UNIVERSIDAD POLITÉCNICA
DE MADRID

INSTITUTO UNIVERSITARIO DE MICROGRAVEDAD
“IGNACIO DA RIVA”

TRABAJO FIN DE MÁSTER

High-order methods for the CFD simulations of hypersonic flows over a reentry capsule

DAVID HUERGO PEREA

MÁSTER UNIVERSITARIO EN SISTEMAS ESPACIALES

JULIO 2022

Escuela Técnica Superior de Ingeniería
Aeronáutica y del Espacio

Departamento de Matemática Aplicada a la
Ingeniería Aeroespacial

HIGH-ORDER METHODS FOR THE CFD SIMULATIONS OF HYPERSONIC FLOWS OVER A REENTRY CAPSULE

---

by:

David Huergo Perea

Advisors:

Esteban Ferrer

Juan Antonio Hernández

Gonzalo Rubio

Madrid, July 2022

Contents

List of Tables

List of Figures

Abbreviated terms

BC:	Boundary Condition.
CBR:	Cosmic Background Radiation.
CFD:	Computational Fluid Dynamics.
CG:	Continuous Galerkin.
CHT:	Conjugate Heat Transfer.
DG:	Discontinuous Galerkin.
DNS:	Direct Numerical Simulation.
FEM:	Finite Elements Method.
FFTB:	Flux Forward Temperature Back.
HETC:	High Efficiency, Tantalum-based Ceramic.
hFFB:	heat transfer coefficient Forward Flux Back.
hFTB:	heat transfer coefficient Forward Temperature Back.
HORSES3D:	High-ORder (DG) Spectral Element Solver.
IC:	Integrated Circuits.
ILES:	Implicit Large Eddy Simulation.
IS:	International System.
ISA:	International Standard Atmosphere.
JB-2006:	Jacchia-Bowman 2006.
LES:	Large Eddy Simulation.
MVF:	Mean Value Filter.
NRLMSISE-00:	NRL Mass Spectrometer, Incoherent Scatter Radar Extended Model.
RANS:	Reynolds Averaged Navier-Stokes.
RMSE:	Root Mean Squared Error.
TFFB:	Temperature Forward Flux Back.
TPS:	Thermal Protection System.
TUFROC:	Toughened Uni-piece Fibrous Reinforced Oxidation-resistant Composite.

1 Introduction

From the beginning of the space race, the development of *reentry vehicles* has gained importance due to the need of recovering space payloads for different purposes. Though these vehicles were first created for military research in the field of ballistic technologies, soon the paradigm drastically changed with the first manned missions. Nowadays, even unmanned missions require this technology for scientific research, especially for interplanetary missions, which have become of great importance during these last years [14].

Unlike scientific or military missions, manned mission are the most critical; hence, the requirements to be accomplished are also the most severe. These requirements are even more specific for long-time missions, as it could be a manned spaceflight to Mars [15]. A major requirement is established to assure endurance over the extreme thermal loads during the descent flight through the atmosphere, as a combination of conductive, convective and radiative heat transfer. These environmental conditions require the use of TPS (*Thermal Protection Systems*) to assure a successful landing and the safety of payloads [4].

The analysis of this fluctuating and extreme environment, along with the specifications and properties of the reentry vehicle, leads to a complex problem of *aerothermodynamics*. In order to carry out the aerothermodynamic design and qualify the space vehicle, it is essential to simulate the *fluid-structure interaction* through different approaches [16]: testing in ground-based facilities (e.g. wind tunnels), numerical analysis (e.g. CFD (Computational Fluid Dynamics) simulation) and flight testing for qualification in a realistic environment.

All these techniques allow to obtain a robust design of reentry vehicles, which can withstand the expected loads in every phase of the mission with an optimal performance.

1.1 Scope

Among the three available options previously explained to solve the aerothermodynamic problem, the CFD simulation is cheaper than testing and is used in massive scale for design purposes in this field. Therefore, in this research several simulations will be carried out to analyse the behaviour of a reentry vehicle for different flight conditions in a hypersonic flow.

The chosen approach for this analysis is described below:

1. The flow will be computed using the CFD code HORSES3D (High-Order (DG) Spectral

Element Solver) [11], developed by the NuMath research group at ETSIAE-UPM. This simulation will allow to understand the behaviour of the hypersonic flow around the reentry vehicle (see sec. 3).

2. The TPS of the vehicle will be analysed through an open-source FEM (*Finite Elements Method*) library called MFEM [17], which will allow to solve the heat equation in the solid domain (see sec. 2).
3. Finally, the aerothermodynamic problem will be solved by coupling both solvers during the simulation, resulting in a CHT (Conjugate Heat Transfer) analysis (see sec. 4). The solution of the fluid-structure interaction will be obtained for different cases (see sec. 5), allowing to understand the thermal effect of the hypersonic flow over the reentry vehicle.

The main disadvantage of this approach is the large computational time required to compute the solution. Therefore, in order to tackle the problem progressively, several analyses will be carried out in ascending degree of complexity, from a CFD simulation on an adiabatic capsule to a full simulation of the whole conjugate heat transfer problem on a real capsule, considering a conductive and radiative TPS.

Furthermore, other techniques will be implemented with the objective of reducing the simulation time, as it could be the use of a weak coupling between the fluid and solid domains, reducing the information exchange between solvers to some key timestamps during the computation.

The established methodology for this research is based on the use of *high-order methods*. Though this approach increases the complexity and the simulation time per iteration, high-order methods also allow to reduce the refinement of the mesh, which implies a decrease in the computational time for a defined error level. The global balance is positive in favour of high-order methods and their use is spreading in this field due to their great performance [18].

Finally, this coupling approach involves to link different programming languages: Fortran, Python and C++. The information transfer between multiple programmes requires the use of interfaces and wrappers, which make possible to connect the algorithms and to compute the whole simulation in a single execution of the main code (see sec. 8). The design of appropriate interfaces helps to generalise this particular problem to a wider range of applications where the use of different solvers and programming languages is necessary.

1.2 Objectives

This research has focused on the analysis of a reentry capsule immersed in a hypersonic flow, which can be considered as the main objective that has to be fulfilled. However, the different analyses developed for this project have taken into account a wider set of objectives that are expected to be accomplished:

1. To test the ability of non-commercial software to solve complex aerothermodynamic problems, including their advantages and limitations. Although the use of open-source solvers can be a challenging task, their extraordinary potential and degree of accuracy can surpass, in some cases, that of commercial applications.
2. The coupling of different solvers, through a programming interface, to solve conjugate heat transfer problems. Different programming languages can be linked into a single simulation, allowing a greater flexibility and facilitating the use of open-source applications.
3. To test the effectiveness of high-order methods to model aerodynamic problems through a CFD simulation. High-order methods allow to obtain accurate solutions with a coarser mesh, reducing the global computational time. An increase in the order of the solution generates a greater reduction of the error than an increase in the degrees of freedom of the mesh, for an equivalent simulation time [11].
4. To test the shock capturing functionality of the CFD solver, HORSES3D, for hypersonic flows. Although this solver has been previously used to solve shock waves, this will be the first attempt to model a shock with a hypersonic Mach number.

All these objectives represent different possibilities that can be achieved through the CFD simulation of real engineering problems, which can be used as a reliable design tool that allows to obtain a high degree of accuracy with a low cost.

However, it is important to mention that, in space missions, every design must be correctly tested, qualified and validated in a real flight before its use is approved for real applications, specially in the case of manned missions.

1.3 Novelty

In order to successfully fulfil all the previous objectives, it is necessary to use different tools and to follow specific methodologies, specially for the correct functioning of the coupling process between different solvers. Therefore, the development of some novel approaches, methodologies or tools has been required, as a complement to the existing state of the art present in the scientific literature.

Furthermore, it is important to clarify the novelty of the present work, highlighting those aspects that have been created from scratch in contrast to those that have been acquired from other researches. The main novel developments in this work are briefly explained below:

1. The programming interface (see sec. 8), which is required to perform the coupling between HORSES3D and MFEM, has been created in this work, as it is the first time both solvers are linked. This interface includes the definition of boundary conditions, the synchronisation among different subroutines in both solvers, the information exchange and the interpolation at the interface among others. However, the master interface with three predefined functions has been developed by Noah Brenowitz [19].
2. The spatial filter (see sec. 4.2.2), used as a stabilisation tool for the CHT analysis, is a novel approach proposed in this work, as opposed to the traditional temporal filter.
3. The CHT analysis of reentry capsules is scarce in the scientific literature. Furthermore, there are not relevant researches regarding the full aerothermodynamic analysis of a reentry capsule with a radiative and conductive TPS, which is a novel analysis performed in this work.
4. Although the shock capturing functionality was already available in HORSES3D, in this work it has been tested to simulate hypersonic flows for the first time.
5. The different meshes used in this research have been created for each particular case. However, the meshing software Gmsh [20] is an existing open-source application.
6. The different issues and challenging problems (see sec. 9) present in this work have been detected and successfully solved, through the use of some novel and traditional methodologies.

1.4 Classes of hypersonic vehicles

The common element among reentry vehicles is the presence of a hypersonic flow around the structure during the descent phase. There are different approaches to design an efficient hypersonic vehicle, according to the particular needs of the mission. Though the design of new configurations is constant, in general it is possible to define three main classes of hypersonic vehicles [1]:

1. **Winged reentry vehicles:** Designed with a blunt shape and prepared to fly at large angles of attack to increase the effective drag and its efficiency as a deceleration system. Some of them are reusable (e.g. Space Shuttle). The presence of wings improves the control during the descent.

2. **Cruise acceleration vehicles with airbreathing propulsion:** Slender vehicles with a large lift/drag ratio which fly at small angles of attack. They are equipped with ramjet or scramjet propulsion systems (e.g. the lower stage of the two-stage-to-orbit system SÄNGER).

3. **Non-winged reentry vehicles:** Very blunt vehicles with limited control (quasi-ballistic reentry), but with a compact design and excellent endurance against high thermal loads. This group includes most of reentry capsules (e.g. HUYGENS, APOLLO, SOYUZ and VIKING among others).

The specific characteristics of these classes, that are summarised in Table 1.1, make their aerothermodynamic analysis very different in each case. Due to these intrinsic differences, in this project the analysis will be constrained to *non-winged reentry vehicles*. The particular interest in reentry capsules is due to their large use for manned missions, where the thermal effects are critical and a detailed analysis is necessary to assure that the thermal loads are below the established limits and, hence, will not damage the astronauts during the descent phase.

Table 1.1. Comparative consideration of particular aerothermodynamic features of the three reference classes of hypersonic vehicles. Features which are common to all classes are not listed. [1]

Item	**Winged reentry vehicles**	**Cruise and acceleration vehicles**	**Non-winged vehicles**
Mach number range	$0-30$	$0-7$	$0-30$
Configuration	blunt	slender	very blunt
Flight time	short	long	short
Angle of attack	large	small	head on
Drag	large	small	large
Lift/drag ratio	small	large	very small
Flow field	compressibility-effects dominated	viscosity-effects dominated	compressibility-effects dominated
Thermal surface effects: viscous	not important/locally important	very important	not important
Thermal surface effects: thermo-chemical	very important	important	very important

1.4.1 Non-winged reentry vehicles

Nowadays, *non-winged reentry vehicles* are extensively used for the transport of payload into space and back to the Earth, due to their simplicity and compact design [1]. The selected geometry for these vehicles, represented schematically in Figure 1.1, involves the use of very blunt bodies, as well as blunted cones and biconics, in order to withstand the extreme thermal loads that are present during the reentry phase.

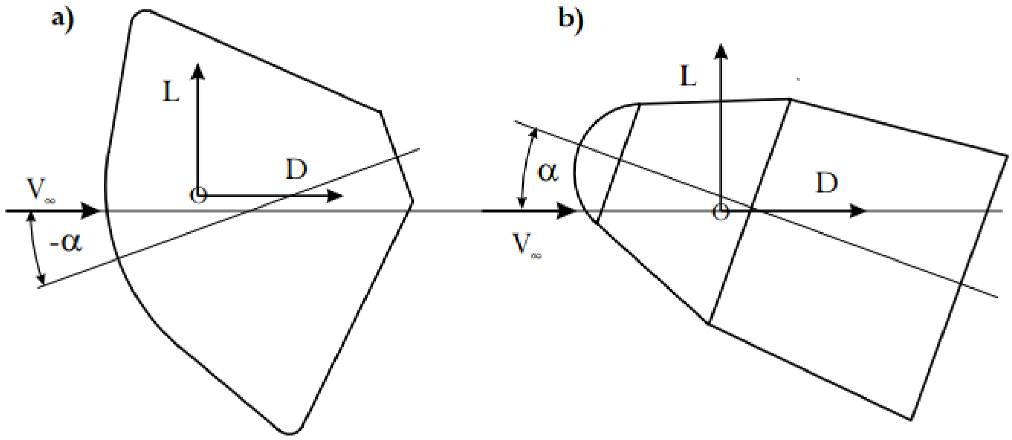


Figure 1.1. Typical geometries of NW-RV: a) blunt capsule, b) bicone. [1]

Among these two possible geometries, the blunt capsules are the traditional approach and this research will be focused on them. However, the use of conic geometries may be interesting in some applications due to the possibility of including body flaps as control surfaces, which leads to more precise descent manoeuvres.

There are two main approaches when designing a capsule, based on the aerodynamic efficiency: ballistic capsules and lifting capsules.

Ballistic capsules are cheaper than lifting capsules, but lack control mechanisms, which leads to a quasi-ballistic reentry. The lift is near zero and the atmospheric drag determines the trajectory through the *ballistic factor*, β:

$$\beta = \frac{m}{SC_D}, \tag{1}$$

with m being the mass, S the reference surface and C_D the drag coefficient. In general, low ballistic factors are preferred due to they provide low thermal loads, a lower mass and a faster reentry [1].

On the other hand, lifting capsules can be guided and controlled during the descent. However, the lift generated by blunt capsules is low, even with high angles of attack, and the aerodynamic efficiency is in the range of $0.3 \leq E \leq 0.4$. It is important to mention, for the understanding of the further analysis, that a positive lift is only obtained with negative angles of attack [1]. Manned capsules are usually lifting (e.g. APOLLO and SOYUZ capsules).

1.5 Thermal protection system

The protection of reentry vehicles from the extreme thermal loads during the descent, requires the use of a TPS (Thermal Protection System). The TPS acts as a shield to assure the safety of the payload inside the capsule. Furthermore, the TPS forms the external surface of the body and, hence, its aerodynamic shape. Therefore, an appropriate design of the thermal protection system is essential in every space mission with a reentry phase.

There are three main methods to design a TPS [4]: passive methods, semi-passive methods and active methods.

1.5.1 Passive methods

They are formed by a heat sink, a hot structure or an insulated structure. A heat sink is a structure, generally metallic, with a very high thermal capacity that absorbs the aerodynamic heat and decreases the temperature increase rate. However, it have to be used for short periods of time [21].

Hot structures have a high emissivity to favour a radiative heat exchange as the surface temperature increases [22]. With this approach, the maximum temperature the vehicle can withstand is an important limiting factor.

Insulated structures consist of an external radiative layer, which radiates back part of the incident heat, and an internal insulation layer (ceramic-fiber matrix), with a low conductivity to reduce the heat transfer process [23].

Some examples of real reentry vehicles with these passive protection systems are shown in Figure 1.2.

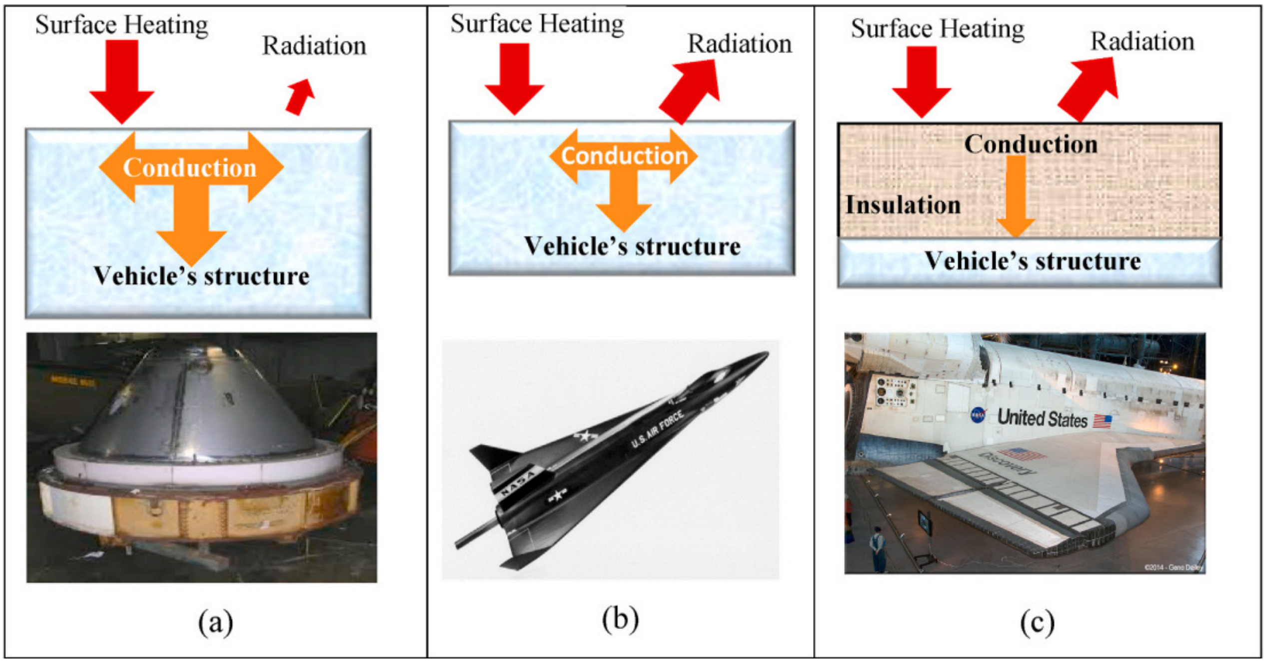


Figure 1.2. Schematics of (a) Heat sink (b) Hot structure (c) Insulated Surface. [4]

1.5.2 Semi-passive methods

They are formed by heat pipes or ablative surfaces. Heat pipes are devices designed to passively transport heat from hot areas to cooler ones, helping to obtain an uniform temperature distribution. They work due to capillarity forces on a fluid and a phase change process. Although heat pipes are frequently used in satellites, they are not massively used in reentry vehicles, due to the efficiency increases in micro-gravity conditions, which are not present during the descent, due to the very high g-forces.

Ablative surfaces absorb thermal energy through latent heat during the vaporisation process, reducing the heat transfer inside the vehicle. The most versatile materials to be used with this purpose are polymeric based ablators [24].

1.5.3 Active methods

There are three main approaches: convective cooling, film cooling and transpiration cooling. Convective cooling is based on pumping a liquid coolant underneath the TPS to avoid excessive heating. This method has been used in Space Shuttle using loops of cold liquid hydrogen [25].

Film cooling consists on the injection of coolant on the surface of the vehicle, insulating the structure and reducing its maximum temperature. Film cooling is more effective at higher speeds [26] and, hence, it is used in hypersonic vehicles.

Transpiration cooling is used for extremely high temperatures (over 1649 °C) and flight times greater than 1.5 hours [27]. In this method, the coolant is pumped through a porous material, towards the hot external surface, cooling the structure down. Furthermore, the coolant forms a film on the surface of the vehicle, reducing the heat flux conducted to the porous structure [28].

The basic functioning of these three methods is shown schematically in Figure 1.3.

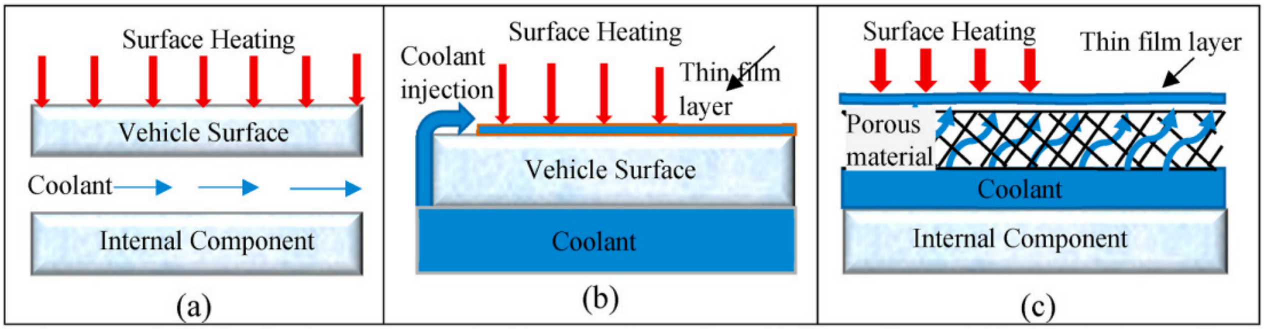


Figure 1.3. Schematic of active TPS methods: (a) convective cooling, (b) film cooling and (c) transpiration cooling. [4]

1.6 Atmospheric model

The reentry analysis and the underlying aerodynamic problem is strongly related to the chosen *atmospheric model.* In general, atmospheric models provide the distribution of density and temperature as a function of the height. More complex models may also include the chemical composition of the atmosphere at different heights. However, as the model becomes more precise, the number of variables which have influence over the model increases, including, among others, the latitude, the solar activity, the seasonal cycles or even the time, for dynamic models.

The ISA (International Standard Atmosphere) model is simple and could be a good choice for a *worst case analysis*, when an extremely high accuracy should not be required. Hence, it will be used as a first approach in this research. However, for detailed solutions in each timestamp during the reentry, it is advisable to use more sophisticated models, recommended by the ECSS [29], such as NRLMSISE-00 (NRL Mass Spectrometer, Incoherent Scatter Radar Extended Model) or JB-2006 (Jacchia-Bowman 2006).

1.6.1 International Standard Atmosphere

The ISA model [2] considers an atmosphere formed by a perfect gas, and the pressure can be obtained as a function of the density and the temperature for each height:

$$p = \rho R T, \tag{2}$$

with p as the pressure, ρ as the density, T as the temperature and R as the gas constant.

This model considers a static atmosphere and the hydrostatic equilibrium can be applied, which provides a relationship between the pressure, the density and the geometric height, z:

$$\frac{\mathrm{d}p}{\mathrm{d}z} = -\rho g(z), \tag{3}$$

where $g(z)$ is the gravitational acceleration.

However, the ISA model considers an uniform gravitational acceleration g_0 equal to its value at sea level. This fact makes convenient to use a *geopotential height*, h, instead of the geometric height, z. Both heights are related by the following equation:

$$h = \frac{1}{g_0} \int_0^z g(z)\,\mathrm{d}z. \tag{4}$$

It is usual to use h as the *standard geopotential height*, where g_0 corresponds to a geographic latitude of 45.5425 degrees [2].

The combination of expressions (2) to (4) allows to obtain an ODE for the pressure:

$$\frac{\mathrm{d}p}{\mathrm{d}h} = -g_0 \frac{p}{RT(h)}. \tag{5}$$

Finally, this model considers a linear variation of the temperature with the height in each layer of the atmosphere:

$$T = T_0 + \lambda(h - h_0). \tag{6}$$

This linear model for the temperature allows to solve the ODE (5) and to obtain the pressure as a function of the geopotential height:

$$\frac{p}{p_0} = \left(\frac{T}{T_0}\right)^{-\frac{g_0}{R\lambda}} = \left[\frac{T_0 + \lambda(h - h_0)}{T_0}\right]^{-\frac{g_0}{R\lambda}} \qquad if \quad \lambda \neq 0, \tag{7}$$

$$\frac{p}{p_0} = \exp\left(-\frac{g_0}{RT_0}(h - h_0)\right) \qquad if \quad \lambda = 0. \tag{8}$$

The constants λ, T_0, p_0 and h_0 are labelled for different layers of the atmosphere. The main layers are the *troposphere* (up to 11 km), the stratosphere (from 11 km to 50 km), the mesosphere (from 50 km to 90 km), the thermosphere (from 90 km to 400 km) and the exosphere (from the upper limit of the thermosphere and up to the end of the atmosphere, with an altitude around 500 km). The first values of λ, T_0, p_0 and h_0 are shown in Table 1.2.

Table 1.2. Parameters for the ISA atmosphere in the first layers. [2]

h_0 [km]	T_0 [K]	p_0 [Pa]	λ [K/m]
0	288.15	101325	$-6.5 \cdot 10^{-3}$
11	216.65	22632	0
20	216.65	5474	$1.0 \cdot 10^{-3}$
32	228.65	868	$2.8 \cdot 10^{-3}$

1.6.2 NRLMSISE-00

This model describes the neutral temperature and composition (species densities) of the atmosphere [29]. Its use is valid from ground level and up to the exosphere. The model defines a dynamic atmosphere and provides data of the spatial and temporal distribution, which comes from satellites, rockets and radars. There are two main indices required in this model [29]:

- $F10.7$: Daily value of the previous day and the 81-day average, centred on the input day.
- A_p: Daily value.

1.6.3 JB-2006

This model describes the temperature and total density in the thermosphere and the exosphere [29]. Its use is valid from 120 km up to the exosphere. It is a very accurate model for the total density. There are four main indices required in this model [29]:

- $F10.7$: Tabular value one day earlier and the 81-day average, centred on the input time.
- A_p: Tabular value 6.7 hours earlier.
- $S10.7$: Tabular value one day earlier and the 81-day average, centred on the input time.
- $M10.7$: Tabular value five days earlier and the 81-day average, centred on the input time.

2 Thermal problem

The thermal problem analyses the temperature distribution in a solid domain which, in this case, is the reentry capsule. The extreme environment during the reentry phase induces the need of a TPS (see sec. 1.5) to withstand the thermal loads. In general, a TPS has different layers, with specific properties and materials [30], in order to improve the general behaviour against a high incident heat flux.

The selected approach to solve the CHT (Conjugate Heat Transfer) problem, through the coupling between fluid and solid, is based on a steady resolution within the solid domain (see sec. 4). Therefore, the selected TPS must be designed for its operation in steady-state conditions, when the temperature has been stabilised in the whole domain. Even though this approach is not completely real, it can be used as a *worst case analysis* of the reentry phase; due to, in general, the critical phase only lasts around 500 s [3].

Among the passive and semi-passive methods previously described to design a TPS, only those that radiates back part of the incident heat can be used indefinitely. These are the *hot structures* and the *insulated structures*. In this case, the problem to be analysed will be that of an insulated structure as a TPS for the solid, because it keeps the advantages of hot structures, while it reduces the internal heating with a low conductivity near the surface. The use of active methods as a TPS is a very interesting field, but the use of a coolant generates an additional fluid domain, whose simulation would be unfeasible with the current tools and, hence, it will not be discussed in this research.

The aerodynamic thermal loads affect this domain through the boundary conditions, that will be used to solve the *heat equation* in the solid with a FEM solver. In this case, the open-source library MFEM [17] will be used with this purpose. Although this library is written in C++, a Pyhton wrapper provided by the developers will be used as a tool for a simpler implementation. Moreover, this Python wrapper will be extremely useful when coupling this FEM solver and the CFD solver, which is written in Fortran (see sec. 8). The FEM solver will be validated with a test case to prove the adequate resolution of a thermal problem.

Furthermore, to deal with the resolution of the thermal problem it is necessary to define a *mesh* in the solid domain, in whose elements the equations will be solved. The software used to mesh the domain will be another open-source application called Gmsh [20], that is compatible with the solid and fluid solvers.

2.1 General equations

The resolution of the thermal problem begins with the *energy balance equation*:

$$\frac{\mathrm{d}E}{\mathrm{d}t} = \dot{W} + \dot{Q}, \tag{9}$$

with E as the energy, t as the time, W as the work and Q as the heat. If the previous expression is constrained to the thermal energy, the following equation is obtained:

$$\frac{\mathrm{d}E_{th}}{\mathrm{d}t} = \dot{W}_{dis} + \dot{Q}. \tag{10}$$

The solution of the equation (10), in a particular case, allows to obtain the evolution of the thermal energy, known the heat and the dissipated power of the system.

The interest of the previous expression lies on the relation between the thermal energy and the temperature. If the selected domain has a constant thermal capacity, without phase changes, and the whole system is incompressible, then the expression can be rewritten:

$$mc\frac{\mathrm{d}T}{\mathrm{d}t} = \dot{W}_{dis} + \dot{Q}, \tag{11}$$

where m is the mass, c the specific thermal capacity and T the temperature.

The expression (11) can be transformed to an integral notation for a continuous medium:

$$\int_V \rho c\frac{\mathrm{d}T}{\mathrm{d}t}\,\mathrm{d}V = \int_V \phi\,\mathrm{d}V + \int_A -\vec{q}\cdot\vec{n}\,\mathrm{d}A, \tag{12}$$

where, following the *Fourier's law*:

$$\vec{q} = -k\nabla T, \tag{13}$$

with k as the thermal conductivity and ϕ as the dissipated power per volume unit.

If the *Gauss divergence theorem* is applied over the equation (12), the *heat equation* is obtained:

$$\rho c\frac{\partial T}{\partial t} = \nabla\cdot(k\nabla T) + \phi \qquad \text{in} \quad \Omega, \tag{14}$$

with Ω as the control volume.

The previous equation is complemented with some BC. For the selected approach to solve the CHT problem, the capsule will be defined through Dirichlet BC, as it will be explained in sec. 4.1:

$$T = T_0(x, y, z) \qquad \text{in} \quad \partial\Omega, \tag{15}$$

where the temperature T_0 is obtained from the solution in the fluid domain. The steady-state analysis is based on the same equation, but taking into account that temporal variations are not considered: $\frac{\partial T}{\partial t} = 0$.

If the chosen TPS were a *heat sink*, then the expression (14) would be the one to be solved in the whole solid domain and no additional considerations would be required. However, the *insulated structure* has a high emissivity surface to increase the radiated heat and cool the structure. In this case, some additional heat fluxes must be taken into account on the boundaries, when exchanging information with the fluid.

2.1.1 Thermal radiation

The total black-body radiation emitted for a given temperature is given by the *Stefan-Boltzmann law*:

$$E_b(T) = \sigma T^4, \tag{16}$$

where $\sigma = 5.67 \cdot 10^{-8} \, \frac{\mathrm{W}}{\mathrm{m^2 K^4}}$ is the Stefan-Boltzmann constant.

Some sources of radiation in space, like the Sun or the CBR (Cosmic Background Radiation), can be considered as black bodies at a constant temperature. However, the thermo-optical properties on the surface of the reentry capsule depend on different factors and the black body hypothesis is not accurate enough in this case. To solve this problem, the capsule is considered to have an emissivity, ε, with $0 < \varepsilon < 1$, which is equal to the absorptance, α, following the *Kirchhoff's law*:

$$\varepsilon(\lambda, \theta, \phi, T) = \alpha(\lambda, \theta, \phi, T), \tag{17}$$

where the emissivity and the absorptance depend, in a general case, on the wavelength, λ, the latitude, θ, and the longitude, ϕ, both computed in body axis, and the temperature, T. The emissivity represents the ratio between the emitted radiation and the radiation of a black body at the same temperature.

However, without an important lack of accuracy, the reentry vehicle can be considered as a *diffuse* body, which implies that the directionality of the radiation, (θ, ϕ) is negligible: emissivity does not significantly change with the direction [31]. Furthermore, the capsule is supposed to be an *opaque* body for the incident radiation, which means that the radiation that is not absorbed is reflected (no radiation is transmitted through the body).

Finally, for a given temperature, it is usual to integrate the considered radiation spectrum at the given temperature to avoid the dependence with the wavelength. The emissivity value obtained with the integrated radiation is the *total hemispherical emissivity*, but it is called simply emissivity because it is the value used into practice:

$$\varepsilon(T) = \frac{\int_0^\infty \varepsilon(\lambda, T) E_b(\lambda, T)\, \mathrm{d}\lambda}{E_b(T)}, \tag{18}$$

where $E_b(T)$ is the total radiation emitted by a black body at the same temperature (see expression (16)) and $E_b(\lambda, T)$ is the black body radiation spectrum, given by *Planck's law*:

$$E_b(\lambda, T) = \frac{2\pi h c^2}{\lambda^5 \left[\exp\left(\frac{hc}{\lambda k T}\right) - 1\right]}, \tag{19}$$

with $h = 6.626 \cdot 10^{-34}$ J $\cdot$ s as the Planck's constant, $k = 1.381 \cdot 10^{-23}$ J/K as the Boltzmann's constant and $c = 3 \cdot 10^8$ m/s as the speed of light in vacuum.

This emissivity value only depends on the temperature and on the surface's thermo-optical properties. However, in most of thermal problems, it is enough to consider a *two spectral band model* [31], corresponding to the *solar band* and the *infrared band*. In each band, the body is considered as a *gray body*, which means that the emissivity and the absorptance are constant. Both spectral bands are separated enough to be considered as independent, as it is shown in Figure 2.1.

The solar band is only used to take into account the effect of solar radiation, which corresponds to a black body at 5800 K. The peak of the radiation spectrum in this band is in the visible range (from 0.4 to 0.7 μm), but it has also energy within the near infrared (from 0.7 to 3 μm) and the ultraviolet (from 0.3 to 0.4 μm). In this situation, the only relevant property is the solar absorptance, α_s, which determines the percentage of solar radiation that is absorbed by the considered surface. The emissivity in the solar band, ε_s, is equal to the absorptance, as it was explained before through expression (17); however, it is never used as no structural material can withstand the Sun's temperature and, hence, can not emit radiation in the solar band.

The infrared band is used to model the thermal radiation emitted by bodies with a temperature between 100 K and 1000 K [31], whose energy is mostly within the far infrared range (from 3 to 30 μm). However, some materials preserve an almost constant value of emissivity for a wider range of temperatures. In this case, the infrared emissivity, ε_{IR}, will be used to determine the amount of radiation emitted by the reentry vehicle. The infrared absorptance, α_{IR}, is equal the the infrared emissivity (see eq. (17)) and it is used to model the radiation emitted in the infrared band by other bodies, like the Earth, that is absorbed by the capsule.

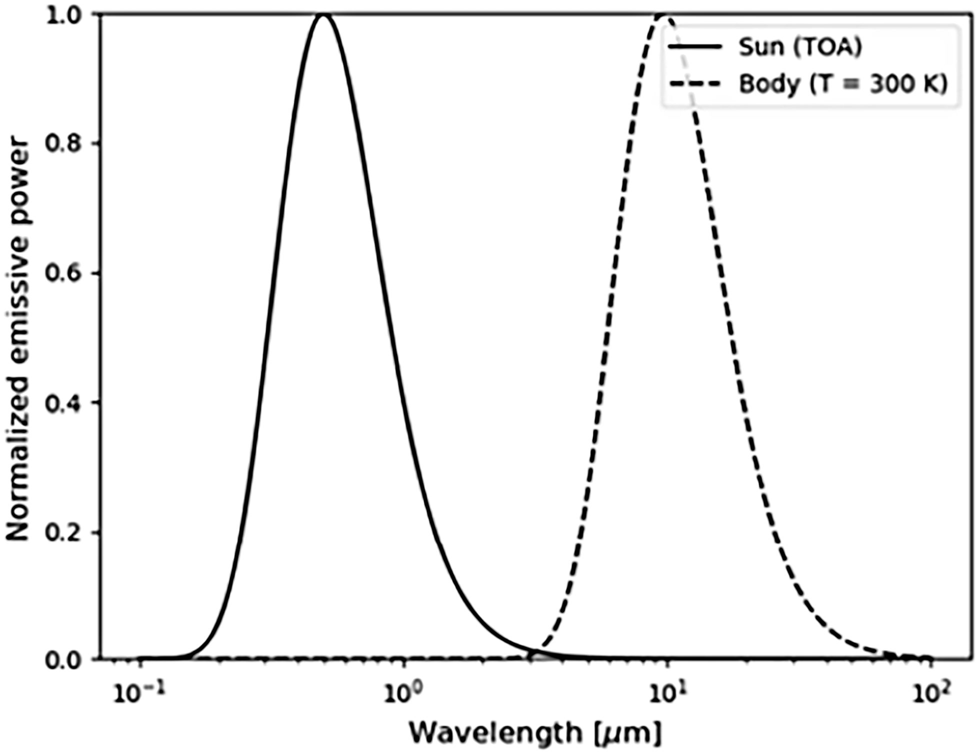


Figure 2.1. Normalised emissive power in a two band radiation model. [5]

Finally, taking all these factors into account, the radiative heat fluxes emitted and absorbed by the capsule can be obtained, considering the surrounding air as *transparent* to thermal radiation:

$$q_{rad} = \alpha_s E_s A_f + \alpha_{IR} A F_{c\to E} \varepsilon_E \sigma T_E^4 + \alpha_{IR} A F_{c\to\infty} \sigma T_\infty^4 - \varepsilon_{IR} A \sigma T^4, \tag{20}$$

where $E_s = 1361\,\mathrm{W/m^2}$ is the solar irradiation at 1 UA, A_f is the frontal area of the capsule (normal to the Sun's direction) that absorbs the solar radiation, A is the total area of the capsule, $F_{c\to E}$ and $F_{c\to\infty}$ are the view factors from the capsule to the Earth and to the void space respectively, ε_E is the Earth's emissivity (considered as a gray body), T_E is the mean Earth's temperature and $T_\infty = 2.7\,\mathrm{K}$ is the temperature of the CBR (Cosmic Background Radiation). The albedo is very fluctuating during the reentry, due to the great variation in length and height during this phase, and will not be considered in this research.

