chapitre d'introduction nous permet d'abord de présenter une description générale autour des enjeux industriels et numériques liés au procédé de trempe. La trempe est une méthode de traitement thermique où un métal chaud est refroidi rapidement à l'aide d'un medium tel que l'air, l'eau, l'huile ou d'autres liquides ou de combinaisons de ceux-ci. Il a pour but de donner au métal une certaine microstructure afin d'atteindre les performances mécaniques requises. Ce procédé a des impacts directs sur l'évolution des propriétés mécaniques, le contrôle de la microstructure et la libération des contraintes résiduelles. Afin de réaliser un procédé optimal, il est essentiel de contrôler correctement les transformations de phase qui ont lieu dans l'alliage, et ainsi obtenir la microstructure présentant les propriétés thermomécaniques souhaitées. Différents paramètres sont contrôlés par les fabricants, comme le temps de trempe, les propriétés chimiques et thermiques du medium, ou le nombre de pièces solides dans le réservoir de trempe.

Cette thèse est réalisée en collaboration avec la société Linamar Montupet spécialisée dans la fabrication de composants complexes en aluminium pour l'industrie automobile. Ils s'intéressent à la trempe des pièces métalliques dans les medium liquides pouvant vaporiser. La vaporisation est généralement le principal phénomène qui anime le système. En effet, le refroidissement de la pièce est fortement conditionné par le comportement du fluide environnant qui extrait la chaleur provenant de la pièce chaude. Le liquide détermine le transfert de chaleur à la surface de la pièce. Ainsi, l'étude des écoulements multiphasiques à changement de phase, des transferts thermiques et des interactions fluide-solide est un premier pas vers une meilleure compréhension des procédés de trempe. En effet, durant le procédé de trempe, une évaporation se produit à l'interface entre la partie solide chaude et le liquide environnant, on parle de changement de phase. Ce phénomène est présent dans divers procédés industriels, mais également dans des phénomènes naturels tels que l'ébullition de l'eau, la condensation qui crée des nuages, le glaçage de ailes d'avion, etc. Chaque type de changement de phase est caractérisé par des mécanismes de déclenchement induits par des variations de température ou de pression. Dans le processus de trempe, le changement de phase est l'ébullition et il est dû au solide qui fournit la chaleur requise. En effet, lors d'une trempe liquide, les ordres de grandeur entre les températures en jeu sont si élevés que les phénomènes d'ébullition se produisent pendant la majeure partie du processus.

L'analyse bibliographique a permis de constater que les informations sur l'ébullition transitoire ne sont pas complètes. En effet, il est connu qu'une fois l'ébullition déclenchée, ce sont les bulles qui sont principalement responsables du transfert de la chaleur de la paroi vers le liquide. Le diamètre des bulles, la fréquence de détachement et la densité des sites de nucléation sont donc importants. À faible surchauffe ou faible densité de flux, les bulles sont formées à partir de quelques sites de nucléation uniquement. Ce régime s'appelle « régime de bulles isolées ». Au transfert par convection dans le liquide hors de la zone d'influence des bulles, s'ajoute le transfert thermique dû aux bulles. Les bulles

naissent à la paroi, grandissent et se détachent et leur cycle de vie améliore les échanges thermiques. Aux deux mécanismes mentionnés, un troisième peut être ajouté, il s'agit de l'évaporation de la microcouche qui existe au pied de la bulle ou ébullition nucléée. Une optimisation expérimentale complète de ce processus n'est pas une stratégie viable en raison du coût des processus impliqués. Bien que la dynamique des fluides soit de plus en plus utilisée dans la conception des réservoirs de trempe, le manque de précision est toujours un problème majeur en raison des hypothèses qui doivent être faites en amont ; particulier l'utilisation de géométries simples et d'un environnement de trempe très approximatif. L'amélioration et le contrôle de ce procédé suscitent un intérêt grandissant et deviennent un axe majeur de progrès pour les industriels.

L'objectif scientifique de ce projet de thèse est d'étendre le premier modèle de trempe pour passer de l'échelle académique à l'échelle industrielle tout en prenant en compte l'ébullition turbulente (avec ou sans agitateurs), le changement de phase, convection naturelle et forcée et donc la complexité des cas industriels à fort gradient thermique et finalement de confronter les nouveaux développements aux expériences proposés en accord avec Montupet. L'objectif industriel sera donc de caractériser les différents régimes d'ébullition observés, de mesurer précisément les transferts thermiques associés, de coupler le champ thermique, et finalement de pouvoir contrôler la vitesse de refroidissement du procédé. En effet il devient nécessaire, pour une simulation complète de trempe, de retravailler les solveurs numériques existants pour prendre en compte plus précisément l'évolution des phases liquide et vapeur : régime de caléfaction, régime d'ébullition nucléée et régime de convection. Les phénomènes étudiés concernent aussi bien la transformation de phase de la pièce traitée et les contraintes résiduelles, le changement de phase aux petites échelles (germination et croissance d'une bulle de vapeur) et les phénomènes d'ébullitions aux grandes échelles intervenant lors de la trempe. En se basant sur les outils existants au sein de la librairie de calcul élément finis de l'équipe Calcul Intensif et Mécanique des Fluides (CFL) du Centre de Mise en Forme des Matériaux (CEMEF) de l'école des Mines de Paris (Mines ParisTech), nous avons identifié les nouveaux besoins numériques. La méthode d'immersion de volume (IVM) développée par l'équipe CFL est adoptée pour prendre en compte les interactions fluide/structure et la distribution des propriétés thermo-mécaniques. Maintenir un bon niveau de précision nécessite un maillage très fin, ainsi l'adaptation de maillage anisotrope est utilisée, ce qui permet une résolution fine des physiques aux interfaces. Ainsi, dans cette thèse, deux aspects principaux seront étudiés: analyser et simuler les interactions liquide-vapeur-solide et mettre en œuvre le couplage entre deux codes (Cimlib-CFD et Z-set) afin de prendre en compte la transformation de phase au sein du solide. Pour ce faire, ces différentes tâches ont été réalisées :

- Développer des méthodes numériques pour décrire les interfaces liquide/gaz/solide
- Coupler les écoulements multiphasiques avec la thermique pour le changement de phase (liquide/gaz)
- Utiliser les derniers travaux sur l'adaptation de maillage et calcul parallèle pour traiter des cas à l'échelle industrielle
- Interpolation conservative des champs entre deux maillages et couplage entre deux codes
- Valider les résultats numériques par des confrontations avec des résultats expérimentaux en collaboration avec Linamar Montupet.

En résumé, l'originalité de ce travail réside dans la simulation de conditions industrielles et de dispositifs de trempe réalistes. Ce nouveau framework permet l'optimisation et le contrôle de ce procédé, ce qui représente un axe majeur de progrès pour les industriels.

1.10 Bibliography

- [1] D. Garwood, J. Lucas, R. Wallis, J. Ward, Modeling of the flow distribution in an oil quench tank, *Journal of Materials Engineering and Performance* 1 (6) (1992) 781–787. [2, 6](#)
- [2] D. M. Wang, A. Alajbegovic, X. Su, J. Jan, Numerical simulation of water quenching process of an engine cylinder head, in: *ASME/JSME 2003 4th Joint Fluids Summer Engineering Conference*, American Society of Mechanical Engineers Digital Collection, 2003, pp. 1571–1578.
- [3] F. Lemmadi, A. Chala, S. Ferhati, F. Chabane, S. Benramache, Structural and mechanical behavior during quenching of 40crmov5 steel, *Journal of Science and Engineering* 3 (1) (2013) 1–6. [5](#)
- [4] C. Şimşir, C. H. Gür, A simulation of the quenching process for predicting temperature, microstructure and residual stresses., *Strojniski Vestnik/Journal of Mechanical Engineering* 56 (2). [2, 5](#)
- [5] S. Nukiyama, The maximum and minimum values of the heat q transmitted from metal to boiling water under atmospheric pressure, *International Journal of Heat and Mass Transfer* 9 (12) (1966) 1419–1433. [3](#)
- [6] V. Dhir, Boiling heat transfer, *Annual review of fluid mechanics* 30 (1) (1998) 365–401. [v, 4](#)
- [7] P. Cavaliere, E. Cerri, P. Leo, Effect of heat treatments on mechanical properties and fracture behavior of a thixocast a356 aluminum alloy, *Journal of Materials Science* 39 (5) (2004) 1653–1658. [5](#)
- [8] B. Taraba, S. Duehring, J. Španielka, Š. Hajdu, Effect of agitation work on heat transfer during cooling in oil isorapid 277hm, *Strojniški vestnik-Journal of Mechanical Engineering* 58 (2) (2012) 102–106. [5](#)
- [9] J. Olivier, B. Clement, J. Debie, F. Moreaux, Stirring of quenchants fluids: design considerations and metallurgical consequences, *Trait. Therm* 206 (1986) 29–42. [5](#)
- [10] N. Chen, L. Han, W. Zhang, X. Hao, Enhancing mechanical properties and avoiding cracks by simulation of quenching connecting rods, *Materials Letters* 61 (14-15) (2007) 3021–3024. [5](#)
- [11] A. Buczek, T. Telejko, Investigation of heat transfer coefficient during quenching in various cooling agents, *International Journal of Heat and Fluid Flow* 44 (2013) 358–364. [5](#)
- [12] S. A. Ebrahim, S. Chang, F.-B. Cheung, S. M. Bajorek, Parametric investigation of film boiling heat transfer on the quenching of vertical rods in water pool, *Applied Thermal Engineering* 140 (2018) 139–146. [5](#)

- [13] N. Kaneyasu, F. Yasunobu, U. Satoru, O. Haruhiko, Effect of surface configuration on nucleate boiling heat transfer, *International Journal of Heat and Mass Transfer* 27 (9) (1984) 1559–1571. [5](#)
- [14] P. Archambault, S. Denis, A. Azim, Inverse resolution of the heat-transfer equation with internal heat source: Application to the quenching of steels with phase transformations, *Journal of materials engineering and performance* 6 (2) (1997) 240–246. [5](#)
- [15] G. Ramesh, K. N. Prabhu, Assessment of axial and radial heat transfer during immersion quenching of inconel 600 probe, *Experimental Thermal and Fluid Science* 54 (2014) 158–170. [5](#)
- [16] D. S. Mackenzie, History of quenching, *International Heat Treatment and Surface Engineering* 2 (2) (2008) 68–73. [6](#)
- [17] V. Srinivasan, K.-M. Moon, D. Greif, D. M. Wang, M.-h. Kim, Numerical simulation of immersion quenching process of an engine cylinder head, *Applied Mathematical Modelling* 34 (8) (2010) 2111–2128. [6](#)
- [18] H. Ramezanzadeh, A. Ramiar, M. Yousefifard, Numerical investigation into coolant liquid velocity effect on forced convection quenching process, *Applied Thermal Engineering* 122 (2017) 253–267. [6](#)

Chapter 2

Eulerian conservative and adaptive framework

Contents

2.1 Introduction	18
2.2 Anisotropic mesh adaptation	19
2.2.1 Edge-based error estimation	19
2.2.2 Gradient recovery procedure	20
2.2.3 Metric construction	21
2.2.4 Mesh adaptation criterion	21
2.3 Conservative interpolation of fields between meshes	23
2.3.1 Linear interpolation	24
2.3.2 Conservation of physical quantities	26
2.4 Parallelization of the conservative adaptive interpolation algorithm	28
2.4.1 Parallel mesh adaptation algorithm	28
2.4.2 Parallel implementation the conservative interpolation algorithm	30
2.5 Numerical examples	32
2.5.1 2D Analytical functions	32
2.5.2 3D Analytical test case	35
2.5.3 2D Analytical test case with mesh adaptation	35
2.5.4 2D Lid driven cavity : interpolation and dynamic mesh adaptation	36
2.5.5 3D Lid driven cavity : interpolation and dynamic mesh adaptation	40
2.5.6 Unsteady flow past a 3D vehicle model	41
2.5.7 Scalability study on the 3D Lid driven cavity	43
2.6 Conclusion	44
2.7 Résumé du chapitre en français	45
2.8 Bibliography	46

2.1 Introduction

In recent years, there have been increasing interest in studying numerically a variety of engineering applications that involve thermal coupling of fluids and solids. However, the modeling and simulation of fluid-solid interactions (FSI) of multiphase applications are considered challenging since they involve the study of the interactions between a solid body and its surrounding media or the interactions between different fluid flows. In this thesis, we are interested in the numerical simulation of thermal coupling problems involving multiphase flows. The most common trend to deal with multi-component domains is issued from the body fitted approach whereby the global domain is partitioned into several local subdomains over each of which a local model is solved [1]. However, during the assembly, the coordination between the meshes can become complicated or even sometimes infeasible.

Immersed methods can be considered as a good tool for reducing the computational cost induced by body-fitted approaches and overcoming the interface treatment step required by non-body fitted techniques. They simplify a number of issues in multiphase applications such as meshing the fluid domain, the use of a fully eulerian algorithm, dealing with complex geometries in terms of curvatures, sharp angles and singularities and topological changes. We can find in the literature three classes of methods for immersing geometries into a computational domain. The first group gathers the methods that enrich the numerical solution's space in a finite element framework [2]. These methods avoid the computational cost needed to construct a mesh that is well representative of the immersed bodies but requires the modification of the finite element solvers.

The second accounts for the effect of the immersed objects by introducing an additional source term into the governing equations, [3–5]. However, when dealing with multiphysical domains, these methods can rapidly become very expensive and time-consuming.

The interface tracking methods constitute a third class of immersion; they propose a different scenario than the previously mentioned techniques [6]. The main idea is to retain the use of the monolithic formulation and coupling it to some additional features that could allow a better and accurate resolution, in particular at the interface between the fluid and solid. Recall that the monolithic resolution is based on the level set approach and consists in considering a single mesh for all components; fluid and solid domains. The interfaces between the different multiphysical domains are tracked by signed distance functions called level set. The immersed volume method (IVM) [6–10] was defined in that sense and can be applied on a wide range of multimaterial applications. The three main ingredients of the immersed volume methods are : (i) the level set, a signed distance function that is used to localize and immerse the solids inside the global domain, (ii) then an anisotropic mesh adaptation is employed to provide a good capture of the interfaces between the solid and fluid parts, (iii) and finally mixing the different material properties using a homogeneous distribution along the interface. The IVM method can be applied without modification for any geometry and any physical property. It can be very easily implemented and applied with stabilized finite element methods and in the context of moving structure problems. The level set function as well as the mixing laws will be presented in the next chapter. This chapter is only dedicated to the newly developed anisotropic mesh adaptation technique coupled with a parallel conservative interpolation method.

2.2 Anisotropic mesh adaptation

In order to provide an accurate configuration of the physical problem, immersed objects should be properly defined and material properties should be well distributed. For accurate computations in a multiphase framework, the mesh has to be refined around the interface. Indeed, adapting the mesh to the physical behaviors and studied phenomena is a means to improve the accuracy of numerical results. However, difficulties might arise due to the discontinuities in the material properties between the solid and the fluid parts. In fact, when the discontinuities intersect the mesh elements arbitrarily, they will not be properly captured and hence they might result in the deterioration of the accuracy of the solution. The objective is therefore to combine the level set representation with local mesh refinement around the zero isovalue of the level set function yielding an accurate capture of a fluid-solid interface at a low computational cost. The key to these regards is anisotropic mesh adaptation that generates highly stretched and well oriented elements allowing a good capture of sharp gradients. It implies that we can stretch the elements in certain directions according to the solution features. Several approaches to build anisotropic adaptive meshes often based on local modifications on an existing mesh can be found in the literature [11–13]. They mainly require extending the way to measure lengths following the space directions and can be done using a metric field to redefine the geometric distances. Mesh adaptation techniques based on a posteriori error estimation have also been well developed [14], leading to some standardization of the adaptation process. In this thesis, the mesh adaptation technique is based on the work [15; 16], we focus on anisotropic mesh adaptation driven by directional error estimators. The mesh is dynamically adapted to the solution, i.e. velocity and interface position. The aim is to refine the discretization in the areas of the domain where the solution fields are mostly non-linear. The refinement affects both the density of elements and their shape, which is anisotropically adapted and stretched along the direction where the considered solution field is linear. The mesh becomes locally refined around the zero isovalue of the level set function which enables to sharply define the interface and to save a great number of elements compared to classical isotropic refinement. To do so, we start by performing an error analysis on the mesh. Then, to correlate the error with the geometry, a metric field is defined. From this metric field, an anisotropic error indicator is defined and used as a functional for a re-meshing optimization problem.

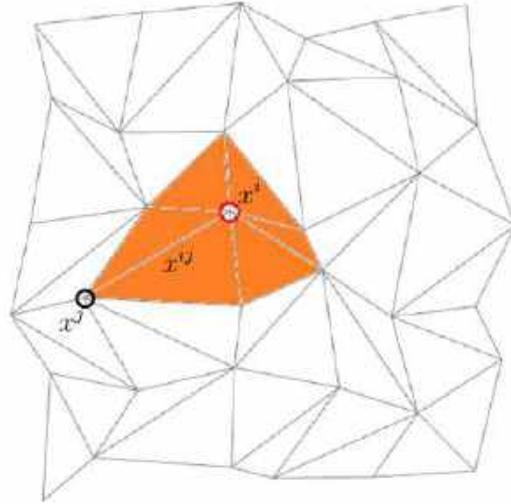
2.2.1 Edge-based error estimation

Let u_h be a P1 finite element approximation obtained by applying the Lagrange interpolation operator to a regular function $u \in C^2(\Omega)$. At each vertex i of the mesh, we have $U^i = u(x^i) = u_h(x^i)$ (where x^i are the coordinates of the vertex i). Let $\Gamma(i)$ be the "patch" associated to a vertex x^i of the mesh defined as the set of nodes which share one edge with x^i , and let us denote by x^{ij} the edge connecting x^i to x^j as seen in Figure 2.1. The gradient $\nabla u_h \cdot x^{ij}$ on the edge x_{ij} is continuous, therefore we can write :

$$U^j = U^i + \nabla u_h \cdot x^{ij} \quad (2.1)$$

which leads to :

$$\nabla u_h \cdot x^{ij} = U^j - U^i \quad (2.2)$$


 Figure 2.1: Patch $\Gamma(i)$ associated with node x^i

Following the work of [16], an a posteriori error estimate based on the length distribution tensor approach and the associated edge based error analysis is defined :

$$\| \nabla u^h \cdot x^{ij} - \nabla u(x^i) \cdot x^{ij} \| \leq \max_{y \in |x^i, x^j|} | x^{ij} \cdot H_u(y) \cdot x^{ij} | \quad (2.3)$$

with H_u the hessian of u . Since we want to compute the recovered gradient g^i of u_h at the node x^i , thus we have :

$$\nabla g_h \cdot x^{ij} = g^j - g^i \quad (2.4)$$

The projection of the Hessian based on the gradient at the extremities of the edge is obtained as follows :

$$(\nabla g_h \cdot x^{ij}) \cdot x^{ij} = (g^j - g^i) \cdot x^{ij} \quad (2.5)$$

$$(H_u \cdot x^{ij}) \cdot x^{ij} = g^{ij} \cdot x^{ij} \quad (2.6)$$

with $g^{ij} = g^j - g^i$. We denote the error along the edges using the following expression :

$$e^{ij} = | g^{ij} \cdot x^{ij} | \quad (2.7)$$

This error sampling is the exact interpolation error along the edge and enables us to evaluate the global L_2 error. The equation (2.7) can be evaluated only when the gradient of u is known and continuous at the nodes of the mesh, thus a recovery procedure has to be considered.

2.2.2 Gradient recovery procedure

The recovery gradient operator is defined by a local optimization problem :

$$G^i = \arg \min_G \left(\sum_{j \in \Gamma(i)} | (G - \nabla u_h) \cdot x^{ij} |^2 \right) \quad (2.8)$$

Denoting by \otimes , the tensor product between two vectors, let us introduce X^i the length distribution tensor at node i :

$$X^i = \frac{1}{|\Gamma(i)|} \left(\sum_{j \in \Gamma(i)} x^{ij} \otimes x^{ij} \right) \quad (2.9)$$

whose purpose is to give an average representation of the distribution of edges in the patch. Let us express the recovered gradient G^i in terms of the length distribution tensor :

$$G^i = (X^i)^{-1} \sum_{j \in \Gamma(i)} U^{ij} x^{ij} \quad (2.10)$$

Therefore, the estimated error is thus written as :

$$e_{ij} = G^{ij} . x^{ij} \quad (2.11)$$

Now, we seek to relate this error indicator e_{ij} to a metric suitable for a mesh adaptation procedure.

2.2.3 Metric construction

It is necessary to take into account the neighborhood of the node so that the best averaging representation is a metric defined at each node [15]. The metric can be regarded as a tensor whose eigenvalues are related to the mesh sizes, and whose eigenvectors define the directions for which these sizes are applied. The metric M is a symmetric positive definite tensor representing a local base that modifies the distance computation from the Euclidean space to the metric space, therefore it takes the following expression :

$$\tilde{M}^i = (\tilde{X}^i)^{-1} \quad (2.12)$$

where

$$\tilde{X}^i = \frac{1}{|\Gamma(i)|} \left(\sum_{j \in \Gamma(i)} s^{ij} \otimes s^{ij} \right) \quad (2.13)$$

The stretching factor s^{ij} of the edge ij is chosen so that the total number of nodes in the mesh is kept fixed. It is defined as the ratio between the original edge length and the length of the adapted edge, more details can be found in the work of jannoun2015anisotropic.

2.2.4 Mesh adaptation criterion

When simulating complex physical phenomena involving turbulent flows and heat transfers like the quenching process, it is highly desirable to accurately capture all the characteristics of the problem, including but not limited to, the flow field, the temperature variations, and the fluid/solid interfaces. Therefore, different fields can be used as a criterion for mesh adaptation: the adaptivity will account for the both the change of direction of the velocity, its magnitude, the level set function and the temperature field.

The common technique to derive a single metric at each node in the mesh accounting for several fields of interest relies on the rigor of computing the metrics corresponding to each of the sensor fields then performing a metric intersection operation [17]. In this work, we simplify this operation and we use one metric that accounts for different variables. Therefore, based on the theory proposed in the previous section, it is possible to extend the definition to account for several sources of error. Thus, the following vector of sources of error is defined:

$$v(x^i) = \left\{ \frac{V^i}{|V^i|}, \frac{|V^i|}{\max_j |V^j|}, \frac{T^i}{T_{max}}, \frac{\phi}{\max(\phi)} \right\} \quad (2.14)$$

Because all fields are normalized (the velocity components v_x, v_y and v_z by the local velocity norm, the velocity norm and the level set function by their respective global maximum), a field that is much larger in absolute value does not dominate the error estimator, and the variations of all variables are fairly taken into account.

In the framework of the immersed volume method, the goal is to provide a good representation of the fluid/solid interfaces for a fixed number of nodes in the mesh. Let us consider the example of two immersed solid bodies. Starting from an isotropic mesh, we generate the multidomain metric and we adapt the mesh to this metric. After several iterations, one can clearly see from Figure 2.2 that the interface between these objects and the surroundings is well adapted. Figure 2.3 clearly shows that the refinement at the interface

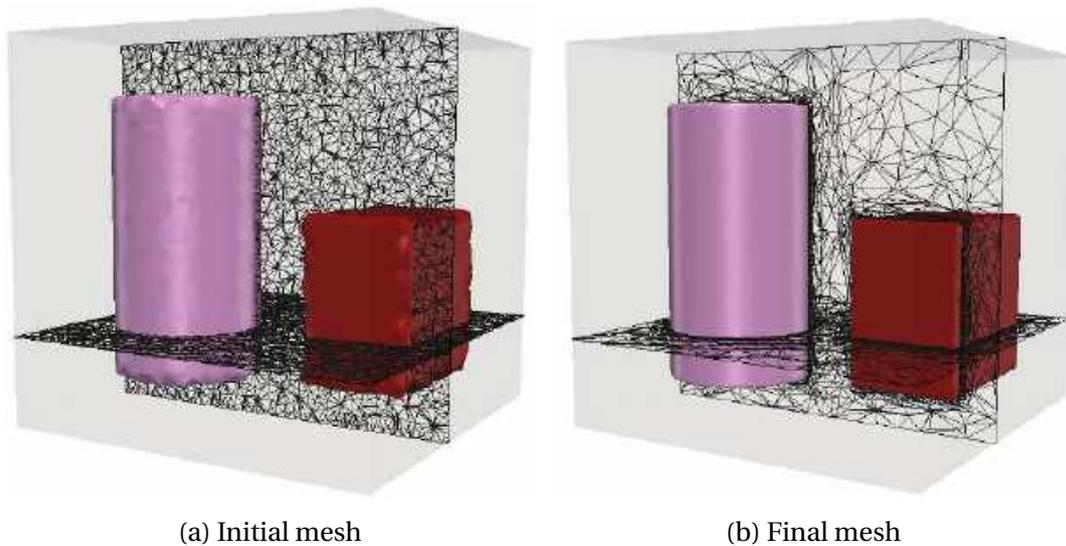


Figure 2.2: Mesh adaptation algorithm applied to two immersed bodies

is anisotropic. The nodes are highly condensed in the vicinity of the interface favoring its accurate capture. This validates how the developed algorithm optimizes the distribution of the nodes to produce a sharp anisotropic mesh that is well adapted based on a given variable (here, the level set function of the immersed solids). Note also, when using an anisotropic mesh, with elements stretched in a 'right' direction, one could allow not only to save a lot of elements but also to well describe the geometry in terms of curvature, angles, etc. An important point to mention here is that this mesh adaptation technique works under the constraint of a fixed number of nodes. With such an advantage, we avoid a drastic increase of mesh complexity and hence computational cost when dealing with complex CFD problems. However, one serious drawback of many mesh adaptivity algorithms on unstructured meshes the necessity of interpolating solutions fields from the initial mesh to the newly adapted mesh. Such interpolation may destroy conservation of important physical quantities and leads to errors in the solution fields. Thus, in the next section, we will give a brief review of the several conservative interpolation methods that can be found in the literature and give details on the conservative interpolation algorithm implemented in this work.

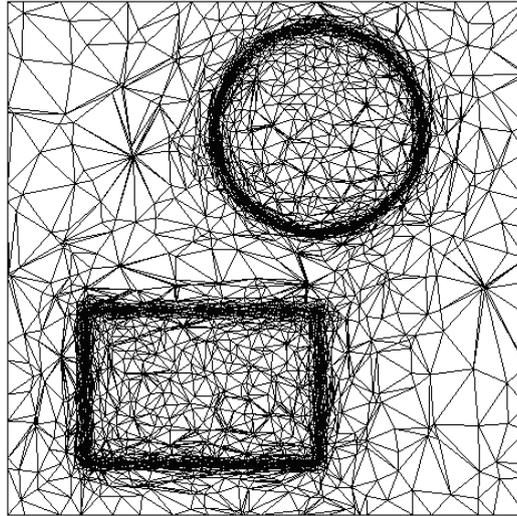


Figure 2.3: Final mesh adapted to the different interfaces

2.3 Conservative interpolation of fields between meshes

During numerical simulations and especially in the context of anisotropic mesh adaptation, with highly stretched elements, the interpolation step is crucial. However, a non-conservative approach increases dissipation and may grow with every operation, thus spoiling the conservation of important physical quantities. Numerous works have addressed such challenge. Indeed, for coupled problems such as fluid structure interactions, the conservation of the load along the interface using a node projection scheme was proposed in [18], while others recommended the use of an auxiliary mesh [19]. Building an intermediate mesh composed of the intersection of the fluid and solid meshes along the interface was introduced in [20; 21] as the term common refinement, combined with the use of a Galerkin projection with an L_2 minimization, that lead to errors which grow more slowly with iteration number compared to other interpolation methods. Another similar approach proposed in [22; 23] is to build a supermesh as the union of two unrelated unstructured meshes in order to use interpolation operators that preserve the global integrals of the solution fields; along with a standard Galerkin projection, a minimally diffusive bounded interpolation scheme is recommended. Others have focused on high order interpolation on quadrilateral elements, [24; 25]. While another approach proposed in [26] proceeds by mapping from a donor to target mesh by calculating the volume of intersection of overlapping polyhedra on the donor and target meshes. This method is similar to performing a Galerkin projection [27] except the donor field is constant across donor elements. In [28; 29] an operator that respects the mass conservation and fulfills the maximum principle by reconstructing the mass field and its gradient with the elemental intersection between both donor and target meshes is proposed. Moreover, the need for accurate transmission of information between meshes has led to the development of mesh independent interpolation methods that conserve global quantities. Indeed, in the work of [30], adding a corrector term to the interpolated solution to ensure mass conservation is discussed. An adaptive remeshing technique based on the diffusive approximation that preserves local equilibrium and ensures conservation of dissipated and strain energy is introduced in [31]. In [32], the conservative interpolation of fluxes between overlapping polyhedra is ensured by applying constraints to integration weights and the interpolations coefficients. In the field of domain decomposition methods, the

conservation of global quantities by resolving an optimization problem is introduced in [33; 34]. The previous method was generalized in our work [35] for problems that require interpolation when using mesh adaptation. The main idea of the method is to impose the continuity of the solution via classical linear interpolation and to relax it by the conservation of global physical quantities. Once a new mesh is obtained with mesh adaptation, the solution fields need to be interpolated from the old mesh \mathcal{T}_1 to the new mesh \mathcal{T}_2 in order to pursue the computations. For each stage, the first mesh is the one where the initial solution u_1^1 is computed. Then, we use it to adapt and obtain a new mesh. We note the interpolated solution u_2^1 . Adapting the mesh, will allow minimizing the interpolation error as explained above. Then, we correct the interpolated solution with an optimization problem based on a given restriction. We note the new corrected solution \tilde{u}_2^1 . In the context of dynamic mesh adaptation, this procedure is repeated till convergence is reached as illustrated in Figure 2.4.

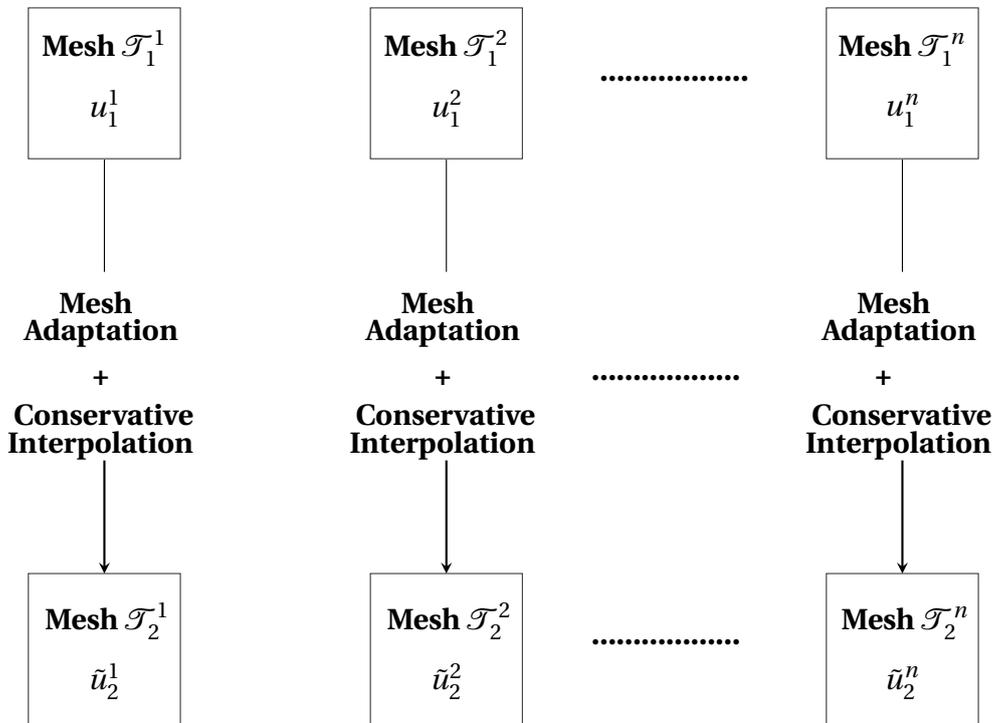


Figure 2.4: Mesh adaptation and fields' interpolation between meshes

2.3.1 Linear interpolation

While resolving a numerical problem, we can distinguish two types of fields (details are shown in Figure 2.5) :

- P_1 variables, defined on the nodes of the mesh (velocity, pressure, temperature)
- P_0 variables, defined on the elements of the mesh (stress and strain tensor)

The transfer of the discrete P_1 fields defined at the nodes level is done in two steps :

1. A localization step, whereby a method of bounding boxes is used to identify the element $K_1 = x_i^K, i = 1, \dots, d + 1$ in the old mesh that contains the new node p_2 . For this work, tree search algorithms are used, [36]. The idea is to check the elements that

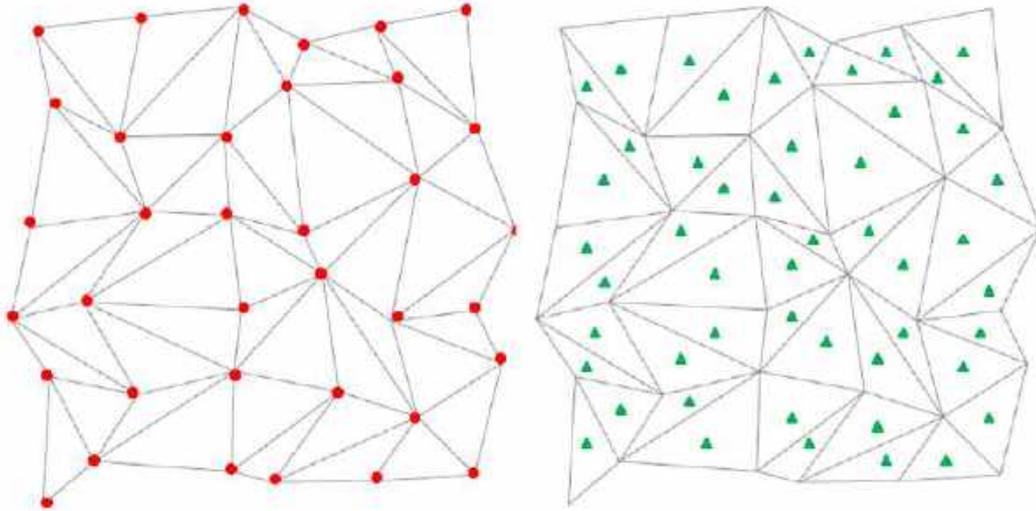


Figure 2.5: Left: P_1 nodal fields - Right: P_0 fields defined at Gauss points

cover the immediate neighborhood of the point to be interpolated by building cubic boxes around the selected node (see Figure 2.6). The new vertex is located in the smallest cubic box that can contain it. In this work, we use an octree algorithm and also R-tree search, [37]. The structure is designed so that a spatial search requires visiting only a small number of nodes. We group nearby elements and represent them with their minimum bounding rectangle.

2. Then, a linear interpolation of the p_1 variable field from the nodes of the element K_1 to p_2 using its barycentric coordinates β_i with respect to K_1 is done :

$$u_2(p_2) = \sum_{i=1}^{d+1} \beta_i(p_2) u_1(x_i^{K_1})$$

This interpolation scheme is P_1 exact, of second order and independent of the mesh topology.

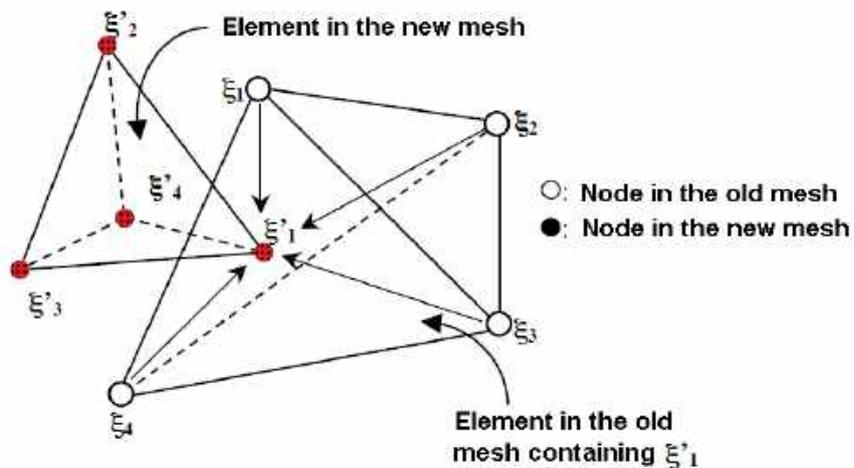


Figure 2.6: P_1 interpolation from an old mesh to a new one, adopted from [38]

As for the **transfer of the of P_0 fields** , two possibilities can be found. On one hand, one can extrapolate the fields from element level onto the nodes then apply a P_1 transfer to the

Gauss points of the elements of the new mesh. On the other hand, one can apply a direct transfer whereby the Gauss points of the new elements are localized inside the elements of the old mesh and take their values. The latter technique showed that it makes rough choices especially for coarse elements whereas the former one induces diffusion due to the extrapolation step. However the projection technique suits better our mesher since the latter proceeds by local modification of topologies, i.e. if a certain region of the mesh is not changed, the data will not be transferred, hence avoiding numerical diffusion.

2.3.2 Conservation of physical quantities

Now that we have computed the interpolated solution $u_{h,2}$ on the new mesh, the idea is to obtain a new conservative solution called $\tilde{u}_{h,2}$ by applying a correction to the interpolated solution $u_{h,2}$. To do so, in an independent subroutine, an optimization problem with Lagrange multipliers is solved. We apply a series of constraints to the interpolated field $u_{h,2}$ through the Lagrange multipliers. Thus, $\tilde{u}_{h,2}$ must satisfy the following conditions :

- $\tilde{u}_{h,2}$ must be the nearest to $u_{h,2}$ in the L_2 -norm
- $u_{h,2}$ must respect some physical properties of $u_{h,1}$, solution on the donor mesh

One of the main issues in the numerical computations of incompressible flows is the conservation of mass, thus, we will formulate a problem with linear restrictions. We are interested in ensuring conservation of linear momentum and mass, to do so we must apply the following constraints :

$$\left\{ \begin{array}{l} \text{Minimize :} \\ \text{Under the constraints :} \end{array} \right. \quad \begin{array}{l} \int_{\Omega_2} |\tilde{u}_{h,2} - u_{h,2}|^2 \\ \int_{\Omega_2} \tilde{u}_{h,2} = \int_{\Omega_1} u_{h,1} \\ \int_{\Omega_2} \nabla \cdot \tilde{u}_{h,2} = \int_{\Omega_1} \nabla \cdot u_{h,1} \end{array} \quad (2.15)$$

The nodal fields are expressed in terms of the shape functions of each finite element partitions :

$$u_{h,n} = \sum_i N_n^i U_n^i, \quad n = 1, 2$$

Let $(\lambda_1, \lambda_2) \in \mathcal{R}$, be the Lagrange multipliers to enforce respectively the restrictions in 2.15. Given $u_{h,2} \in \mathcal{V}_{h,2}$ the interpolation of $u_{h,1} \in \mathcal{V}_{h,1}$ on $\mathcal{T}_{h,2}$. Let us define by $\mathcal{L} : \mathcal{V}_{h,2} \times \mathcal{R}^2 \rightarrow \mathcal{R}$ the following functional :

$$\begin{aligned} \mathcal{L}(\tilde{U}_2, \lambda) = & \frac{1}{2} \int_{\Omega_2} \left(\sum_{n \in \mathcal{T}_2} N_2^n (\tilde{U}_2^n - U_2^n) \right)^2 - \sum_{j=1}^d \lambda_{1+j} \int_{\Omega_2} \left(\sum_{n \in \mathcal{T}_2} N_2^n \tilde{U}_{2,j}^n - \sum_{n \in \mathcal{T}_1} N_1^n \tilde{U}_{1,j}^n \right) \\ & - \lambda_2 \int_{\Omega_2} \left(\sum_{n \in \mathcal{T}_2} \sum_{j=1}^d \partial_j N_2^n \tilde{U}_{2,j}^n - \sum_{n \in \mathcal{T}_1} \sum_{j=1}^d \partial_j N_1^n \tilde{U}_{1,j}^n \right) \end{aligned} \quad (2.16)$$

Taking the derivatives of \mathcal{L} with respect to the nodal unknowns and the Lagrange multipliers, we have :

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \lambda_{1+j}} = 0 &\longrightarrow \int_{\Omega_2} \sum_{n \in \mathcal{T}_2} N_2^n \tilde{U}_{2,j}^n = \int_{\Omega_1} \sum_{n \in \mathcal{T}_1} N_1^n U_{1,j}^n \quad j = 1, \dots, d \\ \frac{\partial \mathcal{L}}{\partial \lambda_2} = 0 &\longrightarrow \int_{\Omega_2} \sum_{n \in \mathcal{T}_2} \sum_{j=1}^d \partial_j N_2^j \tilde{U}_{2,j}^n = \int_{\Omega_1} \sum_{n \in \mathcal{T}_1} \sum_{j=1}^d \partial_j N_1^j U_{1,j}^n \\ \frac{\partial \mathcal{L}}{\partial \tilde{U}_2^n} = 0 &\longrightarrow \sum_{n \in \mathcal{T}_2} \int_{\Omega_2} N_{2,i}^n N_{2,k}^n \tilde{U}_{2,j}^n - \lambda_{1+j} \int_{\Omega_2} N_2^n - \lambda_2 \int_{\Omega_2} \partial_j N_2^n = \sum_{n \in \mathcal{T}_2} \int_{\Omega_2} N_{2,i}^n N_{2,k}^n U_{2,j}^n \end{aligned} \quad (2.17)$$

Let us define by $R_{n,1}$ and $R_{n,2}$ the matrices that come from the imposition of linear momentum and mass on the meshes $n = 1, 2$:

$$R_{1,1} = \int_{\Omega_1} N_1^n \quad (2.18)$$

$$R_{2,1} = \int_{\Omega_1} N_2^n \quad (2.19)$$

$$R_{1,2} = \int_{\Omega_1} \partial_j N_1^n \quad j = 1, \dots, d \quad (2.20)$$

$$R_{2,2} = \int_{\Omega_1} \partial_j N_2^n \quad j = 1, \dots, d \quad (2.21)$$

The system that needs to be solved at each iteration is the following (the reader refers to [34] for more details) :

$$\begin{bmatrix} M_2 & -R_{2,1} & -R_{2,2} \\ R_{2,1}^t & 0 & 0 \\ R_{2,2}^t & 0 & 0 \end{bmatrix} \begin{bmatrix} \tilde{U}_2 \\ \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} M_2 \tilde{U}_2 \\ R_{1,1}^t U_1 \\ R_{1,2}^t U_1 \end{bmatrix} \quad (2.22)$$

with $M_2 = \int_{\Omega_2} N_{2,i}^n N_{2,k}^n$ the mass matrix associated with the target mesh, $R_{1,1}^t U_1$ and $R_{1,2}^t U_2$ are scalars that represent the global quantities one wishes to conserve (here, linear momentum and mass). Instead of solving the system using iterative methods, which is very costly, the mass matrix M_2 is approximated by a diagonal matrix which reduces considerably the computational cost.

This system has been implemented in our parallel library Cimlib-CFD. In our finite element code, we use the Portable Extensible Toolkit for Scientific computation (PETSc) library to store and solve large systems in parallel. PETSc, [39] is an open-source suite of data structures and routines to solve scientific applications modeled by PDEs. To solve the system our non-symmetric problem, the Generalized Minimal RESidual (GMRES) method introduced by Saad and al. in [40] has proven to be more suitable.

Now, before moving to the parallel implementation of the proposed metric-based conservative adaptation method, let us recall the steps of the conservative mesh adaptation in the following algorithm.

Algorithm 1 Conservative mesh adaptation algorithm on a given field u_1

Input : A mesh \mathcal{T}_1 , a target number of nodes N and a field u_1 defined on the nodes of the mesh

Output : An optimal anisotropic mesh \mathcal{T}_2 well adapted to accurately capture the variation of u_1

while not converged **do**

for each node x^i in \mathcal{T}_1 **do**

 Compute the recovery gradient G^i

 Determine the error estimate e_{ij} on the edges x^{ij} connected to x^i

 Compute the stretching factors s_{ij} on the edges x^{ij}

 Construct the metric tensor M^i

end for

 Generate the optimal mesh \mathcal{T}_2

 Interpolate u_1 from \mathcal{T}_1 to \mathcal{T}_2

 Apply the conservation of physical quantities to the interpolated field u_2

end while

2.4 Parallelization of the conservative adaptive interpolation algorithm

Computational fluid dynamics (CFD) simulations aim at solving more complex, more real, more detailed and bigger problems. To achieve this, they must rely on high-performance computing (HPC) systems as it is the only environment where these kinds of simulations can be performed. Especially in the context of industrial applications, computational time is a major issue. In this section, we will start by briefly presenting the algorithm behind parallel mesh adaptation in our code Cimlib-CFD. Then give details on the parallel implementation of the conservative interpolation algorithm.

2.4.1 Parallel mesh adaptation algorithm

Mesh adaptation is one of the key technologies to reduce both the computational cost and the approximation errors of PDE-based numerical simulations. The parallelization of mesh adaptation methods goes back to the end of the 20th century, [41–43]. In our library, Message passing interface (MPI) is used for efficient parallel communications between processors. A schematic representation of the algorithm behind parallel mesh adaptation adopted in our library Cimlib-CFD is presented in Figure 2.7.

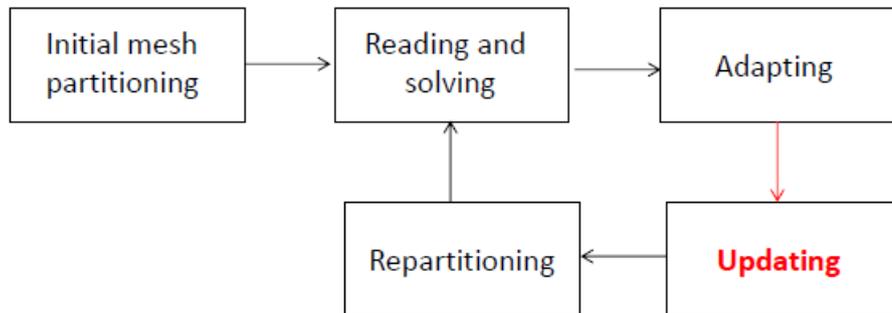


Figure 2.7: Conservative mesh adaptation procedure in parallel

Initial mesh partitioning: At the beginning of the simulation, the initial mesh is partitioned into multiple subdomains over the allocated resources and an initial load balancing is done based on the mesh topology and on the geometry of the domain. The load balancing consists in distributing efficiently and uniformly the workloads uniformly across all the processors.

Reading and solving: After, the subdomains are read by the program and different operations are done within each partition. These operations include the building and the solving of linear systems for the finite element code. For instance, solving the Navier-Stokes VMS system (see Chapter 3) or the thermal convection-diffusion-reaction problem (see Chapter 4) but, more importantly in the case of anisotropic mesh adaption, the computation of the local error estimates.

Adapting: Then, we adapt the mesh with respect to a solution field. Using the method presented in the previous section, we compute a local error estimator and to correlate the error to the mesh geometry using a metric space in order to generate a new adapted mesh. An error estimate is computed for each subdomain. According to the derived error estimator and under the constraint of a given number of elements in the new adapted mesh, an optimal mesh is generated. The constraint could be considered as local to each subdomain. In this case, all computations are local and no data need to be exchanged between processors. If we consider a local constraint on the number of elements, each processor will have the numbers of elements and thus no heavy load balancing is needed after each mesh adaptation. One drawback to mention here is that in some subdomains, no heavy computations are needed and thus there is no need to have a high mesh density. To overcome this problem, we consider a global constraint for the number of elements. The global number of elements over the entire domain is distributed with respect to the mesh density prescribed by the error estimator. And in this case, reload balancing is needed after each mesh adaptation stage. The parallel behavior of the mesh adaptation is very close to the serial one and the error analysis is still the same.

Updating: As explained previously, the solution computed on the previous mesh then needs to be updated on the new mesh. At this step, we use interpolation methods to transport the variables from the previous mesh to the new one. In the context of anisotropic mesh adaptation, the CPU times devoted to the interpolation stage is very small (less than one percent) relative to the total adaptive CPU time. When using the conservative interpolation algorithm, we must solve a linear system in parallel. The parallel implementation of the algorithm will be explained in the next section.

Repartitioning: Finally, the new mesh is repartitioned over the allocated CPUs to take into account the changes of mesh topology in the computational loads distribution. The remeshing operations are performed using a sequential mesh adaptator on each subdomain with an extra treatment of the interfaces. At the first iteration, remeshing is performed concurrently on each processor while the interfaces between subdomains are locked to avoid any modification in the sequential remeshing kernel. Then, a new partitioning is calculated to move the interfaces and remesh them at the next iteration. The algorithm iterates until all items have been remeshed.

2.4.2 Parallel implementation the conservative interpolation algorithm

In order to assemble the system (2.22) in parallel, a loop over the elements of the mesh is needed. In our library Cimlib-CFD, the parallelization of such assembly process is done using distributed memory parallelism, based on MPI. The values of the matrix are com-

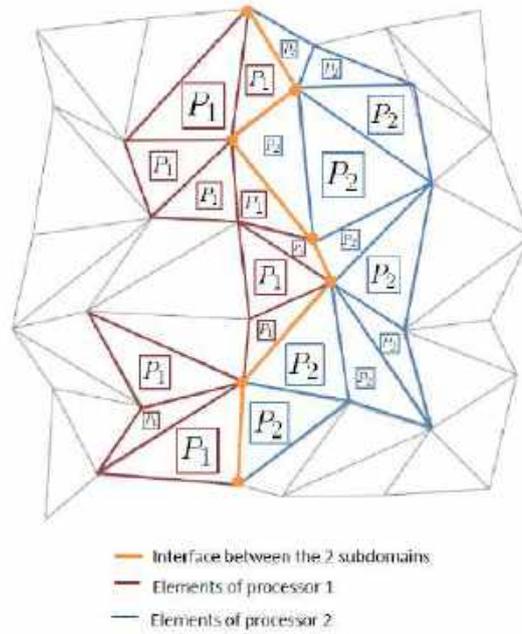


Figure 2.8: Interface between two subdomains, and respective connected cells

puted on each cell of the mesh from the values of the neighboring edges or vertices. Here in this work, we assign rows exclusively to one MPI process. And in order to obtain the complete coefficients of the rows of interface nodes and to maintain the correctness of the numerical results, the cells located at the frontiers between subdomains need to know the values of their neighbors located in another subdomain. For instance, if we consider a domain partitioned between two processors (see Figure 2.8), in order to compute the values at the interface node, we must know the contribution of all the neighboring cells. Therefore, the slice of cells at the frontiers, called ghost cells or halos, are duplicated among the neighbor subdomains so that they can add their contributions, see Figure 2.9. Then, these halos are exchanged at each time step to reduce the different contributions and obtain the final correct result, see Figure 2.10. A halo element is necessary to fully assemble the row of node 3 in subdomain 1 and the row of node 4 in subdomain 2. This process consists in packing all required data into the communication buffers, sending to all neighbors the corresponding interface, and finally receiving and unpacking the data from all neighbors. The system is then assembled with PETSc [39] and using a parallel AIJ Sparse Matrix using the following command :

- `MatCreateAIJ = (MPI Comm, PetscInt m, PetscInt n, PetscInt M, PetscInt N, PetscInt dnz, PetscInt dnnz, PetscInt onz, PetscInt onnz, Mat A);`

A is the newly created matrix, while the arguments **m**, **M**, and **N**, indicate the number of local rows and the number of global rows and columns, respectively. In the PETSc partitioning scheme, all the matrix columns are local and **n** is the number of columns corresponding to local part of a parallel vector. The matrix is stored with a fixed number of rows on each process, given by **m**. Since preallocation of memory is critical for achieving good performance during matrix assembly, the parameters user **dnz**, **onz**, **dnnz**, and

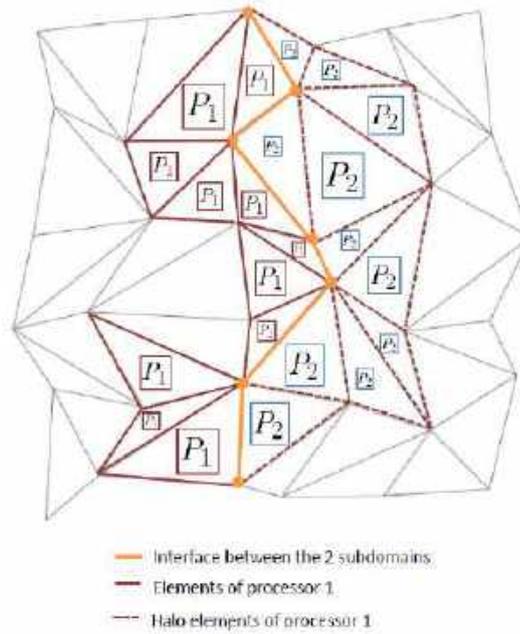


Figure 2.9: Cell nomenclature in the subdomain processor 1

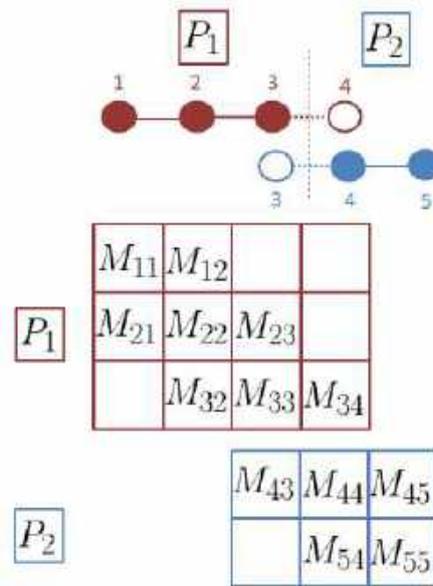


Figure 2.10: Assembly of the system in parallel

onnz are set to control the dynamic allocation of the matrix memory space. An important to mention here, is that the nodes are numbered locally on each MPI processor and the matrices are assembled in a distributed way, the parallel solvers in PETSc need the global row and global column indices of the entries in the distributed matrix. Thus, a 'Local to Global renumbering' is necessary in order to assign a global number for each node of the mesh.

