Politecnico di Torino

ATMOSPHERIC RE-ENTRY OF END-OF-LIFE SATELLITES AND NUMERICAL MODELLING OF HEAT DIFFUSION WITHIN ANISOTROPIC MATERIALS

PIERLUIGI PERTOSO

ONERA

THE FRENCH AEROSPACE LAB

Master's degree in aerospace engineering

Supervisor
Prof. Domenic D'AMBROSIO - Politecnico di Torino

Tutors
Guillaume PUIGT - ONERA
Nicolas DELLINGER - ONERA
Marta DE LA LLAVE PLATA - ONERA

5 October 2022 – Torino

ABSTRACT

Today, the control of the end of life of a satellite is essential in order to limit the pollution of earth orbits on the one hand, and to reduce the risks of material and/or human damage linked to their uncontrolled re-entry into the atmosphere on the other hand. It is thus imperative to preserve some energy to deorbit the satellite at the end of its life and place it on an appropriate atmospheric re-entry trajectory.

During the re-entry, the flow and the wall of the object are in strong interaction: the parietal heating linked to the transformation of kinetic energy into heat leads to the degradation of the material by pyrolysis, oxidation, erosion and even sublimation. The surface of the satellite is then altered, which modifies its aerodynamic characteristics in return. The degradation must be sufficient on the trajectory to burn completely and thus destroy the satellite before it reaches the ground.

ONERA is developing numerical tools to simulate the interaction between aerothermodynamics (hypersonic flow with chemical kinetic effects) and the thermochemical response of the material. In particular, the *MoDeTheC* code allows to model the heat and mass transfers in a porous anisotropic composite material, as well as its volumetric or surface degradation under the effect of external aggressions. An important component of this code is the modeling of heat diffusion.

The objective of the work described in this thesis is to revisit the diffusive scheme already implemented in MoDeTheC, namely the scheme proposed by Leterrier in 2003, and replace it with a more advanced diffusive scheme proposed by Jacq in his 2015 thesis work. An in-depth study of the two schemes was conducted by implementing both of them on a code called DEEP-DIVE, which served as a testbed to analyze the numerical characteristics of the two schemes and understand their peculiarities. In a second step, using the MoDeTheC code, some simulations were carried out to study the atmospheric reentry of a sphero-conical body. The main focus of this second part of the work was to understand the influence of anisotropy on the physics of atmospheric reentry by comparing the results obtained in the isotropic case with those obtained by modifying the thermal conductivity matrix in such a way as to introduce anistropy into the material.

ACKNOWLEDGEMENTS

CONTENTS

LIST OF FIGURES

Part I

BIBLIOGRAPHIC STUDY

In this part, we will analyse the physical phenomena relating to the diffusion of heat within anisotropic materials during atmospheric re-entry phase. Furthermore, the numerical methods generally used to discretize and solve the differential equations describing the physics of the problem will be explained in detail. In particular, two new methods introduced by *Leterrier* and *Jacq* will be described, analysing their principles and studying their numerical properties.

1

INTRODUCTION

The purpose of this document is to present the results of the work carried out during the internship period at ONERA in Toulouse. This internship is the final step of a double degree programme started at the *Politecnico di Torino* and concluded at *Isae-SUPAERO* in Toulouse. ONERA is the French aerospace research laboratory whose main mission is to develop and guide research activities in the aerospace field.

In particular, the internship work was done at the DMPE (Multi-physics and energetics department) which carries out research in the field of multiphysics and energetics for a variety of applications, including experimental, theoretical and numerical activities. In particular, the DMPE works on improving and validating the physical models implemented in ONERA's CFD codes as well as on studying innovative concepts in test facilities under representative conditions.

The entire internship work was carried out under the supervision of DMPE researchers *Guillaume Puigt*, *Nicolas Dellinger* and *Marta De La Llave Plata* and of the professor at the Polytechnic of Turin, prof. *Domenic D'Ambrosio*.

1.1 CONTEXT

Nowadays, the atmospheric re-entry of inhabited and uninhabited vehicles is an increasingly occurring phase in various space missions. We can cite several situations in which an accurate knowledge of the physical phenomena involved during an atmospheric re-entry phase is of crucial importance for mission success. For example, control of a satellite's end-of-life is essential in order to limit the pollution of Earth orbits on the one hand, and to reduce the risks of material and/or human damage linked to their uncontrolled re-entry into the atmosphere on the other. We may also think about the importance of carefully designed thermal protection, consisting of layers of ablative material, for the re-entry of inhabited vehicles. In addition, adequately predicting the effects of a supersonic flow on a body is crucial to improve the technologies of reusable launchers, which must return to Earth with minimal damage to make reuse as cost-effective as possible, or of Mars probes, which must pass through Mars' atmosphere (albeit less dense than Earth's) in order to land on the ground undamaged.

At the DMPE, we are particularly interested in studying the end-of-life satellite trajectory in order to predict and avoid dangerous situations. In the case of satellite's re-entry, the material degradation should be sufficient on the trajectory to burn out completely and thus destroy the satellite before it reaches the ground. However, as this is not always possible,

some satellite fragments may fall to Earth causing risks to people and property. For this reason, it is essential to predict the satellite's trajectory as best as possible, considering all the phenomena to which it will be exposed during its atmospheric re-entry.
During this phase, the flow and the wall of the object are in strong interaction: the parietal heating due to the transformation of kinetic energy into heat leads to the degradation of the material by pyrolysis, oxidation, erosion and even sublimation. The surface of the satellite is then altered, which in turn changes its aerodynamic characteristics and affects its trajectory. Predicting the degree of degradation of space debris is therefore of great concern for space agencies such as ESA and NASA not only for safety reasons, but also for possible environmental risks [9]. Indeed, the thermal degradation of the surface of the debris' materials can generate chemical by-products that may potentially contribute to atmosphere pollution and trigger adverse chemical reactions, such as those affecting the ozone layer. [16].

In the following chapters, the physics of the problem will be explored. The phenomena associated with the hypersonic regime to which the body is subjected during atmospheric re-entry will be analysed. In addition, special emphasis will be placed on the phenomenon of ablation, which is strongly driven by the significant heat fluxes generated during this phase.

1.2 HYPERSONIC FLOW

In aerodynamics, by convention, a flow is defined as hypersonic when the fluid velocity exceeds 5 times the local speed of sound (i.e. the Mach number relative to the flow is 5 or greater). However, in contrast to the definition of the supersonic regime where the lower limit of $M = 1$ is an exact physical boundary, for the hypersonic regime this value is not so clear-cut.

Hypersonic flow is best defined as that regime in which certain phenomena gradually become more important as the Mach number increases. The fluid dynamics phenomena associated with this regime, such as shock waves and boundary layer, share some general characteristics with the supersonic regime, but present physical processes that can no longer be neglected and therefore change the physics of the problem [1].

1.2.1 *Shock Waves*

From the theory of oblique shock waves, it is well known that as the Mach number increases upstream, the shock wave approaches the wall. From a qualitative point of view, this is due to the fact that the stronger the shock, the greater the increase in density, and thus the possibility to handle the same flow rate in a smaller section.

In the hypersonic regime, this distance between the shock wave and the body becomes extremely small. If, in addition, the reacting chemical effects due to high temperatures are taken into account, the distance becomes even smaller.

For low-Reynolds numbers, having a shock layer that is too thin can cause complications from a physical point of view, as there is an interaction between the shock wave and the boundary layer. However, for high Reynolds numbers, where the shock layer is practically inviscid, the thinness of this zone is used to simplify the analytical treatment.

1.2.2 *Viscous Boundary Layer*

A high-speed hypersonic flow contains a large amount of kinetic energy. Through a process called *viscous dissipation* that occurs within the boundary layer, this kinetic energy is partly transformed into internal energy of the gas as it approaches the wall. As a result, there is a significant increase in temperature, which in turn affects the characteristics of the hypersonic boundary layer. For example, an increase in temperature causes an increase in fluid viscosity and thus a thickening of the boundary layer. This has also an effect on the density. Since the pressure must remain constant in the direction normal to the wall (Prandtl equation $\frac{\partial p}{\partial y} = 0$ where y is the perpendicular direction to the wall), if the temperature increases, the density must decrease due to the perfect gas law $\rho = \frac{p}{RT}$. This causes a further increase in the thickness of the boundary layer.

In a flat-plate compressible laminar boundary-layer, the thickness increases as

$$\delta \propto \frac{M_\infty^2}{\sqrt{Re_x}},$$

where M_∞ is the freestream Mach number and Re_x is the local Reynolds number, respectively defined as $M_\infty = \frac{u_\infty}{a_\infty}$ (where a_∞ is the speed of sound) and $Re_x = \frac{u_\infty \rho_\infty x}{\mu_\infty}$.

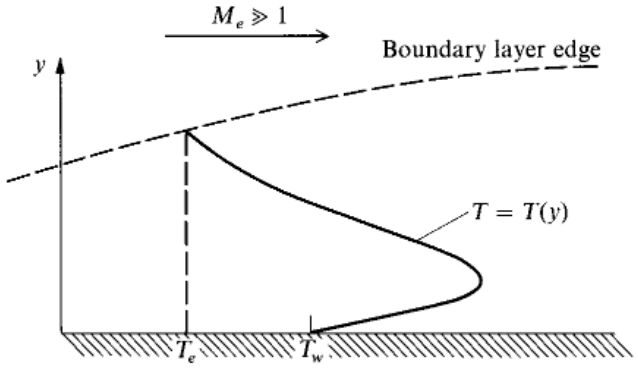


Figure 1: Temperature profile in a hypersonic boundary layer. Image taken from [1].

The thick boundary layer can have a significant displacement effect on the inviscid flow outside the boundary layer, giving the appearance that a given body shape is much larger

than it actually is. The outer inviscid flow is significantly altered due to the boundary layer flow's extreme thickness, which in turn has an impact on how the boundary layer grows.

The major interaction between the boundary layer and the outer inviscid flow is called *viscous interaction*. This phenomenon is extremely interesting as it directly influences the surface-pressure distribution and thus all the aerodynamic characteristics of the body upon atmospheric re-entry, i.e. lift, drag and stability. These modifications have a significant effect on the re-entry trajectory.

1.2.3 *High-Temperature Flow*

As mentioned above, the high heat flux that arises during the entry phase is the most interesting aspect of the internship work. The high viscous dissipation that develops in the hypersonic boundary layer creates very high temperatures, which can cause dissociation and even ionisation of the gas. Furthermore, if the body is protected by ablative material, the products of ablation are present in the boundary layer, giving rise to compressed chemical reactions. This is why we speak of a *chemically reacting boundary layer*.

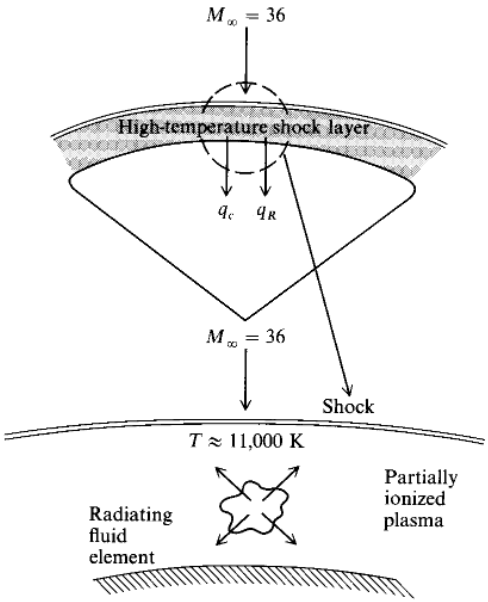


Figure 2: High-temperature shock layer. Image taken from [1].

When considering bodies in atmospheric re-entry, in addition to the high temperatures developed due to the phenomena described above, it is also necessary to take into account the significant temperature increase due to the shock wave generated at the front of the body.

If we consider a blunt body, we can refer to Figure 2 to understand the phenomena that develop in the area downstream of the bow shock. As is clear from the figure, for a flow characterised by a mach number of 36, the temperature can reach as high as 11000 K.

Furthermore, it is important to note that in this regime, the assumption of a calorically perfect gas is no longer valid and therefore all chemical reactions that develop must be taken into account. Thus, not only the boundary layer, but the entire shock layer is dominated by *chemically reacting flow*. In particular, the assumption that $\gamma = c_p/c_v$ is constant and equal to 1.4 is no longer valid, because the specific heats become functions of the temperature (for air, this effect is important above a temperature of 800 K).

When discussing hypersonic aerodynamics, in addition to convective heat transfer (called q_c in the figure) due to aerodynamic phenomena, another heat contribution must also be taken into account: radiative heating q_R. For very high temperatures, this contribution (which may be considered negligible in supersonic regimes) becomes increasingly important and in some cases even predominant. As can be seen in Figure 3, after a certain velocity,

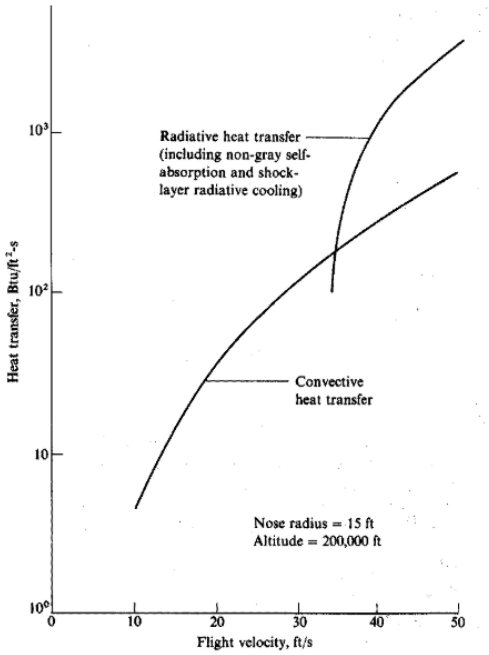


Figure 3: Nature of heating contribution as a function of the velocity. Image taken from [1].

the heat flux contribution due to radiation becomes overwhelmingly dominant compared to the convective contribution. It is therefore necessary to pay close attention to these thermal contributions when studying the thermodynamics of space re-entry.

1.3 HEAT DIFFUSION WITHIN ANISOTROPIC MATERIALS

The heat fluxes to which the body is subjected during atmospheric re-entry are very high and exert enormous stress on the structure and protective thermal layers. It is therefore of great interest to study the diffusion of heat within the satellite structure and protective layers. In this way, the most accurate possible prediction of ablation, degradation and break-up/fragmentation phenomena can be obtained.

To simulate these phenomena numerically, numerous mathematical models using partial differential equations have been developed, which can be simple or gradually more complex depending on the number and nature of phenomena modelled. However, while the problem of heat diffusion within isotropic and non-porous materials is a topic already abundantly covered in the literature, modelling and solving the heat diffusion within anisotropic and porous materials is still a subject of study.

The importance of accurately predicting heat diffusion within anisotropic materials is evident when dealing with satellites. The use of composite materials in the aerospace industry is increasingly common as these materials ensure both a decrease in the weight of the structure while maintaining high structural and thermal performance. In general, therefore, the materials that constitute the satellite's bearing structure will be composed of several layers, each consisting of a matrix and fibres of different materials. Each component reacts differently to thermal stresses, giving rise to the need to adequately distinguish each of them.

An important contribution to the modelling and study of this phenomenon was made by Valentin Biasi [4], who addresses thermal modelling within a composite material in his thesis work. This work eventually resulted in the MoDeTheC code, which was further developed at ONERA. ONERA is developing numerical tools to simulate the interaction between the aerothermo-dynamics (hypersonic flow with chemical kinetic effects) and the thermochemical response of the material. In particular, MoDeTheC allows for the modelling of heat and mass transfers in an anisotropic porous composite material, as well as its volumetric or surface degradation under the effect of external aggressions.

Finally, we can mention further codes that aim to evaluate phenomena related to heat diffusion in porous and anisotropic materials. These include the NASA open source code *PATO* [18] and the python library developed at the Von Karman Institute called *Mutation++* [8].

At this point, it is quite clear that the greatest difficulty will be related to the discretisation of the domain, which will have to take into account non-uniform geometry. The mesh and the numerical model will have to be capable of effectively taking into account the different layers and materials of which the body is composed in order to best simulate the heat flow through the outer shell.

This brings us back to our internship work, whose main objective is to implement and validate new numerical models for calculating diffusive flows that can best possible fit the problem and provide the most precise and accurate results.

The two models that will be studied are cell-centred finite volume schemes adapted to solve anisotropic diffusion on unstructured mesh (see Section 2.3 and Section 2.4). The first method developed by Nikos Leterrier [12], already implemented in MoDeTheC, will be implemented in another code called *DEEP-DIVE* in order to be studied and validated (see Section 3.4). Later on, the second method, developed by Pascal Jacq [11], will also be implemented in *DEEP-DIVE* and will be analysed to understand its advantages and to be later implemented in MoDeTheC (see Section 3.5).

2

NUMERICAL METHODS

2.1 HEAT DIFFUSION AND GOVERNING EQUATION

Heat exchange within a body is carried out in three different ways: conduction, convection and radiation. If one considers heat transport inside a non-porous solid body, it is possible to neglect the last two forms of heat transfer and only take conduction into account.
From a microscopic point of view, heat conduction, also called diffusion, consists of an exchange of kinetic energy by collisions between the particles of which the material is made. This exchange is driven by the temperature gradient that determines the direction of heat propagation: spontaneous heat transfer always occurs from a region with a higher temperature to a region with a lower temperature, as stated in the Second Law of thermodynamics. Finally, heat flow ceases when thermal equilibrium is reached.

The governing equation is obtained by carrying out an energy balance on the elementary volume V.

$$\frac{\partial}{\partial t}\int_V e\,dV = -\int_S \vec{\Phi}\cdot\vec{n}\,dS + \int_V \rho r\,dV \tag{1}$$

where

- e internal energy $[J/m^3]$
- $\vec{\Phi}$ heat flow in three dimensions $[J/m^2/s]$
- $\vec{n}$ normal vector to the surface
- S surface of the elementary volume $[m^2]$
- ρ density $[kg/m^3]$
- r heat source $[J/kg/s]$

Conductive heat flow is related to the temperature field through *Fourier's law*

$$\vec{\Phi} = -\mathbb{K}\vec{\nabla}T \tag{2}$$

where $\mathbb{K}$ is the thermal conductivity, which is a 3x3 tensor for anisotropic materials. The $(-)$ sign shows that the heat flow is defined as positive when the temperature gradient is negative, i.e. the direction of the heat flow is directed from warmer to cooler side. Indeed, considering K always positive definite, the model consistency with the Second Law of thermodynamics can be written as $\vec{\Phi}\cdot\vec{\nabla}T \leqslant 0$.
Applying the Gauss theorem at the conductive flow term of Equation 1

$$\int_S \vec{\Phi}\cdot\vec{n}\,dS = \int_V \vec{\nabla}\cdot\vec{\Phi}\,dV \tag{3}$$

it is possible to write Equation 1 in a local form. This leads to the classical heat equation formulation

$$\rho C_p \frac{\partial T}{\partial t} = \vec{\nabla} \cdot (\mathbb{K}\vec{\nabla}T) + \rho r \tag{4}$$

where ρ and C_p are assumed constant.
Furthermore, taking $\mathbb{K}$ as scalar (times identity matrix I) and constant and $\alpha = \frac{k}{\rho C_p}$ (thermal diffusivity), the previous equation can be rewritten as follows.

$$\frac{\partial T}{\partial t} = \alpha \Delta T + \frac{r}{C_p} \tag{5}$$

To close the problem and be able to solve this partial derivative equation with the temperature T as unknown, it is necessary to define an initial condition and some boundary conditions. Generally, an initial condition determines the solution for $t = 0$.

$$T(\vec{x}, 0) = T^0(\vec{x}) \qquad \vec{x} \in \mathcal{D}$$

To determine the field in the boundaries of the domain, one should impose other conditions, namely Dirichlet that is a condition on the exact value of the temperature at the boundary

$$T(\vec{x}, t) = T^*(\vec{x}, t) \qquad \vec{x} \in \partial\mathcal{D}_D$$

or the Neumann condition in order to impose the value of the heat flux at the boundary

$$\vec{q}(\vec{x}, t) \cdot \vec{n} = q_N^*(\vec{x}, t) \qquad \vec{x} \in \partial\mathcal{D}_N$$

or finally, the Robin condition that takes into account the coupling between convection and diffusion and can be seen as a weighted combination between Dirichlet and Neumann conditions.

$$\alpha T(\vec{x}, t) + \beta \vec{q}(\vec{x}, t) \cdot \vec{n} = q_R^*(\vec{x}, t) \qquad \vec{x} \in \partial\mathcal{D}_R$$

where α and β are two constants which determine the weight of each contribution. In our notation, $\partial\mathcal{D}_D$, $\partial\mathcal{D}_N$ and $\partial\mathcal{D}_R$ are the boundaries on which the Dirichlet, Neumann and Robin conditions are applied, respectively.
The cases in which this equation can be solved analytically are very rare and correspond to case studies rather than problems of practical interest. Therefore, it is necessary to develop methods to be able to solve the equation numerically.