Furthermore, taking into account that $T_\infty \ll T$, the previous equation can be simplified:

$$q_{rad} = \alpha_s E_s A_f + \alpha_{IR} A_c F_{c\to E} \varepsilon_E \sigma T_E^4 - \varepsilon_{IR} A_c \sigma T^4 \qquad \text{in} \quad \partial\Omega. \tag{21}$$

This expression applies to all the considered boundaries, $\partial\Omega$, and, together with the expression (14), that applies to the inner volume, Ω, it allows to obtain the temperature distribution in the whole capsule.

However, the equation (14), together with the Dirichlet BC (15) allows to define completely the temperature distribution in the solid domain. Hence, to take into account these radiative fluxes on the boundaries, they will be applied as an input to the fluid domain, together with the conductive heat fluxes, which allow to update the boundary conditions for the fluid. As it will be explained later in sec. 4, the coupling between fluid and solid is an iterative process and, therefore, the effect of the radiative flux over the fluid affects also the solid domain the next iteration, until the equilibrium in both domains is achieved.

2.2 Geometry and physical properties

Once the thermal problem has been defined in the solid domain, it is important to define the geometrical and physical properties of the considered reentry capsule that will be used to solve the CHT problem in a hypersonic flow.

For the present analysis, the Apollo Command Module, which is one of the most emblematic capsules of all history, will be the capsule used to solve the problem and to validate the proposed methodology. This reentry vehicle, whose geometry is represented schematically in Figure 2.2 and Table 2.1, is a lifting capsule (see sec. 1.4.1) and, hence, a positive lift is obtained with negative angles of attack. In this case, an angle of attack of $-18°$ will be used as representative of the usual behaviour of the vehicle.

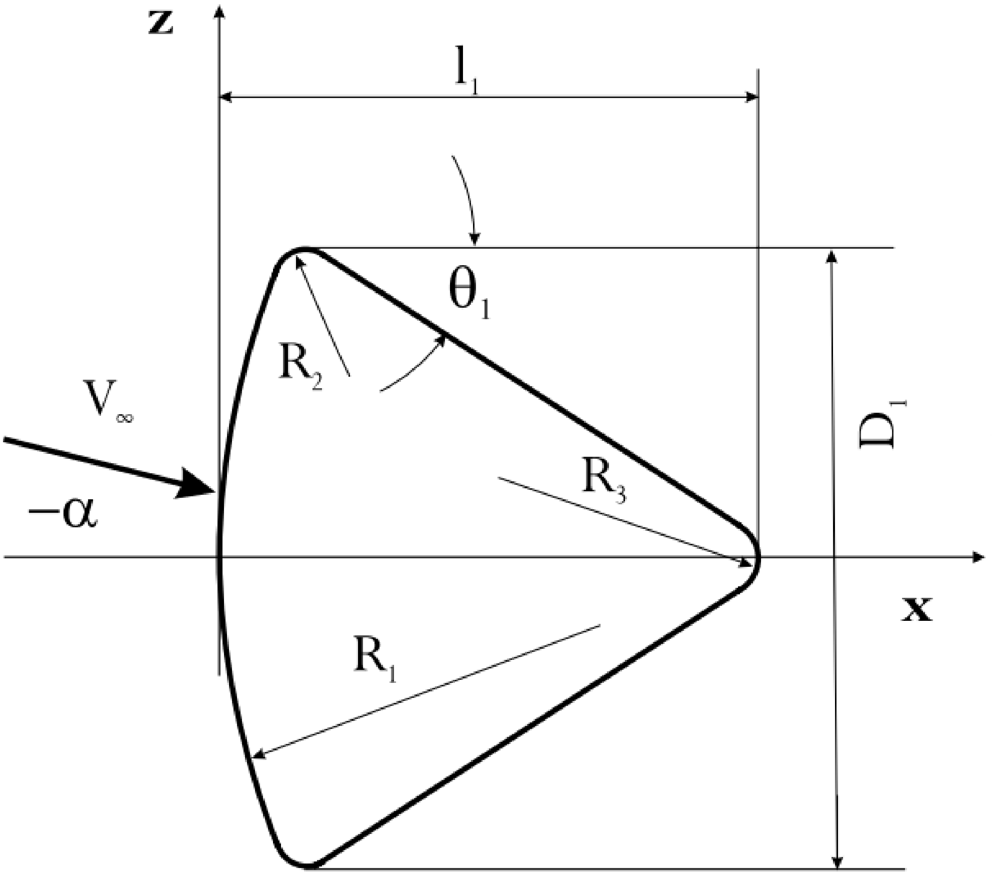


Figure 2.2. Shape definitions of the lifting capsule APOLLO. [1]

Table 2.1. Geometrical data of the Apollo capsule. [1]

$\mathbf{l_1}$ [mm]	$\mathbf{D_1}$ [mm]	$\mathbf{R_1}$ [mm]	$\mathbf{R_2}$ [mm]	$\mathbf{R_3}$ [mm]	θ_1 [°]
3529	3912	4694	196	232	33

This capsule has been selected due to two main reasons:

1. The simplicity of the geometry allows to define a more uniform mesh and it provides a simpler way to monitor the coupling process.
2. The high descent velocity of this capsule, leads to the generation of extreme heat fluxes and incredibly high temperatures after the shock wave. Therefore, this is one of the most critical scenarios and it will allow to test the effectiveness of the radiative insulation structure as a thermal protection system.

The meshing process has been performed assuring a higher density of elements near the surface to improve the resolution of the coupling with the fluid. The mesh is represented in Figure 2.3. This mesh is formed by the thermal protection system only, while the inside of the capsule will be modelled through different boundary conditions (see sec. 5). The whole mesh contains 4880 elements and 4400 nodes.

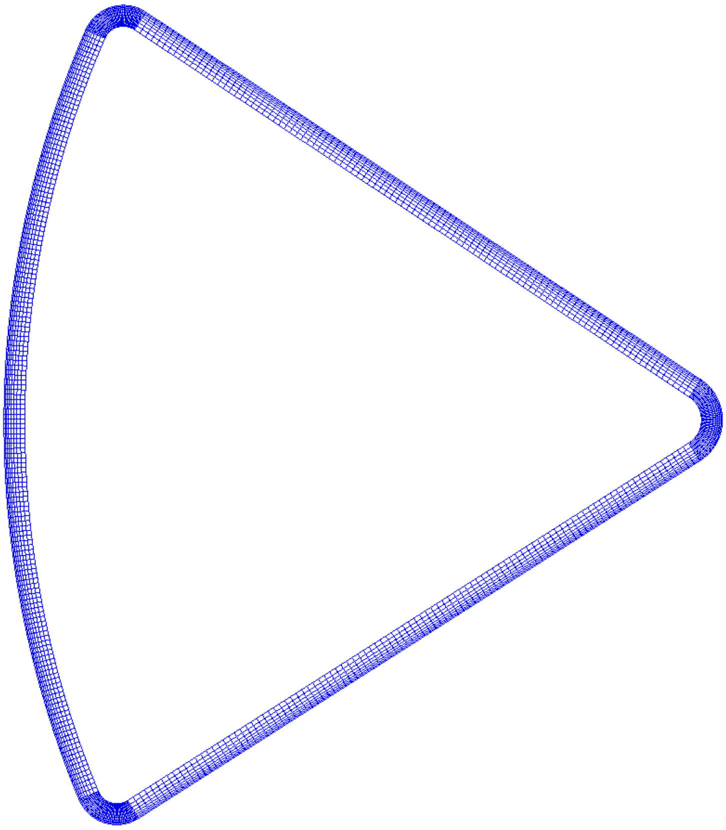


Figure 2.3. Mesh for the TPS of the Apollo capsule.

The external TPS is the part in charge of the thermal control of the capsule and it has a double functionality: it radiates part of the heat to the space, as the surrounding air is considered as transparent to thermal radiation (see sec. 2.1.1), and it insulates the interior of the capsule from the external incident heat. Therefore, the selected material must be able to withstand high temperatures, must have a low thermal conductivity and its emissivity should be as high as possible.

Taking into account these considerations, the selected material for the TPS is a material, developed by NASA, called TUFROC (Toughened Uni-piece Fibrous Reinforced Oxidation-resistant Composite), which was specifically created as a non-ablative insulated structure for reentry vehicles. The key of this material is the ceramic coating HETC (High Efficiency, Tantalum-based Ceramic), which is capable of withstanding much higher temperatures than the glass-based coating employed on Shuttle tiles [32]. Besides, several tests have demonstrated that it can resist heat fluxes up to $3\,\mathrm{MW/m^2}$ and maximum temperatures of $2000\,\mathrm{K}$ approximately [33]. The main physical and geometrical properties of the TPS used in the capsule are compiled in Table 2.2 and their value is almost constant for the whole operative range of temperatures. Specially, the emissivity does not significantly change with temperature.

Table 2.2. Physical and geometrical properties of the TPS used to protect the capsule.

thickness [cm]	**density** [$\mathrm{kg/m^3}$]	**conductivity** [$\mathrm{W/(m \cdot K)}$]	**emissivity**	**absorptance**
10	400	0.1	0.9	0.9

Finally, in order to compute the radiative fluxes emitted by the surface of the capsule, it is necessary to estimate the view factor from the vehicle to the Earth, as it is present in the expression (21). This view factor represents the percentage of the capsule that is affected by the Earth's radiation. During the critical phase of the descent, the capsule is close to the Earth and, hence, this view factor will be approximately 0.5, which means that half of the capsule may be influenced by the radiation emitted by the Earth.

However, it is possible to obtain a better estimation based on the assumption that the geometry of the capsule, seen from the Earth and with the considered angle of attack, is similar to a small sphere. In this situation, there is an analytical approach to compute the view factor as a function of the height of the vehicle over the Earth's surface [31]:

$$F_{c \to E} = \frac{1}{2}\left(1 - \sqrt{1 - \frac{1}{h^2}}\right), \tag{22}$$

where $h = H/R_E$, H is the distance from the capsule to the centre of the Earth and R_E is the Earth's radius. As it will be explained later in sec. 3.2, the selected height over the Earth will be $54.6\,\mathrm{km}$, which is included in the critical phase of the descent trajectory. In this case, the value of the view factor is $F_{c \to E} = 0.435$.

Although this value is not exact, it is a good approximation. If more precision was required, the best way to compute the view factor is through Monte Carlo methods [34] for the generation of random rays from the capsule, in order to detect the percentage that impacts on the Earth. However, the Earth radiation is a very small contribution compared to the heating effect of the fluid, and there is no need to improve the accuracy of this view factor.

2.3 FEM validation: PCB test

The way to assure that the FEM solver works correctly and provides an accurate solution, is to perform a validation through a test case simulation. Once the validation test has been computed successfully, the inherent risk associated to the use of this solver will decrease, leading to more reliable solutions when coupled with the CFD solver.

The chosen test is the analysis of a PCB with three IC (Integrated Circuits), which dissipate power. This test has been solved in a previous research [6], and allows to verify different performances of the solver, such as the correct modelling of Dirichlet and Neumann boundary conditions, the use of multiple materials (Cu and FR4 for the baseplate and the IC material), the power dissipation and the thermal radiation.

The problem focuses on obtaining the steady-state temperature distribution for the given conditions. In particular, the maximum temperature is highly related to the system functioning due to, if a certain threshold is exceeded, the failure probability increases. The PCB to be analysed and the boundary conditions are shown schematically in Figure 2.4.

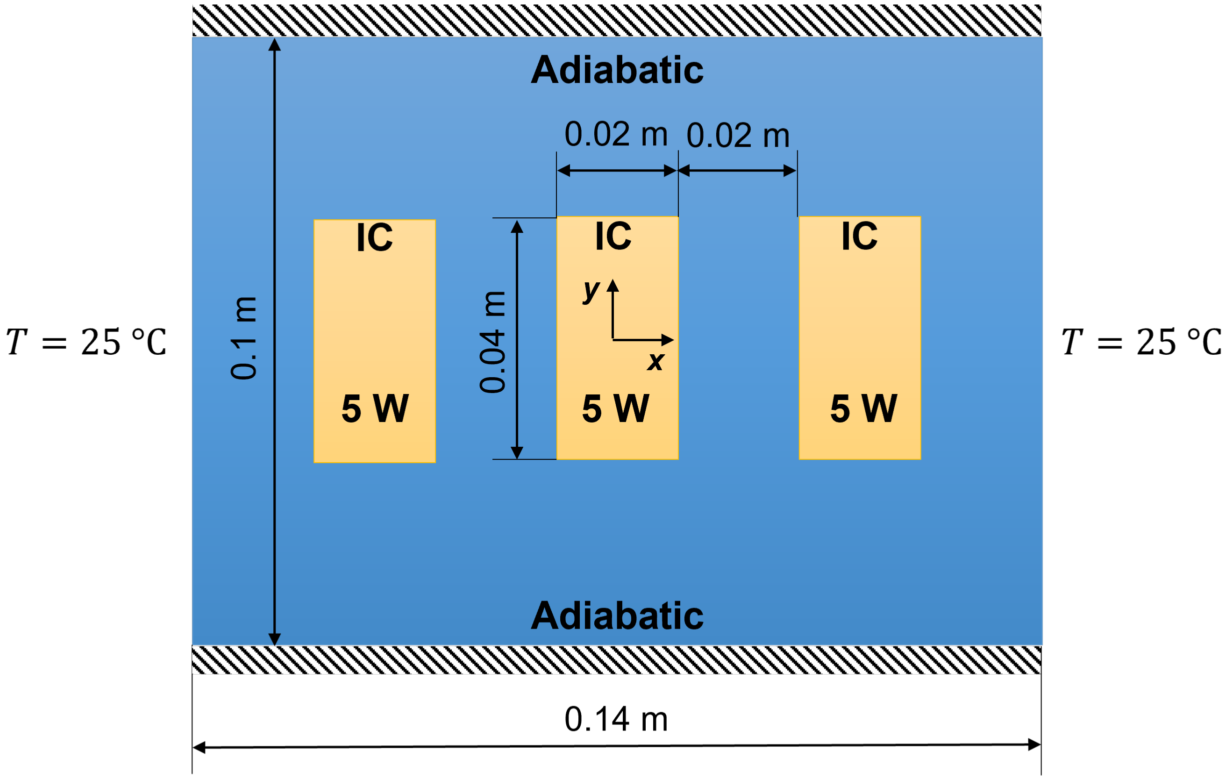


Figure 2.4. Schematics of the analysed PCB.

The required properties to solve the thermal problem are compiled in Table 2.3. All variables with the subscript BP are referred to the baseplate of the PCB only (without IC) and those with the subscript IC are referred to the PCB, with baseplate and integrated circuits. Furthermore, to simulate the thermal radiation, the PCB is considered to be surrounded by a black body at 45 °C, which allows a radiative heat exchange on both sides of the PCB. Due to the heat equation is solved in a 2D domain and the z axis is not considered for the integration, the effect of the radiation is not imposed as a BC, but as a source term in the heat equation. In this case, the equation to be solved is given by the following expression:

$$\rho c \frac{\partial T}{\partial t} = \nabla \cdot (k \nabla T) + \phi + \varepsilon_{up} A \sigma (T_\infty^4 - T^4) + \varepsilon_{down} A \sigma (T_\infty^4 - T^4), \tag{23}$$

where the area $A = L_{x,BP} L_{y,BP}$.

Table 2.3.
General data of the test problem: geometry, physical and thermal properties; imposed temperatures and dissipated power.

Geometry

$L_{x,BP}$ [m]	$L_{y,BP}$ [m]	z_{BP} [m]	$L_{x,IC}$ [m]	$L_{y,IC}$ [m]	z_{IC} [m]
0.14	0.1	$1.6 \cdot 10^{-3}$	0.02	0.04	$3 \cdot 10^{-3}$

Density and thermal capacity

$\rho_{BP} c_{BP}$ $\left[\frac{\mathrm{J}}{\mathrm{m^3 \cdot K}}\right]$	$\rho_{IC} c_{IC}$ $\left[\frac{\mathrm{J}}{\mathrm{m^3 \cdot K}}\right]$
$1.4 \cdot 10^{6}$	$5.92 \cdot 10^{6}$

Thermal conductivity

k_{BP} $\left[\frac{\mathrm{W}}{\mathrm{m \cdot K}}\right]$	k_{IC} $\left[\frac{\mathrm{W}}{\mathrm{m \cdot K}}\right]$
13.9	37.4

Emissivity

ε_{up} (upper side with IC)	ε_{down} (lower side without IC)
0.7	0.5

Temperature and dissipated power

T_0 (wall) [°C]	$\dot{W}_{dis}$ (each IC) [W]	ϕ (only for ICs) $\left[\frac{\mathrm{W}}{\mathrm{m^3}}\right]$	T_∞ [°C]
25	5	$1.36 \cdot 10^{6}$	45

The solution of the test problem will be obtained with a FEM solver and, hence, it is necessary to define a mesh before the temperature distribution is computed. The selected mesh is represented in Figure 2.5, where the size of each element is, approximately, defined by $\Delta x = \Delta y = 1.4 \cdot 10^{-3}\,\mathrm{m}$.

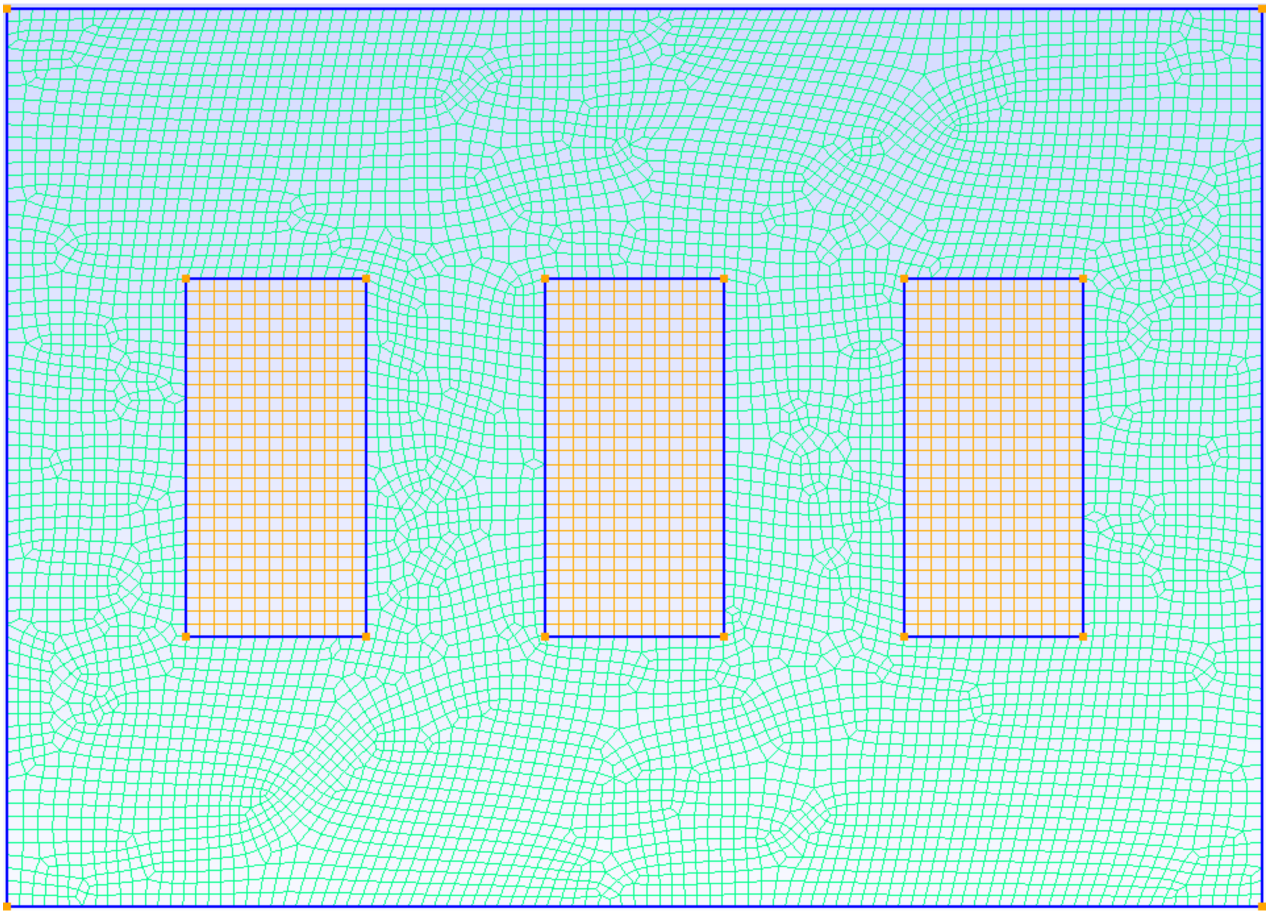


Figure 2.5. Mesh used to solve the PCB test thermal problem.

The final steady-state temperature is shown in Figure 2.6 and the reference distribution [6] is shown in Figure 2.7. The maximum temperature is reached in the middle of the PCB, with a value of 382 K in both cases (this analysis and the reference solution).

These results show that both figures are almost identical, which implies that the FEM solver MFEM is adequate to solve a thermal problem in a solid domain.

Furthermore, it is possible to compare the temperature evolution along the central line of the PCB. The obtained result is represented in Figure 2.8, together with the reference result. Again, both solutions are very similar and, hence, the FEM solver has been successfully validated.

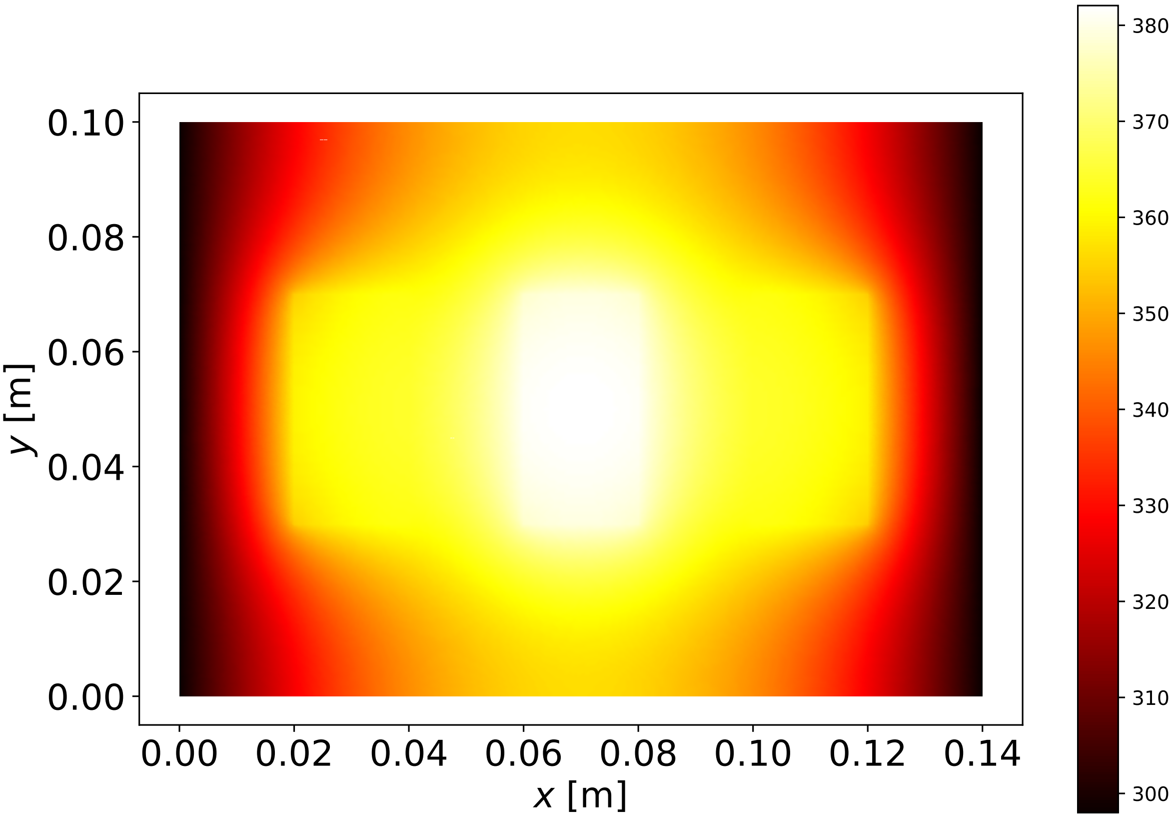


Figure 2.6. Temperature distribution in steady-state.

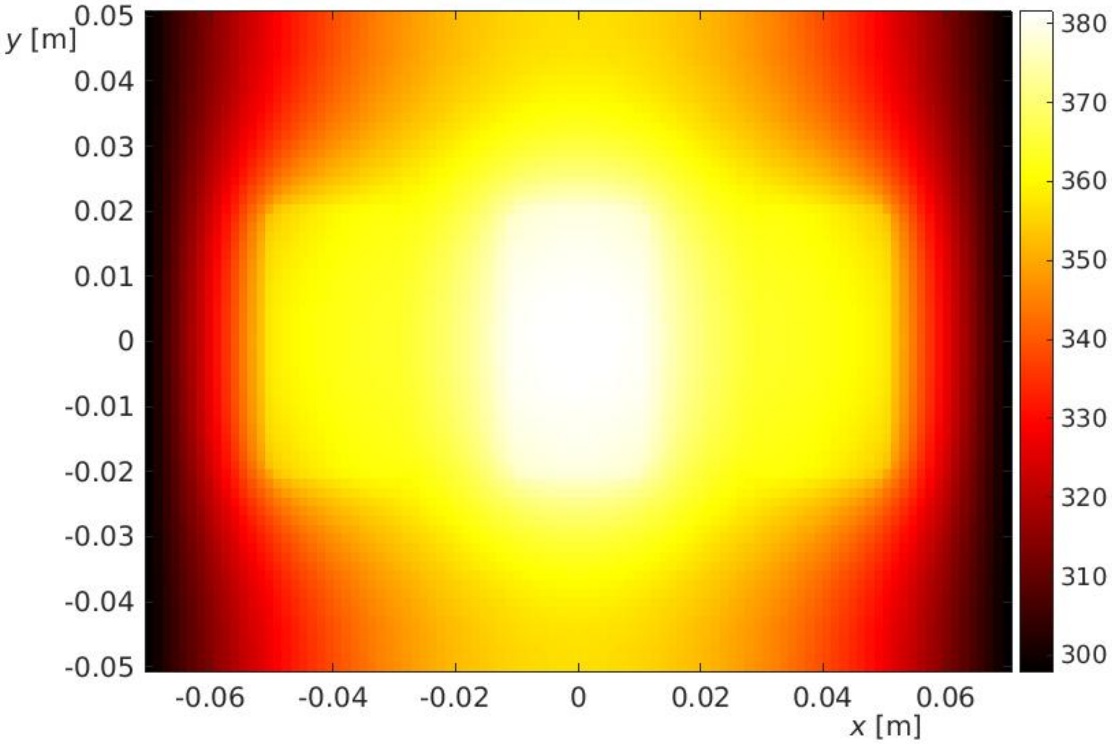


Figure 2.7. Reference temperature distribution in steady-state. [6]

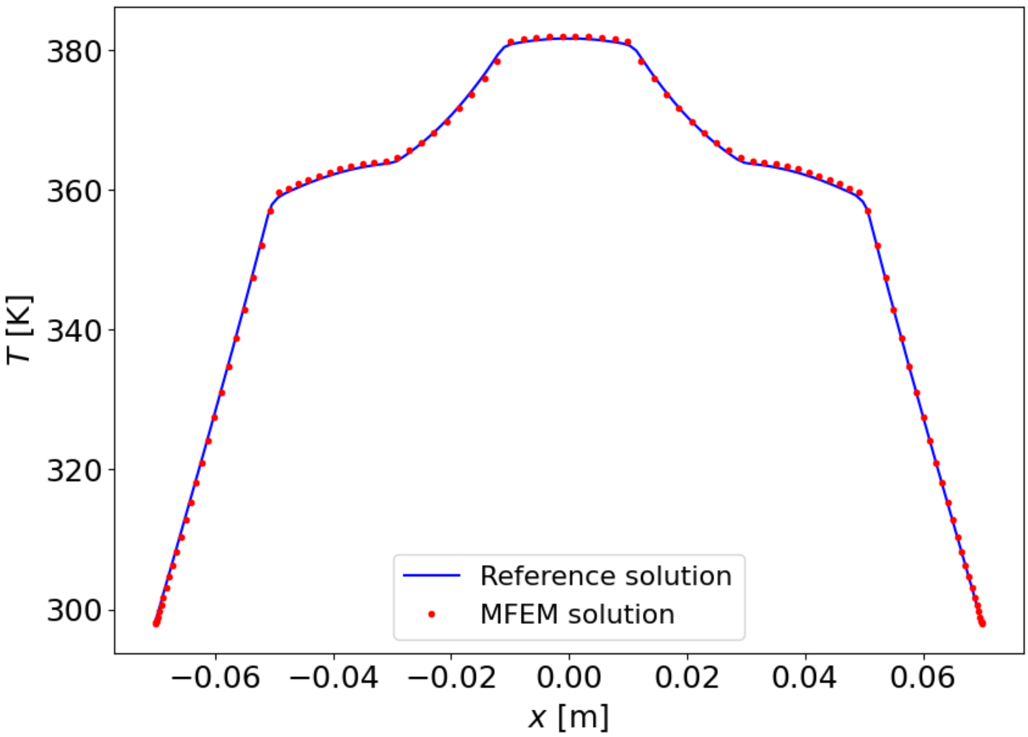


Figure 2.8. Temperature evolution along the central line: MFEM solution compared to the reference solution [6].

3 Aerodynamic problem

The aerodynamic problem analyses the evolution of different fluid variables (e.g. density, velocity or temperature) in a fluid domain. The extremely high velocity of the reentry capsule during the descent generates a hypersonic flow around the body and, also, a shock wave. This shock wave significantly decreases the speed of the air near the body, below $M = 1$, but it increases the pressure and the temperature. This is the main source of thermal loads, that the reentry capsule must withstand during the whole phase.

The CHT (Conjugate Heat Transfer) problem involves the simultaneous resolution of the fluid and the solid domain, exchanging the thermal information through the boundary conditions (see sec. 4). Therefore, an adequate characterisation of the shock wave is needed in order to perform the coupling correctly.

The shock wave simulation can be addressed in many different ways. The simplest approach is to consider the fluid conditions behind a normal shock wave in the whole domain. More complex approaches involve the use of artificial viscosity and internal pressure (or density) sensors to capture the real shock wave while solving the Navier-Stokes equations.

Furthermore, the use of high order DG (Discontinuous Galerkin) methods allows to obtain a high resolution temperature distribution near the capsule, with a relatively coarse mesh, reducing the computational time and obtaining a small numerical error.

The solution of the aerodynamic problem will be obtained by solving the Navier-Stokes equations. In addition, a LES (Large Eddy Simulation) method will be used to model turbulence. All these equations will be solved with the solver HORSES3D (High-ORder (DG) Spectral Element Solver), developed at ETSIAE-UPM [11]. This solver allows to simulate several flows in different regimes and over arbitrary geometries. It is written in Fortran and, hence, a Python interface is needed to perform the coupling with the solver MFEM (see sec. 8).

The resolution of the problem begins with the meshing of the fluid domain, that allows to solve the equations over small elements. This task will be performed with the open-source library Gmsh [20], the same that was used for the solid domain.

Finally, the CFD solver will be validated through the analysis of several test cases, which assure its correct functioning and the accuracy of the results. This way, a more robust coupling can be achieved later and the sources of possible errors are reduced.

3.1 General equations

The aerodynamic problem is based on the resolution of the non-dimensionalised compressible Navier-Stokes equations:

$$\frac{\partial \rho^*}{\partial t^*} + \frac{\partial \rho^* u_j^*}{\partial x_j^*} = 0, \tag{24}$$

$$\frac{\partial \rho^* u_i^*}{\partial t^*} + \frac{\partial \rho^* u_i^* u_j^*}{\partial x_j^*} = -\frac{\partial p^*}{\partial x_i^*} + \frac{1}{Re}\frac{\partial \tau_{ij}^*}{\partial x_j^*} + \frac{1}{Fr^2} g_i^*, \tag{25}$$

$$\frac{\partial \rho^* e^*}{\partial t^*} + \frac{\partial u_j^*(\rho^* e^* + p^*)}{\partial x_j^*} = \frac{1}{Re}\left[\frac{\partial \tau_{ij}^* u_j^*}{\partial x_i^*} + \frac{1}{(\gamma - 1) Pr M_\infty^2}\frac{\partial}{\partial x_j^*}\left(k^* \frac{\partial T^*}{\partial x_j^*}\right)\right], \tag{26}$$

where expression (24) is the continuity equation, expression (25) represents the 3 momentum equations, for $i = 1, 2, 3$ and $j = 1, 2, 3$; and expression (26) is the energy equation. These equations are complemented with the equation of state, considering an ideal gas:

$$p = \rho R T, \tag{27}$$

with $R = 287.15\,\mathrm{J/(kg \cdot K)}$ as the gas constant for the air.

The superscript * represents that the variable is non-dimensional, with t as the time, x as the position vector, ρ as the density, u as the velocity vector, p as the pressure, τ as the viscous stress tensor, g as the gravity acceleration vector, e as the internal energy, k as the thermal conductivity and T as the temperature. The transformation between dimensional and non-dimensional variables has been done as follows:

$$t^* = t u_\infty / L_c, \quad x^* = x / L_c, \quad \rho^* = \rho / \rho_\infty, \quad u^* = u / u_\infty, \quad p^* = p / (\rho_\infty u_\infty^2), \tag{28}$$

$$\tau_{ij}^* = \tau_{ij} L_c / (u_\infty \mu_\infty), \quad g^* = g / g_0, \quad e^* = e(\gamma - 1)/(R T_\infty), \quad k^* = k / k_\infty, \quad T^* = T / T_\infty. \tag{29}$$

Furthermore, the non-dimensional numbers of the Navier-Stokes equations are defined as:

$$Re = \frac{\rho_\infty u_\infty L_c}{\mu_\infty}, \quad M_\infty = \frac{u_\infty}{\sqrt{\gamma R T_\infty}}, \quad Pr = \frac{\gamma R}{\gamma - 1}\frac{\mu_\infty}{k_\infty}, \quad Fr = \frac{u_\infty}{\sqrt{g_0 L_c}}. \tag{30}$$

The subscript $_\infty$ is used for the upstream reference values, g_0 is the sea level gravity acceleration, L_c the characteristic length and γ and R depend on the fluid (for the air, $\gamma = 1.4$ and $R = 287.058\,\mathrm{J/(kg \cdot K)}$).

It is important to notice that the dynamic viscosity, μ varies with the temperature following the *Sutherland's law*:

$$\mu(T) = \mu_{ref} \left(\frac{T}{T_{ref}} \right)^{3/2} \frac{T_{ref} + S}{T + S}, \tag{31}$$

where, in the case of the air, $\mu_{ref} = 1.716 \cdot 10^{-5}\,\mathrm{kg/(m \cdot s)}$, $T_{ref} = 273.15\,\mathrm{K}$ and $S = 110.4\,\mathrm{K}$.

The thermal conductivity, k, also changes with the temperature, as it is shown in Figure 3.1, where the relation between k and T is almost linear for a constant pressure, even for high temperatures. Furthermore, when the temperature is below a value of 3000 K, the thermal conductivity of the air does not significantly change with the pressure.

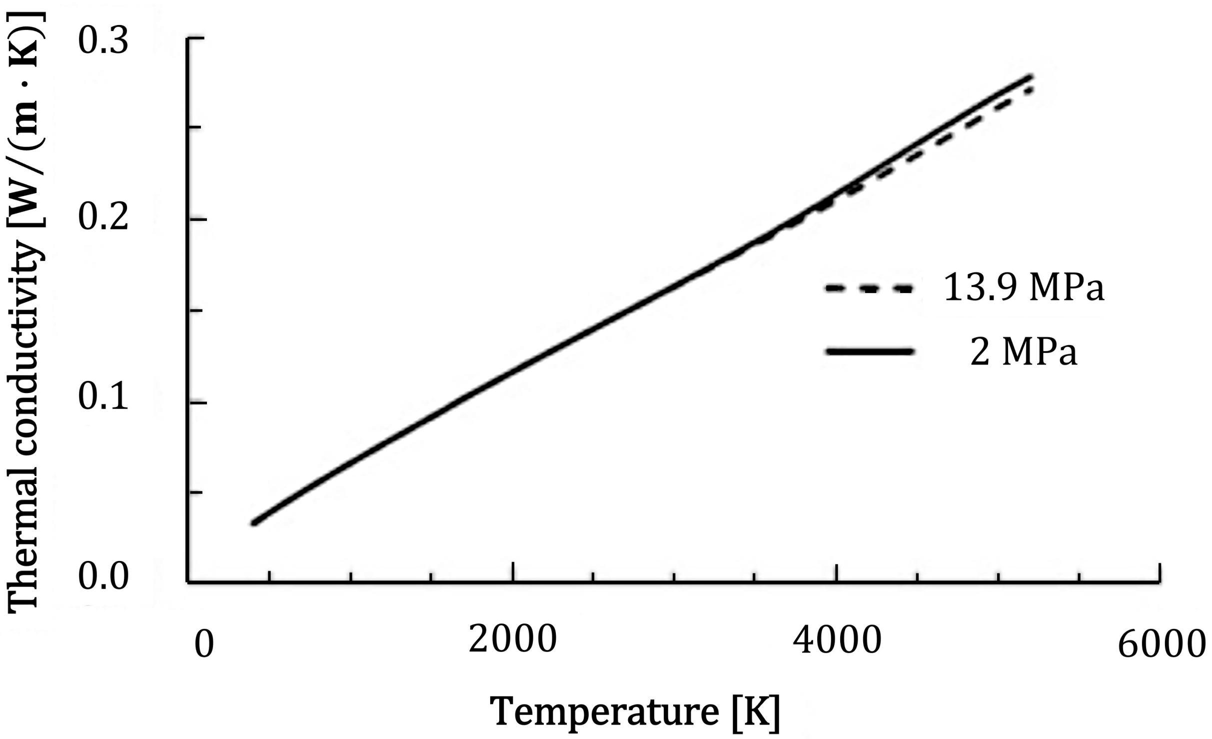


Figure 3.1. Thermal conductivity of the air as a function of the temperature for two different values of the pressure. [7]

Taking into account these considerations, and given the upstream conditions, it is possible to solve numerically the Navier-Stokes equations, using a *Discontinuous Galerkin* spatial discretisation and a temporal integrator. However, it is important to add a turbulence model to achieve a more realistic result. In this case a LES model will be used with this aim, which allows to obtain the effect of turbulent fluctuations in small scales over the flow field. The chosen model to solve the problem with this approach is the *Vreman* model [35].