The conservative adaptive interpolation algorithm has been parallelized and applied to several 2D and 3D cases. It has been implemented in the C++ CimLib-CFD finite element library dedicated for parallel mesh adaptation and for Fluid-Structure Interaction problems. All the following examples have been executed on a 64-bits DELL with a cluster

Intel that consists of 2000 heterogeneous cores interconnected with infiniband network. It is a cluster with a linux system with two Intel Xeon generations E5-2670 and E5-2680 chips. The jobs are scheduled using the OAR environment presented by Capit and al. in [44]. OAR is a versatile resource and task manager, also called a batch scheduler. It allows managing computational resources on a multi-users and multi-tasks system. In this environment, the user can: (i) choose the computational resources needed, (ii) plan his jobs by choosing starting dates and wall-times, (iii) reserve computational resources in order to have exclusivity on them for a certain amount of time. The policy lab allows the user to choose a maximum wall time of 3000 hours (i.e. 125 days, i.e. 4 months) which is highly sufficient for the computations performed in this thesis.

2.5 Numerical examples

In this section we present six test cases with validations to test the performance of the proposed conservative eulerian framework. The first one focuses on 2D analytical functions with comparison between linear and conservative interpolation. The second test case is a 3D analytical function, with emphasize on the behavior of each interpolation operator (linear and conservative) after several interpolation steps. The next test case, focuses on a 2D analytical function with dynamic mesh adaptation. Then we present the time-dependent test case of the lid driven cavity in 2D and 3D using different Reynolds numbers, the obtained results are compared with precise results found in the literature. In the last case, we test the conservative adaptation-interpolation method on a more challenging 3D problem with turbulent flow using immersed methods.

2.5.1 2D Analytical functions

To test the effectiveness of the conservative eulerian framework, we propose in this section several 2D test cases. We compare the solutions of the linear interpolation and the implemented interpolation with restrictions using several meshes with different number of elements. Following the work of [28], four analytical functions are first studied. These functions are representative of several physical phenomena encountered in computational fluid dynamics. The first function in Figure 2.11a is a Gaussian defined by :

$$f_1(x, y) = \exp(-30(x^2 + y^2)) \quad (2.23)$$

It is a smooth function that is representative of the vortices that are encountered in CFD. The second one in Figure 2.11b is a smooth sinusoidal shock :

$$f_2(x, y) = \tanh(100(y + 0.3\sin(-2x))) \quad (2.24)$$

It represents a continuous model of a shock which can be assimilated to the numerical capture of a shock with a dissipative flow solver. In this case, two constant domains are separated by a small region where a gradient variation happens continuously. Then, we studied a multiscale smooth function in Figure 2.11c :

$$f_3(x, y) = \begin{cases} 0.01\sin(50xy) & \text{if } xy \leq -\pi/50 \\ \sin(50xy) & \text{if } -\pi/50 < xy \leq 2\pi/50 \\ 0.01\sin(50xy) & \text{if } 2\pi/50 < xy \end{cases} \quad (2.25)$$

With this function, the smooth sinusoidal variations are at different scales, which is representative when dealing with multiscale problems. There are two order of magnitude

between small and large scales variations. Finally, to test the ability of the algorithm to deal with discontinuous fields, we studied a discontinuous function (see Figure 2.11d):

$$f_4(x, y) = \begin{cases} 1 & \text{if } x \geq 0 \text{ and } y \geq 0 \\ 2 & \text{if } x \geq 0 \text{ and } y < 0 \\ 3 & \text{if } x < 0 \text{ and } y \geq 0 \\ 4 & \text{if } x < 0 \text{ and } y < 0 \end{cases} \quad (2.26)$$

The solution here is constant in four squared regions and is discontinuous in the interface between each. All the functions are defined in a 2D domain $[-1, 1]^2$ and are computed on the nodes of the mesh. We consider a series of embedded meshes denoted by $(\mathcal{T}_1^i)_{i=1\dots 5}$

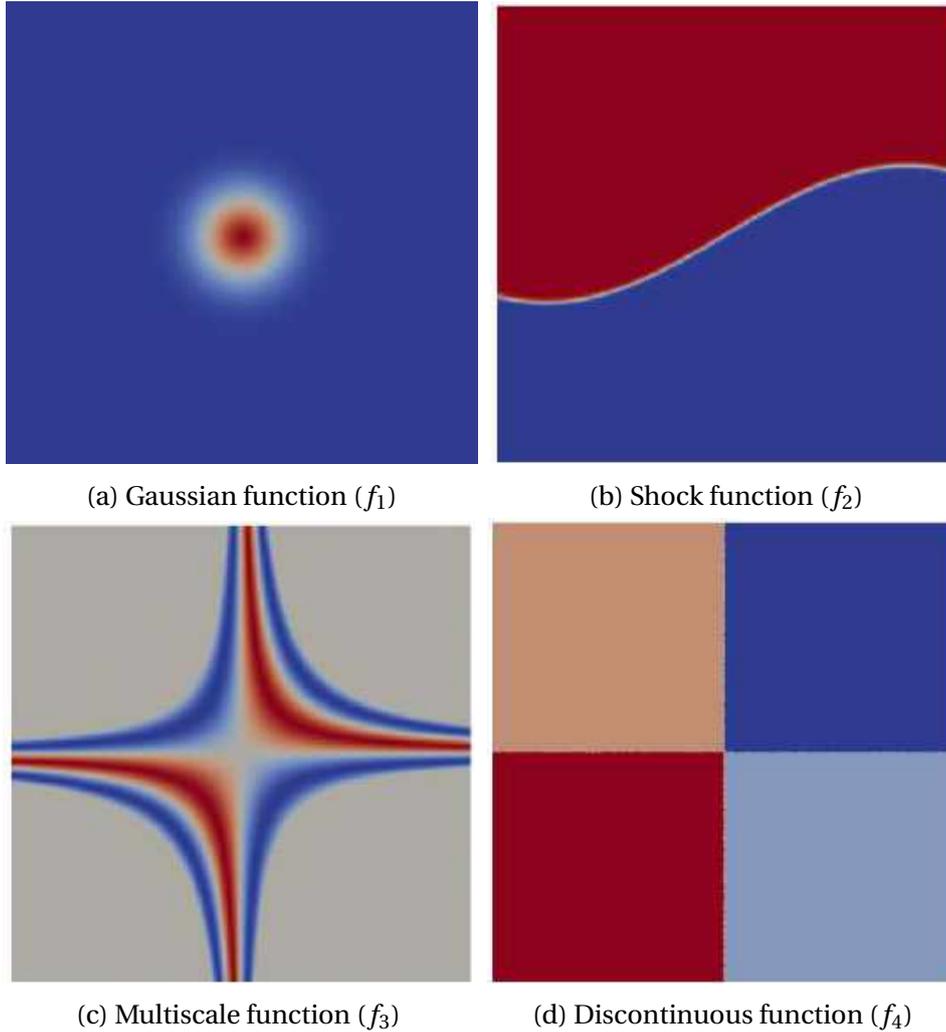


Figure 2.11: 2D representation of the functions

and $(\mathcal{T}_2^i)_{i=1\dots 5}$. For each analytical case, we apply the function on the first mesh \mathcal{T}_1^i which provides us with a solution field u_1^i . This solution field is then transferred to the second mesh \mathcal{T}_2^i , and we get u_2^i . The series of meshes used for the different transfers are summarized in table 3.3. To investigate the performance of the algorithm, for each analytical case, we computed the relative mass variation for each transfer :

$$\frac{|m_{\mathcal{T}_1^i} - m_{\mathcal{T}_2^i}|}{m_{\mathcal{T}_2^i}} \quad (2.27)$$

Step	Number of nodes in \mathcal{T}_1^i	Number of nodes in \mathcal{T}_2^i
1	651	587
2	2235	2184
3	9713	9206
4	32413	30300
5	128631	101746

Table 2.1: 2D meshes used for the different transfers

A plot is given in 2.12, the P1-linear interpolation is represented in red, as for the conservative interpolation, it is represented in blue. We can clearly see that for meshes with more

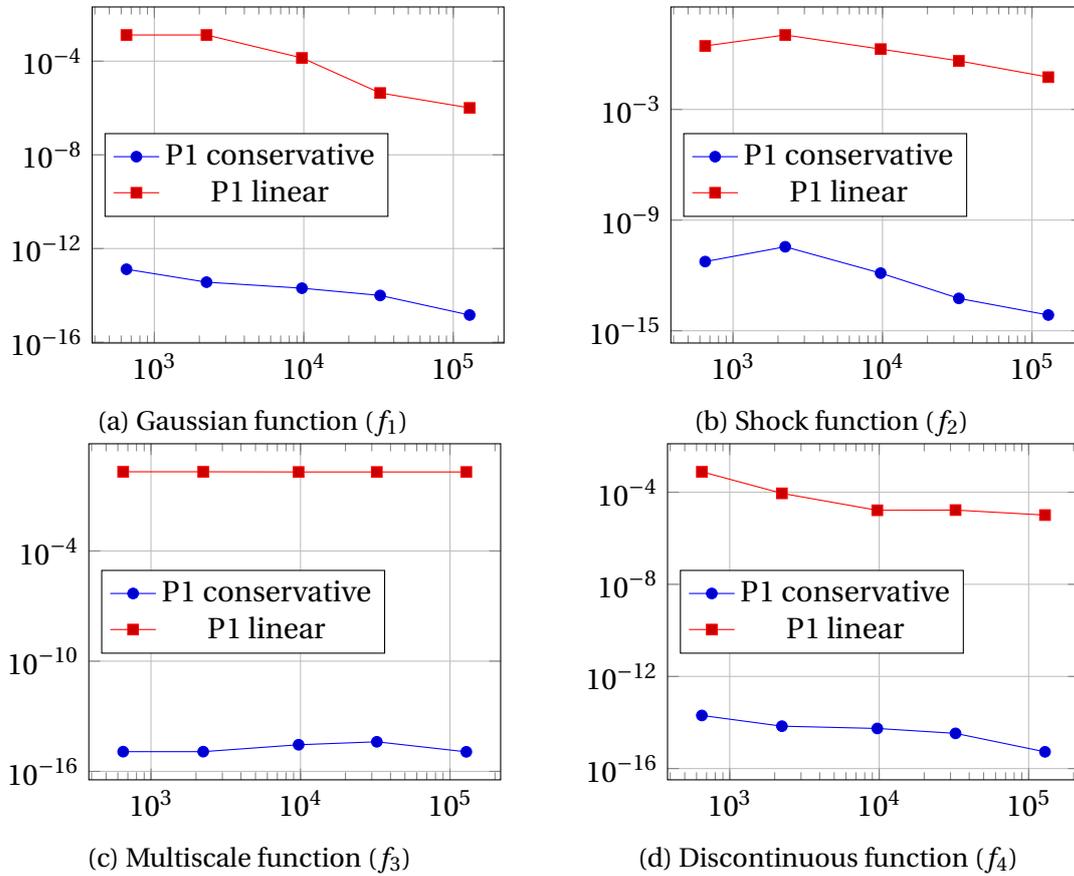


Figure 2.12: Mass variation vs. number of nodes

than 30000 vertices, the mass variation with linear interpolation is about 10^{-5} , against 10^{-13} for the conservative method. It is important to note that with the conservative interpolation method, the mass variation is of the order of the numerical zero for all the interpolation steps. We have a mass variation of order 10^{-2} with the linear interpolation against 10^{-12} with the conservative one, and this for coarse meshes. When using meshes with a higher number of nodes, we still have a large difference between the two methods for all the functions studied above. We note also that all these results are in good agreement with the work of [28].

2.5.2 3D Analytical test case

For completeness, we analyze the behavior of each interpolation operator (linear and conservative) after several interpolation steps using a new 3D analytical test case.

To this end, we define the following perturbation in a cubic domain $[-0.5, 0.5]^3$:

$$f(x, y, z) = \tanh(20(x + 0.3\sin(-10y) - 0.3\sin(-5(z - 0.1)))) \quad (2.28)$$

We want to emphasize a complex 3D situation using (i) anisotropic meshes with stretched elements and (ii) dynamic mesh adaptation. Therefore, we interpolate the analytical

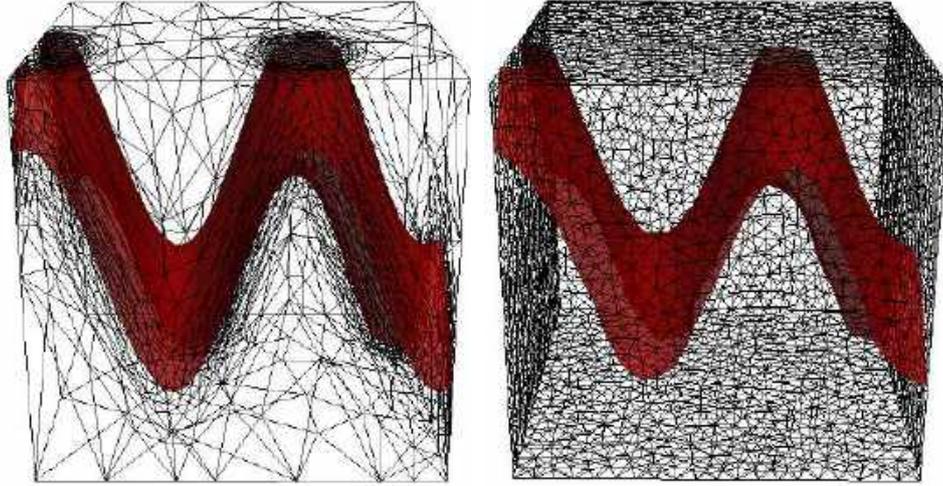


Figure 2.13: Left: 3D Anisotropic mesh - Right: 3D Isotropic mesh

function back and forth from one initial anisotropic mesh \mathcal{T}_1 of 31799 elements to an isotropic mesh \mathcal{T}_2 of 29414 tetrahedrons (see Figure 2.13). Interpolating back and forth from one mesh to another allows us to analyze the accumulated error after each interpolation step. This is crucial for mesh adaptation where a large number of interpolation steps are needed. To this end, the transfer from $\mathcal{T}_1^i \rightarrow \mathcal{T}_2^i \rightarrow \mathcal{T}_1^i$ is performed 5 times leading to a total of 10 interpolation steps. To this end, we analyzed the error when the solution is re-interpolated back to the donor mesh by computing the gap between the analytical function applied on \mathcal{T}_2^i , i.e $u_{a,2}^i$ and the interpolated one u_2^i in the L1-norm :

$$\varepsilon_i = \int_{\mathcal{T}_2^i} |u_{a,2}^i - u_2^i| \quad (2.29)$$

Table 2.2 gives details on the error for each number of interpolation. As expected, the error remains fairly larger when using the linear interpolation. Note that similar conclusions were underlined in the literature, mainly in [28; 35].

2.5.3 2D Analytical test case with mesh adaptation

In this test case, we increase the complexity by adding dynamic mesh adaptation. First, we compute an analytical function on a 2D domain $[-1, 1]^2$ composed of 4 elements:

$$f_6(x, y) = \tanh(100(y + 0.1\sin(-20x))) \quad (2.30)$$

The targeted number of elements is 10000. The total number of iterations to obtain a converged mesh is 10. Therefore, at each step, the proposed global approach gradually refines

	Linear interpolation	Conservative interpolation
$\mathcal{T}_1 \rightarrow \mathcal{T}_2 \rightarrow \mathcal{T}_1$	0.0932863	0.0776079
$2 \times \mathcal{T}_1 \rightarrow \mathcal{T}_2 \rightarrow \mathcal{T}_1$	0.125586	0.093279
$3 \times \mathcal{T}_1 \rightarrow \mathcal{T}_2 \rightarrow \mathcal{T}_1$	0.156789	0.109485
$4 \times \mathcal{T}_1 \rightarrow \mathcal{T}_2 \rightarrow \mathcal{T}_1$	0.18643	0.125575
$5 \times \mathcal{T}_1 \rightarrow \mathcal{T}_2 \rightarrow \mathcal{T}_1$	0.214496	0.141359

Table 2.2: Error for the transfer from anisotropic to isotropic

the mesh along the high variation of the gradient of the solution. This latter is then interpolated on each new adapted mesh. Figure 2.14 shows all the details starting from the initial to the final adapted mesh. The results for each adaptation-interpolation step using linear and conservative interpolations are presented in table 2.3. Again, we can clearly highlight the benefit of the new implemented global conservative method compared to the linear interpolation approach.

Step	Linear interpolation	Conservative Interpolation
1	0,0014683	0.0007411
2	0.00428246	0.00311578
3	0.01074291	0.007144506
4	0.00335502	0.00222403
5	0.00072268	0.00026588
6	0.0018356	0.0011621
7	0.00131129	0.00122882
8	0.0006134	0.00052663
9	0.00237049	0.00204828
10	0.00027782	0.0002267

Table 2.3: Comparison of the mass variation between both methods

2.5.4 2D Lid driven cavity : interpolation and dynamic mesh adaptation

In this section, we assess the performance of the method by bench-marking the driven cavity at different Reynolds numbers and by comparing to very accurate reference solutions. This test has been widely used as a benchmark for numerical methods and has been analyzed by various authors [45–47]. The considered test case has been sketched in Figure 2.15. A fluid is contained in a square domain with no slip boundary conditions at three borders, while a constant tangent unity velocity is imposed at the last border. The zero level pressure has been imposed to the vertex opposite to the applied shear velocity, and the gravity is set to zero. The source term is identical to zero. The time step is fixed to 0.1s. The density is fixed to $1 \text{ kg} \cdot \text{m}^{-3}$ and the viscosity is adjusted in order to obtain Reynolds number of 1000 and 5000. The Navier-Stokes equations are solved by a variational multiscale approach in the framework of stabilized finite-elements (see Chapter 3). An important point to mention before we start is that the following mesh adaptations are always performed for a fixed number of nodes. It avoids a drastic increase in the number of unknowns and thus computational cost. Therefore, for Reynolds number of 1000

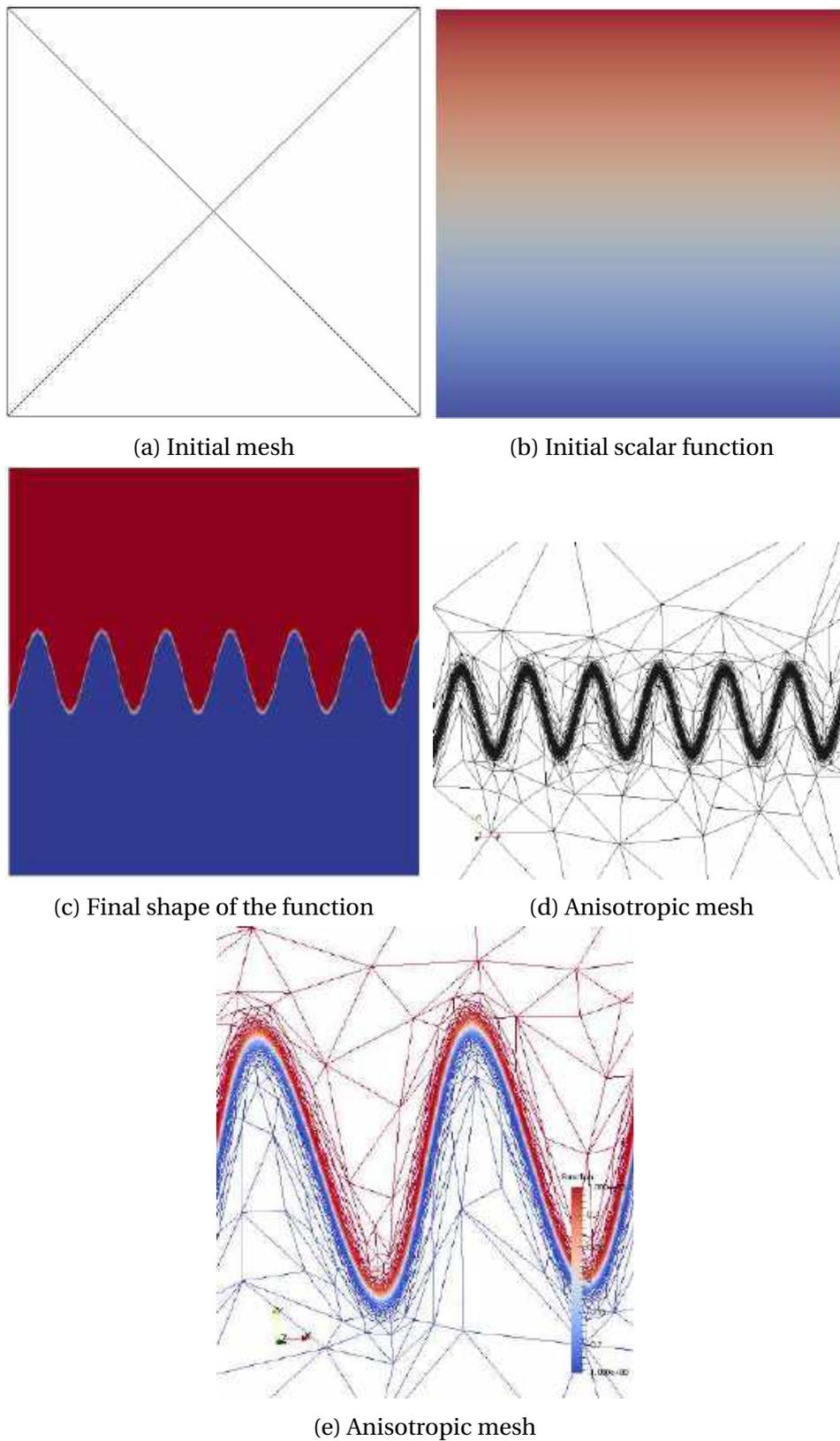


Figure 2.14: Meshes and 2D representations of the function

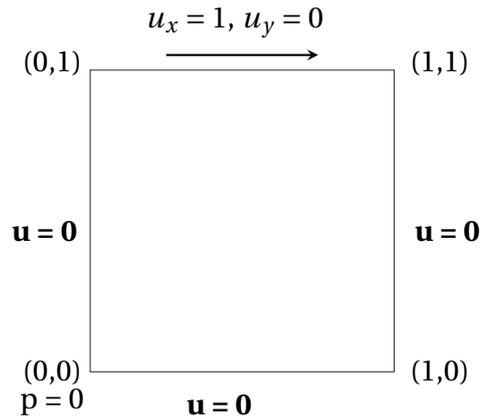


Figure 2.15: Set up of the 2D lid-driven cavity

and 5000 we fixed the number of nodes to be equal to 10000. The adaptive process always starts from the uniform mesh presented in Figure 2.16. The mesh is adapted to the velocity field at each stage using the global conservative method. The results on the two respective converged meshes ($\approx 10\,000$ nodes) can be seen in Figure 2.17. The elements

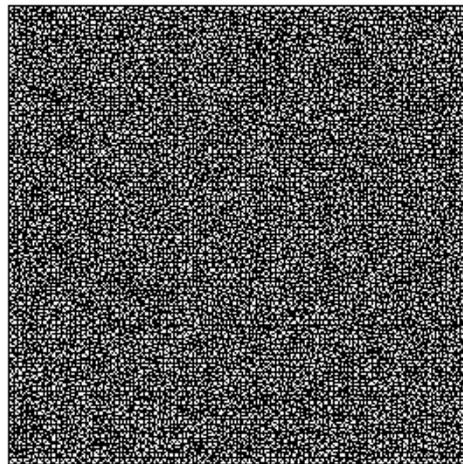


Figure 2.16: Initial mesh for the 2D driven cavity flow

at the central bulk of the cavity region around the primary vortex are mostly isotropic and increase in size as the Reynolds number increases. We first plot the velocity profiles for V_x along $y = 0.5$ and for V_y along $x = 0.5$ for Reynolds number 1000 and 5000. Figure 2.18 shows respectively that all the results are in good agreement with precise results found in [45]. Recall that the objective of these simulations is to also emphasize the benefits of using the new implemented method when dealing with both dynamic and anisotropic meshes. Therefore, we simply computed the accumulation of loss of linear momentum with each method. Three simulations have been done, each with a different adaptation frequency, while keeping a Reynolds number equal to 1000. The accumulated error using both approach are depicted in tables 2.4 and 2.5. A significant difference is noticed in particular when we adapt every time step, which that the fields are being interpolated at each iteration. This latter confirms the benefit of the implemented global conservative approach when solving the unsteady incompressible Navier-Stokes equations using linear stabilized finite element method.

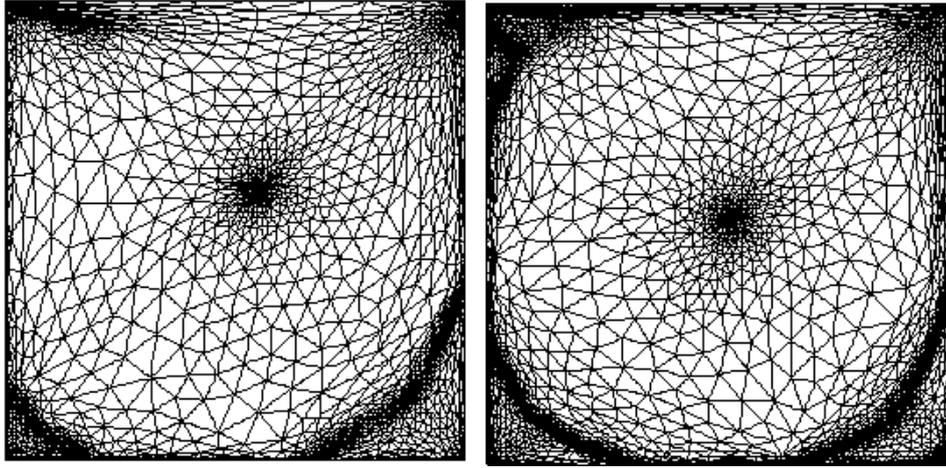


Figure 2.17: Anisotropic meshes at Reynolds 1000 and 5000.

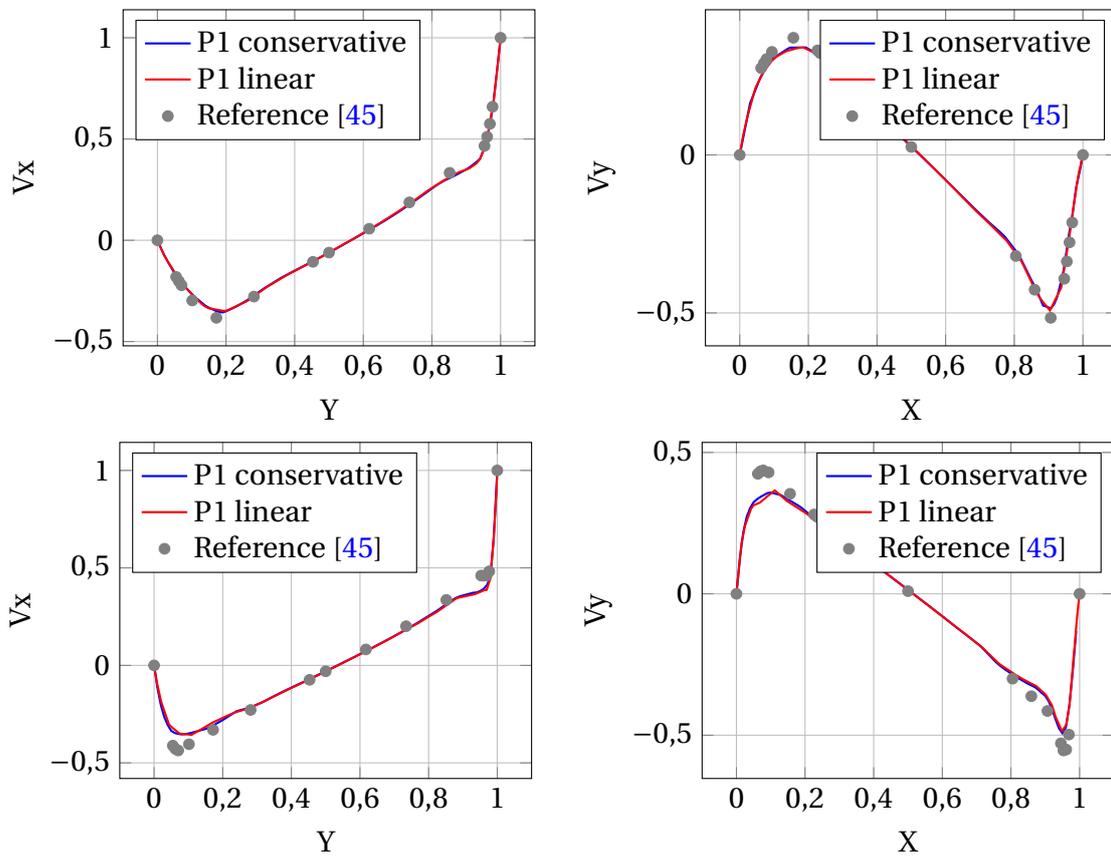


Figure 2.18: Comparison of velocity profiles in the mid-planes for $Re = 1000$ (top), for $Re = 5000$ (bottom). Left: Velocity profiles for V_x along $y = 0.5$. Right: Velocity profiles for V_y along $x = 0.5$

Adaptation Step	P1 linear	P1 conservative	% Variation
1	6	1.8	70
5	2.5	0.6	76
10	1.7	0.3	71

Table 2.4: Final momentum loss in direction x

Adaptation Step	P1 linear	P1 conservative	% Variation
1	1	0.45	55
5	0.83	0.47	43
10	0.4	0.21	47.5

Table 2.5: Final momentum loss in direction y

2.5.5 3D Lid driven cavity : interpolation and dynamic mesh adaptation

In this section, we test again the conservative adaptation-interpolation method on the 3D lid driven cavity. Same as in 2D, Dirichlet boundary conditions are imposed on the velocity. At the upper wall at $z = 1$ it is equal to one, and zero elsewhere, the set up is shown in Figure 2.19. The source term is identical to zero. The time step is fixed to 0.05 s. The viscosity is adjusted in order to obtain Reynolds number of 1000. The numerical example is done with a fixed number of elements ($\approx 300\,000$). The mesh is adapted to the velocity at each time step using our combined mesh adaptation algorithm and conservative interpolation. Figure 2.20 highlights the stretching of the elements near the walls

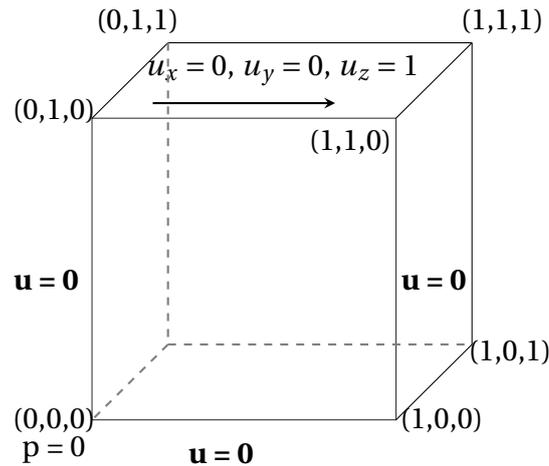


Figure 2.19: Set up of the 3D lid-driven cavity

and the corners. In the same way as the 2D test case, we computed the accumulation of momentum loss with respect to each direction. Table 2.6 highlights as expected a significant difference between the two approaches, confirming again the important role of the proposed global conservative method.

Direction	P1 linear	P1 conservative	% Variation
x	1.1	0.5	54.5
y	0.2	0.1	50
z	0.002	0.001	50

Table 2.6: Final momentum loss in each direction for the 3D lid driven cavity

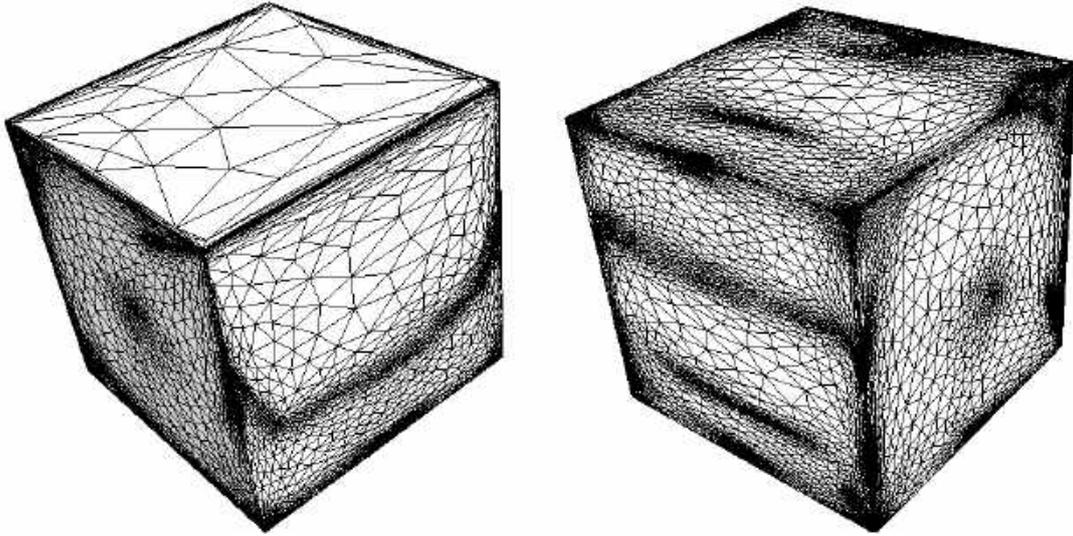
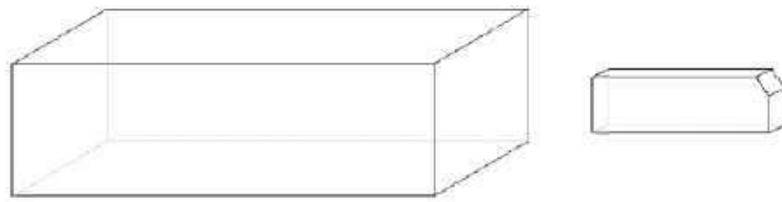


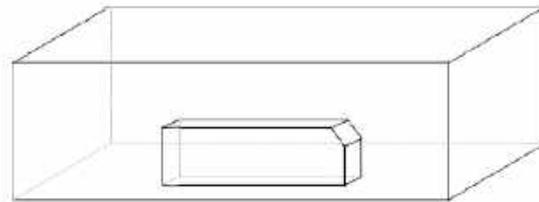
Figure 2.20: 3D Anisotropic meshes at Reynolds 1000

2.5.6 Unsteady flow past a 3D vehicle model

In order to demonstrate the efficiency of the conservative eulerian framework, we consider in this example a 3D computational domain that presents heterogeneous properties. This domain is discretized using a single mesh on which we use an immersion technique to place an object inside. The flow past a 3D vehicle model is often employed to assess new implementations in CFD. Indeed, the sharp and the localized flow features that they develop remain challenging to simulate. For the same reason, they represent a good test for mesh adaptation procedures. Here, we simulate the flow past an immersed vehicle model as proposed in [48]. The computation is done using the immersed volume method. The surface mesh of the vehicle geometry is immersed in a rectangular 3D volumic meshed domain. Figure 2.21 provides a schematic representation of the mesh generation process accounting for an immersed body. This test case is one of the first benchmarks proposed to investigate the stability and fuel consumption of an auto-mobile at high cruising speeds. It represents a simplified car geometry ($0.89 \times 0.35 \times 0.197m$) that can be used to study the automotive aerodynamics and isolate relevant flow phenomena. A critical slant angle of 30° was found to lead to a dramatical change in the flow pattern which leads to more complex flow behaviors. The numerical example is done with a fixed number of elements ($\approx 200\,000$). The mesh is adapted to the velocity at each time step using our combined mesh adaptation algorithm and conservative interpolation. The main objective here is again to only highlight the usefulness and the applicability of the proposed conservative method. Figure 2.23 shows the obtained anisotropic adapted mesh around the immersed vehicle, we can clearly see that the elements are stretched along the interface of the geometry allowing a good capture of the shape of the vehicle and the flow behavior near that region. Table 2.7 highlights as expected the improvement in terms of conservation of the proposed method when solving the unsteady incompressible Navier-Stokes equations coupled with dynamic mesh adaptation when dealing with immersed methods.



(a) 3D domain and vehicle geometry



(b) Vehicle geometry immersed in the 3D domain

Figure 2.21: Immersion of the vehicle geometry in a 3D domain

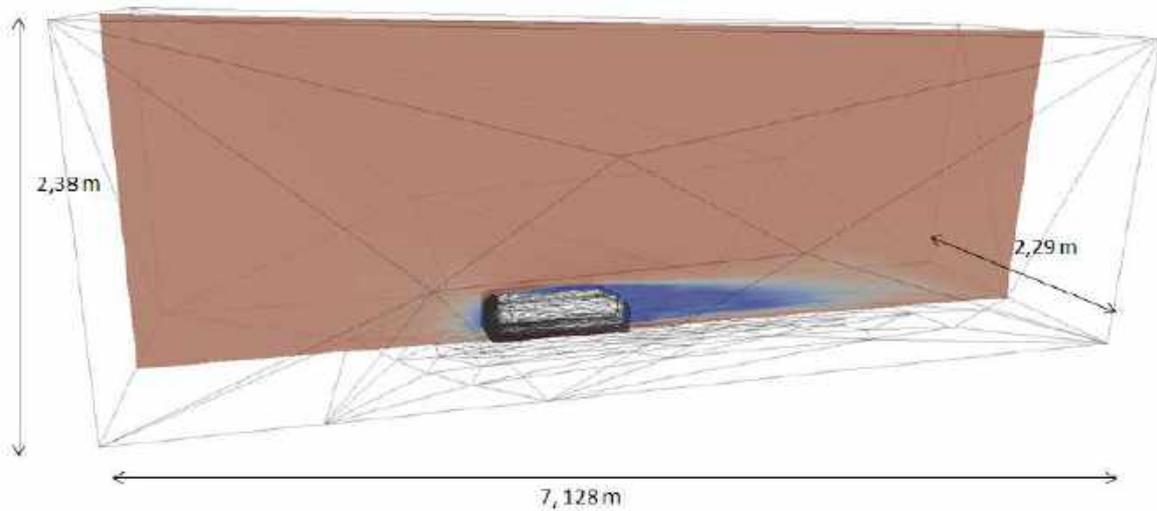


Figure 2.22: Set up of the 3D vehicle model

Direction	P1 linear	P1 conservative	% Variation
x	8.109	1.124	86
y	1.124	0.211	81
z	0.743	0.298	59

Table 2.7: Final momentum loss in each direction for the unsteady flow past a vehicle

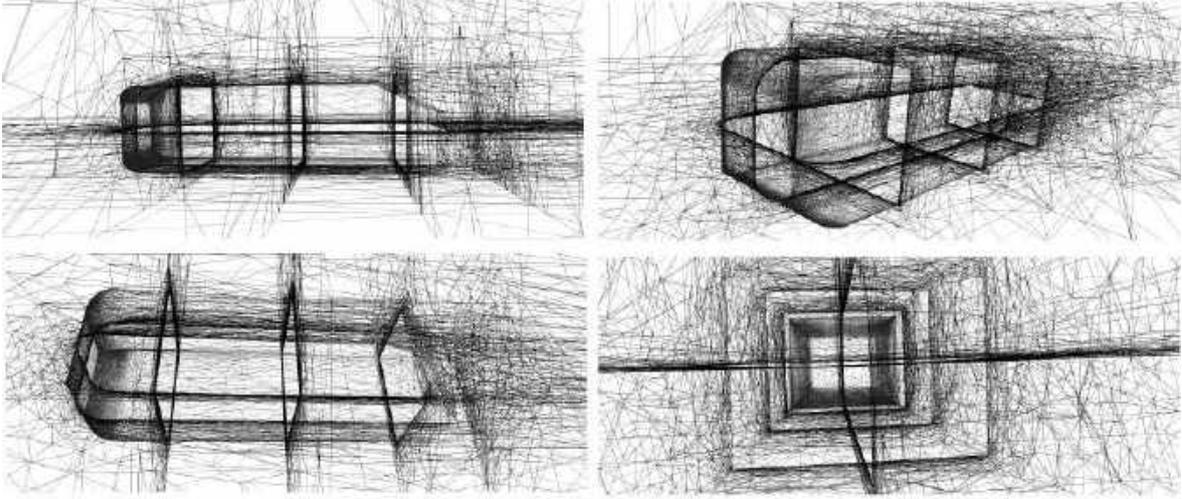


Figure 2.23: 3D Anisotropic meshes for the unsteady flow past a vehicle

2.5.7 Scalability study on the 3D Lid driven cavity

We propose in this section a scalability analysis for the 3D Lid driven cavity presented before. Therefore, we use a mesh of 600 000 elements, we keep a fixed problem size but we vary the number of processors. Indeed, 4 different computations are launched on our Cluster Intel over 8, 12, 28 and 56 processors. The computational time of each computation is given in Table 2.8. The computational time data are extracted at the end of each simulation.

Number of processors	Computational time in minutes	Speed up
8	20	1
12	11.6	1.7
28	6.8	2.9
56	4.08	4.9

Table 2.8: Computational time speed up on the Cluster Intel

To investigate the performance of the parallel conservative framework, along this table, we present in Figure 2.24 the scalability chart of the conservative interpolation algorithm. In particular, we plot the speed-up realized by doubling, tripling and quadrupling the allocated resources. The time reference to compute these speed-ups is the computational time corresponding to 8 processor. This speed-up chart show linear behavior with a determination coefficient close to one. This information give us confidence on the predictability and the scalability of the implemented conservative interpolation algorithm.

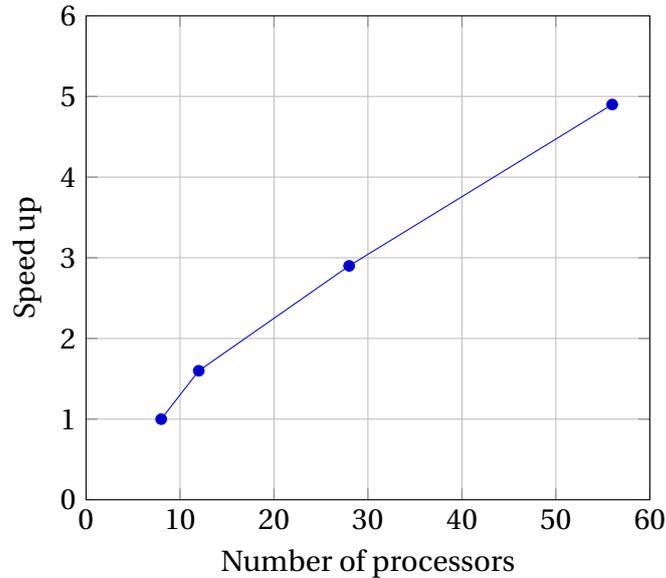


Figure 2.24: Scalability study on the conservative interpolation algorithm

2.6 Conclusion

In this Chapter, we briefly explained the different components of the immersed volume method, with an emphasize on the mesh adaptation and conservative interpolation procedure. The proposed mesh generation algorithm allows the creation of meshes with extremely anisotropic elements stretched along the interface, which is an important requirement for multiphase applications such as the quenching process. The mesh adaptation strategy relies on the a posteriori definition of a metric field able, in some specified sense, to drive the remeshing procedure so that the interpolation error on the Finite Element solution is minimized. In order to relate the length distribution tensor to the interpolation error, we define an edge based error estimator based on a gradient recovery procedure. And we emphasized the fact that a key feature of the proposed adaptation method is its ability to enhance the accuracy while preserving a fixed number of nodes which is a great advantage for industrial applications. We explained that with every mesh adaptation step, we must transfer the solutions fields from one initial mesh to the newly adapted one. We the, presented a new conservative interpolation method based on the conservation of the global physical quantities. We imposition the continuity of the solution using the L_2 norm and relax the solution by imposing the conservation by solving an optimization problem using Lagrange multipliers. This method involves assembling and solving a linear system. We implemented this method in our library Cimlib-CFD with the help of the external library PETSc. Since we are dealing with heavy complex industrial problems, all computations must be launched on several processors. Thus, the parallel mesh adaptation framework was briefly presented. Then, the parallel implementation of the method was detailed in this Chapter. To assemble the system in parallel, a special treatment of the nodes at the interfaces between the different processors has been used. As for the assembly of the system, a global numbering of the nodes of the mesh had to be taken into account. Thus, a local (from each processor) to global (all the processors) mapping of the different nodes was used, all these elements were presented in this Chapter. A scalability study was performed in order to test the parallel implementation of the conservative interpolation algorithm and the results were satisfying. Finally, in the

last section, we have validated the accuracy and efficiency of the conservative Eulerian framework on several analytical functions presenting smooth variations, sharp discontinuities, and multiscale features. Then, we presented various numerical examples and benchmarks taken from the literature. To assess the accuracy and the validity of the presented methods, several comparisons with the literature were proposed. The objective in the upcoming chapter is to apply the developed adaptation tool in the context of multiphase applications, by introducing the level set technique and the multiphase Navier Stokes equations.

2.7 Résumé du chapitre en français

Dans ce chapitre, nous avons brièvement introduits les différents composants de la méthode d'immersion de volume, en mettant l'accent sur l'adaptation de maillage et l'interpolation conservative. La méthode proposée permet la création de maillages avec des éléments extrêmement étirés au long de l'interface, ce qui est très important pour des applications multiphasiques telles que les procédés de trempe industrielle. Nous avons ainsi construits le tenseur métrique associé à l'espace métrique sur chaque nœud du maillage. Ce facteur a été déterminé en fonction de l'erreur induite le long des arêtes connectées au nœud. Un estimateur d'erreur a été proposé et a été exprimé en fonction du gradient du paramètre de remaillage projeté sur les arêtes. Et nous avons souligné le fait qu'une caractéristique clé de la méthode d'adaptation proposée est sa capacité à améliorer la précision tout en préservant un nombre fixe de nœuds ; ceci constitue un avantage non négligeable pour les applications industrielles. Par la suite, nous avons expliqué qu'à chaque étape de l'adaptation du maillage, nous devons transférer les champs de solutions d'un maillage initial à celui qui a été adapté. Nous avons présenté une nouvelle méthode d'interpolation conservative. Cette méthode est basée sur la conservation des quantités physiques globales. Nous imposons la continuité de la solution en utilisant la norme L_2 et on impose la conservation en résolvant un problème d'optimisation à l'aide de multiplicateurs de Lagrange. Cette méthode consiste à assembler et à résoudre un système linéaire. Nous avons implémenté cette méthode dans notre bibliothèque Cimlib-CFD avec l'aide de la bibliothèque externe PETSc. Comme nous traitons de problèmes industriels complexes, tous les calculs doivent être lancés sur plusieurs processeurs. Ainsi, l'adaptation de maillage dans un cadre parallèle a été brièvement présenté. Ensuite, la mise en œuvre parallèle de la méthode a été détaillée dans ce chapitre. Pour assembler le système en parallèle, un traitement spécial des nœuds aux interfaces entre les différents processeurs a été utilisé. En ce qui concerne l'assemblage du système, une numérotation globale des nœuds du maillage doit être prise en compte. Ainsi, un 'mapping' local (de chaque processeur) à global (tous les processeurs) des différents nœuds a été utilisé. Tous ces éléments ont été présentés dans ce chapitre. Enfin, dans la dernière section, nous avons validé la précision et l'efficacité du cadre eulérien sur plusieurs fonctions analytiques présentant des variations lisses, des discontinuités ainsi que des fonctionnalités multi-échelles. Nous avons ensuite présenté divers exemples numériques et benchmarks tirés de la littérature. Pour évaluer l'exactitude et la validité des méthodes présentées, plusieurs comparaisons avec la littérature ont été proposées. L'objectif dans le chapitre suivant consiste à appliquer l'outil d'adaptation développé dans le contexte d'applications multiphasiques, en introduisant la technique de level set et les équations multiphasiques Navier Stokes.