In particular, the most widely used numerical scheme for solving the heat equation and in general transport equations (such as the *Navier-Stokes equations*) is the Finite Volume Method, which will be detailed below.

2.2 THE FINITE VOLUME METHOD

The Finite Volume Method[1] is a method for representing and evaluating partial differential equations in the form of algebraic equations. The finite volume method is widely used in the field of fluid dynamics as it allows transport (convection, conduction) and diffusion phenomena to be optimally model.

In this section, we will discuss in a general way the characteristics and requirements for a numerical scheme (Section 2.2.1), and then go into the specifics of the Finite Volume Method and how it can be applied to modelling physical phenomena described by partial differential equations.

2.2.1 *Stability, Consistency and Accuracy*

When using a numerical method it is fundamental to examine accurately the properties of the solutions obtained, or more specifically the discretization errors one can expect [7, 21]. When designing a numerical method two issues immediately come to mind:

1. under what conditions the results are reliable (i.e. stability and accuracy of the method);
2. to what level of accuracy the problem is solved.

In order to answer these questions, we can use the *Lax equivalence theorem* which states that *consistency and stability are necessary and sufficient conditions for convergence.* It is important to note that this theorem truly holds for linear schemes, while in general it is not always correct. In his paper [6], Despres emphasises an important property of the Finite Volume Method: it is not formally consistent, so Lax theorem does not apply. However, this work proves that cancellation in time of the error is a reason why finite volume methods converge. This result may be considered as a variation on the Lax theorem.

By consistency we mean that a method is a viable approximation to the original differential equation we are solving. Moreover, a method is stable if the solutions are bounded and any perturbation does not grow beyond these limits (all perturbations are damped).

On the other hand, the concept of accuracy involves the estimation of the quality of the approximation as a function of the mesh size h. The order of accuracy of the discretization method can be found either analytically, using the Taylor series expansion, or numerically, by varying the grid size, h, and computing the solution on each of the considered meshes.

Given the discrete formulation of the method and the Taylor series expansion of the solution at $\vec{x} + \Delta x$

$$u(\vec{x} + \Delta x) = u(\vec{x}) + \Delta x \frac{\partial u}{\partial x} + \frac{1}{2}(\Delta x)^2 \frac{\partial^2 u}{\partial x^2} + \frac{1}{6}(\Delta x)^3 \frac{\partial^3 u}{\partial x^3} + o((\Delta x)^4)$$

1 In this report, attention will be focused on the Finite Volume Method applied to the heat equation, hence to diffusion phenomena. The treatment of the convection equation, although similar to that described for diffusion, is neglected in this work. Interested reader can refer to [7, 21]

it is possible to study the accuracy of the method by substituting this expansion in the discrete equation for u_k (with $k = n+1, n, n-1, n-2, ...$) at the mesh points and expanding around a single point, $u(x)$. Furthermore, substituting the Taylor series expansion in the method formulation produces a working definition of *consistency : starting from the discrete equation, the key for consistency is the recovery of the original differential equation plus a sequence of ordered error terms.* And these error terms are exactly what provide us with the order of convergence of the analysed method.

An alternative way to determine the order of accuracy of the method is numerically, by implementing the method and analysing the error between the numerical solutions and the exact one. Once the method is implemented, a numerical analysis can be performed to estimate the order of convergence. For a spatial convergence study it is necessary to define a mesh resolution (or characteristic length) as

$$h = \left(\frac{|\mathcal{D}|}{\mathcal{C}_{\mathcal{D}}} \right)^{1/d} \tag{6}$$

where $|\mathcal{D}|$ is the volume (or the surface) of the computational domain, $\mathcal{C}_{\mathcal{D}}$ denotes the number of cells that paved the computational domain and d is the dimension of the space ($d = 2$ in 2D, $d = 3$ in 3D). By decreasing the value of h (i.e. by increasing the number of elements of the mesh), a convergence analysis can be carried out to establish the rate of convergence of the analysed method. If one has an analytical solution $\hat{u}(\vec{x})$, the numerical errors based in the discrete L_2 and L_∞ norms can be defined as follows:

$$E_h^2 = \sqrt{\sum_{c=1}^{\mathcal{C}_{\mathcal{D}}} \left(u_h^{(c)} - \hat{u}_h^{(c)} \right)^2 \parallel \mathcal{T}_c \parallel} \tag{7}$$

$$E_h^\infty = \max_{c=1 \ldots \mathcal{C}_{\mathcal{D}}} |u_h^{(c)} - \hat{u}_h^{(c)}| \tag{8}$$

where $\hat{u}_h$ is the discrete analytical solution evaluated in the mesh points, u_h is the numerical solution and $\parallel \mathcal{T}_c \parallel$ is the volume of a single cell c.

Assuming a general error distribution $\varepsilon = O(h^p)$ on a sequence of refined meshes keeping the same topology, p is the order of accuracy of the considered method. In order to compute this parameter, we can write

$$\log(\varepsilon) = p \cdot \log(h) + C$$

where C is a constant.

In the Part 2 of this report, this same analysis will be performed in order to find the order of accuracy of Leterrier's and Jacq's schemes.

To conclude, it is important to point out that the convergence study can be carried out not only in space but also in time. Indeed, it is possible to define an order of convergence

even for schemes used for time integration. In this case, the characteristic dimension will be a time step Δt, instead of a length. We will discuss temporal integration in more detail in Section 2.5.

2.2.2 *Discrete Variational Formulation*

In this section, we will briefly illustrate how a semi-discrete variational formulation can be obtained using the Finite Volume Method and how to calculate the individual terms of the resulting equations.

Starting from Equation 1, it is possible to obtain a discrete variational formulation by discretizing the space with a mesh generally composed of standard elements (in 2D: triangles, quadrilaterals; in 3D: tetrahedra, prisms, pyramids and hexahedra). For the sake of simplicity, it is assumed that the mesh elements do not overlap and that the mesh does not move in time.

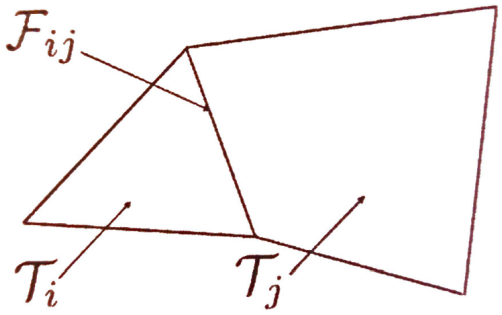


Figure 4: Example of a 2D mesh composed of two cells that share a common interface

The principle of the Finite Volume approximation is to compute unknowns in each mesh cell $\mathcal{T}_c$ by evaluating the fluxes contribution, here for the heat equation.

$$\rho C_p \frac{\partial}{\partial t} \int_{\mathcal{T}_c} T dV = - \sum_{f \in \mathcal{F}_c} \int_{S_f} \vec{\Phi} \cdot \vec{n} dS + \int_{\mathcal{T}_c} \rho r dV \tag{9}$$

where $\mathcal{T}_c$ is the way to indicate the considered cell in concordance with notation of Figure 4. The averaged quantity $\bar{u}$ over $\mathcal{T}_c$ is introduced

$$\int_{\mathcal{T}_c} T dV = \bar{T} \cdot \| \mathcal{T}_c \|$$

and the flux (the first right-hand side term of Equation 9) is expressed using the Fourier's law. Thus, we can write

$$\rho C_p \frac{\partial \bar{T}}{\partial t} \cdot \| \mathcal{T}_c \| = - \sum_{f \in \mathcal{F}_c} \int_{S_f} -\mathbb{K} \vec{\nabla} T \cdot \vec{n} dS + \int_{\mathcal{T}_c} \rho r dV \tag{10}$$

If we consider that the gradient $\vec{\nabla}u$ at the centre of the face is constant over the whole face $\mathcal{F}$ and the source r is a constant in the considered volume $\mathcal{T}$, we can write

$$\rho C_p \frac{\partial \bar{T}}{\partial t} \cdot \| \mathcal{T}_c \| = - \sum_{f \in \mathcal{F}_c} -\mathbb{K}(\vec{\nabla}T)_f \cdot \vec{n}_f \| \mathcal{S}_f \| + \rho r_c \| \mathcal{T}_c \| \tag{11}$$

where ρ and C_p are respectively the density and the specific heat coefficient of the considered gas (or solid, if we are dealing with heat diffusion within a body). Furthermore, $(\vec{\nabla}T)_f$ is the gradient in the center of the face f and $\vec{n}_f$ is the normal at the face f and $\| \mathcal{S}_f \|$ is the area of surface f.

As it can be noticed from the last equation, the Finite Volume Method is a conservative method as it is based on the concept of flows entering and leaving a cell: since the flux entering is equal in modulus to the flux leaving, we can speak of conservation of the quantity transported. This is an important property that must be verified when applying numerically the method.

Equation 11 is constituted by 3 different terms: the unsteady term that will be discretized using the numerical methods described in Section 2.5, the production term that is generally known *a priori*, and the diffusive term in which the gradient at the centre of the mesh cell faces is present and which constitutes the main unknown that needs to be derived.

There are several methods for determining the flux at the centre of the face. In cell-centred schemes, the simplest method is to average the gradients at the centre of the cells adjacent to the face. Using a general notation, we can write:

$$(\nabla u)_{f_{ij}} = \frac{(\nabla u)_i + (\nabla u)_j}{2}, \tag{12}$$

where the index refer to Figure 4.

The approximation of the gradient at the centre of the face using gradients at the centre of adjacent cells gives rise to a new need: the calculation of the cell-centred gradient.

2.2.2.1 *Evaluation of cell-centred gradients*

As shown above, a key ingredient in the calculation of fluxes at the interface between two cells is the gradient of the transported quantity. In particular, the gradient is calculated at the center of gravity of the cells using three classic methods found in the literature, such as the Green-Gauss method, the Quasi-Green method and the Least Square method.

- **Green-Gauss method**

  The starting point for this method is the application of Green's formula, which expresses the integral of the gradient ∇u in the cell $\mathcal{T}$ with an integral along the surface $\mathcal{S}$.

$$\int_{\mathcal{T}} \vec{\nabla} \cdot u dV = \int_{\mathcal{S}} u \cdot \vec{n} dS = \sum_{f \in \mathcal{F}_i} \left(\int_f u \cdot \vec{n} dS \right) \tag{13}$$

If we consider just one direction (for example $\vec{x}$ direction), the integral can be rewritten as

$$\sum_{f \in \mathcal{F}_i} \left(\int_f u \cdot \vec{n} dS \right) = \sum_{f \in \mathcal{F}_i} u_f n_x \|S_f\|$$

where $\|S_f\|$ is the area of the face f and n_x is the component of the normal vector $\vec{n}$ in the considered direction. u_f is the value of the unknown u in the face center which can be easily computed as $u_f = \frac{u_i + u_j}{2}$ (with u_i and u_j the values of u at the left and right cell centers).
The extremely basic approach of this method doesn't allow us to take into account irregularities of unstructured meshes, such as curvature or heterogeneity. Assuming the interface center in the middle of the segment whose vertices are the cell centers can lead to very inaccurate gradients in an unstructured mesh. This method is generally avoided, in favor of other more accurate methods.

- **Quasi-Green method**

In order to deal more adequately with irregular meshes, it is necessary to introduce a scheme that takes into account the characteristics of the cells and weights the contribution from each element.

$$\nabla u\big|_i = \sum_{j \in \mathcal{C}_\mathcal{D}} \alpha_{i,j} u_j$$

There are several solutions in the literature for choosing the most suitable coefficients $\alpha_{i,j}$. In their work [22], *Pont et al.* propose to choose a particular set of coefficients, writing the above equation under the form

$$\nabla u\big|_i = \sum_{F_{ik} \in \mathcal{F}_i} \left(\gamma_{i,k} u_k + (1 - \gamma_{i,k}) u_i \right) \vec{n} \|F_{ik}\| \tag{14}$$

which means that the stencil they consider is based on the current cell and on the ones that share a face with it.
Then, they define the γ coefficients by

$$\gamma_{i,k} = \frac{C_k C_f}{C_k C_f + C_i C_f}$$

where C_i and C_k are centers of cells i and k and C_f is the center of the face F_{ik} shared by cells i and k. $C_i C_f$ (or $C_k C_f$) represents the length of the segment that links the points C_i and C_f (or C_k and C_f).
This set of coefficients only respects the notion of 0-exactness. In order to account also for the notion of 1-exactness, a correction must be applied to Equation 14. The authors of the paper define as correction factor a matrix M_1 such that

$$M_1 = \sum_{f_{ik} \in \mathcal{F}_i} \gamma_{i,k} \begin{pmatrix} x_k - x_i \\ y_k - y_i \\ z_k - z_i \end{pmatrix} \otimes \vec{n} \|S_{f_{ik}}\|,$$

where x, y, z are the spatial coordinates of points C_i and C_k.

Finally, the formulation for the cell-centred gradient is the following one.

$$\nabla u\big|_i = (M_1)^{-1} \sum_{f_{ik} \in \mathcal{F}_i} \left(\gamma_{i,k} u_k + (1 - \gamma_{i,k}) u_i \right) \vec{n} \| S_{f_{ik}} \| \tag{15}$$

where $\| S_{f_{ik}} \|$ is the area of the face f_{ik}, i.e. the face between cells $\mathcal{T}_i$ and $\mathcal{T}_k$.

- **Least-Square method**

The third method is based on the Least-Square technique and was introduced by *Ollivier-Gooch and Van Altena* in 2002 [15]. The Taylor expansion of the unknown u in any mesh cell $\mathcal{T}_i$ is

$$u(x, y, z) \simeq u(x_i, y_i, z_i) + \nabla u\big|_i \cdot \begin{pmatrix} x - x_i \\ y - y_i \\ z - z_i \end{pmatrix}. \tag{16}$$

If we integrate over cell i, it comes

$$\frac{1}{\| \mathcal{T}_i \|} \int_{\mathcal{T}_i} u dV = u(x_i, y_i, z_i), \tag{17}$$

since by definition of the cell center we have:

$$\frac{1}{\| \mathcal{T}_i \|} \int_{\mathcal{T}_i} (x - x_i) dV = \frac{1}{\| \mathcal{T}_i \|} \int_{\mathcal{T}_i} (y - y_i) dV = \frac{1}{\| \mathcal{T}_i \|} \int_{\mathcal{T}_i} (z - z_i) dV = 0, \tag{18}$$

This is called *Conservation of the mean* in [14].

Secondly, integrating Equation 16 over cell j, one obtains

$$\frac{1}{\| \mathcal{T}_i \|} \int_{\mathcal{T}_i} u dV = u(x_i, y_i, z_i) + \nabla u\big|_i \cdot \begin{pmatrix} \hat{x}_{ji} \\ \hat{y}_{ji} \\ \hat{z}_{ji} \end{pmatrix}, \tag{19}$$

where $\hat{x}_{ji}$ is defined as a very simple algebraic realtion:

$$\hat{x}_{ji} = \frac{1}{\| \mathcal{T}_j \|} \int_{\mathcal{T}_j} (x - x_i) dV \tag{20}$$

From an ease computation of Equation 20 and exploiting Equation 18, we obtain that

$$\hat{x}_{ji} = x_j - x_i$$

and the same kind of relation is obtained for the terms $\hat{y}_{ji}$ and $\hat{z}_{ji}$.

Finally, a system of equations must be built to compute the components of the gradient at the center of cell i. For any cell j in the considered stencil, the relation is:

$$\begin{pmatrix} \hat{x}_{ji} \\ \hat{y}_{ji} \\ \hat{z}_{ji} \end{pmatrix} \cdot \begin{pmatrix} \frac{\partial u}{\partial x}\big|_i \\ \frac{\partial u}{\partial y}\big|_i \\ \frac{\partial u}{\partial z}\big|_i \end{pmatrix} = \overline{u_j} - \overline{u_i}. \tag{21}$$

If Equation 21 is applied to any cell in the considered stencil of the cell i, a general linear system of equations of the form $Ax = b$ must be inverted. If one restricts the stencil to the first row of neighboring cells, the system is over-conditioned and it then solved using the least-square method detailed in [15].

The final result gives us the evaluation of the cell-centred gradient for each cell of the domain.

2.2.2.2 *Alternative schemes for interface gradient evaluation*

The cell-centred scheme proposed in Equation 12 is the simplest method found in literature. However, for this approximation to be correct, it is necessary for the mesh to be extremely regular and that the center of gravity of the face is exactly at the centre of the segment joining the two cell centres. For structured meshes, the approximation may be correct in some cases, but if unstructured meshes are used, the approximation is inaccurate or invalid. To improve the solution in the case of meshes with deformations (related to the geometry to be discretized or the type of elements used), it is preferable to use schemes in which the contributions of the gradients at the centre of adjacent cells are weighted appropriately.

As we will see in Section 2.3, Leterrier's method is extremely suitable for the calculation of the gradient at the centre of the face since centre cell gradients are weighted in such a way that distortions and imperfections in the mesh are taken into account.