3.1.1 Discontinuous Galerkin

Discontinuous Galerkin is a finite element method widely used in the resolution of CFD problems in which *convection* plays an important role [36]. Non-linear convective problems may lead to the development of discontinuities in a finite time and, near these discontinuities, the solution of the flow field may be rich and complex. Therefore, the selected method must be able to handle discontinuities, but ensuring an appropriate degree of accuracy to capture the fluid's structure.

The DG method computes the approximated solution in each element of the mesh, which is a tesselation of non-overlapping hexahedral elements (for the 3D case), through a Lagrange polynomial function. In order to perform the interpolation through the Lagrange polynomials, the elements of the mesh have to be transformed from their original shape to a cube, for the case of a 3D problem, which is used as a reference element $E = [-1, 1]^3$. This procedure is shown graphically in Figure 3.2. Once the equations are solved over the reference elements, the inverse transform is applied to recover the original geometry of the problem. The geometrical transformation from the physical to the reference space allows to use curvilinear meshes up to an arbitrary order.

The resolution procedure is based on the *weak formulation* of the integral Navier-Stokes equations, where the integrals are approximated with Gauss quadratures. These quadratures are solved using Legendre-Gauss nodes or Legendre-Gauss-Lobatto nodes to interpolate the Lagrange polynomials. The Gauss points provide an exact integration for polynomials of order $2N + 1$ or lower, while the Gauss-Lobatto points provide an exact solution up to order $2N - 1$ [37], with N as the selected order for the Lagrange polynomials. However, only the Gauss-Lobatto nodes have points on the boundaries of each element; in contrast with the Gauss nodes, which are all of them inside the elements. The existence of some nodes on the boundaries is very useful to perform the coupling between different domains and, hence, Gauss-Lobatto nodes will be used to solve the CHT problem of the capsule in a hypersonic flow.

This approach leads to a discontinuous solution along the interfaces between elements, which is a characteristic feature of DG methods, as it is shown schematically in Figure 3.3. These discontinuities lead to the presence of a *Riemann problem* at the interfaces between elements, which has to be solved to assure a correct coupling in the whole domain.

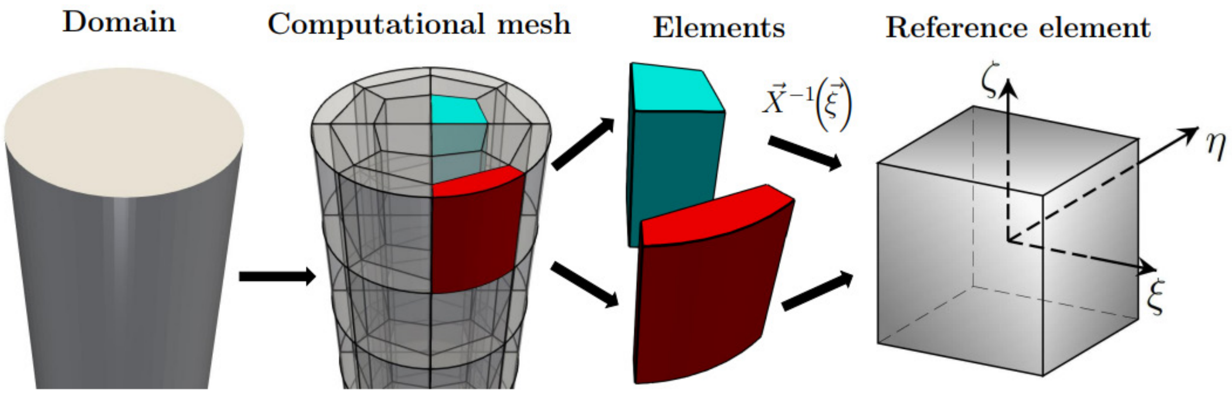


Figure 3.2. Geometrical transformations in the DG method. [8]

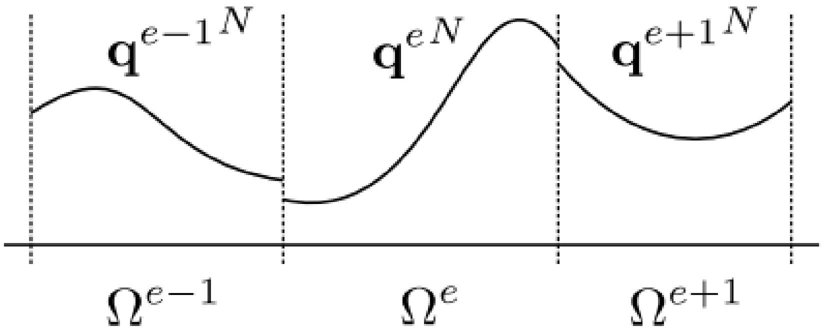


Figure 3.3. Example of a 1D subdivision into finite elements with piece-wise solutions. [9]

The main properties of DG methods are briefly explained bellow [8]:

1. **High-order accuracy:** The approximated solution can be improved by increasing arbitrarily the polynomial order.
2. **Dissipation:** High-order DG sets a low numerical dissipation for well-resolved scales and high numerical dissipation for badly-resolved scales.
3. **Complex geometries:** DG methods allow the use of unstructured meshes[1] with curvilinear elements.

---

[1] The solver HORSES3D only works with hexahedral elements.

4. **Adaptation:** Use of local refinement of the mesh (h-adaptation) and local increasing of the polynomial order (p-adaptation) to increase the accuracy in particular regions.

5. **Robustness:** DG methods can be designed and tailored to be stable for industrial applications.

6. **Parallelization:** The required operations are highly localized and, hence, DG methods can take advantage of memory parallelization.

All these properties make DG methods an adequate approach to the high-order resolution of aerodynamic problems.

3.1.2 Turbulence model

The presence of *turbulence* is usual in many aerodynamic problems and the correct modelling of this phenomenon is of great interest, due to the wide variety of engineering applications in which the effect of turbulence is important.

There are several approaches to model a turbulent flow through computational fluid dynamics. The most straightforward approach is the DNS (Direct Numerical Simulation). This method is based on the numerical resolution of the Navier-Stokes equations without any additional assumption regarding the behaviour of the flow field. However, the use of this methodology requires a very fine mesh to obtain accurate and reliable solutions, due to the characteristic length of the smallest turbulent eddies, which dissipate a significant amount of energy, is close to the *Kolmogorov scale*.

The resolution of these small scales is unaffordable in most cases, as the number of nodes in the mesh should be proportional to the 9/4 power of the Reynolds number to resolve all scales of motion [38], which would require large computational resources for several real applications.

An alternative to DNS is the use of a RANS (Reynolds Averaged Navier-Stokes) model, which allows to compute the evolution of the mean quantities of the flow. However, RANS models have some deficiencies when dealing with flows with large streamline curvatures and time dependant characteristics [39]. Furthermore, these models are usually tuned by steady state mean flow data, which leads to a detriment to the accuracy for unsteady problems. In these cases, there are other more realistic options which allow to obtain a better resolution of the flow field: the LES (Large Eddy Simulation) models.

LES models represent an intermediate point between the direct simulation of turbulent flows and the solution of the Reynolds-averaged equations [38]. In LES, the large fluid structures are resolved exactly, and only the effect of small turbulent scales is modelled. The main advantage of this approach is that, usually, smaller scales are more homogeneous and universal, and the effect of the boundary conditions is also less significant over these structures. Therefore, LES models can be easily generalised and they can be tuned with few parameters to be adapted to different flows.

LES models are based on the distinction between *large scales* (resolved scales) and *sub-grid scales*, through the use of a filter. This filter is formed by a *kernel* or filter function (e.g. Gaussian filter) and a *window length*, which determines the spatial range of the kernel. Depending on the selected window length, more or less wavelengths are completely resolved and, hence, this length is related to the mesh and the polynomial order used by the solver.

If this filter is applied over the Navier-Stokes equations, it is possible to obtain the filtered equations of motion, which govern the evolution of large scales [38]. Sub-grid scales are removed by the filter and they have to be modelled to mimic the energy dissipation associated to small eddies.

In this case, the LES model proposed by Vreman [35] will be used. As it is explained by Vreman, it is an eddy-viscosity model expressed in first-order derivatives and it does not require the use of explicit filtering, averaging or clipping procedures and, furthermore, it is rotationally invariant if the filter width (window length) is isotropic. In addition to all these desirable properties, this model is capable of reducing the eddy viscosity in laminar and transitional regions, which leads to accurate solutions for inhomogeneous flows, in contrast with other simpler models like the well-known *Smagorinsky* model.

Eddy viscosity models approximate the turbulent stress tensor τ_{ij}:

$$\tau_{ij} = -2\nu_e S_{ij} + \tau_{kk}\delta_{ij}/3, \tag{32}$$

where $S_{ij} = \frac{1}{2}(\partial_i \bar{u}_j + \partial_j \bar{u}_i)$ is the large scale strain-rate stress tensor, with $\bar{u}$ as the filtered velocity, and ν_e is the eddy viscosity.

Vreman models the eddy viscosity as follows [35]:

$$\nu_e = c\sqrt{\frac{B_\beta}{\alpha_{ij}\alpha_{ij}}}, \tag{33}$$

where c is a constant related to the Smagorinsky constant C_s by $c \approx 2.5C_s$ and $\alpha_{ij} = \partial_i \bar{u}_j$.

The scalar B_β is an invariant of the matrix β:

$$\beta_{ij} = \Delta_m^2 \alpha_{mi}\alpha_{mj}, \qquad B_\beta = \beta_{11}\beta_{22} - \beta_{12}^2 + \beta_{11}\beta_{33} - \beta_{13}^2 + \beta_{22}\beta_{33} - \beta_{23}^2, \tag{34}$$

with Δ_m as the filter width following the axis m.

3.1.3 Shock capturing

The shock capturing functionality has been recently included in HORSES3D, which allows to perform the direct simulation of shock waves. Although Discontinuous Galerkin methods are capable of handling discontinuities, the modelling of strong shock waves, generated in presence of hypersonic flows, may require a very fine mesh and a small time step to achieve a convergence of the method. Therefore, this new functionality can be added to improve the performance of the solver in this case.

The *shock capturing* module helps to stabilise cases with discontinuous solutions [11]. This module is based on a sensor which detects problematic regions, where the stabilisation algorithm should be applied. The sensor discriminates between three different regions depending on the sensor value, s, which is mapped into the interval $a \in [0, 1]$:

$$a = \begin{cases} 0 & \text{if } s \le s_0 - \Delta s/2, \\ \frac{1}{2}\left[1 + \sin\left(\frac{s - s_0}{\Delta s}\right)\right] & \text{if } s_0 - \Delta s/2 < s < s_0 + \Delta s/2, \\ 1 & \text{otherwise,} \end{cases} \tag{35}$$

with $s_0 = (s_1 + s_2)/2$ and $\Delta s = s_2 - s_1$, where s_1 and s_2 are user-defined sensor thresholds, which have to be tuned manually for each particular application. The value of the sensor, s, can be selected to capture different regions. For this analysis, the *density gradient* is used as a sensor value. This module introduces an artificial viscosity, defined by the user, in the middle and the upper region of the sensor (if $a = 0$ no artificial viscosity is required) to improve the stability of the problem.

3.2 Geometry and physical properties

Once the selected approach to solve the aerodynamic problem has been defined, it is necessary to define the geometrical and physical properties of the hypersonic flow; which, together with the reentry capsule (see sec. 2.2), completely define the CHT problem.

As it was explained in previous sections, the selected capsule for the present analysis is the Apollo Command Module, which represents a challenging engineering task from the point of view of the thermal loads that have to be withstood. The meshing process has been performed to capture the fluid around the geometry of this capsule, as it is shown in Figure 3.4. The whole mesh is a circle, with a radius of 30 m, centred on the capsule. The location of the capsule in the centre of the mesh allows to simulate the flow field for different angles of attack without modifying the mesh.

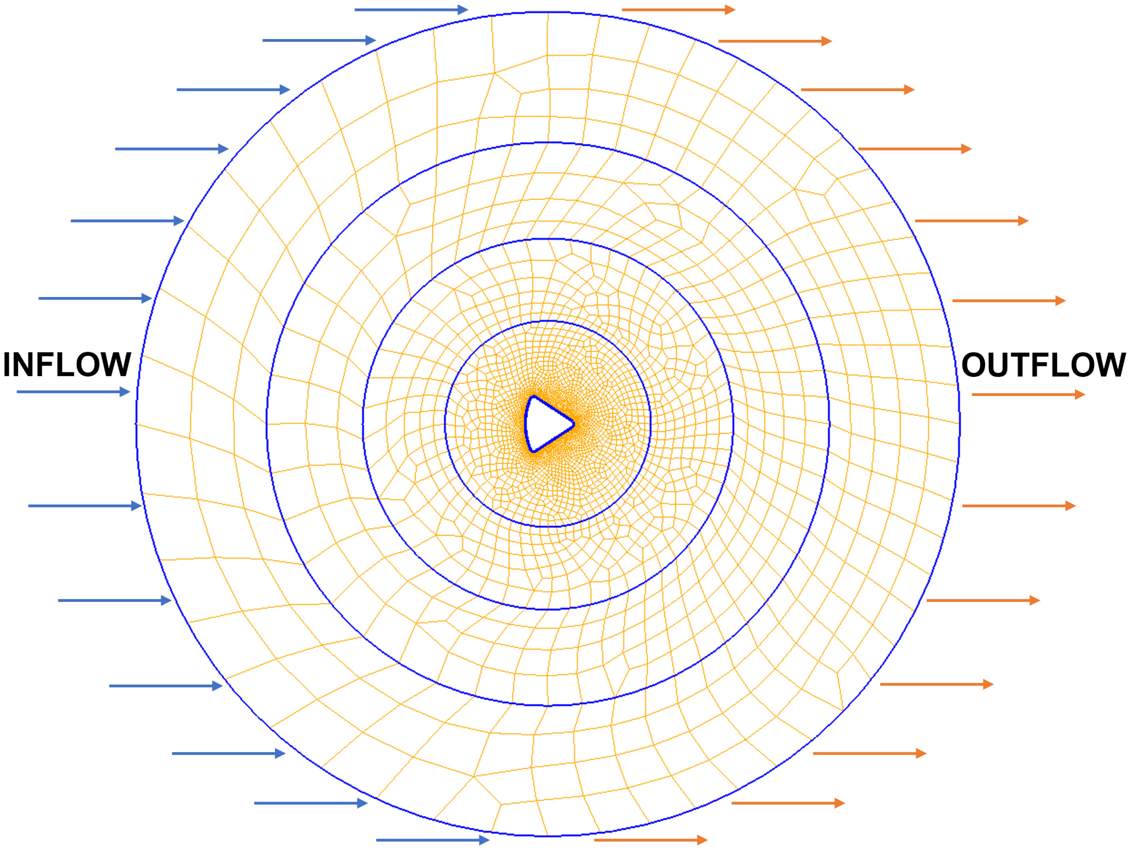


Figure 3.4. Mesh for the fluid domain around the reentry capsule.

As it can be appreciated in the previous figure, the elements become smaller as they get closer to the body. Furthermore, downstream elements are also smaller than upstream elements, due to the turbulence is expected to be more significant in the first region.

Near the body, a boundary layer has been created through a mesh refinement process. The height of each layer near the body increases with the normal distance to the boundary, following a geometrical progression. This meshed boundary layer is represented in the detail of Figure 3.5.

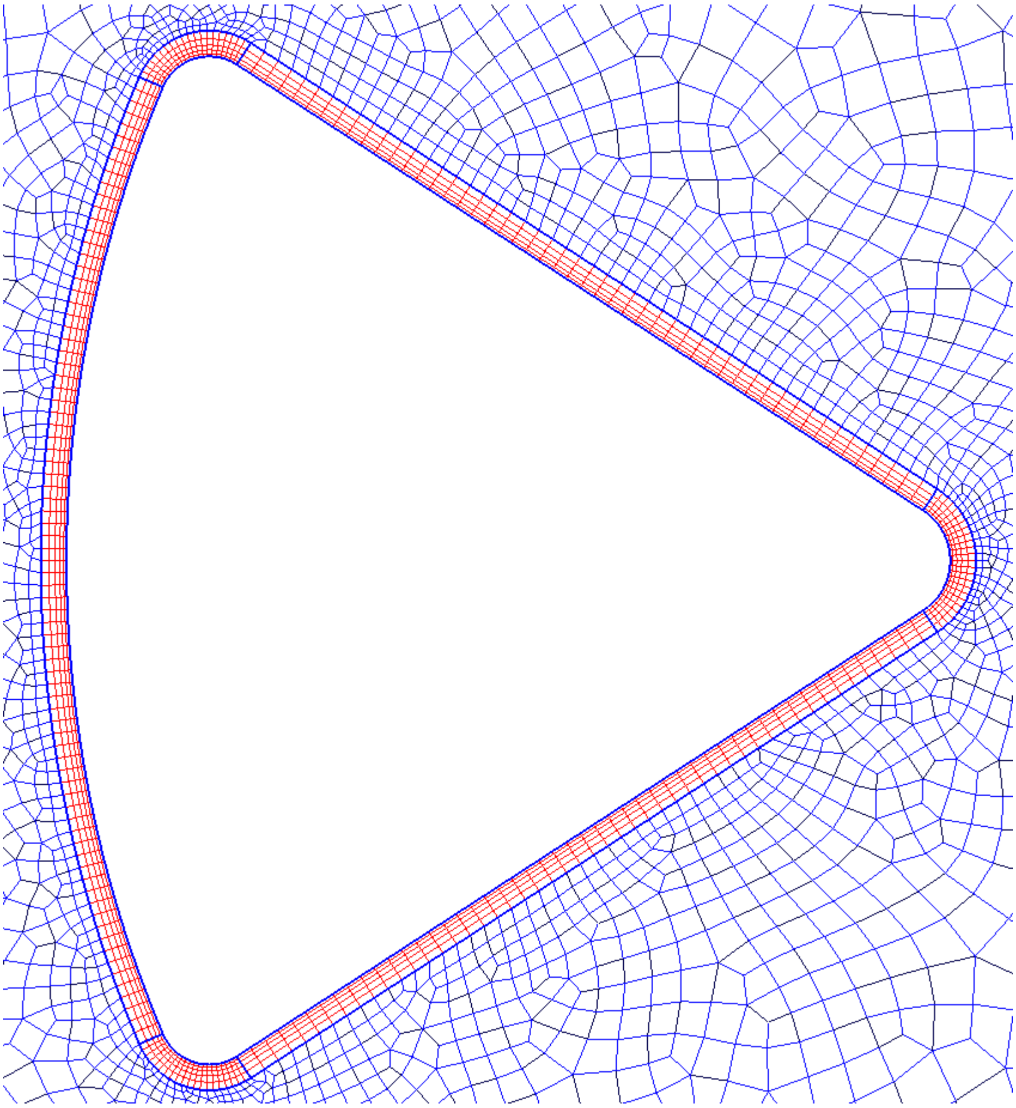


Figure 3.5. Detail of the meshed boundary layer around the reentry capsule.

Although this problem is thought to be analysed as a two dimensional problem, the turbulence can only be captured correctly in a three dimensional domain. However, the resolution of a full 3D capsule, and the associated mesh, would require an incredibly large computational time, specially if the CHT problem has to be performed simultaneously.

Therefore, the selected approach is a middle point between the 2D problem and the 3D problem. The 2D mesh, which has been described previously, has been extruded with 5 layers of hexahedral elements following the third axis, z, to obtain a three dimensional mesh. Then, over the top and the bottom faces of the extruded mesh, *periodic boundary conditions* have been defined. This way, the problem is similar to that of a 2D flow field extruded infinitely, which allows the coupling with a 2D solid capsule in a middle section and, also, the three dimensional nature of the turbulent flow is preserved with this approach.

The complete three dimensional mesh contains 40426 hexahedral non-overlapping elements and 223719 nodes. It is important to mention that this mesh has been generated using second order *curvilinear elements* to increase the accuracy, specially near the body. However, the order of the curvilinear elements and, hence, the number of nodes, will be modified to fit the selected polynomial order of each analysis.

Once the mesh has been completely defined, the conditions at the inflow have to be established to close the aerodynamic problem. The free stream conditions are selected from the descent trajectory of the Apollo reentry vehicle, which is represented in Figure 3.6.

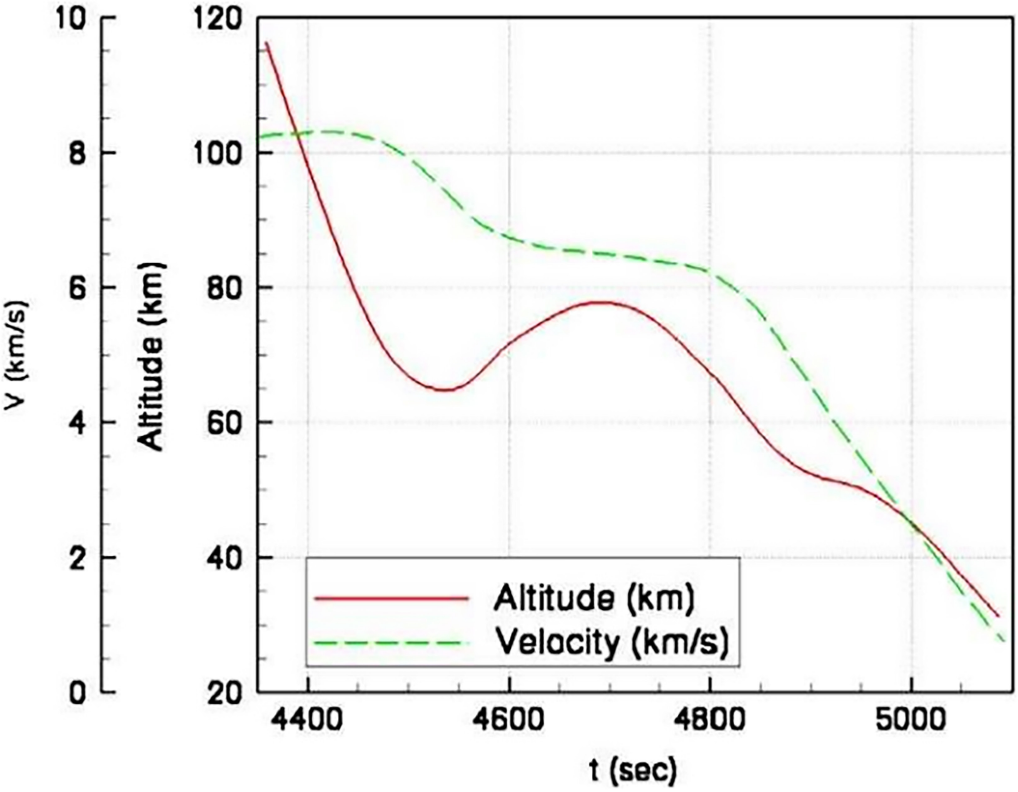


Figure 3.6. Altitude and velocity as a function of time from launch for AS-202 mission. [3]

In this case, the analysis will be focused on a concrete instant, $t = 4875\,\mathrm{s}$, of the descent, when the altitude is equal to 54.6 km and the velocity has a value of 5.07 km/s. From this values and using the ISA atmosphere equations (see sec. 1.6.1) it is possible to obtain a very good estimation of the free stream conditions. In Table 3.1, ISA results are compared with the real values measured during the mission AS-202. As it was expected, both values are very similar, which demonstrates that the ISA model, even though it is very simple, offers accurate results regarding the atmospheric conditions.

Table 3.1. Free stream conditions: ISA estimation and real measured values [3] in mission AS-202.

	Temperature [K]	**Density** [kg/m^3]	**Pressure** [Pa]
ISA	261	$5.6 \cdot 10^{-4}$	42
Measured	262	$6.16 \cdot 10^{-4}$	46

Although these are the real conditions at the inflow, the capsule is surrounded by a very different flow field due to the effect of the *shock wave*. For the given temperature and velocity values, the Mach number is $M = 15.6$, which is a very high value, even for hypersonic flights.

The CFD solver, HORSES3D, has a shock capturing functionality, as it was explained previously in sec. 3.1.3 , but it has been recently developed and it is not mature enough to accurately model these extreme conditions with very high Mach numbers, together with the coupling between fluid and solid domains and preserving the stability of the whole problem. Therefore, the selected approach to solve the CHT problem is to consider the after-shock conditions, considering a normal shock wave, as the free stream values. This way, even though the solution will not be exact, it is possible to obtain a very good approximation considering the software limitations and the available time.

3.2.1 Inflow conditions: Normal shock wave

Known the upstream conditions before the normal shock wave, the variables after the shock are given by the resolution of the following equations:

$$\rho_1 u_1 = \rho_2 u_2, \tag{36}$$

$$p_1 + \rho_1 u_1^2 = p_2 + \rho_2 u_2^2, \tag{37}$$

$$h_1 + \frac{u_1^2}{2} = h_2 + \frac{u_2^2}{2}, \tag{38}$$

which can be solved in conjunction with two additional expressions:

$$h_2 = C_p T_2, \tag{39}$$

$$p_2 = \rho_2 R T_2, \tag{40}$$

where ρ is the density, p is the pressure, h is the enthalpy, u is the velocity, C_p is the specific heat at constant pressure and R is the gas constant.

The previous system of equations is formed by 5 equations and 5 unknown variables. The system's solution links the upstream variables (subscript 1) with the downstream variables (subscript 2) and it can be obtained analytically:

$$M_2^2 = \frac{1 + \frac{\gamma-1}{2} M_1^2}{\gamma M_1^2 - \frac{\gamma-1}{2}}, \tag{41}$$

$$\frac{\rho_2}{\rho_1} = \frac{u_1}{u_2} = \frac{(\gamma+1)M_1^2}{2 + (\gamma-1)M_1^2}, \tag{42}$$

$$\frac{p_2}{p_1} = 1 + \frac{2\gamma}{\gamma+1}(M_1^2 - 1), \tag{43}$$

$$\frac{T_2}{T_1} = \frac{h_2}{h_1} = \frac{p_2}{p_1}\frac{\rho_1}{\rho_2}. \tag{44}$$

For the free stream conditions of the Apollo capsule, the after shock conditions that will be used for the present CHT analysis are compiled in Table 3.2. For the Mach computation, it has been considered $\gamma = 1.4$ and $R = 287.15\,\mathrm{J/(kg \cdot K)}$; and for the Reynolds computation, $L_c = 1\,\mathrm{m}$ and $\mu = 10^{-5}\,\mathrm{Pa \cdot s}$. The inflow temperature has a value of 12645 K, which is a huge value and the ideal gas hypothesis might not be accurate enough in this situation. However, the CFD solver does not provide the option of simulating a chemically reactive flow and the implementation of this new functionality exceeds the scope of this analysis.

Table 3.2. Free stream conditions after the shock wave for the Apollo capsule analysis.

Velocity [m/s]	**Temperature** [K]	**Density** [kg/m^3]	**Pressure** [Pa]	**M**	**Re**
862.4	12645	$3.62 \cdot 10^{-3}$	13151	0.382	312330

3.2.2 Inflow conditions: Shock capturing

The complete aerodynamic problem will be solved to test the capability of the CFD solver to capture shock waves. For this analysis, the CHT problem will not be considered and the reentry vehicle will be simulated through an adiabatic boundary condition.

Even with this simplification, for the inflow conditions previously defined, the Mach number has a value of $M = 15.6$, which is very high. As this is the first hypersonic simulation performed with the solver HORSES3D, another point of the Apollo descent trajectory (see Figure 3.6) will be selected in order to reduce the Mach number and improve the convergence and the stability of the problem.

In this case, the instant $t = 5000\,\mathrm{s}$, has been selected, when the altitude is equal to 45 km and the velocity has a value of 2500 m/s. The inflow conditions, which have been obtained using the ISA atmosphere equations (see sec. 1.6.1), are compiled in Table 3.3. These new conditions involve a Mach number with a value of $M = 7.66$, which is still hypersonic, but less severe than the previous condition and, hence, it will be easier to obtain a stable solution for the problem.

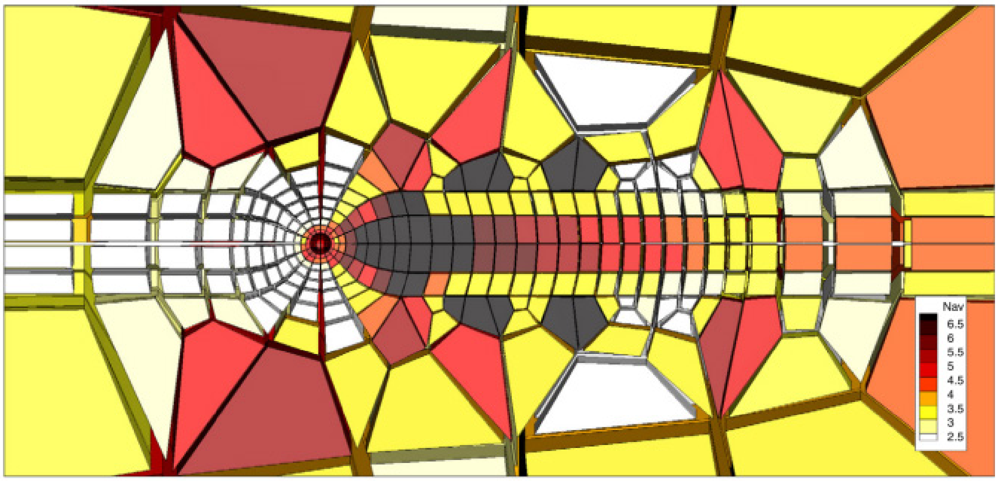
Table 3.3. Free stream conditions at $t = 5000\,\mathrm{s}$, for the Apollo capsule analysis.

Velocity [m/s]	**Temperature** [K]	**Density** [kg/m^3]	**Pressure** [Pa]	**M**	**Re**
2500	265.1	$1.88 \cdot 10^{-3}$	143	7.66	277450

3.3 CFD validation: test cases

HORSES3D has been developed to solve a wide variety of flows: compressible, incompressible, multiphase and turbulent flows, using different available techniques. Therefore, there are several published papers regarding the different functionalities of this solver, which can be used as a validation tool for the present work; avoiding the need to run additional test cases which would require a significant amount of time to be correctly simulated.

This solver provides *high-order* methods for the resolution of the Navier-Stokes equations. Furthermore, it is possible to use different polynomial orders in different regions of the mesh, which allow to obtain more accurate results only where needed. A truncation error adaptation algorithm [10] automatically discriminates among different regions and performs the mesh p-adaptation in steady or unsteady flows. An example of this functionality is represented in Figure 3.7.



(a) Average polynomial order (N_{av}).

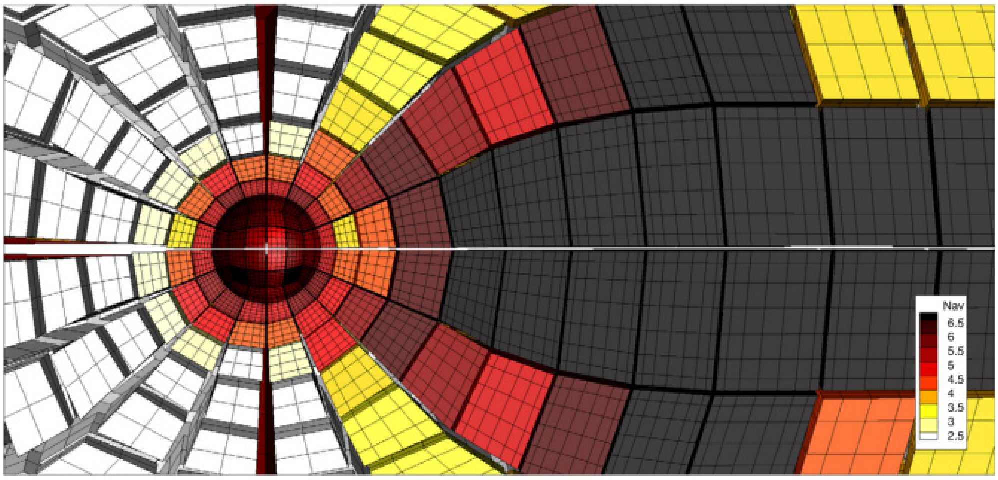
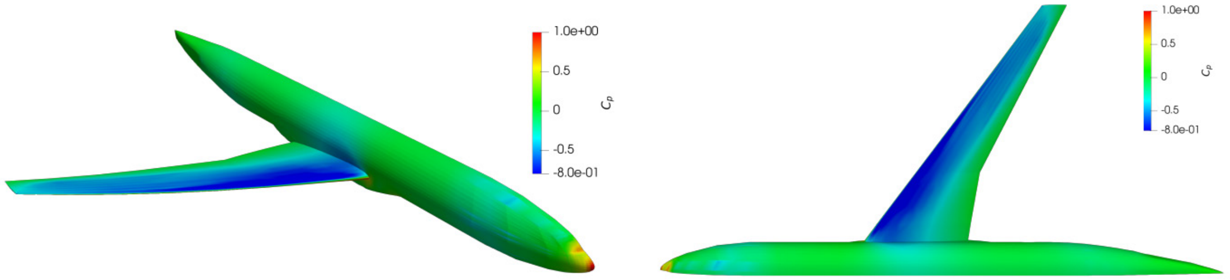


Figure 3.7. Example of a p-adapted mesh for the flow around a sphere at Reynolds 200. The contours indicate the average polynomial order (Nav=(N1 +N2 +N3)/3). [10]

The computation of the flow around a reentry capsule requires an adequate model for the turbulence. HORSES3D provides RANS and LES models for the modelling of turbulent flows.

The RANS solver uses the negative part of the well-known model Spalart-Allmaras [11]. An example of this functionality, successfully used to solve the flow field around an aircraft, is shown in Figure 3.8.

The LES solver provides different models [11], including Smagorinsky, Wale and Vreman. Also ILES (Implicit Large Eddy Simulation) has been implemented in HORSES3D. This approach uses the numerical dissipation to model the turbulent effects of unresolved scales. An example of an ILES simulation is represented in Figure 3.9, where the lift and drag coefficients have been obtained for a NACA 0012 airfoil at high Reynolds numbers.



(a) Isometric view. (b) Top view.

Figure 3.8. Pressure contour over the surface of the Common Research Model (CRM) for $M = 0.85$, $Re = 5 \cdot 10^6$ and $AoA = 2.3°$ using the HORSES3D RANS module with split-form inviscid fluxes. [11]

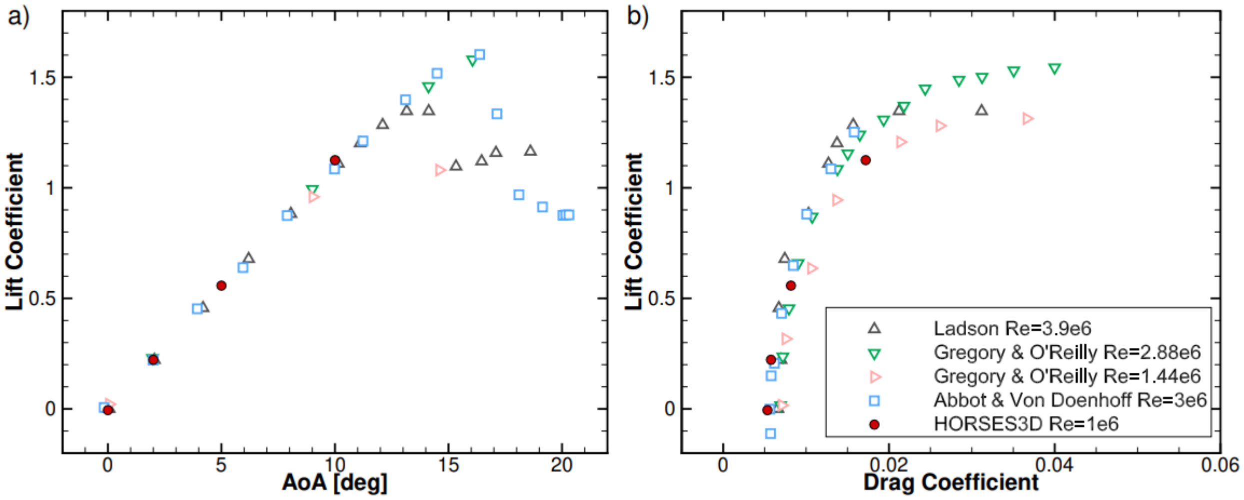


Figure 3.9. Lift and drag for a NACA 0012 at $Re = 10^6$; comparisons between HORSES3D (ILES version) with experimental data. [11]

Finally, the latest version of HORSES3D provides a *shock capturing* functionality. This approach includes an additional artificial viscosity to handle shocks [11]. In order to increase the viscosity near critical regions only, a sensor is defined (e.g. based on the density gradient) to detect the location of shock waves. This functionality has been tested in a flow at Mach 3, which is shown in Figure 3.10. However, it has never been used to model hypersonic flows yet and it is possible that the strong shock wave generated by the Apollo capsule can not be correctly captured.