2.8 Bibliography

- [1] E. Aulisa, S. Manservigi, P. Seshaiyer, A computational multilevel approach for solving 2d navier–stokes equations over non-matching grids, *Computer methods in applied mechanics and engineering* 195 (33-36) (2006) 4604–4616. [18](#)
- [2] N. Moës, J. Dolbow, T. Belytschko, A finite element method for crack growth without remeshing, *International journal for numerical methods in engineering* 46 (1) (1999) 131–150. [18](#)
- [3] L. Lee, R. J. LeVeque, An immersed interface method for incompressible navier–stokes equations, *SIAM Journal on Scientific Computing* 25 (3) (2003) 832–856. [18](#)
- [4] R. P. Fedkiw, T. Aslam, B. Merriman, S. Osher, A non-oscillatory eulerian approach to interfaces in multimaterial flows (the ghost fluid method), *Journal of computational physics* 152 (2) (1999) 457–492.
- [5] R. Glowinski, T.-W. Pan, A. J. Kearsley, J. Periaux, Numerical simulation and optimal shape for viscous flow by a fictitious domain method, *International Journal for Numerical Methods in Fluids* 20 (8-9) (1995) 695–711. [18](#)
- [6] D. Lakehal, M. Meier, M. Fulgosi, Interface tracking towards the direct simulation of heat and mass transfer in multiphase flows, *International Journal of Heat and Fluid Flow* 23 (3) (2002) 242–257. [18](#)
- [7] M. Bernacki, Y. Chastel, T. Coupez, R. E. Logé, Level set framework for the numerical modelling of primary recrystallization in polycrystalline materials, *Scripta Materialia* 58 (12) (2008) 1129–1132.
- [8] R. Valette, T. Coupez, C. David, B. Vergnes, A direct 3d numerical simulation code for extrusion and mixing processes, *International Polymer Processing* 24 (2) (2009) 141–147.
- [9] J. Bruchon, H. Digonnet, T. Coupez, Using a signed distance function for the simulation of metal forming processes: Formulation of the contact condition and mesh adaptation. from a lagrangian approach to an eulerian approach, *International journal for numerical methods in engineering* 78 (8) (2009) 980–1008.
- [10] E. Hachem, T. Kloczko, H. Digonnet, T. Coupez, Stabilized finite element solution to handle complex heat and fluid flows in industrial furnaces using the immersed volume method, *International Journal for Numerical Methods in Fluids* 68 (1) (2012) 99–121. [18](#)
- [11] T. Coupez, A mesh improvement method for 3d automatic remeshing, *Numerical Grid Generation in Computational Fluid Dynamics and Related Fields* (1994) 615–626. [19](#)
- [12] X. Li, M. S. Shephard, M. W. Beall, 3d anisotropic mesh adaptation by mesh modification, *Computer methods in applied mechanics and engineering* 194 (48-49) (2005) 4915–4950.
- [13] J.-F. Remacle, X. Li, M. S. Shephard, J. E. Flaherty, Anisotropic adaptive simulation of transient flows using discontinuous galerkin methods, *International Journal for Numerical Methods in Engineering* 62 (7) (2005) 899–923. [19](#)

- [14] L. Formaggia, S. Perotto, Anisotropic error estimates for elliptic problems, *Numerische Mathematik* 94 (1) (2003) 67–92. [19](#)
- [15] T. Coupez, Génération de maillage et adaptation de maillage par optimisation locale, *Revue européenne des éléments finis* 9 (4) (2000) 403–423. [19](#), [21](#)
- [16] T. Coupez, Metric construction by length distribution tensor and edge based error for anisotropic adaptive meshing, *Journal of computational physics* 230 (7) (2011) 2391–2405. [19](#), [20](#)
- [17] F. Alauzet, P. Frey, Estimateur d’erreur géométrique et métriques anisotropes pour l’adaptation de maillage. partie i: aspects théoriques. [21](#)
- [18] J. R. Cebal, R. Lohner, Conservative load projection and tracking for fluid-structure problems, *AIAA journal* 35 (4). [23](#)
- [19] R. Jaiman, X. Jiao, P. Geubelle, E. Loth, Assessment of conservative load transfer for fluid–solid interface with non-matching meshes, *International Journal for Numerical Methods in Engineering* 64 (15) (2005) 2014–2038. [23](#)
- [20] X. Jiao, M. T. Heath, Common-refinement-based data transfer between non-matching meshes in multiphysics simulations, *International Journal for Numerical Methods in Engineering* 61 (14) (2004) 2402–2427. [23](#)
- [21] X. Jiao, M. T. Heath, Overlaying surface meshes, part i: Algorithms, *International Journal of Computational Geometry & Applications* 14 (06) (2004) 379–402. [23](#)
- [22] P. Farrell, M. Piggott, C. Pain, G. Gorman, C. Wilson, Conservative interpolation between unstructured meshes via supermesh construction, *Computer Methods in Applied Mechanics and Engineering* 198 (33) (2009) 2632–2642. [23](#)
- [23] P. Farrell, J. Maddison, Conservative interpolation between volume meshes by local galerkin projection, *Computer Methods in Applied Mechanics and Engineering* 200 (1) (2011) 89–100. [23](#)
- [24] T. Q. Bui, D. Q. Vo, C. Zhang, et al., A consecutive-interpolation quadrilateral element (cq4): formulation and applications, *Finite Elements in Analysis and Design* 84 (2014) 14–31. [23](#)
- [25] D. Ribarić, G. Jelenić, Higher-order linked interpolation in quadrilateral thick plate finite elements, *Finite Elements in Analysis and Design* 51 (2012) 67–80. [23](#)
- [26] J. Grandy, Conservative remapping and region overlays by intersecting arbitrary polyhedra, *Journal of Computational Physics* 148 (2) (1999) 433–466. [23](#)
- [27] R. H. Bailey, An algorithm for the conservative interpolation of data between two-dimensional structured or unstructured triangular meshes, Ph.D. thesis, University College of Swansea (1987). [23](#)
- [28] F. Alauzet, M. Mehrenberger, P1-conservative solution interpolation on unstructured triangular meshes, *International Journal for Numerical Methods in Engineering* 84 (13) (2010) 1552–1588. [23](#), [32](#), [34](#), [35](#)

- [29] F. Alauzet, A parallel matrix-free conservative solution interpolation on unstructured tetrahedral meshes, *Computer Methods in Applied Mechanics and Engineering* 299 (2016) 116–142. [23](#)
- [30] S. Chippada, C. Dawson, M. Martinez, M. Wheeler, A projection method for constructing a mass conservative velocity field, *Computer Methods in Applied Mechanics and Engineering* 157 (1-2) (1998) 1–10. [23](#)
- [31] D. Brancherie, P. Villon, A. Ibrahimbegovic, A. Rassineux, P. Breitkopf, Field transfer in nonlinear structural mechanics based on diffuse approximation, in: *International Conference on Computational Plasticity COMPLAS VIII*, 2005. [23](#)
- [32] G. Chesshire, W. D. Henshaw, A scheme for conservative interpolation on overlapping grids, *SIAM Journal on Scientific Computing* 15 (4) (1994) 819–845. [23](#)
- [33] G. Houzeaux, R. Codina, Transmission conditions with constraints in finite element domain decomposition methods for flow problems, *International Journal for Numerical Methods in Biomedical Engineering* 17 (3) (2001) 179–190. [24](#)
- [34] A. Pont, R. Codina, J. Baiges, Interpolation with restrictions between finite element meshes for flow problems in an ale setting, *International Journal for Numerical Methods in Engineering*. [24](#), [27](#)
- [35] C. Bahbah, Y. Mesri, E. Hachem, Interpolation with restrictions in an anisotropic adaptive finite element framework, *Finite Elements in Analysis and Design* 142 (2018) 30–41. [24](#), [35](#)
- [36] R. Löhner, Robust, vectorized search algorithms for interpolation on unstructured grids, *Journal of computational Physics* 118 (2) (1995) 380–387. [24](#)
- [37] A. Guttman, R-trees: a dynamic index structure for spatial searching, Vol. 14, ACM, 1984. [25](#)
- [38] R. Boussetta, T. Coupez, L. Fourment, Adaptive remeshing based on a posteriori error estimation for forging simulation, *Computer methods in applied mechanics and engineering* 195 (48-49) (2006) 6626–6645. [v](#), [25](#)
- [39] S. Balay, S. Abhyankar, M. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, A. Dener, V. Eijkhout, W. Gropp, et al., *Petsc users manual*. [27](#), [30](#)
- [40] Y. Saad, M. H. Schultz, Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems, *SIAM Journal on scientific and statistical computing* 7 (3) (1986) 856–869. [27](#)
- [41] H. De Cougny, M. S. Shephard, Parallel refinement and coarsening of tetrahedral meshes, *International Journal for Numerical Methods in Engineering* 46 (7) (1999) 1101–1125. [28](#)
- [42] T. Coupez, H. Dignonnet, R. Ducloux, Parallel meshing and remeshing, *Applied Mathematical Modelling* 25 (2) (2000) 153–175.
- [43] R. Said, N. Weatherill, K. Morgan, N. Verhoeven, Distributed parallel delaunay mesh generation, *Computer methods in applied mechanics and engineering* 177 (1-2) (1999) 109–125. [28](#)

- [44] N. Capit, G. Da Costa, Y. Georgiou, G. Huard, C. Martin, G. Mounié, P. Neyron, O. Richard, A batch scheduler with high level components, in: CCGrid 2005. IEEE International Symposium on Cluster Computing and the Grid, 2005., Vol. 2, IEEE, 2005, pp. 776–783. [32](#)
- [45] U. Ghia, K. N. Ghia, C. Shin, High-re solutions for incompressible flow using the navier-stokes equations and a multigrid method, *Journal of computational physics* 48 (3) (1982) 387–411. [36](#), [38](#), [39](#)
- [46] M. Sahin, R. G. Owens, A novel fully implicit finite volume method applied to the lid-driven cavity problem?part i: High reynolds number flow calculations, *International journal for numerical methods in fluids* 42 (1) (2003) 57–77.
- [47] E. Hachem, B. Rivaux, T. Kloczko, H. Dignonnet, T. Coupez, Stabilized finite element method for incompressible flows with high reynolds number, *Journal of Computational Physics* 229 (23) (2010) 8643–8665. [36](#)
- [48] E. Hachem, S. Feghali, R. Codina, T. Coupez, Anisotropic adaptive meshing and monolithic variational multiscale method for fluid–structure interaction, *Computers & Structures* 122 (2013) 88–100. [41](#)

Chapter 3

Moving interface capturing

Contents

3.1 Introduction	52
3.2 Level set function	53
3.2.1 Basic definition of the level set function	53
3.2.2 Convective reactive level set method	55
3.3 Mixing laws	59
3.4 The incompressible Navier Stokes equation	60
3.4.1 Governing equations	60
3.4.2 Galerkin finite element formulation	61
3.4.3 Variational multiscale (VMS) approximation	62
3.5 Numerical examples	66
3.5.1 2D dam break	67
3.5.2 2D droplet splashing on thin liquid film at different Reynold numbers	69
3.5.3 2D rising bubble	72
3.5.4 3D simulations of the axisymmetric and non-axisymmetric merging of two bubbles	74
3.6 Conclusion	80
3.7 Résumé du chapitre en français	80
3.8 Bibliography	81

3.1 Introduction

Multiphase flows is the term given for fluids with different properties. Multiphase flows appear in a wide variety of natural processes and industrial applications such as geophysical flows, water waves, drop impacts, micro-fluidics, bio mechanics and many others. These applications typically involve immiscible fluids that are separated by thin layers known as the interface. The interface is the region across which the fluids properties as well as some of the flow variables are subjected to variations and deformations. It is a sharp front where density and viscosity change abruptly. In spite of the maturity and popularity of numerical formulations, they are still characterized by a high computational cost and may lack of reliability and generality. The interaction among many entities such as bubbles, drops, or particles immersed in a fluid is not the only source of complexity usually exhibited by multiphase flows. The challenge in the numerical simulation of multiphase flows is how to represent the interface and to model its kinematics, or the discretization mesh for moving interfaces.

To follow moving interfaces, mesh-based methods still hold a prevalent position [1]. They can be divided into two categories : interface tracking and interface capturing depending on whether the interface is explicitly described or defined implicitly by an auxiliary function. For instance, front tracking methods rely on considering marker particles to explicitly keep track of the interface [2–8]. These methods provide very accurate interface evolution; however changes in topology need to be handled explicitly as well, which adds to the complexity of those methods. On the contrary, implicit representation treats changes in topology in a straightforward manner. In the context of two-fluid flows, the volume of fluid (VOF) [9–14] method allows mass conservation by construction, but its discontinuous description of the interface makes the computation of the curvature and the normal to the interface difficult. Although the level set method [15–18] does not have the same conservation properties as the volume of fluid method, it offers many advantages. To start, the implicit representation of the interface allows to accurately compute changes in topology when breakup or coalescence occur. Moreover, since we do not need to explicitly reconstruct the interface, the method is suitable when a calculation of geometric features such as normal or curvature is needed. However, the mass conservation, the efficiency and the robustness of the computations are difficult to achieve without dynamic adaptive methods and quantitative error estimation.

We have described in the previous chapter the mesh adaptation technique used to adapt the interface between two different materials. Recall that the monolithic resolution, based on the level set approach consists in considering a single grid for both fluid and solid for which only one set of equations need to be solved. Consequently, different subdomains are treated as a single fluid with variable material properties. Therefore one needs to compute a signed distance function to separate the different phases. In this chapter, we start by illustrating the level set function used for capturing the interfaces between immersed bodies and their surrounding. We give details on how to compute the signed distance function and how to follow the evolution of these interfaces over time. The steps towards a more conservative and less dissipative level set method will be described. In the next section, we give details on how to mix the different material properties using a homogeneous distribution along the interface. The level set function will be used to distribute in space the respective material properties. Finally, in order to follow the evolution of the different interfaces and the motion of the fluid, we solve the Navier Stokes equations. The

finite element implementation of the unsteady Navier Stokes equation will be described and analyzed in the last section. Since the standard Galerkin formulation is not robust enough to model complex convection dominated flows, a stabilization algorithm will be used to guarantee the convergence of the solution.

3.2 Level set function

Level set methods were first developed and used in the context of computer graphics and image restoration [19]. They were further developed and applied to incompressible flows by [20]. The method has been used in a large variety of applications such as phase change and boiling [21; 22], fluid structure interaction [23], two-fluid flows [18], etc. It is important to mention at this point that if we are dealing with simple geometries, for instance like a square or a sphere we can compute simple analytical functions to define the solids and build their corresponding meshes. However, in the context of industrial applications, we are dealing with more complex geometries, therefore we resort to the use of a surface mesh of the object. The surface mesh is then embedded in the computational domain which is in turn remeshed resulting in a single fluid/solid mesh.

3.2.1 Basic definition of the level set function

The most straightforward way for computing distance fields is through the use of a geometric brute force algorithm where the point-to-point distance is computed throughout the computational grid and the minimum distance for each point is stored. The basic idea of the level set method is that the interface can be implicitly tracked as the zero-isovalue of a scalar function named α . Let us define Ω as the whole computational domain and Γ the interface between the different sub-domains Ω_1 and Ω_2 . The level set function is defined as follows:

$$\alpha(x) = d(x, \Gamma) \quad \forall x \in \Omega \quad (3.1)$$

where d stands for the signed Eulerian distance function operator. As for the interface Γ , we define it as follows :

$$\Gamma = \{x \in \Omega \mid \alpha(x) = 0\} \quad (3.2)$$

In order to identify the different sub-domains, we check the sign of the level set function. The distance is positive if the node is located inside the solid, negative outside it, and zero at the level of the interface :

$$\begin{cases} \alpha(x) > 0 & \text{if } x \in \Omega_1 \\ \alpha(x) = 0 & \text{if } x \in \Gamma \\ \alpha(x) < 0 & \text{if } x \in \Omega_2 \end{cases} \quad (3.3)$$

A schematic representation of the level set function on a 2D circle shaped solid object is presented in Figure 3.1.

One very important point to mention here is that for any Eulerian distance function, the following property must always be verified :

$$\|\nabla\alpha\| = 1 \quad (3.4)$$

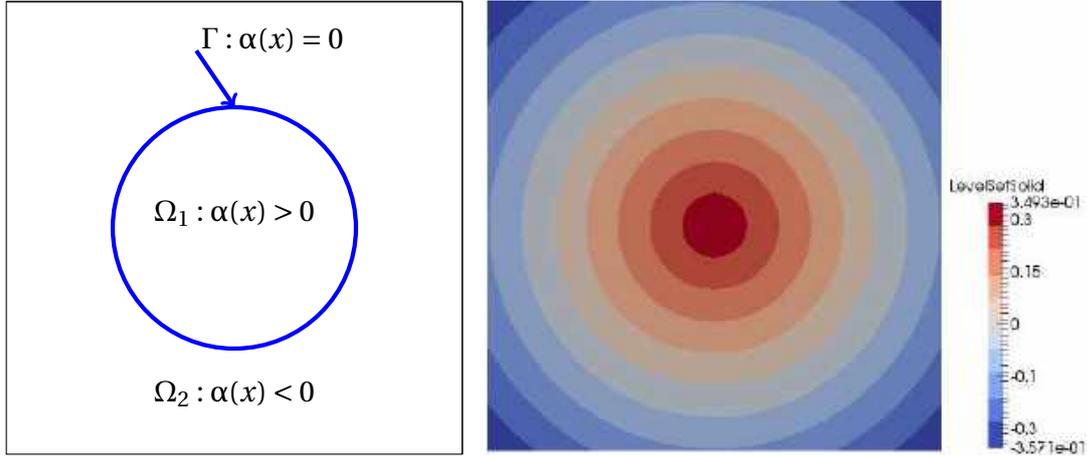


Figure 3.1: Right: schematic representation of the level set function for multi-domain problems. Left: definition of the level set function.

This property is very important since all physical properties will be distributed in space according to the level set. The level set function evolves according to the flow through the transport equation :

$$\frac{\partial \alpha}{\partial t} + u \cdot \nabla \alpha = 0 \quad (3.5)$$

It should be mentioned here that while α is initially a distance function, it will not remain so. While equation 3.5 will move the level set at the correct velocity v , α will no longer be a distance function ($\|\nabla \alpha\| \neq 1$), α can become irregular after some period of time. Therefore, one might notice a diffusion of the isovalues near the interface. And, if we use a non distance-function to distribute the material properties in the computational domain, we will change the physics of the problem since a nonphysical thickness is created near the interface. Hence, maintaining α as a distance function is essential for providing the interface with a width fixed in time.

The most common way to overcome this problem, is to reinitialize the level set function by solving a Hamilton-Jacobi problem [18]. Let us define by $s(\alpha)$ the sign of the level set function. The Hamilton-Jacobi problem is defined as follows :

$$\frac{\partial \alpha}{\partial \tau} + s(\alpha)(\|\nabla \alpha\| - 1) = 0 \quad (3.6)$$

where τ is a virtual time space in which the re-initialization equation is solved at each increment of the physical time domain. The steady state of equation 3.6 is reached when the gradient of the level set recovers its analytical value. If α was already close to a distance function, then one should not have to evolve too far in time. The solution remains its physical meanings since the zero isovalue of the the non re-initialized and the re-initialized level set functions are coincident. Note that the steady state solution for 3.6 will be the value of α for the following physical time-step.

At this point, many concerns arise. On one hand, we need to solve the level function on the whole domain, the CPU time needed for that process will grow very fast with large computations. On the other hand, due to numerical dissipation in simulations, the mass of each phase is not conserved which is crucial when modeling realistic multi-domains

simulations. The steps towards a more efficient, robust and conservative level set method will be presented in the following section.

3.2.2 Convective reactive level set method

In the literature, many approaches have been introduced in order to overcome the conservation issue. Olson and al, [24; 25] proposed to advect the level set method using a conservative scheme with an intermediate step that helps keeping the shape and the width of the profile across the interface constant. On the other hand, Sussman and al. [26; 27] proposed to combine the level set method with a VOF method in order to conserve mass when the interface is convected. Russo and Smereka [28] completed this work by introducing a subcell-fix that consists in using the location of the interface to compute the fluxes with second-order accuracy for cells containing the interface. This approach was later extended to fourth-order accuracy by [29]. In a different approach, the use of massless particles to complement the level set representation was introduced in [30]: it is called hybrid particle level set method and consists in correcting the level set function when the particles cross the interface. Others have taken advantage of the fact that the mass conservation error reduces when using adaptive mesh refinement, either by solving the level set equations on an auxiliary high-resolution equidistant cartesian grid [31] or refining the mesh in the regions close to the interface [32; 33]. In this work, we will use a convective reactive level set method. At first, a reaction term is added to the convection equation to reach higher precision [34]. Then, conservative mesh adaptation technique presented in the previous chapter is used to remedy the mass loss or increase.

Since the level set function itself is only used close to the interface, while for the rest of the domain only its sign counts, it is not necessary to compute the level set function far from the interface. For that reason, the first step towards a more conservative and less time consuming level set method is to apply a filter close to the interface. Filtering the level set, will allow us to have a gradient of the level set close to zero far from the interfaces and thus save computational time. Let us denote by $\tilde{\alpha}$, the new filtered level set function defined as follows :

$$\tilde{\alpha} = \frac{1}{1 + e^{-\frac{\alpha}{\varepsilon}}} \quad (3.7)$$

where ε is a scalar parameter defining the thickness of the truncation. Figure 3.2, shows an example of the filtering of the level set function. It is important to mention here that in order to provide a good capture of the interface of an object, we need to create a sharp gradient in its surrounding. Thus, the filtering process allow us to obtain a sharp gradient of $\tilde{\alpha}$. Note that, the parameter ε controls the steepness of the gradient and thus the anisotropy of the mesh in the vicinity of the interface. Therefore, the smaller ε , the sharper the gradient of α and hence the higher the stretching of the elements near the interface.

The standard level set method is a two step procedure: we start by solving the transport equation 3.5, then the Hamilton-Jacobi equation 3.6 should be solved to re-initialize the function. This approach is accurate, however, it is very time consuming. One way to overcome this difficulty is to embed the re-initiliazation equation into the transport one, [35]. So first of all, let us compute the derivative of the new filtered level set function $\tilde{\alpha}$, with respect to the basic level set function α :

$$\nabla \tilde{\alpha} = \nabla \alpha \frac{e^{-\frac{\alpha}{\varepsilon}}}{\varepsilon(1 + e^{-\frac{\alpha}{\varepsilon}})^2} \quad (3.8)$$

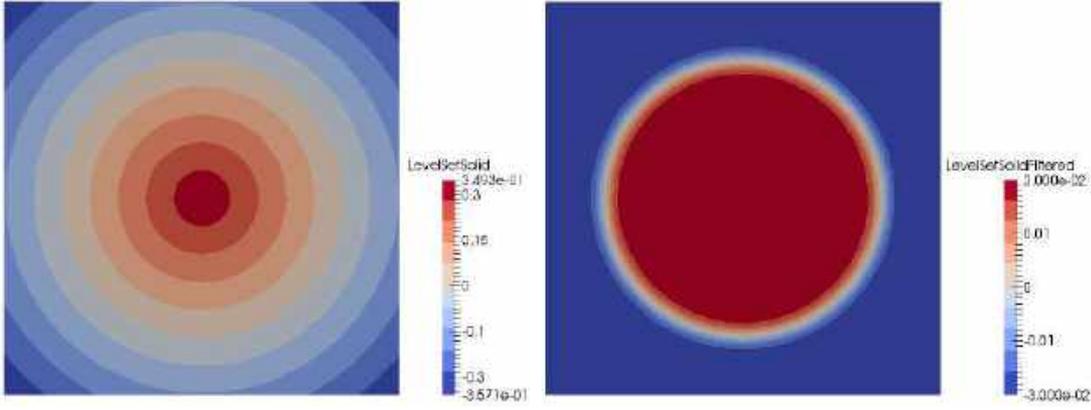


Figure 3.2: Right: basic level set function. Left: truncated level set function.

The truncated level set function now verifies the following distance property :

$$\|\nabla\alpha\| = 1 \longrightarrow \|\nabla\tilde{\alpha}\| = \frac{1}{\varepsilon}(1 - \tilde{\alpha})\tilde{\alpha} \quad (3.9)$$

As a consequence, the Hamilton-Jacobi equation 3.6 is now written as:

$$\frac{\partial\alpha}{\partial\tau} = s(\alpha)(1 - \|\nabla\alpha\|) \longrightarrow \frac{\partial\tilde{\alpha}}{\partial\tau} = s(\tilde{\alpha})\left(\frac{1}{\varepsilon}(1 - \tilde{\alpha})\tilde{\alpha} - \|\nabla\tilde{\alpha}\|\right) \quad (3.10)$$

Let us introduce by U , the re-initialization velocity :

$$U = s(\tilde{\alpha})\frac{\nabla\tilde{\alpha}}{\|\nabla\tilde{\alpha}\|} \quad (3.11)$$

Now, equation 3.10 can be written as :

$$\frac{\partial\tilde{\alpha}}{\partial\tau} + U \cdot \nabla\tilde{\alpha} = s(\tilde{\alpha})\frac{1}{\varepsilon}(1 - \tilde{\alpha})\tilde{\alpha} \quad (3.12)$$

Following the work of [35], the re-initialization step is now included in the convection step and the final one-equation model of evolution for the fluid interface can be written as :

$$\frac{\partial\tilde{\alpha}}{\partial t} + (u + \lambda U) \cdot \nabla\tilde{\alpha} = s(\tilde{\alpha})\frac{\lambda}{\varepsilon}(1 - \tilde{\alpha})\tilde{\alpha} \quad (3.13)$$

where λ is constant proportional to a velocity and ε is the thickness of the truncation. We apply then an implicit temporal discretization, thus the one-equation model in 3.13 becomes :

$$\frac{3\tilde{\alpha}^{n+1} - 4\tilde{\alpha}^n + \tilde{\alpha}^{n-1}}{2\Delta t} + (u^{n+1} + \lambda U^n) \cdot \nabla\tilde{\alpha}^{n+1} - s(\tilde{\alpha})\frac{\lambda}{\varepsilon}(1 - \tilde{\alpha}^n)\tilde{\alpha}^{n+1} = 0 \quad (3.14)$$

Finally, we introduce a stabilized finite element variational formulation to deal with the new obtained convection-reaction equation (3.14):

$$\begin{aligned} & \left(\frac{3\tilde{\alpha}_h^{n+1} - 4\tilde{\alpha}_h^n + \tilde{\alpha}_h^{n-1}}{2\Delta t}, \omega_h \right)_\Omega + ([u_h^{n+1} + \lambda U_h^n] \cdot \nabla\tilde{\alpha}_h^{n+1}, \omega_h)_\Omega - \left(s(\tilde{\alpha}_h)\frac{\lambda}{\varepsilon}(1 - \tilde{\alpha}_h^n)\tilde{\alpha}_h^{n+1}, \omega_h \right)_\Omega \\ & + \sum_K (\mathcal{R}(\tilde{\alpha}_h^{n+1}), \tau^n [u_h^{n+1} + \lambda U_h^n] \cdot \nabla\omega_h)_K + \sum_K (\mathcal{R}(\tilde{\alpha}_h^{n+1}), \tau^n |s(\tilde{\alpha}_h)\frac{\lambda}{\varepsilon}(1 - \tilde{\alpha}_h^n)\tilde{\alpha}_h^{n+1}| \cdot \omega_h)_K = 0, \end{aligned} \quad (3.15)$$

with \mathcal{R} , the residual coming from equation (3.14) defined as follows :

$$\mathcal{R} = \frac{\partial \tilde{\alpha}_h}{\partial t} - (u_h + \lambda U_h \nabla \tilde{\alpha}_h) - s(\tilde{\alpha}_h) \frac{\lambda}{\varepsilon} (1 - \tilde{\alpha}_h) \tilde{\alpha}_h \quad (3.16)$$

Finally, once we have a mesh well adapted along the zero-isovalue of the level set function (see Figure 5.9), and the interface between the different fluids is well captured, we use different mixing laws in order to distribute the material properties between the different sub-domains.

Number of elements	Number of processors	Time step	Computational time
6 million	112	$1 \cdot 10^{-4} s$	2 months

Table 3.1: Pourcentage of error evolution for both level set methods

Number of elements	Number of processors	Time step	Computational time
800 000	56	$1 \cdot 10^{-3} s$	2 weeks

Table 3.2: Pourcentage of error evolution for both level set methods

Number of elements	Number of processors	Time step	Computational time
4 million	112	$1 \cdot 10^{-4} s$	1 week

Table 3.3: Pourcentage of error evolution for both level set methods

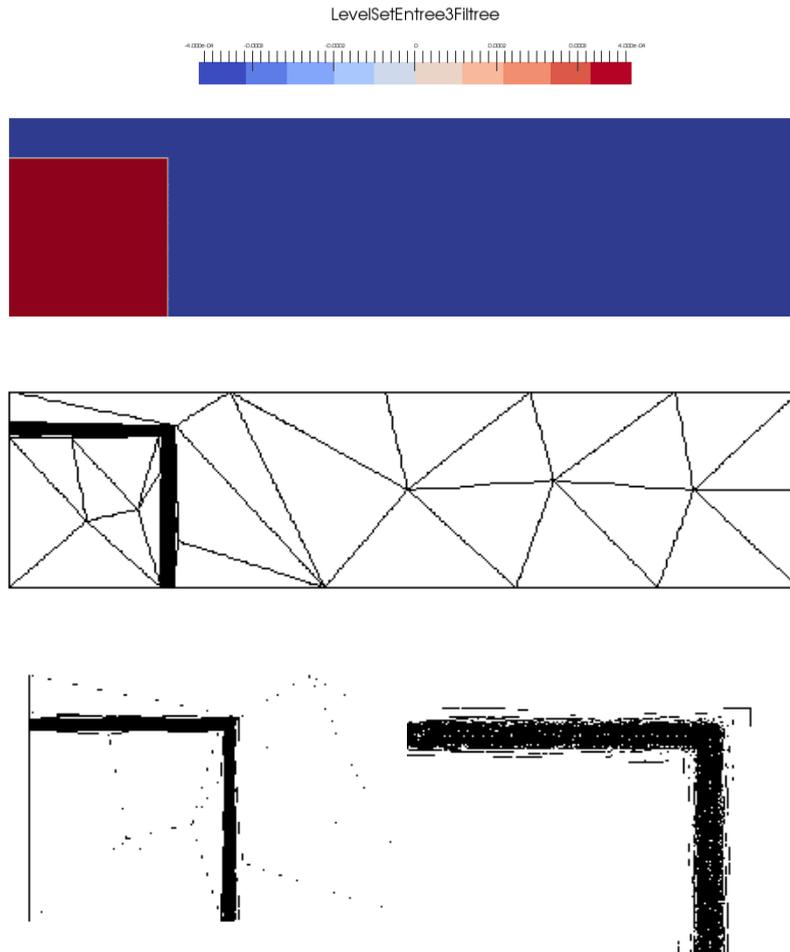


Figure 3.3: Filtered level set function and interface refinement using anisotropic mesh adaptation.

3.3 Mixing laws

As explained before, in the monolithic approach, we consider one single domain for all the different phases. In order to separate the different sub domains and to capture the different interfaces between them, we use level set functions. Now, we need to assign the physical properties corresponding to each sub domain. To do so, the material distribution between each physical domain is determined using the sign of the level set function and a Heaviside function. Therefore, the global material properties such as density ρ , dynamic viscosity μ , heat capacity c_p and the initial temperature T_0 are expressed by the following laws :

$$\begin{aligned}\rho &= \rho_1 H_e(\tilde{\alpha}) + \rho_2 (1 - H_e(\tilde{\alpha})), \\ \mu &= \mu_1 H_e(\tilde{\alpha}) + \mu_2 (1 - H_e(\tilde{\alpha})), \\ \rho c_p &= \rho c_{p1} H_e(\tilde{\alpha}) + \rho c_{p2} (1 - H_e(\tilde{\alpha})), \\ \rho c_p T_0 &= \rho c_{p1} T_1 H_e(\tilde{\alpha}) + \rho c_{p2} T_2 (1 - H_e(\tilde{\alpha})),\end{aligned}\tag{3.17}$$

As for the thermal conductivity κ , one can not use a linear distribution. In fact, Patankar [36] pointed out that it will lead to abrupt changes along the interfaces and will lead to inaccurate results. The use of a harmonic law is preferred in order to ensure the conservation of the heat flux along the interface. As a result, the appropriate expression for the global thermal conductivity is given by :

$$\kappa = \left(\frac{H_e(\tilde{\alpha})}{\kappa_1} + \frac{1 - H_e(\tilde{\alpha})}{\kappa_2} \right)^{-1}\tag{3.18}$$

The Heaviside function H_e is usually defined as :

$$H_e(\tilde{\alpha}) = \begin{cases} 0 & \text{if } \tilde{\alpha} < 0 \\ \frac{1}{2} & \text{if } \tilde{\alpha} = 0 \\ 1 & \text{if } \tilde{\alpha} > 0 \end{cases}\tag{3.19}$$

However, the classic Heaviside function (3.19) will lead to sharp variations of the material properties at the interface, and numerical solvers will suffer from stability problems. To avoid these discontinuities at the interfaces, we use a certain thickness proportional to the spatial mesh size, that will allow us to smooth the transition. Moreover, the classic Heaviside 3.19 is replaced by a smoothed function given by :

$$H_e(\tilde{\alpha}) = \begin{cases} 0 & \text{if } \tilde{\alpha} < -e \\ \frac{1}{2} \left(\frac{1}{\pi} \sin\left(\frac{\pi\tilde{\alpha}}{e}\right) + \frac{\tilde{\alpha}}{e} + 1 \right) & \text{if } |\tilde{\alpha}| \leq e \\ 1 & \text{if } \tilde{\alpha} > e \end{cases}\tag{3.20}$$

where $e = O(h_m)$ is the thickness parameter for smoothing the Heaviside function and h_m is the average mesh size in the normal direction to the interface. Another interesting Heaviside function can be found in the literature and has seemingly been considered as able to reduce spurious current at the interface, [37]. It is based on density scaling and is defined as follows :

$$H_e(\tilde{\alpha}) = \begin{cases} 0 & \text{if } \tilde{\alpha} < -e \\ \frac{1}{2} \left(\frac{1}{2} + \frac{\tilde{\alpha}}{e} + \frac{\tilde{\alpha}^2}{2e^2} - \frac{1}{4\pi^2} \left(\cos\left(\frac{2\pi\tilde{\alpha}}{e}\right) - 1 \right) + \frac{\varepsilon + \tilde{\alpha}}{e\pi} \sin\left(\frac{\pi\tilde{\alpha}}{e}\right) \right) & \text{if } \tilde{\alpha} \leq e \\ 1 & \text{if } \tilde{\alpha} > e \end{cases}\tag{3.21}$$

Note that the mesh size can be computed as follows :

$$h_m = \max_{\{i,j \in K\}} \nabla \tilde{\alpha} \cdot X^{ij} \quad (3.22)$$

where K is an element of the mesh in the close area to the interface and X^{ij} in an edge of that element. It should be underlined here that the the anisotropic mesh adaption procedure reduces the thickness of the transition, therefore, the distribution of the material properties will be more accurate.

3.4 The incompressible Navier Stokes equation

In fluid mechanics, the flow motion of imcompressible fluids is governed by the Navier Stokes equations. A flow is considered incompressible if we neglect the density changes (for instance liquids, in particular water). Over the time, the Navier Stokes equations were extraordinarily successful in obtaining quantitative understanding of shock waves, turbulence, aerodynamics, microfluids, etc. The solution of the equations is a flow velocity and a pressure field. To every point in a fluid, at any moment in a time interval, it gives a vector whose direction and magnitude are those of the velocity of the fluid, and a pressure field at that point in space and at that moment in time. Very few analytical solutions based on model simplifications were introduced in the literature. However, these simplifications lead to non-physical solutions. Several attempts to solve these equations numerically were addressed in the literature. In this section, we will present the finite element implementation of the unsteady Navier-Stokes equations. We we will start by introducing the basic definition of the Navier Stokes equations, then give details on the weak formulation and finally the variational multiscale approximation used to overcome the numerical problems that are encountered when using a Galerkin formulation.

3.4.1 Governing equations

The numerical solution of the unsteady incompressible Navier Stokes equations require discretization in both space and time. Let us define by $\Omega \subset \mathcal{R}^d$ the spatial domain (d being the space dimension). The motion of a Newtonian fluid can be described by the velocity $u(x, t)$ and the pressure $p(x, t)$. We need to find the pair velocity/pressure such that :

$$\begin{cases} \rho \partial_t u + \rho u \cdot \nabla u - \nabla \cdot \sigma = f & \text{in } \Omega \times [0, T] \\ \nabla \cdot u = 0 & \text{in } \Omega \times [0, T] \end{cases} \quad (3.23)$$

where $t \in [0, T]$ is an interval of time, ρ the density, f a given vector of external forces acting on the flow and σ the Cauchy stress tensor. The conservation of mass is modeled by the first equation. And the second one, called continuity equation reflects the incompressibility of the fluid. The stress tensor σ associated with a Newtonian fluid is defined by the following constitutive law:

$$\sigma = 2\mu \epsilon(u) - p I_d, \quad (3.24)$$

where μ is the dynamic viscosity, I_d the identity tensor and $\epsilon(u)$ the strain rate tensor, defining the symmetric part of the velocity gradient :

$$\epsilon(u) = \frac{1}{2}(\nabla u + {}^t\nabla u), \quad (3.25)$$

Combining the Navier Stokes equations (3.23), the constitutive law for the stress tensor (3.24) and the expression of the strain rate tensor (3.25), we obtain the following momentum conservation equation :

$$\begin{cases} \rho (\partial_t u + u \cdot \nabla u) - 2\mu \nabla \cdot \epsilon(u) + \nabla p = f & \text{in } \Omega \times [0, T] \\ \nabla \cdot u = 0 & \text{in } \Omega \times [0, T] \end{cases} \quad (3.26)$$

In order to obtain a well-posed problem, the initial condition on velocity at $t = 0$ must satisfy a divergence free condition ($\nabla \cdot u_0 = 0$) and has the following form :

$$u = u_0 \quad \text{in } \Omega \times (0, 0) \quad (3.27)$$

On the opposite, for incompressible flows, there is no need to define an initial condition on the pressure. Let $\Gamma = \partial\Omega$ denotes the boundary of Ω . And let us define by Γ_D and Γ_N , two complementary subset of the domain boundary Γ , where Dirichlet and Neumann boundary conditions will be imposed. The set of boundary conditions is expressed as follows :

$$\begin{cases} u = u_D & \text{on } \Gamma_D \times (0, T) \\ u \cdot n = h_N & \text{on } \Gamma_N \times (0, T) \end{cases} \quad (3.28)$$

where n is the unit outward normal vector to Γ_N .

3.4.2 Galerkin finite element formulation

In this work, a finite element method (FEM) approach has been used to solve the Navier Stokes equations. To derive the variational formulation of the problem (3.26,3.27,3.28), let us first define the function spaces for the velocity, the scalar function space for the pressure, and the weighting function space for the velocity :

$$\mathcal{U} = \left\{ u(x, t) \mid u(x, t) \in (H^1(\Omega))^d, u = u_D \text{ on } \Gamma_D \right\},$$

$$\mathcal{P} = \left\{ p(x, t) \mid p(x, t) \in L^2(\Omega) \right\}, \text{ and}$$

$$\mathcal{U}_0 = \left\{ u(x, t) \mid u(x, t) \in (H^1(\Omega))^d, u = 0 \text{ on } \Gamma_D \right\}$$

Let us define by $(w, q) \in (\mathcal{U}_0, \mathcal{P})$, two test functions and let us introduce the following integral notation :

$$(a, b)_\Omega := \int_\Omega ab dx$$

After applying the Green theorem, we can derive the weak formulation of equation 3.26:

$$\begin{cases} \text{Find } (u, p) \in \mathcal{U} \times \mathcal{P} \text{ such that:} \\ (\rho \partial_t u, w) + (\rho u \cdot \nabla u, w)_\Omega + (2\mu \epsilon(u) : \epsilon(w))_\Omega - (p, \nabla \cdot w)_\Omega = (f, w)_\Omega + (h_N, w)_{\Gamma_N}, \quad \forall w \in \mathcal{U}_0 \\ (\nabla \cdot u, q)_\Omega = 0, \quad \forall q \in \mathcal{P}. \end{cases} \quad (3.29)$$

Note that the Neumann boundary term appears naturally in the formulation since we integrated by part the viscous and pressure terms of the first equation of 3.26.

For the discretization, we use a classic Galerkin approach. Let us consider the finite element partition \mathcal{T}_h of the domain Ω subdivided into simplex elements K . We use triangular elements, in case $\Omega \subset \mathcal{R}^2$ and tetrahedral ones for $\Omega \subset \mathcal{R}^3$. Using the partition \mathcal{T}_h , the functional spaces defined previously can be approximated to finite dimensional spaces :

$$\begin{aligned}\mathcal{U}_h &= \left\{ u_h \mid u_h \in (C^0(\Omega))^d, u_{h|K} \in P^1(K)^d, \forall K \in \mathcal{T}_h \right\}, \\ \mathcal{P}_h &= \left\{ p_h \mid p_h \in C^0(\omega), p_{h|K} \in P^1(K), \forall K \in \mathcal{T}_h \right\}, \text{ and} \\ \mathcal{U}_{0,h} &= \left\{ w_h \mid w_h \in \mathcal{U}_h, w_{h|\Gamma} = 0 \right\}\end{aligned}$$

The Galerkin discrete problem associated with the finite element approximation consists now in solving the following mixed problem :

$$\left\{ \begin{array}{l} \text{Find } (u_h, p_h) \in (\mathcal{U}_h, \mathcal{P}_h) \text{ such that:} \\ (\rho \partial_t u_h, w_h)_\Omega + (\rho u_h \cdot \nabla u_h, w_h)_\Omega + (2\mu \epsilon(u_h) : \epsilon(w_h))_\Omega \\ \quad - (p_h, \nabla w_h)_\Omega = (f, w_h)_\Omega + (h_N, w_h)_{\Gamma_N}, \quad \forall w_h \in \mathcal{U}_{0,h} \\ (\nabla \cdot u_h, q_h)_\Omega = 0, \quad \forall q_h \in \mathcal{P}_h. \end{array} \right. \quad (3.30)$$

It is well known that simulations using the variational formulation in 3.30 may suffer from spurious oscillations arising from two main sources. The first is due to the dominance of the non-linear convective term of the equations, oscillations may contaminate the velocity field for high advection problems, or, for high Reynolds number. The second source has to do with the mixed formulation character of the equations, which limits the choice of combinations of the finite element interpolations used to approximate the velocity and pressure fields. To obtain stable finite elements solutions of the Navier Stokes problem, the velocity and the pressure field must satisfy a compatibility condition (or inf-sup, or BabuSka-Brezzi [38; 39]). However, the standard Galerkin method uses P1/P1 elements (i.e. the same piecewise linear space for \mathcal{U}_h and \mathcal{P}_h), and this may yield undesirable pathologies in the approximation of pressure and velocity. For this reason a stabilization algorithm must be used in order to guarantee the convergence of the solution.

3.4.3 Variational multiscale (VMS) approximation

To deal with the instabilities coming from the advection-dominated regime and the velocity/pressure compatibility condition, many alternatives to the standard Galerkin finite element method can be found in the literature. Arnold and al, [40] first proposed the concept of mini-element. The idea is to enrich the functional space with space bubble functions that will vanish on each element boundary, thus, they are statically condensed and allow to have a stabilized formulation for equal order linear element. However, [41] pointed out that while this method gives acceptable results in diffusion dominant regimes, in the case of high convection terms, the performance of the method is reduced and require additional stabilization. To overcome this issue, [41; 42] proposed a selection of the optimal bubble: the bubble will have different shapes on the diffusive dominated regime than the advection-dominated flow regime. The popular method streamline-upwind/Petrov-Galerkin (SUPG) was proposed in [43; 44]. To minimize the difficulties associated with numerical simulation of incompressible flows, it was proposed to add stabilization terms in the Galerkin formulation, the basic idea is to modify the test functions by adding weighted residuals to the variational formulation of the problem.

Such schemes, which introduce minimal numerical diffusion, have been successfully applied to various fluid dynamics problems. The stabilized Finite Element methods are very promising and considered to be an open door to turbulence. Codina and others, [45–48] introduced lately recent developments of residual based stabilization methods using orthogonal subscales and time dependent subscales. Finally, multiscale approaches were proposed by Hughes and al, [49–51]. While, stabilized finite element methods are centered on the modification of the variational formulation, the Variational Multiscale method modify the finite element basis. However, it is shown in the literature, that they address the same shortcomings of the classical Galerkin method : they add weighted residual terms to the standard weak formulation of the problem to enhance its stability while maintaining consistency.

In this thesis, the Variational Multiscale approach will be used for avoiding the spurious solutions for convection dominated regimes. While satisfying the inf-sup condition, the VMS method will enrich the functions spaces. Moreover, it has been proofed to be equivalent to LES methods for the modeling of turbulence effects. The effect of the small scale turbulence is modeled implicitly, without the need of its explicit resolution. The VMS approach provides a natural stabilization by an orthogonal decomposition of the solution (velocity and pressure) spaces. The basic idea is to consider that the unknowns can be split into two components, a coarse one and a fine one, corresponding to different scales or levels of resolution. First, we solve the fine scales in an approximate manner and then we replace their effect into the large-scale equation. We present here only an outline of the method, and the reader is referred to [52] for extensive details about the formulation. First of all, we apply an orthogonal decomposition of the functional spaces for the velocity, the pressure and the weighting function. The enrichment of the functional spaces is performed as follows:

$$\begin{aligned}\mathcal{U} &= \mathcal{U}_h \oplus \mathcal{U}', \\ \mathcal{P} &= \mathcal{P}_h \oplus \mathcal{P}', \text{ and} \\ \mathcal{U}_0 &= \mathcal{U}_{0,h} \oplus \mathcal{U}'_0\end{aligned}$$

Let us split the velocity and the pressure fields into resolvable coarse-scale and unresolved fine-scale components:

$$\begin{aligned}u &= u_h + u', \\ p &= p_h + p'\end{aligned}$$

In the same manner, we obtain the following weighting functions:

$$\begin{aligned}w &= w_h + w', \\ q &= q_h + q'\end{aligned}$$

Subscript h is used to denote the finite element (coarse) component, whereas the prime is used for the so called scale (fine) component of the unknowns. Now, the mixed-finite element approximation of problem 3.26 can be written as :

$$\left\{ \begin{array}{l} \text{Find } (u, p) \in \mathcal{U} \times \mathcal{P} \text{ such that:} \\ (\rho \partial_t (u_h + u'), (w_h + w'))_{\Omega} + (\rho (u_h + u') \cdot \nabla (u_h + u'), (w_h + w'))_{\Omega} \\ \quad + (2\mu \varepsilon (u_h + u') : \varepsilon (w_h + w'))_{\Omega} \\ \quad - ((p_h + p'), \nabla \cdot (w_h + w'))_{\Omega} = (f, (w_h + w'))_{\Omega} + (h_N, (w_h + w'))_{\Gamma_N}, \quad \forall w \in \mathcal{U}_0 \\ (\nabla \cdot (u_h + u'), (q_h + q'))_{\Omega} = 0, \quad \forall q \in \mathcal{P}. \end{array} \right. \quad (3.31)$$

To derive the stabilized formulation, we split equation 3.31 into two sub-problems, namely the coarse-scale problem and the fine-scale problem .

$$\left\{ \begin{array}{l} \mathbf{Coarse-scale\ problem} \\ (\rho \partial_t (u_h + u'), w_h)_\Omega + (\rho (u_h + u') \cdot \nabla (u_h + u'), w_h)_\Omega \\ \quad + (2\mu \varepsilon(u_h) : \varepsilon(w_h))_\Omega \\ \quad - ((p_h + p'), \nabla \cdot w_h)_\Omega = (f, w_h)_\Omega + (h_N, w_h)_{\Gamma_N}, \quad \forall w \in \mathcal{U}_0 \\ (\nabla \cdot (u_h + u'), q_h)_\Omega = 0, \quad \forall q \in \mathcal{P}. \end{array} \right. \quad (3.32)$$

$$\left\{ \begin{array}{l} \mathbf{Fine-scale\ problem} \\ (\rho \partial_t (u_h + u'), w')_\Omega + (\rho (u_h + u') \cdot \nabla (u_h + u'), w')_\Omega \\ \quad + (2\mu \varepsilon(u') : \varepsilon(w'))_\Omega \\ \quad - ((p_h + p'), \nabla \cdot w')_\Omega = (f, w')_\Omega + (h_N, w')_{\Gamma_N}, \quad \forall w \in \mathcal{U}_0 \\ (\nabla \cdot (u_h + u'), q')_\Omega = 0, \quad \forall q \in \mathcal{P}. \end{array} \right. \quad (3.33)$$

To get the stabilized formulation, we begin by approximating the fine scale problem (3.32), defined on the sum of element interiors and written in terms of the time-dependent large scale variables. Afterwards, we substitute the fine scale solution back into the coarse problem (3.33), by that, the fine-scales will no longer appear but we shall still model their effects. To do so, let start by rearranging the terms of equation (3.33) :

$$\left\{ \begin{array}{l} (\rho \partial_t u', w')_\Omega + (\rho (u_h + u') \cdot \nabla u', w')_\Omega \\ \quad + (2\mu \varepsilon(u') : \varepsilon(w'))_\Omega \\ \quad + (\nabla p', w')_\Omega = (\mathcal{R}_M, w')_\Omega, \quad \forall w \in \mathcal{U}_0 \\ (\nabla \cdot u', q')_\Omega = (\mathcal{R}_C, q')_\Omega \quad \forall q \in \mathcal{P}. \end{array} \right. \quad (3.34)$$

with \mathcal{R}_M and \mathcal{R}_C the momentum and continuity residuals , respectively :

$$\begin{aligned} \mathcal{R}_M &= f - \rho \partial_t u_h - \rho u_h \cdot \nabla u_h - \nabla p_h \\ \mathcal{R}_C &= -\nabla \cdot u_h \end{aligned} \quad (3.35)$$

Here, following the work of [52], three assumptions regarding the time-dependency and the non-linear terms in equations (3.34) must be made :

- The subscales are not tracked in time, hence a quasi-static approximation has been adopted in this work. However, since the subscale equation is driven by large-scale time-dependent residual, it remains quasi time-dependent (this choice is justified in [53]);
- As the large-scale gradients are dominant compared to the small scales, the convective velocity of the non-linear term may be approximated using only large-scale part so that $(u_h + u') \cdot \nabla (u_h + u') \approx u_h \cdot \nabla (u_h + u')$;
- As the fine-scale space is assumed to be orthogonal to the finite element space, the crossed viscous terms vanish in 3.32 and 3.33, [54];

Therefore, the fine-scale problem reduces to the following :

$$\begin{cases} (\rho u_h \cdot \nabla u', w')_{\Omega} + (2\mu \varepsilon(u') : \varepsilon(w'))_{\Omega} \\ \quad + (\nabla p', w')_{\Omega} = (\mathcal{R}_M, w')_{\Omega}, \quad \forall w \in \mathcal{U}_0 \\ (\nabla \cdot u', q')_{\Omega} = (\mathcal{R}_C, q')_{\Omega} \quad \forall q \in \mathcal{P}. \end{cases} \quad (3.36)$$

However, solving the small-scale equation (3.36) is complicated. So, rather than solving it, the subscales are approximated within each element $K \in \mathcal{T}_h$ using the residual-based terms and stabilization parameters (τ_u and τ_p) as follows :

$$\begin{aligned} u' &= \tau_u \mathcal{R}_M \\ p' &= \tau_p \mathcal{R}_C \end{aligned} \quad (3.37)$$

Several definitions are given in the literature for the stabilization parameters, for instance, Codina in [45] proposed the following stabilizing coefficients:

$$\begin{aligned} \tau_u &= \left[\left(\frac{c_1 \mu}{\rho h_m^2} \right)^2 + \left(\frac{c_2 \rho \|u_h\|_K}{h_m} \right)^2 \right]^{-\frac{1}{2}} \\ \tau_p &= \left[\left(\frac{\mu}{\rho} \right)^2 + \left(\frac{c_2 \rho \|u_h\|_K}{c_1 h_m} \right)^2 \right]^{\frac{1}{2}} \end{aligned} \quad (3.38)$$

where h_m is the characteristic length of the element K , $\|u_h\|_K$ a characteristic norm of u_h in K and c_1, c_2 two constants independent from h_m . We take them as $c_1 = 4$ and $c_2 = 2$ for linear elements (see [48]). Another possible way is to include the time step size of the temporal discretization in the expression of τ_u . This improves the convergence behavior of the algorithm to deal with the nonlinearity of the problem :

$$\tau_u = \left[\left(\frac{2\rho}{\Delta t} \right)^2 + \left(\frac{c_1 \mu}{\rho h_m^2} \right)^2 + \left(\frac{c_2 \rho \|u_h\|_K}{h_m} \right)^2 \right]^{-\frac{1}{2}} \quad (3.39)$$

Now, let us go back to the coarse problem in (3.32). Taking into account the quasi-static assumption on the fine-scales, and integrating by parts the third term of the first equation and the second term in the second equation of (3.33) and substituting the expressions of both the fine-scale pressure and the fine-scale velocity of (3.37), the coarse-scale system becomes :

$$\begin{cases} (\rho \partial_t u_h, w_h)_{\Omega} + (\rho u_h \cdot \nabla u_h, w_h)_{\Omega} - \sum_{K \in \mathcal{T}_h} (\tau_u \mathcal{R}_M, \rho u_h \cdot \nabla w_h)_K + (2\mu \varepsilon(u_h) : \varepsilon(w_h))_{\Omega} \\ \quad - (p_h, \nabla \cdot w_h)_{\Omega} + \sum_{K \in \mathcal{T}_h} (\tau_p \mathcal{R}_C, \nabla \cdot w_h)_K = (f, w_h)_{\Omega}, \quad \forall w \in \mathcal{U}_0 \\ (\nabla \cdot u_h, q_h)_{\Omega} - \sum_{K \in \mathcal{T}_h} (\tau_u \mathcal{R}_M, \nabla q_h)_K = 0, \quad \forall q \in \mathcal{P}. \end{cases} \quad (3.40)$$

Finally, substituting the residuals by their explicit formulas onto the large-scale problem, the coarse-scale's equations are now modified and expressed only in terms of coarse scale functions :

$$\begin{cases} (\rho \partial_t u_h + u_h \cdot \nabla u_h, w_h)_{\Omega} + (2\mu \varepsilon(u_h) : \varepsilon(w_h))_{\Omega} - (p_h, \nabla \cdot w_h)_{\Omega} + (\nabla \cdot u_h q_h)_{\Omega} - (f, w_h)_{\Omega} \\ \quad + \sum_{K \in \mathcal{T}_h} \tau_u (\rho (\partial_t u_h + u_h \cdot \nabla u_h) + \nabla p_h - f, \rho u_h \nabla w_h)_K \\ \quad + \sum_{K \in \mathcal{T}_h} \tau_u (\rho (\partial_t u_h + u_h \cdot \nabla u_h) + \nabla p_h - f, \nabla q_h)_K \\ \quad + \sum_{K \in \mathcal{T}_h} \tau_p (\nabla \cdot u_h, \nabla \cdot w_h)_K = 0 \quad \forall w \in \mathcal{U}_0, \quad \forall q \in \mathcal{P} \end{cases} \quad (3.41)$$

The new problem in (3.41) can now be decomposed into four main terms: the first one is the standard Galerkin formulation, the second and the third terms represent the fine-scale velocity and the last term models the influence of the fine-scale pressure. All the added terms provide additional stabilization in convection dominated regimes and make it possible to overcome instabilities caused by incompatible approximation spaces. Equations (3.41) are discretized in time by a semi-implicit scheme. The convective term, the viscous term and the pressure term in the momentum equation, as well the divergence term in the continuity equation, are integrated implicitly through a backward Euler scheme. All other contributions (i.e. the source term and the stabilization terms) are integrated explicitly by a forward Euler scheme.

An important point to mention here is that the element characteristic length h_m significantly influences the behavior of the stabilization parameter. This parameter is involved in the stabilization terms for each element K as shown by equation (3.41). A common choice would be to simply define this measure as the minimum edge length of K or as the diameter of the element. Nevertheless, in the case of strongly anisotropic meshes with highly stretched elements, the definition of h_m is still an open problem and plays a critical role in the design of the stabilizing coefficients [45]. Indeed, when a classical characteristic length is used, an undesired behavior can be observed where one edge of the element K remains unchanged while the others are refined. An example is shown in Figure 3.4; clearly the two triangles are different, one is isotropic while the other is anisotropic, but the same characteristic length is associated to both. In this work, since we are dealing with

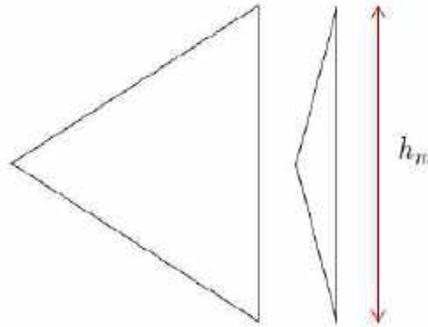


Figure 3.4: Characteristic length for isotropic and anisotropic element based on a classical formula

convection dominated problems, we adopted the definition proposed in [55] to compute h_m . It consists in computing h_m as the diameter of the element K in the direction of the velocity field u_h :

$$h_m = \frac{2|u_h|}{\sum_{i=1}^{N_K} |u_h \cdot \nabla \Phi_i|} \quad (3.42)$$

where N_K is the number of vertices of the element K and Φ_i is the shape function that is non-zero on the element K . Note that this approach takes into account the information on the projected flow along the gradient of the basis functions relative to the element.

3.5 Numerical examples

In order to validate the proposed moving interface capturing method, four time-dependent numerical test cases will be presented in this section. The first one is the famous dam

break. Then we present the test case of a 2D droplet splashing on a thin liquid. Three simulations, with different Reynolds numbers are performed. The next test case is a 2D rising bubble in a channel. Finally, we perform two simulations of the coalescence of two rising bubbles (axisymmetric and non-axisymmetric). The results obtained with the proposed approach are compared with either analytic solutions or with those obtained by other approaches that can be found in the literature.

3.5.1 2D dam break

The dam-break benchmark represents the study of the collapse of a fluid column in an air cavity [56]. It is a widely used benchmark in the field of multiphase flows. The set up of the dam break is presented in Figure 3.5. In this simulation, a column of water of size $l = 0.06$ m and $h = 0.12$ m is located in the left side of a tank ($l = 0.50$ m and $h = 0.15$ m). Water has a density of 1000 kg.m^{-3} and the air has a constant density of 1 kg.m^{-3} . A free slip boundary condition is prescribed on the vertical walls and a no-slip condition is prescribed on the top and the bottom of the domain. As mentioned earlier in this chapter, mesh adaptation can be performed using several variables all at once. Therefore, for this test case we adapt the mesh according to the level set function and the velocity field. Figure 3.6 illustrates the evolution of the free surface of the breaking dam and highlights

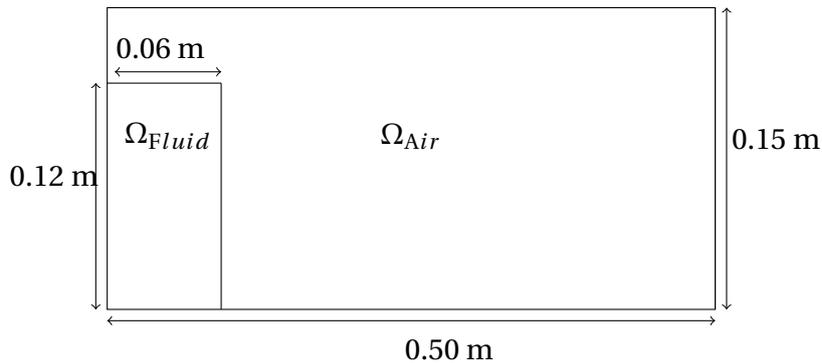


Figure 3.5: Set up of the 2D dam break

the stretching of the elements near the interface. It is clear how the automatic adaptation refines the mesh close to the interface, and in the regions where the spatial variation of velocity is maximum. It shows that the proposed framework deals naturally with complex topological changes of the interface. Recall that the mesh adaptation technique works under a fixed number of elements. When the level set surface spreads further, the elements will follow the interface, thus some elements are coarsened in some regions while others are stretched further at the interface. Anisotropic mesh adaptation generates highly stretched and well oriented elements allowing a good capture of sharp gradients. For completeness, the kinematics of the fall has been studied and compared to experimental data [56] and other numerical works [57–59]. Figure 3.7 shows the evolution of the non-dimensional front position versus the non-dimensional time, and in Figure 3.8, the height of the liquid column is plotted. To present the numerical results, the same non-dimensional units have been used in order to allow a direct comparison with the original publication of [56]. The obtained results are in good agreement with both experimental and numerical benchmarks.