2.3 LETERRIER'S DIFFUSION SCHEME

This paragraph describes a cell-centred scheme for heat diffusion proposed by Nikos Leterrier in his PhD thesis work in 2003 [12]. This scheme is extremely interesting as it allows to

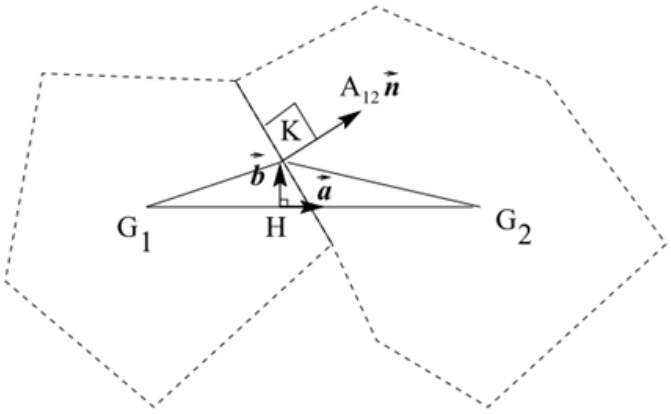


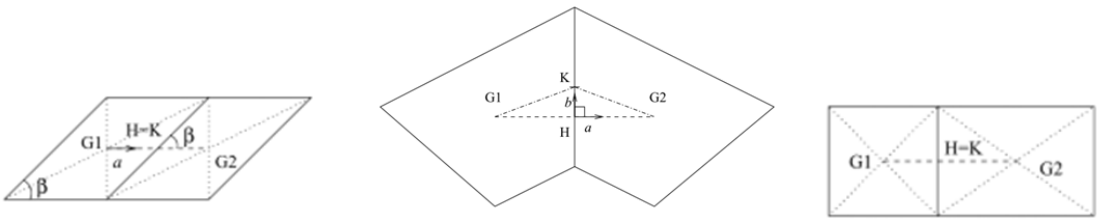
Figure 5: Geometric reference used for building Leterrier's scheme. Image taken from [12].

consider all the characteristic deformations of a structured or unstructured mesh. Furthermore, it is a scheme that is versatile enough to be adapted to hybrid meshes, i.e. composed of different cell types. Having well in mind the purpose of the study under consideration in this thesis, it is clear that it is of great importance to have a scheme of these characteristics. In fact, given the particularities of anisotropic and multilayer materials, it will be necessary to study in detail the diffusion of heat even at the interface between two layers with different orientation and consequently characterised by different meshes.

As shown in Figure 5, it is possible to define a number of geometrical quantities on which Leterrier's schemes relies. Firstly, we define the segment $G_1\vec{G}_2$ connecting the centres of the two adjacent cells and oriented along the vector $\vec{a} = \frac{G_1\vec{G}_2}{\|G_1\vec{G}_2\|}$. Point K is the central point of face f_{12} (interface between cell 1 and cell 2) characterised by the normal vector $\vec{n}$ and the surface of absolute value A_{12}. Finally, the point H is defined as the orthogonal projection of K onto $G_1\vec{G}_2$.

Based on these quantities, it is possible to define different types of geometrical properties in the mesh (see Figure 6).

- the non-orthogonality, i.e. angle between the directions $\vec{a}$ and $\vec{n}$ (irregularity bringing the worst consequences on stability and accuracy of the simulation);
- the relative curvature between one element and another, taken into account by the distance $\overline{HK}$;
- the heterogeneity in the direction $\vec{a}$, i.e. ratio between $\overline{HG_1}$ and $\overline{HG_2}$.



(a) Non-orthogonality (b) Curvature (c) Heterogeneity

Figure 6: Geometrical deformations in a 2D mesh. Images taken from [12].

The analytical formulation of the numerical scheme makes it possible to derive the gradient at the midpoint knowing the gradient at the H point, the quantities at the centre of the cells (we will indicate with $(\cdot)_i$ and $(\cdot)_j$ the two neighbour cells) and the geometrical characteristics of the mesh:

$$(\vec{\nabla}u)_K = (\vec{\nabla}u)_H + \theta_g \left[\frac{u_j - u_i}{\overline{G_i G_j}} - (\vec{\nabla}u)_H \cdot \vec{a} \right] \vec{v} \tag{22}$$

The parameters θ_g and $\vec{v}$ are a priori defined local variables that provide a general form of the scheme. The appropriate choice of these parameters prevents local distortions in certain areas of the mesh from leading to instability or loss of accuracy.

$(\vec{\nabla}u)_H$ is the barycentric gradient evaluated at point H by the first order interpolation formula between the gradients of adjacent elements.

$$(\vec{\nabla}u)_H = \alpha_i (\vec{\nabla}u)_i + \alpha_j (\vec{\nabla}u)_j = \frac{\overline{HG_j}}{\overline{G_i G_j}} (\vec{\nabla}u)_i + \frac{\overline{HG_i}}{\overline{G_i G_j}} (\vec{\nabla}u)_j \tag{23}$$

The gradients at the centre of the mesh cells are evaluated using Quasi Green or Least Squares methods that were presented earlier (see paragraph 2.2.2.1).
This class of schemes is consistent at order 2 and the presented scheme is the linear scheme with this most general characteristic using u_i, u_j, $(\vec{\nabla}u)_H$. Demonstrations of these properties can be found in Leterrier's thesis. This formulation of the scheme, which is as general as possible, is applicable to 3D, 2D and 1D geometries.

2.3.1 *Schemes for Diffusion Fluxes*

The objective of the method developed by Leterrier is to calculate the fundamental diffusive flux term to solve the Equation 10. In the present work, the aim is to study and solve the heat equation, and thus the state variable u will be temperature T. The diffusion flux Φ is therefore defined as

$$\Phi_{ij} = \vec{\phi}_{ij} \cdot A\vec{n} = -k(\vec{\nabla}u)_K \cdot A\vec{n} = -kA(\vec{\nabla}u)_H \cdot \vec{n} - kA\theta_g \left[\frac{u_j - u_i}{G_i G_j} - (\vec{\nabla}u)_H \cdot \vec{a} \right] \vec{v} \cdot \vec{n}, \tag{24}$$

where k is the material conductivity at the considered interface, A is the surface of the interface and $\vec{n}$ is the normal vector to it. These are local variables which are different from one interface to another.

It can be noticed that the diffusion flux depends on the product $\omega_{ij} = \theta_g \vec{v} \cdot \vec{n}$, which can be seen as a local variable depending on each interface. The choice of this term is important to distinguish between different schemes, belonging to the same family.

In his thesis [12], Leterrier proposes four[2] different schemes which are summarized in Table 1.

SCHEME	ω_{ij}	CHARACTERISTICS
1	0	Not very accurate
2	$\theta_g \vec{a} \cdot \vec{n}$	Initially in CEDRE
3	$\frac{\theta'_g}{\vec{a} \cdot \vec{n}}$	Optimal scheme
4	$\frac{(\vec{\nabla} u)_H \cdot \vec{n}}{(\vec{\nabla} u)_H \cdot \vec{a}}$	Non-linear scheme

Table 1: Numerical schemes proposed in [12] as a function of the local variable ω_{ij}.

In this report, the focus is exclusively on the study and implementation of scheme number 3, which guarantees the best results in terms of stability and stationary state quality.

2.3.2 *Characteristics of the optimal scheme*

Scheme 3 is considered to be the optimal one as it is compact without losing consistency.

The compactness of a scheme has several advantages. The error will necessarily be smaller compared to a scheme which uses values of elements far from the considered cell. Moreover, in the case of an unstructured mesh, compactness limits the influence of geometric irregularities since only the closest cells are used. Finally, it also limits the influence of possible discontinuities (shocks, etc.), and the schemes used for the boundary cells become consistent with those used for the internal cells. In Leterrier's thesis is seen that this compactness property allows for a better spatial and temporal stability, as well as the suppression, in some 1D cases, of the oscillations of the stationary solution.

Using this scheme for computing the gradient, one obtains

$$(\vec{\nabla} u)_K \cdot \vec{n} = (\vec{\nabla} u)_H \cdot \vec{n} + \frac{\theta_0}{\vec{a} \cdot \vec{n}} \left[\frac{u_j - u_i}{G_i G_j} - (\vec{\nabla} u)_H \cdot \vec{a} \right] \cdot \vec{n} \tag{25}$$

where the best choice for the constant θ_0 is 1. We will numerically analyse this method in Section 3.4 by implementing Leterrier's scheme in the DEEP-DIVE code and analysing its characteristics.

2 Actually, in Leterrier's thesis [12], we can find eight different schemes. In four of them, $\vec{v} = \vec{a}$ and if the other four $\vec{v} = \vec{n}$. In this report, we will focus on the four methods using $\vec{v} = \vec{a}$.

2.4 JACQ'S DIFFUSION SCHEME

This paragraph describes the cell-centred scheme for heat diffusion developed by Pascal Jacq in his PhD thesis work in 2015 [11]. The main purpose of this thesis is to develop efficient parallel numerical methods on unstructured grids to contribute to the improvement of the numerical modeling of the global atmospheric re-entry problem.

This scheme is a Cell-Centred Finite Volume scheme which means that the unknowns are placed at the center of the grid cells. The main feature of the scheme is the particular partitioning of the domain based on the vertices of each cell rather than its faces. Each mesh element is divided into sub-cells and each face is subdivided into sub-faces. The generation of sub-cells allows for better handling complex unstructured meshes that are used to simulate heat diffusion problems within anisotropic materials composed of different layers.

The sub-face normal fluxes impinging at a vertex are expressed in terms of the difference between sub-face temperatures and the cell-centred temperature. This approximation of the sub-face fluxes results from a local variational formulation written over each sub-cell. The sub-face temperatures, which are auxiliary unknowns, are locally eliminated by invoking the continuity of the temperature and the normal component of the heat flux across each cell interface.
Gathering the contribution of each vertex allows to construct easily the global sparse diffusion matrix (Equation 39). The scheme stencil is local and for a given cell consists of the cell itself and its node-based neighbors. This Cell-Centred Finite Volume scheme can be used to solve the anisotropic heat diffusion both in 2D and 3D geometries.

2.4.1 *Geometry Discretisation*

The main feature of this finite volume scheme relies on the 3D (2D) partition of each polyhedral (polygonal) cell of the computational domain into sub-cells and on the partition of each cell face into sub-faces, which in the three dimensional case are composed by triangular faces.

2D PARTITION: As displayed in Figure 7a, for each vertex $p \in \mathcal{P}(c)$, a sub-cell (named ω_{pc}) is defined by connecting the centroid of ω_c to the midpoints of edges $[p^-, p]$ and $[p, p^+]$ impinging at node p. In 2D, the sub-cell is always a quadrilateral regardless of the type of cells that compose the underlying grid.

$\mathcal{P}(c)$ is the list of vertices of the cell c, (named ω_c)

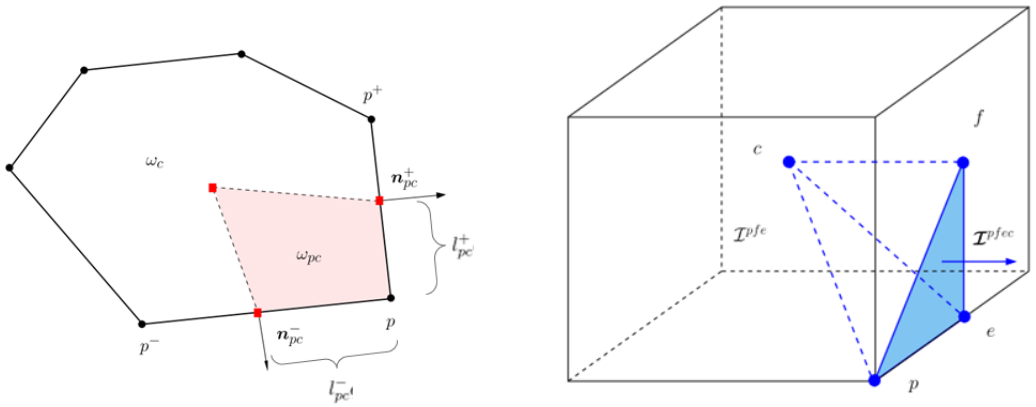
3D PARTITION: The polyhedral cell is firstly partitioned in elementary tetrahedrons $\mathcal{I}^{pfec}$, called *iota*, related to point $p \in \mathcal{P}(c)$, face $f \in \mathcal{F}(c)$ and the edge e. This elementary volume is constructed by connecting point p, the centroid of cell c, the centroid of face f and the midpoint of edge e (as displayed in Figure 7b).

$\mathcal{F}(c)$ is the list of faces of the cell c

The sub-cell ω_{pc} is obtained by gathering the iotas attached to point p as follows

$$\omega_{pc} = \bigcup_{f \in \mathcal{F}(p,c)} \bigcup_{e \in \mathcal{E}(p,f)} \mathcal{I}^{pfec}$$

$\mathcal{E}(p, f)$ is the set of edges of face f impinging on p



(a) 2D geometry discretisation (b) 3D geometry discretisation

Figure 7: Partition of cell c into sub-cells impinging at node p named ω_{pc}. Images taken from [11].

2.4.2 *Sub-cells variational formulation*

The starting point to derive the sub-cell based variational formulation consists in writing the partial differential equation satisfied by the heat flux. From heat flux definition in Equation 2, it follows that $\vec{\Phi}$ satisfies:

$$\mathbb{K}^{-1}\vec{\Phi} + \vec{\nabla}T = 0$$

As fully detailed in Jacq's thesis [11], following some algebraic steps it is possible to obtain a local explicit expression of the half-edge fluxes in terms of the half-edge temperatures and the mean cell temperature. The approach followed by Jacq is strongly linked to the mixed formulation utilized in the context of mixed finite element discretization. In particular, both in 3D and in 2D, the main steps followed are

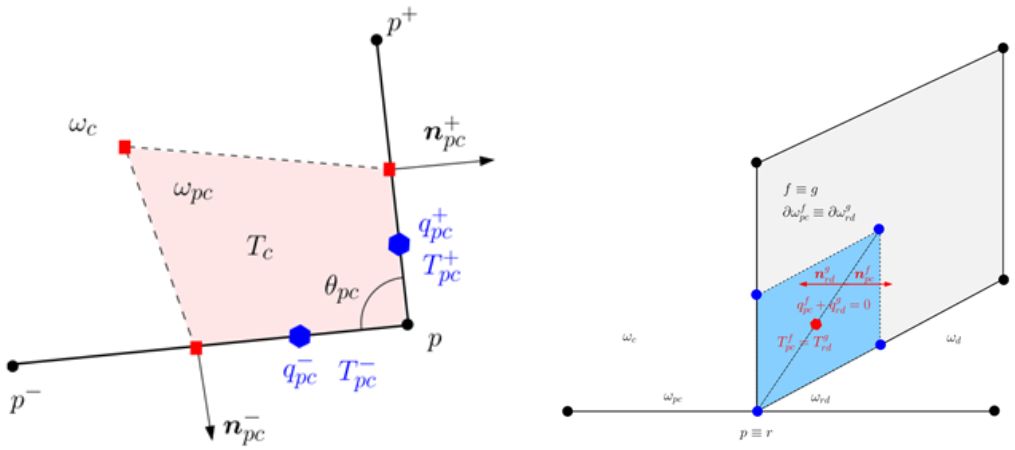
1. multiplication of the equation for an arbitrary function;
2. integration over the cell ω_{pc};
3. discretization by using geometry and function properties (e. g.piecewise-constant functions).

All notations used in the following refer to Figure 9.

2D LOCAL VARIATIONAL FORMULATION. For each sub-cell ω_{pc}, the half-edge normal fluxes related to this volume are two, one for each sub-face.

$$\begin{pmatrix} q^-_{pc} \\ q^+_{pc} \end{pmatrix} = -\frac{1}{\omega_{pc}} (\mathbb{J}^t_{pc} \mathbb{K}_c \mathbb{J}_{pc}) \begin{bmatrix} l^-_{pc}(T^-_{pc} - T_c) \\ l^+_{pc}(T^+_{pc} - T_c) \end{bmatrix} \tag{26}$$

where the local normal vectors matrix is defined as $\mathbb{J}_{pc} = [\vec{n}^-_{pc}, \vec{n}^+_{pc}]$. l_{pc} is the length of the sub-face and ω_{pc} is the volume of the sub-cell (in the 2D case is the surface of the sub-cell).



(a) 2D notation (b) 3D notation

Figure 8: Local notation for sub-cells. Images taken from [11].

3D LOCAL VARIATIONAL FORMULATION. For each sub-cell ω_{pc}, the half-edge normal fluxes related to this volume are three, one for each sub-face.

$$\begin{pmatrix} q^1_{pc} \\ q^2_{pc} \\ q^3_{pc} \end{pmatrix} = -\frac{1}{\omega_{pc}} (\mathbb{J}^t_{pc} \mathbb{K}_c \mathbb{J}_{pc}) \begin{bmatrix} A^1_{pc}(T^1_{pc} - T_c) \\ A^2_{pc}(T^2_{pc} - T_c) \\ A^3_{pc}(T^3_{pc} - T_c) \end{bmatrix} \tag{27}$$

where the local normal vectors matrix is defined as $\mathbb{J}_{pc} = [\vec{n}^1_{pc}, \vec{n}^2_{pc}, \vec{n}^3_{pc}]$. A_{pc} is the surface of the sub-face coloured in blue in Figure 8b and ω_{pc} is the volume of the sub-cell related to the vertex p.

It is important to notice that the corner conductivity tensor $\mathbb{K}_{pc} = \mathbb{J}^t_{pc} \mathbb{K}_c \mathbb{J}_{pc}$ inherits all the properties of the conductivity tensor $\mathbb{K}_c$.

In order to have a simpler expression for Equation 26 and Equation 27, the author introduces some new local notations.

2D LOCAL GENERALISED FORMULATION. For the two-dimensional case, the new local notation can be understood from the schema in Figure 9.
Equation 26 can be rewritten as

$$\begin{pmatrix} q^c_c \\ q^c_{c+1} \end{pmatrix} = -\alpha_{pc} \mathbb{K}_{pc} \begin{bmatrix} l_c(\bar{T}_c - T_c) \\ l_{c+1}(\bar{T}_{c+1} - T_c) \end{bmatrix} \tag{28}$$

where $\alpha_{pc} = \frac{1}{\omega_{pc}}$.

3D LOCAL GENERALISED FORMULATION. For each face f in the list F(p) of the faces impinging at the node p two tuples *(c,i)* and *(d,j)* are associated. Indeed, the face f belongs to two different cells, i.e. c and d, and it is identified by a local index, respectively i and

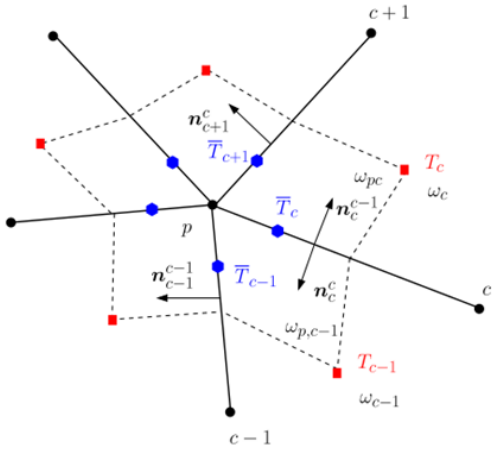


Figure 9: Changing in local notation for the 2D case. Image taken from [11].

j. If we consider the most common types of cells in meshes, the total number of faces impinging at a node and belonging to a single cell are 3, and so index i (and j) varies in a range between 1 and 3.
A particular case is represented by pyramidal cells, where faces which contain the top vertex and belong to the cell are 4. This exception must be treated differently than usual and a specific study will be conducted in Chapter 3. Pyramidal cells are fundamental in a mesh as they are used as a transition layer between hexahedral and tetrahedral cells.

Using this notation, Equation 27, which defines the heat flux approximation, rewrites

$$q_c^i = -\alpha_c \sum_{k=1}^{3} \mathbb{K}_{ik}^c A_c^k (\bar{T}_c^k - T_c) \tag{29}$$

where A_c^k is the surface of the face k (local numbering) of cell c. In the same way, $\bar{T}_c^k$ is the sub-face temperature relative to the face k of cell c.