All these tests and different functionalities have demonstrated that HORSES3D is a robust and accurate solver and, hence, it is adequate to be used for the simulation of a reentry capsule in a hypersonic flow.

Figure 3.10. Facing forward step case at $\tau = 10$ with an incoming flow at Mach 3 and 3653 elements with polynomial degree $N = 7$. [11]

4 Conjugate Heat Transfer

The *fluid-structure interaction* applied to the analysis of a reentry capsule in a hypersonic flow is defined as a *Conjugate Heat Transfer* problem. CHT problems are based on the exchange of the temperature and heat fluxes between a fluid domain and a solid domain. This coupling, which is performed through the *boundary conditions*, allows to obtain the thermal effect of each domain on the other. This way, the whole energy of the system is shared and it is necessary to solve the solid heat equation along with the fluid energy equation [40]. A graphical example of a CHT problem is shown in Figure 4.1.

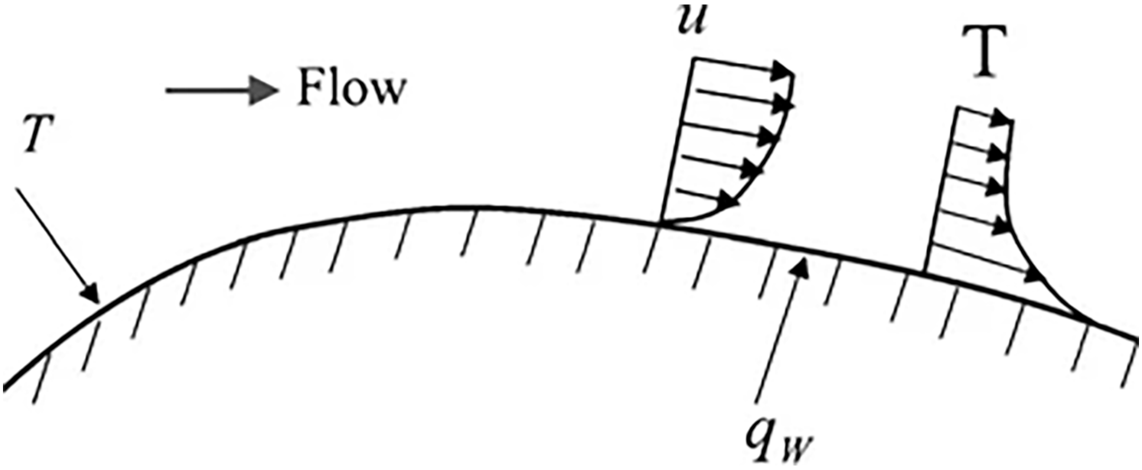


Figure 4.1. Example of conjugate heat transfer: convective heat transfer to external fluid flow from a solid surface with internal heat conduction. [12]

Conjugate heat transfer problems are an important field of analysis in aerospace engineering [41], where the aerodynamic loads are coupled with the thermal loads and even with the structural response of the aerospace vehicle. There have been numerous researches in this field [42, 43, 44, 45, 46, 47]; some of them centred on a hypersonic flow [48, 49]. In the area of reentry capsules, there are several analyses of the hypersonic flow field, considering the capsule as adiabatic [50], isothermal [51, 3] or catalytic [52]; using a model of ablative material on the surface [53]; or even analysing the fluid domain and the solid domain separately [54]. However, there are very few researches based on the complete CHT analysis of a reentry capsule in a hypersonic flow [55], which is the main objective of this project.

There are several ways to solve CHT problems. Simple 1D problems have a simple analytical solution at the fluid-solid interface [41]; however, multi-dimensional problems have to be modelled and solved numerically. A widely used method is the *monolithic* approach, also called *strong coupling*, in which solid and fluid equations are solved simultaneously, with a single solver. This approach improves stability, due to the continuity in heat fluxes and temperature is ensured at the interfaces, and it is the desired solution in many cases. However, for real applications, it is usual to use specific CFD solvers in the fluid domain and a different FEM solver in the solid domain, which is known as the *partitioned* approach or *weak coupling.* The use of this method provides more flexibility, but some stability problems may show up (see sec. 4.2). This approach is common due to most of commercial solvers are centred on a single physical problem and, furthermore, the complexity of the monolithic approach leads, in general, to a considerable increment of the computational time [41].

Therefore, in this research the CHT problem will be solved with a partitioned approach and two different solvers: HORSES3D as the CFD solver for the fluid and MFEM as the FEM solver for the solid. The CFD solver uses a DG (Discontinuous Galerkin) method, which provides stable solutions in convection dominated problems, while the FEM solver for the solid uses a CG (Continuous Galerkin) method, where no stabilisation is required, due to the smoothness of the solution, for the *heat equation* [56].

In this case, the fluid-structure interaction between the reentry capsule and the hypersonic flow will be focused on the quasi steady-state solution, as a perfect steady-state is not possible to be obtained in presence of turbulence. The quasi steady-state problem allows to use a weak coupling that significantly reduces the computational time required to obtain the solution.

The thermal analysis of a reentry capsule in a hypersonic flow is a complex problem and the difficulty of the analysed problem will be increased progressively. First, a test case of a thermal boundary layer will be simulated to validate the proposed approach to the CHT problem. Then, different analyses can be carried out to analyse the range of application of the proposed methodology: from simple cases, with an adiabatic or isothermal BC to model the capsule, to more complex CHT problems, with a conductive and/or radiative TPS (see sec. 5).

Finally, the whole coupling between both solvers will be performed through a Python interface, which is needed to adjust the information exchange from HORSES3D, which is written in Fortran, to MFEM, which is written in C++ (see sec. 8).

4.1 Conjugate Heat Transfer approach

Partitioned Conjugate Heat Transfer problems, in which a weak coupling is established between the fluid and the solid, can be classified in relation to how the information is exchanged from one domain to the other: FFTB (Flux Forward Temperature Back), TFFB (Temperature Forward Flux Back), hFTB (heat transfer coefficient Forward Temperature Back) and hFFB (heat transfer coefficient Forward Flux Back) [13]. The difference among these methods is the way the thermal information (temperature and heat fluxes) is sent from the fluid domain to the solid domain and back, through the boundary conditions. The general idea behind these four methods is shown in Figure 4.2.

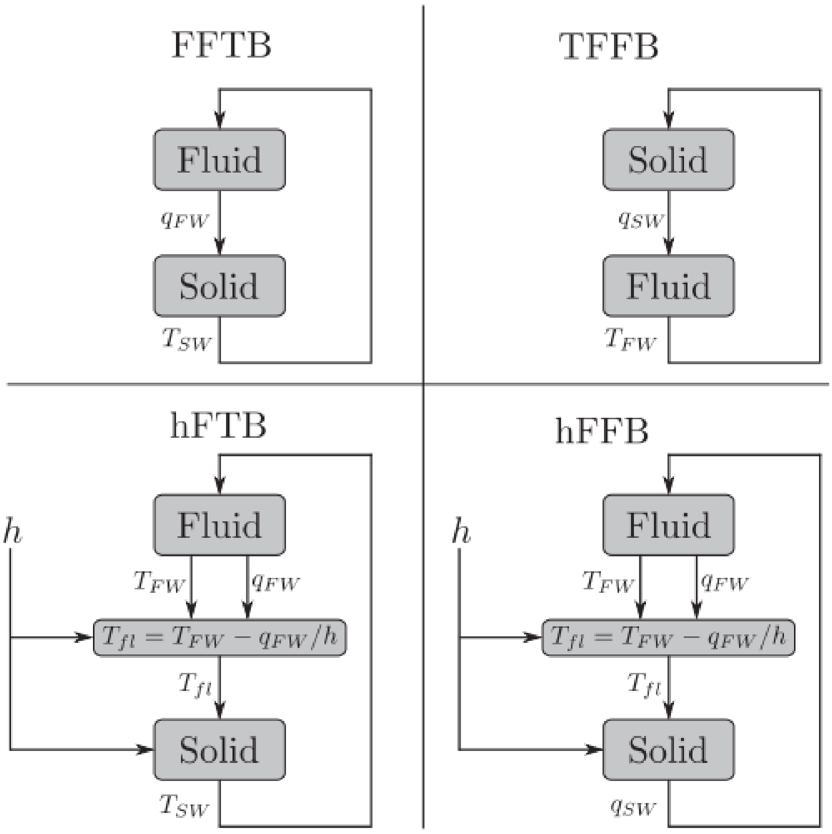


Figure 4.2. Overview of different coupling methods. [13]

In each of these methods, the selected approach is based on solving the steady-state heat equation in the solid (*Laplace equation* if a constant conductivity is considered) once each N_s iterations of the CFD solver in the fluid domain. This way, the boundary conditions of the fluid are updated only after several time steps, which drastically reduces the required computational time to converge the complete system. However, the value of N_s has to be tailored for each application, due to if N_s is increased too much, the convergence of the system may deteriorate, while if N_s is very low, the solution may oscillate continuously because the fluid requires some time to assimilate the new BC [13].

4.1.1 Flux Forward Temperature Back (FFTB)

In this method, each N_s iterations of the CFD solver, the heat fluxes are computed on the BC of the fluid domain. These fluxes are used as Neumann BC to solve the heat equation in the whole solid domain. Then, the temperature obtained on the interface of the solid is used as the new BC to update the fluid domain.

This process is repeated several times until convergence is achieved; that is, the temperature and heat fluxes are the same in both domains and, hence, the temperature remains the same in two successive updates.

Although this methodology may be interesting for some applications, it is not suitable from a practical point of view because, if the solid is completely immersed in the fluid, the Laplace equation for the solid would have only Neumann boundary conditions, which implies that there would be several possible solutions and the problem would be ill-posed. Therefore, this method has been discarded to deal with the CHT problem of the reentry capsule in a hypersonic flow.

4.1.2 Temperature Forward Flux Back (TFFB)

In this method, each N_s iterations of the CFD solver, the temperature distribution of the fluid at the interface is used as Dirichlet BC to solve the heat equation within the solid domain. In this case, there is only one solution for the Laplace equation and the temperature is well defined in the whole domain. Then, the heat fluxes are computed on the solid's boundaries and they are used as the new BC to update the fluid solution. It is important to notice that these heat fluxes have a contribution from the conductive fluxes, obtained from the solution of the heat equation (14), and a contribution from the radiative fluxes (21), which represent the heat that is radiated to space.

This process is iterative and it has to be repeated until the temperature and heat fluxes in both domains are equal and, hence, convergence has been achieved. This is the most common approach and it will be used to solve the CHT problem of the reentry capsule in this research.

4.1.3 Heat transfer coefficient Forward Temperature Back (hFTB)

In this method, each N_s iterations of the CFD solver, the heat transfer coefficient, h, is computed from the solution in the fluid domain using the following expression:

$$q_w = h(T_w - T_f), \tag{45}$$

where q_w and T_w are the heat flux and the temperature on the BC, respectively, and T_f is the ambient fluid temperature, far from the boundary layer.

Once the value of h is known, the same equation (45) is used as a Robin BC for the solid domain, taking into account that:

$$q_w = -k_s \frac{\partial T_w}{\partial n}, \tag{46}$$

with k_s as the solid's thermal conductivity.

The unknowns, q_w and T_w, are obtained as part of the solution for the heat equation within the solid domain. Then, the new T_w distribution on the boundaries is sent back to the CFD solver to be used as the new BC in the fluid domain.

Following the same principle as previous methods, these steps must be repeated each N_s iterations until the temperature and heat fluxes change no more.

The main problem of this approach is the need of knowing the temperature T_f far from the boundary layer, which is not possible in every situation. Furthermore, a positive value of h must be assured because, if h falls below zero, it would make the conduction problem ill-posed [13].

These problems, together with the additional complexity in the resolution of the Laplace equation with Robin BC, are the reason why this approach has been discarded.

4.1.4 Heat transfer coefficient Forward Flux Back (hFFB)

In this method, each N_s iterations of the CFD solver, the heat transfer coefficient, h, is computed from the solution in the fluid domain, using the same methodology as in the previous method hFTB. The only difference between hFFB and hFTB is that, in the case of hFFB, once the solid domain has been solved with Robin BC, the heat fluxes, q_w, are used to update the fluid as Neumann BC.

The main advantage of hFFB over hFTB is that, in this case, the BC for the fluid are Neumann BC, which means that the boundary conditions only affect the value of the derivatives and, hence, the fluid's temperature will be modified progressively and the convergence may be improved.

However, the same problems regarding the temperature T_f and the h value are also present in this case and, therefore, it is not the optimal approach for the chosen reentry analysis.

4.2 Conjugate Heat Transfer stability

The CHT stability depends on the selected approach to solve the thermal problem. In this case, a stability analysis will be performed based on the method TFFB (Temperature Forward Flux Back), which is the selected methodology in this case. The basis of the stability problem are compiled in [13].

The TFFB method is initialised by guessing a BC for the fluid, as it could be an adiabatic wall. With this condition, N_s iterations are performed in the CFD solver and a wall temperature, T_w^0, is obtained on the boundaries. Let α_f be a parameter defined as the difference between the estimated temperature, T_w^0, and the stable temperature once the problem has converged, T_w:

$$T_w^0 = T_w + \alpha_f. \tag{47}$$

Then, T_w^0 is used as a Dirichlet BC in the solid domain and the heat equation is solved. The heat flux on the boundaries is obtained as follows:

$$q_w^0 \approx -\frac{k_s}{L}\Delta T = q_w - \frac{k_s}{L}\alpha_f, \tag{48}$$

where k_s is the solid's thermal conductivity, L is a characteristic length, q_w is the steady-state value of the heat flux and ΔT is the difference of temperature between two points, separated a distance L, following the normal direction from the boundary.

The heat flux q_w^0 is used to update the BC of the fluid domain. Using the previous expression (45), the new equation for the boundary can be obtained:

$$T_w^1 = T_f + \frac{q_w^0}{h} = T_f + \frac{q_w}{h} - \frac{k_s}{hL}\alpha_f = T_w - \frac{\alpha_f}{Bi}, \tag{49}$$

with T_f as the fluid temperature far from the boundary layer, h as the heat transfer coefficient of the fluid, $Bi = hL/k_s$ as the Biot number and T_w^1 as the new estimation for the wall temperature. If this same coupling analysis is repeated n times, the previous expression (49) can be rewritten:

$$T_w^n = T_w - \frac{\alpha_f}{Bi^n}. \tag{50}$$

Therefore, the temperature T_w^n will converge to T_w only if $|Bi| > 1$. This criterion will be fulfilled if the convective heat transfer in the fluid is greater than the conductive heat transfer in the solid. Although this stability condition should not be a problem for the reentry analysis, there can be small regions where the local Biot number falls below the unit and, hence, it is necessary to apply different techniques in order to stabilise the coupling method. Among the existing possible techniques, the following explanation will be focused on *temporal filters* and *spatial filters.*

4.2.1 Temporal filter

The temporal filter, also known as *relaxation coefficient* method, allows to reduce the change in the temperature of the fluid on the boundary after each coupling with the solid domain. This way, the stability of the problem can be improved, even if the Biot number is not appropriate.

In this methodology, a relaxation coefficient β is defined and the filter is applied over the wall temperature or the heat fluxes, depending on the selected CHT approach. For the present analysis, a TFFB approach is used and, hence, the temporal filter will be applied over the heat flux, which is the input for the fluid domain.

The application of the temporal filter over the heat flux, q_w, that sets the fluid's BC, is performed as follows:

$$q_w^i = \beta q_w^{i-1} + (1-\beta) q_w^i, \tag{51}$$

with $0 < \beta < 1$.

Therefore, the new value of the heat flux is obtained as a weighted sum of the present and the previous heat flux. The greater the value of β, the slower the convergence; however, the stability also increases with β and a trade-off has to be achieved between speed and stability.

Although this method is widely used to improve stability [13, 57]; the use of a relaxation coefficient leads to a slower convergence, as the heat fluxes are updated progressively. Furthermore, the final heat fluxes, after many iterations, only converge to the real ones for steady-state analyses. However, in presence of turbulence, the flow field is not perfectly steady and, hence, the heat fluxes at the interface may have a higher error with this methodology. The increase in the computational time and the error in the heat fluxes are not desirable features and there may be other approaches with a better performance.

4.2.2 Spatial filter

As an alternative to the temporal filter, which has some inconvenient features, a novel approach is proposed in this research: the use of a spatial filter. This methodology is based on the use of a filter over the values of the temperature or the heat flux for the same time step.

This approach takes advantage of the smoothness of the solution obtained from the heat equation in the solid domain, where no discontinuities are expected and, hence, the use of a spatial filter does not modify the physical meaning of the solution. The selected filter to be used in this application was developed in a previous research and it is called MVF (Mean Value Filter) [58].

The MVF has only one independent parameter, α, which can be tuned to filter a different range of wavelengths. The key behind this filter is that smooth solutions have a gain of one and they are not modified. Only high wavelength sinusoidal components of the solution are affected by this filter. Although it is not exactly a low-pass filter and the Fourier transform is not involved in the process, the properties are similar among both filters.

The main difference between the spatial filter and the temporal filter is that, in the case of the temporal filter, the relaxation coefficient, β, is used to prevent the appearance of any instability in the problem; while the spatial filter destroys the instabilities when they first appear and they are too small to significantly modify the physical behaviour of the problem.

For the TFFB approach, the MVF has to be applied iteratively over the heat flux of every node on the solid-fluid interface, before updating the fluid's BC. The number of times this filter is applied, N, modifies the filter's behaviour: if N increases, more dampening is achieved and more wavelengths are affected. However, in this case, the purpose is to remove the instabilities when they are very small; therefore, a value of $N = 10$ is enough for this application.

An advantage of this method over the temporal filter is that, in this case, the convergence is faster, as the spatial filter preserves the real behaviour of the heat fluxes in each time step, instead of updating their value progressively. Furthermore, the time is not involved in this filter and, hence, the error associated to non-steady problems does not appear in this case. These features are desirable and they lead to an improvement in the general performance of the coupling process. Therefore, this will be the selected approach to stabilise CHT problems in this research.

The Mean Value Filter

Let $f(t)$ be a continuous function within the interval $[a, b]$; there is at least one point c at which the value of the function, $f(c)$, is equal to the average value of the function over the interval:

$$f(c) = \frac{1}{b-a} \int_a^b f(x)\,\mathrm{d}x. \tag{52}$$

In order to turn this integral into a filter, it is necessary to apply this theorem to every sample of the signal. Furthermore, the main functionality of the filter is to smooth the function; so it may be possible to remove the noise. To achieve this smoothness, the new smoothed value must depend on its previous value and the value of adjacent samples.

Having into account these factors, it is possible to rewrite the mean value theorem using a three-point stencil numerical integration [58]:

$$\hat{Y}_j = \frac{1}{x_{j+1} - x_{j-1}} \int_{x_{j-1}}^{x_{j+1}} Y(x)\,\mathrm{d}x = \frac{Y_{j-1} + 2\alpha Y_j + Y_{j+1}}{2(\alpha+1)}, \tag{53}$$

where $\hat{Y}_j$ is the smoothed value of Y_j and α is the order of the numerical integration. The higher the order, the softer the filter and smaller is the change between Y_j and $\hat{Y}_j$.

This filter has to be applied recursively to be effective; therefore, there are only two independent parameters: the order of the filter, α, and the number of times it is applied, N.

4.3 CHT validation: thermal boundary layer

The CHT approach (described previously in sec. 4.1) has to be validated through a test case simulation before the analysis of the reentry capsule can be carried out. The validation increases the reliability of the proposed methodology and allows to check the coupling procedure and the use of high-order methods in the CFD solver.

The selected validation test is a thermal boundary layer, where a fluid flow is cooled by the effect of a solid, as it is represented in Figure 4.3. The selection of this test case was decided due to the scarcity of test cases for compressible CHT problems in the scientific literature. Most of test cases are prepared for an incompressible fluid, which is not representative of the reentry capsule. The simulation will be performed assuming a laminar compressible flow.

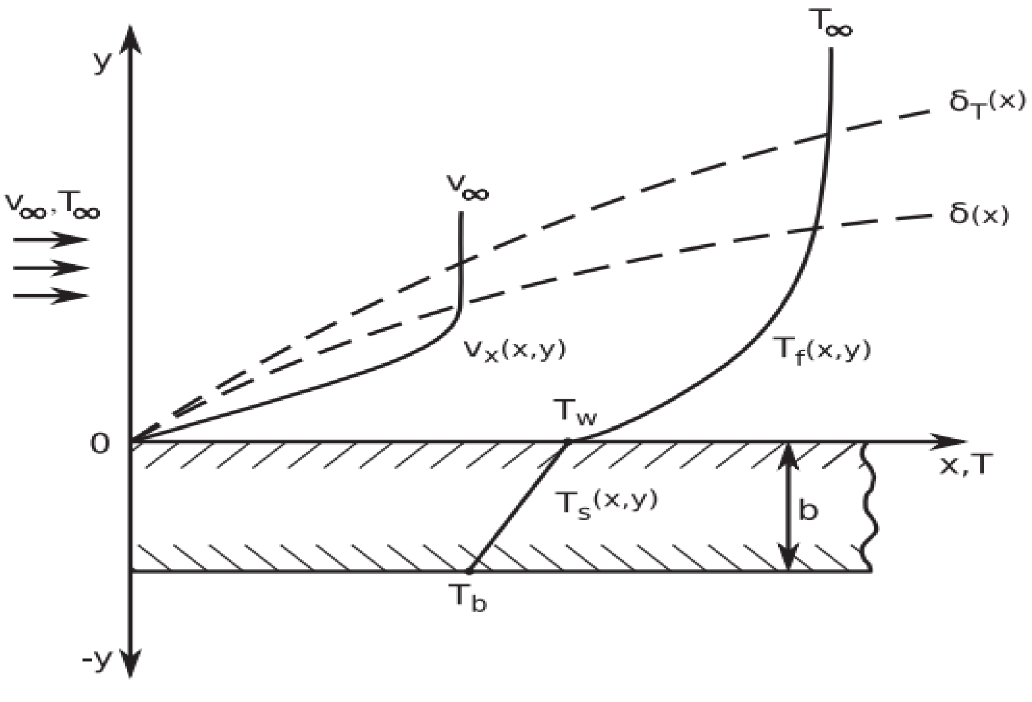


Figure 4.3. Validation test case: thermal boundary layer. [13]

The required data to solve the selected problem is compiled in the Table 4.1 and the geometry and the boundary conditions of the problem are represented in Figure 4.4. The main idea behind this problem is that a hot fluid ($T_\infty = 1000\,\mathrm{K}$) flows through a rectangular channel, from $x = -0.1$ m to $x = 0.2$ m. In $x = 0$ m, 0.1 m far from the inlet, a solid, which is cooled on its base ($T_b = 600\,\mathrm{K}$), appears below the fluid. Therefore, the solid will extract heat from the fluid, reducing the wall temperature at the interface and generating a thermal boundary layer within the fluid domain.

The CHT problem is centred on the resolution of the wall temperature at the solid-fluid interface, following the CHT approach (see sec. 4.1). To improve stability, a spatial filter will be used along the interface to smooth the imposed heat fluxes on the fluid's BC.

The advantage of the selected problem is that there are two analytical approximated solutions [59] and a numerical solution [13], which can be used as a reference to validate the proposed methodology.

Table 4.1.
General data of the CHT test problem: geometry, physical and thermal properties.

Geometry

L_f [m]	H_f [m]	L_s [m]	H_s [m]
0.3	0.09	0.2	0.01

Fluid upstream properties

U_∞ [m/s]	T_∞ [K]	ρ_∞ [kg/m^3]	μ_∞ [Pa $\cdot$ s]	γ	R [J/(kg $\cdot$ K)]
12	1000	0.3525	$3.95 \cdot 10^{-5}$	1.335	287.15

Thermal conductivity

k_f $\left[\frac{\mathrm{W}}{\mathrm{m \cdot K}}\right]$	k_s $\left[\frac{\mathrm{W}}{\mathrm{m \cdot K}}\right]$
0.06808	0.2876

Non-dimensional numbers

Re	M	Pr
21418	0.019	0.663

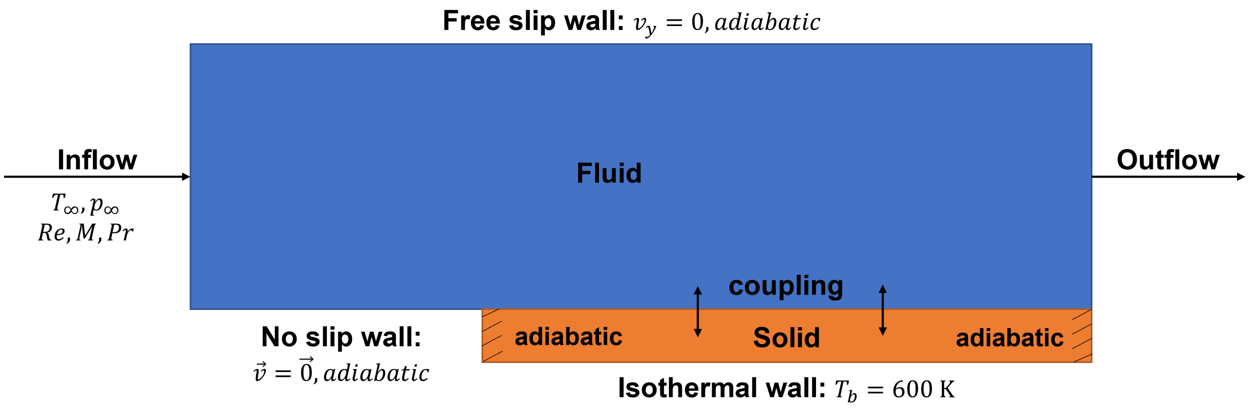


Figure 4.4. Schematic of the validation test case with boundary conditions.

The first step to solve the mentioned problem is to mesh the whole fluid and solid domains. The mesh has to be refined near the interface to accurately capture the boundary layer, as it is represented in Figure 4.5. It is important to notice that the solid's mesh is very fine compared to the fluid's mesh. The reason of this fact is that the solid domain will be solved using a method of order 1 only, due to some limitations of the FEM library (see sec. 9).

However, in the fluid domain, high-order methods will be used and, hence, the mesh can be coarser for the same error. Furthermore, the nodes at the interface of both meshes do not have to be coincident because the temperature distribution and the heat fluxes will be interpolated along the interface every time the coupling is performed.

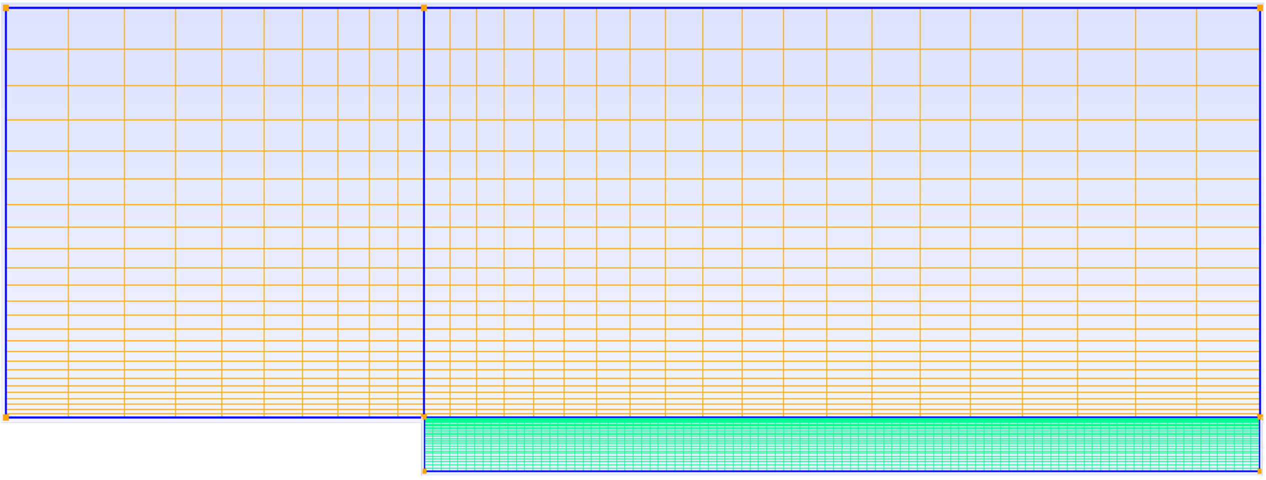


Figure 4.5. Mesh definition for the fluid and solid domains.

Once the mesh has been defined, it is possible to solve the CHT problem in both domains. As it was explained in sec. 4, the coupling process will be performed each N_s iterations of the CFD solver. In this case, the selected value has been $N_s = 1000$ iterations, which allows to speed up the computational time while the convergence is not significantly deteriorated. The selected CHT approach has been TFFB (see sec. 4.1); hence, each time the coupling is performed, the steady-state heat equation will be solved in the solid, taking the fluid's temperature at the interface as Dirichlet BC. Then, the heat flux obtained in the solid domain is used to update the fluid's Neumann BC. This process will be repeated several times until both domains remain in equilibrium.

The most accurate solution has been obtained with a 6th order polynomial method in the fluid domain. The temperature distribution in the fluid is represented in Figure 4.6 and a detailed view of the thermal boundary layer is shown in Figure 4.7. The temperature distribution in the solid domain is shown in Figure 4.8.

As it can be observed in these figures, from $x = 0$ m onward, the solid extracts heat from the fluid and the thermal boundary layer grows progressively. At the same time, the temperature at the interface is reduced as the value of x increases.

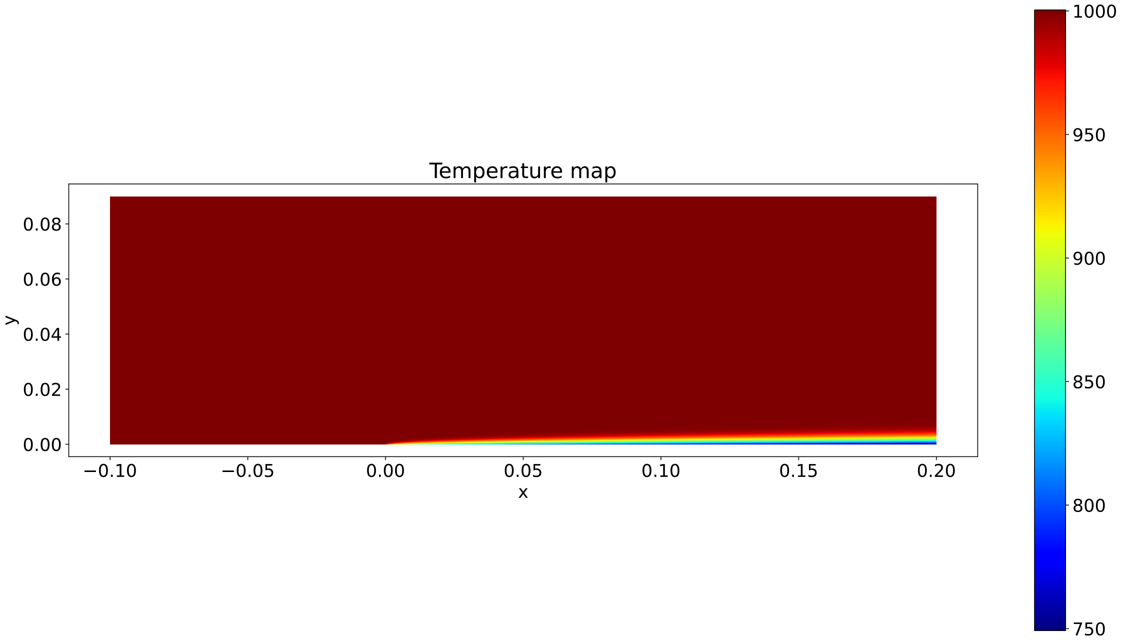


Figure 4.6. Steady-state temperature distribution in the fluid domain with a 6th order polynomial, for the inflow conditions compiled in Table 4.1.

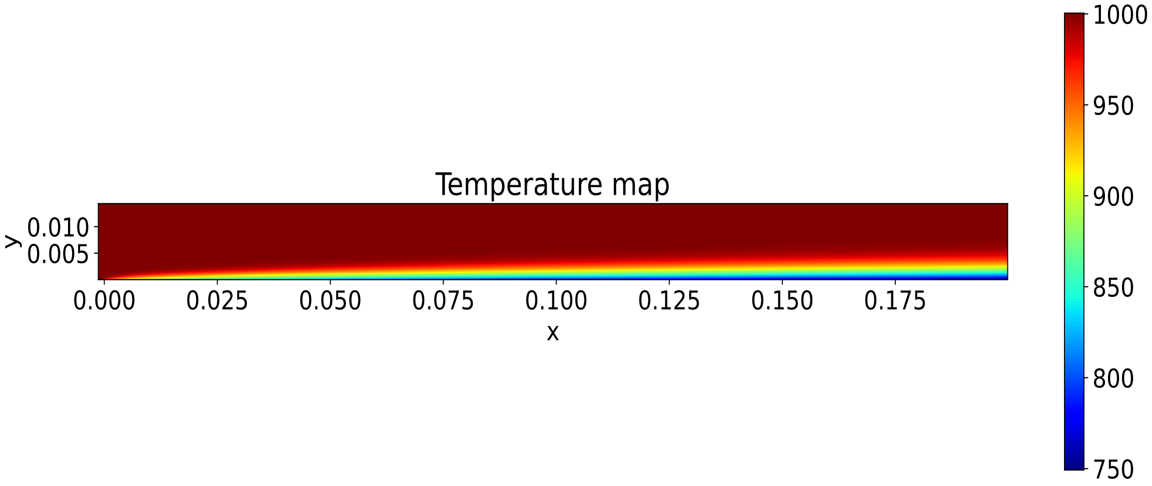


Figure 4.7. Detail of the thermal boundary layer in the fluid domain with a 6th order polynomial, for the inflow conditions compiled in Table 4.1.

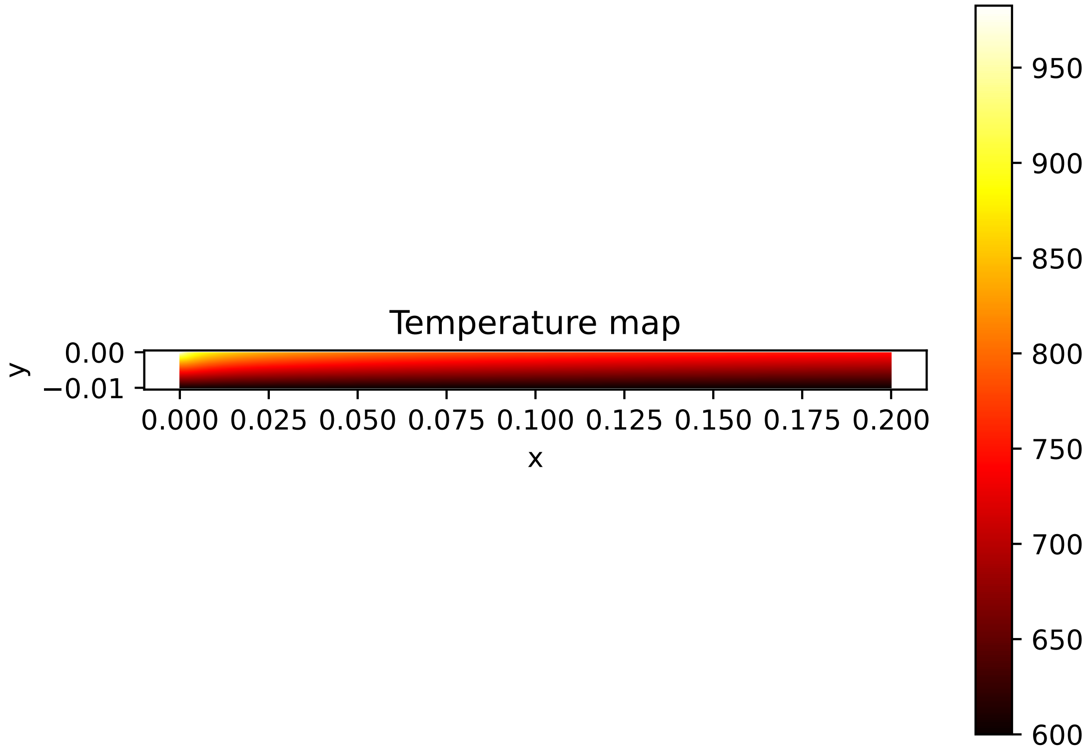


Figure 4.8. Steady-state temperature distribution in the solid domain, for the BC described in Figure 4.4.

In order to have a deeper understanding of the fluid's behaviour, the velocity following the x axis is represented in Figure 4.9, while a detailed view of the velocity boundary layer is shown in Figure 4.10. In this case, the behaviour of the velocity field is very similar to the temperature map, with the thermal and the velocity boundary layer overlapping each other.

This test has allowed to verify the correct functioning of the Python programming interface, which couples both solvers (see sec. 8). Furthermore, the use of a spatial filter has proven to be adequate to remove any possible instability present at the interface during the coupling process and a converged steady-state solution has been obtained with the proposed methodology.

Additionally, these results prove that it is possible to use different solvers, polynomial orders and even different meshes in both domains to compute the final solution. This flexibility is essential to be able to implement Conjugate Heat Transfer problems with different tools and it allows to take advantage of the robustness of preexisting solvers, while user-defined solutions can be checked at the same time in a conjugate problem.