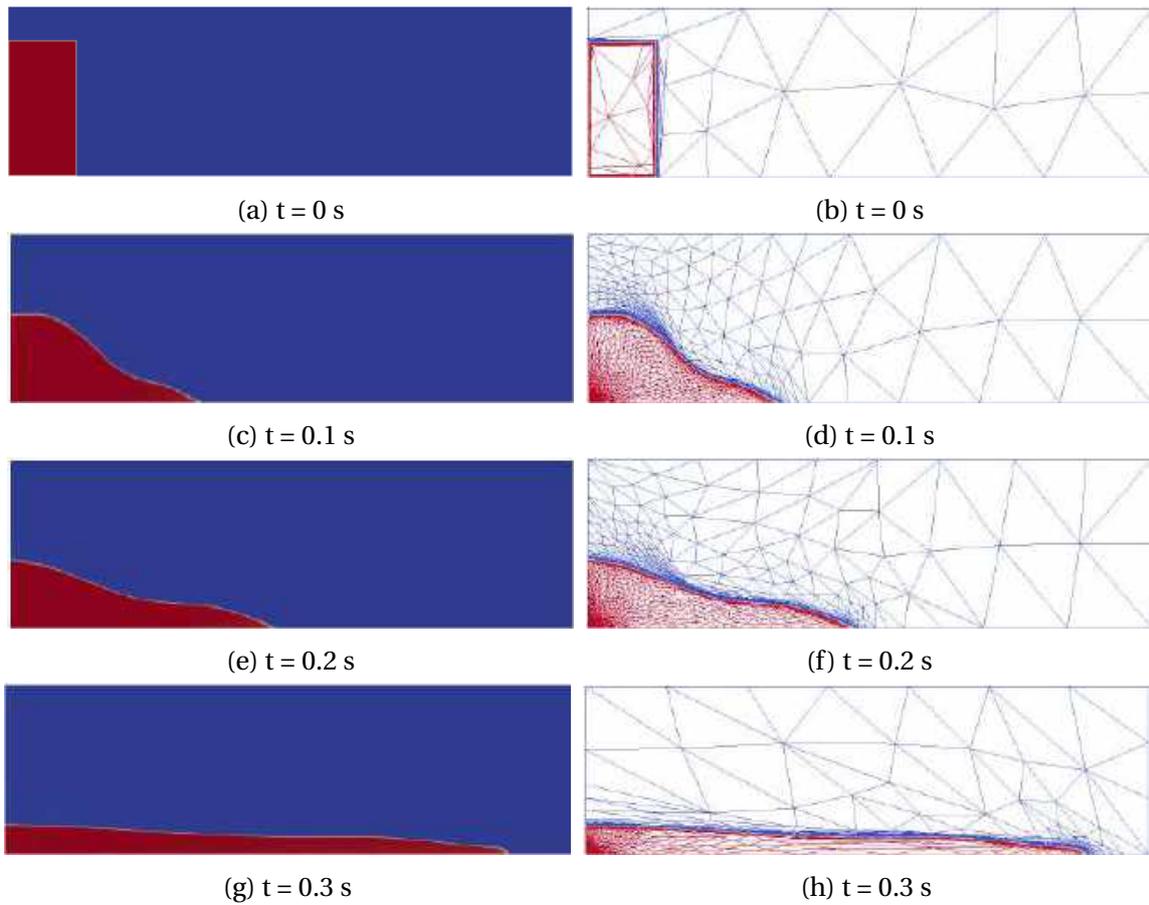


Figure 3.6: Column fall evolution and refined meshes

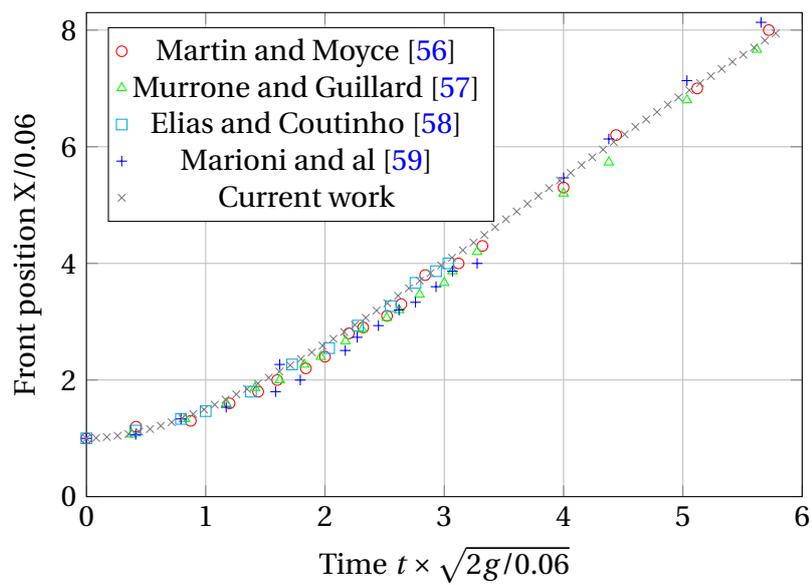


Figure 3.7: Non-dimensional front position evolution

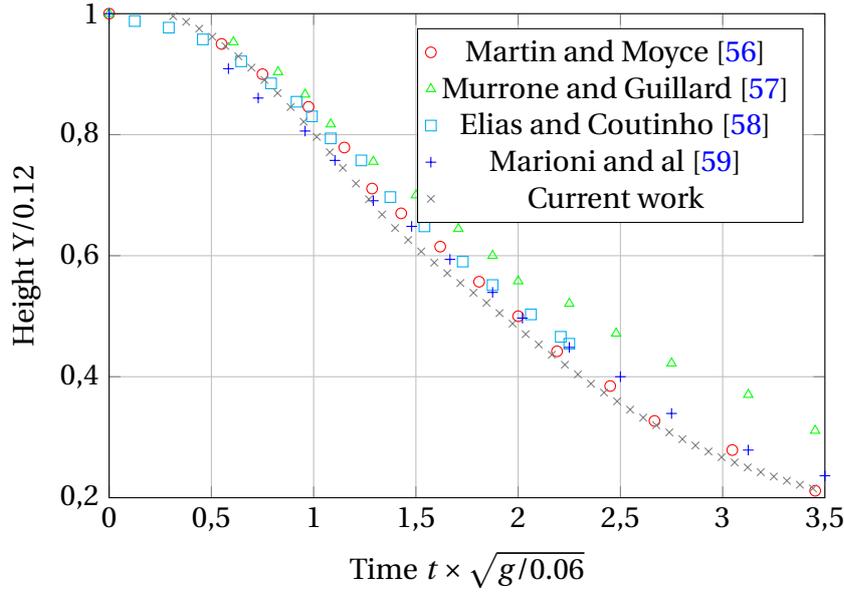


Figure 3.8: Non-dimensional column height evolution

3.5.2 2D droplet splashing on thin liquid film at different Reynolds numbers

In this section, the problem of a droplet splashing on a thin liquid film with large density ratio and high Reynolds numbers is investigated [60; 61]. The set up of the problem is shown in Figure 3.9 : a liquid droplet of diameter $d = 6 \times 10^{-3} m$ is moving downward with a velocity u to a thin liquid film of height $h = 8 \times 10^{-4} m$ in an ambient vapor field. The dimensions of the computational domain are $[0, 2.5d] \times [0, 10d]$. We denote by $(\rho_{Fluid}, \mu_{Fluid})$ and (ρ_{Air}, μ_{Air}) the density and the viscosity inside the fluid domain and the air around it. We set $\rho_{Fluid}/\rho_{Air} = 10^3$ and $\mu_{Fluid}/\mu_{Air} = 10^2$. This is a very challenging test case since the two fluids have large density and viscosity differences. Three simulations at different Reynolds numbers have been carried out; the parameters used for each one are depicted in table 3.4.

Re = 20	Re = 100	Re = 1000
We = 2000	We = 2000	We = 2000
$u = 0.003 m.s^{-1}$	$u = 0.02 m.s^{-1}$	$u = 0.2 m.s^{-1}$
$\sigma = 1,35 \times 10^{-8} N.m^{-1}$	$\sigma = 1,2 \times 10^{-6} N.m^{-1}$	$\sigma = 1,2 \times 10^{-4} N.m^{-1}$

Table 3.4: Physical parameters used for the different simulations of the 2D droplet splashing on thin liquid film

We recall that we combine the level set representation with local mesh refinement around the zero-isovalue of the level set function. Figures 3.10-3.15 show the time evolution of the droplet and the thin liquid film after the instant of impact. The interface of the droplet and the thin liquid film are perfectly represented by the dynamic mesh adaptation with multi criteria. It yields to an accurate capture of the interface between the different phases at a low computational cost. Moreover, we can clearly see the impact of using different Reynolds numbers. When the Reynolds number is low (Figures 3.10-3.11), the impact does not result in splashing, the droplet only spreads gently on the surface, it is called the deposition process. As the Reynolds number is increased to 100 (Figures 3.12-3.13), clear splashing phenomenon is observed : two liquid fingers at the end of the

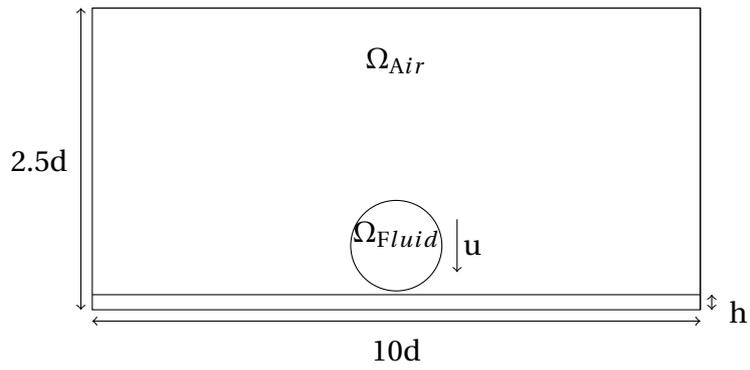


Figure 3.9: Set up of the 2D droplet splashing on thin liquid film

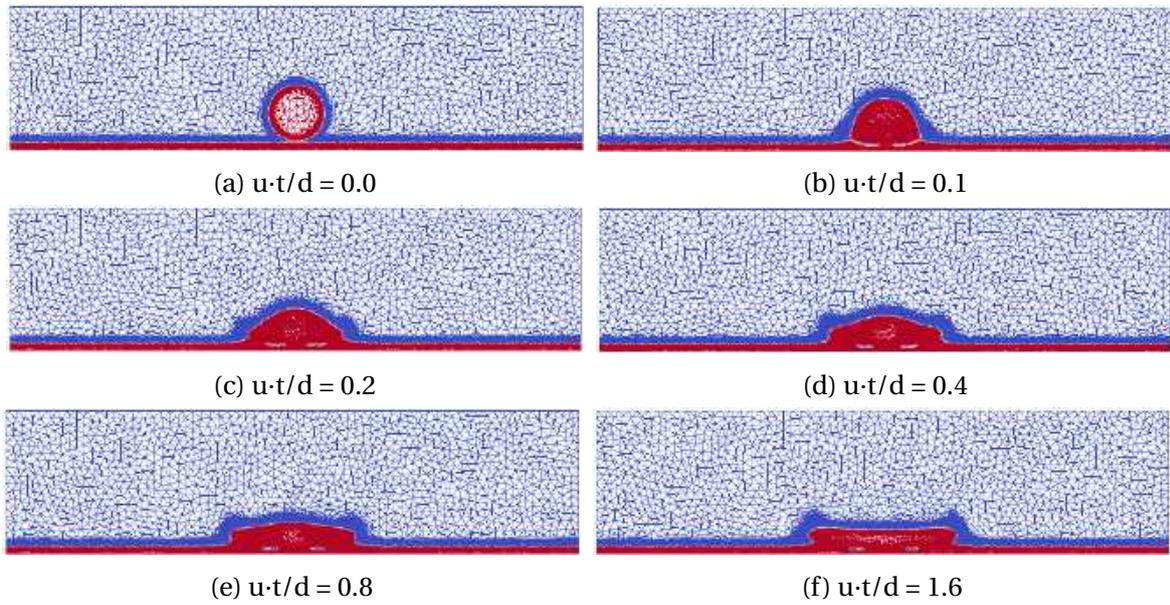


Figure 3.10: Meshes evolution for the droplet splashing on a thin film at $Re = 20$, $We = 2000$

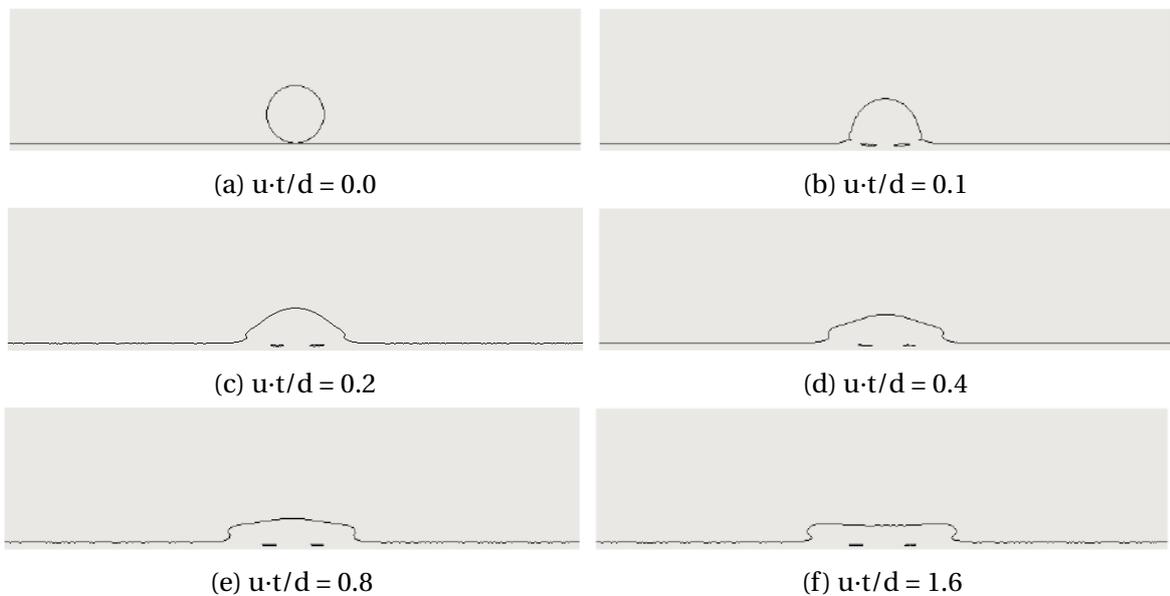


Figure 3.11: Evolution of the zero-isovalue of the level set for the droplet splashing on a thin film at $Re = 20$, $We = 2000$

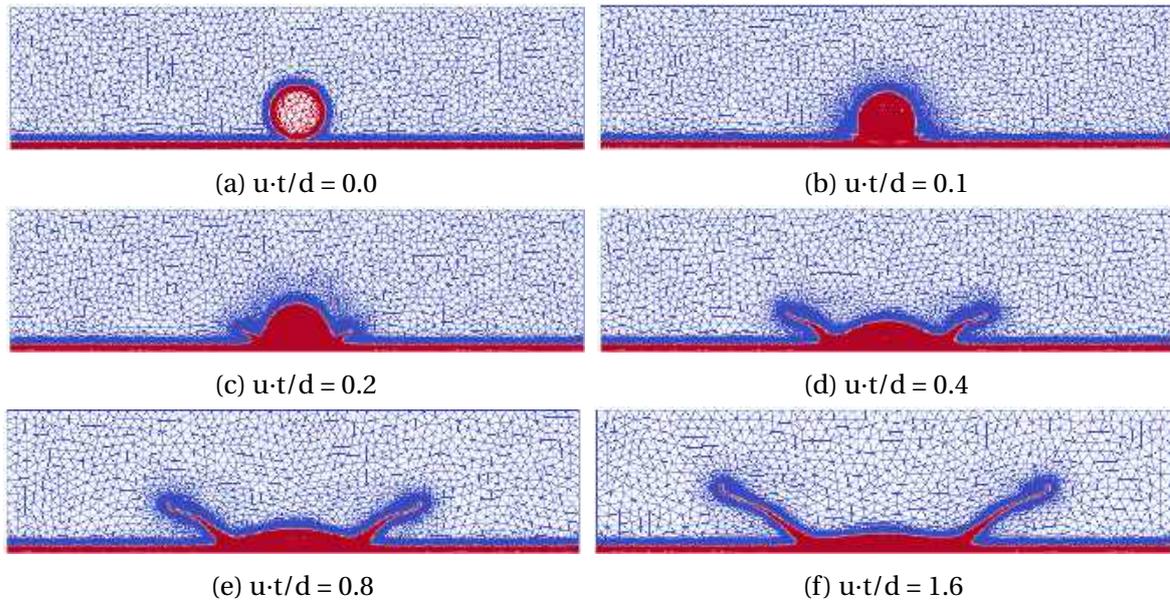


Figure 3.12: Meshes evolution for the droplet splashing on a thin film at $Re = 100$, $We = 2000$

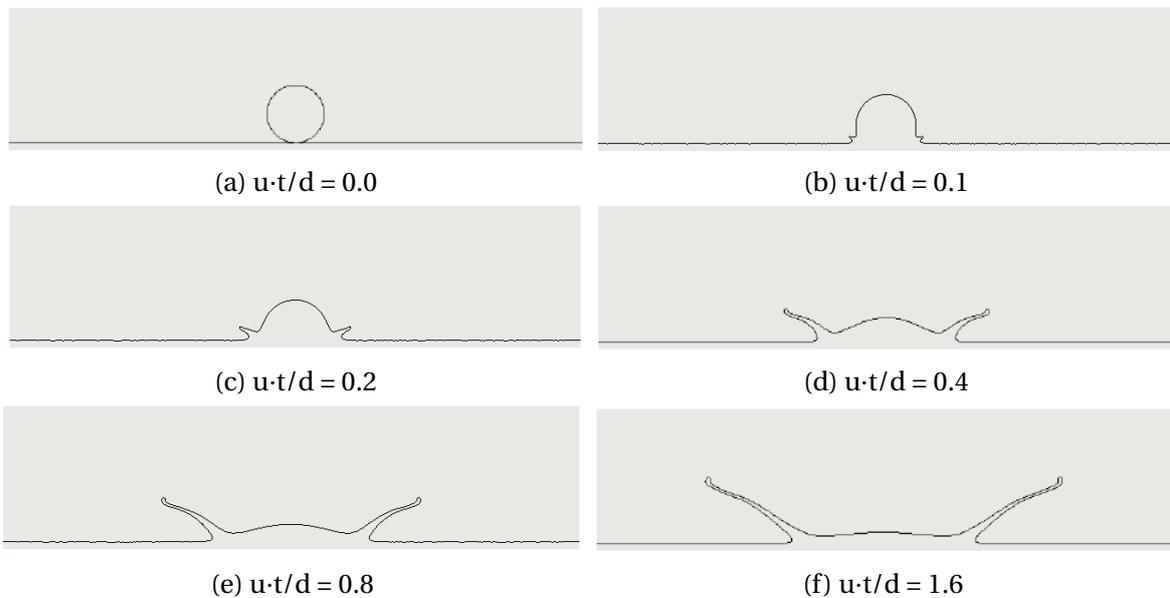


Figure 3.13: Evolution of the zero-isovalue of the level set for the droplet splashing on a thin film at $Re = 100$, $We = 2000$

rim of the splashing are generated after the impact of the droplet and are perfectly captured by the anisotropic mesh adaptation technique. Similar observations were reported in the work of [63]. As the Reynolds number increases to 1000 (Figures 3.14-3.15), the height and the speed of the splashing is higher and we can also note that the thickness of the fingers becomes smaller, which forms a thin film in the direction perpendicular to the plane. It is important to mention that in this work the simulation of the whole domain was carried out, unlike in the work of [62] where they consider half of the computational domain, see Figure 3.16. Still the results reflect the same symmetry assumed in their work.

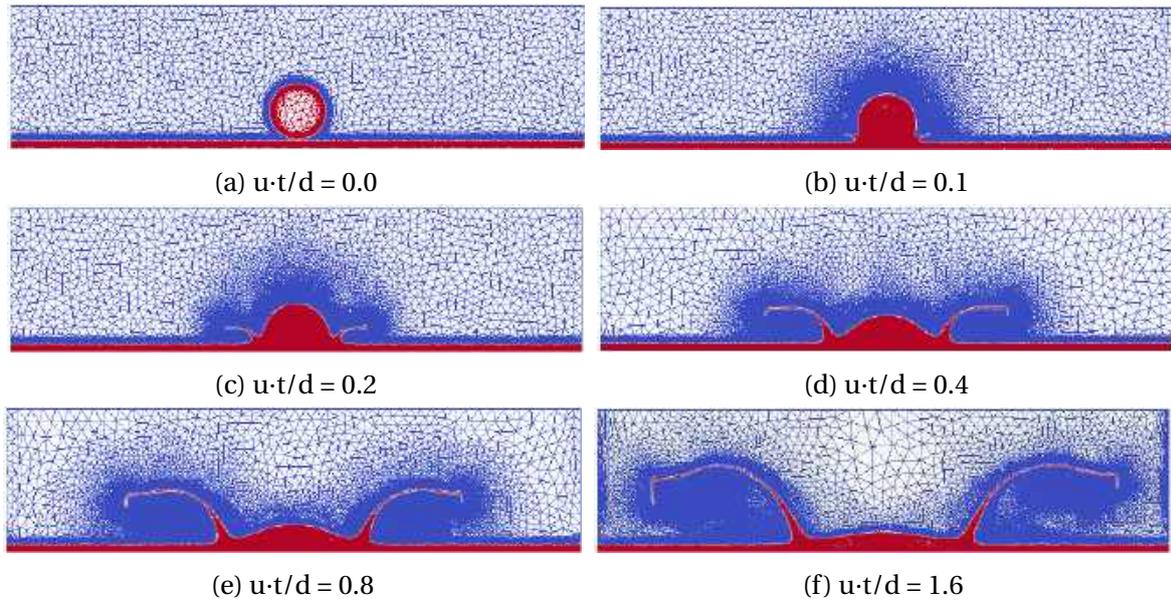


Figure 3.14: Meshes evolution for the droplet splashing on a thin film at $Re = 1000$, $We = 2000$

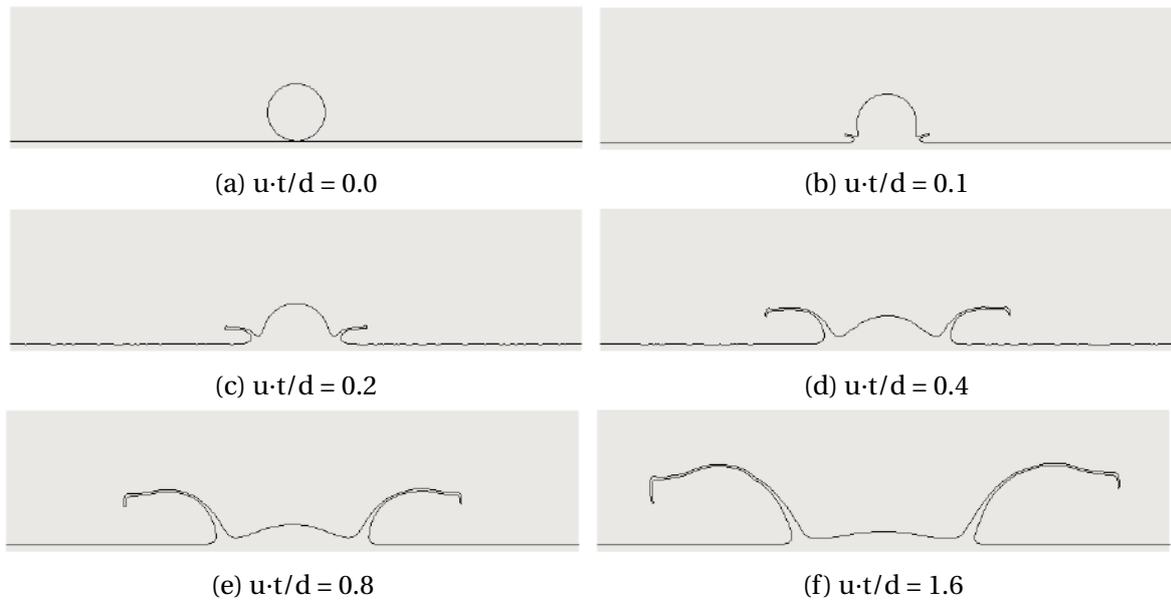


Figure 3.15: Evolution of the zero-isovalue of the level set for the droplet splashing on a thin film at $Re = 1000$, $We = 2000$

3.5.3 2D rising bubble

In this section, we conducted a 2D simulation of a single rising bubble under buoyancy force [64]. This test case has already been investigated by our research team in [65]. A rising bubble of diameter $d = 0.5 m$ is in water with a density ratio of 1000 and a viscosity ratio of 100. We use a rectangular domain of sizes $[0, 2d] \times [0, 4d]$, the set up is shown in Figure 3.17. A free slip boundary condition is prescribed on the vertical walls and a no-slip condition is prescribed on the top and the bottom of the cavity.

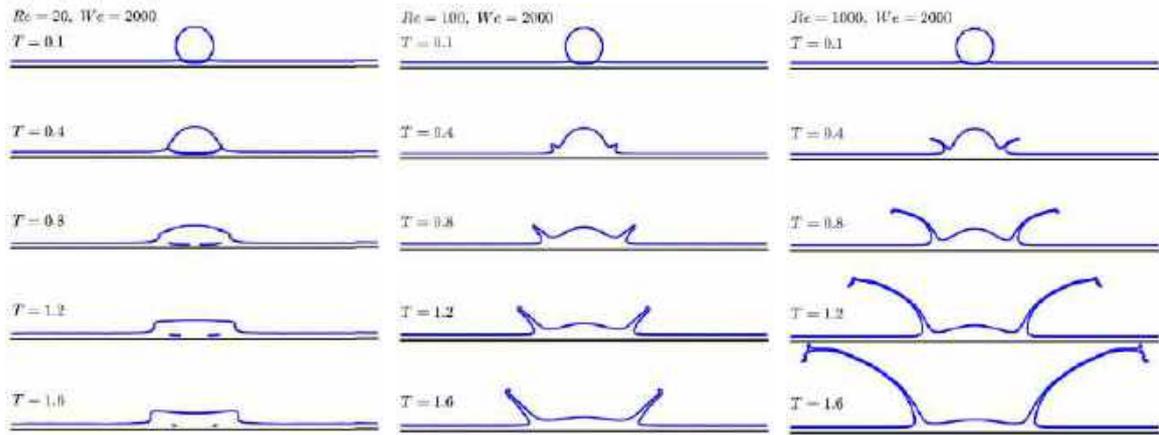


Figure 3.16: Evolution of the instantaneous interface for the droplet splashing on a thin film for the different Reynolds numbers, adopted from the work of [62]

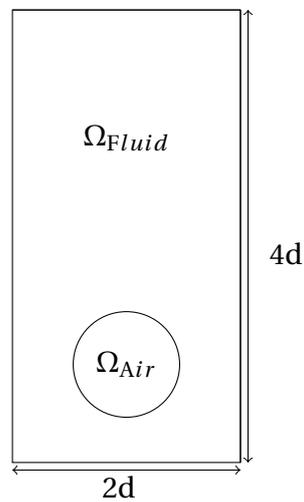
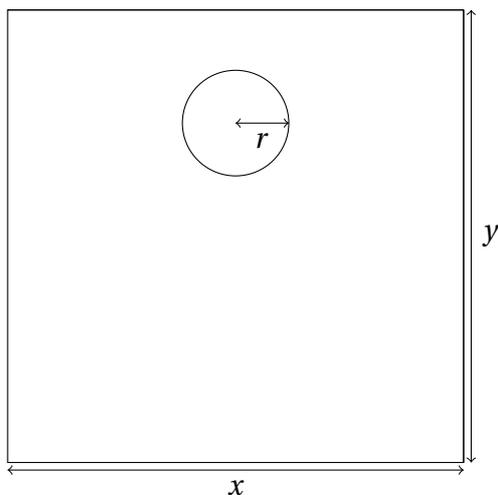


Figure 3.17: Set up of the 2D rising bubble



We performed two simulations, using adaptive anisotropic meshing, under the constraint of a fixed number of elements. The criteria chosen for the adaptive meshing is again both the velocity (in direction and magnitude) and the level set function. Figure 3.18 shows the evolution of the shape of the bubble. The high surface tension coefficient

maintains the bubble as an ellipsoid. In Figure 3.19, we can see the associated anisotropic meshes at different time steps. It shows that the finest elements are concentrated near the interface of the bubble. In this section, we want to emphasize the benefits in terms of conservation of our numerical framework and this even when dealing with both dynamic and anisotropic meshes. To this end, four simulations have been carried out, two are performed using about 5000 elements and the others were performed using about 10 000 elements, while changing the adaptation frequency (we adapt every time step or every 5 steps). The accumulation of mass loss when using mesh adaptation with linear interpolation and with conservative interpolation are depicted in tables 3.5-3.6. As expected, the tables highlight a significant difference between both approaches, confirming again the important role of the proposed global conservative framework.

Adaptation Frequency	Linear interpolation	Conservative interpolation
1	29.04 %	6.68 %
1/5	19.70 %	4.77 %

Table 3.5: Mass variation for the 2D rising bubble with 5000 elements

Adaptation Frequency	Linear interpolation	Conservative interpolation
1	19.05 %	3.86 %
1/5	16.82 %	2.37 %

Table 3.6: Mass variation for the 2D rising bubble with 10 000 elements

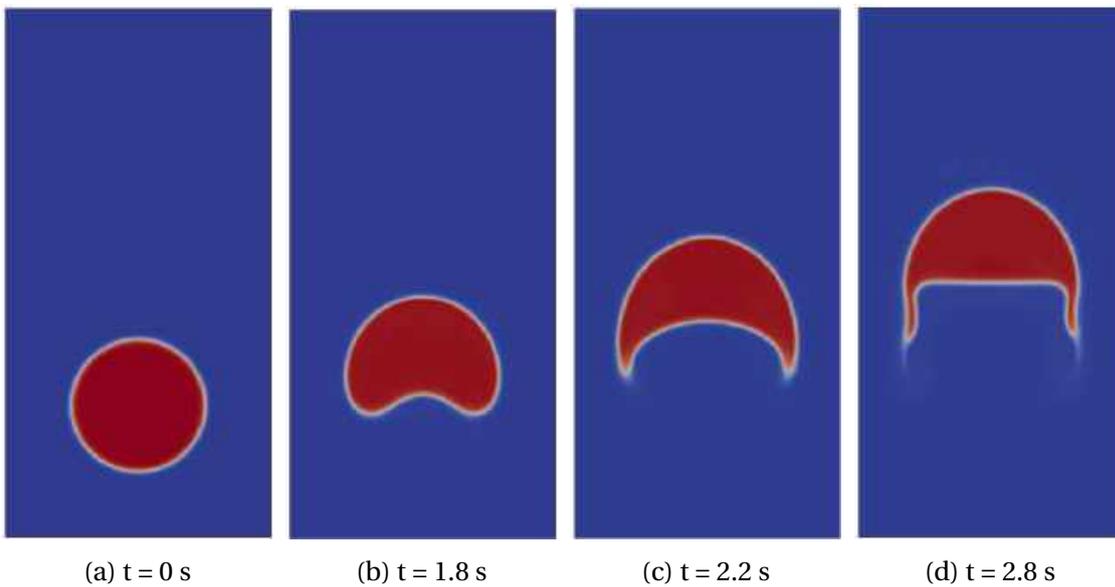


Figure 3.18: 2D rising bubble evolution

3.5.4 3D simulations of the axisymmetric and non-axisymmetric merging of two bubbles

To test high efficiency of the method for two-fluid flows involving complex topological changes of the interface, we consider the merging of two air bubbles with first co-axial

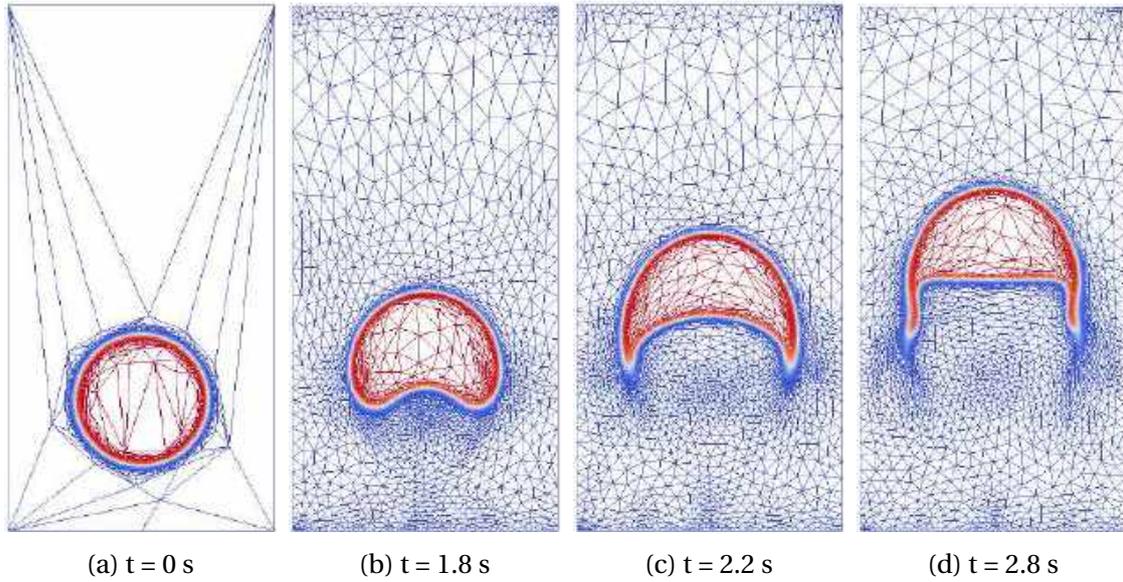


Figure 3.19: Refined meshes for the 2D rising bubble

and then oblique coalescence, the set up is shown in Figure 3.20. This test case was studied by [66; 67] and the results suggested that due to surface tension a merging will occur. We consider two gas bubbles with a diameter $d = 0.01$ m that are initially set in a quiescent liquid. The density and viscosity ratios (gas to liquid) are respectively 0.001 and 0.01. The dimensionless parameters we use for this problem are the Eotvos number $Eo = 16$ and the Morton number $M = 2 \times 10^{-4}$. The dimensions of the computational domain are $[-2d, 2d] \times [0, 8d] \times [-2d, 2d]$. For the co-axial case, the center of the upper bubble is at $[0, 2.5d, 0]$ and the lower bubble is at $[0, d, 0]$. As for the non-axisymmetric case, the lower bubble is shifted to the position $[0.8d, d, 0]$. We use free-slip boundary conditions on all sides of the domain. It can easily be observed in Figures 3.23 and 3.24 that the two bub-

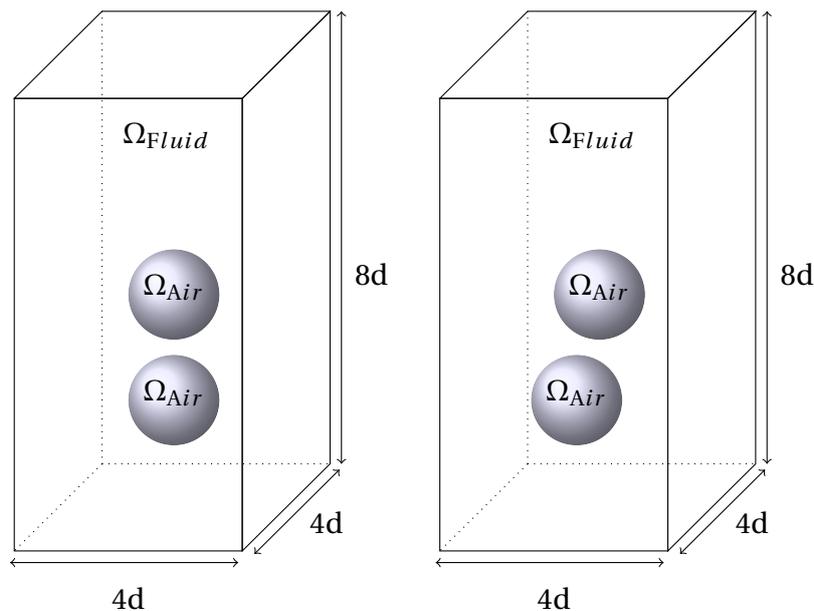


Figure 3.20: Set up of the 3D simulations of the axisymmetric and non-axisymmetric merging of two bubbles

bles gradually merge together and form a larger bubble under capillary force. For both

cases, the computed evolution of the bubble shape is well captured by the dynamic mesh adaptation. The interfaces of the two bubbles are very well captured and the boundary layers as well as the detachments are automatically detected. This shows the ability of the proposed adaptive meshing procedure to accurately solve the problem, not only by refining the mesh close to the interface but also wherever it is relevant from a physical point of view. Our results agree with the experimental photographs in [66] (see Figures 3.21, 3.22) and the numerical results of [68]. We can see that the evolution of the lower bubble is completely different from the leading one. The trailing bubble catches the leading bubble later to form one single bubble. For the non-axisymmetric case shown in Figure 3.24, the dynamics are similar to the previous one except we can note that the flow field is clearly asymmetrical.

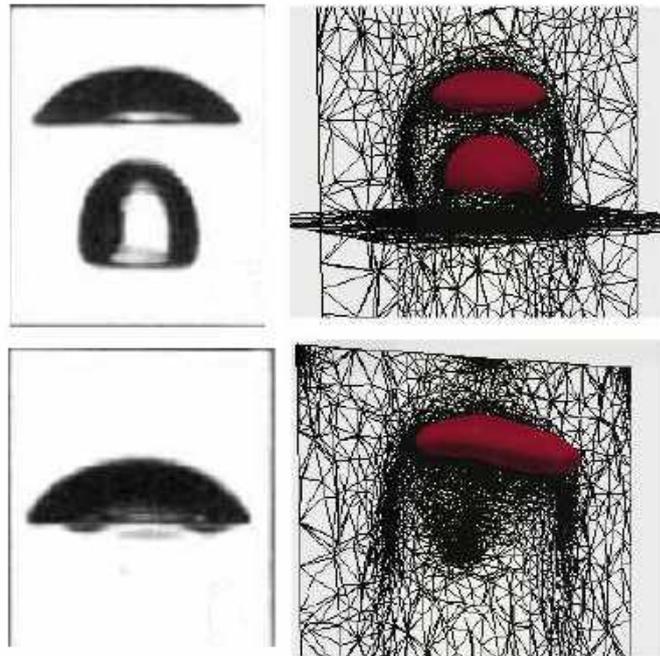


Figure 3.21: Comparison between the experimental measurement of [66] and the simulation for the co-axial coalescence of two bubbles

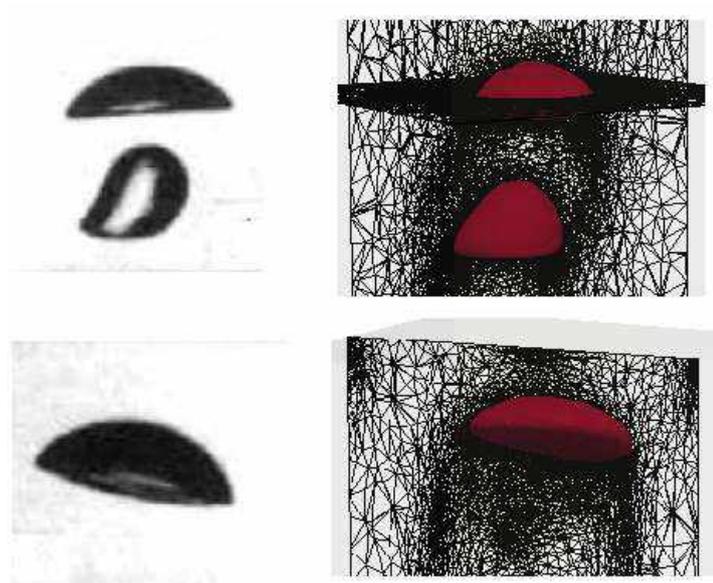


Figure 3.22: Comparison between the experimental measurement of [66] and the simulation for the oblique coalescence of two bubbles

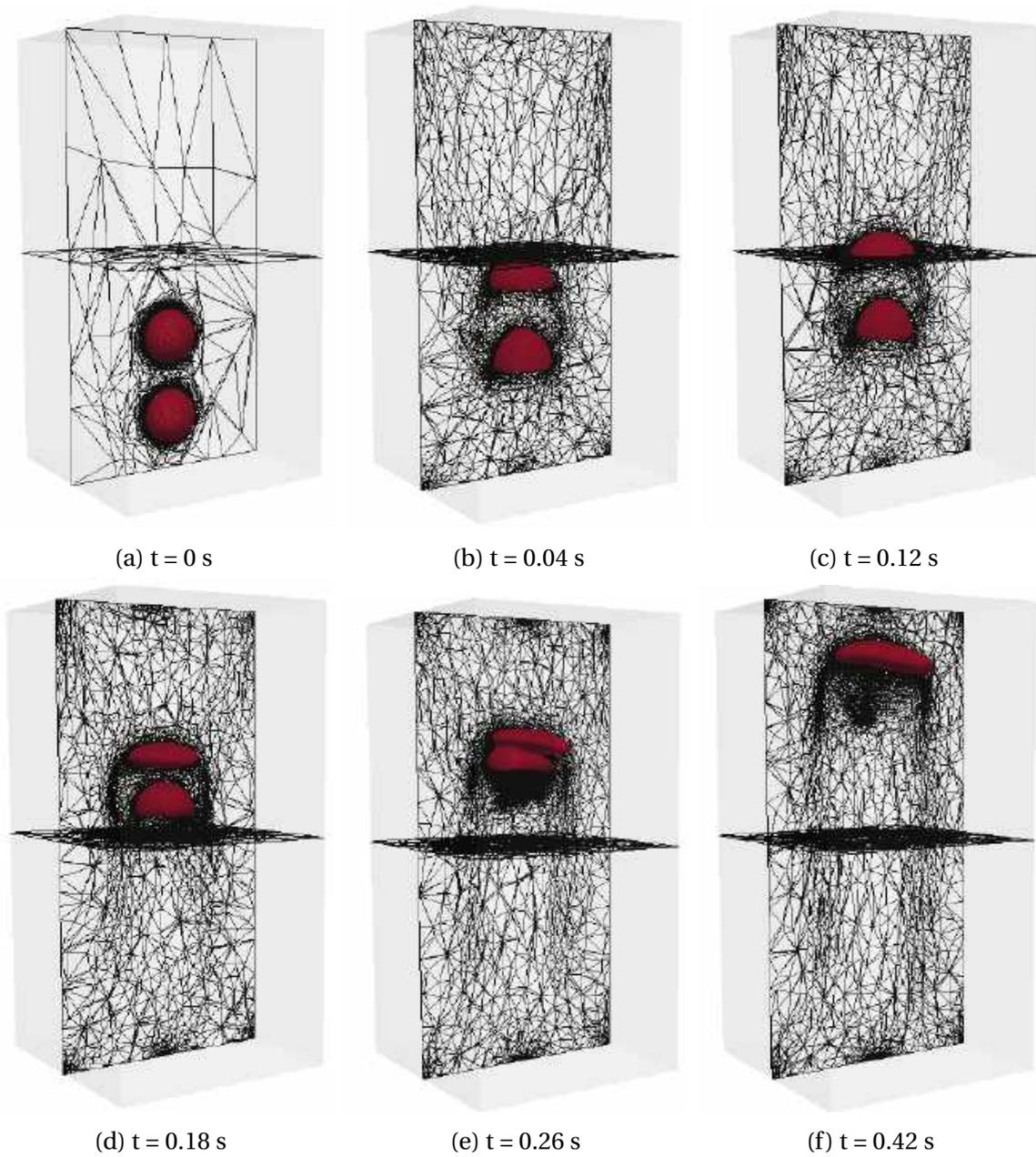


Figure 3.23: Evolution of the refined meshes the 3D bubble shape for the co-axial coalescence

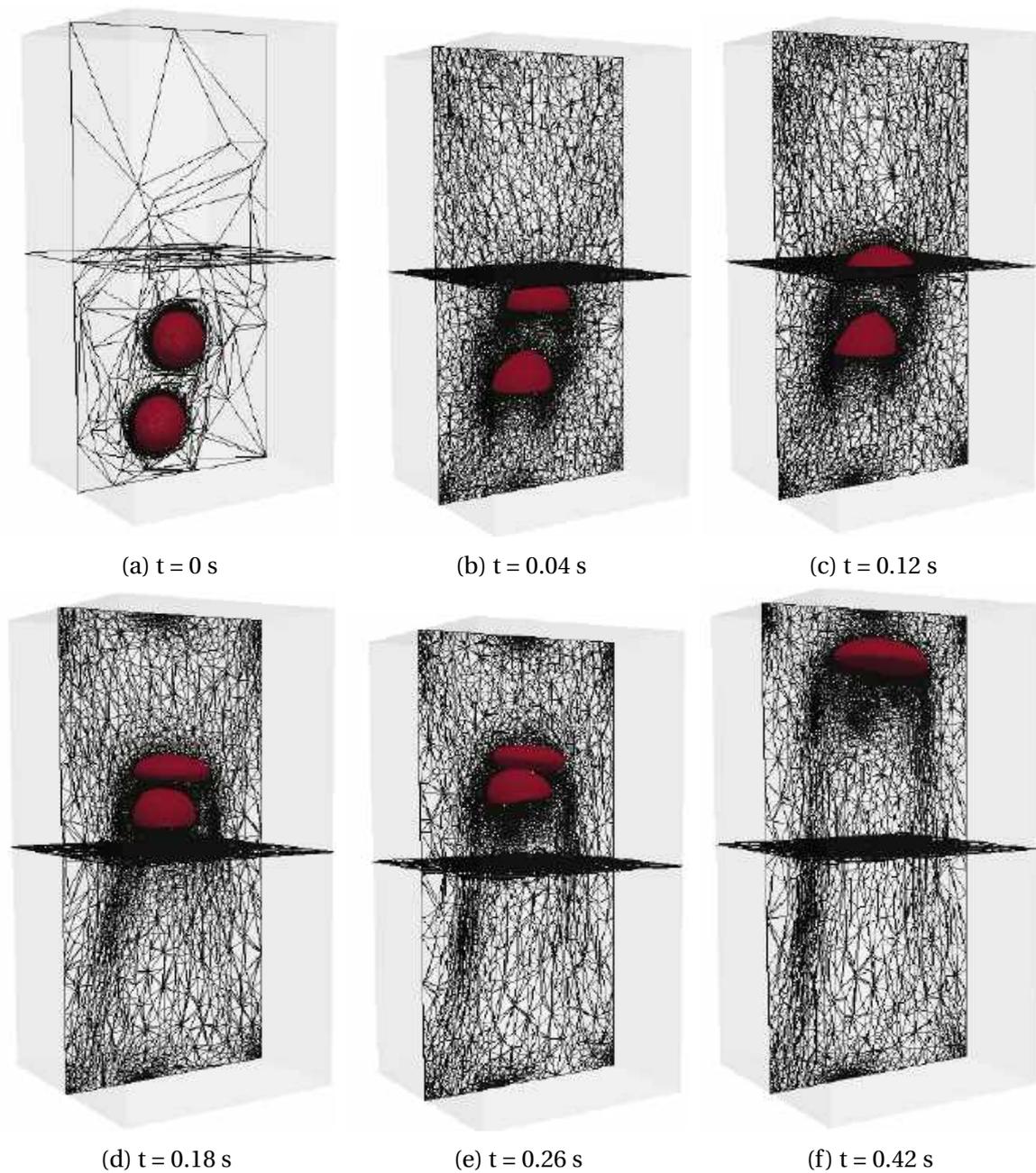


Figure 3.24: Evolution of the refined meshes the 3D bubble shape for the oblique coalescence

3.6 Conclusion

In this Chapter, we presented the method used to follow the evolution of the interfaces in multiphase applications. The interfaces between the different subdomains are rendered by the zero-isovalue of a level set function. A single mesh is constructed for both subdomains whereby a single set of equations is being solved. We introduced in this Chapter, a new definition of the level set function by adding a reaction term in the definition. These changes allow to have a more conservative level set method. Physical properties are distributed to their respective phases using mixing laws that enable a smooth transition over a small thickness at the interface. A harmonic mean formulation is applied to the conductivity to handle abrupt changes in the material properties. We have showed that the dynamic anisotropic mesh adaptation permitted a sharp capture of the interfaces and an accurate material distribution over the different subdomains. The gradient of the level set function will guide the spatial adaptation to follow the movement of the interfaces over time. Then, we have described a stabilized finite element method for the transient incompressible Navier-Stokes equations based on the decomposition of the unknowns into large scale and fine scale. The motivation of using these techniques comes from the desire of solving problems for high Reynolds numbers. The bottom line of this approach is to take into account the small-scale pressure and to add the convection terms into the fine scale equations. Finally, the performance and the efficiency of the overall moving interface capturing method have been demonstrated using four benchmarks and conclusions were drawn. These test cases are well suited for validation since both experimental data and numerical results from several authors are available. The results were in good agreement with experimental and numerical references. The results have shown a promising tool for simulating boiling and the simulation of quenching processes, and this, will be handled in the next chapters. Several test cases were performed to assess the efficiency of the proposed moving interface capturing method.

3.7 Résumé du chapitre en français

Dans ce chapitre, nous avons présenté la méthode utilisée pour suivre l'évolution des interfaces dans les applications multiphasiques. Les interfaces entre les différents sous-domaines sont représentés par l'isovaleur zéro de la fonction distance signée appelé level set. Dans ce travail, nous utilisons une approche Monolithique. C'est-à-dire que l'on considère qu'un seul domaine de calcul avec des propriétés physiques variées pour les différents sous-domaines. Un seul maillage est alors utilisé et ne nécessite pas de coïncidence avec la frontière. Dans cette thèse nous avons adopté la méthode d'immersion de volume, par laquelle les objets solides sont immergés et localisés dans la partie fluide en utilisant la fonction level set. Nous avons introduits dans ce chapitre une nouvelle définition de la fonction level set en ajoutant un terme de réaction dans la définition. Ces modifications permettent d'avoir une méthode level set plus conservative. Ensuite, les propriétés physiques et thermiques des différentes composantes sont distribuées à l'aide d'une fonction Heaviside lissée, ce qui permet une transition en douceur sur une faible épaisseur à l'interface. L'avantage de cette méthode réside dans sa capacité de s'affranchir de la nécessité de déterminer les coefficients d'échanges thermiques au niveau de l'interface. Une formulation de moyenne harmonique est appliquée à la conductivité pour traiter les changements brusques dans les propriétés du matériau. Nous avons par la suite mis en évidence, que l'adaptation de maillage anisotrope permet une nette capture des interfaces, ainsi, une bonne précision du maillage à ce niveau per-

met une bonne distribution des propriétés. Le gradient de la fonction level set guidera l'adaptation de maillage pour suivre le mouvement des interfaces dans le temps. Ensuite, nous avons décrit une méthode éléments finis stabilisés pour les équations de Navier Stokes incompressibles. Cette méthode est basée sur la décomposition des inconnues en grande et fine échelles. L'idée est de prendre en compte la pression à petite échelle et d'ajouter les termes de convection dans les équations des petites échelles. Enfin, la performance et l'efficacité de la méthode globale de capture d'interface mobile a été mis en évidence à l'aide de plusieurs cas tests et benchmarks tirés de la littérature. Ces cas tests sont bien adaptés à la validation car les données expérimentales et les résultats numériques de plusieurs auteurs sont disponibles. Les résultats ont montré un outil prometteur pour simuler l'ébullition et la simulation de procédés de trempe, et cela sera traité dans les prochains chapitres.