Sub-face temperature is an auxiliary variable which is not needed to be computed. For this reason, we may express it as a function of the cell-center temperature. In order to do that, the flux continuity at the interface between cells has been exploited.

$$(\mathbb{K}\nabla T)_1 \cdot \vec{n}_{12} = (\mathbb{K}\nabla T)_2 \cdot \vec{n}_{12} \tag{30}$$

2D: CONTINUITY AT THE INTERFACE. In two dimension and with the local notation introduced before, Equation 30 can be expressed as

$$l_c q_c^{c-1} + l_c q_c^c = 0 \tag{31}$$

If we substitute the expressions found for the heat flux, we obtain the extended expression for the continuity equation.

$$l_c^i q_c^i + l_d^j q_d^j = 0 \tag{32}$$

3D: CONTINUITY AT THE INTERFACE. We apply the same procedure to the case 3D and we obtain

$$-\alpha_c A_c^i \sum_{k=1}^{3} \mathbb{K}_{ik}^c A_c^k (\bar{T}_c^k - T_c) - \alpha_d A_d^j \sum_{k=1}^{3} \mathbb{K}_{jk}^d A_d^k (\bar{T}_d^k - T_d) = 0 \tag{33}$$

Both expressions can be written using a matrix formulation in a more compact form, remembering that T is the cell-center temperature while $\bar{T}$ is the sub-face temperature.

$$\mathbb{N}\bar{T} = \mathbb{S}T \tag{34}$$

where $\mathbb{N}$ is a $F_p x F_p$ matrix and $\mathbb{S}$ is a $F_p x C_p$ matrix. The formulation of the single element of these matrices is shown in the annex.

F_p and C_p are respectively the number of faces and cells impinging at point p.

Finally, thanks to this formulation we are able to express the sub-face temperature as a function of the cell-center one. To do so, it is important to be sure that matrix $\mathbb{N}$ can be inverted. In Jacq's thesis [11], the invertibility of this matrix is demonstrated and in particular $\mathbb{N}$ is found to be a positive-definite matrix.

$$\bar{T} = (\mathbb{N}^{-1}\mathbb{S})T \tag{35}$$

2.4.3 *Diffusion matrix definition*

Using considerations of the previous paragraph, we have all the elements to achieve the spatial discretization of the local diffusion matrix and obtain an analytical expression for the diffusion flux.

Starting from Equation 10 and taking into account space discretization explained previously, we are able to construct the space discretization of our diffusion problem.

Basing on the sub-cells formulation for the cell c, one can write

$$m_c C_{pc} \frac{d}{dt} T_c + \sum_{f \in \mathcal{F}(c)} \int_{\partial \omega_c^f} \vec{\Phi} \cdot \vec{n} dS = m_c r_c,$$

where m_c is the mass of the cell defined as $m_c = \rho_c \parallel \mathcal{T}_c \parallel$. Furthermore, recalling the partition of face f intro sub-cells, i.e. $\partial\omega_c^f = \cup_{p \in \mathcal{P}(c,f)} \omega_{pc}^f$, leads to write the surface integral as

$$\sum_{f \in \mathcal{F}(c)} \int_{\partial \omega_c^f} \vec{\Phi} \cdot \vec{n} dS = \sum_{p \in \mathcal{P}(c)} \sum_{f \in \mathcal{F}(p,c)} \int_{\partial \omega_{pc}^f} \vec{\Phi} \cdot \vec{n} dS.$$

Finally, denoting by q_{pc}^f the piecewise constant representation of the normal component of the heat flux over the sub-face $\partial\omega_{pc}^f$

$$q_{pc}^f = \frac{1}{A_{pc}^f} \int_{\partial \omega_{pc}^f} \vec{\Phi} \cdot \vec{n} dS$$

and gathering the above equations, it is possible to write the variational formulation of Equation 4 as

$$m_c C_{pc} \frac{d}{dt} T_c + \sum_{p \in \mathcal{P}(c)} \sum_{f \in \mathcal{F}(p,c)} A^f_{pc} q^f_{pc} = m_c r_c. \tag{36}$$

We define the contribution of the sub-cell ω_{pc} to the diffusion flux as

$$Q_{pc} = \sum_{f \in \mathcal{F}(p,c)} A^f_{pc} q^f_{pc} = \sum_{k=1}^{3} A^k_c q^k_c.$$

Substituting the heat flux approximation (29), we obtain

$$Q_{pc} = -\sum_{k=1}^{3} A^k_c \left[\alpha_c \sum_{i=1}^{3} \mathbb{K}^c_{ki} A^i_c (\bar{T}^i_c - T_c) \right].$$

Interchanging the order of the summations in the right-hand side and defining the matrix $\tilde{S}$ whose entries write $\tilde{S}_{fc} = \alpha_c \sum_{k=1}^{3} (A^i_c \mathbb{K}^c_{ki} A^k_c)$, we obtain a more compact form of Q_{pc}

$$Q_{pc} = -\sum_{f \in \mathcal{F}(p)} \tilde{S}^t_{cf} (\bar{T}^i_c - T_c). \tag{37}$$

The auxiliary variable $\bar{T}_c$ can be eliminated by means of Equation 35, leading to the compact formulation

$$Q_{pc} = -\sum_{d \in \mathcal{C}(p)} \mathbb{G}^p_{cd} (T_d - T_c), \tag{38}$$

where $\mathbb{G}^p$ is a $C_p x C_p$ matrix defined at point p by

$$\mathbb{G}^p = \tilde{S}^t \mathbb{N}^{-1} S, \tag{39}$$

$\mathbb{G}^p$ have the physical dimension of a thermal conductivity and it can be considered as the effective conductivity tensor at point p. Furthermore, the single term $\mathbb{G}^p_{cd}$ stands for the conductivity between cells c and d through the sub-face $(c,i) = (d,j)$ belonging to the sub-cell impinging at p.

The semi-discrete scheme over cell c reads

$$m_c C_{pc} \frac{d}{dt} T_c + \sum_{p \in \mathcal{P}(c)} \sum_{d \in \mathcal{C}(p)} \mathbb{G}^p_{cd} (T_d - T_c) = m_c r_c, \tag{40}$$

This equation allows to construct the generic entries of the global diffusion matrix, D, as follows

$$\mathbb{D}_{cc} = \sum_{p \in \mathcal{P}(c)} \sum_{d \in \mathcal{C}(p)} \mathbb{G}^p_{cd},$$

$$\mathbb{D}_{cd} = -\sum_{p \in \mathcal{P}(c)} \mathbb{G}^p_{cd}, c \neq d.$$

The vector of cell-centred temperatures, $\mathcal{T} \in \mathcal{R}^{C_D}$ is a solution of the system of differential equations

C_D is the number of cells composing the computational grid

$$\mathbb{M}\mathbb{C}_p \frac{d}{dt}\mathcal{T} + \mathbb{D}\mathcal{T} = \mathbb{M}R, \tag{41}$$

where $R \in \mathcal{R}^{C_D}$ is the source term vector, $\mathbb{M}$ and $\mathbb{C}_v$ are the diagonal matrices whose entries are respectively the cell mass m_c and the cell heat capacity C_{vc}.

2.4.4 *Boundary Conditions*

As in any numerical model, it is essential to define a methodology for imposing boundary conditions that allow different phenomena to be simulated. In order to take the boundary terms into account, the vector B containing the boundary conditions was introduced.

$$\mathbb{N}\bar{T} = \mathbb{S}T + B. \tag{42}$$

The terms of this vector depends strictly on the boundary conditions types under consideration. The modifications to bring to the matrices and boundary vector are described in the annex.

Solving the system 42 we obtain a new formulation for $\bar{T}$ as a function of T, where boundary conditions are taken into account.

$$\bar{T} = (\mathbb{N}^{-1}\mathbb{S})T + \mathbb{N}^{-1}B.$$

If we substitute this expression into Equation 37, the diffusion flux Q_{pc} turns into

$$Q_{pc} = -\sum_{d \in \mathcal{C}(p)} \mathbb{G}^p_{cd}(T_d - T_c) - (\tilde{\mathbb{S}}^t\mathbb{N}^{-1}B)_c, \tag{43}$$

where the effective conductivity tensor $\mathbb{G}^p$ is defined by Equation 39.

Finally, the global linear system (36) corresponding to Jacq finite volume scheme becomes

$$\mathbb{M}\mathbb{C}_p \frac{d}{dt}\mathcal{T} + \mathbb{D}\mathcal{T} = \mathbb{M}R + \Sigma, \tag{44}$$

where Σ is the vector containing the boundary condition contributions, whose cth entry is given by $\Sigma_c = (\tilde{\mathbb{S}}^t\mathbb{N}^{-1}B)_c$.

2.5 TIME INTEGRATION

In this paragraph, we will discuss various approaches to achieving time integration. When speaking of temporal discretization, it is necessary to distinguish the existing methods into

two broad categories: explicit and implicit methods. In explicit methods, the solution at step $k+1$ is easily obtained by solving an equation that depends only on the time step and the solution at the previous step u_k. On the contrary, for implicit methods, the solution at time step u_{k+1} depends on the solution u_{k+1} itself and therefore it will be necessary to solve a linear system at each time step. If from a computational cost point of view explicit methods are extremely convenient, implicit methods can assure an important characteristic: unconditioned stability. This means that the stability of the solution does not depend on the time step chosen and so even relatively high time steps can be used.

Furthermore, integration in time can be achieved through one-step methods (like Euler method) or multi-step methods (like Runge-Kutta or Linear Multi-step Methods).

- One-step methods.
  Explicit: $u_{k+1} = u_k + \Delta t \cdot \mathcal{F}(t_k, u_k; \Delta t)$
  Implicit: $u_{k+1} = u_k + \Delta t \cdot \mathcal{F}(t_k, u_k, u_{k+1}; \Delta t)$

- Multi-step methods.
  Explicit: $u_{k+1} = u_k + \Delta t \cdot \mathcal{F}(t_k, u_{k-p}, ..., u_{k-1}, u_k; \Delta t)$
  Implicit: $u_{k+1} = u_k + \Delta t \cdot \mathcal{F}(t_k, u_{k-p}, ..., u_{k-1}, u_k, u_{k+1}; \Delta t)$

In the next section, we will briefly describe the *Runge Kutta* methods, which are mainly used in the numerical applications detailed in the second part of the report.

2.5.1 *Runge-Kutta Multi-Step Methods*

Runge-Kutta methods are commonly used for integrating ODEs where accuracy is built up through solving the ODE in a series of steps (or stages). The simplest method belonging to the Runge-Kutta family of methods is the Euler method, which corresponds to a one-step Runge Kutta scheme. However, this numerical method is seldom used as it is not very accurate (having a time convergence order equal to 1) but above all has a very low stability limit. To improve the convergence order and stability, it is possible to increase the number of steps of the method.

There are several Runge-Kutta type schemes depending on the number of steps used, but the most widely known member of the Runge–Kutta family is the 4-steps method (generally referred to as *RK4*).

$$\begin{cases} \frac{u_1 - u_k}{\Delta t} & = \frac{1}{2}\mathcal{F}(u_k, t_k) \\ \frac{u_2 - u_k}{\Delta t} & = \frac{1}{2}\mathcal{F}(u_1, t_{k+1/2}) \\ \frac{u_3 - u_k}{\Delta t} & = \frac{1}{2}\mathcal{F}(u_2, t_{k+1/2}) \\ \frac{u_{k+1} - u_k}{\Delta t} & = \frac{1}{6}\left[\mathcal{F}(u_k, t_k) + 2\mathcal{F}(u_1, t_{k+1/2}) + 2\mathcal{F}(u_2, t_{k+1/2}) + \mathcal{F}(u_3, t_{k+1})\right] \end{cases} \tag{45}$$

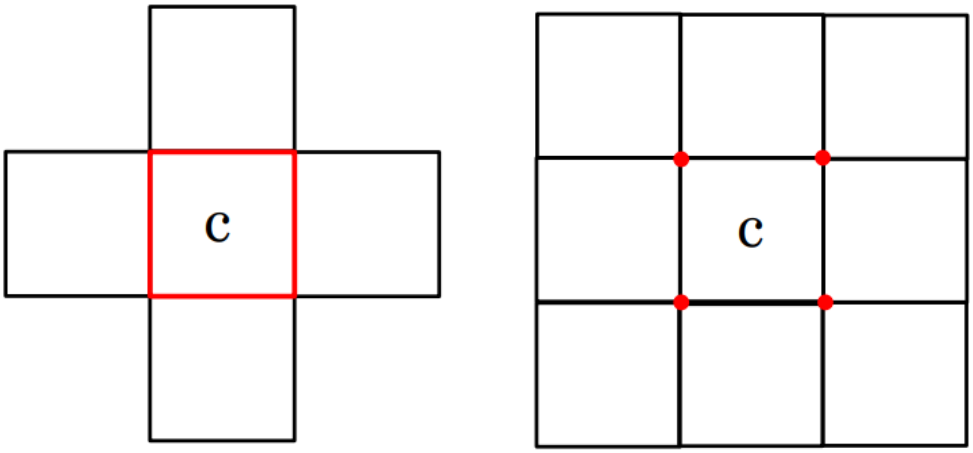
For the numerical study performed in the second part, the RK4 method was used, which is already implemented in DEEP-DIVE. This made it possible to reduce calculation times while maintaining good stability.

2.6 SUMMARY AND REMARKS

The two schemes detailed in Section 2.3 and Section 2.4 are based on the same principle, namely that of the Finite Volumes Method. In particular, they allow the domain to be discretized into cells, diffusive fluxes to be calculated across the cell faces and the quantities within each cell to be updated at each time step. To do this, both use only the variable calculated at the centre of the cell, hence the definition of cell-centred schemes.

However, the two methods show significant differences that distinguish them. In particular, Leterrier's method is based on a classical domain discretization and on a variational formulation widely used in finite volume schemes. The variable at the centre of the cell depends on the cell itself and neighbouring cells that share a face with the main cell. Leterrier's scheme differs from classical methods in that it allows for the gradient at the centre of the cell faces to be defined in a way that takes into account imperfections in the mesh.

By contrast, Jacq's method is different by construction. In fact, even if it is still a cell-centred method, it is no longer based on a classical discretization, but the mesh is further subdivided into sub-cells and sub-faces. This further subdivision of the mesh makes it possible to take into account to a greater extent the irregularity of the mesh and the geometry and thus to calculate all quantities more precisely. The variational formulation that follows is therefore different from the classical formulation and allows us to obtain a linear system that provides the unknowns of our problem. The conservation of the solution is guaranteed by the condition of continuity imposed on each face, which also allows the only unknown in the problem to be the variable considered (in this case the temperature at the centre of the cells). Another difference with Leterrier's scheme is the scheme stencil. In this case, the solution depends on the cell itself and on the neighbouring cells that share a vertex with the current cell.



(a) Leterrier's stencil composed by the main cell and the neighbouring cells by faces.

(b) Jacq's stencil composed by the main cell and the neighbouring cells by vertex.

Figure 10: Schematic stencil used in the two schemes for the main cell c.

Part II

NUMERICAL APPLICATIONS

This section will illustrate how the numerical models described previously have been applied in practice. 2D and 3D cases for structured and unstructured meshes will be analysed both for Leterrier's and Jacq's scheme. Furthermore, the code in which these methods were introduced, namely *DEEP-DIVE*, will be briefly described. Finally, we will summarise all results, analyse critical issues and define future objectives.

3

DEEP-DIVE

3.1 INTRODUCTION

DEEP-DIVE is a numerical calculation code, conceived and written in *Fortran* by *Guillaume Puigt*. It allows 2D and 3D Navier-Stokes equations to be solved numerically using a finite volume scheme and Runge Kutta time integration. Due to its simple and intuitive structure, it was used during the stage work to test and validate the new numerical methods described in the first part. During the internship at ONERA, the algorithm for solving the heat equation, the algorithm for the Leterrier method and the Jacq method in both two and three dimensions were implemented in this code.

3.2 ANALYTICAL TEST CASE

As mentioned in part 1, the first objective of the internship work is to test and validate Leterrier's and Jacq's methods and find their order of convergence. The best way to proceed is to use an analytical case study for which the exact solution is known, with which to compare the numerical solutions obtained from the numerical models. The case study considered is the one described in [19].

The paper provides the stationary analytical solution of the heat equation (Equation 5) in two dimensions on a square $[0, L]x[0, L]$. Considering the source term equal to 0 and by imposing the following boundary conditions

$$T_B(x, y) = \begin{cases} 0 & \forall x \in [0, L] \quad \text{and} \quad y = 0 \\ \sin(\pi \frac{x}{L}) & \forall x \in [0, L] \quad \text{and} \quad y = L \\ 0 & \forall y \in [0, L] \quad \text{and} \quad x = 0 \\ \sin(\pi \frac{y}{L}) & \forall y \in [0, L] \quad \text{and} \quad x = L \end{cases} \tag{46}$$

the exact stationary solution of the problem is

$$u^{ex}(x, y) = \frac{1}{\sinh \pi} \left[\sinh(\pi \frac{x}{L}) \sin(\pi \frac{y}{L}) + \sinh(\pi \frac{y}{L}) \sin(\pi \frac{x}{L}) \right] \tag{47}$$

It can easily be verified that the solution meets the boundary conditions and that the Laplacian of the stationary solution is null ($\Delta u^{ex} = 0$). The solution is shown in Figure 11a

Taking this 2D analytical solution as a starting point, the aim was to find a 3D analytical solution for a similar problem, so that the three-dimensional cases for the numerical

schemes studied could also be tested and validated. This time we will look for the exact analytical solution on a cube $[0, L]x[0, L]x[0, L]$

The new boundary conditions are

$$T_B(x, y) = \begin{cases} 0 & \forall x, z \in [0, L] \quad \text{and} \quad y = 0 \\ \sin(\pi\frac{x}{L})\sin(\pi\frac{z}{L}) & \forall x, z \in [0, L] \quad \text{and} \quad y = L \\ 0 & \forall y, z \in [0, L] \quad \text{and} \quad x = 0 \\ \sin(\pi\frac{y}{L})\sin(\pi\frac{z}{L}) & \forall y, z \in [0, L] \quad \text{and} \quad x = L \\ 0 & \forall x, y \in [0, L] \quad \text{and} \quad z = 0 \\ \sin(\pi\frac{x}{L})\sin(\pi\frac{y}{L}) & \forall x, y \in [0, L] \quad \text{and} \quad z = L \end{cases} \tag{48}$$

The exact stationary solution is

$$\begin{aligned} u^{ex}(x, y, z) = \frac{1}{\sinh(\sqrt{2}\pi)} \Big[& \sinh(\sqrt{2}\pi\frac{x}{L}) \sin(\pi\frac{y}{L}) \sin(\pi\frac{z}{L}) + \\ & \sin(\pi\frac{x}{L}) \sinh(\sqrt{2}\pi\frac{y}{L}) \sin(\pi\frac{z}{L}) + \\ & \sin(\pi\frac{x}{L}) \sin(\pi\frac{y}{L}) \sinh(\sqrt{2}\pi\frac{z}{L}) \Big] \end{aligned} \tag{49}$$

Again, it can be verified that the solution meets the boundary conditions and is a solution of the Equation 5. The solution is shown in Figure 11b.