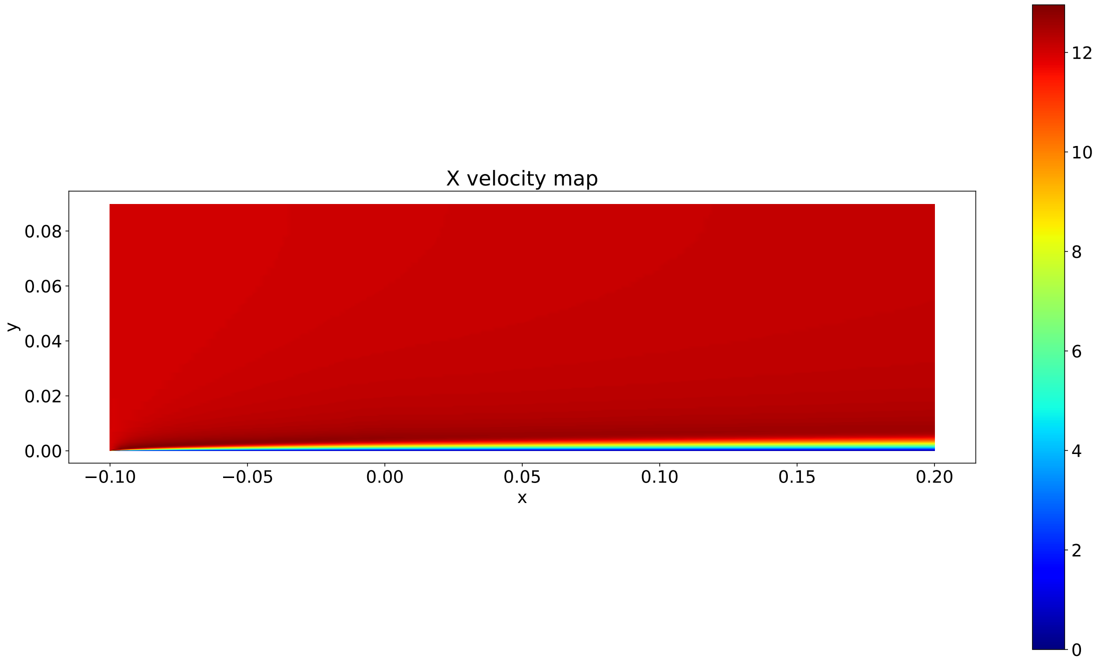


Figure 4.9. Steady-state x-velocity field in the fluid domain with a 6th order polynomial, for the inflow conditions compiled in Table 4.1.

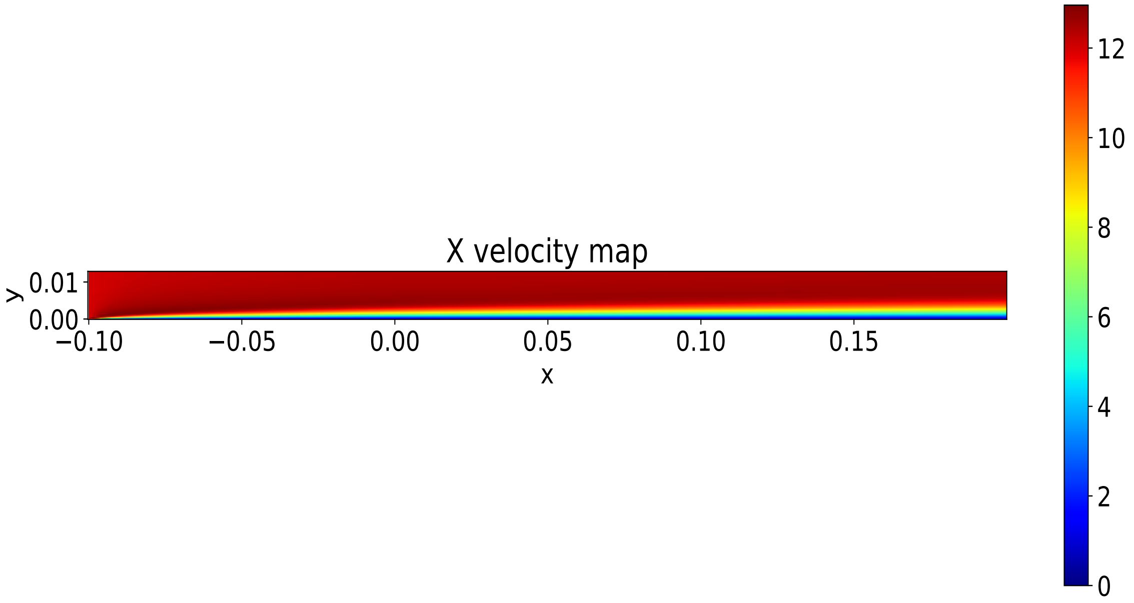


Figure 4.10. Detail of the velocity boundary layer in the fluid domain with a 6th order polynomial, for the inflow conditions compiled in Table 4.1.

Finally, as a complement the previous analysis, the solution will be obtained for different polynomial orders, for the same mesh, and the results will be compared to a reference solution. Then, fixed the polynomial order, the mesh will be modified to estimate the numerical error through a *Richardson extrapolation*.

4.3.1 Results for p-order methods

The use of high-order methods to solve the fluid equations allows to obtain accurate results with a coarser mesh, which leads to a reduction of the global computational time for a same error. In order to check the effect of the use of different polynomial orders over the final solution, different analyses have been performed for a single mesh (see Figure 4.5) and the results have been compared with the solutions proposed in [59] and [13].

First, Figure 4.11 represents the evolution of residuals in the fluid equations during the simulation of the CHT problem with a 1st order polynomial method. The steady-state convergence is achieved with 150000 iterations approximately, using a 3rd order Runge-Kutta temporal integrator and a time step $dt = 10^{-5}$ s.

This first solution is very useful, as it can be used as an initial condition to restart the CFD solver and speed-up the convergence for higher order methods. This functionality can be used in the CFD solver, HORSES3D, if the mesh remains the same, which is the case for the present analysis.

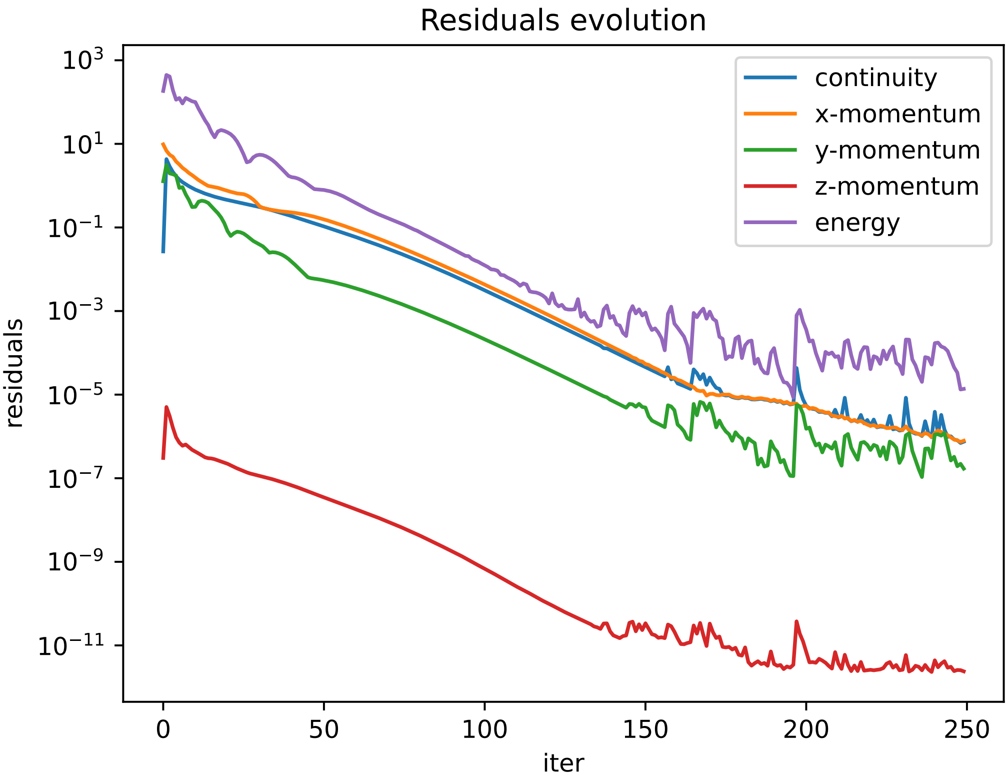


Figure 4.11. Residuals in the fluid equations during the simulation with a 1st order polynomial method, in thousands of iterations.

The solution of the CHT problem has been obtained for different polynomial orders and the solution is represented in Figure 4.12. As it can be observed, the solution is slightly different for each polynomial order, even though the mesh remains the same. These differences are highlighted below, in Figure 4.13, where the error between different solutions is represented.

The solution obtained with the proposed approach is very close to the reference solutions, which determines the success of the validation test case for the CHT problem.

It is important to notice that the reference solutions are not exact: the analytical solutions are approximated solutions, obtained by [59] through two different methodologies, while the numerical reference solution has been obtained by [13] using their own CHT methodology and a second order method. In general, the behaviour among the different solutions is very close, and the differences are small in relation to the reference temperature of 1000 K.

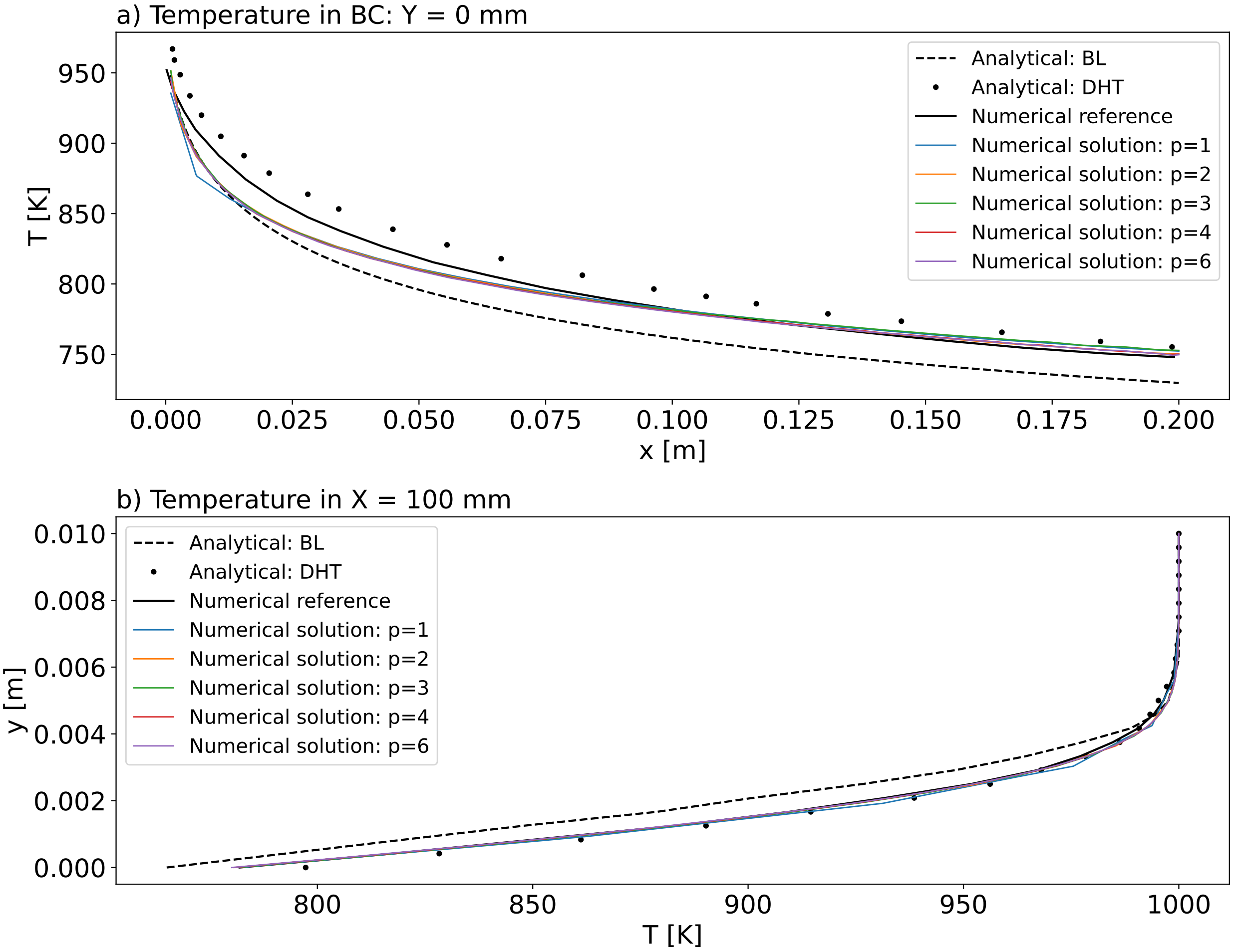
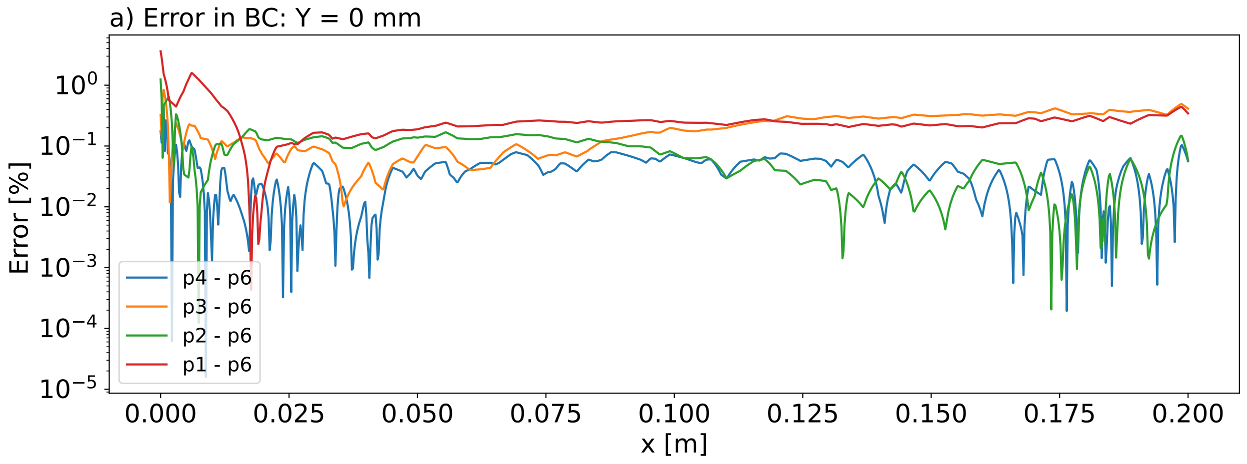


Figure 4.12. a) Temperature distribution at the fluid-solid interface for different polynomial orders, together with the reference solutions. b) Temperature distribution normal to the boundary, at $x = 100$ mm, for different polynomial orders, together with the reference solutions.

Furthermore, the analytical solutions shown in Figure 4.12 a), have a singularity at $x = 0$ m, which generates a higher error in that region. Therefore, a possible option to avoid this problem is to compare the results at $x = 100$ mm, in the direction normal to the boundary, as it is shown in Figure 4.12 b). That point is in the middle of the interface and, hence, the effect of the singularity, the inlet and the outlet are negligible and the comparison is more reliable. In this second case, the solution obtained with different polynomial orders, the numerical reference solution and one of the analytical solutions are almost identical and the error among them is very close to zero. Therefore, the proposed approach is reliable as a possible methodology to solve Conjugate Heat Transfer problems.

As a complement to the previous analysis, it is important to know the effectiveness of the selected polynomial order. To obtain an indicator of each result, every solution from $p = 1$ to $p = 4$ have been compared to the solution $p = 6$, which is the most accurate. The error of the temperature for each polynomial order is represented in Figure 4.13.


a) Error in BC: Y = 0 mm

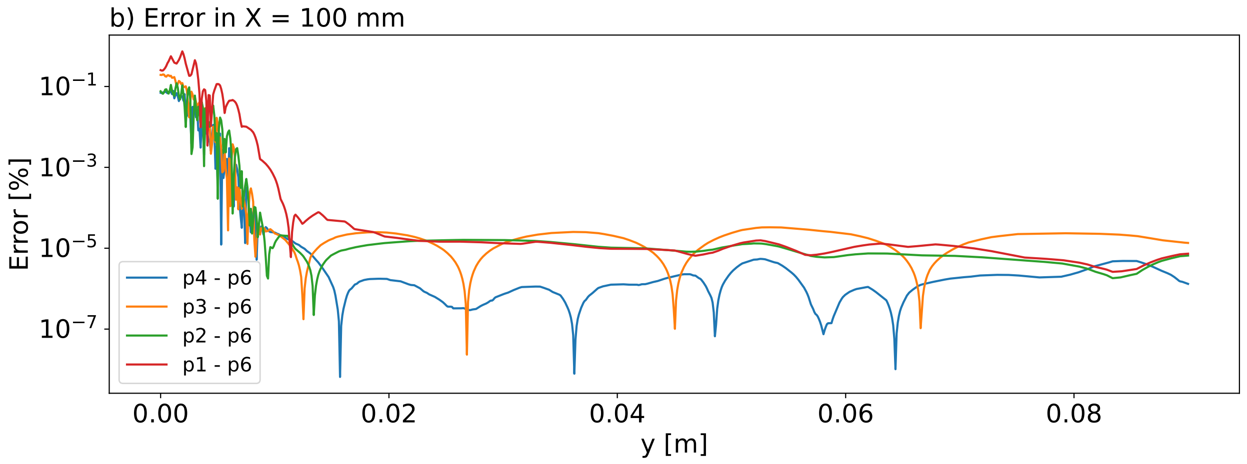

b) Error in X = 100 mm

Figure 4.13. a) Error along the fluid-solid interface compared to the solution $p = 6$. b) Error normal to the boundary, at $x = 100$ mm, compared to the solution $p = 6$.

The previous error, E_{pi-p6}, measured in %, has been computed through the following expression:

$$E_{pi-p6} = \frac{|T_{pi} - T_{p6}|}{T_{\infty}} \cdot 100, \tag{54}$$

where $T_{\infty} = 1000\,\mathrm{K}$ is the reference temperature, T_{pi} is the temperature obtained with a polynomial order $p = i$, for $i = 1, 2, 3$ and 4; and T_{p6} is the temperature obtained with a polynomial order $p = 6$.

As it is shown in Figure 4.13, in general, the error decreases as the polynomial order increases. The maximum error is obtained near the beginning of the boundary layer, at $x = 0$ m, and near the interface, at $y = 0$ m.

Furthermore, the 4th order polynomial solution has an outstanding behaviour, while the 3rd order solution is not as good as it was expected and, even, it is surpassed by the 2nd order polynomial solution in several regions.

Another difference among the different p-order solutions is the behaviour of the heat fluxes, obtained by solving the heat equation in the solid domain. These heat fluxes are used as a Neumann BC for the fluid and they are a key aspect for the convergence. These differences are shown in Figure 4.14, where the temperature gradient distribution at the solid interface is shown for different polynomial orders and before the spatial filter has been applied.

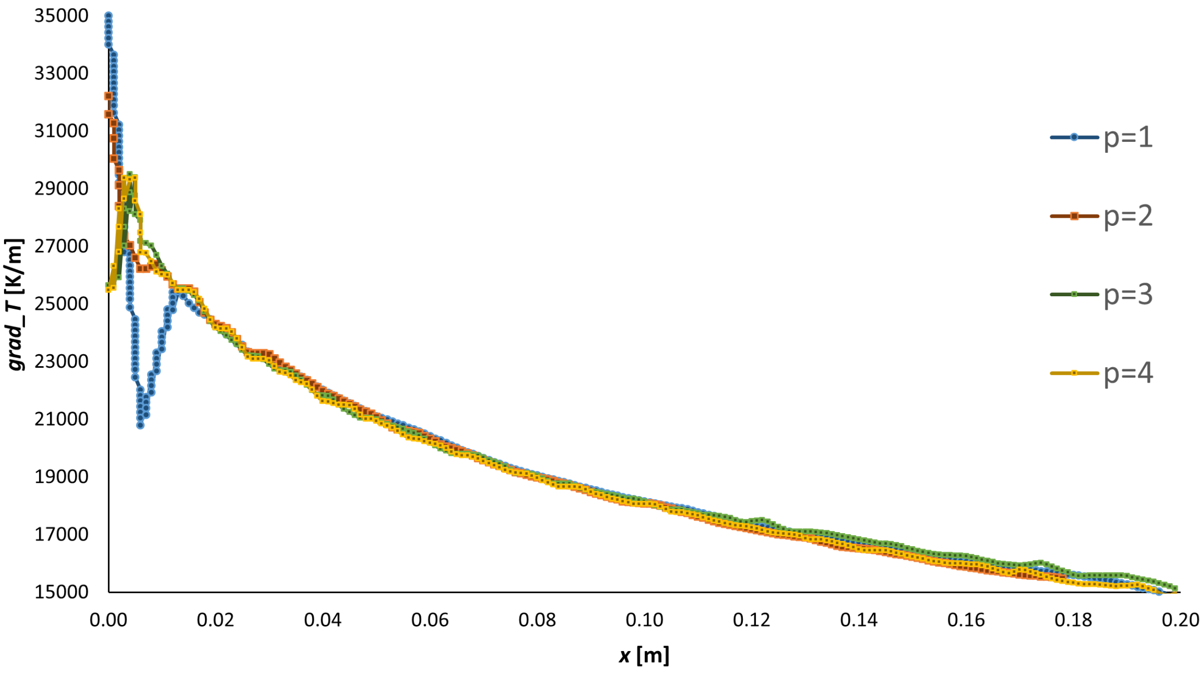


Figure 4.14. Temperature gradient at the solid interface for different polynomial order solutions.

Although the shape is similar for $p = 3$ and $p = 4$, the behaviour changes for low order polynomials. Specially, the 1st order polynomial solution would generate a divergence of the method if the spatial filter had not been used over this distribution before updating the fluid's BC. Furthermore, the maximum value of the gradient is reduced progressively from $p = 1$ to $p = 3$. From $p = 3$ onward, the maximum value and the general shape remains the same, with only small variations.

4.3.2 Error convergence: Richardson extrapolation

The last analysis of this validation test case is focused on the estimation of the numerical spatial error of the proposed methodology. There are not analytical exact solutions to obtain the error and, hence, other approaches have to be used to estimate the error. In this case, the *Richardson extrapolation* will be used with this aim.

This analysis is based on the definition of 3 meshes: $M1$, $M2$ and $M3$, each one more refined than the previous one. The values of Δx and Δy for these meshes verify:

$$\Delta x_1 = \Delta y_1 = 0.01\,\mathrm{m}, \qquad \Delta x_2 = \Delta y_2 = \Delta x_1/2, \qquad \Delta x_3 = \Delta y_3 = \Delta x_2/2. \tag{55}$$

The previous definition of Δx and Δy assure that every node of $M1$ is present in $M2$, and every node of $M2$ is present in $M3$. This way, the solution of each mesh can be compared, as it will be explained below. The first mesh, $M1$, is shown graphically in Figure 4.15, where the fluid domain has been meshed completely with squares.

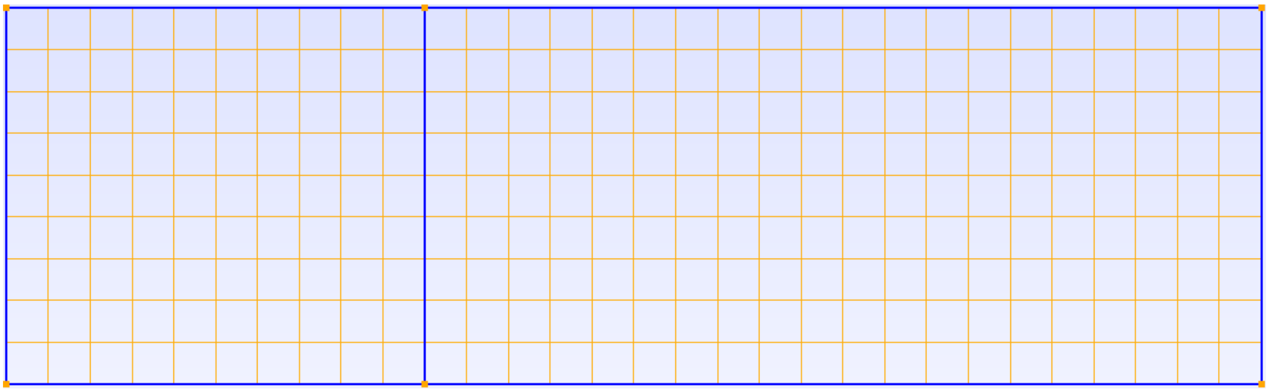


Figure 4.15. Mesh $M1$ used for the error estimation through Richardson extrapolation.

Let T_j be the exact solution of the temperature in the node j of the fluid domain, and T_j^i the numerical solution of the temperature in the node j of the fluid domain, computed through the mesh i. The error, E, of the numerical solution of order q can be defined as follows:

$$E_j^i = T_j - T_j^i = k_{x,j}\Delta x_i^q + k_{y,j}\Delta y_i^q = k_j \Delta x_i^q. \tag{56}$$

If the previous expression of the error is compared between two consecutive meshes:

$$E_j^i - E_j^{i+1} = T_j^{i+1} - T_j^i = k_j \left(\Delta x_i^q - \Delta x_{i+1}^q\right). \tag{57}$$

The key of this methodology is that the value of k_j is not dependent on the mesh and it is unique for each node. Therefore, the value of k_j can be obtained from the previous expression:

$$k_j = \frac{T_j^{i+1} - T_j^i}{\Delta x_i^q - \Delta x_{i+1}^q}. \tag{58}$$

Known the solution for two meshes, and selecting the nodes j that are common in both solutions, all the k_j values can be obtained. Then, using these same values in the expression (56), it is possible to plot a map of the numerical error in each node.

This process has been performed between the meshes $M1$ and $M2$, and between $M2$ and $M3$, using a 4th order polynomial, $p = 4$, in both cases. It is important to notice that the order of the error, q, and the polynomial order, p, are not necessarily equal, as it will be shown later. The value of q depends on the approach followed for the spatial discratisation and other issues related to the concrete problem to be solved (e.g. discontinuities, singularities or unresolved regions may decrease the expected order q, that determines the error convergence). The results are shown in Figure 4.16, where the error maps are represented. As it was expected, a refinement of the mesh reduces the global error. Furthermore, the highest errors are located near the boundary layer, where the temperature gradients are maximum.

In the case of the third mesh, $M3$, the maximum absolute error is near 10^{-1}, which is a very low value taking into account that the temperature at the inlet is $T_\infty = 1000\,\mathrm{K}$. Therefore, an error of $0.1\,\mathrm{K}$ is negligible in this case.

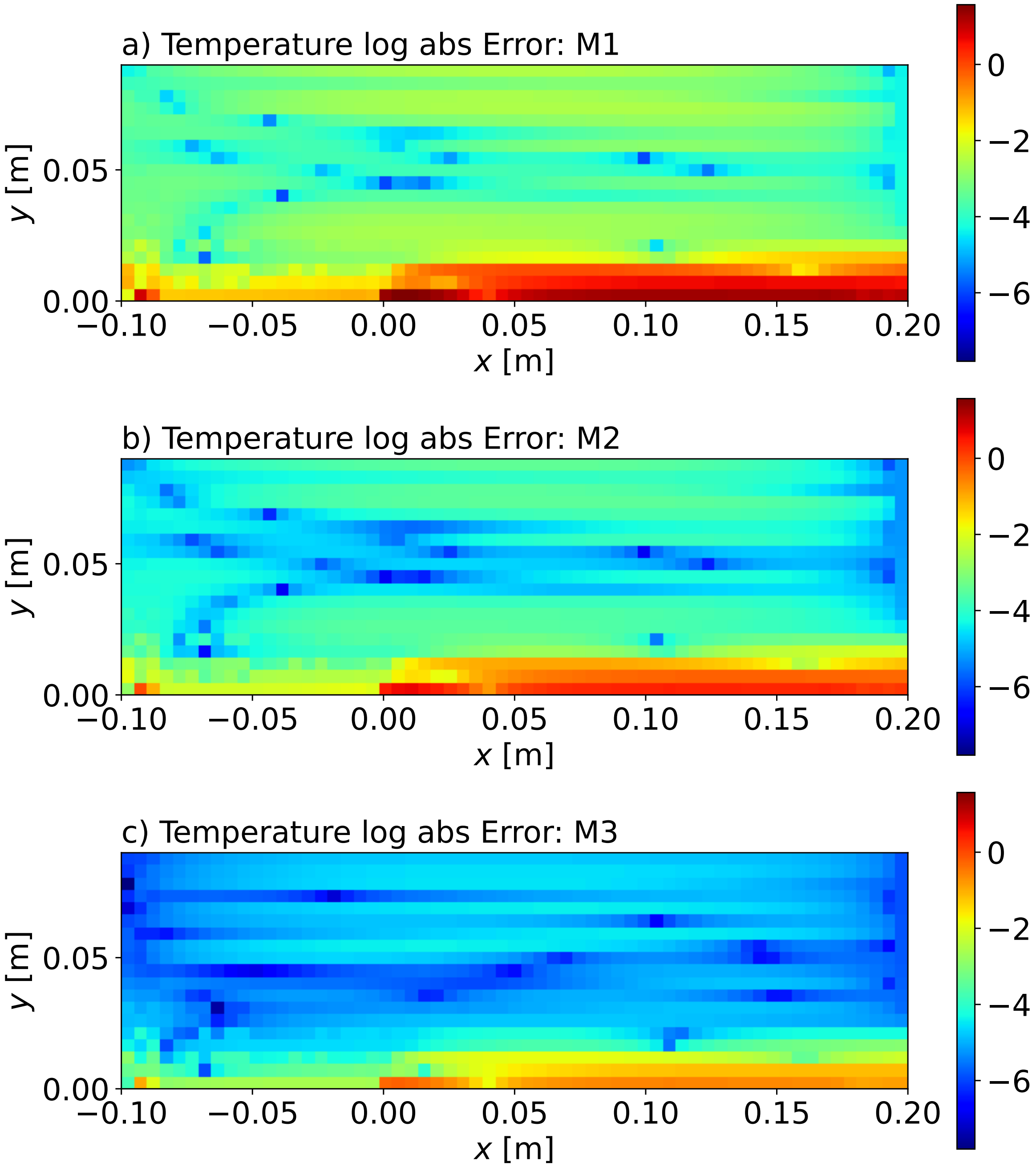


Figure 4.16. Temperature error map, in logarithmic scale, using the results from the meshes $M1$, $M2$ and $M3$.

Finally, these error maps can be compressed into a global value in order to analyse the convergence of the numerical error as a function of the mesh refinement. Three different global values have been obtained for each error map as follows:

$$E^{i}_{abs} = \sqrt{\frac{\sum_{j=1}^{N_i} {E^{i}_{j}}^{2}}{N_i}} \qquad \text{for } i = 1, 2 \text{ and } 3, \tag{59}$$

$$E_{rel}^{i} = \sqrt{\frac{\sum_{j=1}^{N_i} \left(E_j^i / T_\infty\right)^2}{N_i}} \qquad \text{for } i = 1, 2 \text{ and } 3, \tag{60}$$

$$E_{max}^{i} = \max(E_j^i) \qquad \text{for } i = 1, 2 \text{ and } 3, \tag{61}$$

with E_{abs}^i as the absolute RMSE (Root Mean Squared Error) for the mesh i, E_{rel}^i as the relative RMSE for the mesh i, E_{max}^i as the maximum error for the mesh i, E_j^i as the error in the node j for the mesh i, and N_i as the number of nodes within the mesh i.

This convergence is shown graphically in Figure 4.17. The more refined is the mesh, the smaller is the error. This methodology is useful to estimate the error convergence order, q, which may be unknown when using a new solver.

In this case, the slope of the curves is very close to $q = 3$, which is different from the polynomial order $p = 4$, in a double logarithmic scale. This result is due to the singularity present at the beginning of the boundary layer, which increases locally the error and reduces the convergence order below the polynomial order. However, in a general problem, if there were not singularities, the error convergence order, q, could be equal to $p + 1$ for a DG discretisation [36], when Gauss quadratures are used to approximate the integrals that appear in the weak formulation of the Navier-Stokes equations, which would significantly improve the accuracy of the solution.

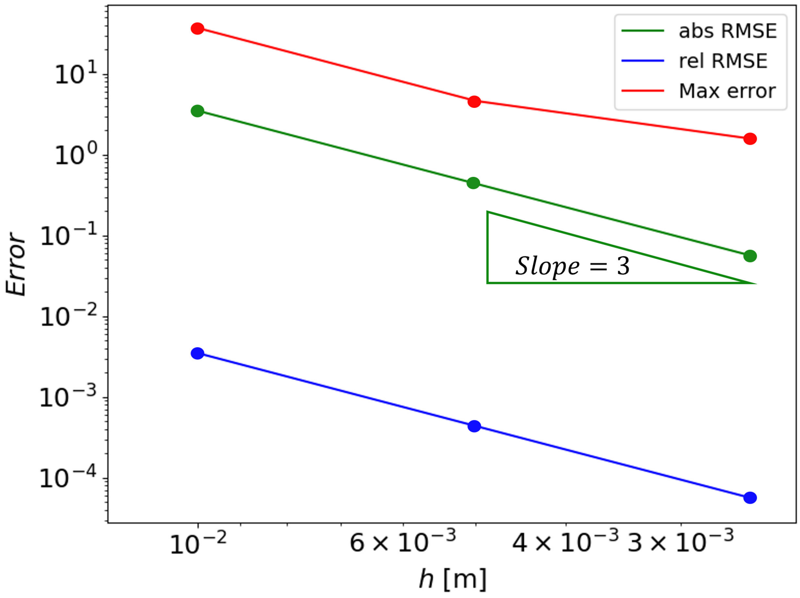


Figure 4.17. Maximum error, absolute error and relative error (with respect to T_∞) convergence in relation to the mesh refinement h, using a $p = 4$ method.

5 Results

Once the bases of the aerodynamic problem and the thermal problem have been established, it is possible to solve the hypersonic flow around the reentry capsule Apollo and the associated conjugate heat transfer problem, which takes into account the heat equation within the solid domain. The meshes used to solve the equations through a FEM approach were previously described in sec. 2.2 for the solid domain (see Figure 2.3), and in sec. 3.2 for the fluid domain (see Figure 3.4), except for the shock capturing analysis, as it will be shown later.

It is important to notice that all the following analyses will be performed to obtain a *quasi steady-state* solution, as the perfect steady solution does not exist in presence of turbulence, for the upstream conditions previously explained in sec. 3.2 (see Table 3.2), with an angle of attack of $-18°$, which is an usual value for the Apollo capsule, and taking into account the conditions after the shock wave. The real reentry conditions are not steady and the critical phase of the descent only lasts around 500 s. Therefore, the problem analysed in this research can be considered as a worst-case analysis, due to the thermal loads will increment the maximum temperature as the reentry time increases.

All the following results have been developed to be considered as a guide to the analysis of a reentry capsule and to show the potential of non-commercial open-source software, including the coupling between different solvers through a programming interface (see sec. 8), to solve complex conjugate heat transfer problems, whose main difficulties are described in Annex 2 (see sec. 9). Finally, the use of high-order methods will improve the accuracy of the solution, allowing to use a coarser mesh and reducing the computational time for a threshold error.

5.1 Aerodynamic problem: capsule as an adiabatic BC

The first analysis is the resulting 3D flow field around the reentry capsule, considered as an adiabatic BC. This analysis will be useful to verify the correct implementation of the turbulence model and to have an idea of the fluid's behaviour for further analyses.

In this case, the adiabatic BC sets the heat flux at the interface to zero. Hence, the maximum temperature will be located at the boundary of the capsule, where the velocity is zero due to the no-slip condition, as the kinetic energy is transformed into thermal energy. Due to this adiabatic BC, the highest temperatures will be reached in this first analysis.

The solution is represented in Figures 5.1 to 5.6, where the Mach number, the temperature, the pressure, the density and the velocity (u following the x axis and v following the y axis), are shown. All these results are represented using the IS (International System) of units.

This solution has been computed using a $p = 3$ polynomial order, which is equivalent to a 4th order method. DG methods have proven to have an error convergence rate proportional to Δx^{p+1} [36] if Gauss quadratures are used to approximate the integrals that appear in the weak formulation of the Navier-Stokes equations, and if the solution is smooth, which is a reasonable assumption in this case, due to the considered upstream conditions have been calculated after the shock wave.