3.8 Bibliography

- [1] S. Elgeti, H. Sauerland, Deforming fluid domains within the finite element method: five mesh-based tracking methods in comparison, *Archives of Computational Methods in Engineering* 23 (2) (2016) 323–361. [52](#)
- [2] J. Glimm, J. W. Grove, X. L. Li, K.-m. Shyue, Y. Zeng, Q. Zhang, Three-dimensional front tracking, *SIAM Journal on Scientific Computing* 19 (3) (1998) 703–727. [52](#)
- [3] J. Glimm, J. W. Grove, X. Li, N. Zhao, Simple front tracking, *Contemporary Mathematics* 238 (2) (1999) 133–149.
- [4] D. Juric, G. Tryggvason, A front-tracking method for dendritic solidification, *Journal of computational physics* 123 (1) (1996) 127–148.
- [5] D. Juric, G. Tryggvason, Computations of boiling flows, *International Journal of Multiphase Flow* 24 (3) (1998) 387–410.
- [6] G. Tryggvason, B. Bunner, A. Esmaeeli, D. Juric, N. Al-Rawahi, W. Tauber, J. Han, S. Nas, Y.-J. Jan, A front-tracking method for the computations of multiphase flow, *Journal of Computational Physics* 169 (2) (2001) 708–759.
- [7] S. O. Unverdi, G. Tryggvason, A front-tracking method for viscous, incompressible, multi-fluid flows, *Journal of computational physics* 100 (1) (1992) 25–37.
- [8] H. Udaykumar, W. Shyy, M. Rao, Elafint: a mixed eulerian–lagrangian method for fluid flows with complex and moving boundaries, *International journal for numerical methods in fluids* 22 (8) (1996) 691–712. [52](#)
- [9] D. J. Benson, Computational methods in lagrangian and eulerian hydrocodes, *Computer methods in Applied mechanics and Engineering* 99 (2-3) (1992) 235–394. [52](#)
- [10] D. J. Benson, Volume of fluid interface reconstruction methods for multi-material problems, *Applied Mechanics Reviews* 55 (2) (2002) 151–165.
- [11] C. W. Hirt, B. D. Nichols, Volume of fluid (vof) method for the dynamics of free boundaries, *Journal of computational physics* 39 (1) (1981) 201–225.
- [12] J. Brackbill, D. B. Kothe, C. Zemach, A continuum method for modeling surface tension, *Journal of computational physics* 100 (2) (1992) 335–354.

- [13] N. V. Deshpande, Fluid mechanics of bubble growth and collapse in a thermal ink jet printhead, in: *Hard Copy Output*, Vol. 1079, International Society for Optics and Photonics, 1989, pp. 273–282.
- [14] E. G. Puckett, A. S. Almgren, J. B. Bell, D. L. Marcus, W. J. Rider, A high-order projection method for tracking fluid interfaces in variable density incompressible flows, *Journal of computational physics* 130 (2) (1997) 269–282. [52](#)
- [15] S. Osher, R. P. Fedkiw, Level set methods: an overview and some recent results, *Journal of Computational physics* 169 (2) (2001) 463–502. [52](#)
- [16] J. A. Sethian, *Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science*, Vol. 3, Cambridge university press, 1999.
- [17] J. A. Sethian, P. Smereka, Level set methods for fluid interfaces, *Annual review of fluid mechanics* 35 (1) (2003) 341–372.
- [18] M. Sussman, P. Smereka, S. Osher, A level set approach for computing solutions to incompressible two-phase flow, *Journal of Computational physics* 114 (1) (1994) 146–159. [52](#), [53](#), [54](#)
- [19] S. Osher, R. Fedkiw, *Level set methods and dynamic implicit surfaces*, Vol. 153, Springer Science & Business Media, 2006. [53](#)
- [20] Y.-C. Chang, T. Hou, B. Merriman, S. Osher, A level set formulation of eulerian interface capturing methods for incompressible fluid flows, *Journal of computational Physics* 124 (2) (1996) 449–464. [53](#)
- [21] E. Hachem, M. Khalloufi, J. Bruchon, R. Valette, Y. Mesri, Unified adaptive variational multiscale method for two phase compressible–incompressible flows, *Computer Methods in Applied Mechanics and Engineering* 308 (2016) 238–255. [53](#)
- [22] E. Hachem, R. Valette, M. Khalloufi, High fidelity anisotropic adaptive fem towards physical couplings occurring in turbulent boiling, *Numerical Modeling of Liquid-Vapor Interfaces in Fluid Flows*. [53](#)
- [23] G.-H. Cottet, E. Maitre, A semi-implicit level set method for multiphase flows and fluid–structure interaction problems, *Journal of Computational Physics* 314 (2016) 80–92. [53](#)
- [24] E. Olsson, G. Kreiss, A conservative level set method for two phase flow, *Journal of computational physics* 210 (1) (2005) 225–246. [55](#)
- [25] E. Olsson, G. Kreiss, S. Zahedi, A conservative level set method for two phase flow ii, *Journal of Computational Physics* 225 (1) (2007) 785–807. [55](#)
- [26] M. Sussman, E. G. Puckett, A coupled level set and volume-of-fluid method for computing 3d and axisymmetric incompressible two-phase flows, *Journal of computational physics* 162 (2) (2000) 301–337. [55](#)
- [27] M. Sussman, A second order coupled level set and volume-of-fluid method for computing growth and collapse of vapor bubbles, *Journal of Computational Physics* 187 (1) (2003) 110–136. [55](#)

- [28] G. Russo, P. Smereka, A remark on computing distance functions, *Journal of Computational Physics* 163 (1) (2000) 51–67. [55](#)
- [29] A. du Chéné, C. Min, F. Gibou, Second-order accurate computation of curvatures in a level set framework using novel high-order reinitialization schemes, *Journal of Scientific Computing* 35 (2-3) (2008) 114–131. [55](#)
- [30] D. Enright, R. Fedkiw, J. Ferziger, I. Mitchell, A hybrid particle level set method for improved interface capturing, *Journal of Computational physics* 183 (1) (2002) 83–116. [55](#)
- [31] M. Herrmann, A balanced force refined level set grid method for two-phase flows on unstructured flow solver grids, *Journal of computational physics* 227 (4) (2008) 2674–2706. [55](#)
- [32] C. Min, F. Gibou, A second order accurate level set method on non-graded adaptive cartesian grids, *Journal of Computational Physics* 225 (1) (2007) 300–321. [55](#)
- [33] F. Gibou, R. Fedkiw, S. Osher, A review of level-set methods and some recent applications, *Journal of Computational Physics*. [55](#)
- [34] C. Bahbah, M. Khalloufi, A. Larcher, Y. Mesri, T. Coupez, R. Valette, E. Hachem, Conservative and adaptive level-set method for the simulation of two-fluid flows. [55](#)
- [35] L. Ville, L. Silva, T. Coupez, Convected level set method for the numerical simulation of fluid buckling, *International Journal for numerical methods in fluids* 66 (3) (2011) 324–344. [55](#), [56](#)
- [36] S. Patankar, *Numerical heat transfer and fluid flow*, CRC press, 1980. [59](#)
- [37] K. Yokoi, A density-scaled continuum surface force model within a balanced force formulation, *Journal of Computational Physics* 278 (2014) 221–228. [59](#)
- [38] I. Babuška, The finite element method with lagrangian multipliers, *Numerische Mathematik* 20 (3) (1973) 179–192. [62](#)
- [39] F. Brezzi, On the existence, uniqueness and approximation of saddle-point problems arising from lagrangian multipliers, *Publications mathématiques et informatique de Rennes (S4)* (1974) 1–26. [62](#)
- [40] D. N. Arnold, F. Brezzi, M. Fortin, A stable finite element for the stokes equations, *Calcolo* 21 (4) (1984) 337–344. [62](#)
- [41] L. P. Franca, A. Nesliturk, On a two-level finite element method for the incompressible navier–stokes equations, *International Journal for Numerical Methods in Engineering* 52 (4) (2001) 433–453. [62](#)
- [42] C. Canuto, V. Van Kemenade, Bubble-stabilized spectral methods for the incompressible navier-stokes equations, *Computer methods in applied mechanics and engineering* 135 (1-2) (1996) 35–61. [62](#)
- [43] A. N. Brooks, T. J. Hughes, Streamline upwind/petrov-galerkin formulations for convection dominated flows with particular emphasis on the incompressible navier-stokes equations, *Computer methods in applied mechanics and engineering* 32 (1-3) (1982) 199–259. [62](#)

- [44] T. J. Hughes, L. P. Franca, M. Balestra, A new finite element formulation for computational fluid dynamics: V. circumventing the babuška-brezzi condition: A stable petrov-galerkin formulation of the stokes problem accommodating equal-order interpolations, *Computer Methods in Applied Mechanics and Engineering* 59 (1) (1986) 85–99. [62](#)
- [45] R. Codina, Stabilization of incompressibility and convection through orthogonal sub-scales in finite element methods, *Computer methods in applied mechanics and engineering* 190 (13-14) (2000) 1579–1599. [63](#), [65](#), [66](#)
- [46] R. Codina, J. Blasco, Stabilized finite element method for the transient navier–stokes equations based on a pressure gradient projection, *Computer Methods in Applied Mechanics and Engineering* 182 (3-4) (2000) 277–300.
- [47] R. Codina, Pressure stability in fractional step finite element methods for incompressible flows, *Journal of Computational Physics* 170 (1) (2001) 112–140.
- [48] R. Codina, J. Principe, Dynamic subscales in the finite element approximation of thermally coupled incompressible flows, *International journal for numerical methods in fluids* 54 (6-8) (2007) 707–730. [63](#), [65](#)
- [49] T. J. Hughes, Multiscale phenomena: Green’s functions, the dirichlet-to-neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods, *Computer methods in applied mechanics and engineering* 127 (1-4) (1995) 387–401. [63](#)
- [50] T. J. Hughes, G. R. Feijóo, L. Mazzei, J.-B. Quincy, The variational multiscale method—a paradigm for computational mechanics, *Computer methods in applied mechanics and engineering* 166 (1-2) (1998) 3–24.
- [51] T. J. Hughes, A. A. Oberai, L. Mazzei, Large eddy simulation of turbulent channel flows by the variational multiscale method, *Physics of fluids* 13 (6) (2001) 1784–1799. [63](#)
- [52] E. Hachem, B. Rivaux, T. Kloczko, H. Digonnet, T. Coupez, Stabilized finite element method for incompressible flows with high reynolds number, *Journal of computational physics* 229 (23) (2010) 8643–8665. [63](#), [64](#)
- [53] T. Dubois, F. Jauberteau, R. Temam, *Dynamic multilevel methods and the numerical simulation of turbulence*, Cambridge University Press, 1999. [64](#)
- [54] T. Coupez, Stable-stabilized finite element for 3d forming calculation. cemef, Tech. rep., ENSMP, internal report (1996). [64](#)
- [55] T. E. Tezduyar, Y. Osawa, Finite element stabilization parameters computed from element matrices and vectors, *Computer Methods in Applied Mechanics and Engineering* 190 (3-4) (2000) 411–430. [66](#)
- [56] J. C. Martin, W. J. Moyce, W. G. Penney, A. Price, C. Thornhill, Part iv. an experimental study of the collapse of liquid columns on a rigid horizontal plane, *Phil. Trans. R. Soc. Lond. A* 244 (882) (1952) 312–324. [67](#), [68](#), [69](#)
- [57] A. Murrone, H. Guillard, A five equation reduced model for compressible two phase flow problems, *Journal of Computational Physics* 202 (2) (2005) 664–698. [67](#), [68](#), [69](#)

- [58] R. N. Elias, A. L. Coutinho, Stabilized edge-based finite element simulation of free-surface flows, *International Journal for Numerical Methods in Fluids* 54 (6-8) (2007) 965–993. [68](#), [69](#)
- [59] L. Marioni, M. Khalloufi, F. Bay, E. Hachem, Two-fluid flow under the constraint of external magnetic field: Revisiting the dam-break benchmark, *International Journal of Numerical Methods for Heat & Fluid Flow* 27 (11) (2017) 2565–2581. [67](#), [68](#), [69](#)
- [60] C. Josserand, S. Zaleski, Droplet splashing on a thin liquid film, *Physics of fluids* 15 (6) (2003) 1650–1657. [69](#)
- [61] Q. Li, K. Luo, X. Li, Lattice boltzmann modeling of multiphase flows at large density ratio with an improved pseudopotential model, *Physical Review E* 87 (5) (2013) 053301. [69](#)
- [62] H. Yuan, Z. Chen, C. Shu, Y. Wang, X. Niu, S. Shu, A free energy-based surface tension force model for simulation of multiphase flows by level-set method, *Journal of Computational Physics* 345 (2017) 404–426. [vi](#), [71](#), [73](#)
- [63] Y. Wang, C. Shu, H. Huang, C. Teo, Multiphase lattice boltzmann flux solver for incompressible multiphase flows with large density ratio, *Journal of Computational Physics* 280 (2015) 404–423. [71](#)
- [64] S.-R. Hysing, S. Turek, D. Kuzmin, N. Parolini, E. Burman, S. Ganesan, L. Tobiska, Quantitative benchmark computations of two-dimensional bubble dynamics, *International Journal for Numerical Methods in Fluids* 60 (11) (2009) 1259–1288. [72](#)
- [65] M. Khalloufi, Y. Mesri, R. Valette, E. Massoni, E. Hachem, High fidelity anisotropic adaptive variational multiscale method for multiphase flows with surface tension, *Computer Methods in Applied Mechanics and Engineering* 307 (2016) 44–67. [72](#)
- [66] G. Brereton, D. Korotney, I. Sahin, G. Tryggvason, Dynamics of bubbles and vortices near a free surface, New York: ASME, AMD 119. [vi](#), [75](#), [76](#), [77](#)
- [67] M. Sussman, A. S. Almgren, J. B. Bell, P. Colella, L. H. Howell, M. L. Welcome, An adaptive level set approach for incompressible two-phase flows, *Journal of Computational Physics* 148 (1) (1999) 81–124. [75](#)
- [68] Z. Xie, D. Pavlidis, P. Salinas, J. R. Percival, C. C. Pain, O. K. Matar, A balanced-force control volume finite element method for interfacial flows with surface tension using adaptive anisotropic unstructured meshes, *Computers & Fluids* 138 (2016) 38–50. [76](#)

Chapter 4

Boiling multiphase flows : liquid-vapor-solid interactions

Contents

4.1 Introduction	88
4.2 Phase Change	88
4.2.1 Derivation of the governing equation without the phase change model	90
4.2.2 Derivation of the governing equation with the phase change model	91
4.2.3 Definition of the mass transfer rate	92
4.3 Surface tension	93
4.3.1 Standard definition	93
4.3.2 Semi-implicit time integration	94
4.4 Convection Diffusion Reaction equation	95
4.4.1 Governing equation	96
4.4.2 Standard Galerkin formulation	96
4.4.3 Streamline Upwind Petrov-Galerkin (SUPG) method	97
4.5 Numerical Examples	98
4.5.1 3D simulations of horizontal film boiling	98
4.5.1.1 Single film boiling	98
4.5.1.2 Multi film boiling	103
4.5.2 2D Quenching	107
4.6 Conclusion	110
4.7 Résumé du chapitre en français	111
4.8 Bibliography	111

4.1 Introduction

Numerical simulations for two-phase flows coupled with phase change and fluid-solid interactions is still an open topic, and no complete numerical models accurately including every phenomena can be found. In the case of the quenching process, we have the interaction between three phases with different material properties : the immersed solid, the vapor and the liquid since a vapor film surrounds the part because of the high thermal gradient at the interface. In our case, we focus on liquid coolants that can vaporize. Generally, vaporization is the leading phenomenon that drives the system. Thus the study of two-phase flows with phase change, thermal transfers and fluid-solid interactions is a first step to a better understanding of quenching processes. The classical well known methods for thermal treatment of a solid generally refer to the use of experimentally deduced heat transfer coefficients. This allow to simply and quickly deduce the cooling of a solid, without any knowledge of the fluid behavior. However, these latter are only useful for a particular configuration in term of geometry, orientation of the surface, range of temperature and flow motion. This is not very accurate and gives no information on the underlying physics; only a direct numerical simulation can take into account all the physics and all the scales involved in boiling.

When modeling the quenching process, several challenging questions have to be tackled to be able to correctly model quenching processes. On one hand, the complexity arises from the modeling of the liquid/vapor phase change at the interface. On the other hand, the surface tension force at the liquid/vapor interface as a result of the discontinuity of attractive forces plays an essential role in the mechanical behavior of this interface; if it is treated explicitly, then we have a capillary time step restriction condition, which is very inconvenient since we are dealing with industrial test cases. Thus, this chapter is dedicated to the methods used to model the liquid/vapor/solid interactions. We will first give details on the phase change model to predict precisely the liquid to vapor phase change during boiling, then we introduce a semi-implicit time integration of the surface tension force designed to circumvent the time step restriction condition. And then we present the stabilized finite-element method used to solve the convection diffusion reaction equation (CDR). Finally, we assess the behavior and accuracy of the framework in the simulation of several 2D and 3D time-dependent challenging test cases.

4.2 Phase Change

The first attempt to simulate the phase change at the interface between two phases is attributed to [1], where it was proposed to use a moving triangular grid system to represent the liquid-vapor interface. However, this method was limited to two-dimensional problems and didn't handle the topology changes of the interface. The use of interface tracking methods for phase-change simulations was later introduced. The use of the Volume Of Fluid (VOF) method coupled with a mass transfer model and a model for surface tension was proposed by [2]. The authors in [3] first proposed the capture of the interface by a level set method which is modified to include the liquid-vapor phase change effect, an this phase change model has been continuously improved over the years,[4–7]. These methods are popular due to the simplicity of implementation and the fact that complex topology changes such as the coalescence of bubbles are handled naturally. Phase fields approaches seem to be the most promising ones, [8–10]. The key idea is to use a field, namely the phase field, which is defined on the entire computational domain, and is a

marker of the location of the different phases. The phase field varies smoothly on the computational domain, and naturally produces thin layers which represent interfaces between phases. However, from the computational point of view, the main disadvantage of the phase-field approach is that it typically leads to higher-order partial-differential equations that are stiff in space and time, and produce thin layers which evolve dynamically over the computational domain.

Following the work of [11], we propose in the section an adaptive Eulerian framework to follow the evolution of a fluid going through phase change. We present the different steps towards a phase-change model coupled with the level set function that accounts for the mass transfer at the interface. In order to have a fully operational two phase flow with phase change numerical mode, let us first recall the level set formalism explained in Chapter 3. As explained before, a level set function is used to separate the different phases (liquid and vapor). In order to distribute the material properties in space, the associated indicator function $H_e(\alpha)$ allows us to determine the fraction of vapor and liquid in every point of space. The Heaviside function is determined as follows :

$$H_e(\alpha) = \begin{cases} 0 & \text{if } \alpha < -e \\ \frac{1}{2} \left(\frac{1}{\pi} \sin\left(\frac{\pi\alpha}{e}\right) + \frac{\alpha}{e} + 1 \right) & \text{if } |\tilde{\alpha}| \leq -e \\ 1 & \text{if } \alpha > e \end{cases} \quad (4.1)$$

And in order to define properties at the interface between the liquid and the vapor phase, we define the smoothed Dirac function $\delta_e(\alpha)$ as the derivative of the Heaviside function with respect to α :

$$\delta_e(\alpha) = \begin{cases} 0 & \text{if } |\alpha| > e \\ \frac{1}{2e} \left(\cos\left(\frac{\pi\alpha}{e}\right) + 1 \right) & \text{if } |\alpha| < -e \end{cases} \quad (4.2)$$

This formalism allows us to determine the characteristic of a fluid particle regarding the characteristics of the two phases. The derivation of the model relies on the mixing law used to distribute the density in space :

$$\rho = (1 - H_e(\alpha))\rho_l + H_e(\alpha)\rho_v \quad (4.3)$$

where ρ_l and ρ_v are respectively the density of the liquid and vapor phase.

First of all, let us recall the mass conservation equation :

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho u) = 0 \quad (4.4)$$

where v and ρ are respectively the velocity and the density. In order to explain the phase change model used in this thesis, we will first start by explaining the two-phase model without considering a phase change at the interface

4.2.1 Derivation of the governing equation without the phase change model

For the moment, let us first recall the equations defined for two-phase flows without any phase change. We write equation (4.4), for each phase (liquid and vapor) :

$$\left\{ \begin{array}{l} \frac{\partial(1 - H_e(\alpha))\rho_l}{\partial t} + \nabla \cdot ((1 - H_e(\alpha))\rho_l u) = 0 \\ \frac{\partial H_e(\alpha)\rho_v}{\partial t} + \nabla \cdot (H_e(\alpha)\rho_l u) = 0 \end{array} \right. \quad (4.5)$$

Dividing both equations of (4.5) by their respective density, we obtain the following system :

$$\left\{ \begin{array}{l} \frac{\partial(1 - H_e(\alpha))}{\partial t} + \nabla \cdot ((1 - H_e(\alpha))v) = 0 \\ \frac{\partial H_e(\alpha)}{\partial t} + \nabla \cdot (H_e(\alpha)v) = 0 \end{array} \right. \quad (4.6)$$

Summing both equations of (4.6), leads to the incompressibility equation :

$$\nabla \cdot u = 0 \quad (4.7)$$

If we sum the equations in the system (4.5) and consider the incompressibility equation, we obtain the following equation :

$$\frac{\partial H_e(\alpha)}{\partial t} + u \cdot \nabla H_e(\alpha) = 0 \quad (4.8)$$

Considering the derivative in time of the Heaviside function, we have :

$$\frac{\partial H_e(\alpha)}{\partial t} = \frac{\partial H_e(\alpha)}{\partial \alpha} \frac{\partial \alpha}{\partial t}, \quad (4.9)$$

thus, we have the following relation between H_e and δ_e :

$$\frac{\partial H_e(\alpha)}{\partial t} = \delta_e(\alpha) \frac{\partial \alpha}{\partial t} \quad (4.10)$$

And the derivative in space gives the following relation :

$$\nabla H_e(\alpha) = \delta_e(\alpha) \nabla \alpha \quad (4.11)$$

Combining equations (4.8), (4.10) and (4.11) we get the following relation :

$$\delta_e(\alpha) \frac{\partial \alpha}{\partial t} + \delta_e(\alpha) v \cdot \nabla \alpha = 0 \quad (4.12)$$

Extending equation (4.12) to the whole domain, we obtain the transport equation for the level set function:

$$\frac{\partial \alpha}{\partial t} + v \cdot \nabla \alpha = 0 \quad (4.13)$$

In the next section, we will extend the model to account for the liquid-to-vapor phase change occurring at the interface.

4.2.2 Derivation of the governing equation with the phase change model

Now, let us rewrite the mass conservation equation for the two phases while taking into account the phase change at the interface. We first define a mass transfer rate \dot{m} . And since the vapor is gaining mass due to vaporization, whereas the liquid phase is losing mass, a term appears on the right hand side of each equation in (4.5).

$$\begin{cases} \frac{\partial(1-H_e(\alpha))\rho_l}{\partial t} + \nabla \cdot ((1-H_e(\alpha))\rho_l u) = -(\dot{m} \cdot \nabla \alpha) \delta_e(\alpha) \\ \frac{\partial H_e(\alpha)\rho_v}{\partial t} + \nabla \cdot (H_e(\alpha)\rho_l u) = (\dot{m} \cdot \nabla \alpha) \delta_e(\alpha) \end{cases} \quad (4.14)$$

Developing equation (4.14) leads to :

$$\begin{cases} -\rho_l \frac{\partial H_e(\alpha)}{\partial t} + \rho_l(1-H_e(\alpha))\nabla \cdot u - \rho_l u \cdot \nabla H_e(\alpha) = -(\dot{m} \cdot \nabla \alpha) \delta_e(\alpha) \\ \rho_v \frac{\partial H_e(\alpha)}{\partial t} + \rho_v H_e(\alpha)\nabla \cdot u + \rho_v u \cdot \nabla H_e(\alpha) = (\dot{m} \cdot \nabla \alpha) \delta_e(\alpha) \end{cases} \quad (4.15)$$

Dividing both equations of (4.15) by their respective density, we obtain the following system :

$$\begin{cases} -\frac{\partial H_e(\alpha)}{\partial t} + (1-H_e(\alpha))\nabla \cdot u - u \cdot \nabla H_e(\alpha) = -\frac{1}{\rho_l}(\dot{m} \cdot \nabla \alpha) \delta_e(\alpha) \\ \frac{\partial H_e(\alpha)}{\partial t} + H_e(\alpha)\nabla \cdot u + u \cdot \nabla H_e(\alpha) = \frac{1}{\rho_v}(\dot{m} \cdot \nabla \alpha) \delta_e(\alpha) \end{cases} \quad (4.16)$$

And finally summing both equations of (4.16), we obtain the new mass conservation equation :

$$\nabla \cdot u = \left(\frac{1}{\rho_v} - \frac{1}{\rho_l} \right) (\dot{m} \cdot \nabla \alpha) \delta_e(\alpha) \quad (4.17)$$

Summing both equations of (4.16), we have :

$$(\rho_v - \rho_l) \frac{\partial H_e(\alpha)}{\partial t} + (\rho_v - \rho_l) u \cdot \nabla u + \nabla \cdot ((1-H_e(\alpha))\rho_l u) + \nabla \cdot (H_e(\alpha)\rho_l u) = 0 \quad (4.18)$$

Considering the mixing law in (4.3) to distribute the density in space, the equation in (4.18) becomes :

$$\frac{\partial H_e(\alpha)}{\partial t} + u \cdot \nabla u = \frac{\rho}{(\rho_l - \rho_v)} \nabla \cdot u \quad (4.19)$$

Using relations (4.10) and (4.11), equation (4.19) becomes :

$$\delta_e(\alpha) \frac{\partial \alpha}{\partial t} + \delta_e(\alpha) u \cdot \nabla \alpha = \frac{\rho}{\rho_l - \rho_v} \nabla \cdot u \quad (4.20)$$

Extending it to the whole domain and replacing the expression of $\nabla \cdot u$, we get :

$$\frac{\partial \alpha}{\partial t} + u \cdot \nabla \alpha = \frac{\rho}{\rho_l - \rho_v} \left(\frac{1}{\rho_v} - \frac{1}{\rho_l} \right) (\dot{m} \cdot \nabla \alpha) \quad (4.21)$$

Finally, the new transport equation accounting for the phase change is defined in the following manner :

$$\frac{\partial \alpha}{\partial t} + \left[u - \frac{\rho}{\rho_l \rho_v} \dot{m} \right] \cdot \nabla \alpha = 0 \quad (4.22)$$

Moreover, we also have to take account the absorbed or released heat at the interface due to phase change. Thus, the new heat equation is written as follows :

$$\rho c_p \left(\frac{\partial T}{\partial t} + u \cdot \nabla T \right) - \nabla \cdot (k \nabla T) = -\rho \left[L + (c_p^v - c_p^l)(T - T_{sat}) \right] \left(\frac{\rho}{\rho_l \rho_v} \dot{m} \cdot \nabla \alpha \right) \delta_e(\alpha) \quad (4.23)$$

where L is the enthalpy of vaporization (latent heat) and c_p^* the heat capacity for the liquid and the vapor phases. Here, one important information is missing here, we must compute the value of the mass transfer rate \dot{m} .

4.2.3 Definition of the mass transfer rate

The mass transfer rate is calculated from the heat flux at the interface. To do so, let us consider a domain composed of water and vapor (see Figure 4.1). Let us define by q_l and q_v the heat fluxes (W/m^2) coming from the liquid and the vapor sides of the interface :

$$q_l = k_l (\nabla T_l) \cdot \vec{n}, \text{ and} \quad (4.24)$$

$$q_v = -k_v (\nabla T_v) \cdot \vec{n} \quad (4.25)$$

where ∇T_l and ∇T_v are respectively the temperature gradients in liquid and vapor phases, \vec{n} is the normal vector to the interface, pointing from vapor to liquid. A representation

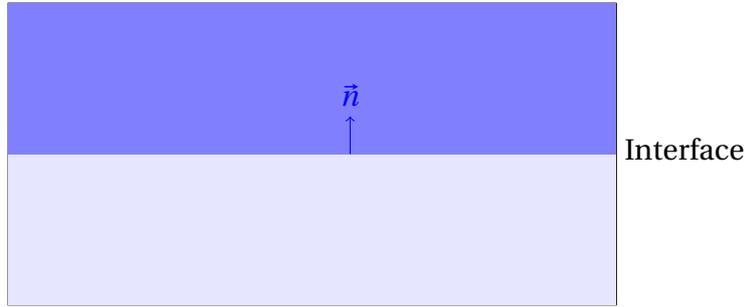


Figure 4.1: Heat fluxes from liquid and vapor phases to interface

of the different phases and the corresponding heat fluxes is given in Figure 4.1. If we consider a sharp interface, the mass transfer rate \dot{m} ($kg \cdot m^{-2} \cdot s$) is defined as :

$$\dot{m} = \frac{q_l + q_v}{L} \quad (4.26)$$

where L is the latent heat of vaporization (J/kg). However, since we are using level functions to model the interface, the evaluation in equation (4.26) is not straightforward when using an implicit definition of the interface. To tackle this issue, we suggest here to approximate the value of \dot{m} by using the expression on the whole interface layer by means of the dirac function δ_e . For this, let us consider the same domain composed of water and vapor (see Figure 4.2). Let us define by V , the volume of water that has vaporized in the time interval $[t, t + \Delta t]$. Thus, the position of the interface between the vapor and liquid phases changes. Therefore, we have :

$$\dot{m} \int_V \delta_e(\alpha) dV = \frac{\int_V \delta_e(\alpha) dV (q_l + q_v)}{L} \quad (4.27)$$

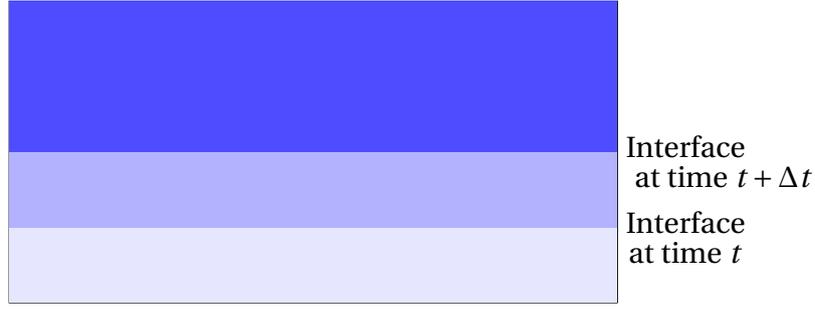


Figure 4.2: Interface position for a volume of liquid that has vaporized in the time interval $[t, t + \Delta t]$

which leads to :

$$\dot{m} = \frac{\int_V \delta_e(\alpha) dV (q_l + q_v)}{L \int_V \delta_e(\alpha) dV} \quad (4.28)$$

Finally, we recall the whole system of equations solved for the phase change model:

$$\begin{cases} \rho \left(\frac{\partial u}{\partial t} + u \cdot \nabla v \right) - 2\mu \nabla \cdot \varepsilon(u) + \nabla p = f + f_\sigma(\alpha) \\ \nabla u = \left(\frac{1}{\rho_v} - \frac{1}{\rho_l} \right) (\dot{m} \cdot \nabla \alpha) \delta_e(\alpha) \\ \frac{\partial \alpha}{\partial t} + \left[u - \frac{\rho}{\rho_l \rho_v} \dot{m} \right] \cdot \nabla \alpha = 0 \\ \rho c_p \left(\frac{\partial T}{\partial t} + u \cdot \nabla T \right) - \nabla (k \nabla T) = -\rho \left[L + (c_p^v - c_p^l)(T^{n-1} - T_{sat}) \right] \left(\frac{\rho}{\rho_l \rho_v} \dot{m} \cdot \nabla \alpha \right) \delta_e(\alpha) \\ \dot{m} = \frac{\int_V \delta_e(\alpha) dV (q_l + q_v)}{L \int_V \delta_e(\alpha) dV} \end{cases} \quad (4.29)$$

The implementation of the surface tension force f_σ will be explained in the next section.

4.3 Surface tension

4.3.1 Standard definition

Surface tension appears whenever two immiscible fluids (vapor and liquid in our case, see Figure 4.3) are in contact, and is a consequence of the increased energy of the molecules of the liquid (respectively, the vapor) at the interface, as compared to those comfortably surrounded by identical neighbors at the bulk of the liquid (respectively, the vapor).

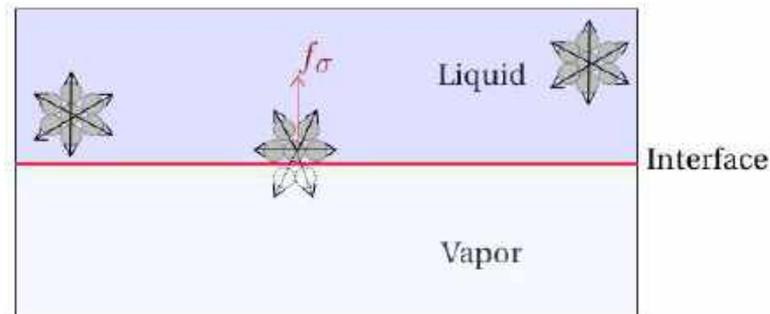


Figure 4.3: Representation of the surface tension force

The surface tension is usually implemented explicitly as a source term in the Navier Stokes equations :

$$f_{\sigma_{explicit}}(\alpha) = -\sigma\kappa\hat{n}\delta_e(\alpha) \quad (4.30)$$

where f_{σ} is the surface tension present on the interface and is proportional to the local curvature of the interface. σ is the surface tension coefficient, κ the interface curvature, $\delta_e(\alpha)$ the delta Dirac function used to represent the interface Γ between the liquid and the vapor and \hat{n} a unit normal vector. Since the interface is a moving (internal) boundary, a numerical method has to take this into account in order to handle arbitrarily shaped interfaces which may collide and break up. Thus, the use of a level set function allows us to compute directly the normal to the interface as follows :

$$\hat{n} = \frac{\nabla\alpha}{|\alpha|} \quad (4.31)$$

and in the same manner, we can express the curvature κ with respect to the level set function :

$$\kappa = \nabla \cdot \frac{\nabla\alpha}{|\nabla\alpha|} \quad (4.32)$$

Difficult to solve directly at the molecular scale, different approaches were proposed in the literature to solve the surface tension as a surface stress condition [12; 13]. The continuum surface force (CSF) method of Brackbill and al.[14] has been very popular over the last decade to model surface tension in eulerian mesh formulations, in particular in the volume of fluids methods [15–17], or level set techniques[18; 19] and front tracking [20; 21] methods. There, the surface tension forces acting on the interface are transformed to volume forces in regions near the interface via delta functions, thus the discontinuous interfacial jump conditions are modeled as smooth. However, surface tension is almost exclusively implemented with explicit time integration, this has the drawback that the time step size is limited by both the grid size and coefficient of surface tension. For numerical stability, the time step size Δt must satisfy the following condition :

$$\Delta t \leq \sqrt{\frac{\bar{\rho}(\Delta x)^3}{2\pi\sigma}} \quad (4.33)$$

where $\bar{\rho}$ is the average density of the tow phases (liquid and vapor) and Δx is the mesh size. The time step restriction due to surface tension is an order of magnitude smaller than the restriction imposed by the CFL condition, this was pointed out by many authors [22–24]. Thus, if this restriction could be removed, or 'softened', then we would be able to use larger time steps, and computational time would be significantly reduced.

4.3.2 Semi-implicit time integration

The surface tension force in this method consists of an explicit part, which is the regular continuum surface force, and an implicit part which represents the diffusion of velocities induced by surface tension on fluids interfaces. Following the work of [11], let us first introduce some definitions from differential geometry. If we define an identity mapping Id_{Γ} on the interface Γ , then we have the following relation :

$$\underline{\Delta}Id_{\Gamma} = -\kappa\hat{n} \quad (4.34)$$

where $\underline{\Delta}Id_{\Gamma}$ is the tangential Laplacian operator. If we consider a backward Euler scheme, we can approximate the position of the interface in time :

$$Id_{\Gamma}^{t+1} = Id_{\Gamma}^t + \Delta t v^{t+1} \quad (4.35)$$

with the index $t + 1$ representing the current time and t the previous time step. Thus, we can write :

$$\underline{\Delta}Id_{\Gamma}^{t+1} = \underline{\Delta}Id_{\Gamma}^t + \Delta t(\underline{\Delta}v^{t+1}) \quad (4.36)$$

Which leads to :

$$-(\kappa \hat{n})^{t+1} = -(\kappa \hat{n})^t + \Delta t(\underline{\Delta}v^{t+1}) \quad (4.37)$$

Thus, we can rewrite the expression of the surface tension :

$$f_{\sigma_{Semi-implicit}}(\alpha) = f_{\sigma_{Explicit}}(\alpha) + \Delta t(\underline{\Delta}v^{n+1})\sigma\delta_e(\alpha) \quad (4.38)$$

The tangential Laplacian can be interpreted as the standard Laplacian minus the second derivative in the normal direction, i.e., anisotropic diffusion minus diffusion in the normal direction :

$$\underline{\Delta}v^{t+1} = \nabla^2 v^{t+1} - \frac{\partial^2 v^{t+1}}{\partial n^2} - \kappa \frac{\partial v^{t+1}}{\partial n} \quad (4.39)$$

Or :

$$f_{\sigma_{Explicit}}(\alpha) = \frac{v^{t+1} - v^t}{\Delta t} \quad (4.40)$$

Thus :

$$v^{t+1} = f_{\sigma}(\alpha)\Delta t + v^t \quad (4.41)$$

Therefore :

$$\frac{\partial^2 v^{t+1}}{\partial n^2} = \frac{\partial^2 v^t}{\partial n^2}, \text{ and} \quad (4.42)$$

$$\frac{\partial v^{t+1}}{\partial n} = -\sigma\kappa\delta_e(\alpha)\Delta t + \frac{\partial v^t}{\partial n} \quad (4.43)$$

Which leads to the following surface tension expression :

$$f_{\sigma_{Semi-implicit}}(\alpha) = f_{\sigma_{Explicit}}(\alpha) - \sigma \left[\frac{\partial^2 v^t}{\partial n^2} \Delta t + \kappa \frac{\partial v^t}{n} \Delta t + \kappa \Delta^2 t - \Delta t \nabla^2 v^{t+1} \right] \delta_e(\alpha) \quad (4.44)$$

We can neglect the term in $\Delta^2 t$, thus the final semi-implicit expression of the surface tension is expressed as follows :

$$f_{\sigma_{Semi-implicit}}(\alpha) = f_{\sigma_{Explicit}}(\alpha) - \sigma \Delta t \left[\frac{\partial^2 v^t}{\partial n^2} + \kappa \frac{\partial v^t}{n} - \nabla^2 v^{t+1} \right] \delta_e(\alpha) \quad (4.45)$$

The classic explicit expression of the surface tension is now completed with terms proportional to the time step. This formulation has proven to be very stable [11]. With a small surface tension, both schemes (explicit and semi-implicit) work well. With large surface tension, the semi-implicit scheme can still give correct results due to the stabilization of the implicit term, while the explicit scheme immediately becomes highly unstable.

4.4 Convection Diffusion Reaction equation

In this section the general equation of convection-diffusion-reaction is described and solved. The solution of the convection diffusion reaction (CDR) problem is an important feature for the numerical modeling of a wide range of fluid mechanics applications such as heat transfer equations and turbulence models. Over the years, there has been a continuous growth in the development of numerical methods for the resolution of this

type of equations. Usually the Galerkin Finite Element (FE) method is the first one mentioned among the various numerical techniques available to solve these problems due to its simplicity and easy implementation. However, a standard Galerkin formulation leads to global spurious oscillations in convection-dominated regimes, especially in the regions near sharp gradients; thus causing a loss of accuracy and stability. We can find in the literature many papers with different methods proposed to avoid these numerical oscillations. Stabilization methods for transient convection-diffusion-reaction equations are discussed in [25–27]. In this section, we start with the description of the governing equation for the modeling of the convection-diffusion-reaction problem. Then we introduce the standard Galerkin finite element formulation and point out the need for stabilized finite element methods. The third subsection is devoted to the stabilized formulation Streamline Upwind Petrov-Galerkin (SUPG) to overcome the non-physical oscillations.

4.4.1 Governing equation

The convection-diffusion-reaction equation over a bounded domain $\Omega \subset \mathcal{R}^d$ with boundary domain Γ (d being the space dimension) consists in finding a scalar $\phi(x, t)$ such that:

$$\left\{ \begin{array}{ll} (\partial_t \phi + u \cdot \nabla \phi - \nabla \cdot (k \nabla \phi) + r \phi = f & \text{in } \Omega \times [0, T] \\ \phi(\cdot, 0) = \phi_0 & \text{in } \Omega \times [0, T] \\ \phi = g & \text{on } \Gamma \times [0, T] \end{array} \right. \quad (4.46)$$

where $k > 0$ is the diffusion factor, ϕ a given divergence free velocity, $r > 0$ the reaction coefficient, f is a given source term, ϕ_0 is the initial data and g is a given boundary condition. Here, we only consider Dirichlet boundary conditions, $\phi = 0$ on Γ .

4.4.2 Standard Galerkin formulation

In this work, a finite element method (FEM) approach has been used to solve the convection-diffusion-reaction equation. To derive the variational formulation of the problem (4.46), let us first define the function spaces :

$$\mathcal{H}_0^1(\Omega) = \left\{ w \mid w \in \mathcal{H}^1(\Omega), w = 0 \text{ on } \Gamma \right\},$$

with

$$\mathcal{H}^1(\Omega) = \left\{ w \mid w \in \mathcal{L}^2(\Omega), \|\nabla w\| \in \mathcal{L}^2(\Omega) \right\},$$

and $\mathcal{L}^2(\Omega)$ being the Hilbert vector space given by :

$$\mathcal{L}^2(\Omega) = \left\{ w(x) \mid \int_{\Omega} |w|^2 dx < \infty \right\}$$

The Galerkin variational formulation of the convection-diffusion-reaction is obtained by multiplying equation (4.46) by a test function $w \in \mathcal{H}_0^1(\Omega)$ and integrating over the computational domain. Let us introduce the following integral notation :

$$(a, b)_{\Omega} := \int_{\Omega} a b dx$$

The weak formulation now reads :

$$\left\{ \begin{array}{l} \text{Find } \phi \in \mathcal{H}_0^1(\Omega) \text{ such that:} \\ (\partial_t \phi, w) + (k \nabla \phi, \nabla w) + (u \cdot \nabla \phi, w) + (r \phi, w) = (f, w) \quad \forall w \in \mathcal{H}_0^1(\Omega) \end{array} \right. \quad (4.47)$$

Let us now consider the finite element partition \mathcal{T}_h of the domain Ω subdivided into simplex elements K . Using the partition \mathcal{T}_h , the functional spaces are now approximated by discrete spaces called $\mathcal{H}_{0,h}^1(\Omega)$ and $\mathcal{H}_h^1(\Omega)$. With these definitions, the Galerkin finite element formulation reads as:

$$\left\{ \begin{array}{l} \text{Find } \phi_h \in \mathcal{H}_{0,h}^1(\Omega) \text{ such that:} \\ (\partial_t \phi_h, w_h) + (k \nabla \phi_h, \nabla w_h) + (\phi \cdot \nabla \phi, w_h) + (r \phi_h, w_h) = (f, w_h) \quad \forall w_h \in \mathcal{H}_{0,h}^1(\Omega) \end{array} \right. \quad (4.48)$$

As explained before, the standard Galerkin formulation is unstable and leads to spurious oscillations when the convective term of the equation is dominant. For this reason, we stabilize the formulation using the SUPG numerical scheme. This method has proved its efficiency in eliminating the spurious oscillations related to the Galerkin formulation and is now a standard for heat transfer Finite Element solver.

4.4.3 Streamline Upwind Petrov-Galerkin (SUPG) method

In this work, we use the streamline upwind Petrov-Galerkin (SUPG) technique to stabilize the solution and enhance its stability. This method modifies the test functions by adding weighted residuals to the variational formulation of the problem. In brief, the basic idea of the SUPG method is to add artificial diffusion which acts only in the direction of the flow. The method is globally stable and has good order of accuracy in regions where the solution is smooth. To define, the SUPG approximation of equation (4.48), we start by introducing an additional weight $\tau_K u \nabla w_h$ to the standard Galerkin weighting functions w_h in the upwind direction for all terms in the equation. This modification adds more weight to the nodes in the upstream direction and reduces the weight of the nodes in the downstream direction. Thus, the modified test function is defined by:

$$\tilde{w}_h = w_h + \tau_K u \nabla w_h$$

As a result, the stabilized variational formulation of equation (4.48) reads:

$$\left\{ \begin{array}{l} \text{Find } \phi_h \in \mathcal{H}_{0,h}^1(\Omega) \text{ such that:} \\ (\partial_t \phi_h + u \cdot \nabla \phi_h, w_h) + \sum_{K \in \mathcal{T}_h} \tau_K (\partial_t \phi_h, u \cdot \nabla w_h)_K + (k \nabla \phi_h, \nabla w_h) + (r \phi_h, w_h) \\ + \sum_{K \in \mathcal{T}_h} \tau_K (\mathcal{R}_{\phi_h}, u \cdot \nabla w_h)_K = (f, w_h) + \sum_{K \in \mathcal{T}_h} \tau_K (f, u \cdot \nabla w_h)_K \quad \forall w_h \in \mathcal{H}_{0,h}^1(\Omega) \end{array} \right. \quad (4.49)$$

where τ_K is a stabilization parameter that tunes the amplitude of the added weight and \mathcal{R}_{u_h} the appropriate residual of equation (4.48) given by :

$$\mathcal{R}_{\phi_h} = \partial_t \phi_h - \nabla \cdot (k \nabla \phi_h) + u \cdot \nabla \phi_h + r \phi_h - f$$

The latter stabilization adds numerical diffusion in the neighborhood of sharp gradient and boundary layers. Concerning the choice of the stabilizing parameter τ_K , we refer to Hughes and al. in [28] and we choose:

$$\tau_K = \frac{h_m}{2 \|u\|_2} \left(\coth(\text{Pe}_K) - \frac{1}{\text{Pe}_K} \right) \quad (4.50)$$

where Pe_K is the local Peclet number defined as :

$$\text{Pe}_K = \frac{\|u\|_2 h_m}{2|K|}$$

and h_m is the element size computed as described in Chapter 2. Note that other possible suggestions for the choice of the stabilizing parameter can be found in [26; 29; 30].

4.5 Numerical Examples

In order to validate the proposed framework that handles boiling multiphase flows, we present in this section three test cases. We are first interested in the 3D simulation of horizontal film boiling, we first consider a single film boiling mode and then mutli film boiling. The obtained results are compared with results taken from the literature. The next test case is a 2D quenching simulation of a hot metal part.

4.5.1 3D simulations of horizontal film boiling

4.5.1.1 Single film boiling

In this section, we present a 3D simulation of a single film boiling on a horizontal surface where a thin vapor layer covers a heated plate and phase change takes place at the liquid/vapor interface, the set up is shown in Figure 4.4. Several authors [2; 31; 32] have

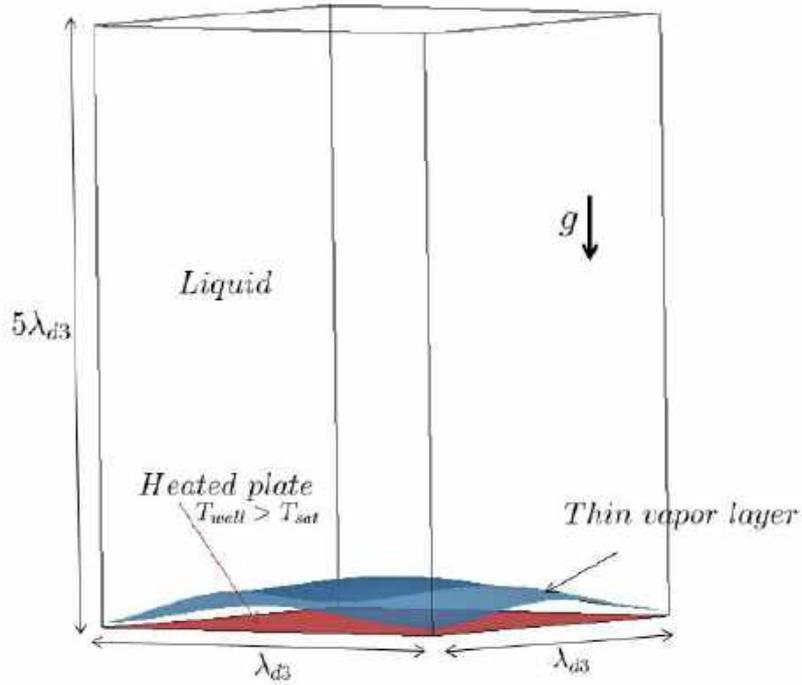


Figure 4.4: Set up of the 3D single film boiling

analyzed this test case in order to validate their phase change model. The vapor film completely covers the heated surface and the denser liquid above the vapor film falls under the action of gravity resulting in Rayleigh–Taylor instability. In 2D, the most unstable taylor wavelength is :

$$\lambda_{d2} = 2\pi \left[\frac{3\sigma}{g(\rho_l - \rho_v)} \right]^{\frac{1}{2}} \quad (4.51)$$

where σ is the surface tension coefficient and g the gravity force. In 3D, the domain size in the horizontal directions must be equal or greater than the most dangerous three-dimensional wavelength λ_{d3} , defined as [33] :

$$\lambda_{d3} = \sqrt{2}\lambda_{d2} = 2\sqrt{2}\pi \left[\frac{3\sigma}{g(\rho_l - \rho_v)} \right]^{\frac{1}{2}} \quad (4.52)$$

	$\rho(\text{kg}.m^{-3})$	$\mu(\text{Pa}.s)$	$c_p(\text{J}.kg^{-1}.K^{-1})$	$k(\text{W}.m^{-1}.K^{-1})$	$L(\text{J}.kg^{-1})$
Liquid	200	0.1	400	40	10^4
Vapor	5.0	0.005	200	1.0	

Table 4.1: Physical parameters for the 3D simulation of film boiling

The depth of the liquid must also be large enough to accommodate the growth of the vapor jets, thus we use a domain of size $\lambda_{d3} \times \lambda_{d3} \times 5\lambda_{d3}$. The bottom wall is at a constant temperature $T_{wall} > T_{sat}$ and the vertical temperature gradient is set to zero at the top wall. The temperature at the wall T_{wall} is maintained constant at a temperature of 5K above the saturation temperature. The initial film thickness is perturbed slightly and is given by:

$$z = z_c + \varepsilon \left[\cos\left(\frac{2\pi x}{\lambda_{d3}}\right) + \cos\left(\frac{2\pi y}{\lambda_{d3}}\right) \right] \quad (4.53)$$

where z_c is the unperturbed film thickness, ε the perturbation amplitude and x and y the coordinates of the mesh. Following the work of [2], the physical parameters used for the simulation are depicted in table 4.1. And, the governing dimensionless parameters for this problem are:

- The Prandtl number that represents the ratio of the momentum diffusivity over the thermal diffusivity :

$$Pr = \frac{\mu_l c_{p,l}}{k_l} = 2.0$$

- The Grashof number that represents the ratio of the buoyancy force over the viscous force :

$$Gr = \frac{\rho_v(\rho_l - \rho_v)g\lambda_{d3}}{\mu_v^2} = 18.5$$

- The Jacob number that represents the ratio of sensible heat over latent heat :

$$Ja = (T_w - T_{sat})c_{p,l} = 0.1$$

In film boiling of common liquids, the vapor film is usually very thin. Thus, if we choose a uniform grid to accurately resolve the film, the computational cost will be very high and in some cases maybe prohibitive. Hence, we are using here conservative mesh adaptation to accurately follow the evolution of the liquid/vapor interface. A number of 200 000 elements was used to perform this simulation. Figure 4.5 shows the evolution of the mesh and the interface. We can clearly see that the mesh is very fine at the liquid/vapor interface and at the bottom when the thermal gradient is the highest. The mesh remains coarser far from the interface, where the thermal gradient is null and the velocity field is still zero as depicted in Figure 4.6 that shows the velocity fields during the simulation. As the liquid moves closer to the wall, evaporation at the liquid/vapor interface prevents the liquid from getting in contact with the wall, thus the liquid/vapor interface becomes unstable and a mushroom shaped bubble is formed. Figure 4.7 shows the evolution of the temperature field and the interface for $\Delta T = 5\text{K}$. The vapor film grows to a mushroom shape due to the Rayleigh-Taylor instability. Since we performed 3D computations, a break up occurs due to surface tension.