3.3 TOOLS FOR THE NUMERICAL ANALYSIS

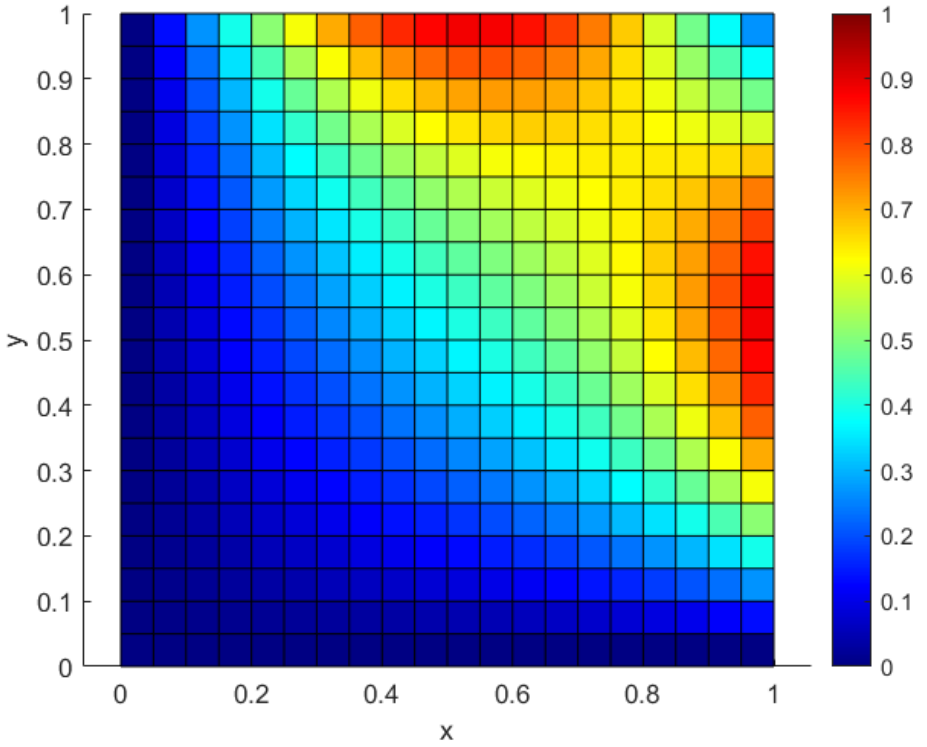
To perform the numerical analysis and determine the order of convergence, it was necessary to generate different types of meshes.

The implementation of a *cgns* reader made it possible to test different structured and unstructured meshes generated using *GMSH* grid generation tool [10] (version 4.9.0) and exported in the *.cgns* format. This tool was fundamental to calculate the order of convergence of the implemented methods and validate them.

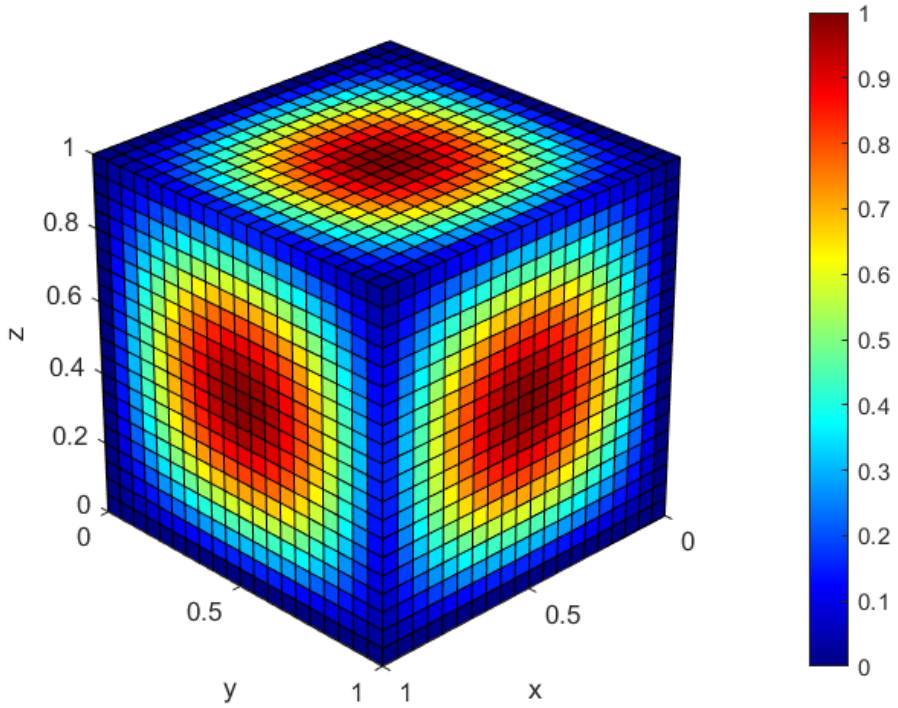
The meshes generated for the study and their main characteristics are shown in Table 2 and Table 3.

In order to test the accuracy of the methods under analysis, meshes were also created that presented criticalities and would be more difficult to handle by the numerical method. For example, hybrid meshes consisting of a first structured part and a second unstructured part were generated in 2D.

The results obtained from the simulation, saved on the same *cgns* file, are visualised using the open source software *paraview*. This made it possible to graphically analyse the consistency of the solution and possibly find critical points where the implemented method struggles to accurately calculate physical quantities.



(a) 2D analytical solution for a square made of 400 quadrangular elements.



(b) 3D analytical solution for a cube made of 8000 hexahedral elements.

Figure 11: Representation of the analytic exact solution of the pure diffusion problem

STRUCTURED	CELLS	H	UNSTRUCTURED	CELLS	H
$Str2D_{24}$	576	0.042	$Unstr2D_{010}$	244	0.064
$Str2D_{32}$	1024	0.031	$Unstr2D_{006}$	736	0.037
$Str2D_{48}$	2304	0.021	$Unstr2D_{005}$	940	0.033
$Str2D_{72}$	5184	0.014	$Unstr2D_{003}$	2738	0.019
-	-	-	$Unstr2D_{002}$	5826	0.013

Table 2: 2D structured and unstructured meshes used for convergence study.

STRUCTURED	CELLS	H	UNSTRUCTURED	CELLS	H
$Str3D_{9}$	729	0.111	$Unstr3D_{020}$	726	0.111
$Str3D_{10}$	1000	0.100	$Unstr3D_{015}$	1568	0.086
$Str3D_{15}$	3375	0.067	$Unstr3D_{010}$	4689	0.059
$Str3D_{17}$	4913	0.588	$Unstr3D_{009}$	8160	0.049
$Str3D_{20}$	8000	0.050	$Unstr3D_{008}$	10192	0.046
$Str3D_{40}$	64000	0.025	$Unstr3D_{005}$	36523	0.030

Table 3: 3D structured and unstructured meshes used for convergence study.

In the following studies, in order to determine the correct numerical error, a tolerance was imposed on the residual calculated in L_2 norm equal to 10^{-12}, which denotes that convergence has been achieved (stationary solution). In addition, the numerical error considered in the analysis is also calculated via the L_2 norm as in Equation 7.

3.4 LETERRIER'S SCHEME IMPLEMENTATION

During a first part of the internship work, the algorithm for solving the heat equation was implemented in the *DEEP-DIVE* code, respecting the block organisation of the code and partly exploiting the structures already existing.

Initially, the calculation of diffusive fluxes and in particular of gradients at the interface between cells was performed using a centred scheme, as described by Equation 12. As already pointed out, this approximation does not allow for inhomogeneities in the mesh to be effectively taken into account, particularly when dealing with unstructured meshes. Keeping in mind that the main objective is to study heat diffusion in anisotropic materials, it was decided to implement the Leterrier's scheme, so as to overcome this issue.

Following the mathematical formulation described in Section 2.3, it has been necessary to compute all the geometrical entities needed to obtain firstly the gradient at point H

and secondly the gradient at the center of the face. The latter can be seen as a weighted translation of the gradient at point H towards point K (centre of the face), as described by Equation 24.

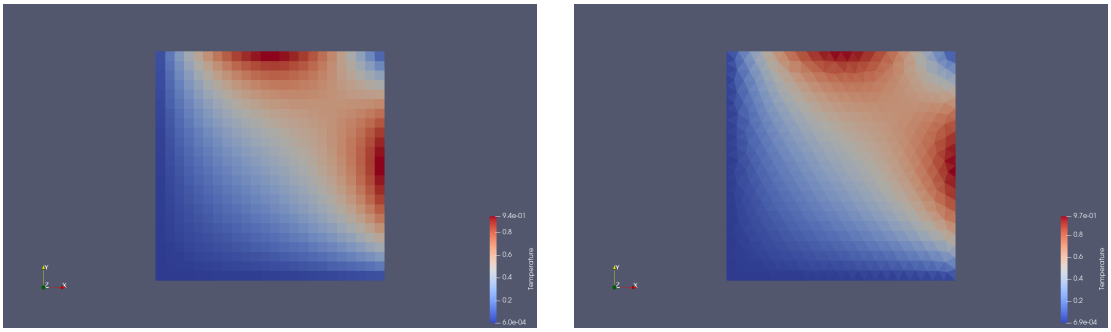
For the calculation of the gradients at the centre of the cell, the Quasi-Green method already implemented in the code was chosen, while for the temporal integration, the 4-step Runge-Kutta method was used.

3.4.1 *Accuracy and order of convergence*

Before studying the order of convergence of the method, it is necessary to determine its fidelity, i.e. whether the results obtained correspond to reality and make physical sense. Referring to the analytical case illustrated in Section 3.2, the boundary conditions for the 2D and 3D cases presented above and an initial condition $T_0 = 0°C$ were imposed throughout the domain.

To understand whether the result is physically acceptable and thus whether the equations have been solved correctly, one can proceed both graphically and numerically. The graphical solution is extremely useful to get an initial indication of fidelity, as it is possible to compare the numerical and analytical solution across the entire domain. Furthermore, in the case of an imprecise solution, it is possible to observe the graphical result to see in which areas of the grid the method does not provide correct results.

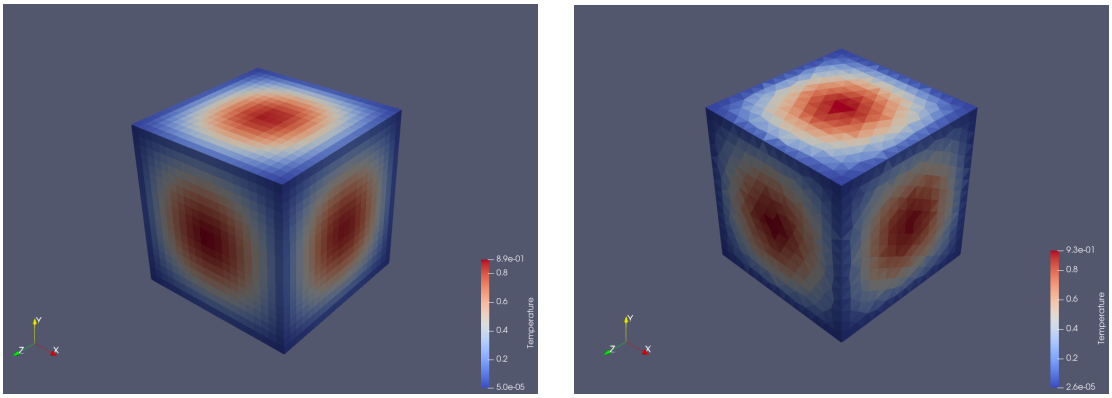
We report below the solution obtained using both an unstructured and a structured mesh having the same characteristic dimension as the analytical solution shown in Figure 11a and Figure 11b so that the two can be compared.



(a) 2D Structured mesh solution ($Str2D_{24}$). (b) 2D Unstructured mesh solution ($Unstr2D_{005}$).

Figure 12: 2D numerical solution obtained by using Leterrier's scheme.

The numerical solutions obtained in 2D using the Leterrier scheme are extremely accurate and faithful to the exact analytical solution (Figure 11a). Graphically, no particular problems with the accuracy of the solution are apparent, as all cells appear to be well treated. Despite a mesh that is not too dense and consists of few elements, the solution is very precise. However, to obtain a better solution, it is necessary to thicken the mesh, which corresponds to the study conducted to define the order of convergence.



(a) 3D Structured mesh solution ($Str3D_{20}$). (b) 3D Unstructured mesh solution ($Unstr3D_{008}$).

Figure 13: 3D numerical solution obtained by using Leterrier's scheme.

Comparing the two 3D numerical solutions with the analytical solution (Figure 11b), it is evident that the result obtained using Leterrier's scheme is extremely precise and accurate. In fact, the scheme succeeds in adequately dealing with edge zones and edges, which can sometimes cause numerical problems.

Having ascertained that the solution is physically consistent and corresponds to the analytical solution, we can proceed to the second study phase corresponding to the *verification* of the approximation order. Thus, we can observe what happens to the error when we decrease the number of cells in the mesh. We summarise the results obtained in the tables Table 6 and Table 7 and plot the evolution of the error as a function of h in the graph in Figure 14a and Figure 14b.

STRUCTURED	ERROR L_2	UNSTRUCTURED	ERROR L_2
$Str2D_{24}$	$2.32 \cdot 10^{-2}$	$Unstr2D_{006}$	$1.67 \cdot 10^{-3}$
$Str2D_{48}$	$1.31 \cdot 10^{-2}$	$Unstr2D_{005}$	$7.82 \cdot 10^{-4}$
$Str2D_{72}$	$9.21 \cdot 10^{-3}$	$Unstr2D_{003}$	$5.65 \cdot 10^{-4}$
-	-	$Unstr2D_{002}$	$1.56 \cdot 10^{-4}$

Table 4: 2D structured and unstructured errors using Leterrier's scheme, computed in norm L_2.

By analysing the data obtained, it is possible to understand how well the scheme works and whether the implementation was carried out correctly.

For a more immediate understanding of the results, we can analyse the curves plotted in Figure 14. In these plots[1], the exact values of the errors in the table are shown as a function of the characteristic mesh size (they are indicated with a small circle). These data were then interpolated using a polynomial of order 1 thanks to the function *polyfit* already

1 All the numerical plots have been made using the software *Matlab*

STRUCTURED	ERROR L_2	UNSTRUCTURED	ERROR L_2
$Str3D_9$	$3.75 \cdot 10^{-2}$	$Unstr3D_{020}$	$2.10 \cdot 10^{-2}$
$Str3D_{10}$	$3.16 \cdot 10^{-2}$	$Unstr3D_{015}$	$1.19 \cdot 10^{-2}$
$Str3D_{15}$	$1.51 \cdot 10^{-2}$	$Unstr3D_{010}$	$6.31 \cdot 10^{-3}$
$Str3D_{17}$	$1.19 \cdot 10^{-2}$	$Unstr3D_{009}$	$4.57 \cdot 10^{-3}$
$Str3D_{20}$	$8.46 \cdot 10^{-3}$	$Unstr3D_{008}$	$3.62 \cdot 10^{-3}$
$Str3D_{40}$	$2.43 \cdot 10^{-3}$	$Unstr3D_{005}$	$1.72 \cdot 10^{-3}$

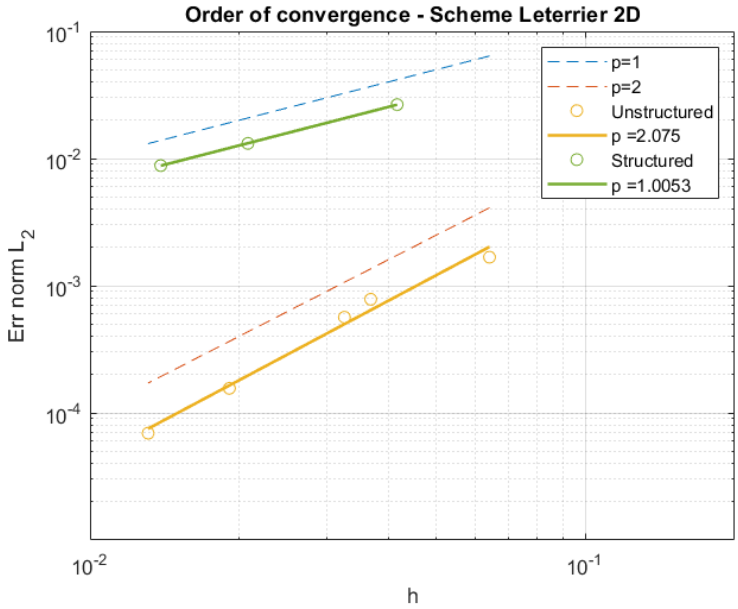
Table 5: 3D structured and unstructured errors using Leterrier's scheme, computed in norm L_2.

implemented in Matlab. By doing this, it was possible to derive the order of convergence of the scheme, which corresponds to the slope of the line. In fact, as mentioned in the equation, the order of convergence p is defined as $\log(\varepsilon)/\log(h)$. This also explains why it was preferred to use logarithmic axes in the graph instead of a graph with standard axes.

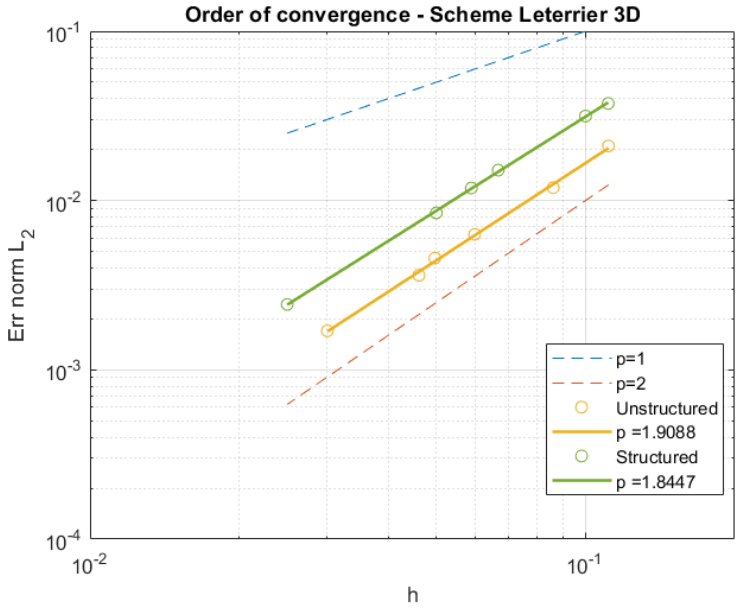
Furthermore, in order to facilitate the interpretation of the graph and immediately deduce the approximate order of convergence, the two lines corresponding to order 1 and order 2 have been drawn with a dotted line. Looking at the graphs, it is evident that for the 3D case (Figure 14b) there is no issue, as the order of convergence of the structured and unstructured case is very close to order 2 (we note that the slope is practically the same). In particular we obtained in the unstructured case $p = 1.9088$ and for the structured one $p = 1.8447$. Thus, it can be affirmed that in 3D, the implemented Leterrier scheme makes it possible to obtain a solution that is not only consistent, but also of order 2. This result served to confirm the value already found by Leterrier in his thesis work, and thus remains valid in the DEEP-DIVE implementation.

On the other hand, regarding the implementation of the Leterrier's method in 2D, although the scheme applied to unstructured meshes provides an expected order of convergence, this is not the case for structured meshes. As is evident from the graph, the yellow line has the same slope as the red dotted line, which confirms that an order of convergence $p = 2.075$ has been found. However, the green line, corresponding to the structured mesh, has a slope $p = 1.005$. This result deviates from the expectations and cannot be considered correct, as the convergence order is significantly lower than order 2. Furthermore, as we are dealing with the structured mesh case, convergence is expected to be at least of the same order as the unstructured case, which is generally the case with the most convergence problems. Although attempts were made to understand the source of the anomaly, it was not possible to solve the problem due to the limited time available. In the future, it will be necessary to thoroughly analyse the code in order to understand what the problem is that does not allow retrieving the correct order of convergence.

Wanting to test Leterrier's method also on more particular meshes characterised by heterogeneity, a hybrid mesh was created consisting of 171 elements and two parts: a structured



(a) 2D Structured and Unstructured mesh convergence.



(b) 3D Structured and Unstructured mesh convergence.

Figure 14: Order of convergence for the Leterrier's scheme implemented within DEEP-DIVE.

half with regular quadrilaterals and a non-structured half composed of triangles. The result showed that the implemented scheme is capable of providing a physically acceptable and accurate result, with an error in L2 norm of $\varepsilon_{\mathrm{HYB}} = 7.98 \cdot 10^{-2}$. The result is shown in Figure 15.