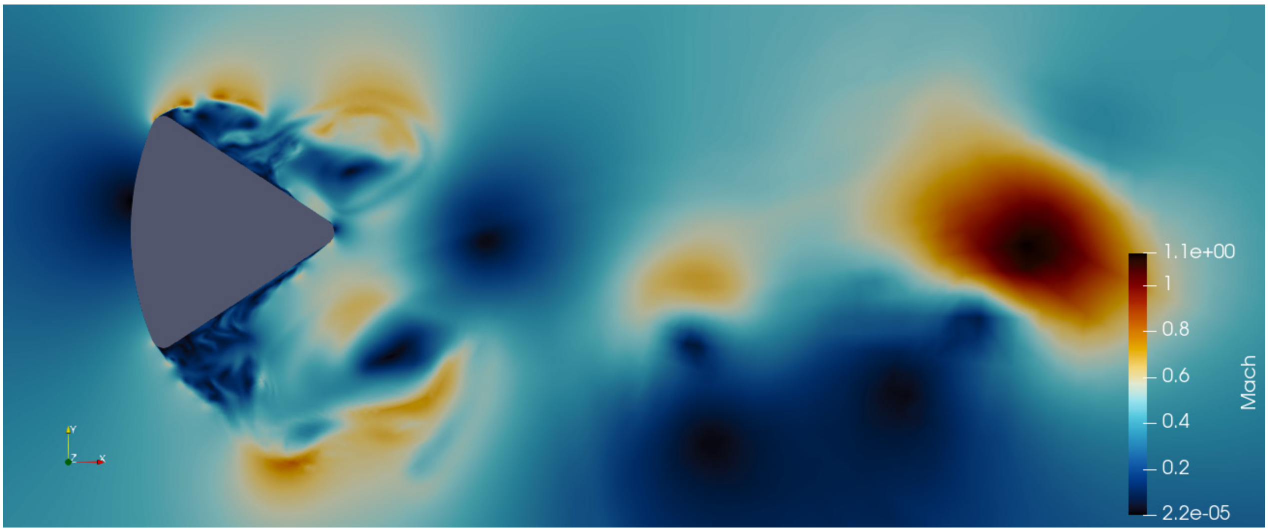


Figure 5.1. Adiabatic capsule: Snapshot of the Mach number distribution in the fluid domain with $p = 3$, for the inflow conditions after the shock wave and $AoA = -18°$.

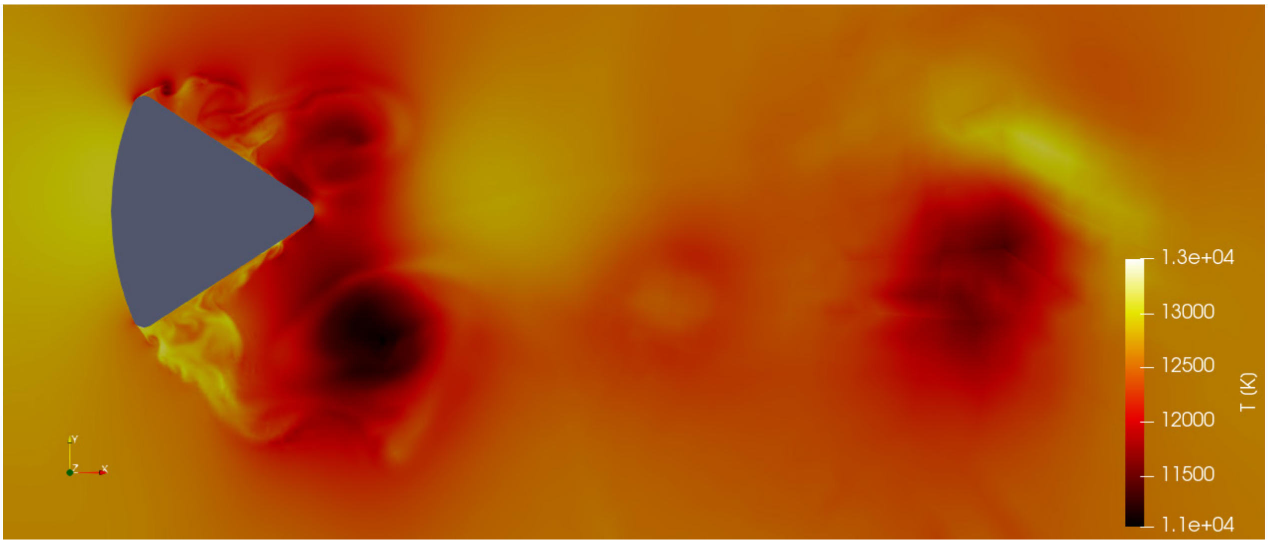


Figure 5.2. Adiabatic capsule: Snapshot of the temperature distribution in the fluid domain with $p = 3$, for the inflow conditions after the shock wave and $AoA = -18°$.

The LES model, Vreman, described in sec. 3.1.2, has allowed to accurately capture the 3D turbulent flow around the capsule, showing a rich distribution of eddies and a well defined flow separation near the sides of the capsule. Furthermore, the effect of the bigger eddies has transformed part of the thermal energy into kinetic energy, reaching a maximum Mach number of $M = 1.1$ downstream. There is also a clear asymmetry of the flow structure due to the effect of the angle of attack, $\alpha = -18°$.

The maximum temperature of 13495 K is reached at the front of the capsule, where the Mach number is zero, and on the lower side of the capsule, where the effect of turbulence leads to the presence of a low velocity region. The cooler part of the capsule is the rear, where the temperature has a value of 12000 K.

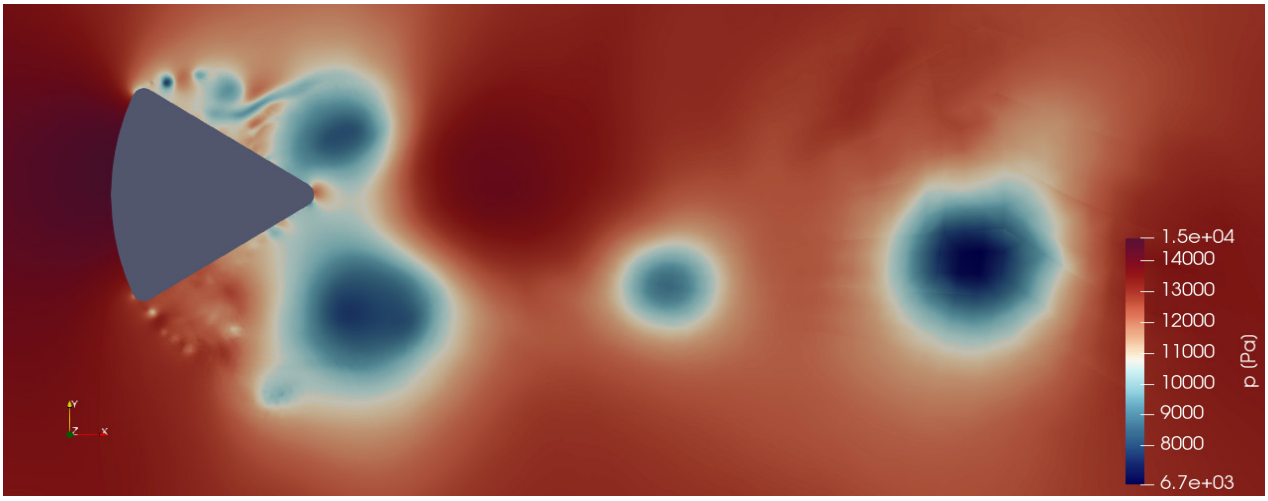


Figure 5.3. Adiabatic capsule: Snapshot of the pressure distribution in the fluid domain with $p = 3$, for the inflow conditions after the shock wave and $AoA = -18°$.

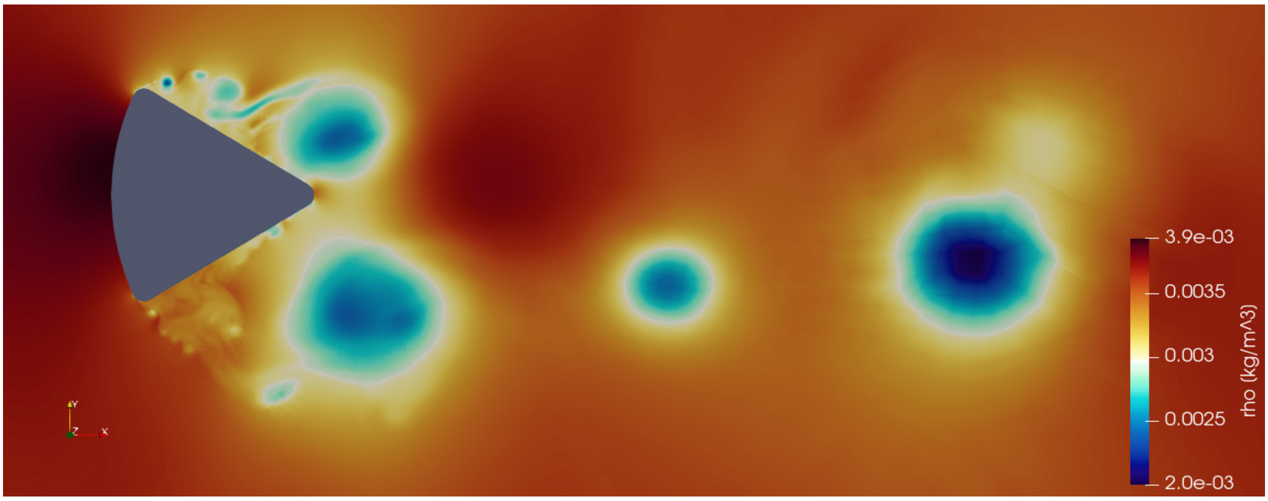


Figure 5.4. Adiabatic capsule: Snapshot of the density distribution in the fluid domain with $p = 3$, for the inflow conditions after the shock wave and $AoA = -18°$.

The pressure and density maps are useful to check visually the location of the main eddies of the flow, where the pressure and the density drop to their minimum values of $p = 6700\,\mathrm{Pa}$ and $\rho = 2 \cdot 10^{-3}\,\mathrm{kg/m^3}$, respectively. Their maximum values are located at the front of the capsule, as expected, with a maximum pressure of $14697\,\mathrm{Pa}$ and a maximum density of $3.92 \cdot 10^{-3}\,\mathrm{kg/m^3}$.

Finally, the velocity maps, which represent the u and v components, give a better idea of the flow field surrounding the capsule. At the rear of the capsule, there is a big region where the air recirculates and travels downstream.

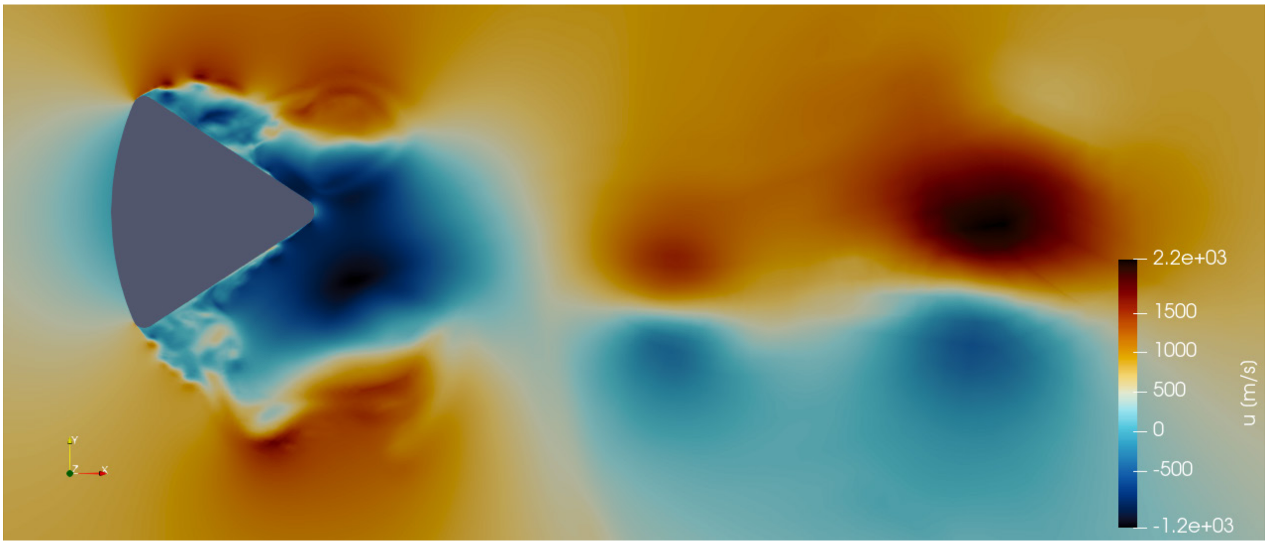


Figure 5.5. Adiabatic capsule: Snapshot of the velocity field, following the x axis, in the fluid domain with $p = 3$, for the inflow conditions after the shock wave and $AoA = -18°$.

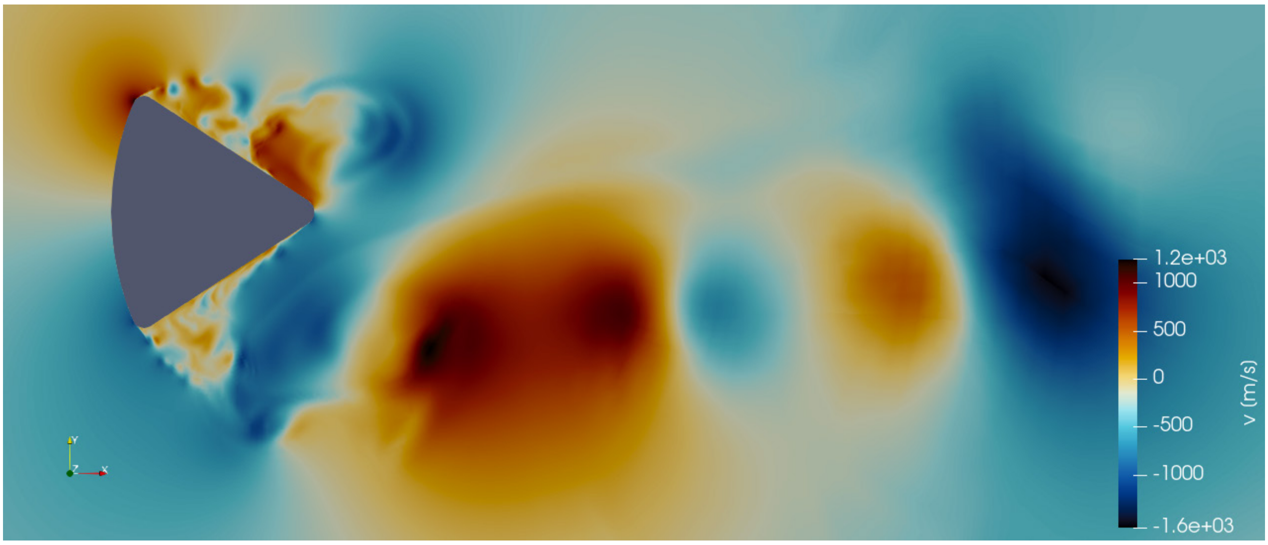


Figure 5.6. Adiabatic capsule: Snapshot of the velocity field, following the y axis, in the fluid domain with $p = 3$, for the inflow conditions after the shock wave and $AoA = -18°$.

To give a better understanding of the previous result and of the trajectory followed by the air flow, Figures 5.7 and 5.8 represent the streamlines and the velocity vectors of the flow field. In this case, the recirculating region at the rear of the capsule is highlighted through the vector representation.

It is important to notice that this flow field is not completely steady. It is constantly fluctuating, following some patterns that are repeated over time, almost periodically. This effect is represented in Figure 5.9, where the Mach number distribution is represented for 3 different time stamps.

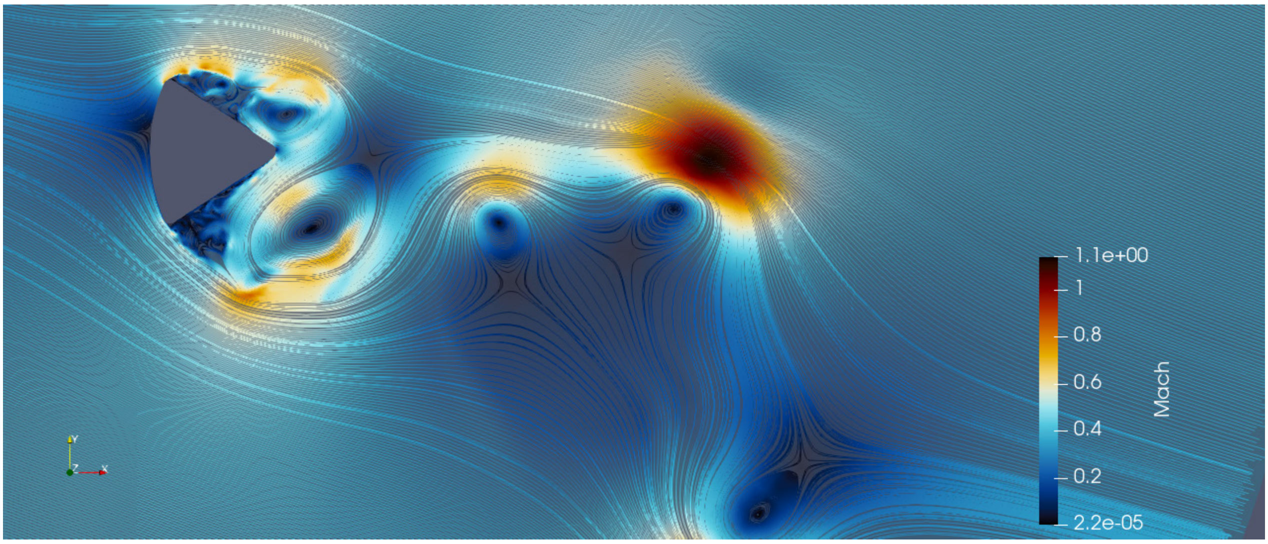


Figure 5.7. Adiabatic capsule: Snapshot of the streamlines of the flow field with $p = 3$, for the inflow conditions after the shock wave and $AoA = -18°$.

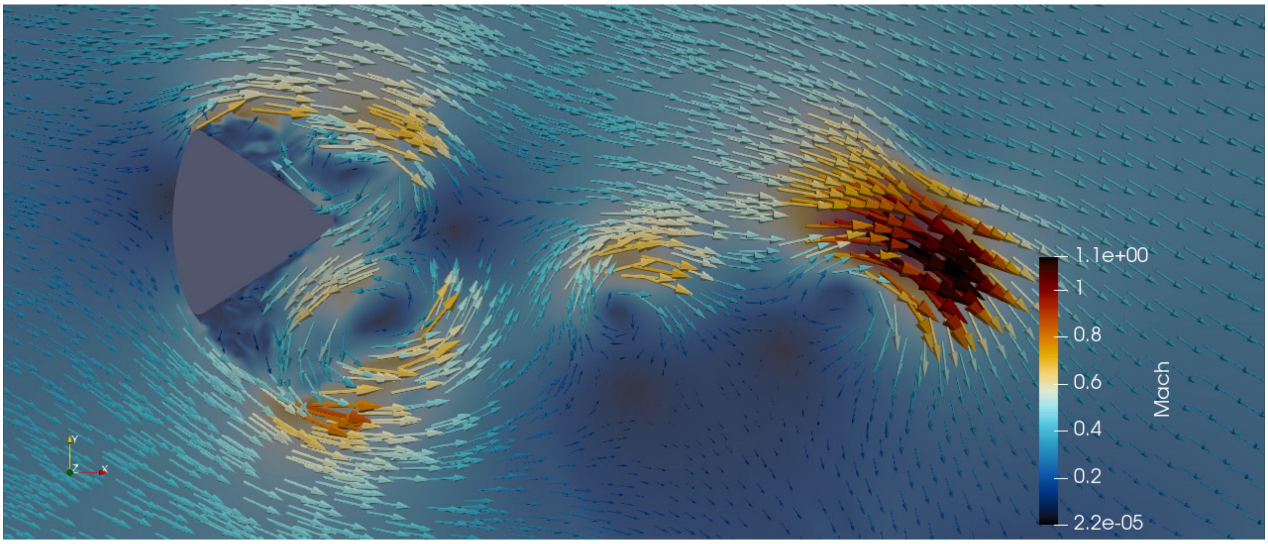
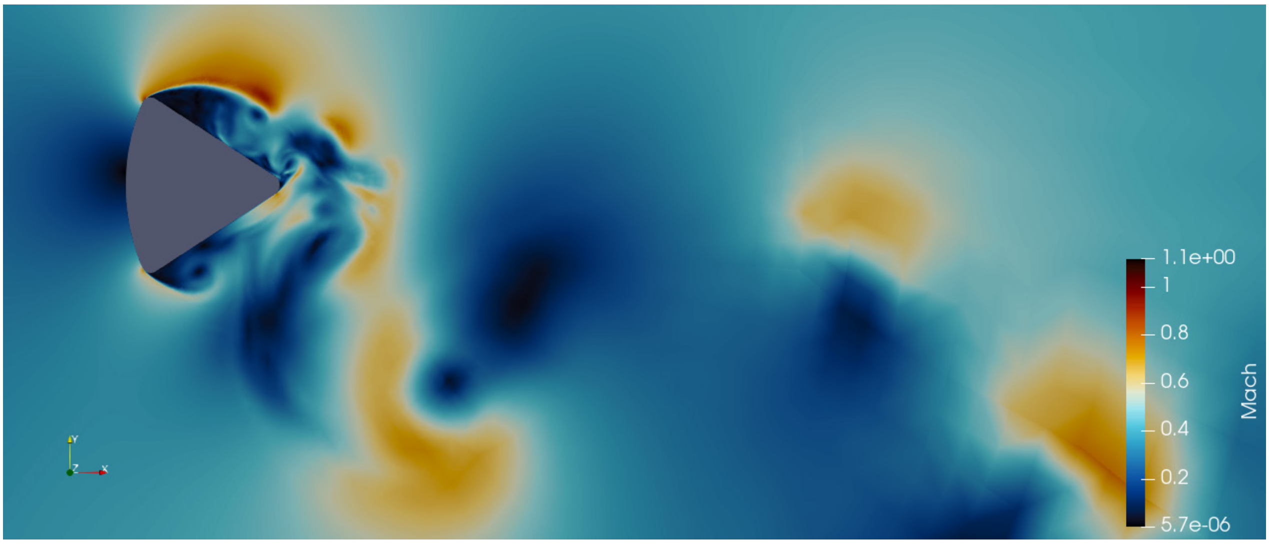
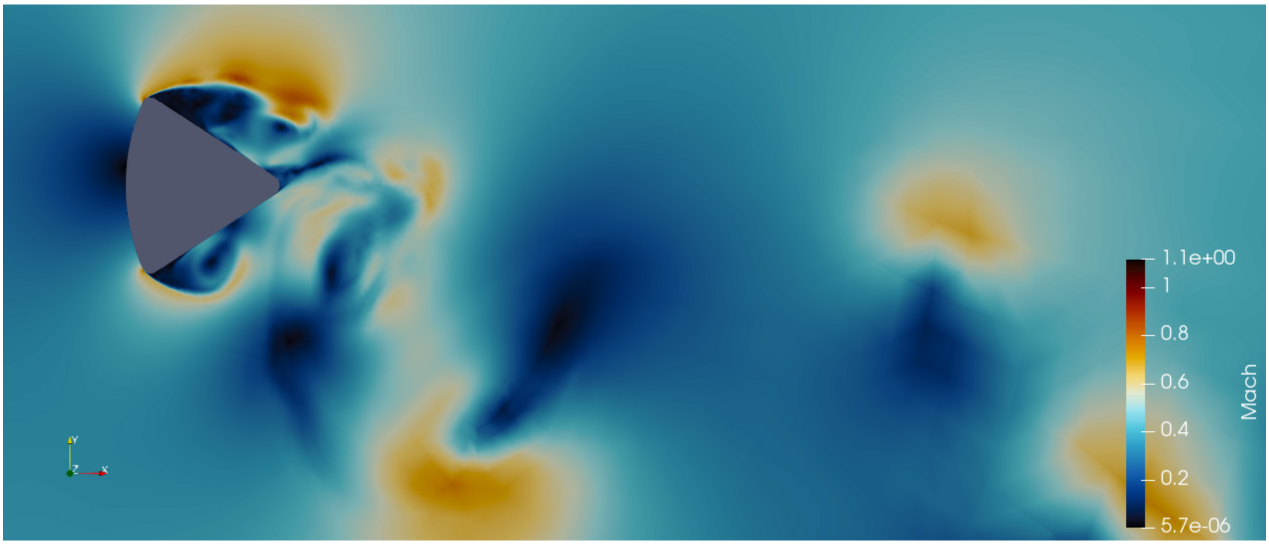


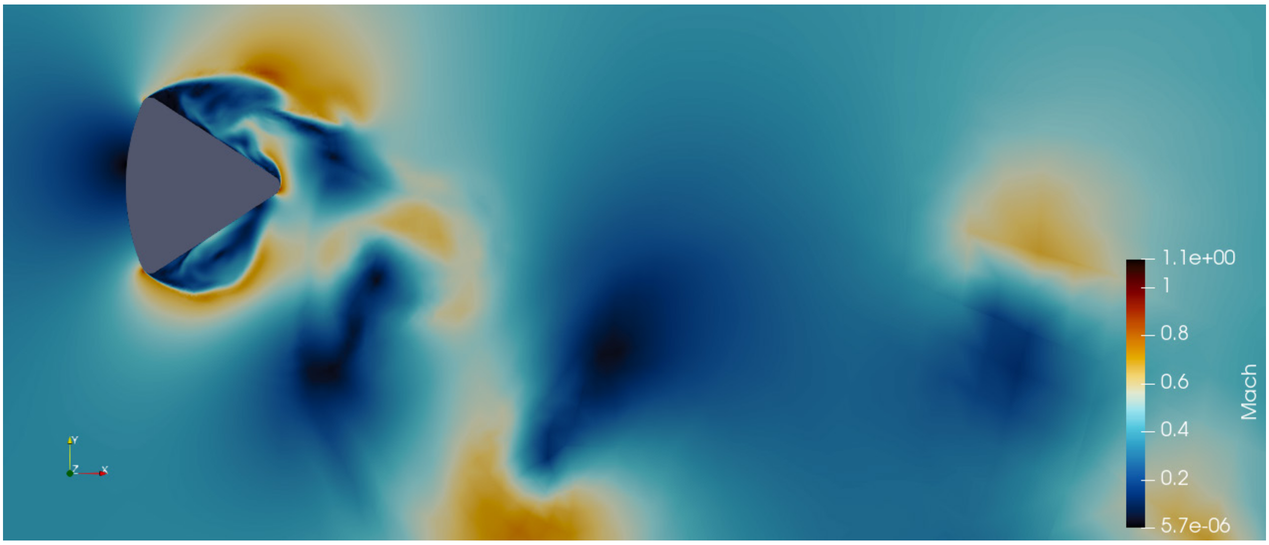
Figure 5.8. Adiabatic capsule: Snapshot of the velocity vectors of the flow field with $p = 3$, for the inflow conditions after the shock wave and $AoA = -18°$.



(a) $t1 = 40.88\,\mathrm{s}$.



(b) $t2 = t1 + 0.6\,\mathrm{s}$.



(c) $t3 = t2 + 0.6\,\mathrm{s}$.

Figure 5.9. Adiabatic capsule: Snapshot of the Mach number distribution in the fluid domain for different timestamps with $p = 3$, for the inflow conditions after the shock wave and $AoA = -18°$.

5.2 Aerodynamic problem: capsule as an isothermal BC

As a complement to the previous analysis, the same problem has been solved considering a capsule modelled as an isothermal BC at 2000 K. This analysis is closer to the real situation, when the capsule is cooled through the effect of thermal radiation. The presence of a thermal boundary layer is expected at the interface, but the general behaviour of the turbulent flow should not be significantly different from the previous case.

This analysis is specially interesting due to the solution can be used as a restart point to solve the conjugate heat transfer problem. The use of the isothermal solution instead of the adiabatic solution as a restart point improves the stability of the coupling process and it is essential if the effect of thermal radiation has to be considered (see sec. 9).

The temperature distribution around the capsule and the thermal boundary layer at the interface are represented in Figures 5.10 and 5.11. In this case, the capsule has cooled the surrounding flow, resulting in a reduction of the temperature in the eddies separated from the body, which travel downstream.

Furthermore, a thermal boundary layer can be clearly appreciated in the detailed figure, as it was expected. This behaviour was not present in the previous analysis due to the adiabatic BC. This temperature distribution is expected to be similar to the analysis considering thermal radiation, as the capsule will be cooled and a similar thermal boundary layer should be present.

Figure 5.10. Isothermal capsule: Snapshot of the temperature distribution in the fluid domain with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

Although the Mach number, the pressure and the velocity field are almost identical to the adiabatic case, the density distribution has been transformed in a similar way that the temperature, due to both magnitudes are related through the equation of state (27).

The density distribution is represented in Figures 5.12 and 5.13. It is important to mention that in Figure 5.12 the maximum density has been reduced artificially to show the density distribution in a clearer way, due to the difference between the maximum and minimum values is important. However, the real scale is shown in the detailed figure below, where it is clear that a huge density gradient has appeared close to the body.

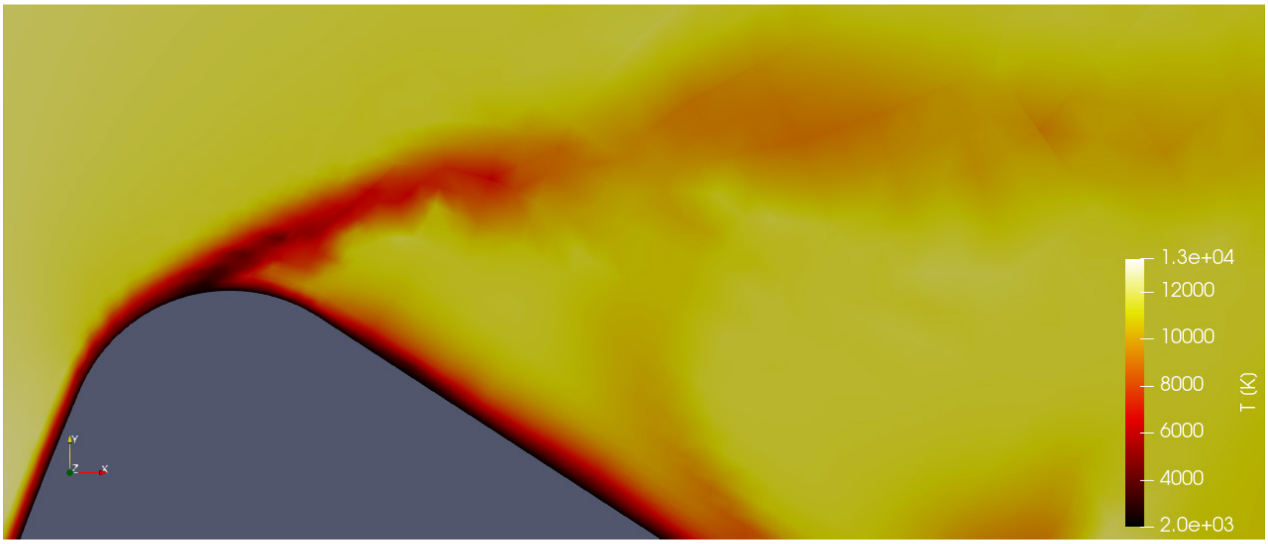


Figure 5.11. Isothermal capsule: Detail of the thermal boundary layer at the interface with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

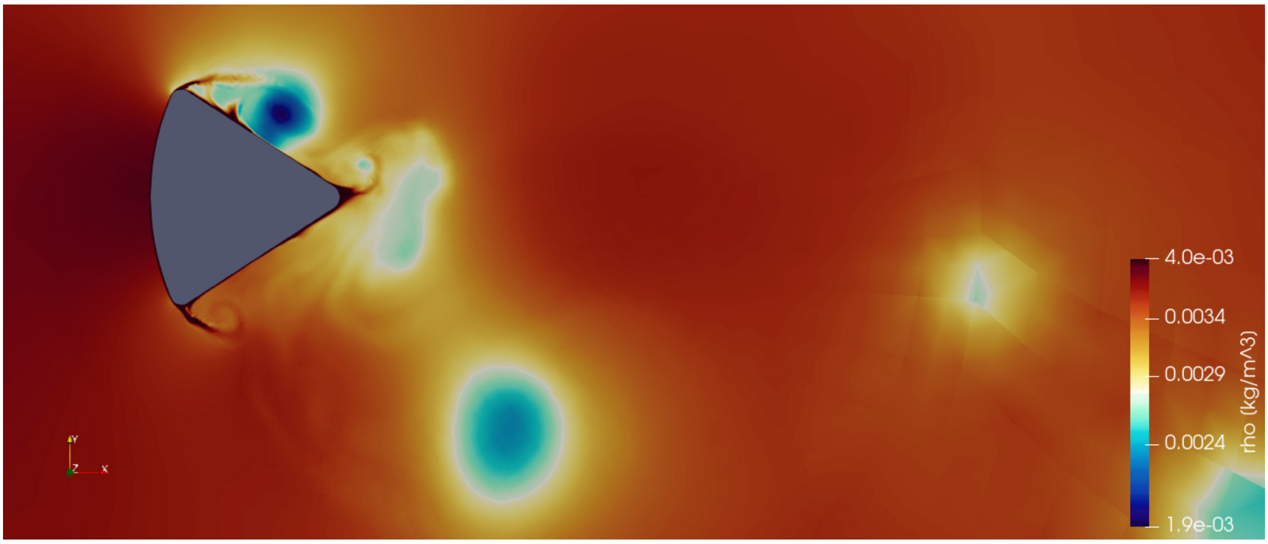


Figure 5.12. Isothermal capsule: Snapshot of the density distribution in the fluid domain (reduced scale) with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

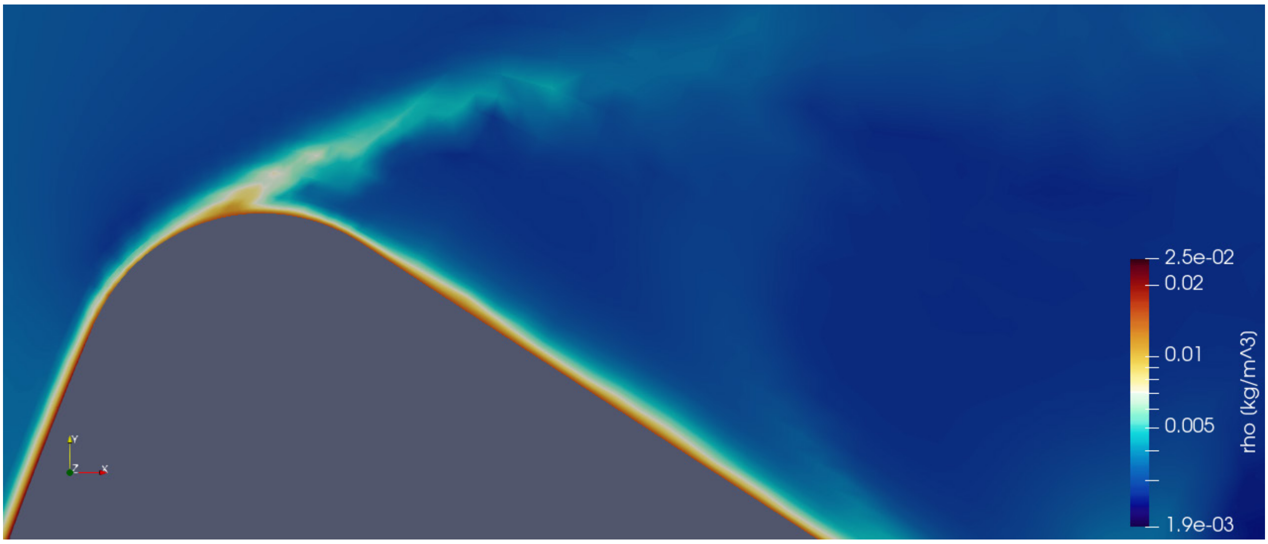


Figure 5.13. Isothermal capsule: Detail of the density distribution at the interface with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

5.3 CHT problem: conductive TPS without radiation

The conjugate heat transfer analysis involves the resolution of the aerodynamic flow field and the solid thermal problem at the same time. This first CHT analysis will be focused on the conductive heat transfer within the TPS of the reentry vehicle. The TPS is the external layer of the capsule and it insulates the interior from the hot air stream. It has a thickness of 10 cm and the selected material has been TUFROC (see Table 2.2 in sec. 2.2), which is a good insulator, $k = 0.1\,\mathrm{W/(m \cdot K)}$, it can withstand high temperatures and it has a high emissivity.

The solution of this problem will be given by the coupling between the fluid and the TPS of the capsule in quasi steady-state conditions. The inside of the capsule will be considered as adiabatic; that is, perfectly insulated. Although this approach is a simplification of reality, for this analysis without thermal radiation, the inside of the capsule is not interesting, due to the temperatures are going to be too high to extract any relevant conclusion. This hypothesis will be improved in following analyses, but in this case it is accurate enough to model the conduction inside the TPS and to verify the coupling between both domains.

The problem will be solved following the CHT approach, explained in sec. 4.1. Once the iterative process has concluded and both domains are in equilibrium, the temperatures and heat fluxes must be the same at both sides of the fluid-solid interface:

$$T_f = T_s, \qquad q_f = k_f \nabla_n T_f = k_s \nabla_n T_s = q_s. \tag{62}$$

The exchange of information between solvers has been established to fit the TFFB (Temperature Forward Flux Back) methodology, as it is schematically represented in Figure 5.14. This simple methodology allows to achieve a convergence state once both domains are in equilibrium. The information is only shared through the boundary conditions and, hence, the use of two different solvers is possible through a programming interface (see sec. 8).

1) The Navier-Stokes equations are solved within the fluid domain.
2) $T_s = T_f$ is used as Dirichlet BC for the solid.
3) The solid heat equation is solved.
4) $q_f = q_s$ is used as Neumann BC for the fluid.
5) Steps 1 to 4 are repeated iteratively.