For completeness, we computed the evolution of the space-averaged Nusselt number defined as the dimensionless heat flux through the wall :

$$Nu = \frac{\lambda_{d3}}{T_w - T_{sat}} \left. \frac{\partial T}{\partial z} \right|_{z=0} \quad (4.54)$$

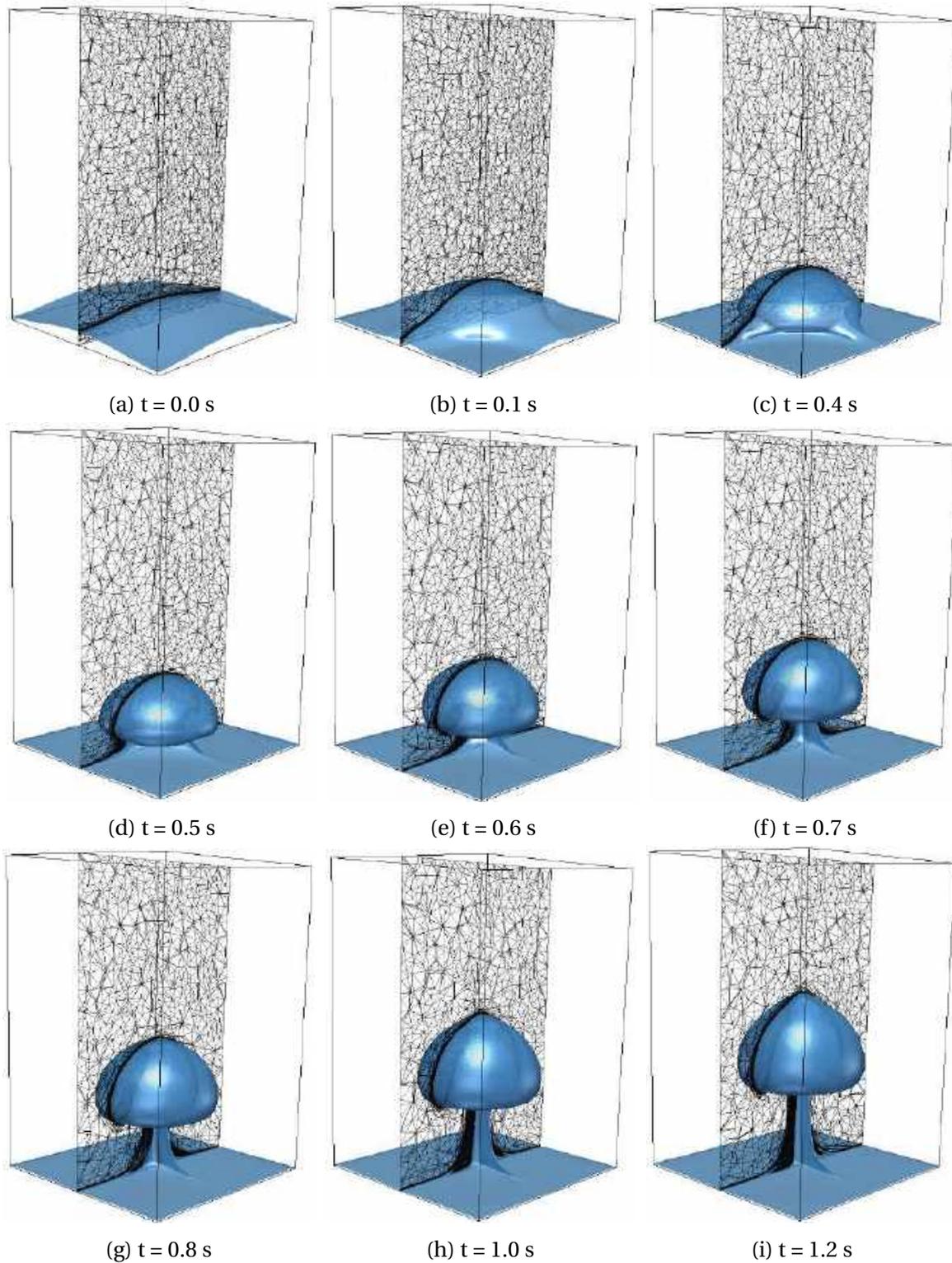


Figure 4.5: Evolution of the liquid/vapor interface and meshes during 3D film boiling

And we compared it to two of the most widely used correlations in literature for the heat transfer coefficient in film boiling on horizontal surfaces. The first one is the correlation of Berenson [34]:

$$Nu_B = 0.425 \left(\frac{GrPr}{Ja} \right)^{\frac{1}{4}} \quad (4.55)$$

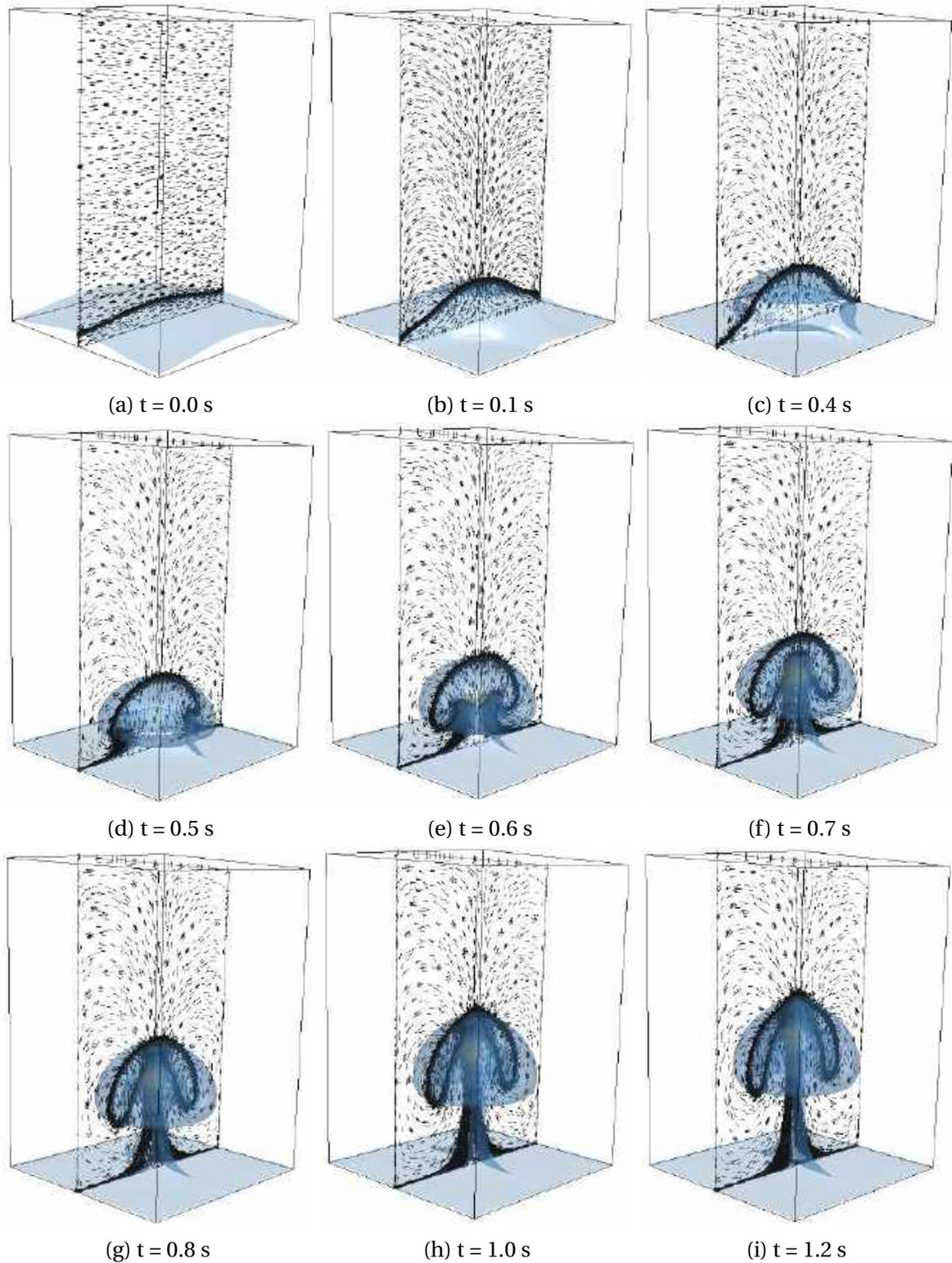


Figure 4.6: Evolution of the liquid/vapor interface and velocity during 3D film boiling

and the second one is the correlation of Klimenko [35] :

$$Nu_{\kappa} = 0.1691 \left(\frac{GrPr}{Ja} \right)^{\frac{1}{3}} \text{ for } Ja < 0.71 \quad (4.56)$$

The evolution of the space-averaged Nusselt number is depicted in Figure 4.8. The comparison with the correlation of Berenson and Klimenko shows a good agreement. The

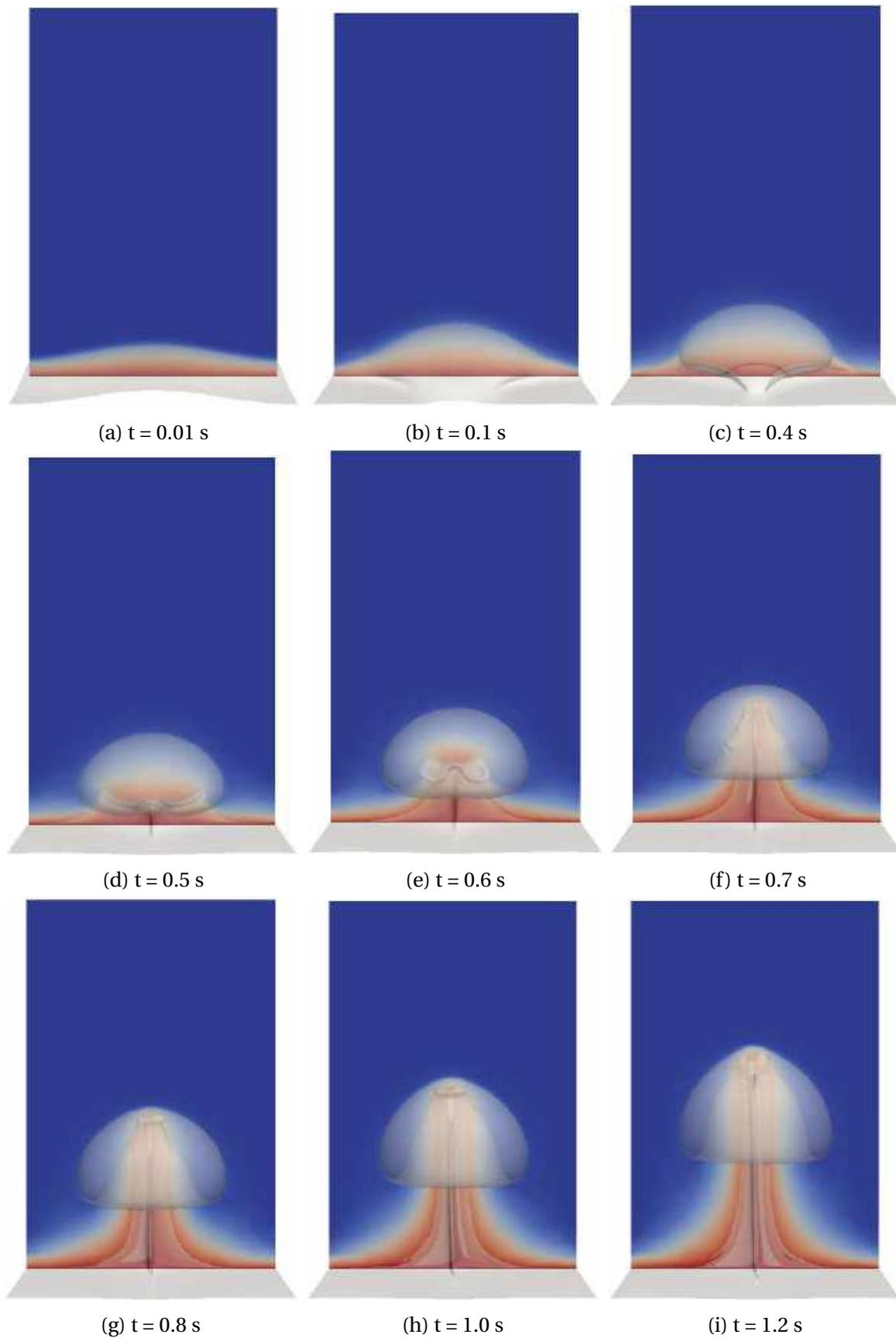


Figure 4.7: Evolution of the temperature field and the interface location for single film boiling

Nusselt number increases at the beginning because of large temperature difference tem-

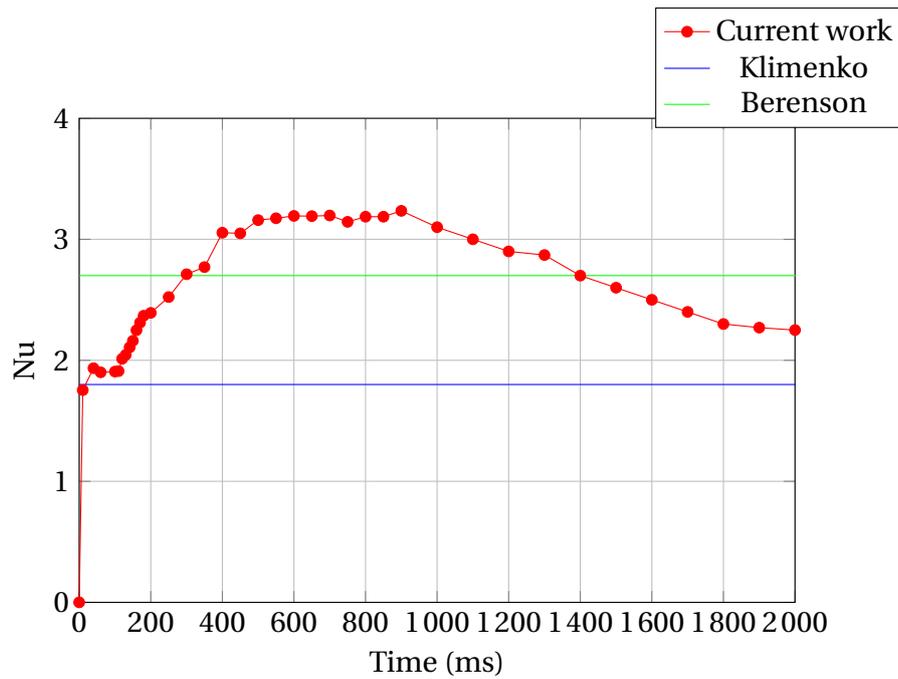


Figure 4.8: Evolution of Nu for the single film boiling in 3D

perature between the wall and vapor in the film. When the vapor begins to warm up, the temperature gradient at the wall drops down and the Nusselt number decreases.

4.5.1.2 Multi film boiling

In this section, we are interested in multi film boiling, where the evolution of the vapor film leads to the formation of different bubbles. We use the same perturbation as before. The set up is shown in Figure 4.9.

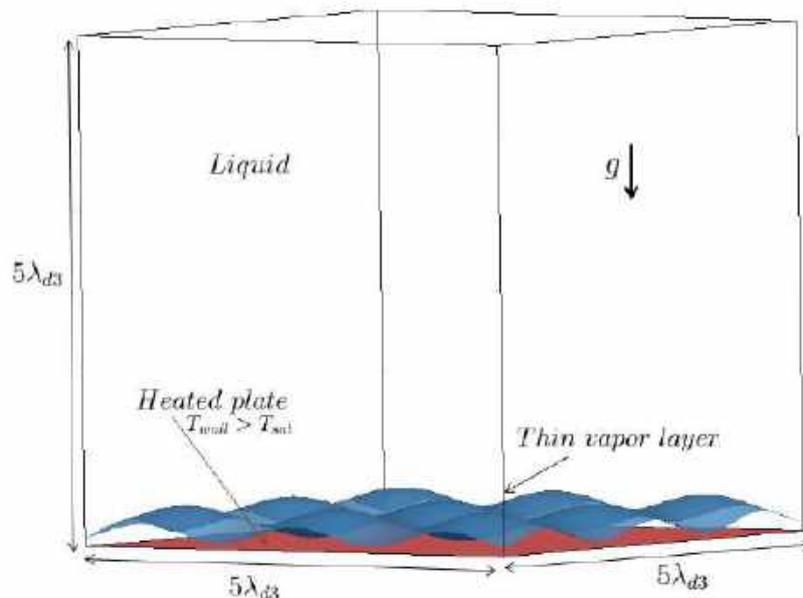


Figure 4.9: Set up of the 3D multi film boiling

As expected, we can see in Figures (4.10, 4.11) that the initial perturbation has grown and bubbles have been formed. This can happen if the size of the heated surface in suf-

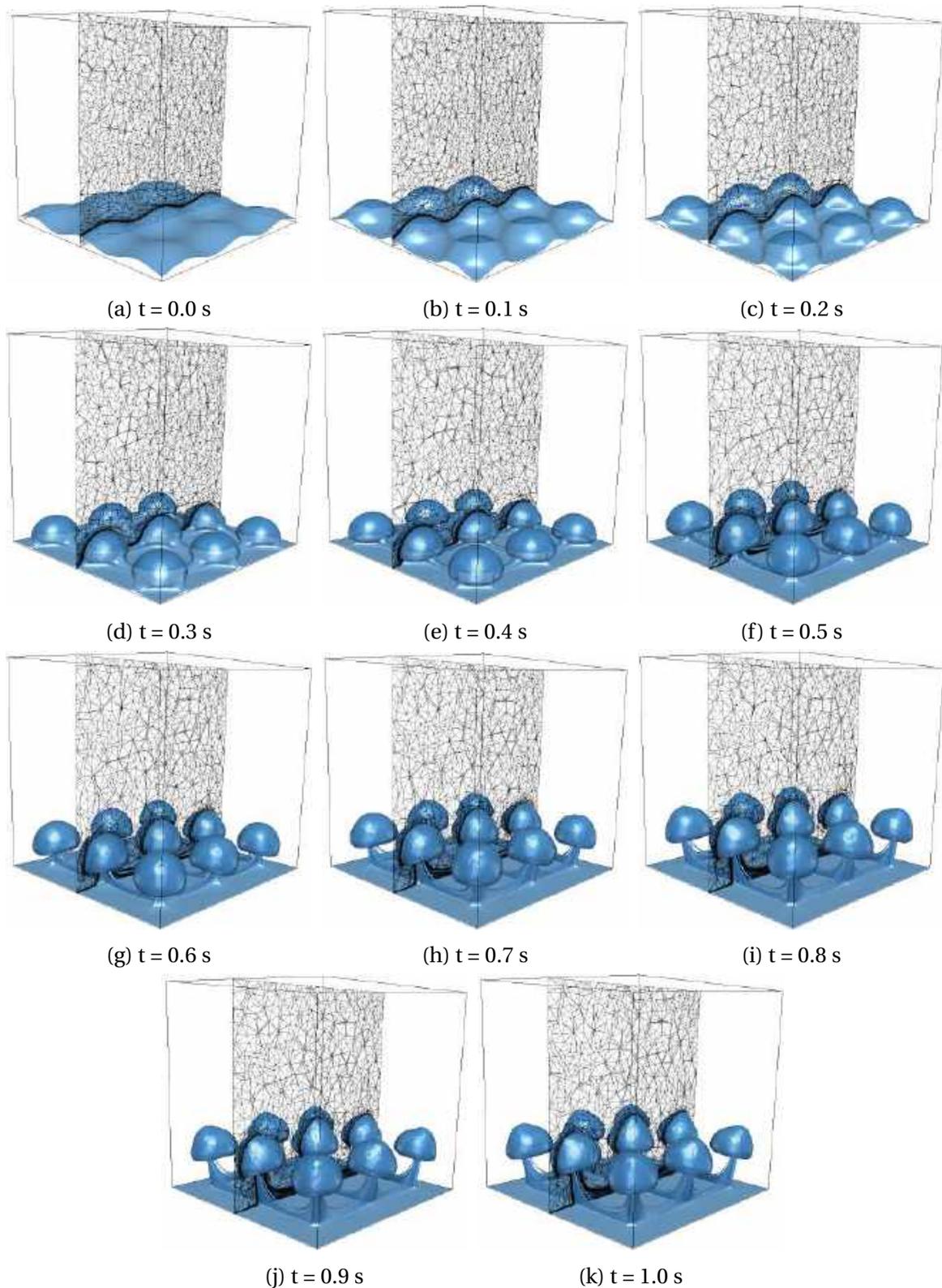


Figure 4.10: Evolution of the liquid/vapor interface and meshes during 3D multi film boiling

ficiently large compared to λ_{d3} . The simulation have been carried out with 900000 elements. Same here, the criteria chosen for the adaptive meshing is again both the velocity (in direction and magnitude) and the level set function and we can clearly see that the mesh is well adapted to the interface of the bubbles. Figure 4.12 shows the evolution of the temperature field and the interface of the bubbles. We can see that a break up occurs

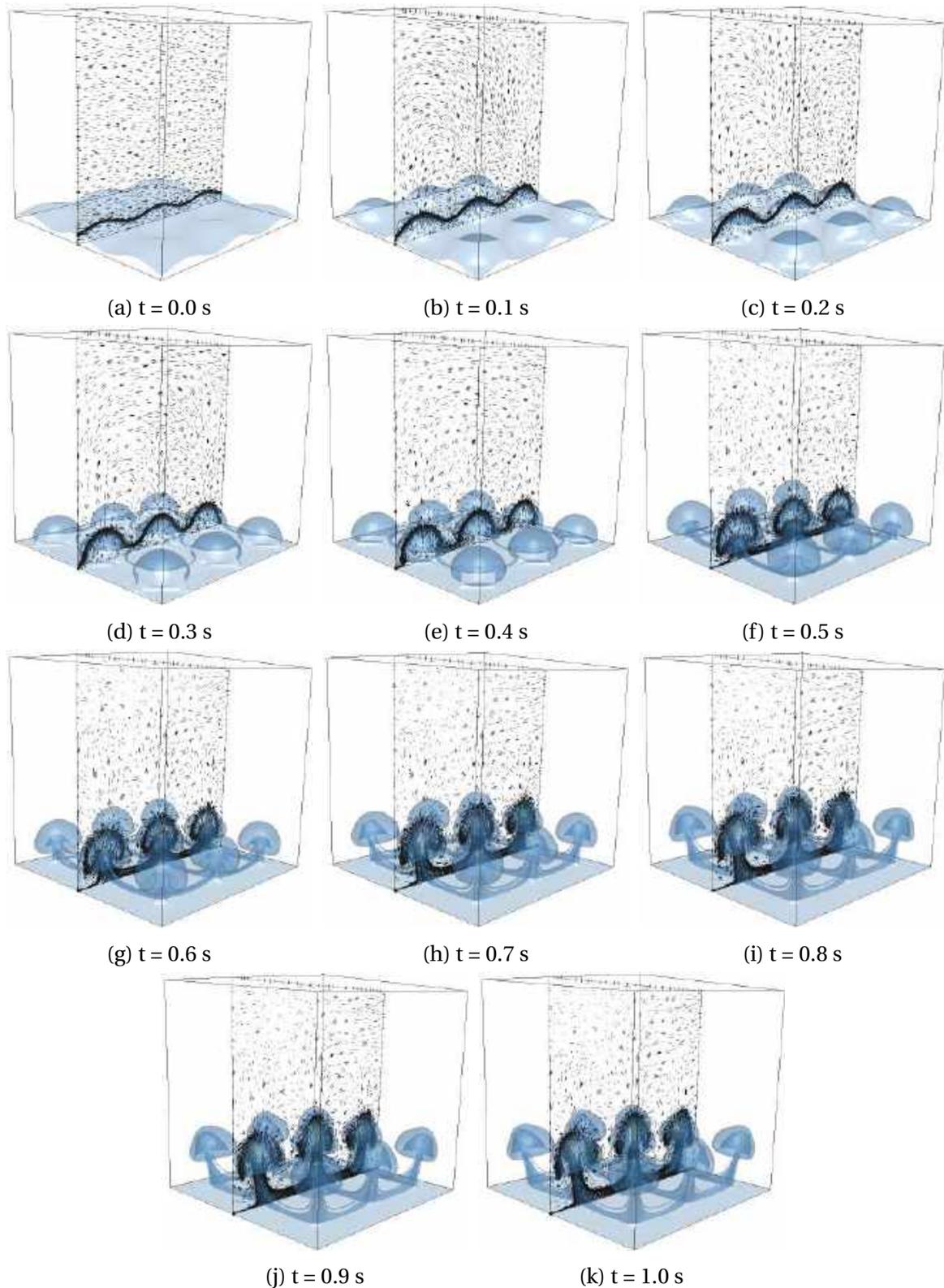


Figure 4.11: Evolution of the liquid/vapor interface and velocity during 3D multi film boiling

due to the surface tension force. No Nusselt number was computed in this case, since no benchmark was available in the literature. The purpose here was to show the ability of the framework to handle different bubble formation and to follow the evolution of the different interfaces accurately.

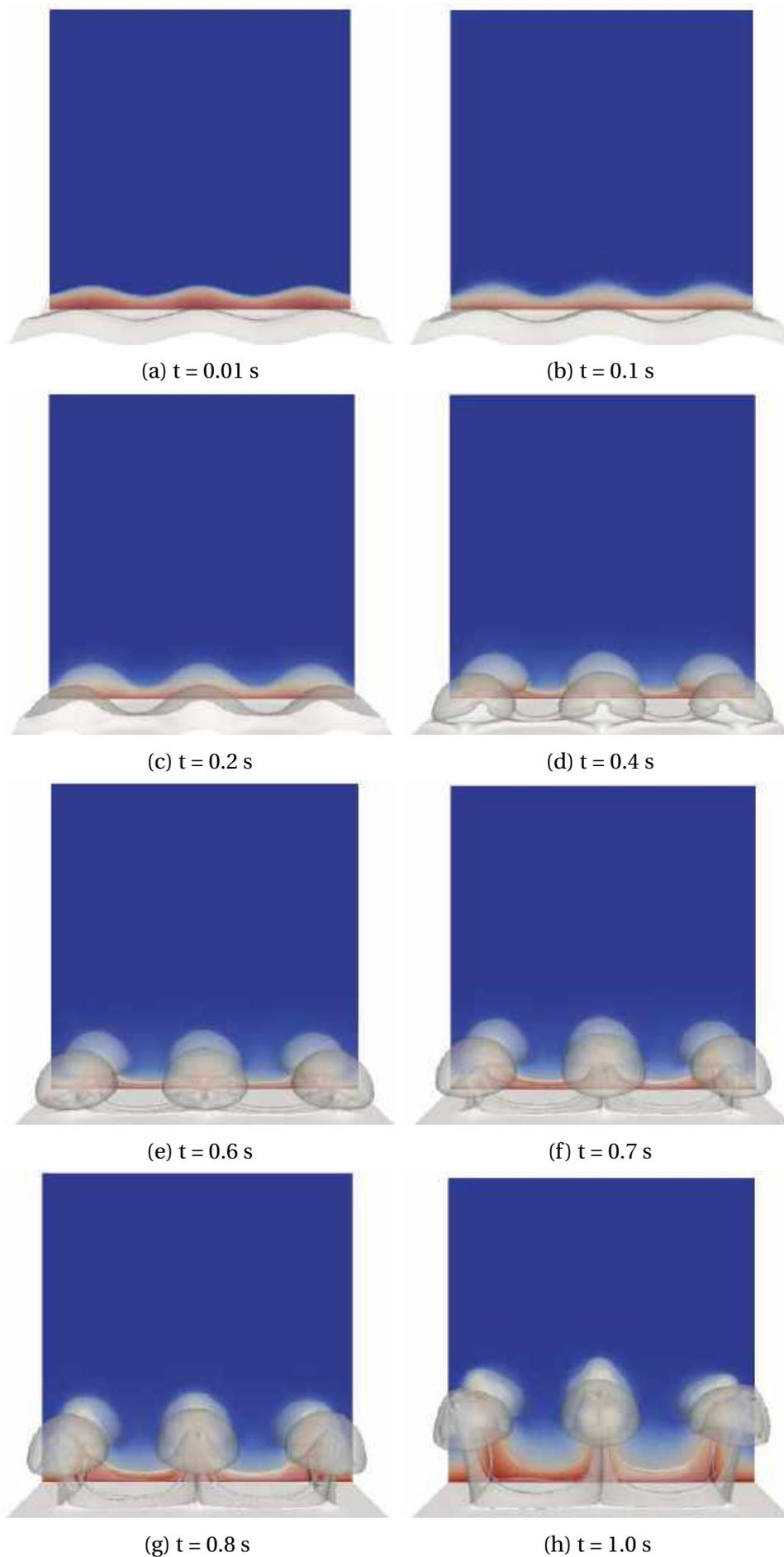


Figure 4.12: Evolution of the temperature field and the interface location for multi film boiling

4.5.2 2D Quenching

In this section, we are interested in the 2D simulation of the quenching process. We will consider a 2D solid in a domain of size $0.6 \times 0.4m^2$, three quarters full of water in which a metallic sample of dimension $0.2 \times 0.045m^2$ is immersed. A vapor initially surrounds the metal part. The set up is depicted in Figure 4.13. A free slip boundary condition is

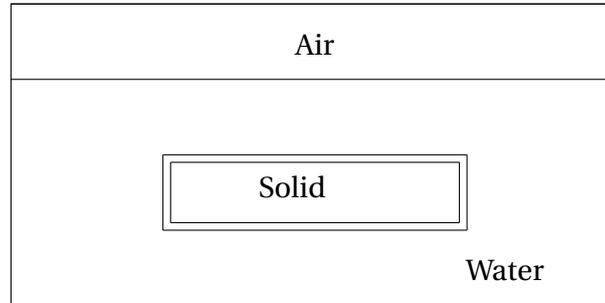


Figure 4.13: Set up of the 2D quenching

prescribed on all the walls. The motion of the vapor film is due to the buoyancy force and the surface tension force. The simulation is performed using the conservative adaptation technique with 25000 elements. All the physical parameters related to each phase are presented in table 4.2.

	$\rho(kg.m^{-3})$	$\mu(Pa.s)$	$c_p(J.kg^{-1}.K^{-1})$	$k(W.m^{-1}.K^{-1})$	$T_0(C)$	$L(J.kg^{-1})$
Liquid	1000	1×10^{-3}	2010	0.025	25	2.26×10^6
Vapor	1.7	1.2×10^{-5}	4185	0.6	100	
Solid	8000		435	11.4	880	

Table 4.2: Physical parameters for the 2D quenching process simulation

The evolution of the liquid/vapor interface is depicted in Figure 4.14, we can see the evolution of the mesh over time. The first picture shows the initial mesh, the elements are mostly concentrated around the part and the free surface. As the simulation starts and bubbles form, the mesh is adapted automatically according to the level set interface but also the velocity and the temperature. While keeping a fixed number of elements, we can see that the mesh is coarsened automatically at the bottom of the tank, where the variation of the solution is small. The anisotropic mesh adaptation enables to capture the high thermal gradient as well as the complexity of the flow. Figure 4.15 shows the evolution of the temperature inside the solid piece. At the beginning of the simulation, when the solid part is still hot, we can see the formation of many bubbles and when the part is colder, the boiling becomes less and less turbulent. For completeness, we plotted the evolution of the temperature history observations at four different monitoring locations, see Figure 4.16. We can clearly see in Figure 4.17 that the cooling is faster near the surface of the solid part, and takes more time the closest we get to the center of the metal part. This test case can serve as a benchmark for future work.

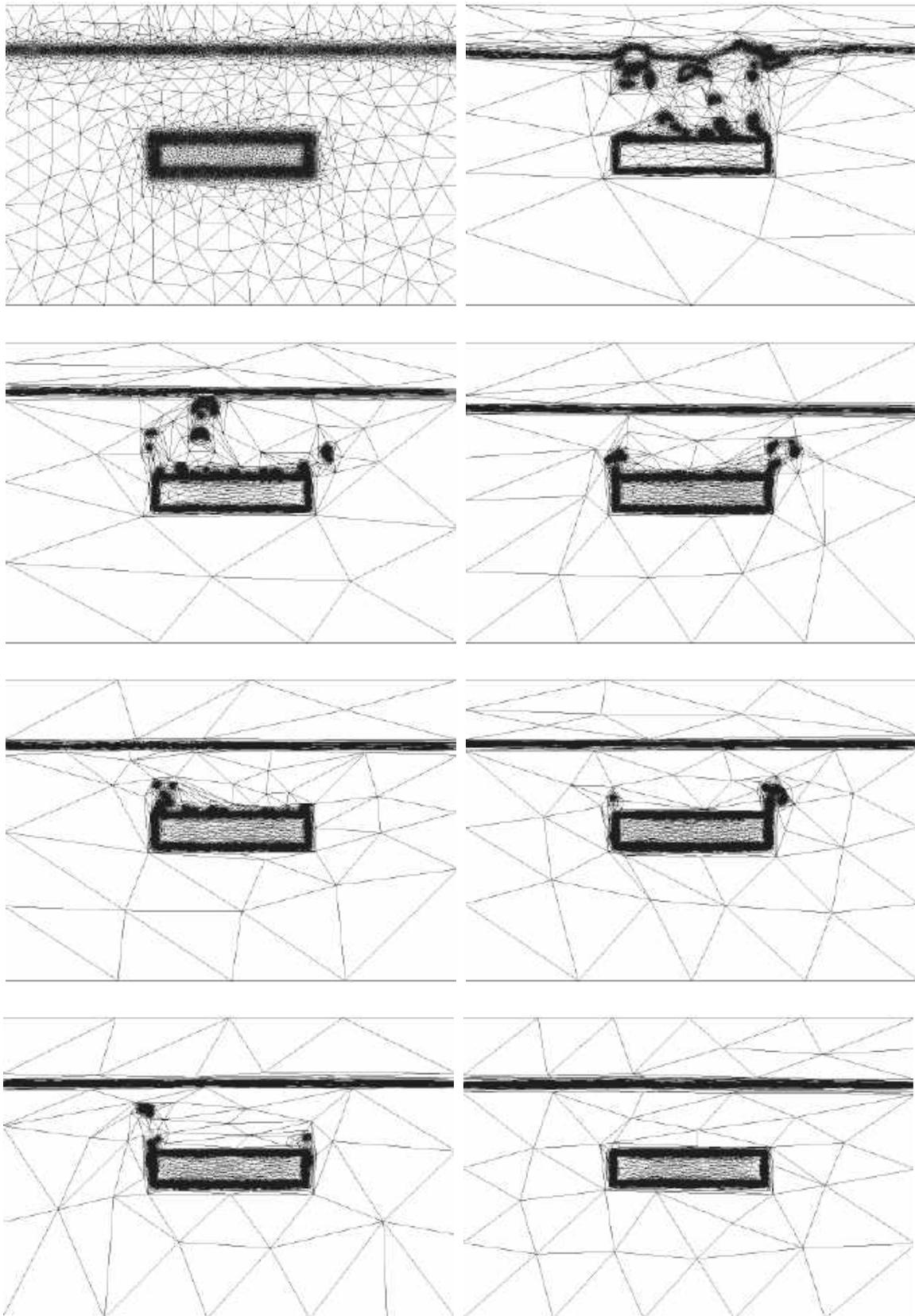


Figure 4.14: Meshes and liquid/vapor interface location at times $t = 0, 1, 6, 25, 40, 60, 70, 90, 120$ s

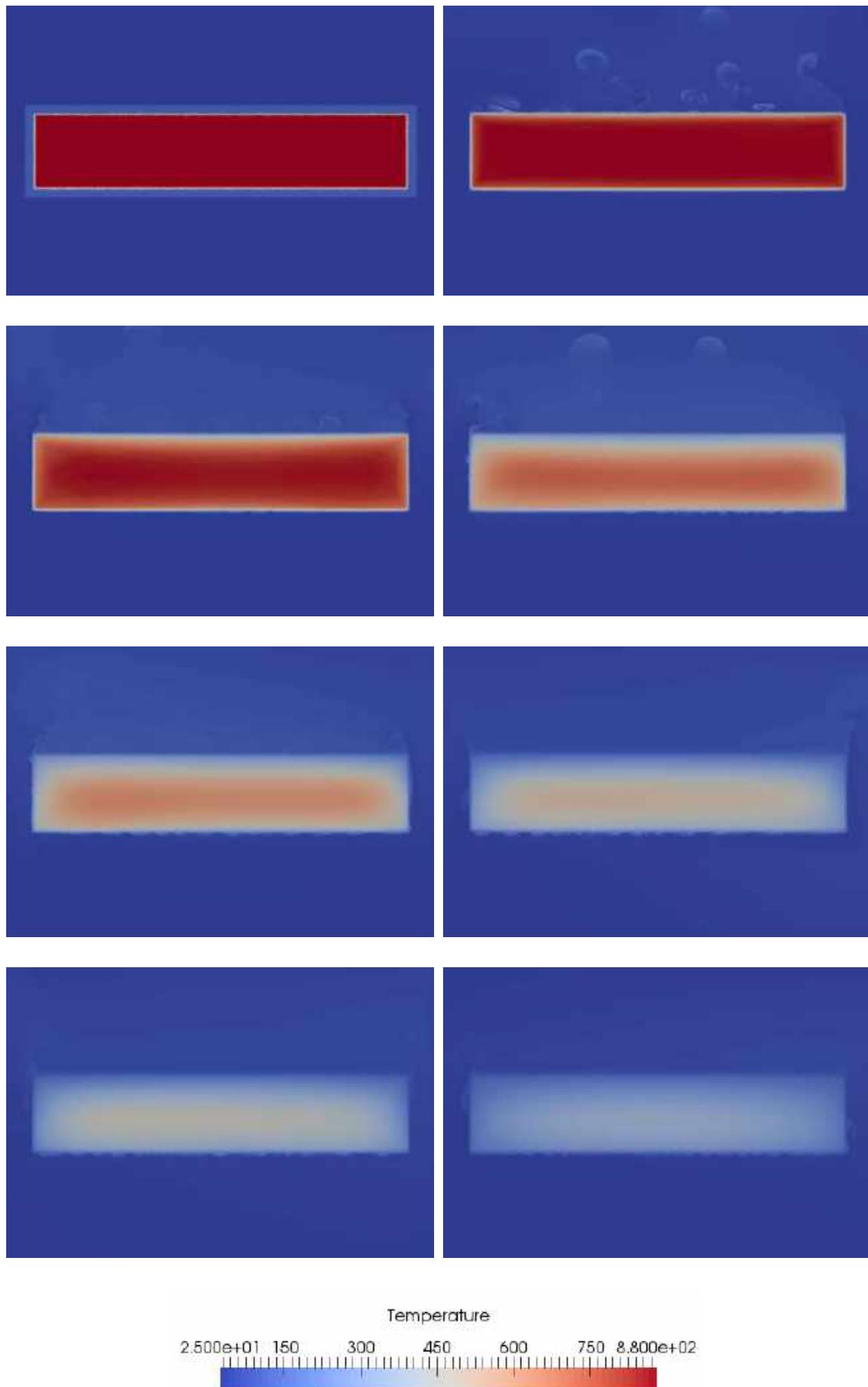


Figure 4.15: Temperature evolution and liquid/vapor interface location at times $t = 0, 1, 6, 25, 40, 60, 70, 90, 120$ s

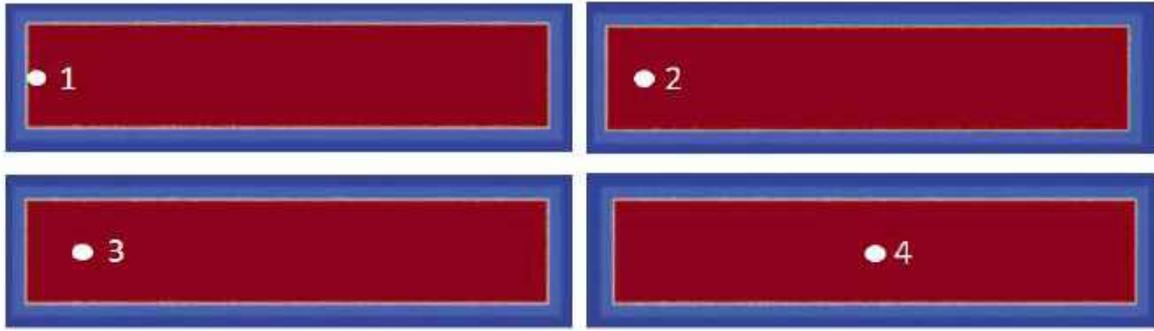


Figure 4.16: Sensors positions inside the solid part

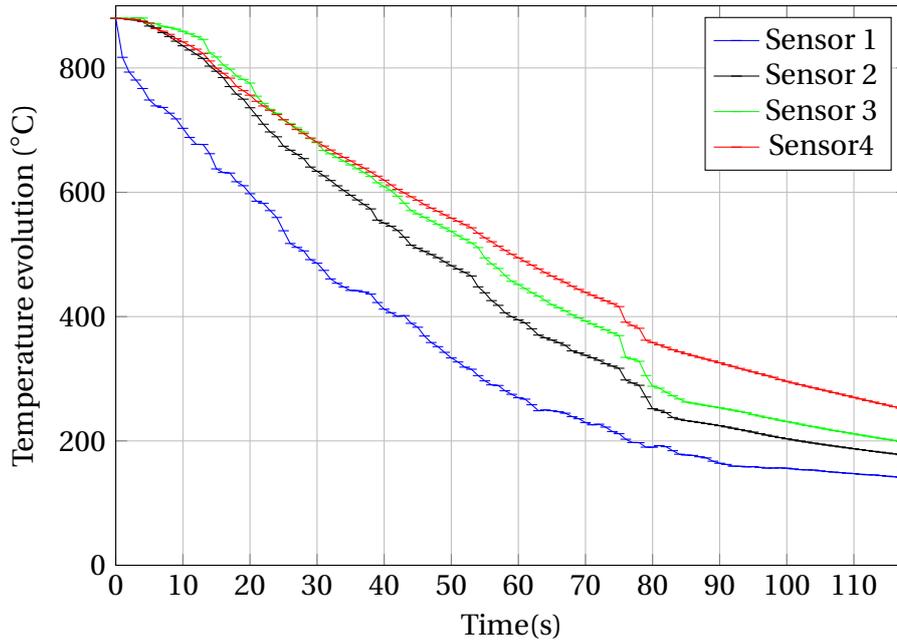


Figure 4.17: Evolution of the temperature inside the solid for different positions

4.6 Conclusion

This Chapter was dedicated to the boiling multiphase applications that involves liquid-vapor-solid interactions. We proposed here a new conservative Eulerian framework for the numerical simulations of multiphase applications that involve a phase change. We have first presented the formulation of a two-phase flow model with thermal exchanges, surface tension and phase change. To do so, we first explained the derivation of the equations for multiphase applications without phase change, then, we explained how to model the phase change and how to integrate it a mathematical framework. We detailed the integration of the phase change terms and derived the mass transfer rate at the liquid/vapor interface. The dynamics of the vapor phase is given by the surface tension, implemented in a semi-implicit way to circumvent the usual severe capillary time step restriction, the computational cost is therefore drastically reduced. We adopted a finite element discretization of the computational domain for its accuracy and efficiency in solving 3D problems with complex geometries. However the standard Galerkin formulation fails to produce an oscillation free solution in convection or diffusion dominated problems especially in the vicinity of sharp gradients. Thus, the Streamline Upwind Petrov-Galerkin (SUPG) method was introduced to deal with convection dominated convection-

diffusion-reaction problems. This method eliminates the instabilities in the solution by adding artificial diffusion terms in the convection direction. Finally, the accuracy of the methods was studied on two and three dimensional problems. The results obtained with the proposed stabilization tools were in very good agreement with the ones found in the literature. These examples consist of a fully coupled system of fluid flows and heat transfer involving phase changes at the interface. More challenging computations, including comparison with experimental data are performed in the next chapter. We will show that the direct numerical simulation of industrial water quenching using the approach developed in the all the previous chapters can be performed accurately.

4.7 Résumé du chapitre en français

Ce chapitre est consacré aux applications multiphasiques impliquant des interactions liquide-vapeur-solide. Nous avons proposé ici un nouveau cadre eulérien pour les simulations numériques d'applications multiphasiques impliquant un changement de phase. Nous avons formulé un modèle d'écoulement à deux phases impliquant des échanges thermiques, de la tension de surface ainsi que du changement de phase. Pour ce faire, nous avons d'abord expliqué la dérivation des équations pour les applications multiphasiques sans changement de phase, puis nous avons expliqué comment modéliser le changement de phase et comment l'intégrer à un cadre mathématique. Nous avons détaillé l'intégration des termes de changement de phase et déduit le taux de transfert de masse à l'interface liquide/vapeur. La dynamique de la phase vapeur est déterminée par la tension de surface superficielle, implémentée de manière semi-implicite afin de contourner la limitation habituelle du pas de temps capillaire, ainsi le coût de calcul est considérablement réduit. En ce qui concerne la discrétisation du domaine de calcul, nous avons adopté une méthode éléments finis pour sa précision et son efficacité dans la résolution de problèmes impliquant des géométries complexes en 3D. Cependant, la formulation Galerkin standard ne permet pas d'obtenir une solution numérique sans oscillations lorsque les phénomènes de convection et diffusion sont importants. Ainsi, la méthode Streamline Upwind Petrov-Galerkin (SUPG) a été introduite afin de résoudre les problèmes de convection-diffusion-réaction à dominante convective. Cette méthode élimine les instabilités de la solution en ajoutant des conditions de diffusion artificielles dans la direction de convection. Enfin, la précision des méthodes a été étudiée sur des cas tests 2D et 3D. Les résultats obtenus avec les outils numériques proposés sont en accord avec ceux trouvés dans la littérature. Ces exemples consistent en un système entièrement couplé d'écoulements de fluide et de transfert de chaleur impliquant des changements de phase à l'interface. Ensuite, des cas tests plus complexes, y compris la comparaison avec des données expérimentales seront effectués dans le Chapitre suivant. Nous montrerons que la simulation numérique directe de la trempe industrielle à l'eau en utilisant l'approche développée dans les chapitres précédents peut être effectuée avec précision.

4.8 Bibliography

- [1] S. W. Welch, Local simulation of two-phase flows including interface tracking with mass transfer, *Journal of Computational Physics* 121 (1) (1995) 142–154. [88](#)
- [2] S. W. Welch, J. Wilson, A volume of fluid based method for fluid flows with phase change, *Journal of computational physics* 160 (2) (2000) 662–682. [88](#), [98](#), [99](#)

- [3] G. Son, V. K. Dhir, Numerical simulation of film boiling near critical pressures with a level set method, *Journal of Heat Transfer* 120 (1) (1998) 183–192. [88](#)
- [4] V. K. Dhir, Mechanistic prediction of nucleate boiling heat transfer—achievable or a hopeless task?, *Journal of Heat Transfer* 128 (1) (2006) 1–12. [88](#)
- [5] G. Son, V. K. Dhir, N. Ramanujapu, Dynamics and heat transfer associated with a single bubble during nucleate boiling on a horizontal surface, *Journal of Heat Transfer* 121 (3) (1999) 623–631.
- [6] G. Son, N. Ramanujapu, V. K. Dhir, Numerical simulation of bubble merger process on a single nucleation site during pool nucleate boiling, *Journal of heat transfer* 124 (1) (2002) 51–62.
- [7] G. Son, V. K. Dhir, A level set method for analysis of film boiling on an immersed solid surface, *Numerical Heat Transfer, Part B: Fundamentals* 52 (2) (2007) 153–177. [88](#)
- [8] J. Liu, C. M. Landis, H. Gomez, T. J. Hughes, Liquid–vapor phase transition: Thermo-mechanical theory, entropy stable numerical formulation, and boiling simulations, *Computer Methods in Applied Mechanics and Engineering* 297 (2015) 476–553. [88](#)
- [9] J. Bueno, H. Gomez, Liquid-vapor transformations with surfactants. phase-field model and isogeometric analysis, *Journal of Computational Physics* 321 (2016) 797–818.
- [10] J. Bueno, C. Bona-Casas, Y. Bazilevs, H. Gomez, Interaction of complex fluids and solids: theory, algorithms and application to phase-change-driven implosion, *Computational Mechanics* 55 (6) (2015) 1105–1118. [88](#)
- [11] M. Khalloufi, Multiphase flows with phase change and boiling in quenching processes, Ph.D. thesis, PSL Research University (2017). [89](#), [94](#), [95](#)
- [12] Y.-C. Chang, T. Hou, B. Merriman, S. Osher, A level set formulation of eulerian interface capturing methods for incompressible fluid flows, *Journal of computational Physics* 124 (2) (1996) 449–464. [94](#)
- [13] T. E. Tezduyar, M. Behr, S. Mittal, J. Liou, A new strategy for finite element computations involving moving boundaries and interfaces—the deforming-spatial-domain/space-time procedure: Ii. computation of free-surface flows, two-liquid flows, and flows with drifting cylinders, *Computer methods in applied mechanics and engineering* 94 (3) (1992) 353–371. [94](#)
- [14] J. U. Brackbill, D. B. Kothe, C. Zemach, A continuum method for modeling surface tension, *Journal of computational physics* 100 (2) (1992) 335–354. [94](#)
- [15] D. Kothe, W. Rider, S. Mosso, J. Brock, J. Hochstein, Volume tracking of interfaces having surface tension in two and three dimensions, in: *34th Aerospace Sciences Meeting and Exhibit*, 1996, p. 859. [94](#)
- [16] Y. Renardy, M. Renardy, Prost: a parabolic reconstruction of surface tension for the volume-of-fluid method, *Journal of computational physics* 183 (2) (2002) 400–421.
- [17] D. Lörstad, M. Francois, W. Shyy, L. Fuchs, Assessment of volume of fluid and immersed boundary methods for droplet computations, *International journal for numerical methods in fluids* 46 (2) (2004) 109–125. [94](#)

- [18] M. Kang, R. P. Fedkiw, X.-D. Liu, A boundary condition capturing method for multiphase incompressible flow, *Journal of Scientific Computing* 15 (3) (2000) 323–360. [94](#)
- [19] M. Sussman, A second order coupled level set and volume-of-fluid method for computing growth and collapse of vapor bubbles, *Journal of Computational Physics* 187 (1) (2003) 110–136. [94](#)
- [20] G. Tryggvason, B. Bunner, A. Esmaeeli, D. Juric, N. Al-Rawahi, W. Tauber, J. Han, S. Nas, Y.-J. Jan, A front-tracking method for the computations of multiphase flow, *Journal of computational physics* 169 (2) (2001) 708–759. [94](#)
- [21] S. Popinet, S. Zaleski, A front-tracking algorithm for accurate representation of surface tension, *International Journal for Numerical Methods in Fluids* 30 (6) (1999) 775–793. [94](#)
- [22] S. Hysing, A new implicit surface tension implementation for interfacial flows, *International Journal for Numerical Methods in Fluids* 51 (6) (2006) 659–672. [94](#)
- [23] M. Raessi, M. Bussmann, J. Mostaghimi, A semi-implicit finite volume implementation of the csf method for treating surface tension in interfacial flows, *International journal for numerical methods in fluids* 59 (10) (2009) 1093–1110.
- [24] G. C. Buscaglia, R. F. Ausas, Variational formulations for surface tension, capillarity and wetting, *Computer Methods in Applied Mechanics and Engineering* 200 (45-46) (2011) 3011–3025. [94](#)
- [25] R. Codina, Comparison of some finite element methods for solving the diffusion-convection-reaction equation, *Computer Methods in Applied Mechanics and Engineering* 156 (1-4) (1998) 185–210. [96](#)
- [26] S. Badia, R. Codina, Analysis of a stabilized finite element approximation of the transient convection-diffusion equation using an ale framework, *SIAM Journal on Numerical Analysis* 44 (5) (2006) 2159–2197. [97](#)
- [27] R. Codina, J. M. González-Ondina, G. Díaz-Hernández, J. Principe, Finite element approximation of the modified boussinesq equations using a stabilized formulation, *International Journal for Numerical Methods in Fluids* 57 (9) (2008) 1249–1268. [96](#)
- [28] T. J. Hughes, G. R. Feijóo, L. Mazzei, J.-B. Quincy, The variational multiscale method—a paradigm for computational mechanics, *Computer methods in applied mechanics and engineering* 166 (1-2) (1998) 3–24. [97](#)
- [29] A. N. Brooks, T. J. Hughes, Streamline upwind/ Petrov-galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations, *Computer methods in applied mechanics and engineering* 32 (1-3) (1982) 199–259. [97](#)
- [30] A. C. Galeão, E. G. D. Do Carmo, A consistent approximate upwind Petrov-galerkin method for convection-dominated problems, *Computer Methods in Applied Mechanics and Engineering* 68 (1) (1988) 83–95. [97](#)
- [31] A. Esmaeeli, G. Tryggvason, Computations of film boiling. part i: numerical method, *International journal of heat and mass transfer* 47 (25) (2004) 5451–5461. [98](#)

- [32] M. W. Akhtar, S. J. Kleis, Boiling flow simulations on adaptive octree grids, *International Journal of Multiphase Flow* 53 (2013) 88–99. [98](#)
- [33] Y.-j. Lao, R. E. Barry, R. E. Balzhiser, A study of film boiling on a horizontal plate, in: *International Heat Transfer Conference 4*, Vol. 24, Begel House Inc., 1970. [98](#)
- [34] P. J. Berenson, Film-boiling heat transfer from a horizontal surface, *Journal of Heat Transfer* 83 (3) (1961) 351–356. [100](#)
- [35] V. Klimenko, Film boiling on a horizontal plate—new correlation, *International journal of heat and mass transfer* 24 (1) (1981) 69–79. [101](#)

Chapter 5

Coupling and industrial applications

Contents

5.1 Introduction	116
5.2 Coupling between Cimlib-CFD and Z-set	117
5.3 Industrial applications	120
5.3.1 Quenching of a solid and comparison to experimental data	120
5.3.1.1 Set up	121
5.3.1.2 Results and discussions	122
5.3.2 Quenching of Cylinder head	127
5.3.2.1 Set up	128
5.3.2.2 Results and discussions	129
5.3.3 Challenging geometry : crossmember of a car	132
5.3.3.1 Set up	133
5.3.3.2 Results and discussions	134
5.4 Conclusion	140
5.5 Résumé du chapitre en français	140
5.6 Bibliography	141

5.1 Introduction

In this chapter, we are interested in evaluating the developed numerical tools in resolving large scale and long time computations. More precisely, we aim at validating the potential of the conservative eulerian framework to deal with complex 3D geometries. The study will include the study of thermal cooling and turbulent boiling inside quenching tanks. We recall that this work was developed in the context of the Linamar Montupet project with the main objective to develop a conservative eulerian framework to perform realistic simulations of quenching process at an industrial scale. As mentioned previously, the objective of this thesis is divided in two parts. First of all, propose a quenching model that accounts for turbulent boiling, phase change and natural convection; and all the different components of the eulerian conservative framework have been presented in the previous chapters of this manuscript. We depicted in Figure 5.1 the general diagram for the simulation of the quenching process. As explained previously, the simulation is very challenging as it involves solving simultaneously heat transfers between the solid body and the surrounding fluid. Recall that prior to this work, finite element stabilized solvers for the modelling of the quenching process were developed within CEMEF [1]. However, with these tools, the thermomechanical behavior and characteristic of the solid part were not explored. Thus, the second objective of this thesis is to develop a generalist "platform" between two codes capable of accurately transferring fields from a fluid-structure mesh to a structure mesh; in order to perform a residual stress analysis. Indeed, a coupling is necessary to account for the thermomechanical behavior and characteristic of the solid part. Indeed, residual stresses are the consequences of heat treatment, deformation processes, etc. They develop when a body undergoes inhomogeneous plastic deformation or is exposed to a non-uniform temperature distribution such as in the case of quenching processes. These residual stresses may induce premature failure through cracking, reduce fatigue strength, or cracking, and cause distortion and dimensional variation during subsequent machining. As a result, machined parts may be out of tolerance and have to be cold worked or re-machined. Therefore, the investigation and understanding of residual stress prediction and reduction is important to improve the quality and reliability of many manufactured parts. Prediction of residual stresses with analytical or numerical methods has been studied extensively since experimental measurement of residual stresses is very lengthy, tedious and expensive. Due to its versatility, accuracy and efficiency, finite element analysis technique is found to be a viable and cost-effective alternative.