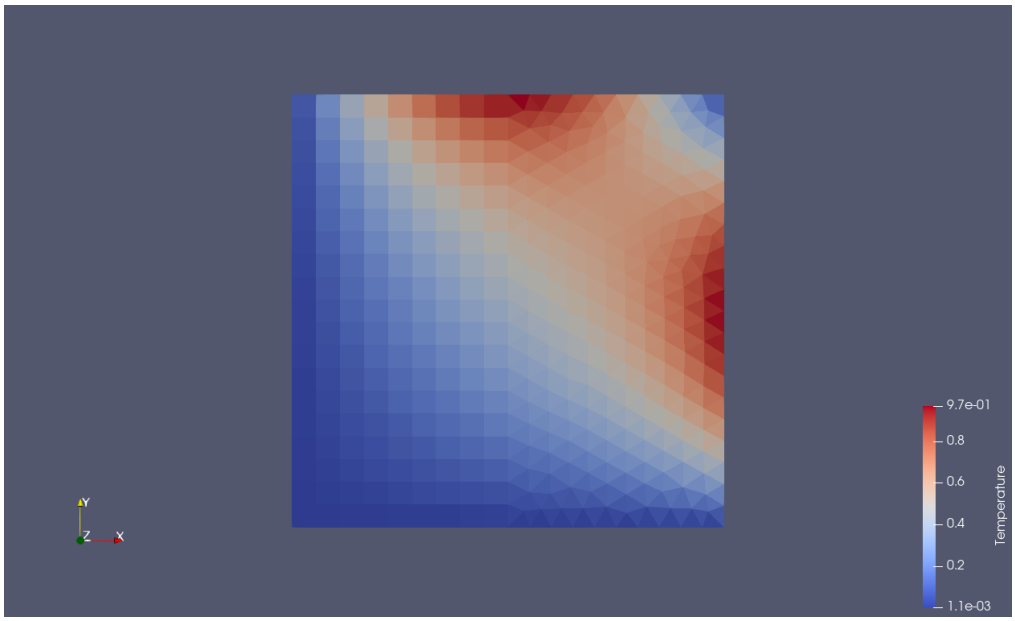


Figure 15: 2D Hybrid mesh solution

3.5 JACQ'S SCHEME IMPLEMENTATION

The numerical implementation of Jacq's scheme was the central part of the internship work. Due to the complexity of the model, a thorough study of the thesis was necessary in order to understand the basic principle of the method and the mathematical formulation that allowed the scheme to be developed.

The first phase of the work, already described in Section 2.4, gave a clear idea of the scheme as a whole and gradually more and more in detail. The second step was to adapt the structures already present in DEEP-DIVE in accordance with Jacq's proposed discretisation of the domain. In other words, we have identified three fundamental matrices useful to construct the discretisation in sub-cells and sub-faces from the information obtained from the mesh file.

Once in possession of all the geometry information, we moved on to the definition of the fundamental matrices introduced by Jacq, namely the $\mathbb{N}$, $\mathbb{S}$ and $\mathbb{S}^t$ matrix. From these matrices, defined for each of the mesh points, it was possible to derive the $\mathbb{G}$ matrix representing conductivity (see Equation 39). The $\mathbb{G}$ matrix is defined for each point and thus represents the conductivity tensor related to each sub-cell impinging at that point.

In addition, following Jacq's discussion, the vector $\mathbb{B}$ defining the boundary conditions was also constructed.

Finally, by assembling all the matrices, at each iteration the diffusive flux through each cell was calculated using the Equation 43.

It is necessary to specify that a precise choice was made in the implementation of the method that deviates slightly from that of Jacq's thesis [11]. In fact, in DEEP-DIVE, the algorithm for

calculating the solution at each time step is based on the calculation of an increment, i.e. the amount by which the solution at the centre of the cell is to be increased based on the sum of the flows entering and leaving the cell.

In order to preserve the existing algorithm, it was decided to calculate as well the increment in the case of Jacq's scheme, which was subsequently used during the time integration to calculate the solution at step $t + \Delta t$. This deviates slightly from the implementation strategy in Jacq's thesis, which is based on solving the linear system of Equation 44 and utilising the $\mathbb{D}$ and Σ matrices, which are not calculated in our case.

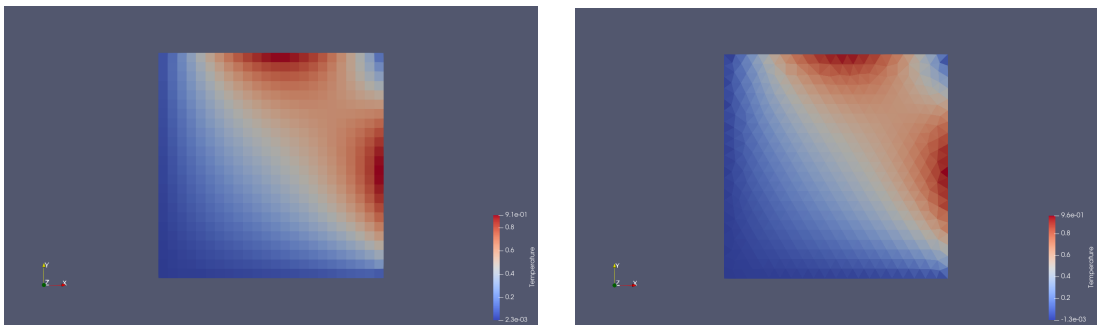
A further point to emphasise is related to the square-based pyramid elements that may be present in 3D meshes. These elements are often used in hybrid meshes, at the interface between hexahedral and tetrahedral elements. As extensively discussed in Jacq's thesis, a modification to the implemented algorithm is necessary in order to deal with these cell types. This is due to the fact that the vertex opposite to the base of the pyramid is shared by 4 faces, instead of the 3 faces common to the vertices of all other cell types. In the internship work, the presence of these elements was not taken into account and therefore the algorithm for dealing with pyramidal elements was not implemented. This implementation could be the subject of future work useful to improve the code by extruding it to more general grids.

Finally, although Jacq's scheme in 2D and 3D are very similar from an algebraic point of view, they cannot be treated equally as they present some substantial differences in geometry and connectivity. For the 2D case, reference was made to the paper of *Maire et al.* [13] on which Jacq also based his 2D and 3D formulation.

3.5.1 *Accuracy and order of convergence*

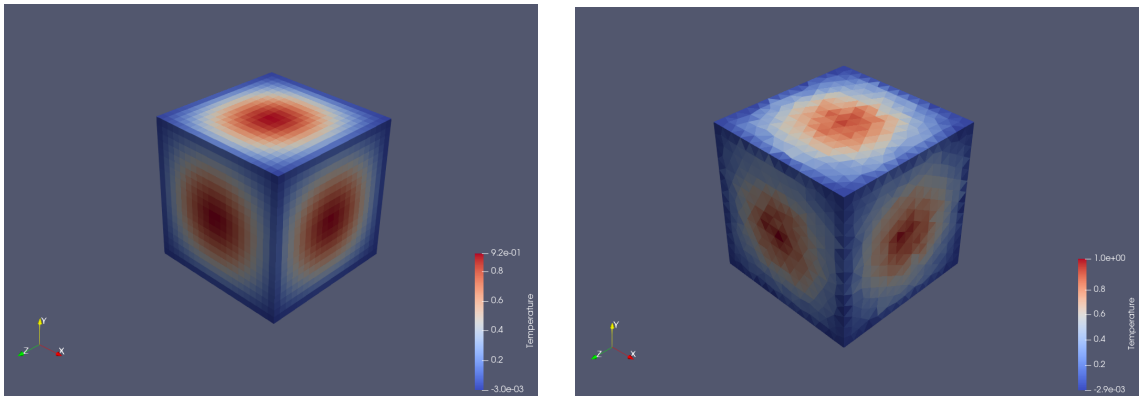
The same procedure was used for the Jacq's scheme as for the Leterrier's scheme. In addition, the same meshes in structured and unstructured were used and therefore in the following we will refer to the tables Table 2 and Table 3. First of all, the fidelity of the results to the exact analytical solution was assessed. This study was carried out both by analysing the error at convergence and by observing the graphical results obtained.

In Figure 16 and Figure 17, we have shown the numerical results obtained by using respectively 2D and 3D Jacq's scheme with structured and unstructured meshes.
The result obtained is quite satisfactory in that it allows the analytical solution of Figure 11 to be found graphically. In the 2D case (Figure 16), we can observe a very accurate solution in the structured case, while the solution for an unstructured mesh is less accurate, as it has cells in which the solution deviates from the exact value (for example, one should notice that the right upper corner solution in Figure 16b presents some inaccurate behaviours). The problem is solved consistently, but further verification of the algorithm will be necessary to understand the origin of these inaccuracies.



(a) 2D Structured mesh solution ($Str2D_{24}$). (b) 2D Unstructured mesh solution ($Unstr2D_{005}$).

Figure 16: 2D numerical solution obtained by using Jacq's scheme.



(a) 3D Structured mesh solution ($Str3D_{20}$). (b) 3D Unstructured mesh solution ($Unstr3D_{008}$).

Figure 17: 3D numerical solution obtained by using Jacq's scheme.

In 3D, although the solution in the case of the structured mesh is cleaner and smoother, a good result is also obtained for the unstructured numerical solution (Figure 17b). It is important to note that the mesh $Unstr3D_{008}$ is composed of only a few elements and a smoother solution would require finer meshes.

However, the unstructured solution, although physically acceptable, is less accurate than the solution obtained for the same mesh but using Leterrier's numerical scheme. This highlights some underlying inaccuracies that it has not yet been possible to identify, but which will be investigated in the future.

These inaccuracies are amplified when studying the order of convergence of Jacq's method. As can be seen from the graph in Figure 18, in both cases, 2D and 3D, the order of convergence found does not meet predictions and in general appears to be inaccurate. In two dimensions (see Figure 18a), the order of convergence in structured and unstructured case is less than 1 ($p = 0.9267$ for the unstructured case and $p = 0.8141$ for the structured one), resulting in a scheme only consistent. The order of convergence in 3D (see Figure 18b) found for the unstructured mesh case is $p = 1.4243$ and for the structured mesh case it is $p = 1.0686$.

These values are far from the expected value, which, according to the calculations in Jacq's thesis [11], should have been close to 2. In fact, this scheme, like Leterrier's scheme, is also theoretically of second order.

STRUCTURED	ERROR L_2	UNSTRUCTURED	ERROR L_2
$Str2D_{24}$	$2.91 \cdot 10^{-2}$	$Unstr2D_{010}$	$4.25 \cdot 10^{-2}$
$Str2D_{32}$	$2.29 \cdot 10^{-2}$	$Unstr2D_{006}$	$2.67 \cdot 10^{-2}$
$Str2D_{48}$	$1.61 \cdot 10^{-2}$	$Unstr2D_{005}$	$2.33 \cdot 10^{-2}$
$Str2D_{72}$	$1.12 \cdot 10^{-2}$	$Unstr2D_{002}$	$9.86 \cdot 10^{-3}$

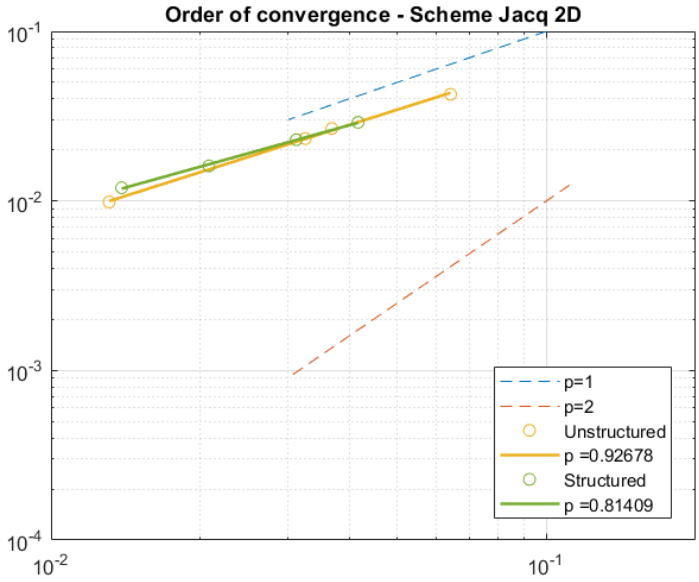
Table 6: 2D structured and unstructured errors using Jacq's scheme, computed in norm L_2.

STRUCTURED	ERROR L_2	UNSTRUCTURED	ERROR L_2
$Str3D_{9}$	$3.45 \cdot 10^{-2}$	$Unstr3D_{020}$	$9.19 \cdot 10^{-2}$
$Str3D_{10}$	$3.25 \cdot 10^{-2}$	$Unstr3D_{015}$	$4.83 \cdot 10^{-2}$
$Str3D_{15}$	$2.48 \cdot 10^{-2}$	$Unstr3D_{010}$	$3.15 \cdot 10^{-2}$
$Str3D_{17}$	$2.25 \cdot 10^{-2}$	$Unstr3D_{009}$	$2.27 \cdot 10^{-2}$
$Str3D_{20}$	$1.25 \cdot 10^{-2}$	$Unstr3D_{008}$	$2.59 \cdot 10^{-2}$
-	-	$Unstr3D_{005}$	$1.28 \cdot 10^{-2}$

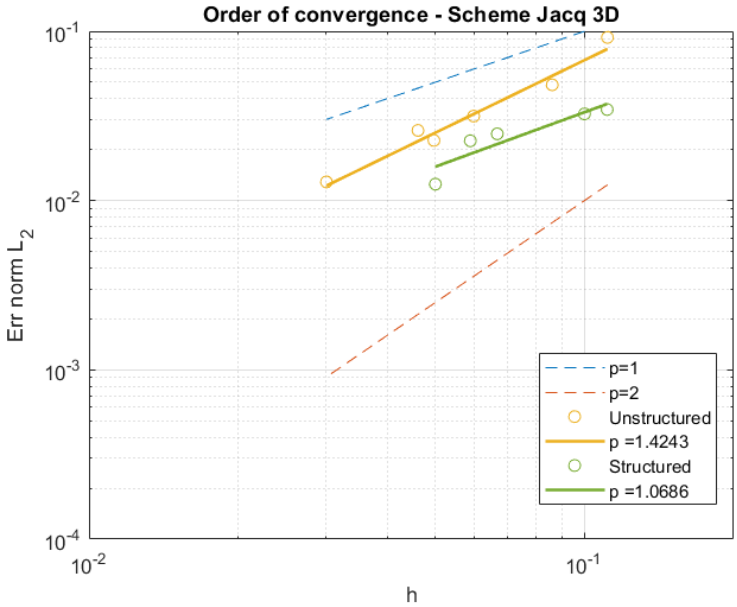
Table 7: 3D structured and unstructured errors using Jacq's scheme, computed in norm L_2.

Considering the complexity of Jacq's method, from the discretization of the mesh into sub-cells and sub-faces to the construction of the $\mathbb{G}$-matrix (see Section 2.4 for more information about Jacq's scheme), it is easy for a small inaccuracy in the code to generate unforeseen phenomena in the solution. For this reason, a very thorough analysis of the algorithm was carried out: by applying Jacq's scheme to very coarse meshes formed by a few elements, it was possible to compare the values obtained numerically with the values obtained in analytical form through hand calculations. The exact correspondence of these values allows us to state that the error may lie in the advanced stages of the code, which cannot be verified analytically by hand.

Now, the causes that lead to these inaccuracies are still being investigated and different strategies have been put in place to understand the origin of the errors. In particular, it has been thought to study the code through the Method of Manufactured Solutions described in the paper [20]. Despite several attempts with different test functions, it has not yet been possible to identify the source of the problem.



(a) 2D Structured and Unstructured mesh convergence.



(b) 3D Structured and Unstructured mesh convergence.

Figure 18: Order of convergence for the Jacq's scheme implemented on DEEP-DIVE.

Furthermore, as it is suspected that the error is due to the way in which the boundary conditions (in particular the Dirichlet conditions) are applied, we are trying to test the code using different strategies that could bypass this step, which are however still being implemented.

3.6 IMPROVEMENT STRATEGY: USE OF GRADE

In order to improve the implementation of Jacq's scheme into DEEP-DIVE and allow sources of errors to be identified, it was decided to use a test code called GRADE. This code was developed by *Guillaume Puigt* and allows the exact value of the Laplacian at the centre of the cell to be calculated for known analytical solutions. Once the Jacq scheme is implemented within the code, it is possible to compare the value of the Laplacian obtained with this diffusive scheme and the exact value of the Laplacian.

Important information on the order of convergence of the scheme can be derived from this comparison: if the numerically calculated Laplacian is equal to the analytically calculated Laplacian, this means that the implemented scheme has an order of accuracy equal to the degree of the analytical function minus 1. For example, in the case where the Laplacian of the analytical function $f(x,y) = x^3$ calculated numerically equals that calculated analytically, this guarantees that the scheme is at least second-order.

The analytical solutions considered are listed in the table. These simple analytical functions make it easy to determine the exact Laplacian by calculating the derivatives. In order to be able to study the numerical scheme in detail, Jacq's method was implemented using the same algorithm and structures designed for DEEP-DIVE. In this way, it was also possible to validate the numerical implementation and find any inaccuracies. Jacq's laplacian in the cell c corresponds to the sum over each vertex of the flux Q_{pc} defined in Equation 38.

$T(x,y)$	$\Delta T(x,y)$
x	0
y	0
$x+2y$	0
x^2	2
y^2	2
x^2+2y^2	6
x^3	$6x$
y^3	$6y$

Table 8: List of analytical functions taken into account and their analytically calculated Laplacian to be evaluated in the centre of the main cell.

The advantages of this type of improvement strategy are numerous:

1. **Easy debugging of the scheme.**
   The GRADE code is based on a very simple stencil, in that the mesh used is composed of only the main cell whose Laplacian we are interested in calculating and the

neighbouring cells that are useful for the scheme we have to use. For example, for Leterrier's scheme, adjacent cells by faces will suffice, whereas for Jacq's scheme we need adjacent cells by vertices (see Section 2.6). In addition, the mesh is defined using a very easy-to-edit ASCII format, which therefore allows the position of points in the domain to be quickly changed to obtain new meshes as required. As is evident, it is extremely easy to manually verify that the results provided by the scheme under analysis actually correspond to the expected results.

2. **Immediate identification of the order of the scheme.**
   By comparing the analytical and numerical Laplacian, the determination of the order of convergence is straightforward. Moreover, in this case, the calculation of the Laplacian is not linked to anything other than the implemented scheme. In fact, this quantity is calculated by means of a single step and does not require, for example, a time integration scheme that could impact the order of convergence.

3. **Study not depending on the implementation of boundary conditions.**
   As pointed out at the end of Section 3.5.1, after some analysis, it is thought that the main problem hindering the obtaining of the correct convergence order is related to the implementation of the Dirichlet boundary conditions. In fact, this aspect is not sufficiently detailed in Jacq's thesis and some assumptions had to be made to implement this type of boundary condition in DEEP-DIVE. Thanks to the GRADE code, it is therefore possible to study the scheme without worrying about boundary conditions, since the calculation of the Laplacian, given the solution, is carried out without taking into account the values at the edge. This simplification allows us to isolate the problem and possibly take more targeted action.

During the work carried out as part of the internship, Jacq's 2D scheme was implemented and validated in GRADE, which works for structured and unstructured meshes. Thanks to this improvement strategy, it was possible to identify some inaccuracies in the 2D algorithm and to adequately modify the scheme implemented in DEEP-DIVE. It is important to note, that once correctly implemented in GRADE, Jacq's diffusive method in two dimensions is indeed a second-order scheme. This confirms the theoretical predictions and highlights the importance of a more careful study of the implementation of boundary conditions, which could be the cause of the loss in convergence order.