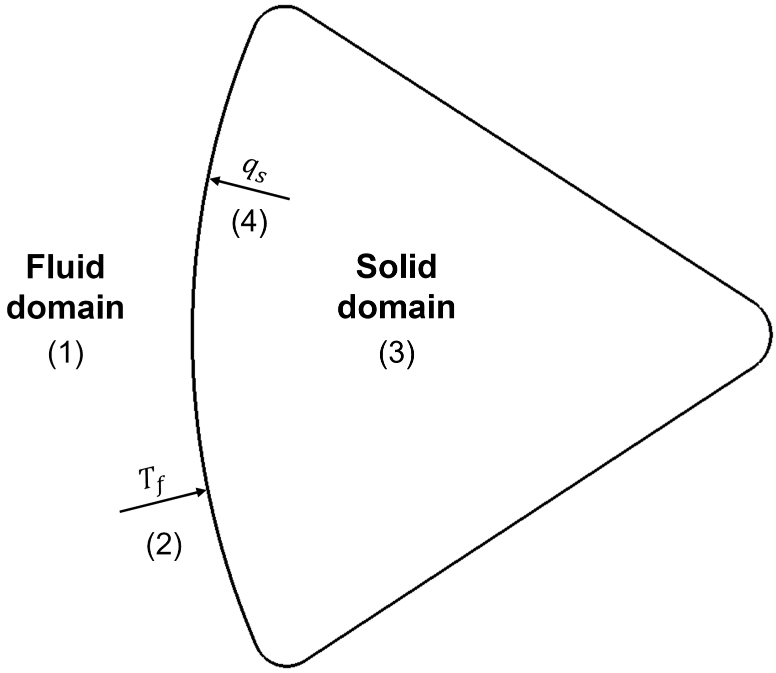


Figure 5.14. CHT approach for the aerothermodynamic problem.

For this analysis, the solution in the fluid domain will be computed using a polynomial order $p = 2$, which is equivalent to a 3rd order method. The use of $p = 2$ instead of $p = 3$ is needed to perform the coupling and the interpolation in the solid correctly, which is required to get the fluxes q_s, due to the limitations of the developed code (see sec. 9).

The Mach number distribution is represented in Figure 5.15, where the result is similar to that of the previous analysis, without TPS. Therefore, the conduction within the TPS of the capsule does not significantly modify the behaviour of the fluid around the reentry vehicle. However, the maximum Mach number has dropped from $M = 1.1$ to $M = 0.98$, as some of the energy of the fluid has been transferred to the capsule. If the inside of the capsule had not been considered as adiabatic, the energy absorbed by the capsule would be higher and, hence, the Mach number should be slightly smaller. However, the difference will be small, due to the conductive fluxes are negligible compared to the heating effect of the convective flow, as it will be shown later.

The temperature distribution is represented in Figure 5.16, where both fluid and solid are represented. The coupling process has been performed correctly, assuring continuity of temperature and heat fluxes along the interface between both domains. The fluid's temperature follows a similar trend compared to the previous case, but the upper side of the capsule is cooler than before. Furthermore, the conduction in the TPS has an homogenising effect on the temperature at the interface; therefore, in this case, there are larger regions with a similar temperature surrounding the capsule.

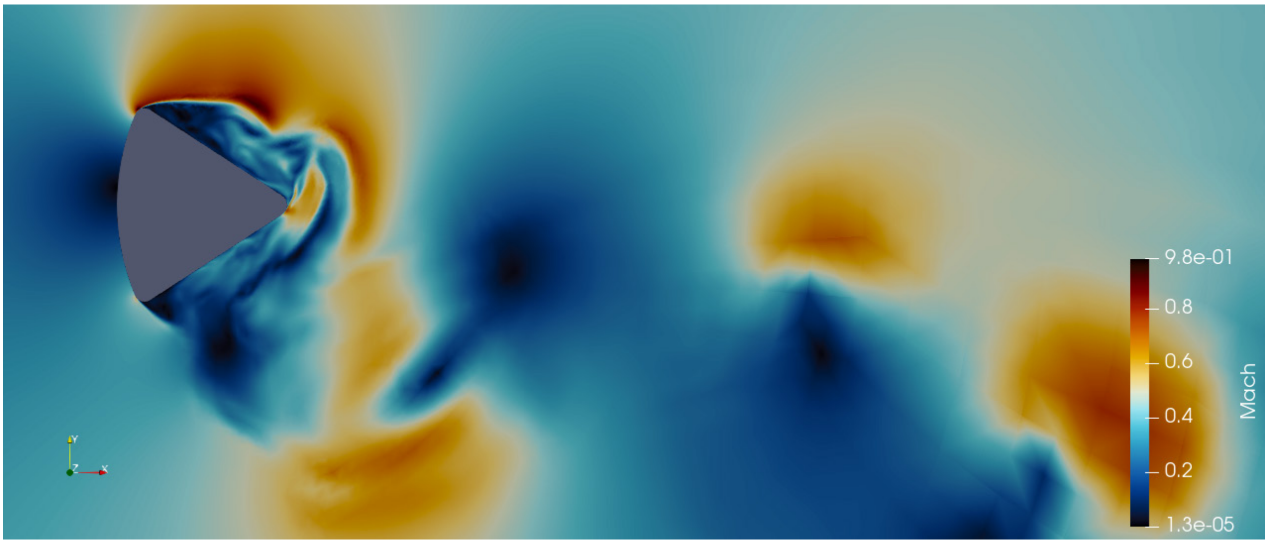


Figure 5.15. CHT with conductive TPS: Snapshot of the Mach number distribution in the fluid domain with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

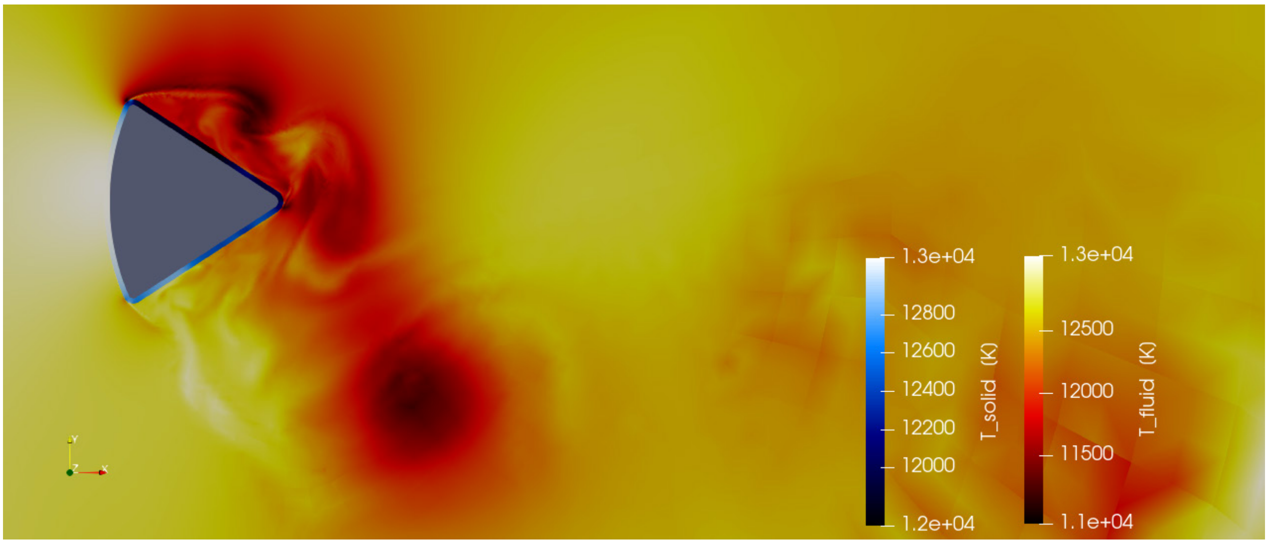


Figure 5.16. CHT with conductive TPS: Snapshot of the temperature distribution in the solid domain (blue scale) and in the fluid domain (red scale) with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

The coupling between both domains is represented with more detail in Figure 5.17. In this figure, it is clear that both domains are in thermal equilibrium and, hence, fluid and solid have the same temperature at each point of the interface.

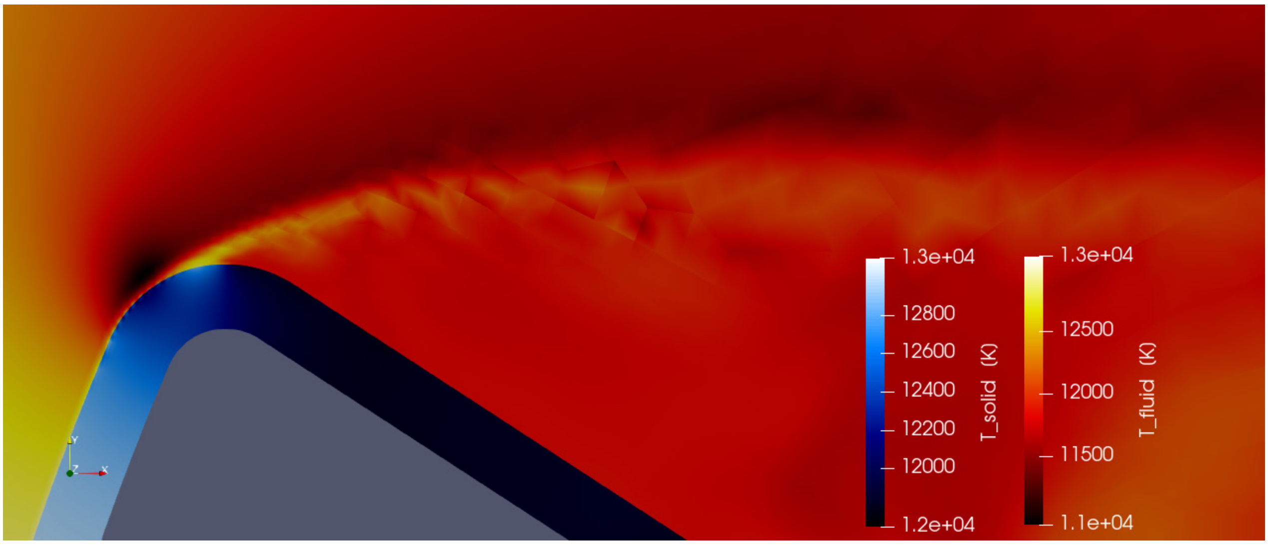


Figure 5.17. CHT with conductive TPS: Detail of the temperature distribution in the solid domain (blue scale) and in the fluid domain (red scale) with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

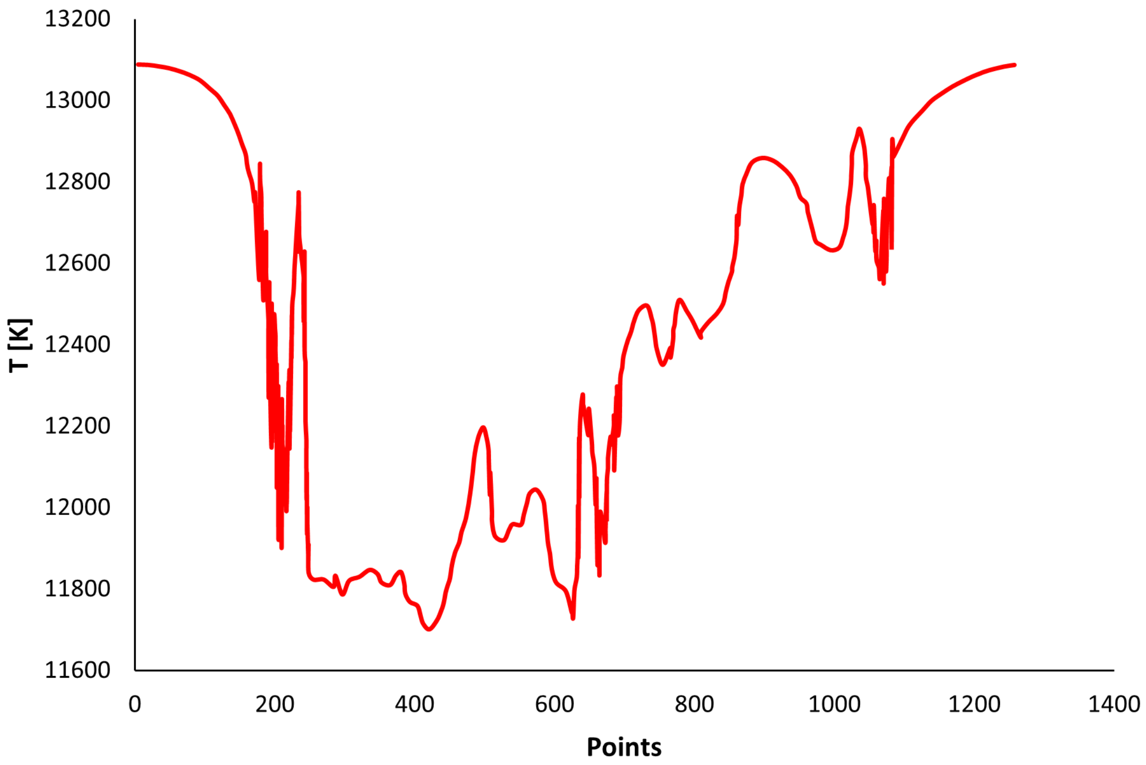


Figure 5.18. CHT with conductive TPS: Temperature distribution along the fluid-solid interface. Samples recorded clockwise, beginning at the front of the capsule.

The temperature distribution at the interface is represented in Figure 5.18, where the maximum temperature of $T = 13089\,\mathrm{K}$ is located on the frontal surface of the capsule, as it was expected. Furthermore, the maximum temperature has dropped $\Delta T = 406\,\mathrm{K}$ in comparison to the analysis without TPS (adiabatic capsule). Therefore, even though the effect of the TPS does not modify the flow field, the energy absorbed by this heat shield and the influence of conduction within the solid domain can significantly reduce the maximum temperature. The minimum temperature has a value of $T = 11702\,\mathrm{K}$, which is also slightly smaller than the 12000 K obtained for the adiabatic analysis. Even with this reduction of temperature, the values are extremely high and the effect of thermal radiation has to be considered to achieve an effective cooling of the capsule.

All these thermal results have been presented considering a situation of equilibrium between both domains. However, as it was explained before, it is impossible to reach a perfect steady-state in presence of a turbulent flow. Therefore, it is necessary to establish a criterion to stop the simulation when a quasi steady-state has been achieved.

In this case, the maximum and minimum temperatures along the interface have been monitored during the simulation process. The first hundreds of iterations, the temperature evolution did not follow any representative pattern, as it is shown in Figure 5.19. However, after many iterations, a pattern appeared in the behaviour of the temperature. This pattern, which is represented in Figure 5.20, is repeated periodically in time. This situation is a quasi steady-state and conclusions can be extracted from this time onward.

This simple criterion can be easily implemented and the equilibrium can be detected through a visual inspection, which is a simple and effective methodology. It is interesting to notice that the maximum temperature is almost constant over time. This behaviour is due to the maximum value is reached at the front of the capsule, where the effect of turbulence is not important and, therefore, it does not significantly change over time.

On the other hand, the minimum temperature, which appears at the rear of the capsule, where the flow is highly turbulent, varies in time following a periodic pattern.

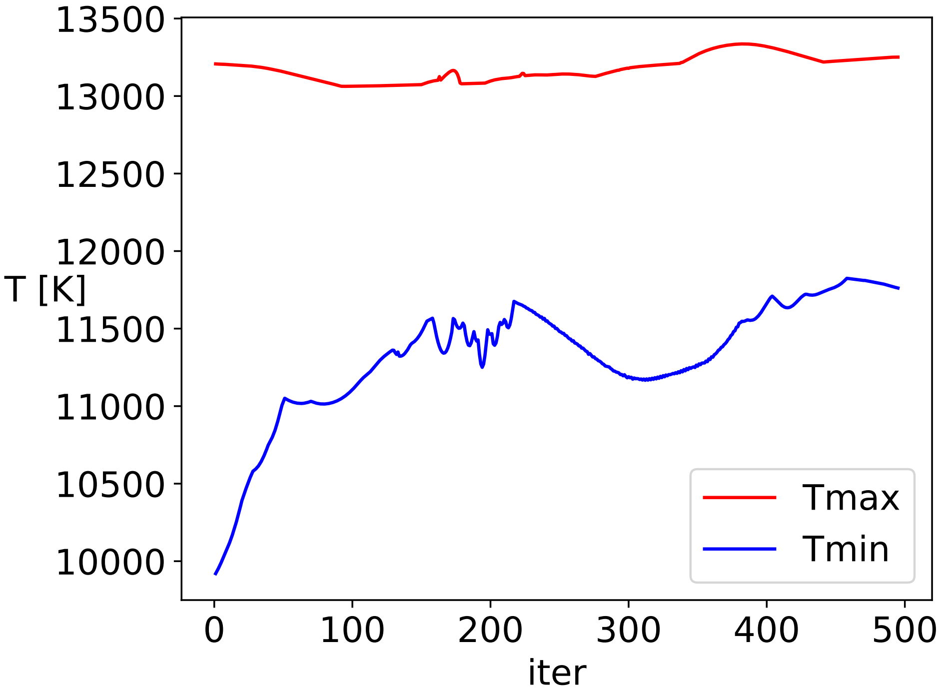


Figure 5.19. CHT with conductive TPS: Maximum and minimum temperature evolution along the fluid-solid interface as a function of the simulation time (in hundreds of iterations).

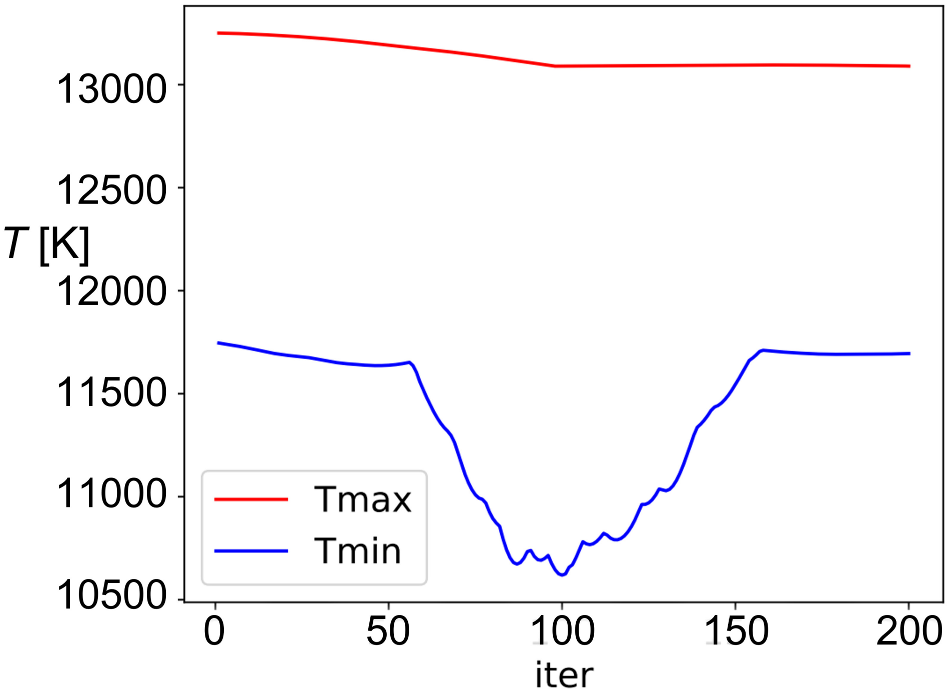


Figure 5.20. CHT with conductive TPS: Maximum and minimum temperature evolution along the fluid-solid interface as a function of the simulation time (in hundreds of iterations). Final pattern used as a stop criterion.

The pressure and density maps are represented in Figures 5.21 and 5.22. Although the general behaviour is similar to the previous case, there is a low pressure and low density region near the upper side of the capsule which was not present in the analysis without TPS. Downstream, the distributions have not shown any significant variation.

The maximum recorded pressure and density are $p = 14818\,\mathrm{Pa}$ and $\rho = 3.94 \cdot 10^{-3}\,\mathrm{kg/m^3}$ respectively, which are almost identical to the values obtained in the adiabatic analysis. The minimum density is equal to its previous value of $2 \cdot 10^{-3}\,\mathrm{kg/m^3}$.

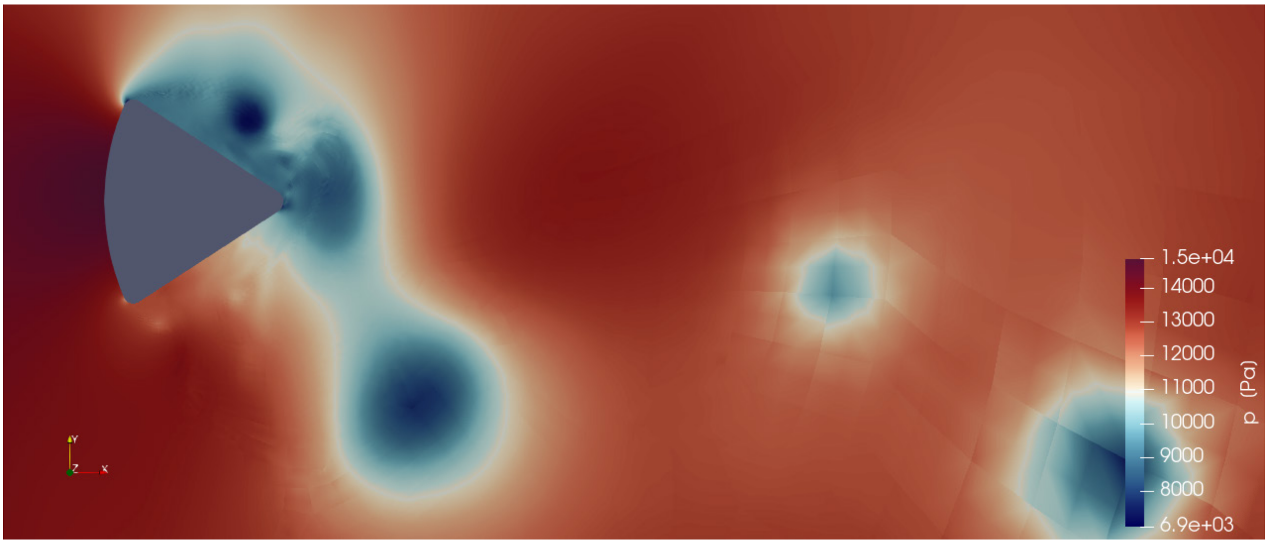


Figure 5.21. CHT with conductive TPS: Snapshot of the pressure distribution in the fluid domain with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

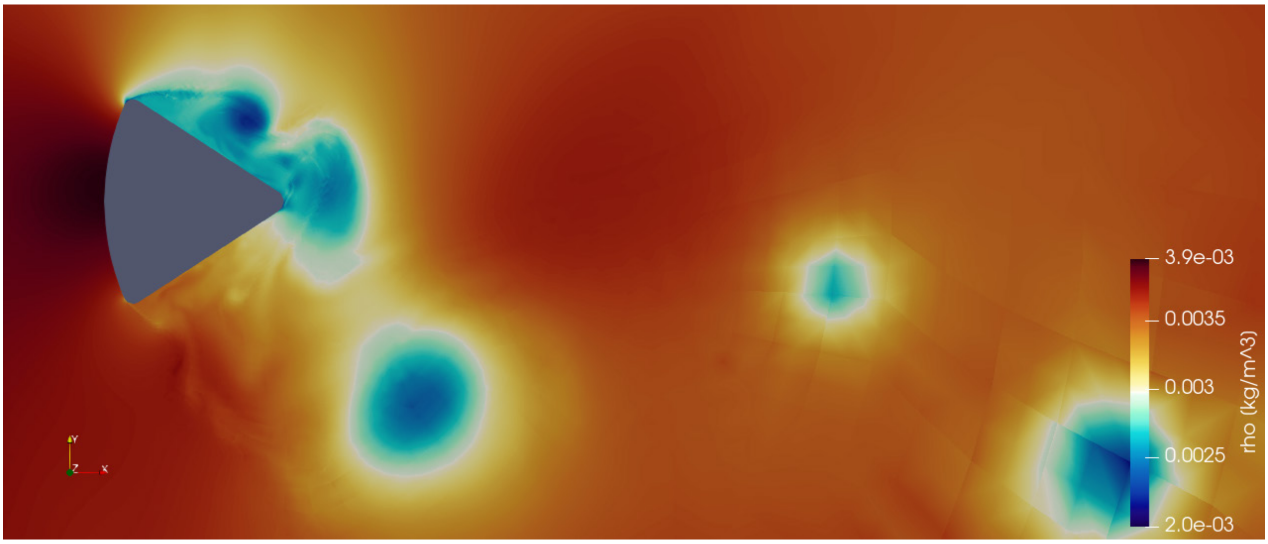


Figure 5.22. CHT with conductive TPS: Snapshot of the density distribution in the fluid domain with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

However, the minimum pressure has a value of 6325 Pa, noticeably lower to the previous analysis. This result was expected, as the temperature recorded in the region close to the upper side of the capsule, where the pressure is minimum, is lower and the perfect gas assumption has been considered in the whole research.

Finally, the velocity field is represented in Figures 5.23 and 5.24. Following the same trend that the Mach number, the maximum velocity is smaller than the value obtained for the adiabatic case. However, the general behaviour is very similar to the previous analysis. It is interesting to highlight the recirculating region present over the upper side of the capsule, where the flow separation is more important in this case.

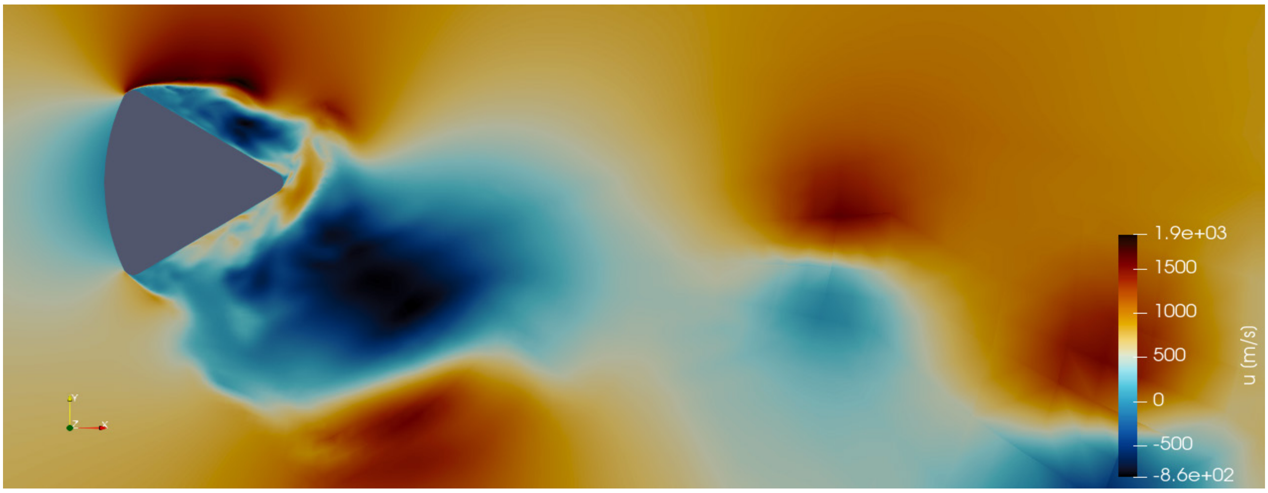


Figure 5.23. CHT with conductive TPS: Snapshot of the velocity field, following the x axis, in the fluid domain with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

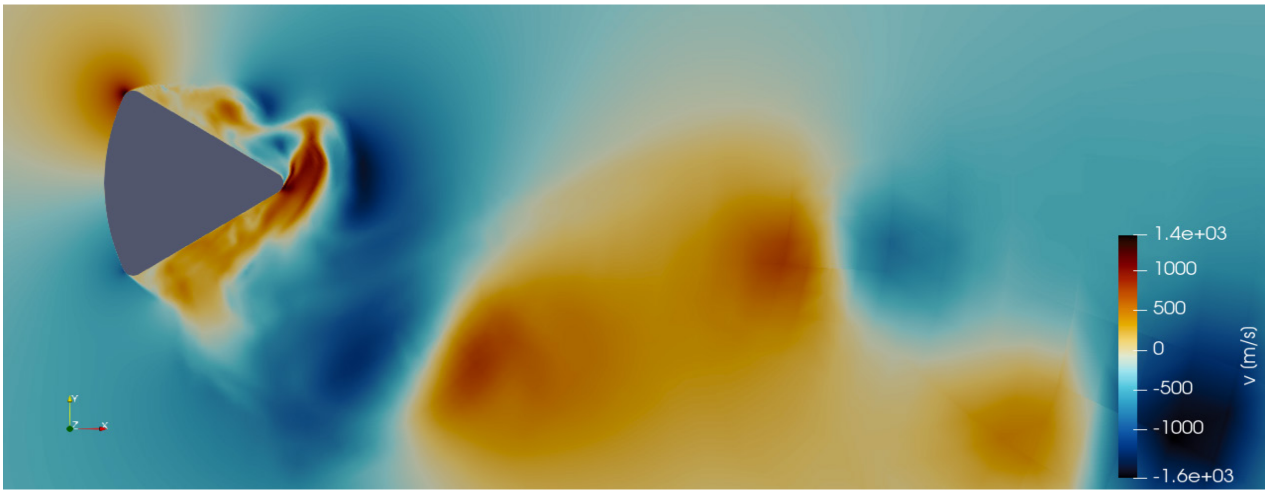


Figure 5.24. CHT with conductive TPS: Snapshot of the velocity field, following the y axis, in the fluid domain with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

5.4 CHT problem: radiative TPS without conduction

The second CHT analysis is the other face of the previous problem. In this case, the interest lies on the effect of thermal radiation and the possibility of using a radiative heat shield to cool the capsule and to assure the survival of the payload during the reentry.

Conduction in the solid domain will not be considered in this analysis, but the mean conductive heat fluxes are expected to be small in comparison with the radiative fluxes, as it will be shown later. Therefore, the solution should not significantly change using this simplification. In this case, the equilibrium at the boundary is given by the following expressions:

$$T_f = T_s, \qquad q_f = k_f \nabla_n T_f = \alpha_s E_s \frac{A_f}{A_c} + \alpha_{IR} F_{c\to E} \varepsilon_E \sigma T_E^4 - \varepsilon_{IR} \sigma T_s^4 = -q_{rad}, \tag{63}$$

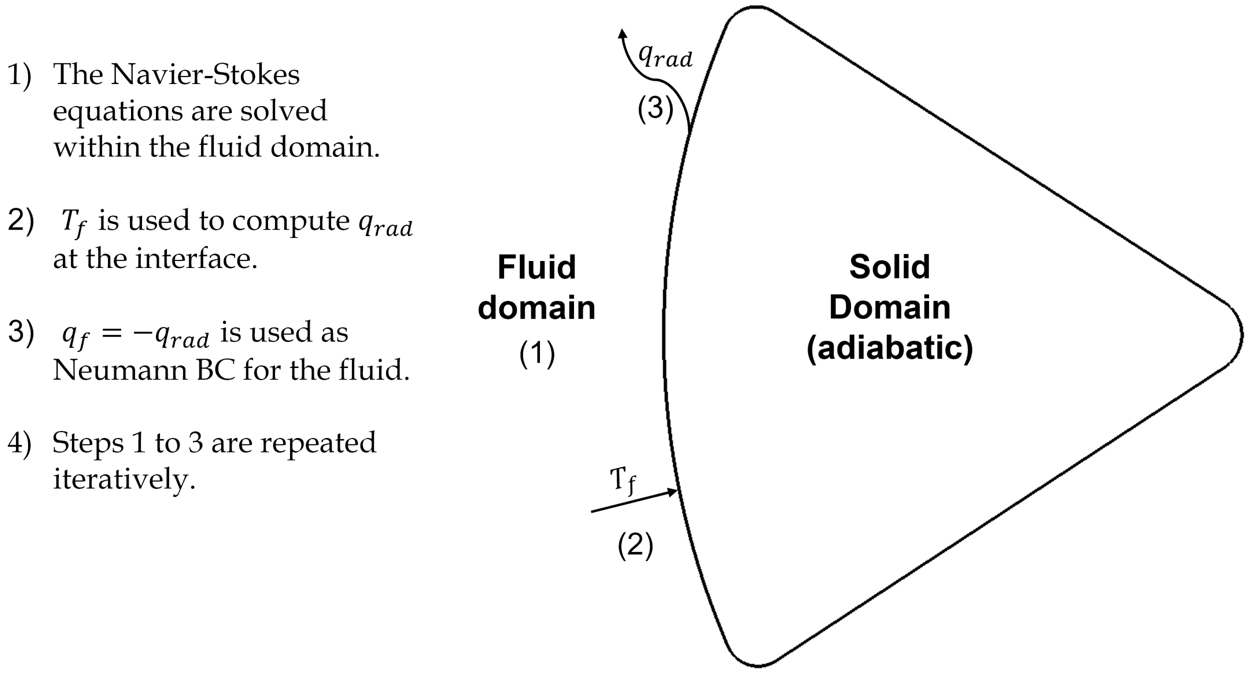
where the radiative flux was described in detail in sec. 2.1.1.

For this first approach to thermal radiation, the radiation from the Sun and the Earth can be discounted as their value will be negligible in relation to the radiation emitted by the capsule, as it will be demonstrated through this analysis. Therefore, the previous expression can be simplified:

$$T_f = T_s, \qquad q_f = k_f \nabla_n T_f = -\varepsilon_{IR} \sigma T_s^4 = -q_{rad}. \tag{64}$$

In this case, the solver MFEM, used to solve the heat equation within the solid domain, will not be necessary as the thermal radiation can be computed directly from the CFD solver. The methodology followed in the present analysis is represented schematically in Figure 5.25. This analysis allows to have into account the radiative heat shield of the capsule, which is the major contribution to the heat fluxes at the interface, with only one CFD solver. Therefore, this approach can be used if the coupling with another solver for the solid domain is not possible.

The solution obtained in this case has been computed using the solution obtained for the isothermal capsule as a restarting point. As it was commented before, this approach improves the stability of the problem, which is essential if the thermal radiation is involved due to the huge heat fluxes that are generated on the boundaries may create a numerical divergence during the simulation (see sec. 9).



1) The Navier-Stokes equations are solved within the fluid domain.
2) T_f is used to compute q_{rad} at the interface.
3) $q_f = -q_{rad}$ is used as Neumann BC for the fluid.
4) Steps 1 to 3 are repeated iteratively.

Figure 5.25. Approach for the aerothermodynamic problem: capsule with a radiative heat shield.

Thermal radiation is emitted from the capsule, effectively reducing the temperature at the interface until a new equilibrium is reached. This behaviour is represented in Figure 5.26. When the simulation starts, the hot air flow increases the temperature around the capsule, due to the isothermal BC is not applied in this case. Then, the radiative heat shield is activated and the temperature of the capsule is progressively reduced. The simulation has been extended some iterations beyond the scale of the figure, but the temperature follows a similar trend.

The Mach number distribution is represented in Figure 5.27, where the solution has been computed using a $p = 2$ polynomial order. This result is similar to the Mach distribution obtained in previous analyses. Again, as it was explained for the isothermal case, the temperature distribution at the interface does not significantly modify the Mach number and the velocity field of the fluid, due to the main variations of the temperature are constrained within the thermal boundary layer.

The angle of attack of $-18°$ creates the asymmetry of the flow that is shown in the figure, which generates a positive lift over the reentry capsule, as it will be demonstrated later through the pressure distribution.

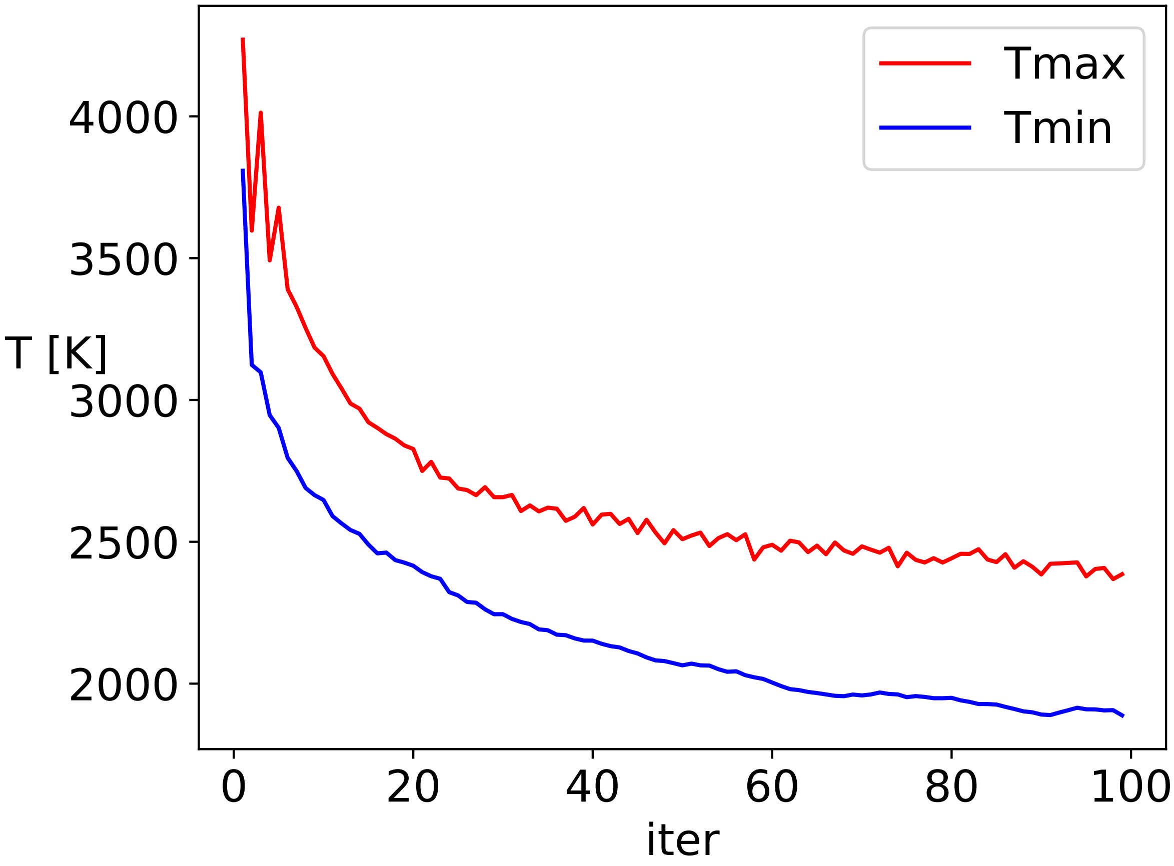


Figure 5.26. Radiative TPS: Maximum and minimum temperature evolution along the fluid-solid interface as a function of the simulation time (in hundreds of iterations).