The close proximity between our research work and the industrial world obliges us to deal with more and more complex simulations involving challenging geometries. Thus, this chapter is dedicated to the coupling of the two codes and the industrial validation. Our industrial partner is using the software Z-set, dedicated to the solid and is able to simulate the heat treatment of mechanical pieces with phase change and analyze the residual stresses. First, we will present the method used for the coupling between the two codes and then we will study complex industrial cases involving high thermal gradients. We have taken into account the real setup in terms of full geometric design of the problem, the material properties of the workpieces, and the true initial and boundary conditions. The first test case will enable to assess the accuracy and robustness of the proposed approach to help us closing the bridge between experiment and numerical simulation. The next test case will show the ability of the framework to handle very complex geometries and more challenging configurations. Finally, the last test case will show the ability of this numerical framework to be used as predictive tool for the industry. It will help us reach

the most optimal quenching process in terms of the orientation of the geometry.

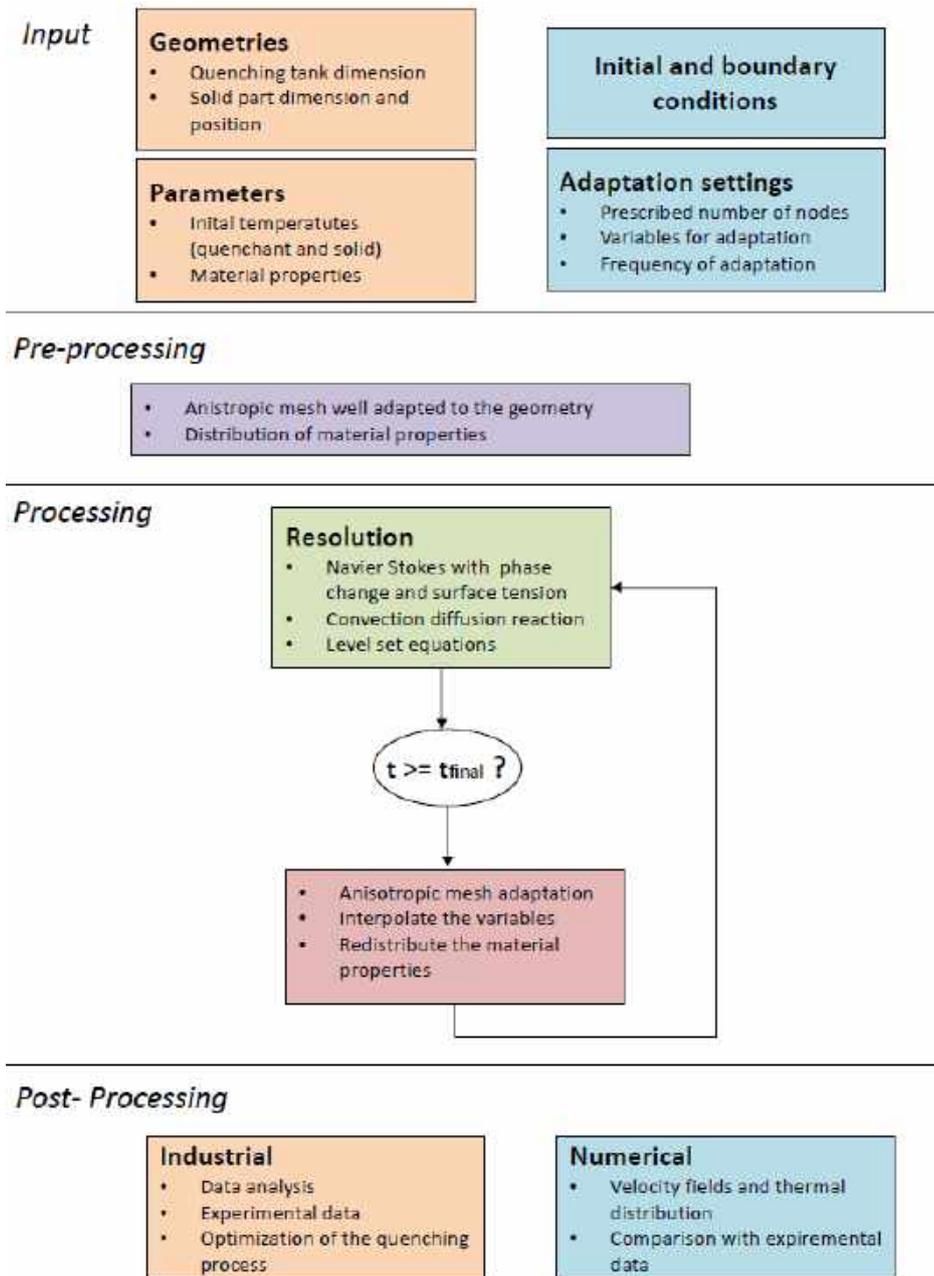


Figure 5.1: General diagram for a quenching process simulation

5.2 Coupling between Cimlib-CFD and Z-set

The initial approach is to consider two domains, see Figure 5.2. First of all, the fluid-solid domain in which the solid is totally immersed in the fluid. For that, we will refer to the use of our immersed volume framework that has been developed in the previous chapters. Indeed, a full Eulerian framework that simulates the quenching process has been established. It takes into account three ingredients that can be resumed to : (i) geometric : flexibility for multidomain simulation, (ii) fluid mechanics: accounting for turbulent boiling, (iii) physics : phase change. As for the solid part, we recall that the quenching leads to important deformations such as cracks and defects inside the solid, so we want to study

the effect of the residual stresses. The first step of this project consists in transferring the

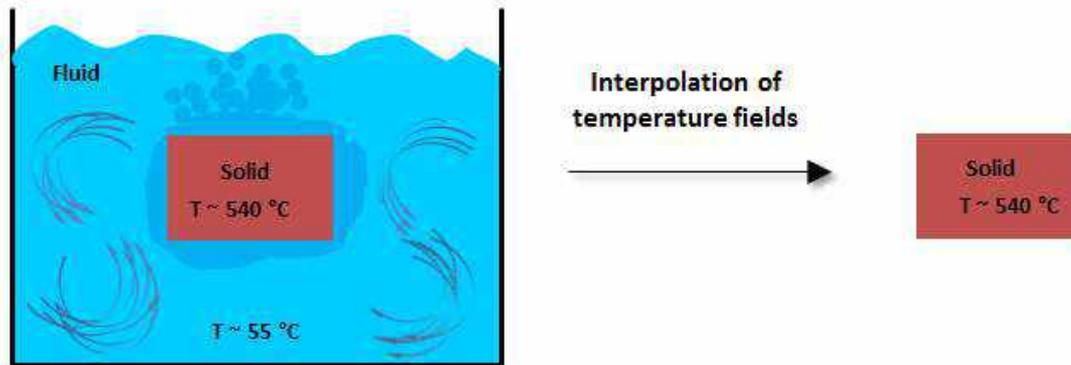


Figure 5.2: Division into two domains : Fluid-Solid and Solid

fields of the first fluid-solid domain to the solid domain. The software Z-set uses only the solid geometry, and since the two domains are completely different, it requires exchanging and treating information based on two different computational grids. Such transfers have to be thought to maintain numerical accuracy. However, the way data are exchanged between the two codes plays an important role in the efficiency of the coupling. In fact, transferring data from one interface to another interface is not straightforward and solutions are not clear.

We can distinguish two types of coupling. The first one is the weak coupling, where the two codes are coupled in an iterative and not time accurate manner. The second one is the strong coupling, which is time accurate since exchanges between the two codes are done at each time step. However, such technique is very heavy to implement, therefore we have first considered indirect coupling approaches which are easier to manage. This so-called weak coupling procedure was widely applied by several researchers between dynamics codes. In the weak coupling approach, the communication is performed by exchanging files between Cimlib-CFD and Z-set. The iterative coupling scheme developed between Cimlib-CFD and Z-set is depicted in Figure 5.3. The different steps towards the coupling are explained here :

1. First of all, a simulation of the quenching process is performed using our library Cimlib-CFD. Such simulation is performed using a numerical multiscale framework that uses an implicit representation of the solid part and of course is able to predict precisely the liquid-to-vapor transition during boiling. But here no solid phase transformation is taken into account.
2. At the end of the computation, the solid part is completely cooled. And thus we have the whole thermal history of the metal part. Recall, that we are using an immersed volume method, thus the solid geometry is completely immersed in the fluid domain and the solid interface is implicit. We are only interested here, in the thermal history of the solid part, thus we must transfer the temperature fields on a completely unrelated mesh of the solid geometry. This step is done using the conservative interpolation algorithm presented in Chapter 2.
3. Now that we have, the temperature fields on the solid geometry, we import the solid mesh and the temperature fields to Z-set. Thus, the files that are sent are : (i) a mesh

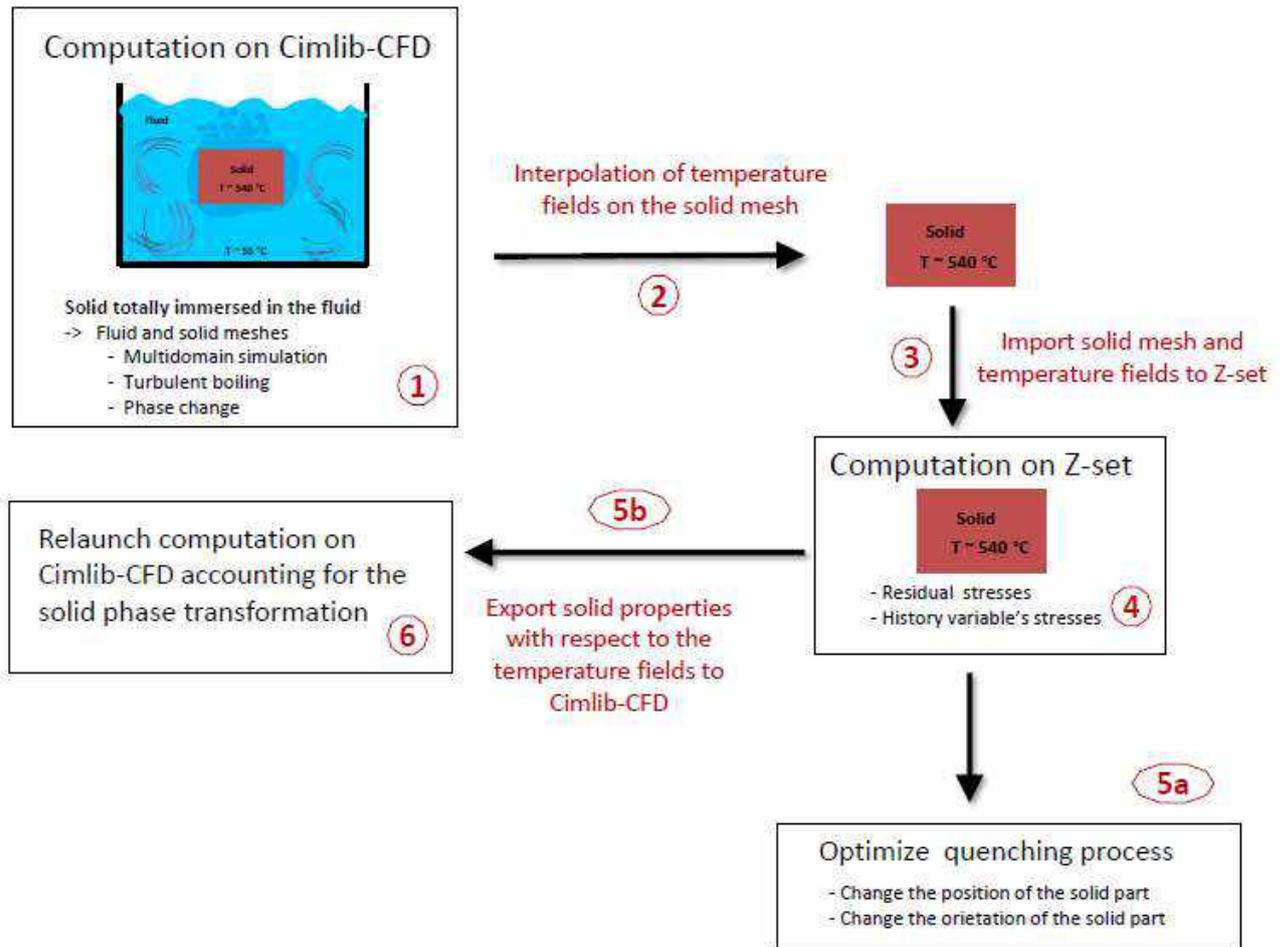


Figure 5.3: Coupling scheme between Cimlib-CFD and Z-set

file in our Cimlib-CFD native format; it contains the number of nodes and elements, then a list of the coordinates and finally a list of the connectivities, (ii) then a text file that contains the temperature fields with respect to the corresponding nodes. Note that the frequency of exchanging files is high at the beginning during the rapid phase of the cooling process since. Indeed, in order to obtain an accurate residual stresses analysis, we need many discretization points as possible.

4. We have been working with the Z-set team in Paris, and now the software is provided with a data reader that is able to read a mesh file in the native Cimlib-CFD format. The prediction of residual stresses is now possible using the software Z-set, and such study will be performed by our industrial partner. The objective of this thesis was only to implement the coupling between the two codes.
5. Now two outcomes are possible :
 - (a) The analysis of the residual stress allows a better understanding of the quenching process. Since, the industrial are interested in the optimization of the quenching process, performing a residual stress analysis will allow us to know which is the optimal orientation and positioning of the solid part in the quenching tank. For instance, depending on the orientation of the geometry, a deformation/distortion might occur and this can be avoided if the part is oriented in the right way, and this can be predicted by doing a preliminary residual stress

analysis.

- (b) We are now able to predict the solid part metallurgical evolution. Indeed, the microstructure of the material is inhomogeneous in the part due to the difference of cooling rate in the different area of a part. Thus, this metallurgical history (density, conductivity, heat capacity...) is exported from Z-set to Cimlib-CFD. These variable's history is written in text files and are now ready for a coupling with Cimlib-CFD.
6. The thermomechanical behavior and characteristic of the solid part are now taken into account. A new computation of the quenching process accounting for the solid phase transformation can now be performed.

5.3 Industrial applications

In this section, three industrial applications will be presented. The first test case is a ring, here the numerical results will be compared with experimental data proposed by the industrial partner Linamar Montupet. A complex engine head geometry will then be studied, it is a cylinder head of a car. It is a challenging geometry since the interior regions of the quenched part is composed of holes. The last industrial test case is a more challenging geometry, it is a crossmember of a car. This is very complex test case, the geometry is very thin and hollow inside, thus the vapor gets stuck inside the solid.

5.3.1 Quenching of a solid and comparison to experimental data

The first test case is a Ring geometry. The geometry is first defined in a CAD environment

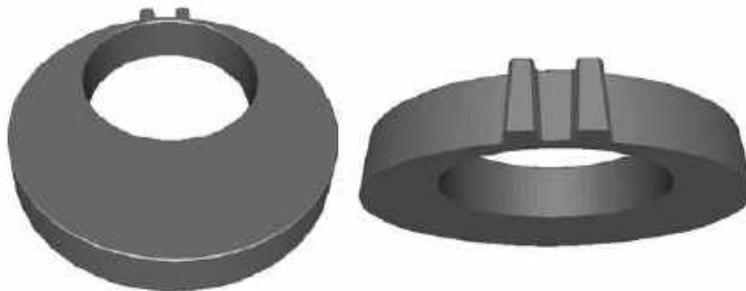


Figure 5.4: Ring geometry

(see Figure 5.4). It is a hollow cylinder, that presents some sharp corners, which can create singularities. After cleaning the CAD geometry (eliminating small details, trimming overlapping surfaces,...), a surface mesh of the object is generated using an STL mesher. The latter is made up from segments in 2D simulations and from triangles in 3D simulations. The surface mesh is then embedded in the computational domain which is in turn remeshed resulting in a single fluid/solid mesh. An accurate representation of the body is obtained through successive mesh adaptations. Figure 5.5 provides a schematic representation of the mesh generation process accounting for the immersed Ring geometry. The fluid/solid interface is perfectly represented by the anisotropic mesh adaptation.

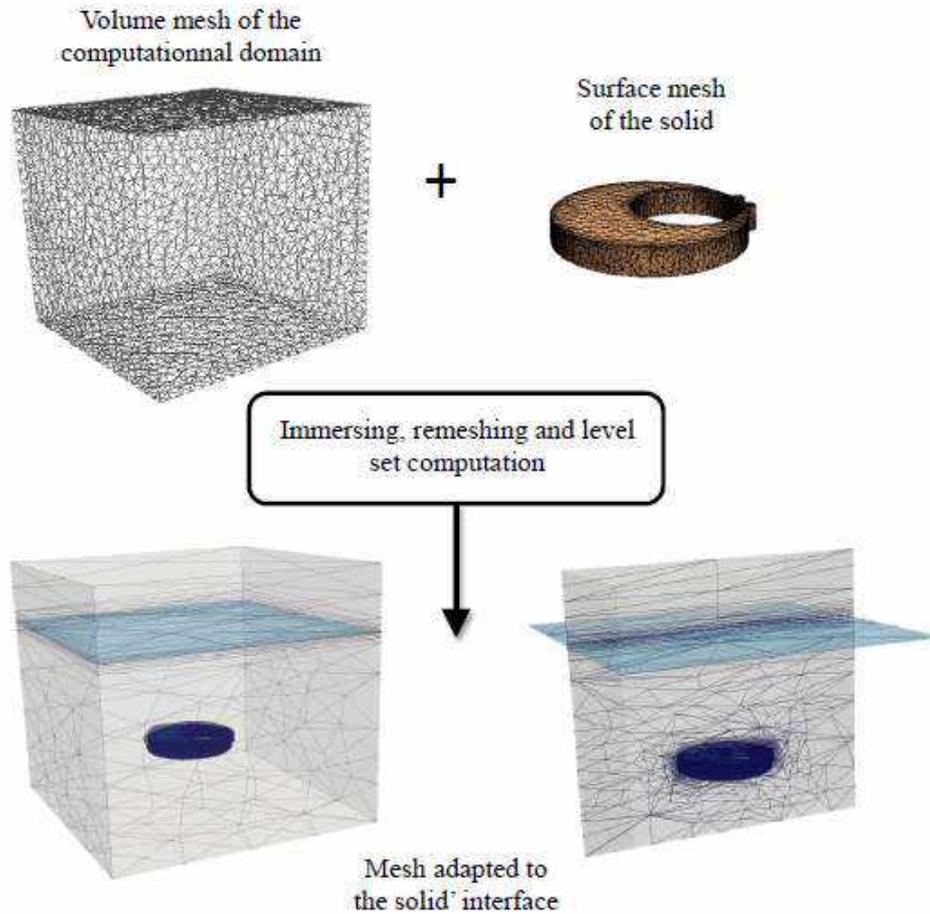


Figure 5.5: Immersion of the solid geometry in the fluid domain

5.3.1.1 Set up

The set up of this test case is shown in Figure 5.6. The solid part is quenched in a water medium. The domain has a size of $0.4 \times 0.4 \times 0.3 m^3$ and is three-quarters full of water. The solid part has a diameter of $125 mm$ and is quenched in a horizontal position at the middle of the tank. The solid part is made of cast aluminum alloy. The water has an initial temperature of $T_{\text{water}} = 55^\circ C$ and the solid part $T_{\text{solid}} = 495^\circ C$. A free slip boundary condition is prescribed on all the walls. Since, a vapor film surrounds instantaneously the part, simulating the creation of the vapor film would be very time consuming, thus, we define a vapor film surrounding the solid part from the beginning of the computation. This vapor film has a thickness of millimeters and is defined with respect to the complexity of the geometry. The motion of the vapor film is due to the buoyancy and the surface tension forces. Two simulations have been carried out, one considering constant solid properties (see table 5.1) and the other one, considering a transformation phase inside the solid. Indeed, most of the metallic alloys experience phases transformation, meaning that the microstructure of the material is inhomogeneous in the part due to the difference of cooling rate in the different area of a part. This leads to variation, sometimes large, of the thermal and mechanical properties of the solid. Thus, the solid parameters used for the second simulation are depicted in table 5.2. A total number of 800000 elements have been used for both simulations. The time step is set to $\Delta t = 1 \cdot 10^{-3} s$ for the first simulation and $\Delta t = 1 \cdot 10^{-4} s$ for the second test case. These simulations were launched on 28 pro-

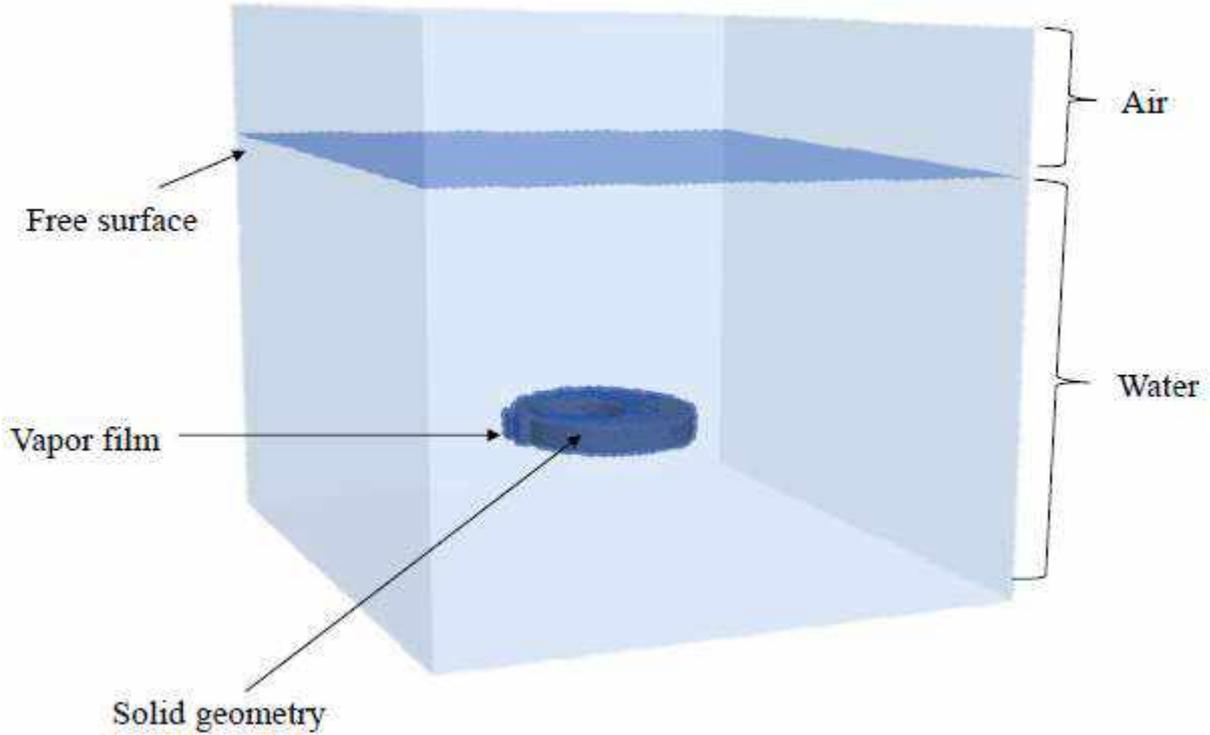


Figure 5.6: Set up for the simulation of the quenching of a Ring geometry

	$\mu(\text{Pa} \cdot \text{s})$	$\rho(\text{kg} \cdot \text{m}^{-3})$	$c_p(\text{J} \cdot \text{kg}^{-1} \cdot \text{K}^{-1})$	$k(\text{W} \cdot \text{m}^{-1} \cdot \text{K}^{-1})$
Solid		2675	1045	174
Vapor	$1.2 \cdot 10^{-5}$	1.0	2010	0.025
Water	$1.0 \cdot 10^{-3}$	1000	4185	0.6

Table 5.1: Physical parameters used for first simulation of the quenching of a Ring

cessors each. In this simulation, as explained in Chapter 3, the Immersed volume method is used to immerse the solid object into the domain and distribute the material properties. A single set of equations is solved simultaneously for both fluid and solid domains. The variational multiscale approach is employed to stabilize the Navier Stokes equations, accounting for a semi implicit surface tension model (Chapter 4). The SUPG method is used to preclude numerical oscillations at the locations of convection domination and sharp gradients especially in the vicinity of the interface (Chapter 4). And finally, the conservative anisotropic mesh adaptation (Chapter 2) was applied. The material properties of the water and the solid are updated dynamically in accordance with their temperatures.

5.3.1.2 Results and discussions

The evolution of the liquid/vapor phase and the corresponding meshes are depicted in Figure 5.7. The solid is represented in purple and the vapor field in blue. As explained previously, the relative temperature difference is so high that the vaporization takes place directly at the interface between the vapor film and the liquid. At the beginning, small bubbles start to form, then the vapor bubbles start moving, coalesce and leave the sur-

$T_{\text{solid}}(^{\circ}\text{C})$	$\rho(\text{kg} \cdot \text{m}^{-3})$	$c_p(\text{J} \cdot \text{kg}^{-1} \cdot \text{K}^{-1})$	$k(\text{W} \cdot \text{m}^{-1} \cdot \text{K}^{-1})$
20	2675	919	154.2
100	2661	940	156.6
200	2642	965	163.5
300	2621	991	170.9
400	2599	1020	172.1
500	2576	1045	174.5

Table 5.2: Physical parametres of the immersed solid while considering the phase transformation

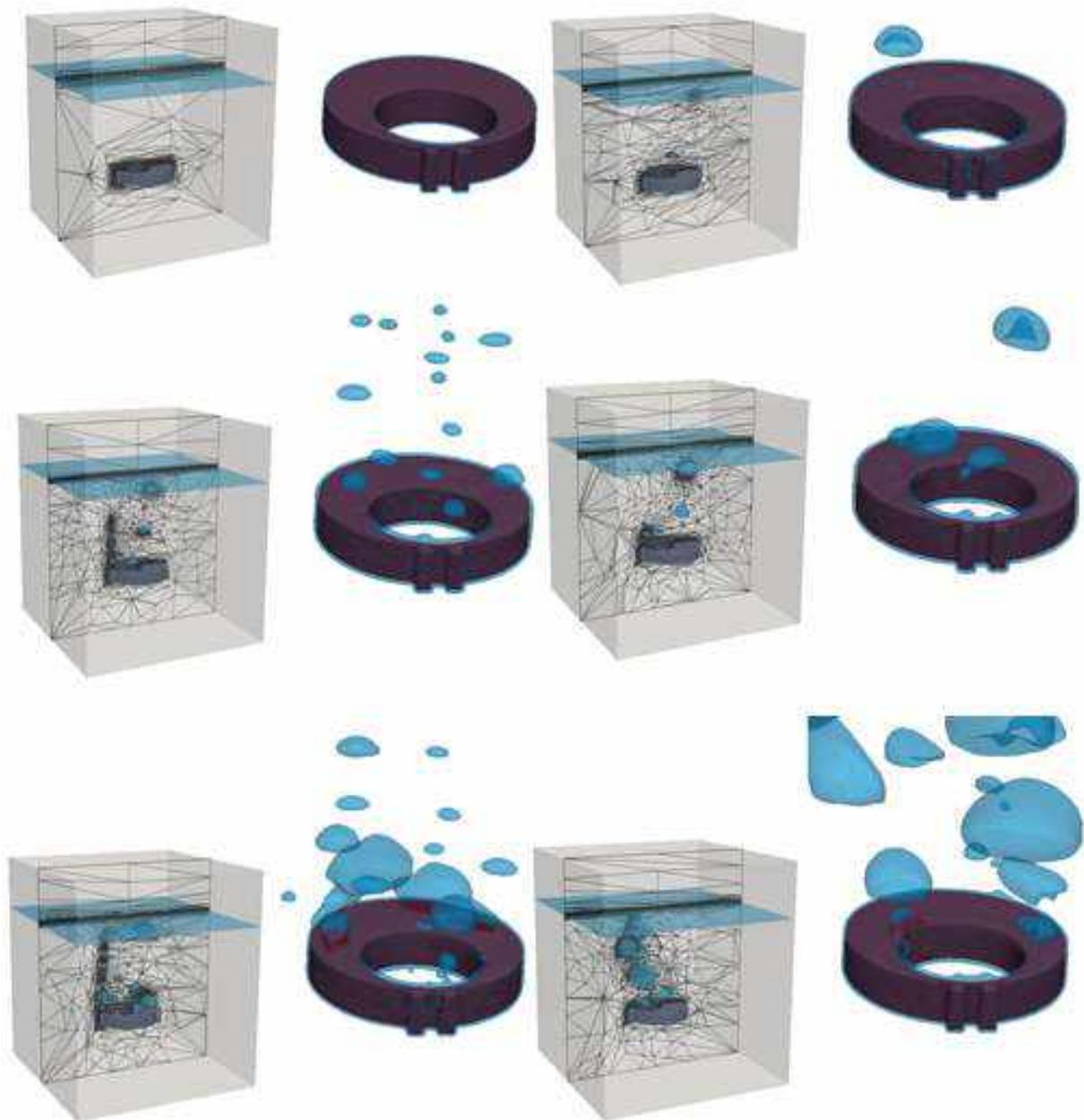


Figure 5.7: Evolution of the liquid/vapor interface during the boiling at times $t = 0, 1, 5, 7, 8$ s

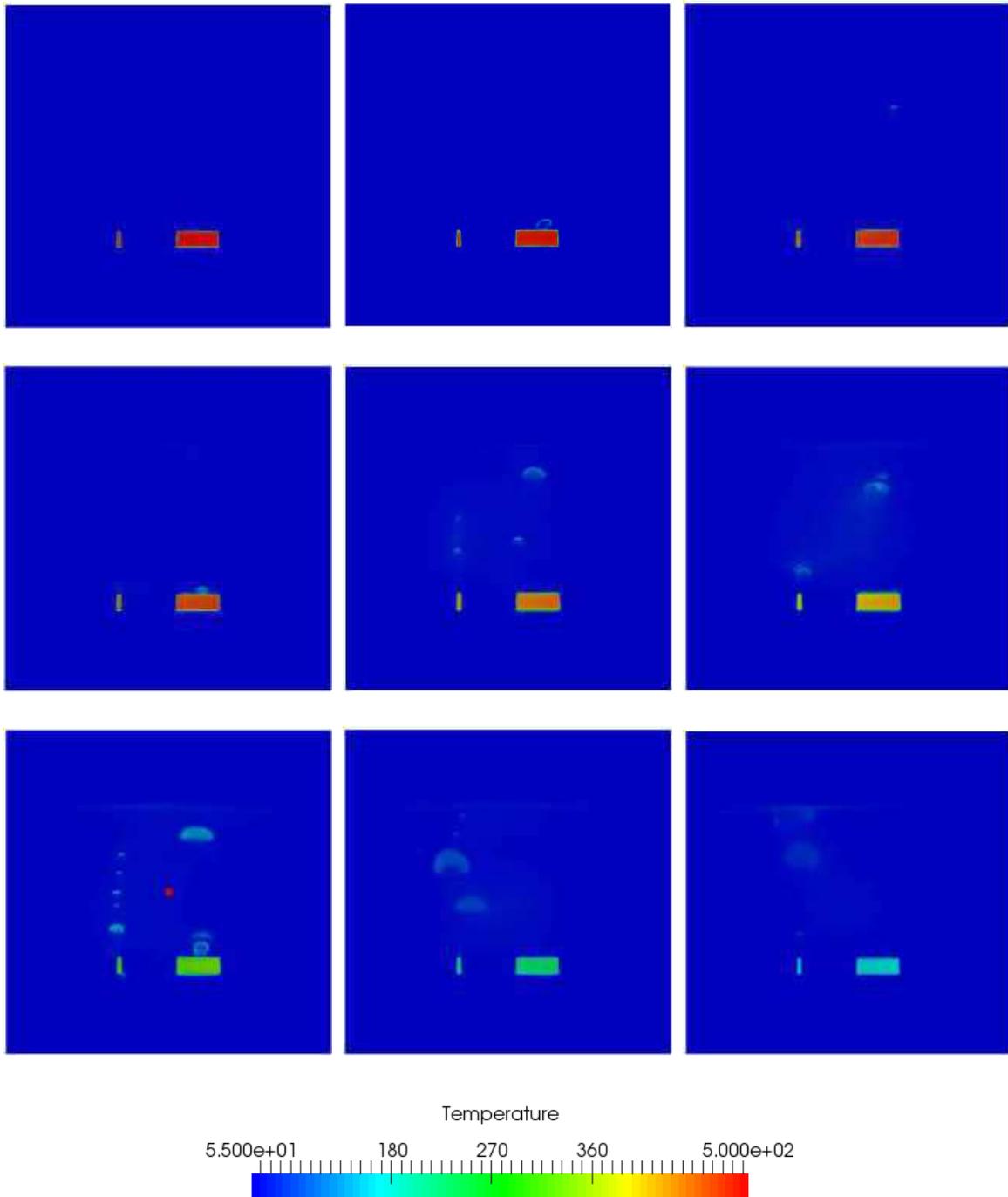


Figure 5.8: Simulation results. Temperature distribution along the sectional plane x at times $t = 0, 0.15, 1, 2, 5, 7.5, 10, 12, 15$ s.

face. Note here that the mesh is adapted to the level set function, the temperature field, the velocity norm and direction every time step, which underline the importance of the conservative interpolation when dealing with multiphase applications. We can clearly see how the mesh refinement is localized and responds to the characteristics of the problem. Indeed, at the beginning the elements are mostly concentrated inside and at the solid interface, and the free surface. As the boiling starts, bubbles start to form, the mesh is adapted automatically according to the level set interface but also the velocity and the temperature. Thus in the regions far from the liquid/vapor interface (at the bottom of the



Figure 5.9: Simulation results. Contour plot of temperature distribution on the solid domain at different times $t = 0, 0.15, 1, 2, 5, 7.5, 10, 12, 15$ s .

tank), the elements are coarsened. That is the ability of the algorithm since we are fixing the desired number of nodes, which automatically coarsens the elements far from the sharp variations of gradients. We report in Figure 5.8 the temperature distribution inside the aluminum workpiece at different time instances, we can observe how the solid cools down and how the heat is transferred between the two subdomains. In the beginning, the solid part is at 495°C , then the boiling start to happen, the temperature decreases, till the piece is cooled. For completeness, we reported in Figure 5.9 the distribution of the



Figure 5.10: Position of the thermocouples for the experimental analysis

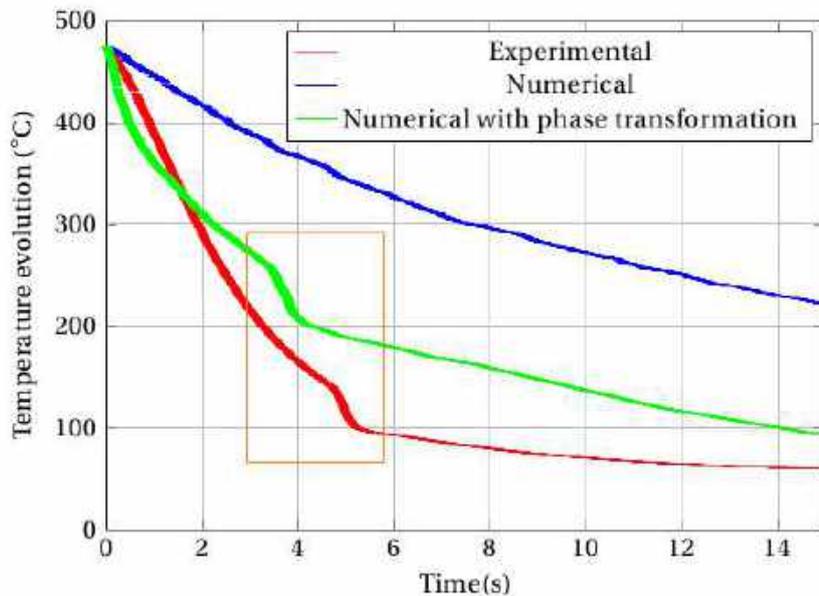


Figure 5.11: Comparison between experimental data and numerical results

nodal temperature fields at the surface of the solid part. Note that, the accuracy of the temperature field at the surface of the solid is linked to the choice of the thickness used for the mixing law. Therefore, to obtain accurate results, we must filter the level set function and choose an optimal mesh size near the interface. And also we recall that harmonic mean formulation enables to well render the sharp discontinuity in the thermal conductivity field. From a quantitative perspective, experimental analysis was also conducted on the Ring geometry which was equipped with several sensors at different positions inside

the work-piece as shown in Figure 5.10. The thermal history has been registered through a data acquisition device and provided to us by our industrial partner. We compare in Figure 5.11 the temperature variations at the sensor 1 (see Figure (5.12) with the results coming from both simulations : with and without considering the phase transformation inside the solid. An important point to mention here before discussing the results is that

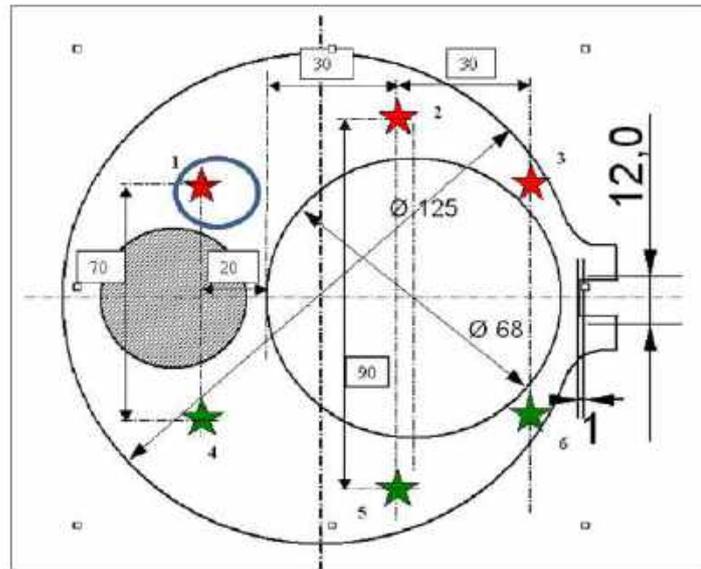


Figure 5.12: Position of sensors on the Ring geometry

in the experimental set up, agitators were used in the quenching tank, which explains the gap between the experimental data and numerical results. In fact, the use of agitators, will encourage the detachment of the vapor film and thus a faster cooling of the metal part. If we compare the results coming from the two simulations, we clearly see that when the transformation phase inside the solid is taken into account, the same temperature jump is observed. And this clearly shows the importance of the coupling between the two codes. Thus, despite some discrepancies observed between the simulation with the transformation phase and the measurement curve, the global results from the thermal calculations can be considered in good agreement with the experiment results.

5.3.2 Quenching of Cylinder head

The objective of this Chapter is to validate the performance of the developed conservative Eulerian framework on a challenging industrial case. Thus, in this section, a real cylinder head quench cooling process is presented. As it can be seen in Figure 5.13, it is a very complex geometry, since interior regions of the quenched part consisting of holes, passages with constrictions or expansion zones. We highlight the challenge in remeshing the domain as it includes curvature and sharp angles on its boundary. Inhere, the same approach have been use, the initial mesh was adapted only on the filtered level set function describing the geometry. The close-up snapshot in Figure 5.14 depict the layers of elements near the object boundaries. We can clearly detect the bandwidth across the interface where the elements are highly condensed. One can also observe the extremely stretched elements along the boundary of the immersed objects whereas the rest of the domain keeps almost the same background mesh size. We can clearly see that the mesh adaptation procedure allows us to handle complex geometries in terms of curvatures, sharp angles and singularities.

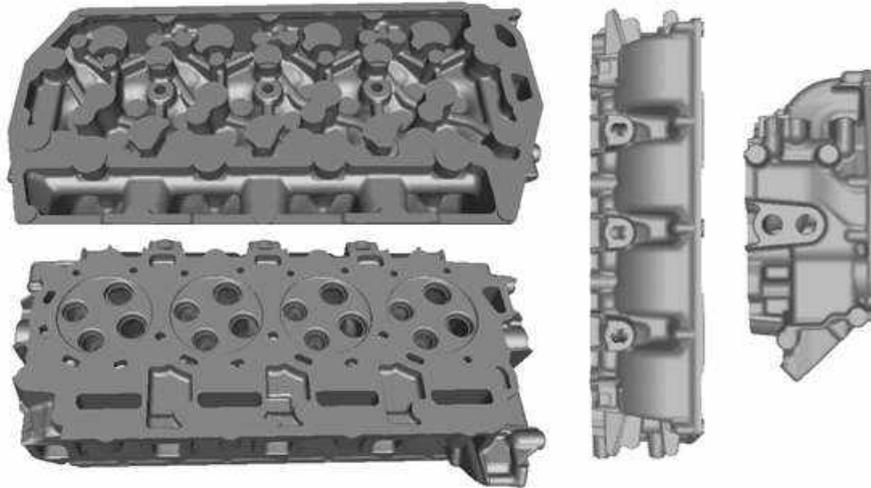


Figure 5.13: Cylinder head geometry

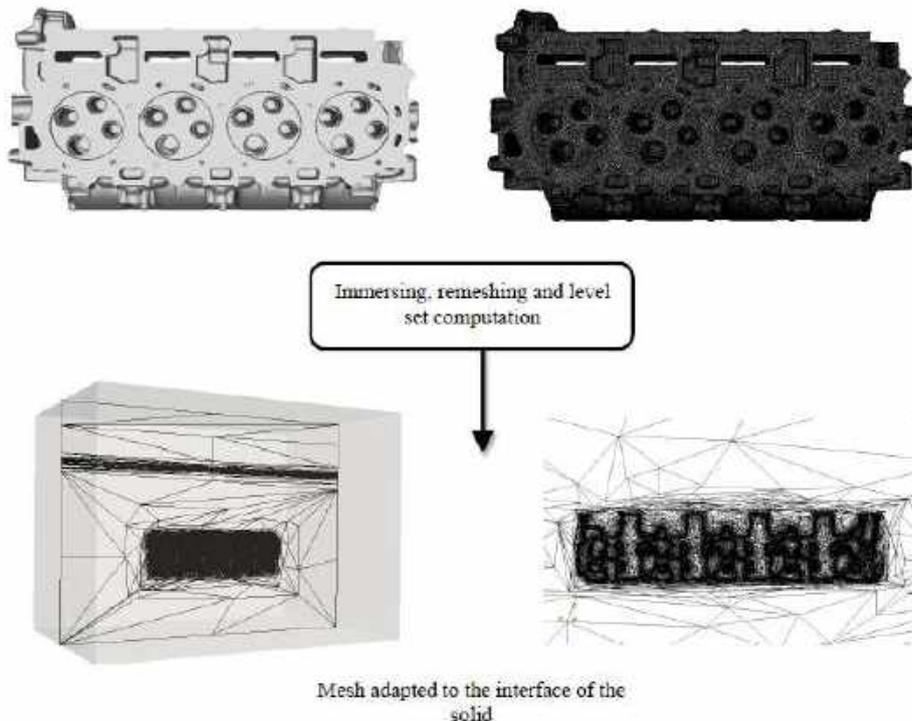


Figure 5.14: Immersion of the solid geometry in the fluid domain

5.3.2.1 Set up

The set up is shown in Figure 5.15. Water was used as the liquid quenchant. The computational domain has a size of $1.22 \times 0.9m^3$. And the geometry has an approximate size of $0.58 \times 0.33 \times 0.13m^3$. Here, the whole domain is full of water, and an outward velocity is defined at the top bottom wall. The water has an initial temperature of $T_{\text{water}} = 80^\circ\text{C}$ and the solid part $T_{\text{solid}} = 540^\circ\text{C}$. We use the same initial and boundary conditions as the previous test case, here again the solid part is comped of aluminum alloy. As the previous test case, heat and thermal conductivity properties in the solid region, as a function of the metal temperature, are used to treat the variation in heat dissipation characteristic. A

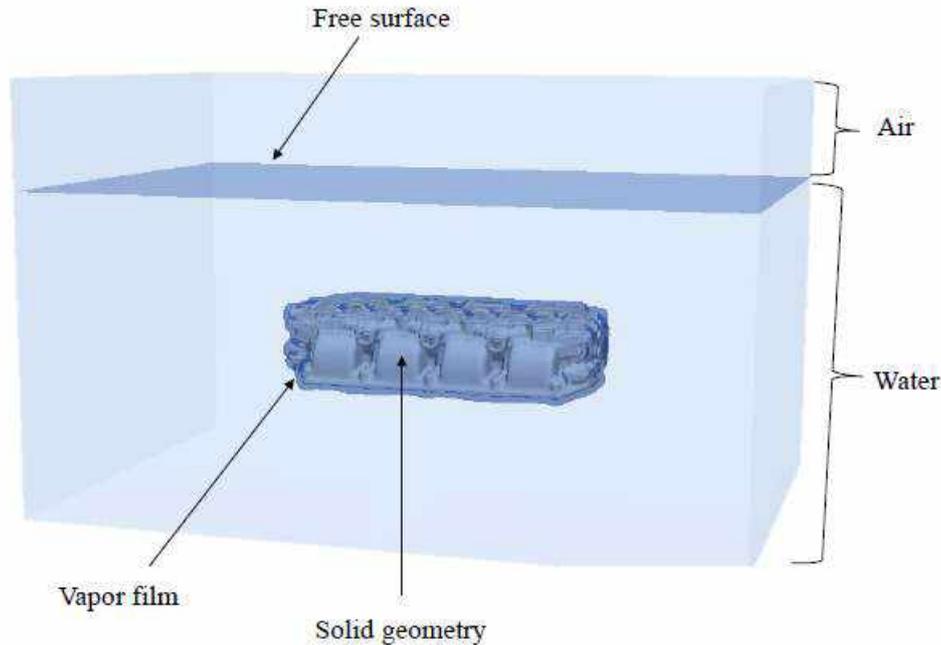


Figure 5.15: Set up for the simulation of the quenching of a cylinder head geometry

total number of 6 million elements was used for this simulation. The time step is set to $\Delta t = 10^{-4}$ s. The computation was launched on 112 processors on our Cluster.

5.3.2.2 Results and discussions

The anisotropic mesh adaptation algorithm responds very well to the thermal variations and the induced flow motion by strongly refining the regions with sharp gradients and highly stretching the elements in the rest of the domain thus optimizing to the best the use of the prescribed nodes. In addition, a precise capture of the interfaces is maintained all over the simulation resulting in an accurate heat transfer between both sides of the different interfaces. We can clearly identify the directional feature of the mesh which reflects very well the characteristics of the problem. As seen in Figure 5.16, the elements are well refined along the gradients of the adaptation fields enabling a very good capture of the growing bubbles. Figure 5.17 shows the evolution of the vapor film around the cylinder head. At the first time steps, the vapor film starts to detach from the geometry, it goes to the top of the geometry. Then bubbles of different sizes and shapes start to grow from the surface. We can see bubbles coming out of the different holes of the cylinder head. All these bubbles are well captured by the dynamic mesh adaptation technique.

In Figure 5.18, we present the temperature profiles all along the computation, a slice along the x direction allows us to see the evolution of the temperature inside the metal part, but also the vapor field inside and around the cylinder head geometry. We can see a considerable vapor motion within the complex geometry since the part is composed of holes and passages. Figure 5.19 shows the evolution of the temperature fields on the surface of the geometry. Even though, we are dealing with a complex geometry in terms of shape and curvature, we can clearly see how the temperature's profiles do not suffer from numerical instabilities or spurious oscillations that appear in the presence of sharp gradients near the interface, and this with a very challenging geometry. This validates the adopted stabilized finite element approaches and the appropriate smoothing of the material properties. No experimental data was provided for this geometry, since a quenching

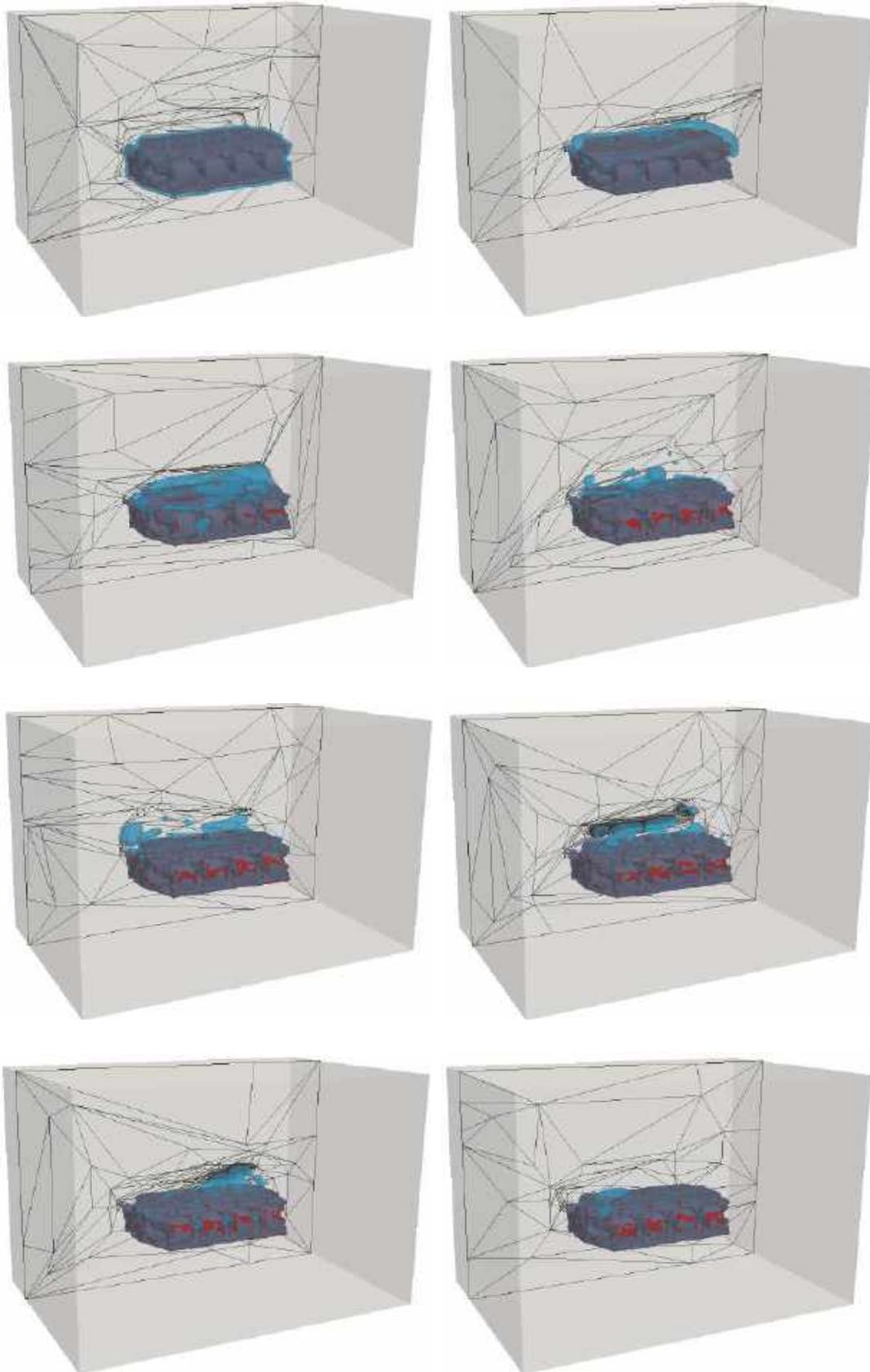


Figure 5.16: Simulation results. Evolution of meshes during the boiling at times $t = 0, 0.005, 0.15, 0.3, 0.5, 2$ s.

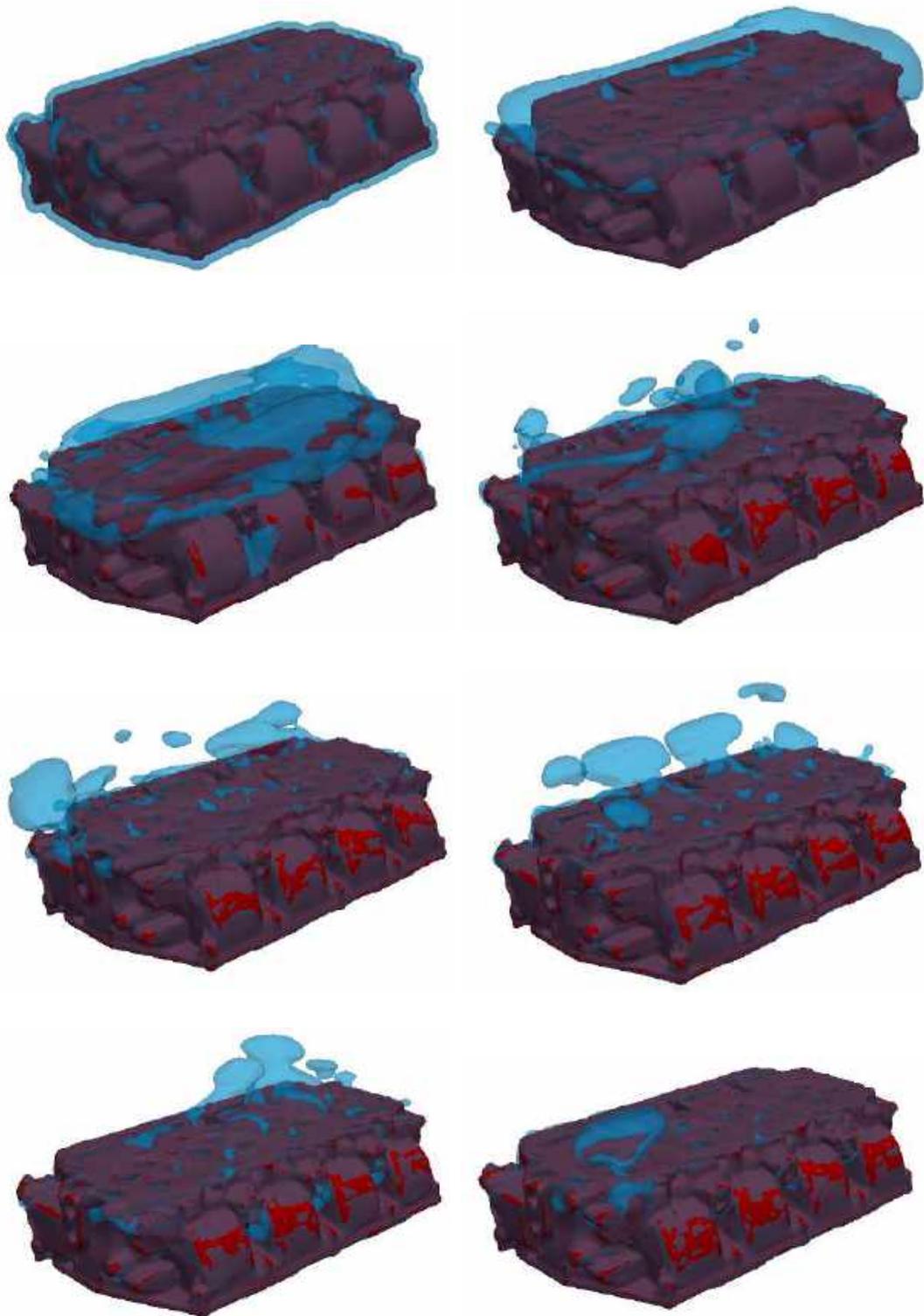


Figure 5.17: Simulation results. Evolution of the liquid/vapor interface during the boiling at times $t = 0, 0.005, 0.15, 0.3, 0.5, 2$ s .