4

CONCLUSIONS AND FUTURE DEVELOPMENTS

The initial objective of the internship was to find an alternative numerical method for the simulation of heat diffusion to be implemented in MoDeTheC, a code developed at ONERA. This code allows the modelling of heat and mass transfer within anisotropic porous materials, which is of particular interest in the space re-entry phase for end-of-life satellites. The new numerical method is to be compared with the numerical scheme for heat diffusion already implemented in MoDeTheC proposed by Leterrier in his PhD thesis work. The numerical scheme chosen to be implemented in MoDeTheC is the one proposed by Jacq in his PhD thesis work.

In the framework of this work, it was decided to proceed by steps in order to better assimilate the key concepts and test the new scheme on a simpler test case. Given the complexity of MoDeTheC, it would have been extremely complicated to implement Jacq's scheme directly on this code. For this reason, it was decided to implement and test the new scheme on a test code called DEEP-DIVE, developed at ONERA by Guillaume Puigt, one of the internship tutor.

Initially, DEEP-DIVE allowed solving the Navier-Stokes equations by the finite volume method, but it did not take a pure heat diffusion into account. Therefore, the first task of the stage work was to implement the algorithm for solving the heat equation in 2D and 3D, taking into account Dirichlet and Neumann boundary conditions.

After successfully implementing this algorithm, it was decided to implement in DEEP-DIVE the same numerical scheme already present in MoDeTheC, so that it could be tested and later compared with Jacq's new scheme. Thus, Leterrier's scheme was first thoroughly studied in order to understand its key concepts, and then implemented in DEEP-DIVE. Leterrier's scheme for calculating the gradient at the interface between cells replaced the centered scheme for calculating the gradient, which does not take into account the characteristics of the mesh and therefore is not suitable for some applications. The numerical results obtained were satisfactory in that the scheme was found to be not only consistent and accurate, but more importantly, it was possible to verify that this scheme was of order 2, as already stated by the author of the thesis. The order of convergence p=2 was found in 3D for the structured and unstructured configurations and in 2D for the unstructured configuration. The structured configuration in 2D did not provide the expected results and a study is underway to understand the causes of these inaccuracies.

The next step was to carefully study Jacq's thesis in order to analyze the scheme and be able to understand its basic principles. Next, meticulous thought was given to the best strat-

egy for constructing the algorithm to be implemented in DEEP-DIVE. This phase was very delicate and important in order to have an overall view of Jacq's method and to be able to proceed in the most correct way.

The scheme was then implemented in DEEP-DIVE, debugged and validated. This phase took a long time as many factors had to be taken into account and some criticality had to be resolved for the implementation of Jacq's method.

Although the Jacq scheme implemented in DEEP-DIVE provides a physically correct solution corresponding to the desired analytical solution, it should be pointed out that the implemented algorithm still has inaccuracies that affect the convergence order of the scheme. In fact, the expected order of convergence could not be obtained despite numerous attempts to find the source of the error, and further studies will be necessary to obtain an optimal solution.

Finally, thanks to the improvement strategy conducted using the GRADE code, it was possible to carry out an even more accurate analysis of the 2D Jacq code, which allowed us to identify some inaccuracies and improve the final result.

4.1 PERSPECTIVES

As evident from the results, although the implemented Jacq method provides correct physical results, the algorithm still needs to be improved in order to resolve certain inaccuracies that lead to errors on the order of convergence. In future work, therefore, a particular focus will be on the accurate analysis and debugging of the Jacq scheme, using different strategies already mentioned in the chapter. Particular attention will be paid to the implementation of boundary conditions that are likely to be a source of error. In order to study Jacq's 3D scheme more accurately, it will be necessary in the future to implement the algorithm in the GRADE code in order to carry out the same analysis as for the 2D version of the scheme.

After having correctly validated Jacq's scheme in DEEP-DIVE, its implementation in MoDeTheC will be quite straightforward. In particular, it will be necessary to re-organise the algorithm in such a way that it represents a stand-alone function with well-defined inputs and outputs. Specifically, the input will be information regarding the geometry of the mesh and the output will be the increment of the unknown variable at each time step.

Part III

MODETHEC

In this section, we will describe the results obtained from simulations carried out using *ARES* (FAST + MoDeTheC), a software that allows the calculation of aerothermodynamics during the re-entry into the atmosphere and the effects of high heat fluxes on the body. The study was carried out on a 3D body composed of a sphere and a cone and subjected to a hypersonic flow. Particular attention was given to the comparison between the isotropic and anisotropic cases.

5

ANISOTROPIC MATERIAL SIMULATION

The objective of the work described in this section is to study the aerothermodynamics to which a body is subjected during the re-entry phase into the atmosphere, with particular emphasis on the ablation of the protective shell.

In particular, the main focus was on studying the differences between the effects that heat flow in the hypersonic regime has on an isotropic and an anisotropic material. To do this, two different cases were tested:

1. A body composed of an isotropic material with a conductivity tensor characterised by equal values of conductivity in the three directions;
2. A body composed of an anisotropic material with a conductivity tensor that had a tenfold lower value along the direction normal to the wall.

To do this, a software developed at ONERA called ARES was used, which allows the two calculation codes FAST and MoDeTheC to be coupled.

5.1 ARES

ARES (Atmospheric Re-Entry Software) is a software developed at ONERA since 2006. Thanks to this tool, it is possible to model the re-entry phase both in terrestrial and martian atmosphere, with a vehicle-oriented approach. ARES joins four independent codes:

- **AtMoS** (Atmosphere Model Software) gives the atmospheric data (pressure, temperature, density, ...) as a function of the planet (Earth or Mars).
- **FAST** (Fast Aerothermodynamic Solver for Trans-atmospheric vehicle) allows to compute convecto-diffusive and 3D radiative heat flux at the wall. Furthermore, it is possible to compute also the wall distributions of pressure and friction coefficients in hypersonic regime. These wall distributions are used to determine aerodynamics force and momentum coefficients.
- **MUSIC** (MUlti SImulators in Combination) is a code used for determining the trajectory during the re-entry phase.
- **MoDeTheC** (*Modélisation de la Dégradation Thermique des Composites* which is the code presented previously.

Thanks to this software, different types of calculations can be performed, as the calculation of a flight point, numerical replication of a wind chamber test for imposed conditions, determination of a ballistic trajectory and calculation of flow characteristics at determined flight points.

5.2 TEST CASE: CAMPHOR SPHERE-CONE BODY

As a case study for our analysis, we analysed the wind-tunnel test of a body consisting schematically of a sphere and a cone, as showed in Figure 19. Initially, a body composed of isotropic material (camphor) was considered and therefore the calculations in Perron's thesis [17] were carried out. In this first case in which the shell material is isotropic, we can compare the results obtained numerically with the results found experimentally by Baker [2]. Subsequently, by changing the physical properties of the material, in particular the conductivity tensor $\mathbb{K}$, the case of an anisotropic material was studied for the same geometry and under the same physical conditions of re-entry. This study made it possible to compare the two cases (isotropic and anisotropic material) and to analyse what influence the change in material properties has on the physics of the problem.

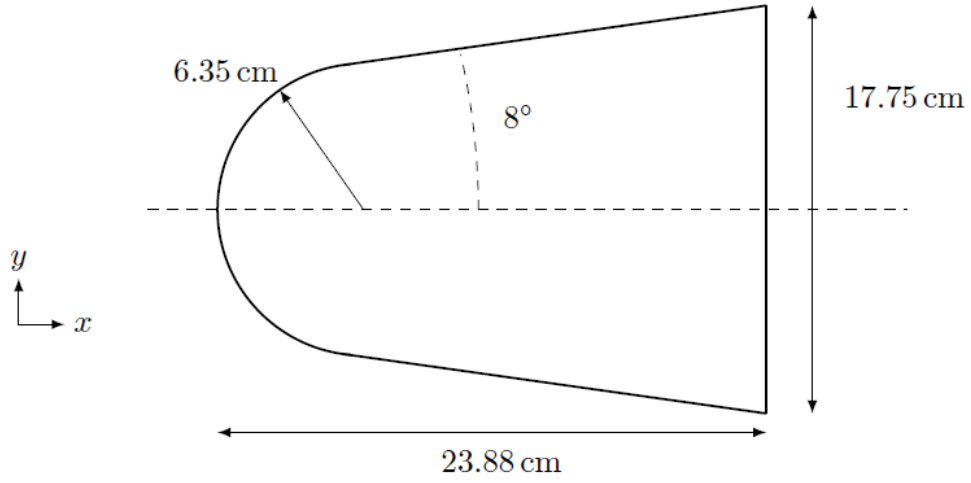


Figure 19: Geometry of the sphere-cone body used for the Baker's experience [2]. Image taken from [17].

5.2.1 *Ablative material testing and Baker's experience*

Ablative material testing on the ground seeks to provide essential information on the behavior of the material under relevant conditions for the mission. This is typically done in plasma wind tunnel facilities (such as ICP torches and arcjets), although the flow velocity is far from being in the hypersonic region typical of the actual flight. As a result, there are significant technological difficulties that must be overcome in order to conduct ablative

tests with meaningful shape change in a hypersonic facility. In fact, although the temperatures reached in the test chamber can reach the same orders of magnitude as the hypersonic regime at atmospheric re-entry, these tests do not take into account the real changes made to the structure that influence the flow around the body. The change in geometry causes a change in pressure point, a change in curvature and consequently in heat flow.

Low-temperature ablators can be utilized to get around the restriction brought on by the decreased overall temperature of hypersonic wind tunnels. In reality, there are a few commonly accessible chemicals that sublimate or ablate at temperatures low enough to allow for equipment and observations and in a way that can be correctly described by theory. The three that stand out the most are naphthalene, camphor, and dry ice. They are simple to handle and some previous experiences [2, 5] seem to make them perfect for usage in continuous wind tunnels at tolerable supersonic Mach values.

In particular, the experience to which we will refer to compare the data obtained from numerical simulations is the one conducted by Baker in 1972 on the sublimation of a body consisting of a sphere-cone and composed of camphor, which is a natural element with the chemical formula $C_{10}H_{16}O$. The main characteristics of the camphor are given in Table 9.

DENSITY	ρ	990 kg/m^3
SPECIFIC HEAT	C_P	1863 $J/(kg \cdot K)$
THERMAL CONDUCTIVITY	λ	0.4 $W/(m \cdot K)$
SUBLIMATION ENTHALPY	H_S	341 kJ/kg

Table 9: Camphor's properties at the solid state.

Baker's paper describes the experience conducted using a camphor model exposed to a flow at Mach 5 in a wind tunnel. The upstream conditions during the experiment are listed in Table 10. Furthermore, Baker compares the results obtained with numerical results ob-

SPEED	v_∞	1155 m/s
PRESSURE	p_∞	1515 Pa
TEMPERATURE	T_∞	132 K
DENSITY	ρ_∞	0.04 kg/m^3

Table 10: Upstream flow conditions during the ablation experience of the cone-sphere camphor body.

tained using analytical models to calculate the heat flow at the wall. The pressure at the wall is calculated using the modified Newton method. The heat flow at the stopping point is then obtained using Cohen's analytical model for laminar flows. The distribution of the parietal heat flow is calculated from these quantities and the Lees local similarity approach

in the zone away from the stopping point. Finally, the ablation mass flow rate is calculated with the Knudsen-Langmuir relation.

5.2.2 *Description of simulations using ARES*

For ARES simulations, the domain outside the body and the body itself must be considered separately. Externally, a mesh is not required, as the models used are not solved by finite volume or finite difference type methods. On the contrary, the mesh of the material domain used consists of 146000 elements and 4000 boundary faces (see Figure 20). An algorithm is also implemented in MoDeTheC to adapt the mesh at each iteration, so that the ablation of the body shell is taken into account by the discretization. Finally, ARES exploits a coupling method that allows a heat flux/temperature exchange at the wall to bind the two domains.

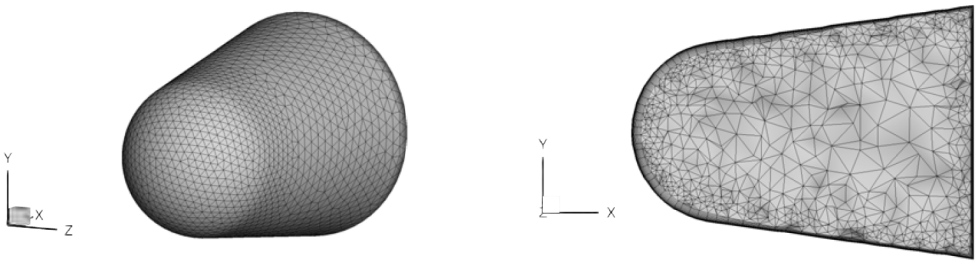


Figure 20: 3D hybrid mesh of the sphere-cone body in camphor for the ARES simulation. On the left the boundary surface and on the right a cut view of the domain. Image taken from [17].

As far as the definition of material properties is concerned, a diagonal conductivity matrix with $\mathbb{K}_{ii} = 0.4\ W/(m \cdot K)$ (with $i = 1, 2, 3$), was considered in the isotropic case. On the contrary, for the anisotropic case, it was decided to change only one component of the matrix. The component chosen is the one normal to the surface while the conductivity value in the other two directions was left unchanged, thus assuming isotropicity in the tangent plane. In fact, spacecraft claddings are often composed of several layers of composite material each consisting of fibres oriented in a certain direction. Therefore, as a first approximation, it can be assumed that in the tangent plane yz the material can be considered isotropic, as it is composed of fibres oriented randomly in different directions. Along the normal, however, the conductivity changes, and in particular has been taken to be $\mathbb{K}_{11} = 0.04\ W/(m \cdot K)$ (i.e. ten times smaller than that considered in the isotropic case).

5.3 NUMERICAL RESULTS

5.3.1 *Isotropic case*

The purpose of the first simulation was to replicate Perron's study in his thesis work and to have a database to compare the isotropic and anisotropic cases. The deformation of the

geometry during the $\Delta t = 320s$ simulation with ARES is shown in Figure 22. Using geometry information, it is possible to derive the regression speed at which the body sublimates. As shown in the thesis, this speed has an initial transient phase before reaching an almost constant value. Considering the amount by which the nose tip point shifted during the duration of the simulation, i.e. $\Delta x = 43$ mm, it is possible to obtain a first rough estimation of the regression speed by considering it constant over time.

$$V_r^{(ISO)} = \frac{43mm}{320s} = 0.1344mm/s$$

In Figure 21 we show the boundary surface for different times during the simulation. As is evident from the figure, the spherical shape of the nose is progressively flattened. This change implies an increase in the radius of curvature at the stagnation point and thus a decrease in the calculated convective heat flow at the nose tip. This flattening is also evident in the two images showing the surface temperature of the three-dimensional body (Figure 22) at two different instants, i.e. $t = 80s$ and $t = 240s$.

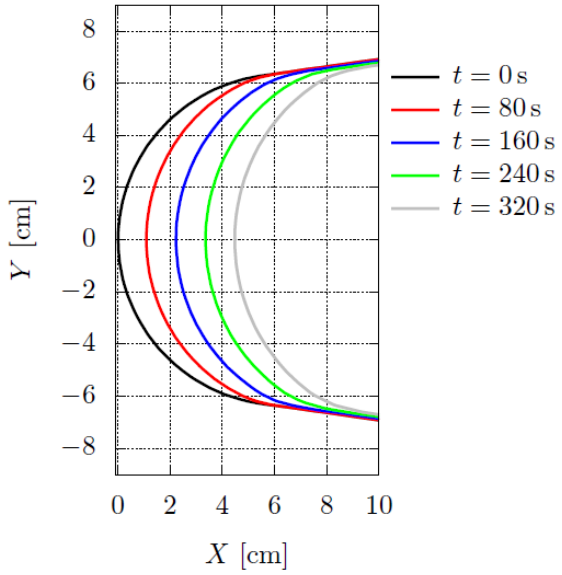
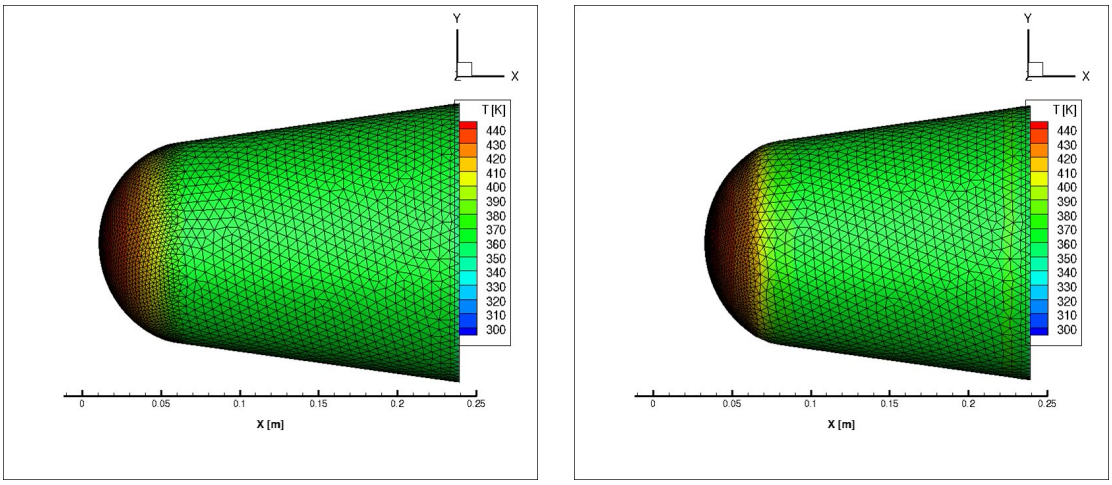


Figure 21: Displacement of the boundary surface of the camphor sphere-cone near the nose tip point for ARES simulation between 0 and 320 s. Image taken from [17].

Finally, it is possible to compare the simulation results carried out using the ARES software with the data obtained by Baker both experimentally and numerically. The comparison, shown in the figure, shows that the simulation with ARES provides excellent results for the spherical part of the geometry, while it presents some inaccuracies at the level of the joint between the sphere and the cone. This error is due to an incorrect prediction of the heat fluxes in this area away from the stagnation point by the models used in FAST.



(a) 3D ISO solution for t = 80s

(b) 3D ISO solution for t = 240s

Figure 22: 3D isotropic solution for the sphere-cone body plotted using TecPlot360. The different colours represent the surface temperature of the body.

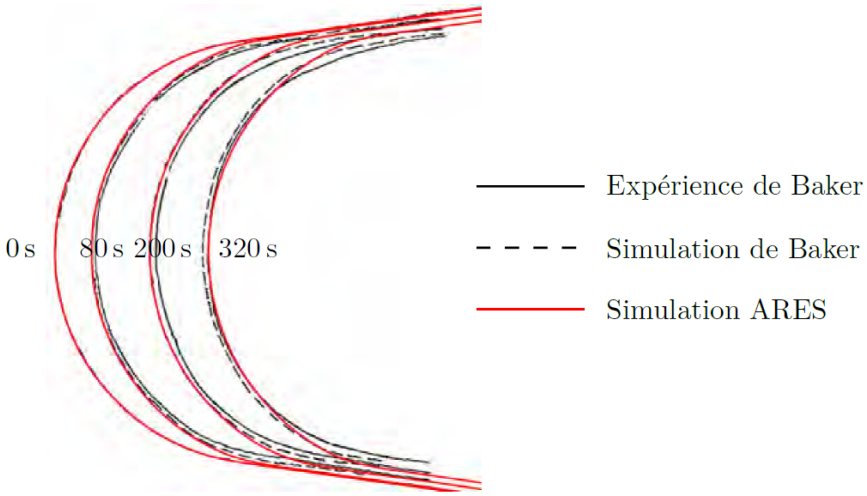


Figure 23: Comparison of the surface displacement of the camphor sphere-cone at different times obtained by the Baker experiment, the Baker simulation and the ARES simulation. Image taken from [17].

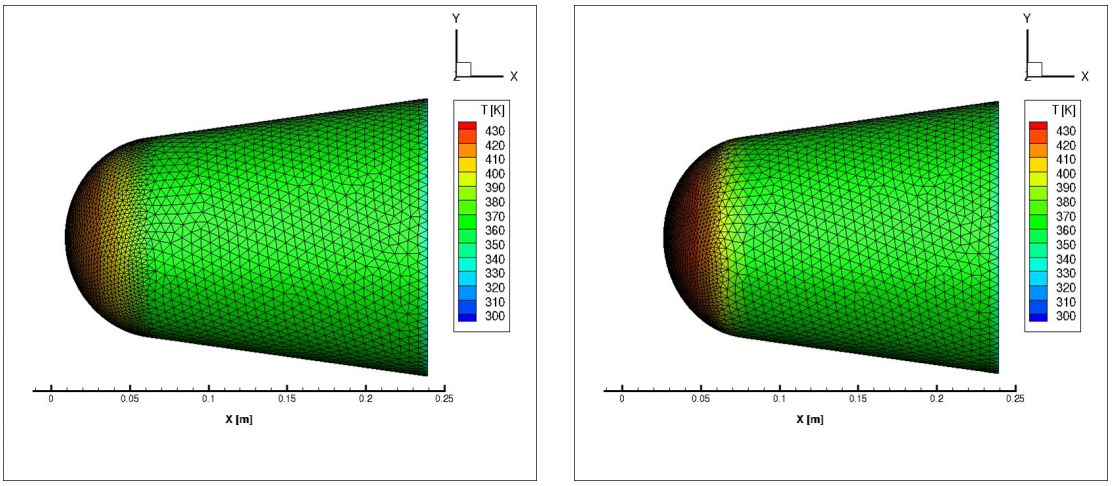
5.3.2 *Anisotropic case*

In contrast to the isotropic case, in the anisotropic case no experimental tests are available to validate the numerical code employed. For this reason, the comparison will be conducted by means of theoretical and physics-related considerations of the problem. First, we measure how far the stagnation point has shifted in order to calculate the average regression rate in the anisotropic case. We can see that the nose tip point has shifted by a smaller amount than

in the isotropic case. In particular, $\Delta x = 33$ mm. This means that on average, the regression speed has decreased.

$$V_r^{(ANI)} = \frac{33mm}{320s} = 0.1031 mm/s$$

In Figure 24, we can see the solution obtained in the anisotropic case at the same t as the isotropic solution shown in Figure 22.



(a) 3D ANI solution for t = 80s

(b) 3D ANI solution for t = 240s

Figure 24: 3D anisotropic solution for the sphere-cone body plotted using TecPlot360. The different colours represent the surface temperature of the body.

5.4 COMPARISON BETWEEN ISOTROPIC AND ANISOTROPIC CASE

In this section, we will try to explain from a physical point of view why the regression speed decreases when the normal component of diffusivity decreases. Let us first write the balance of the main contributions to the heat flow at the body surface basing on Figure 25 [3]. If we

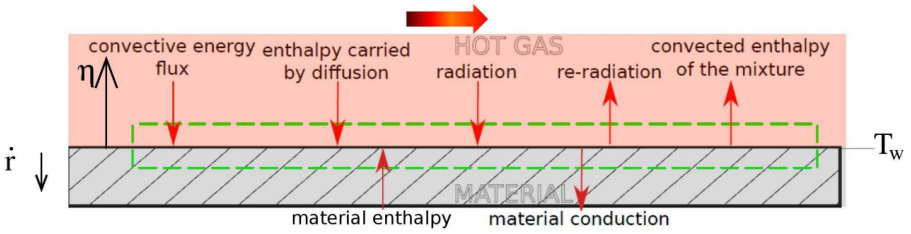


Figure 25: Surface energy balance for an ablating material. Image taken from [3]

consider the contributions due to convection, conduction within the material, ablation and radiation, we obtain

$$q_{ext} = q_{abl} + q_{c_w} + q_{rad} \tag{50}$$

where the left-hand side term is the heat flux due to the aerodynamics around the body (mainly convective heating), the first right-hand side term q_{abl} represents the energy flux due to ablation, q_{c_w} is the solid conduction and finally q_{rad} is the radiative flux from the wall.

- The ablative heat can be defined as a function of the enthaply drop. So we can write $q_{abl} = \dot{m}_w(\Delta h_{abl} + \Delta h_{solid})$ with Δh_{reac} the enthalpy required for ablation to occur and $\Delta h_{solid} = h(T_w) - h(T_i)$ the enthalpy required to heat the solid to the wall ablation temperature;
- The solid conduction can be defined as $q_{c_w} = -k\frac{\partial T}{\partial \eta}$ with k representing the conductivity of the solid material. Considering the discrete form of the conductive flux, one can write $q_{c_w} = -k_n\frac{T_w - T_0}{\Delta x}$ with T_w the wall temperature and T_0 the temperature inside the body and k_n the approximation of the conductivity;
- Assuming the solid material as a grey body with surface emissivity equal to ε, we can define $q_{rad_{out}} = \sigma\varepsilon(T_w^4 - T_\infty^4)$ with T_∞ the temperature of the hot gas and σ the Stefan-Boltzmann constant.

If we substitute in Equation 51 all the terms, we obtain

$$q_{ext} - \sigma\varepsilon(T_w^4 - T_\infty^4) + k\frac{T_w - T_0}{\Delta x} = \dot{m}_w \Delta h_{reac} \tag{51}$$

Considering the density constant, the ablation rate will be given by the sum of the ablative heat flux and the conductive flux divided by the enthalpy jump.

$$V_r = \frac{q_{abl}}{\Delta h_{reac} + h(T_w) - h(T_i)} \cdot \frac{1}{\rho} \tag{52}$$

Let us examine each term using the data obtained from the simulation and the influence this has on the regression speed.

The first observation that can be made when analysing the results is strictly related to the conductivity of the material, which is the physical characteristic that differentiates the two cases, isotropic and anisotropic. Thermal conductivity of a material is a measure of its ability to a particular material conduct heat. Heat transfer occurs at a lower rate in materials of low thermal conductivity than in materials of high thermal conductivity. In the anisotropic case, as conductivity decreases, heat finds it harder to penetrate the interior of the body and consequently the thickness affected by the temperature gradient is smaller than in the isotropic case (see Figure 26). In other words, the ability of the heat flux to penetrate the body (in the normal direction) is hindered by the low conductivity of the material. This phenomenon influences the temperature reached inside the body. Indeed, in the anisotropic case, it can be seen that the internal temperature is lower than T_0 in the isotropic case. In the isotropic case, the heating is much more effective and the inside of the body is heated in less time.

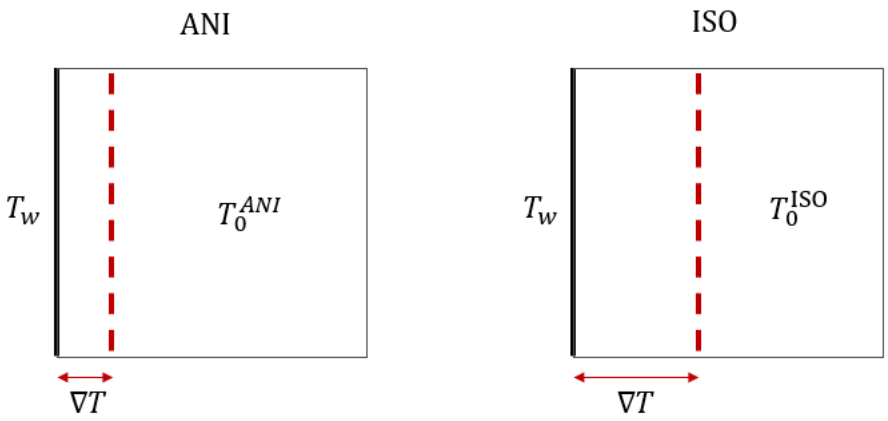


Figure 26: Schematic drawing of what happens inside the body for the two cases anisotropic and isotropic.

Comparing the characteristic times of the phenomena involved, we can state that in the isotropic case the characteristic time for heat conduction in the body is smaller than the characteristic time for ablation. Thus, the body inside is heated as the erosion progresses. In the anisotropic case, on the contrary, the characteristic time for heat flow inside the body is comparable or smaller than the ablation time. This causes rougher erosion. The phenomenon just described manifests itself in the simulation results. We can see that during the first moments of the simulation, ablation starts slightly earlier in the anisotropic case. Although it is activated more quickly, the regression rate is lower in the anisotropic case. Thus, although the ablation process starts earlier for an anisotropic material, after sufficient time (in our case, $t = 320s$) the isotropic material is more eroded than the anisotropic one.

Moreover, in the two cases, the ablation temperature T_w does not change cause we have the same material and the ablation temperature depends on the chemical properties of it. Thus, the temperature gradient is higher in the anisotropic case than in the isotropic case, since the internal temperature is lower ($T_0^{(ANI)} < T_0^{(ISO)}$). However, this increase in the temperature gradient is not high enough to compensate for the decrease in the thermal conductivity of the material in the normal direction. This explains why the heat flow by conduction inside the body is lower in the anisotropic case, as can be seen in the simulation.

The decrease in internal temperature in the anisotropic case leads to a decrease in the body's internal enthalpy. From the data available from the simulation, we have that the heat flow due to ablation is smaller in the anisotropic case. Therefore, a decrease in the regression rate is well justified by this trend in magnitudes (according to Equation 52)

Finally, as already pointed out above, during the ablation process, due to the change in the surface of the body, we consequently have a change in the radius of curvature R_n. This parameter strongly impacts the convective flux q_{ext}, which varies inversely to the radius of curvature. In particular, as is evident from the results, in the anisotropic case ablation causes a smaller decrease in the radius of curvature than in the isotropic case. Thus, for a fixed

instant of time, we will have that $R_n^{(ANI)} > R_n^{(ISO)}$. From this it follows that the external heat flux will be smaller in the anisotropic case than in the isotropic one. In Table 11 we summarise the trends of the different physical quantities by switching from the isotropic to the anisotropic case.

	ISO	ANI
q_{rad}	=	=
q_{abl}	+	-
Δ_h	-	+
V_r	+	-
T_i	+	-
q_{c_w}	+	-
R_n	-	+
q_{ext}	+	-

Table 11: Summary of the increases and decreases in the quantities under examination by switching from the isotropic to the anisotropic case.

6

CONCLUSIONS AND FUTURE DEVELOPMENTS

The second part of the project focused on a practical application of an atmospheric re-entry body. Thanks to the ARES software, which integrates the MoDeTheC calculation code developed at ONERA, it was possible to simulate a sphero-conical body subject to a hypersonic flow and to study the effects of high heat flows on it. In particular, the most evident phenomenon is the ablation by sublimation of the body structure. Given the importance of anisotropic composite materials in the aerospace industry, some simulations were carried out by changing the value of the thermal conductivity of the material of which the body is made. The simulations revealed certain differences between the anisotropic case and the isotropic case, which became evident in particular by a decrease in regression speed as the thermal conductivity decreases.

6.1 PERSPECTIVES

The increasing use of anisotropic materials in the aerospace industry makes it necessary to carry out an in-depth study of the phenomenology associated with heat diffusion within this type of material. For this reason, it would be extremely interesting to obtain experimental data that can validate the numerical results obtained through the ARES code and in particular through MoDeTheC. The aim for the future is to carry out further simulations, changing the physical characteristics of the material, in order to study the behaviour of anisotropic materials subjected to large heat flows in even greater detail. Furthermore, as explained in the second part of the report, currently the numerical scheme implemented in MoDeTheC is that of Leterrier, detailed above. The aim of future work will be to improve this code by implementing new diffusive schemes, such as the Jacq scheme, and study the improvements made.

Part IV

APPENDIX

A

JACQ'S SCHEME: MATRICES DEFINITION

- **Matrix $\mathbb{N}$ and $\mathbb{S}$**
  Below is the definition of the constitutive matrices of Jacq's method $\mathbb{N}$ and $\mathbb{S}$, which depend on the geometry and physical characteristics of the material (matrix $\mathbb{K}$). The 2D and 3D cases are similar but cannot be defined using the same notation.

  2D SCHEME

$$\begin{cases} \mathbb{N}_{c,c-1} = \alpha_{c-1} l_{c-1} l_c (\mathbb{K}_{c-1} n_{c-1}^{c-1} \cdot n_c^{c-1}) \\ \mathbb{N}_{c,c} = \alpha_{c-1} l_c^2 (\mathbb{K}_{c-1} n_c^{c-1} \cdot n_c^{c-1}) + \alpha_c l_c^2 (\mathbb{K}_c n_c^c \cdot n_c^c) \\ \mathbb{N}_{c,c+1} = \alpha_c l_c l_{c+1} (\mathbb{K}_c n_{c+1}^c \cdot n_c^c) \end{cases}$$

$$\begin{cases} \mathbb{S}_{c,c-1} = \alpha_{c-1} l_c [l_{c-1} (\mathbb{K}_{c-1} n_{c-1}^{c-1} \cdot n_c^{c-1}) + l_c (\mathbb{K}_{c-1} n_c^{c-1} \cdot n_c^{c-1})] \\ \mathbb{S}_{c,c} = \alpha_c l_c [l_c (\mathbb{K}_c n_c^c \cdot n_c^c) + l_{c+1} (\mathbb{K}_c n_{c+1}^c \cdot n_c^c)] \end{cases}$$

  3D SCHEME

$$\begin{cases} \mathbb{N}_{ff} = \alpha_c A_c^i \mathbb{K}_{ii}^c A_c^i + \alpha_d A_d^j \mathbb{K}_{jj}^d A_d^j, \\ \mathbb{N}_{fg} = \alpha_c A_c^i \mathbb{K}_{ik}^c A_c^k \quad \text{for} \quad k \in [1,3] \quad \text{and} \quad k \neq i, \\ \mathbb{N}_{fg} = \alpha_d A_d^j \mathbb{K}_{jk}^d A_d^k \quad \text{for} \quad k \in [1,3] \quad \text{and} \quad k \neq j. \end{cases}$$

$$\begin{cases} \mathbb{S}_{fc} = \alpha_c \sum_{k=1}^{3} A_c^i \mathbb{K}_{ik}^c A_c^k, \\ \mathbb{S}_{fd} = \alpha_d \sum_{k=1}^{3} A_d^j \mathbb{K}_{jk}^c A_d^k. \end{cases}$$

- **Boundary conditions: matrix $\mathbb{N}$ and $\mathbb{S}$**
  We also show the changes to be made to the matrices when dealing with cells and boundary faces. These modifications depend on the type of boundary condition to be imposed.

  3D SCHEME

$$\text{DIRICHLET:} \begin{cases} \mathbb{N}_{ff} = A_c^i, \\ \mathbb{N}_{fg} = 0, \\ \mathbb{S}_{fg} = 0. \end{cases}$$

$$\text{NEUMANN:} \begin{cases} \mathbb{N}_{ff} = \alpha_c A_c^i \mathbb{K}_{ii}^c A_c^i, \\ \mathbb{N}_{fg} = \alpha_c A_c^i \mathbb{K}_{ik}^c A_c^k, \\ \mathbb{S}_{fc} = \alpha_c \sum_{k=1}^{3} A_c^i \mathbb{K}_{ik}^c A_c^k. \end{cases}$$

$$\text{ROBIN:} \begin{cases} \mathbb{N}_{ff} = \beta\alpha_c A_c^i \mathbb{K}_{ii}^c A_c^i - \alpha A_c^i, \\ \mathbb{N}_{fg} = \beta\alpha_c A_c^i \mathbb{K}_{ik}^c A_c^k, \\ \mathbb{S}_{fc} = \beta\alpha_c \sum_{k=1}^{3} A_c^i \mathbb{K}_{ik}^c A_c^k. \end{cases}$$

BIBLIOGRAPHY

[1] John D. Jr. Anderson. *Hypersonic and High-Temperature Gas Dynamics*. 2nd. AIAA, 2006.

[2] L. R. Baker. "Low temperature ablator nosetip shape change at angle of attack." In: *10th Aerospace Sciences Meeting* (1972).

[3] Turchi A. Bianchi D. "Numerical analysis on the sublimation of low-temperature ablator models undergoing shape change in a supersonic wind-tunnel." In: *EUCASS association* (2019).

[4] V. Biasi. *Modélisation Thermique de la Dégradation d'un Matériau Composite Soumis au Feu.* Toulouse, France: Université de Toulouse - Isae-SUPAERO, 2014.

[5] F. Charwat. *Exploratory studies on the sublimation of slender camphor and naphthalene models in a supersonic wind-tunnel.* Advanced Research Project Agency, Memorandum RM-5506-ARPA. 1968.

[6] B. Despres. "Lax Theorem and Finite Volume Schemes." In: *Mathematics of Computation* (2003).

[7] Rider W. Drikakis D. *High-Resolution Methods for Incompressible and Low-Speed Flows*. 1rd. 2005.

[8] Von Karman Institute For Fluid Dynamics. "MUTATION++ library, technology transfer from atmospheric entry plasmas to biomass pyrolysis." In: (2018).

[9] ESA. "End-of-life disposal of satellites." In: (2015).

[10] Remacle J.-F. Geuzaine C. *GMSH Reference Manual*. 2022.

[11] P. Jacq. *Méthodes numériques de type Volume Finis sur maillages non structurés pour la résolution de thermique anisotrope et des équations de Navier-Stokes compressibles.* Bordeaux, France: Université de Bordeaux, 2015.

[12] N. Leterrier. *Discrétisation spatiale en maillage non-structuré de type général*. Paris, France: Université Pierre et Marie Curie, 2003.

[13] Breil J. Maire P.-H. "A high-order finite volume cell-centered scheme for anisotropic diffusion on two-dimensional unstructured grids." In: *Journal of Computational Physics* (2011).

[14] ONERA. "Reserved document property of ONERA, Departments DAAA/DMPE." In: (October 2020).

[15] Van Altena M. Ollivier-Gooch C. "A High-Order Accurate Unstructured Mesh Finite-Volume Scheme for the Advection-Diffusion Equation." In: *University of British Columbia.* (2002).

[16] Navarro Laboulais J.-Leyland P. Mischler S. Park S.-H. "Re-entry survival analysis and ground risk assessment of space debris considering by-products generation." In: *Acta Astronautica* (2020).

[17] N. Perron. *Modélisation de la dégradation thermo-structurale des débris spatiaux durant la rentrée atmosphérique.* Toulouse, France: Université de Toulouse - Isae-SUPAERO, 2022.

[18] NASA Technology Transfer Program. "Porous material Analysis Toolbox based on OpenFoam (PATO)." In: (2018).

[19] Auffray V.-Müller J.-D. Puigt G. "Discretisation of diffusive fluxes on hybrid grids." In: *Journal of Computational Physics* 229 (2008), pp. 1425–1447.

[20] P.-J. Roache. "Code Verification by the Method of Manufactured Solutions." In: *ASME* (2002).

[21] E.F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics*. 2nd. 1999.

[22] G. Pont et al. "Multiple-correction hybrid k-exact schemes for high-order compressible RANS-LES simulations on fully unstructured grids." In: *Journal of Computational Physics - Vol. 350, p.45-83* (2017).