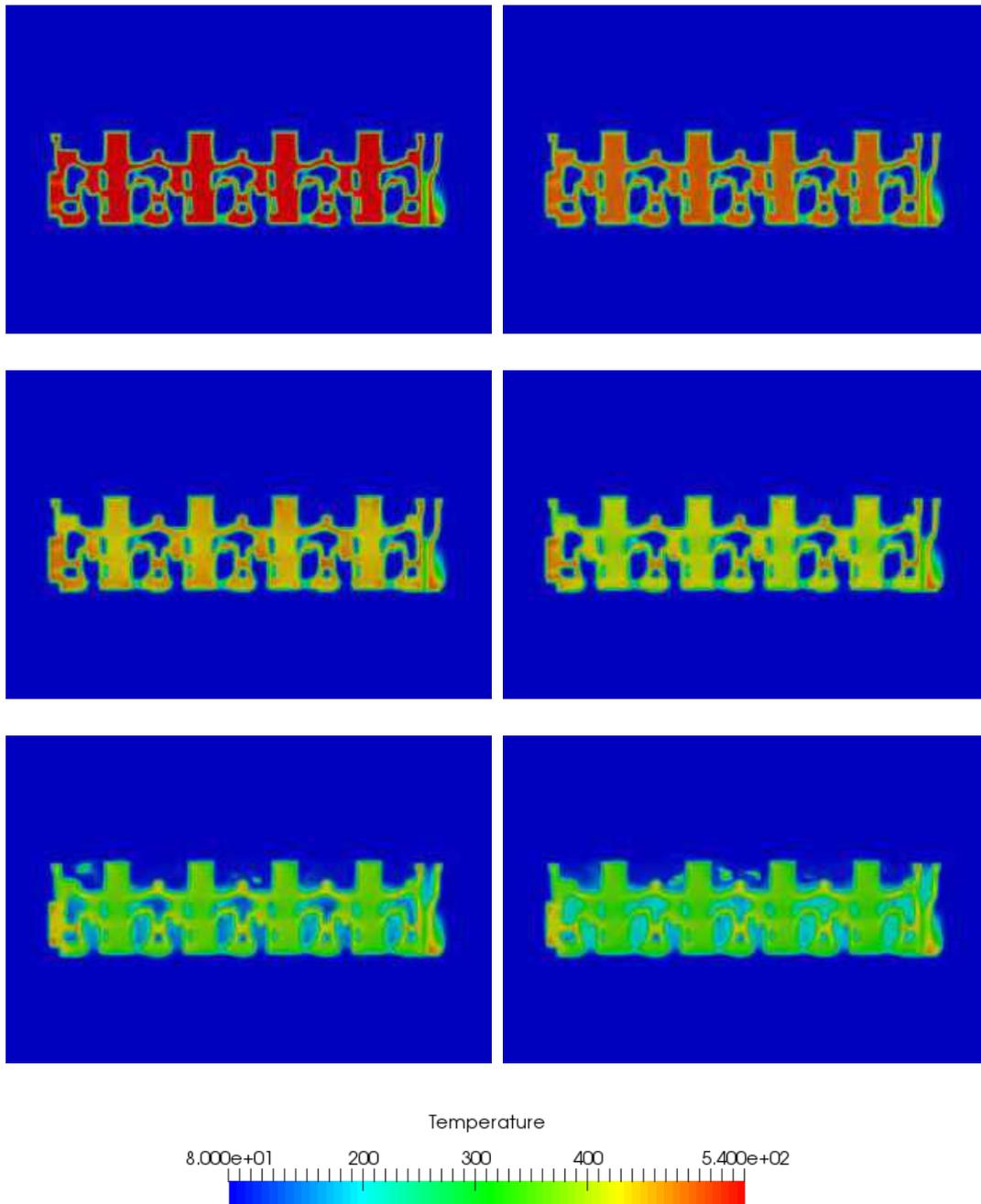


Figure 5.18: Simulation results. Temperature distribution along the sectional plane x at different times $t = 0, 0.2, 0.5, 0.7, 1, 1.2$ s.

with an air medium was preferred at the end by our industrial partner.

5.3.3 Challenging geometry : crossmember of a car

In the brief literature review presented in Chapter 1, we highlighted the fact that the heat transfer at the surface part is very dependent on the orientation and the position of the solid part in the quenching tank. Thus, for industrial purpose, based on the same engine geometry, two different quenching configurations are investigated in this section. The geometry is a crossmember of a car, presented in Figure 5.20. It is a very complex part

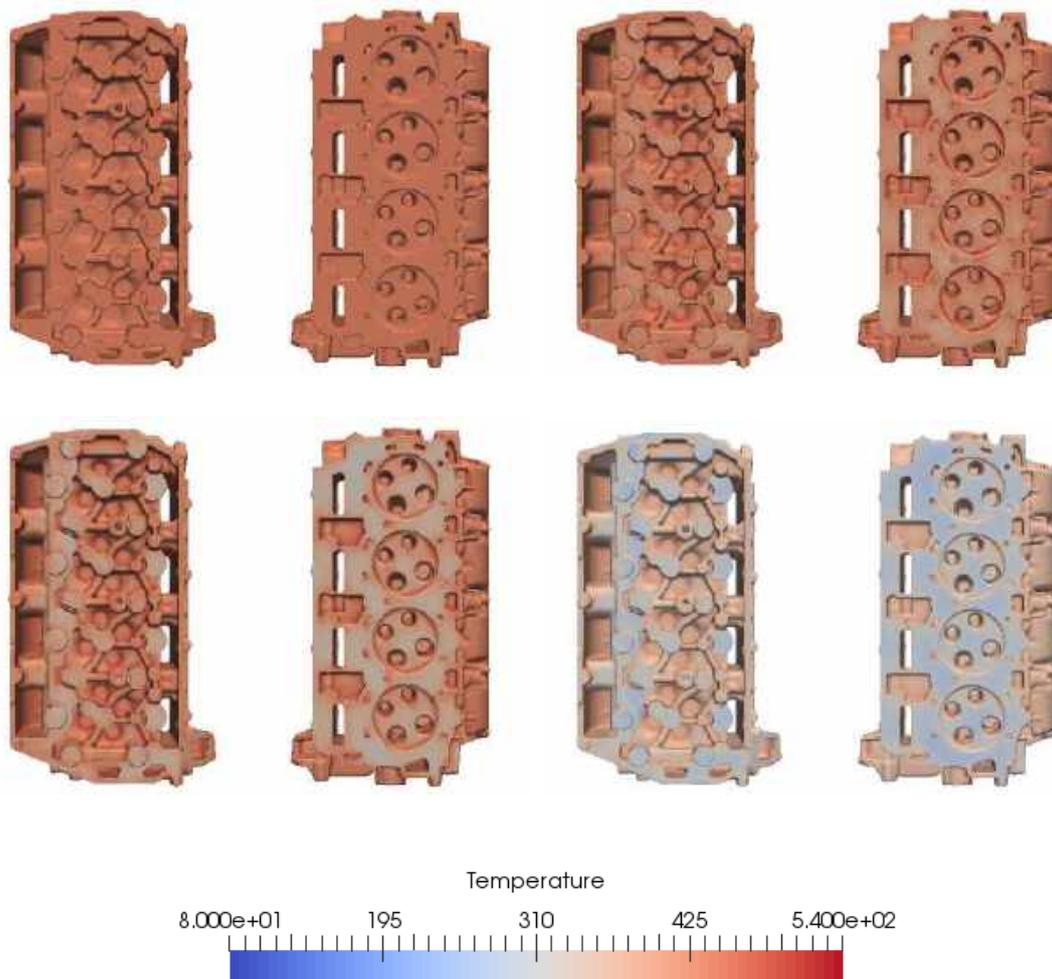


Figure 5.19: Simulation results. Temperature distribution on the surface of the geometry $t = 0, 0.2, 0.5, 0.7, 1, 1.2$ s .

with a very small thickness, moreover, the geometry is hollow inside.

5.3.3.1 Set up

This test case was proposed to us by our industrial partner in order to size the heat treatment tank, and especially to decide on the position of the parts in the baskets. Since, numerical simulation provides some insight of the effects of these parameters on the performance of the quenching, it will help us understand better the quenching process and to locate sensitive areas in the parts. Thus, two different configurations are considered (see Figure 5.21). The geometry is immersed horizontally at mid-height, and the vertically in the quenching tank. Pressed with time, we started by performing two simulations using a fixed mesh (with 4 million elements) in order to provide fast results to the industrial. To do so, we immersed the geometry, adapted the mesh all along the interface of the geometry. Then, defined a boundary mesh around the geometry with a small mesh size, and we coarsened the mesh close to the domain boundary. Which allowed us to provide fast results at a low computational time. Following the non-disclosure agreement with the company, the physical parameters of this test case will not be provided in this manuscript.

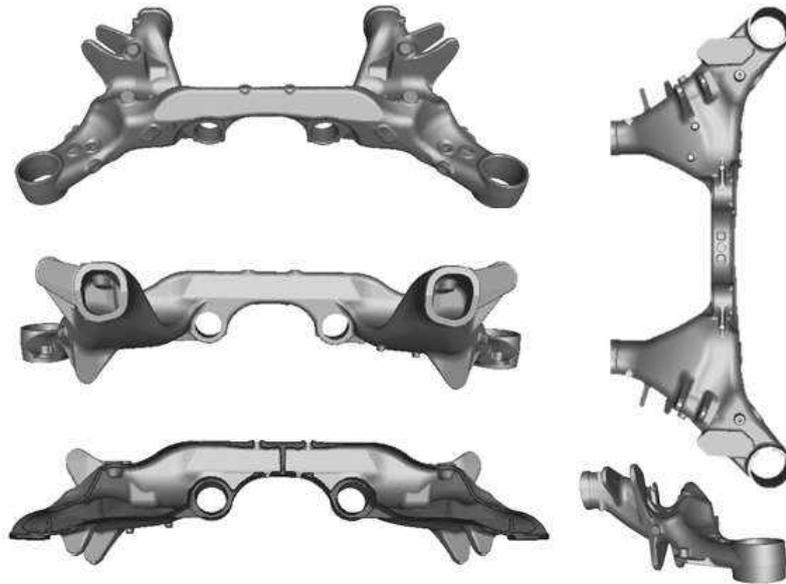
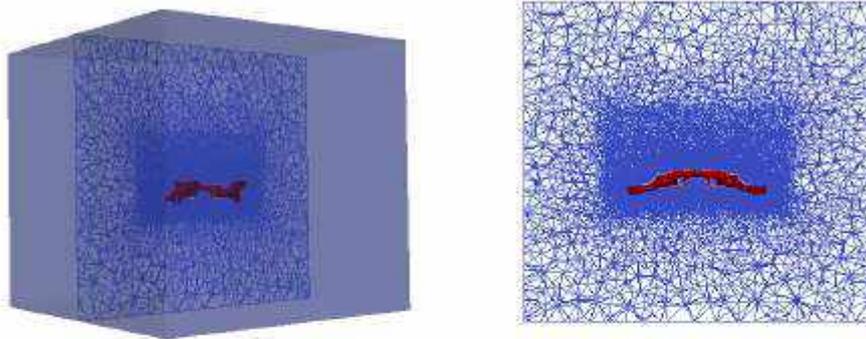


Figure 5.20: Crossmember geometry

Horizontal Quenching



Vertical Quenching

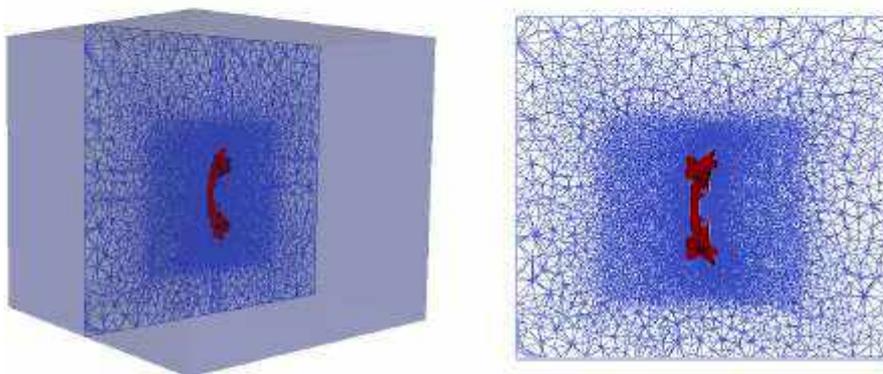


Figure 5.21: Set up for the simulation of the quenching of the crossmember geometry. Horizontal and vertical quenching

5.3.3.2 Results and discussions

In view of the simulations with a fixed mesh, many conclusions were drawn. The primary results for both simulations are show in Figures (5.22, 5.23). We can clearly see the de-

tachment of the vapor film and the formation of bubble, at the surface of the geometry. As expected, the position of the geometry has an important effect on the film evolution and thus on the cooling velocity of the part. The bubbles have different shapes, and this clearly shows the importance of the dynamic mesh adaptation technique that allows a very accurate capture of the different interfaces. We can clearly see that the vapor is stuck inside the solid geometry. And manages to go out from the different holes of the metal part. We depicted in Figure 5.24 the vapor field for the two different quenching posi-

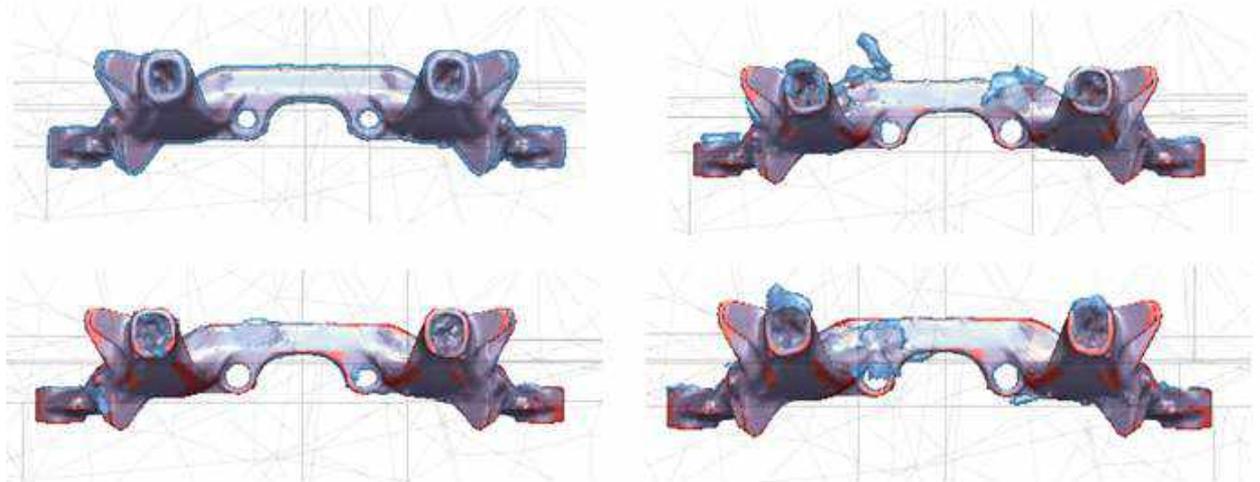


Figure 5.22: Evolution of the liquid/vapor interface for the horizontal quenching

tions. Since most of quenching tanks do not enable to get an inside view of the process, except from the free surface The numerical simulation allowed us to draw some fast conclusions on the optimal quenching position. Indeed, as depicted in Figure 5.24, we can see the vapor field in blue inside the geometry for both configurations. If we compare both quenching configurations, we can see that that during the horizontal quenching, we can note a symmetric propagation of the vapor field. Whereas, during the vertical quenching, the distribution of the flow is more concentrated in the upper part of the geometry. This is due to the vaporization of water nearby but also due to the rise of vapor from the lower part of the solid. Thus, during the vertical quenching more vapor stay stuck inside the cavity of the solid, and this might lead to some distortion of the metal part. Thus the horizontal position was preferred. And therefore, we decided to launch a new simulation of the horizontal quenching and this time using the dynamic mesh adaptation technique. The set up for the new simulation is shown in Figure 5.25. The mesh was adapted on the filtered level set function describing the geometry. The snapshot in Figure 5.25 depict the layers of elements near the object boundaries, even though it is a complex geometry with sharp angles, the mesh adaptation procedure procedure allows a good and accurate capture of the interface of the solid.

Recall that for confidential reasons, no explicit data will be provided for this test case. Figure 5.27 presents the variance in the temperature distribution along the section x , at different simulation instants. We can see the transient motion of the vapor surrounding the geometry. With progress in time, the vapor volume inside and outside the quenched geometry increases dramatically. This effect is attributed to the fact that the mass transfer effects are enhanced ensuring the transition from film to nucleate boiling heat transfer mode. To illustrate the transient nature of the quenching process and the resulting discerned non-uniform temperature distribution within the solid domain, we depicted

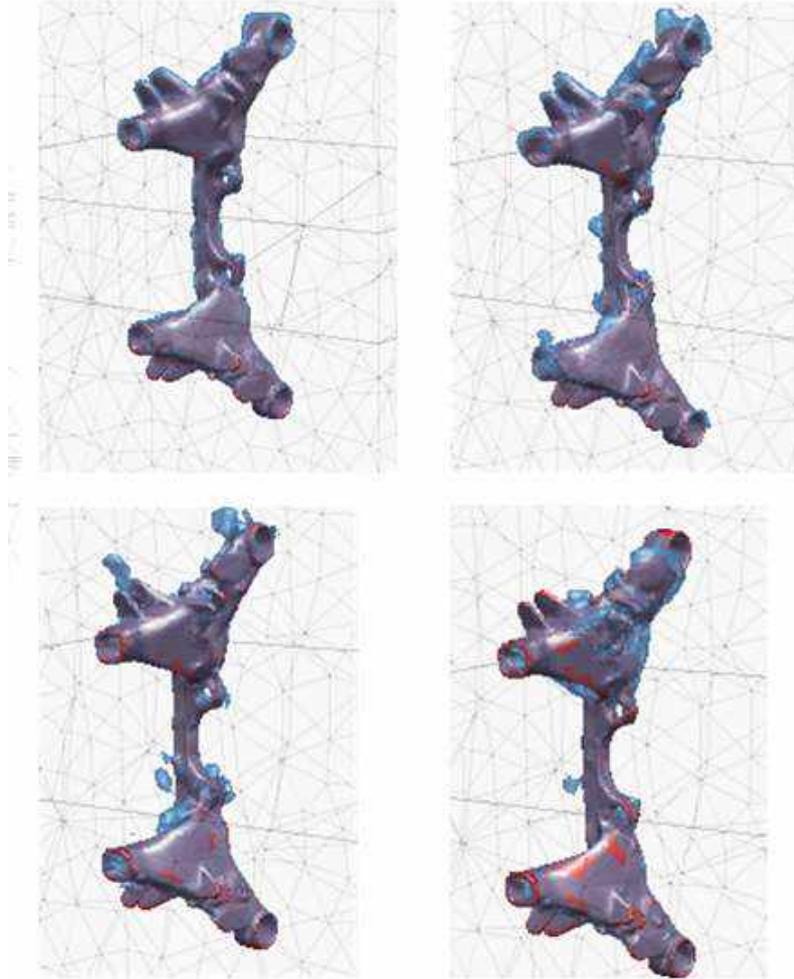


Figure 5.23: Evolution of the liquid/vapor interface for the vertical quenching

in Figure 5.27, the temperature fields at the surface of the solid part. We can see that while some sections of the part have cooled close to the minimum temperature, the mid-sections retain large portions of the applied heat, resulting in higher temperature values. Particularly, interior regions of the quenched part consisting of holes, passages, and this is explained by the considerable vapor motion that is persistent within the complex engine geometry. We can clearly see that the spreading of the cooling dynamics is not uniform all over the surface of the solid. This information provided by this simulation is particularly very convenient to understand and analyze potential zones prone to higher residual stresses and thus minimize and/or eliminate resulting material dynamics such as cracks and fatigue. The results coming from this simulation were given to our industrial partner to perform a residual stress analysis on Z-set. For confidentiality reasons, no data will be presented in this manuscript.

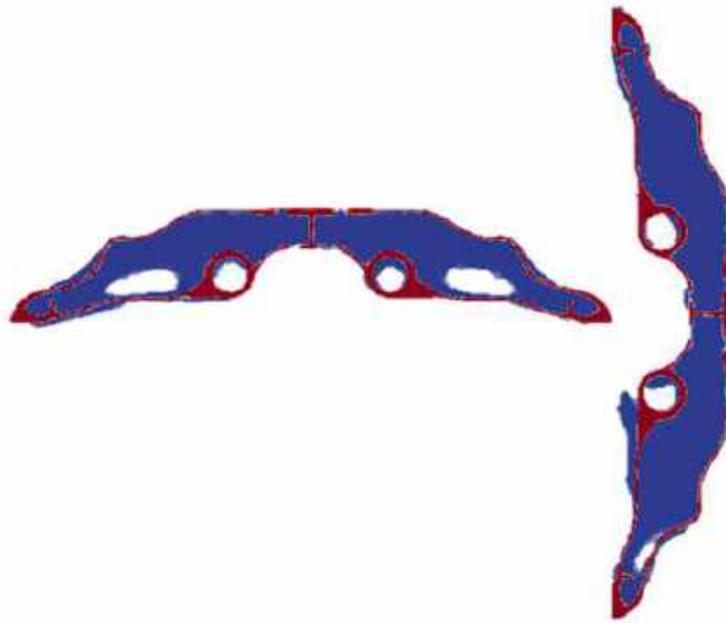


Figure 5.24: Snapshot of the vapor field inside the solid geometry for the horizontal (left) and vertical (right) quenching taken at the same time

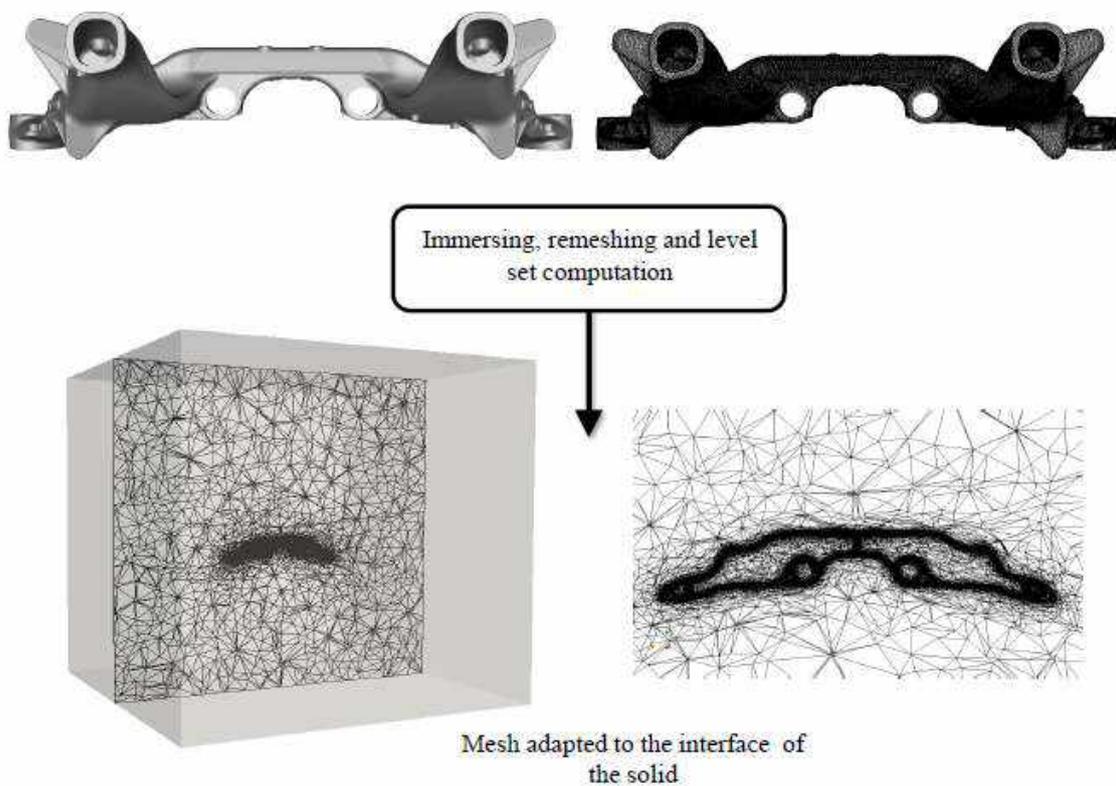


Figure 5.25: Set up for the simulation of the quenching of the crossmember geometry. Horizontal quenching

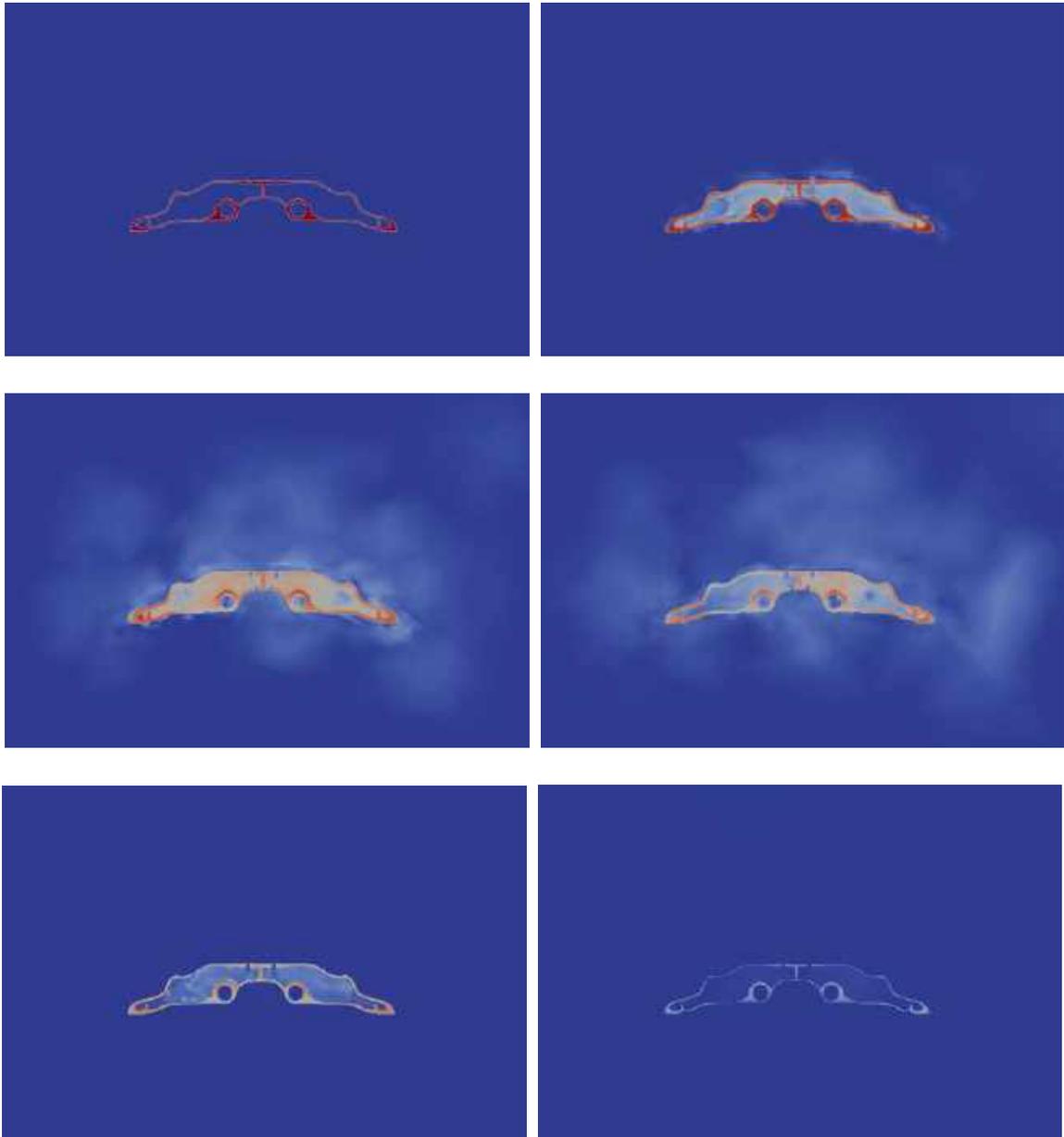


Figure 5.26: Simulation results. Temperature distribution along the sectional plane x at different times.



Figure 5.27: Simulation results. Temperature distribution along the sectional plane x at different times.

5.4 Conclusion

In this last chapter, we have first presented the coupling scheme between our library Cimlib-CFD and the software Z-set. Then, three simulations of industrial applications of the quenching process using the numerical framework developed in this thesis are presented. The cases simulated here represent very challenging cases due to the complexity of the geometries and the coupled physical phenomena involved such as high thermal gradients, high ratio of the physical properties or high convection. The first test case that we studied is a Ring geometry. We performed two simulations, with and without phase transformation inside the solid geometry, and many conclusions were drawn. In fact, the industrial partners provided us with experimental data, thus we compared the results coming from both simulations to the experimental data. When considering the phase transformation inside the solid, the results were more accurate, and this shows the importance of such coupling. Then, a numerical simulation of an automotive engine cylinder head quenching process was presented. Our numerical framework has been successfully employed to calculate the metal temperature histories in the solid domain. Considering the complexity of the boiling phenomena and its effect on the temperature evolution on such complex geometry such as the one tested in this study, we conclude that our results are satisfactory. And finally, the last studied geometry was a crossmember engine geometry. This test case was set to be a predictive tool for our industrial partners. Two different quenching configurations, differentiated by orientation of the immersion process, have been simulated in order to find the optimal quenching position. The results are interesting both from the numerical and industrial point of view. From the numerical point of view, modeling such complex phenomena is still an open topic, and with these complex simulations, we can say that this is a breakthrough. And from an industrial point of view, we provided a predictive tool for manufacturers to investigate and improve their quenching process.

5.5 Résumé du chapitre en français

Dans ce dernier chapitre, nous avons d'abord présenté le schéma de couplage entre notre bibliothèque Cimlib-CFD et le logiciel Z-set. Trois simulations d'applications industrielles du procédé de trempe utilisant le cadre Eulérien multiphasique développé dans cette thèse sont présentées. Les cas présentés dans ce chapitre représentent un défi de taille. En effet, les géométries sont très complexes, les phénomènes physiques tels que des gradients thermiques élevés, des rapports élevés des propriétés physiques ou des phénomènes de haute convection sont complexes et difficiles à modéliser. Le premier cas de test que nous avons étudié est une géométrie en anneau. Nous avons effectué deux simulations, avec et sans transformation de phase au sein du solide, et de nombreuses conclusions ont été tirées. Pour ce cas, les partenaires industriels nous ont fourni des données expérimentales. Nous avons donc comparé les résultats issus des deux simulations aux données expérimentales. En considérant la transformation de phase à l'intérieur du solide, les résultats étaient plus précis, ce qui montre l'importance d'un tel couplage. Ensuite, une simulation numérique d'un procédé de trempe de la culasse d'un moteur automobile a été présentée. Notre cadre numérique a été utilisé avec succès pour calculer l'historique de température d'une telle géométrie pendant le procédé de trempe. Considérant la complexité des phénomènes d'ébullition et leurs effets sur l'évolution de la température sur une géométrie aussi complexe que celle testée dans cette étude, nous concluons que nos résultats sont satisfaisants. Et enfin, la dernière géométrie étudiée représente le train ar-

rière d'une voiture. Ce test était destiné à être un outil prédictif pour nos partenaires industriels. En effet, deux configurations de trempe, différenciées par l'orientation du processus d'immersion, ont été simulées afin de trouver la position de trempe optimale. Les résultats sont intéressants du point de vue numérique ainsi qu'industriel. Du point de vue numérique, la modélisation de tels phénomènes complexes reste à ce jour un sujet très complexe, et proposer un cadre eulérien multiphasique permettant de simuler tel procédé est une avancée. Et du point de vue industriel, nous avons fourni aux fabricants un outil prédictif leur permettant d'étudier et d'améliorer leur procédé de trempe.

5.6 Bibliography

- [1] M. Khalloufi, Ecoulements multiphasiques avec changement de phase et ébullition dans les procédés de trempe, Ph.D. thesis, Paris Sciences et Lettres (2017). [116](#)

Chapter 6

Conclusion and perspectives

Contents

6.1 Conclusion	144
6.2 Perspectives	145
6.3 Résumé du chapitre en français	146

6.1 Conclusion

This PhD thesis was motivated by the need to perform simulations of industrial quenching processes. Such simulations involve turbulent flows, conjugate heat transfer, phase change and turbulent boiling. The tools used in this thesis are the Finite Element Method (FEM) and Computational Fluid Dynamics (CFD). This work was done in the laboratory CEMEF, a research center of MINES ParisTech-PSL Research University and in collaboration with our industrial partner Linamar Montupet. Using the previous numerical methods developed by the CFL team in the library Cimlib-CFD and adding more features we were able to propose an Eulerian multiphase framework able to precisely simulate the industrial quenching process. This manuscript was divided in five chapters. Let us recall in this conclusion the different points developed in each.

We started in Chapter 1 by giving a general introduction on the industrial quenching process. We emphasized the need for simulations to understand and reach a better control for such process. In fact, as explained, quenching processes of metals are widely adopted procedures in the industry, in particular automotive, nuclear and aerospace industries since they have direct impacts on changing mechanical properties, controlling microstructure and releasing residual stresses. A lack of control of such operation might lead to cracks and distortions. Today there is a strong demand from many industrial companies to control this cooling process. This chapter made it possible for us to better understand the complexity of the physical phenomenon occurring during the cooling such as phase change and boiling. We presented a brief literature review to give an idea of the evolution of the different methods used to model the quenching process. Conclusions were drawn, for instance, the major factors that should be taken into account to precisely model this process are the design of the quenching tank, the complexity of the solid geometry, the position and orientation of the solid part, and also the medium state (agitated or not) but also the thermal characteristics of the liquid are very important. A precise numerical model that offers understanding of the complex behavior of fluid flow and its impact on part cooling is still an open topic.

We have presented in Chapter 2 the eulerian conservative and adaptive framework. An anisotropic mesh adaptation technique, working under the constraint of a prescribed fixed number of nodes in the mesh has been detailed in this Chapter. The mesh generation process has been briefly described pointing out its capability to receive metric tensors and generate the corresponding adapted mesh. This mesh generation algorithm allows having meshes with extremely stretched elements at the interface. A detailed description of the adaptation framework has been provided. After explaining the steps behind the mesh adaptation procedure, we have presented a new conservative interpolation method based on the conservation of global physical quantities. We apply a correction to the interpolated solution by solving an optimization problem with Lagrange multipliers. We implemented this method in our library Cimlib-CFD with the help of the external library PETSc. Since we are dealing with heavy complex industrial problems, all computations must be launched on several processors. Thus, the implementation of the method in parallel was detailed. This framework was tested and validated on several benchmarks taken from the literature.

In this thesis, an immersed volume method (IVM) is used, it allows taking into account all the features of the process in a unified way. A full description of the method's compo-

nents was briefly introduced in Chapter 3. The method retains the use of a monolithic formulation whereby the fluid and solid parts of the problem were considered as a single medium with variable material properties. A conservative level set function was used to delimit the interface between the subdomains and to distribute the material properties using mixing laws. Combined with dynamic anisotropic mesh adaptation technique, the IVM lead to accurate simulations of multiphase applications. This method allows a simple and accurate resolution, in particularly at the interface between the fluid and solid. We have described in this chapter a stabilized finite element method for the transient incompressible Navier-Stokes equations based on the variational multiscale (VMS) principle, e.g. the decomposition of the unknowns into large scale and fine scale. Full description of the proposed moving interface capturing method, details and examples about this method are discussed in this chapter. The results were in good agreement with experimental and numerical references.

The next Chapter was dedicated to the modeling of the solid-liquid-vapor interactions in multiphase applications. We have proposed a phase change model to simulate the heat and mass transfer occurring during boiling. We detailed the integration of the phase change terms and derived the mass transfer rate at the liquid/vapor interface. The dynamics of the vapor phase is given by the surface tension, implemented in a semi-implicit way to overcome the usual severe capillary time step restriction, the computational cost is therefore drastically reduced. Since, standard Galerkin formulations exhibit numerical oscillations in convection or diffusion dominated regimes especially in the vicinity of sharp solution gradient, the convection-diffusion-reaction were solved using stabilization techniques to deal with numerical instabilities. A detailed description of such stabilization method is given in Chapter 4. Several examples of a fully coupled system of fluid flows and heat transfer involving phase change at the interface were presented in this Chapter and allowed us to validate this framework for modeling the solid-liquid-vapor interactions in applications that involve phase change at the interface.

All the presented numerical framework has been used to perform simulations of industrial water quenching processes. Indeed, the application of the developed numerical tools on 3D complex quenching problems involving phase change and turbulent flows have been investigated in Chapter 5. We first presented the development of a generalist "gateway" between two codes capable of accurately transferring fields from a fluid-structure mesh to a structure mesh. This structure mesh was used in the software Z-set to perform residual stress analysis. Then different simulations of complex geometries of car engine were presented. We compared some results with experimental data provided by our industrial partner. The obtained results reflected an improvement in the numerical accuracy when considering the phase transformation inside the solid. Considering the complexity of the boiling phenomena and its effect on the temperature evolution in the complex geometries such as the one tested in this Chapter, we concluded that our results are satisfactory.

6.2 Perspectives

From a physical or numerical point of view, several future research prospects can be considered in this thesis, these perspectives are specific to each subject dealt within each chapter:

- Regarding the conservative interpolation method presented in Chapter 2, we can extend the method for nonlinear restrictions. For instance, to consider the conservation of kinetic energy when solving the incompressible Navier Stokes equations. In that case, a linearization of the system must be done.
- In this thesis, all the presented simulations were performed using the level set method presented in Chapter 3. An alternative description of the interfaces is using phase-field methods. These models are based on the assumption about abrupt but continuous variation of the liquid phase density across the interface with non-zero thickness. In this method, the transport equation governing the phase indicator is modified by incorporating physical effects that govern thin interfaces, and this is very interesting when dealing with multiphase applications. A PhD thesis by Ali-Malek BOUBAYA has already started on this topic in 2018.
- The development of physically based robust numerical models for simulating quenching environments requires to get a better understanding of the involved physical phenomena such as contact angle, wetting, nucleation and many others. We must understand the local physics to enrich the macro model of the quenching process. This matter is handled by Charles Brissot in his PhD thesis that started in 2018.
- In Chapter 5, we presented the weak coupling method used to consider the thermo-mechanical behavior of the solid. To obtain a perfect prediction of residual stresses and thermo-elastoplastic stresses developed during heat treatment of the metal part, we must consider the fluid-structure interaction in a strong manner. A weak coupling is interesting, however an exchanges of data between the different fluid and solid domains can be very heavy in terms of computation, such method needs to be optimized. A PhD thesis by Joe Khalil has already started on this topic in 2019.

Others future perspectives can be considered :

- For instance, the radiation effect on the quenching process must be analyzed. Indeed, during the quenching process, the surface temperature is so high that the quenchant evaporates and a vapor film, more or less stable, is formed around the part. At this stage, the heat transfer is quite low as it mainly occurs by radiation. A new PhD student Remi Gerard has started in 2018 to work on radiative heat transfer.
- To obtain realistic simulations, we must consider the heat transfer between the solid and its surrounding environment, from the furnace to the quenching tank. To do so, the use of moving meshes is preferred, to follow the whole thermal history of the solid. The simulation of physical phenomena involving moving bodies undergoing large displacements still represents a real challenge.

6.3 Résumé du chapitre en français

Cette thèse était motivée par la nécessité de simuler des procédés de trempe industriels. Ces simulations impliquent des écoulements turbulents, un transfert de chaleur conjugué, un changement de phase et une ébullition turbulente. Les outils utilisés dans cette thèse sont la méthode des éléments finis (FEM) et la dynamique des fluides numérique (CFD). Ces travaux ont été réalisés dans le laboratoire CEMEF, centre de recherche de

l'université de recherche MINES ParisTech-PSL, en collaboration avec notre partenaire industriel Linamar Montupet. En utilisant les méthodes numériques précédemment développées par l'équipe CFL dans la bibliothèque Cimlib-CFD et en ajoutant plus de fonctionnalités, nous avons pu proposer un cadre eulérien multiphasique capable de simuler avec précision le processus de trempe industrielle. Ce manuscrit a été divisé en cinq chapitres. Rappelons dans cette conclusion les différents points développés dans chacun.

Le premier Chapitre nous a permis d'introduire le procédé de trempe. Nous avons souligné l'importance de la simulation numérique pour comprendre et atteindre un meilleur contrôle d'un tel procédé. La trempe des métaux est un procédé largement adopté par l'industrie, en particulier les industries automobile, nucléaire et aérospatiale. La trempe a un impact direct sur le changement des propriétés mécaniques, le contrôle de la microstructure et la libération de contraintes résiduelles. Si le procédé est mal contrôlé/réalisé, ceci pourrait entraîner des fissures ainsi que des distorsions au sein de la pièce métallique. Aujourd'hui il y a une forte demande de nombreuses industrielles pour contrôler ce processus de refroidissement. Ce chapitre nous a permis de mieux comprendre la complexité du phénomène physique survenant au cours du refroidissement: le changement de phase et l'ébullition turbulente. Nous avons présenté une brève étude bibliographique qui permet d'avoir une idée de l'évolution des différentes méthodes utilisées pour modéliser le procédé de trempe. Des conclusions ont été tirées. Par exemple, les principaux facteurs à prendre en compte pour modéliser précisément ce procédé sont la conception de la cuve de trempe, la complexité de la géométrie du solide, la position et l'orientation de la pièce solide, ainsi que l'état du milieu (agité ou non) mais aussi les caractéristiques thermiques du liquide sont très importantes. Un modèle numérique précis permettant de comprendre le comportement complexe du fluide et son impact sur le refroidissement des pièces est toujours un sujet ouvert.

Nous avons présenté dans le Chapitre 2, le cadre Eulerien adaptatif multiphasique. Une technique d'adaptation de maillage anisotrope, utilisant un nombre fixe d'éléments a été détaillée. Le processus de génération de maillage a été brièvement décrit soulignant sa capacité à recevoir des tenseurs métriques et générer le maillage adapté correspondant. Cet algorithme de génération de maillage permet d'avoir des maillages avec des éléments extrêmement étirés à l'interface. Une description détaillée du cadre d'adaptation a été fourni. Après avoir expliqué les étapes de la procédure d'adaptation du maillage, nous avons présenté une nouvelle méthode d'interpolation conservative basée sur la conservation de quantités physiques globales. Nous appliquons une correction à la solution interpolée en résolvant un problème d'optimisation avec les multiplicateurs de Lagrange. Nous avons implémenté cette méthode dans notre bibliothèque Cimlib-CFD à l'aide de la bibliothèque externe PETSc. Comme nous traitons de gros problèmes industriels complexes, tous les calculs doivent être lancés sur plusieurs processeurs. Ainsi, nous avons détaillé l'implémentation parallèle de la méthode. Ce cadre a été testé et validé sur plusieurs benchmarks tirés de la littérature.

Dans cette thèse, une méthode d'immersion de volume (IVM) est utilisée, elle permet de prendre en compte toutes les différentes échelles du problème et ainsi traiter les différents sous domaines comme étant un seul et unique domaine. Une description complète des composants de la méthode a été brièvement introduite dans le Chapitre 3. Nous utilisons une approche monolithique. C'est-à-dire que l'on considère qu'un seul domaine de calcul avec des propriétés physiques variées pour les différents sous-domaines:

fluide et solide. Un seul maillage est alors utilisé et ne nécessite pas de coïncidence avec la frontière fluide-solide. Dans cette thèse nous avons adopté la méthode d'immersion de volume, par laquelle les objets solides sont immergés et localisés dans la partie fluide en utilisant la fonction level set. Associé à la technique d'adaptation de maillage anisotrope dynamique, l'IVM permet des simulations précises d'applications multiphasiques. Cette méthode permet une résolution simple et précise, en particulier à l'interface entre le fluide et le solide. Nous avons aussi décrit dans ce chapitre une formulation éléments fini stabilisée pour résoudre les équations de Navier-Stokes incompressibles. Cette méthode est basée sur la décomposition des inconnues en grande et fine échelle. Une description complète de la méthode de capture d'interface mobile a été proposée. La méthode a été validée sur plusieurs cas tests, les résultats sont en bon accord avec les résultats expérimentaux et numériques trouvés dans la littérature.

Le chapitre suivant était consacré à la modélisation des interactions solide-liquide-vapeur dans les applications multiphasiques. Nous avons proposé un modèle de changement de phase pour simuler le transfert de chaleur et de masse se produisant pendant l'ébullition. Nous avons détaillé l'intégration des termes de changement de phase et déduit le taux de transfert de masse à l'interface liquide / vapeur. La dynamique de la phase vapeur est donnée par la tension superficielle, implémentée de manière semi-implicite afin de contourner la restriction sur le pas de temps, le coût de calcul est donc considérablement réduit. Étant donné que les formulations Galerkin standards présentent des oscillations numériques dans les régimes à convection ou à diffusion dominante, l'équation de convection-diffusion-réaction a été résolue à l'aide de techniques de stabilisation permettant de traiter les instabilités numériques. Une description détaillée de cette méthode de stabilisation est donnée dans le Chapitre 4. Plusieurs cas tests d'écoulements de fluide et de transfert de chaleur impliquant un changement de phase à l'interface ont été présentés et validés avec des comparaisons tirées de la littérature.

Le cadre eulérien multiphasique présenté dans les différents chapitres a été utilisé pour réaliser des simulations de procédés de trempe industriels à l'eau. En effet, les outils numériques développés ont été validés sur des problèmes complexes 3D impliquant le changement de phase et les écoulements turbulents. Nous avons tout d'abord présenté le développement d'une «passerelle» généraliste entre deux codes capables de transférer avec précision des champs d'un maillage fluide-structure à un maillage structuré. Ce maillage de structure a été utilisé dans le logiciel Z-set pour effectuer une analyse de contrainte résiduelle. Ensuite, différentes simulations de géométries complexes de pièces de voitures ont été présentées. Nous avons comparé certains résultats avec des données expérimentales fournies par notre partenaire industriel. Les résultats obtenus reflètent une amélioration de la de la précision des résultats Lorsque la transformation de phase à l'intérieur du solide est prise en compte. Considérant la complexité des phénomènes d'ébullition et son effet sur l'évolution de la température dans les géométries complexes telles que celle testées dans ce chapitre, nous avons conclu que nos résultats sont satisfaisants.

D'un point de vue physique ou numérique, plusieurs perspectives de recherche futures peuvent être envisagées dans cette thèse, ces perspectives sont spécifiques à chaque sujet traité dans chacun des chapitres:

- En ce qui concerne la méthode d'interpolation conservative présentée au Chapitre 2, nous pouvons étendre la méthode aux restrictions non linéaires. Par exemple,

considérer la conservation de l'énergie cinétique lors de la résolution des équations de Navier Stokes incompressibles. Dans ce cas, une linéarisation du système doit être effectuée.

- Dans cette thèse, les simulations présentées ont été réalisées à l'aide de la méthode level set présentée dans le Chapitre 3. Une autre description des interfaces consiste à utiliser des méthodes à champ de phase. Ces modèles sont basés sur l'hypothèse d'une variation abrupte mais continue de la densité de la phase liquide à l'interface avec une épaisseur non nulle. Dans cette méthode, l'équation de transport régissant la phase est modifiée en incorporant des effets physiques qui régissent les interfaces minces, ce qui est très intéressant lorsque qu'on traite d'applications multiphasiques. La thèse de doctorat d'Ali-Malek BOUBAYA a déjà commencé sur ce sujet en 2018.
- Le développement de modèles numériques robustes pour la simulation des procédés de trempe nécessite une meilleure compréhension des phénomènes physiques impliqués tels que l'angle de contact, le mouillage, la nucléation et bien d'autres. Nous devons comprendre la physique locale pour enrichir le macro-modèle mis en place pour modéliser le procédé de trempe industrielle. Cette question est traitée par Charles Brissot dans sa thèse de doctorat commencée en 2018.
- Au chapitre 5, nous avons présenté la méthode de couplage faible utilisée pour prendre en compte le comportement thermomécanique du solide. Pour obtenir une prédiction parfaite des contraintes résiduelles et thermo-élastoplastiques générées lors du traitement thermique de la pièce métallique, il faut considérer l'interaction fluide-structure de manière forte. Un couplage faible est intéressant, cependant un échange de données entre les différents domaines fluide et solide peut être très lourd en termes de calcul, une telle méthode doit être optimisée. Une thèse de doctorat de Joe Khalil a déjà commencé sur ce sujet en 2019.

D'autres perspectives de recherche peuvent être envisagées:

- Par exemple, l'effet du rayonnement au cours du procédé de trempe doit être analysé. En effet, lors de la trempe, la température de surface est si élevée que le liquide s'évapore et un film de vapeur, plus ou moins stable, se forme autour de la pièce. A ce stade, le transfert de chaleur est assez faible car il se produit principalement par rayonnement. Un nouveau doctorant, Remi Gerard, a commencé en 2018 à travailler sur le transfert de chaleur par rayonnement.
- Pour obtenir des simulations réalistes, nous devons considérer le transfert de chaleur entre le solide et son environnement, du four au réservoir de trempe. Pour ce faire, il est préférable d'utiliser des maillages mobiles afin de suivre toute l'histoire thermique du solide. La simulation de phénomènes physiques impliquant les corps en mouvement soumis à de grands déplacements représente un véritable défi.

RÉSUMÉ

La trempe est une méthode de traitement thermique où un métal chaud est refroidi rapidement à l'aide d'un médium. Le but est de donner au métal une certaine microstructure afin d'atteindre les performances mécaniques requises. Ce procédé a des impacts directs sur l'évolution propriétés mécaniques, contrôle de la microstructure et libération des contraintes résiduelles. Afin de réaliser un procédé optimal, il est essentiel de contrôler correctement les transformations de phase qui ont lieu dans l'alliage, et ainsi obtenir la microstructure présentant les propriétés thermomécaniques souhaitées.

Cette thèse est réalisée en collaboration avec la société Linamar Montupet spécialisée dans la fabrication de composants en aluminium pour l'industrie automobile. Ils s'intéressent à la trempe des pièces métalliques dans les liquides pouvant vaporiser. La vaporisation est le principal phénomène qui anime le système.

L'objectif de cette thèse est donc de définir un cadre numérique capable de simuler le procédé de trempe à l'échelle industrielle. Différents aspects seront étudiés: (i) analyser et simuler les interactions liquide-vapeur-solide avec changement de phase, (ii) simuler des interactions fluide-solide pour pouvoir prédire le comportement thermomécanique du solide. Les résultats des développements numériques seront validés par des confrontations avec les expériences proposées par le partenaire industriel.

MOTS CLÉS

Procédé de trempe, Adaptation de maillage anisotrope, Interpolation conservative, Mécanique des Fluides, Navier Stokes VMS, Level set, Ebullition turbulente, Changement de phase, Transformation de phase.

ABSTRACT

Quenching is a heat treatment method where a hot metal part is cooled down rapidly with the help of a quenchant. The purpose of such process is to give a certain microstructure to the metal in order to achieve the required mechanical performance. This process has direct impacts on changing mechanical properties, controlling microstructure and releasing residual stresses. Good control of quenching is essential for correctly controlling the phase changes that take place within the alloy, and obtain the microstructure exhibiting the desired thermomechanical properties. This Phd is done in collaboration with the company Linamar Montupet specialized in the manufacture of complex cast aluminium components for the automotive industry. They are interested in the quenching of metallic parts in liquid quenchants that can vaporize. The vaporization is generally the leading phenomenon that drives the system.

Thus, the objective of this thesis is to set a numerical framework able to simulate the quenching process at an industrial scale. In this thesis, different aspects will be studied: (i) analyze and simulate the liquid-vapor-solid interactions with phase change, (ii) simulate fluid-solid interactions to be able to predict the thermomechanical behavior of the solid. The results coming from these numerical development will be validated by confrontations with the experiments proposed in agreement with the industrial partner.

KEYWORDS

Quenching process, Anisotropic mesh adaptation, Conservative interpolation, Fluids mechanics, Navier Stokes VMS, Turbulent boiling, Phase change, Phase transformation.