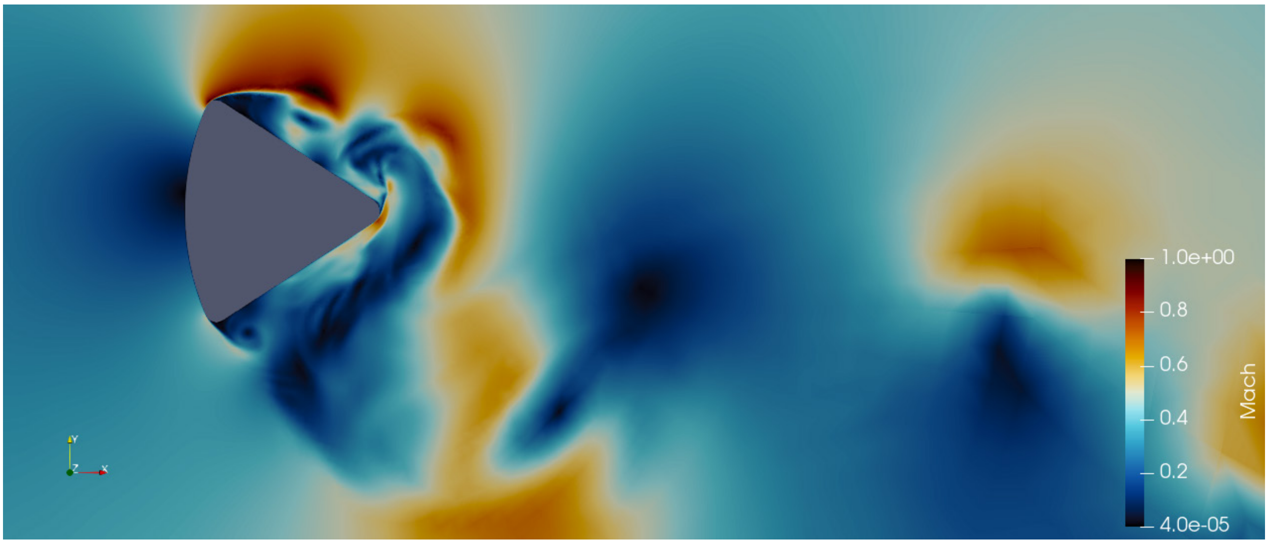


Figure 5.27. Radiative TPS: Snapshot of the Mach number distribution in the fluid domain with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

The temperature distribution is shown in Figure 5.28. The temperature of the flow field reminds of the distribution obtained for the isothermal case, where the whole capsule was at a temperature of 2000 K. Even though the effect of radiation is not uniform, the mean temperature is close to the value of 2000 K, as it is shown in Figure 5.29, where the temperature along the interface is represented. Therefore, an isothermal boundary condition is more real to simulate the effect of thermal radiation than an adiabatic boundary condition.

The cooler air surrounding the capsule is transported downstream through the bigger eddies, whose temperature is smaller than the inflow temperature. However, the main temperature gradients are bounded into the boundary layer, which is represented in detail in Figure 5.30. This figure shows clearly the strong variation of the temperature at the front of the capsule, where the temperature is reduced one order of magnitude in a small distance of some centimetres. Furthermore, the flow separation is well-defined over the upper side of the body, showing how the cooler flow is transported far from the capsule.

Unlike the adiabatic analysis, where the temperature was too high to be withstood by any known material, the radiative heat shield is able to effectively reduce the temperature around 10000 K. In this case, the maximum and minimum temperatures at the interface have a value of 2610 K and 1127 K respectively. These values are more reasonable and there are some refractory materials which can work at this temperature levels. Hence, the use of a radiative heat shield is appropriate to be used as a thermal protection system of a reentry capsule at hypersonic velocities.

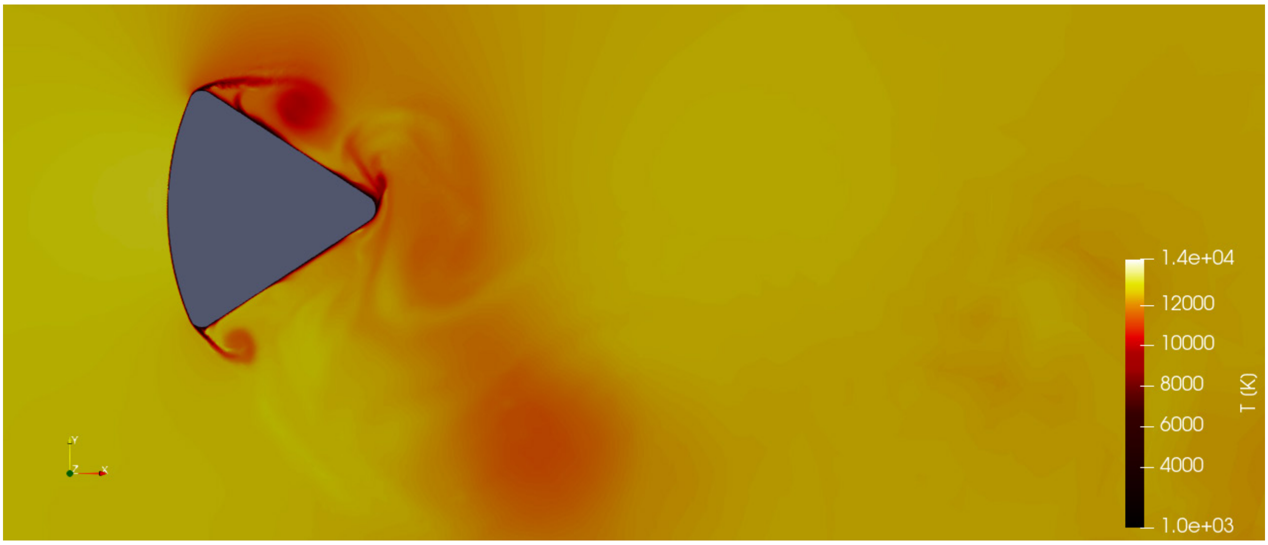


Figure 5.28. Radiative TPS: Snapshot of the temperature distribution in the fluid domain with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

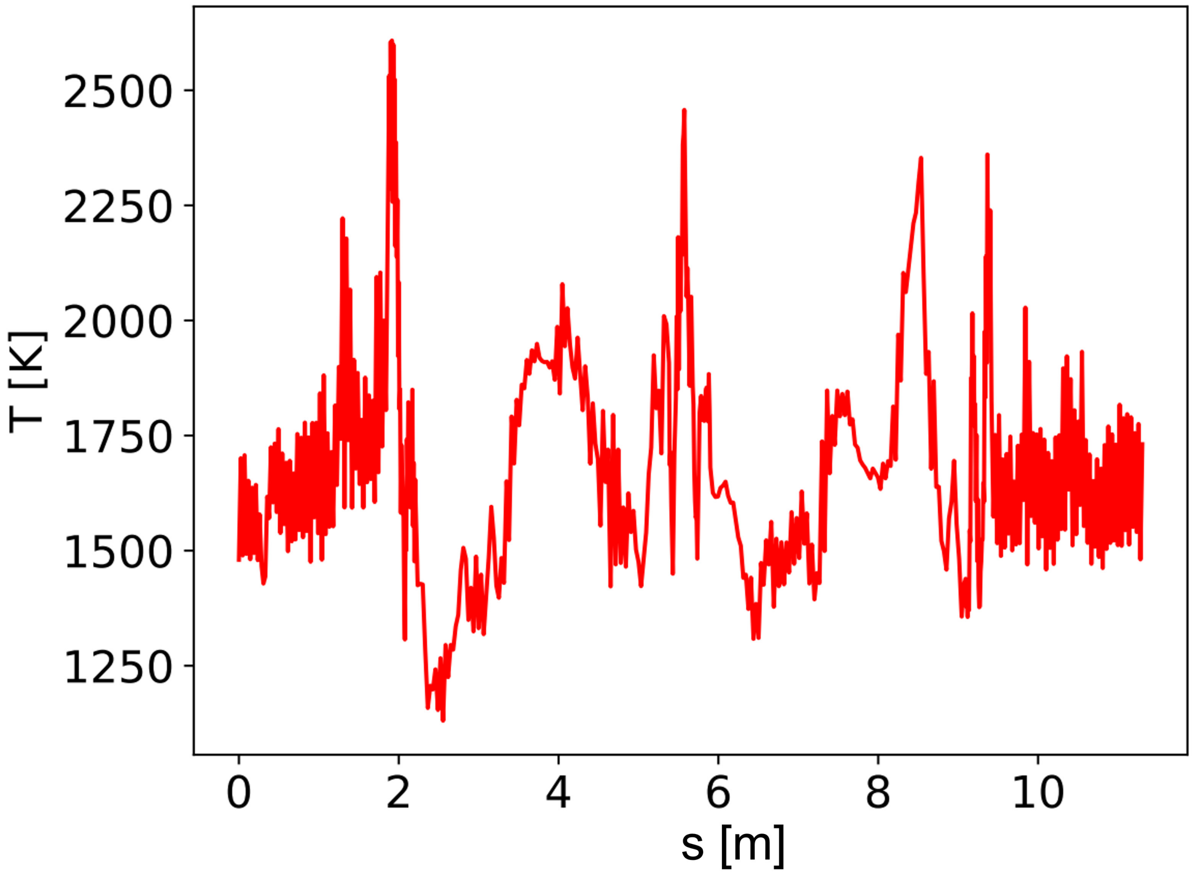


Figure 5.29. Radiative TPS: Temperature distribution along the fluid-solid interface. Samples recorded clockwise, beginning at the front of the capsule.

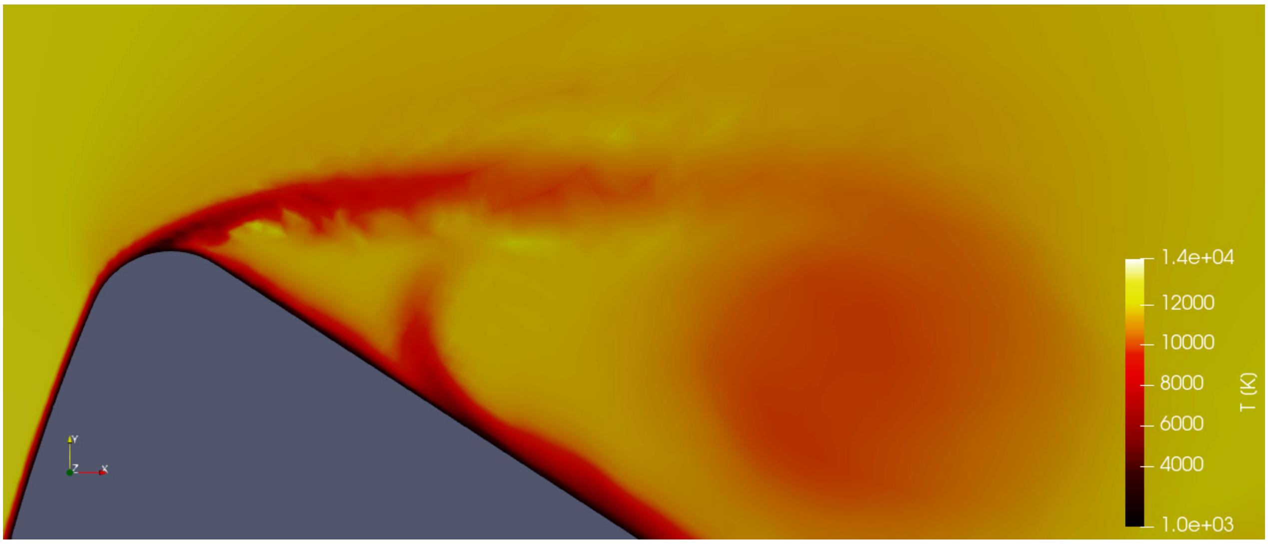


Figure 5.30. Radiative TPS: Detail of the temperature distribution in the fluid domain with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

Once the solution for the temperature has been computed, it is possible to plot the final radiative heat flux along the interface, which is represented in Figure 5.31. This distribution will be useful to decide whether the effect of the Earth and the Sun, that has not been taken into account for this analysis, is negligible or not and, hence, if this additional effect should be considered in the following analyses.

The radiative heat flux has a maximum value of $2.367 \cdot 10^6\,\mathrm{W/m^2}$ and a minimum value of $7.717 \cdot 10^4\,\mathrm{W/m^2}$.

The estimation of the contribution from the Earth and the Sun is computed below:

$$q_{Sun} = \alpha_s E_s \frac{A_f}{A_c} = 370.87\,\frac{\mathrm{W}}{\mathrm{m}^2}, \qquad q_{Earth} = \alpha_{IR} F_{c\rightarrow E} \varepsilon_E \sigma T_E^4 = 80.88\,\frac{\mathrm{W}}{\mathrm{m}^2}. \tag{65}$$

These results show that the Sun and the Earth are completely negligible for this analysis, due to their effect is several orders of magnitude below the heat flux radiated by the capsule.

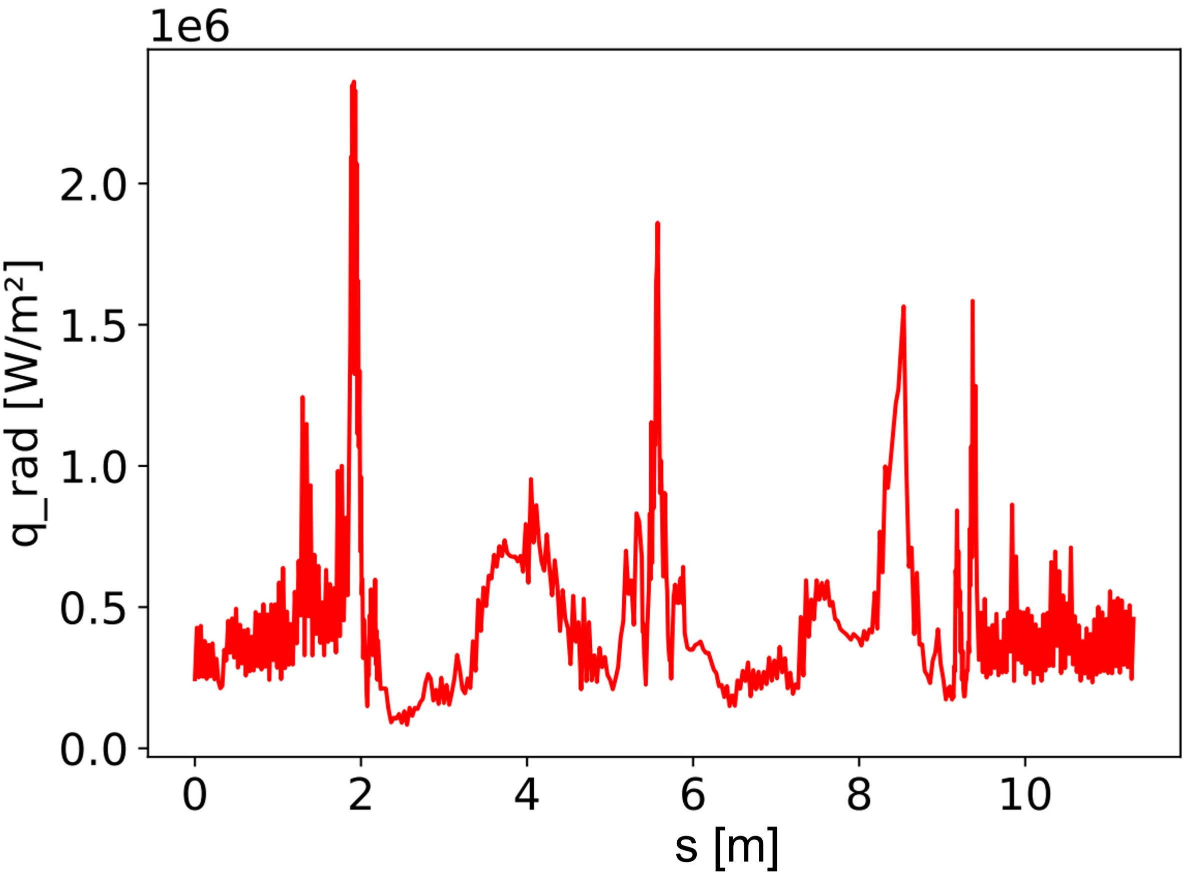


Figure 5.31. Radiative TPS: Radiative heat flux along the fluid-solid interface. Samples recorded clockwise, beginning at the front of the capsule.

The pressure and the density distributions are represented in Figures 5.32 and 5.33. These distributions have a common behaviour with the isothermal analysis, due to the temperatures at the boundary are very similar in both cases.

There is an important low pressure region over the upper side of the capsule, which will produce a positive lift, as it was expected with a negative angle of attack.

The density map has been plotted in a reduced scale to show this distribution in a clearer way, due to there is a huge density gradient near the body. The ranges of the pressure and the density are $5662 - 15056\,\text{Pa}$ and $2 \cdot 10^{-3} - 3.71 \cdot 10^{-2}\,\text{kg/m}^3$, respectively.

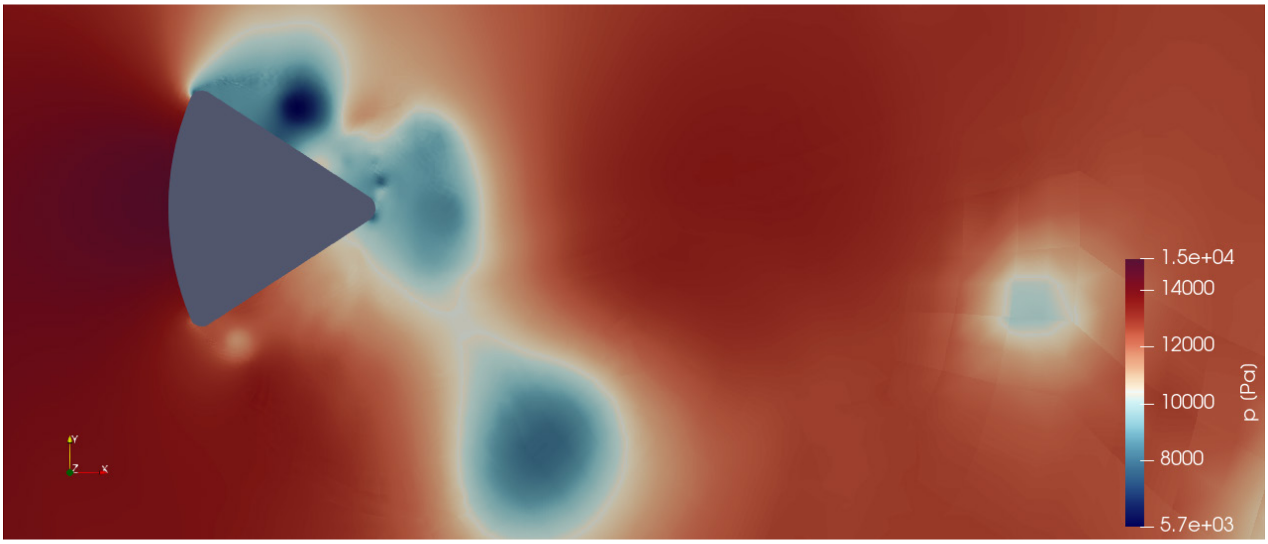


Figure 5.32. Radiative TPS: Snapshot of the pressure distribution in the fluid domain with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

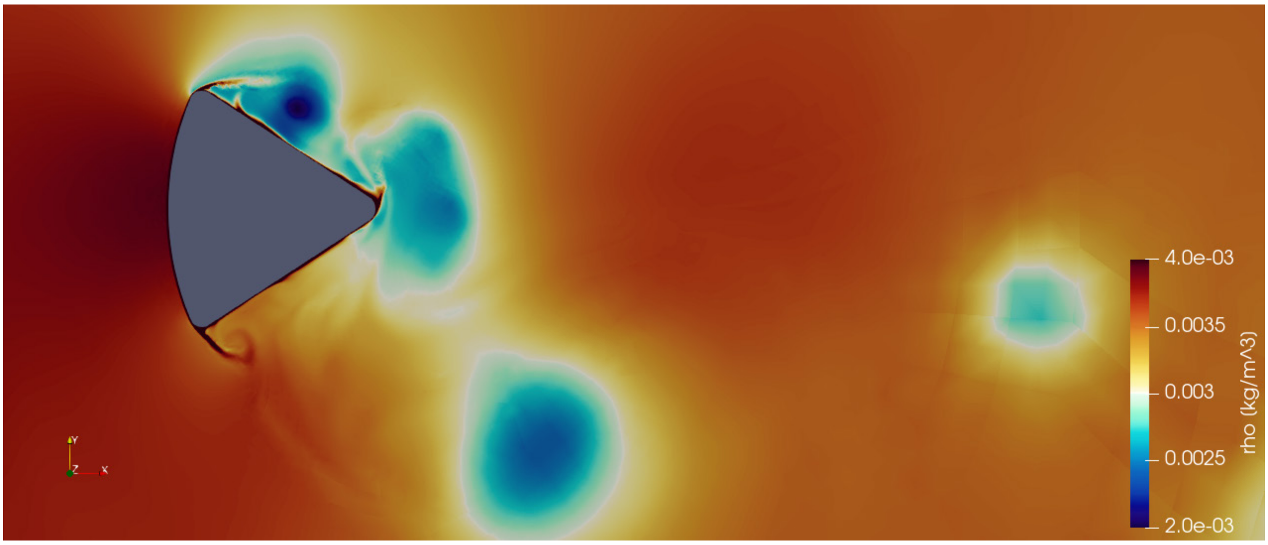


Figure 5.33. Radiative TPS: Snapshot of the density distribution in the fluid domain (reduced scale) with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

Finally, the velocity field is represented in Figures 5.34 and 5.35. These figures allow to acquire a deeper understanding of the flow field, as a complement to the Mach number distribution. These results show an important recirculating area at the rear part of the capsule and a stream separation region on the upper side.

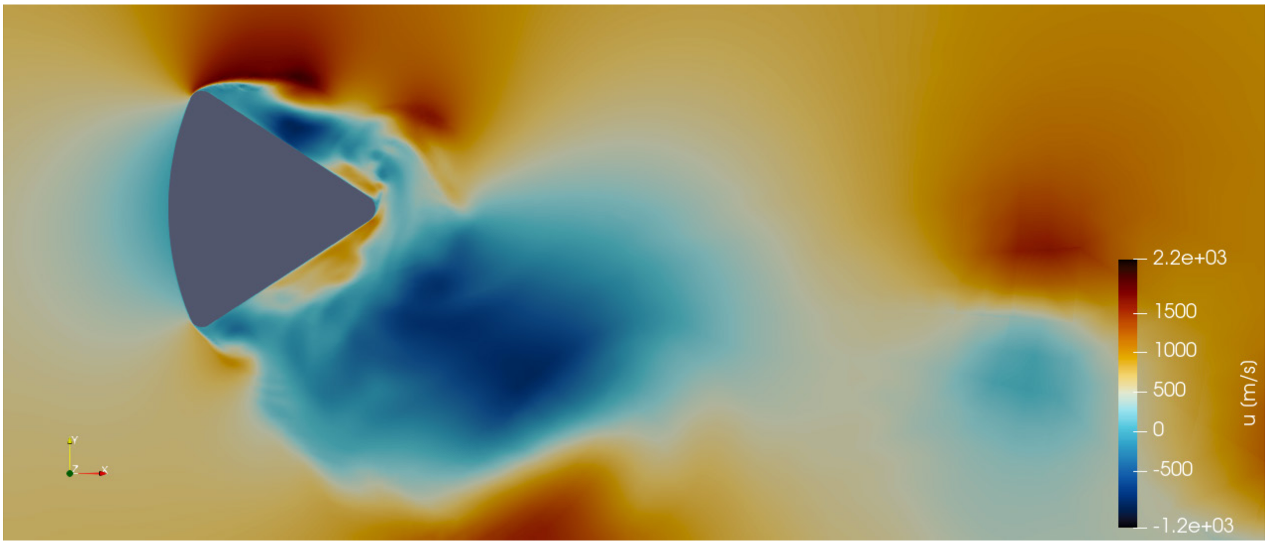


Figure 5.34. Radiative TPS: Snapshot of the velocity field, following the x axis, in the fluid domain with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

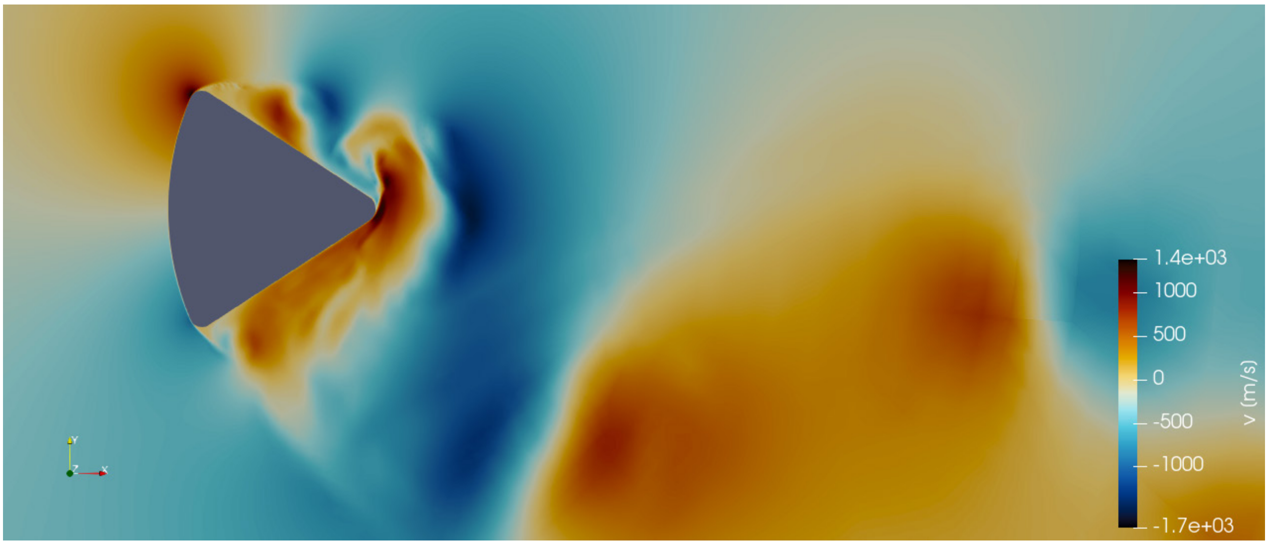


Figure 5.35. Radiative TPS: Snapshot of the velocity field, following the y axis, in the fluid domain with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

The last two Figures 5.36 and 5.37 show a zoom out vision of the same velocity field. In this case, the flow structure downstream is represented. There are four main eddies which define the path for the air stream. The maximum velocities are located near the upper side of the capsule and surrounding these eddies downstream.

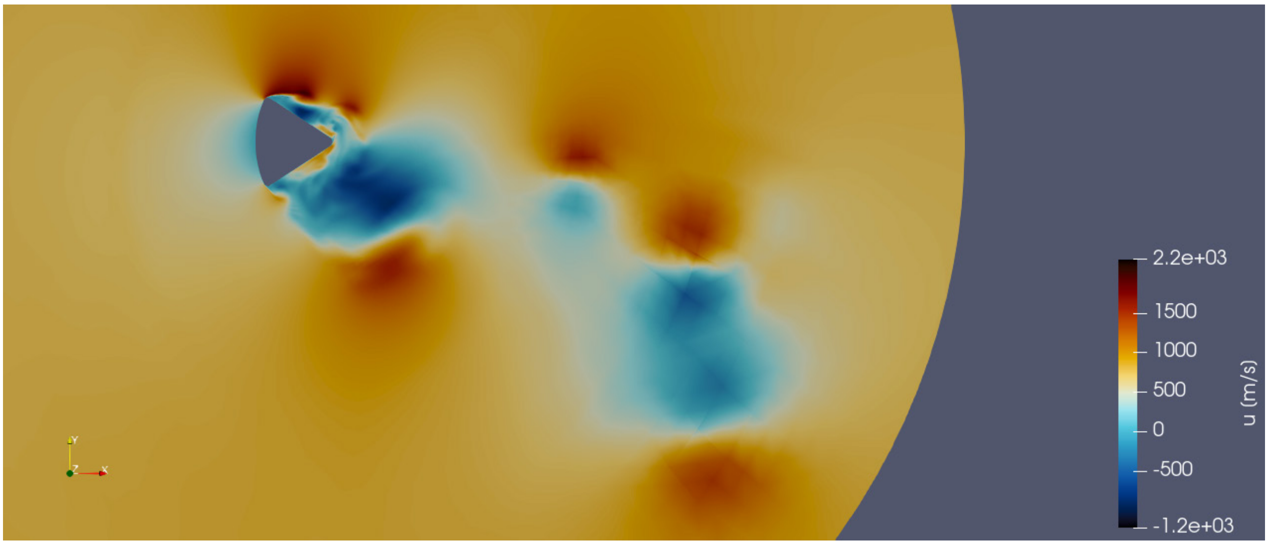


Figure 5.36. Radiative TPS: Snapshot of the velocity field, following the x axis, in the fluid domain (downstream view) with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

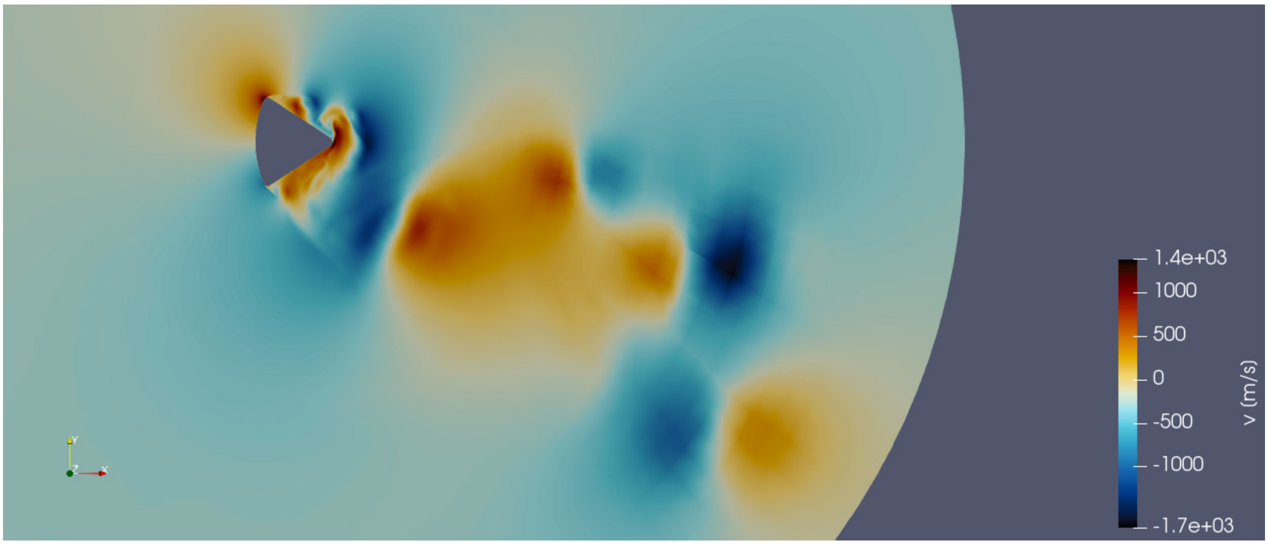


Figure 5.37. Radiative TPS: Snapshot of the velocity field, following the y axis, in the fluid domain (downstream view) with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

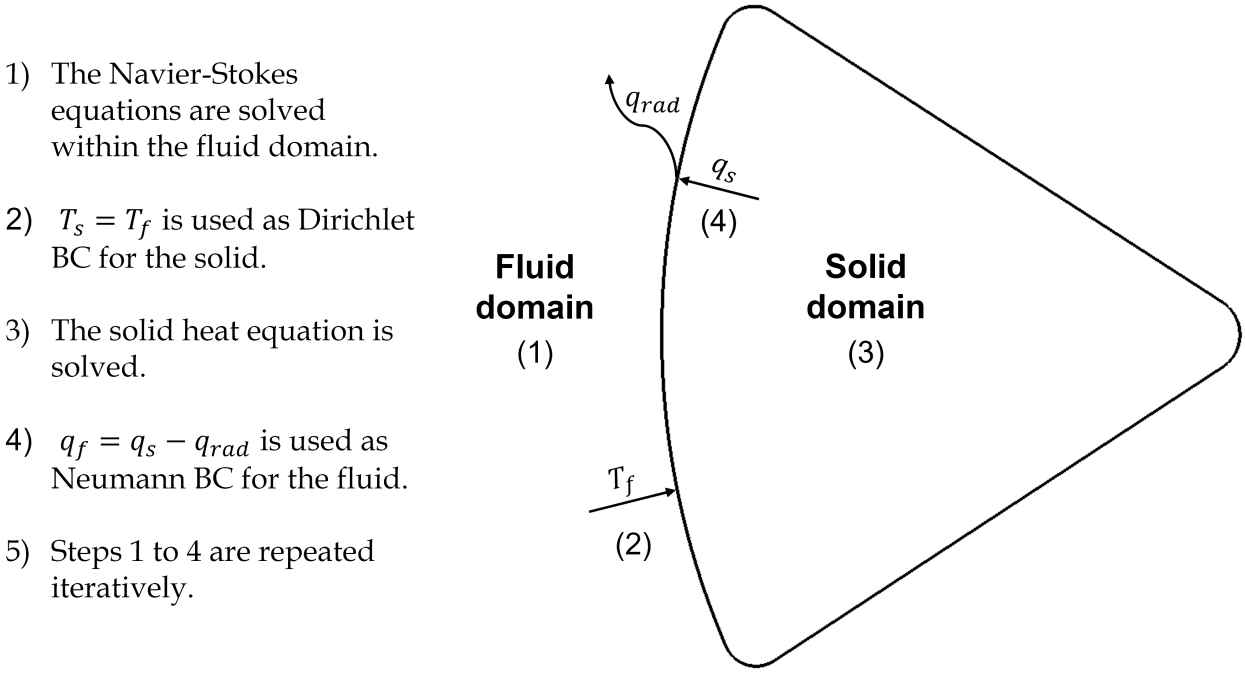
5.5 CHT problem: radiative and conductive TPS

The last conjugate heat transfer problem to be analysed involves the complete coupling problem between the air flow and the reentry capsule. In this case, the TPS of the capsule will be considered as a real heat shield, made of TUFROC (see sec. 2.2); and therefore, it will be conductive, with a thermal conductivity of $k = 0.1\,\mathrm{W/(m \cdot K)}$, and radiative, with an emissivity of $\varepsilon = 0.9$. The thermal equilibrium at the fluid-solid interface is defined by the following expressions:

$$T_f = T_s, \qquad q_f = k_f \nabla_n T_f = k_s \nabla_n T_s - \varepsilon_{IR} \sigma T_s^4 = q_s - q_{rad}. \tag{66}$$

In this case, the thermal radiation emitted by the Sun and the Earth has not been taken into account, due to their value is negligible, as it was demonstrated in the previous analysis.

The selected methodology to solve the CHT problem involves the exchange of information between both solvers: MFEM and HORSES3D, following the approach previously defined in sec. 4.1. The procedure is represented schematically in Figure 5.38. For the complete resolution of the problem, the Neumann BC for the fluid depends on both radiative and conductive heat fluxes computed within the solid domain.



1) The Navier-Stokes equations are solved within the fluid domain.
2) $T_s = T_f$ is used as Dirichlet BC for the solid.
3) The solid heat equation is solved.
4) $q_f = q_s - q_{rad}$ is used as Neumann BC for the fluid.
5) Steps 1 to 4 are repeated iteratively.

Figure 5.38. Approach for the aerothermodynamic problem: capsule with a radiative and conductive heat shield.

The inside of the capsule, which is not part of the TPS, will be considered as isothermal for this analysis, at a constant temperature of 293 K. This simplification avoids to have a detailed model of the inside of the capsule, where there are several devices and materials, or even humans in manned missions. However, the isothermal BC is expected to be appropriate, due to the TPS is a very good insulator and the temperature at the inside should not be significantly modified during the reentry, as it will be shown through this analysis.

This simulation will be restarted from the previous analysis: radiative TPS without conduction. The conductive heat fluxes are expected to be smaller than the radiative fluxes and, hence, the solution will be similar to the previous case, which will improve the convergence.

The Mach number, the pressure, the density and the velocity field have not noticeably changed in this case. Therefore, this analysis will be focused on the temperature distribution and the thermal coupling between both domains.

The temperature distribution is represented in Figure 5.39. As expected, the temperature of the fluid is very similar to the analysis of the radiative TPS without conduction. The thermal boundary layer is defined by the effect of thermal radiation at the interface, while the conduction inside the solid is only important to define the internal heating of the payload and the TPS. This simulation has been stopped when the temperature at the boundary was quasi steady and it was only slightly modified by periodic fluctuations, due to the effect of turbulence.

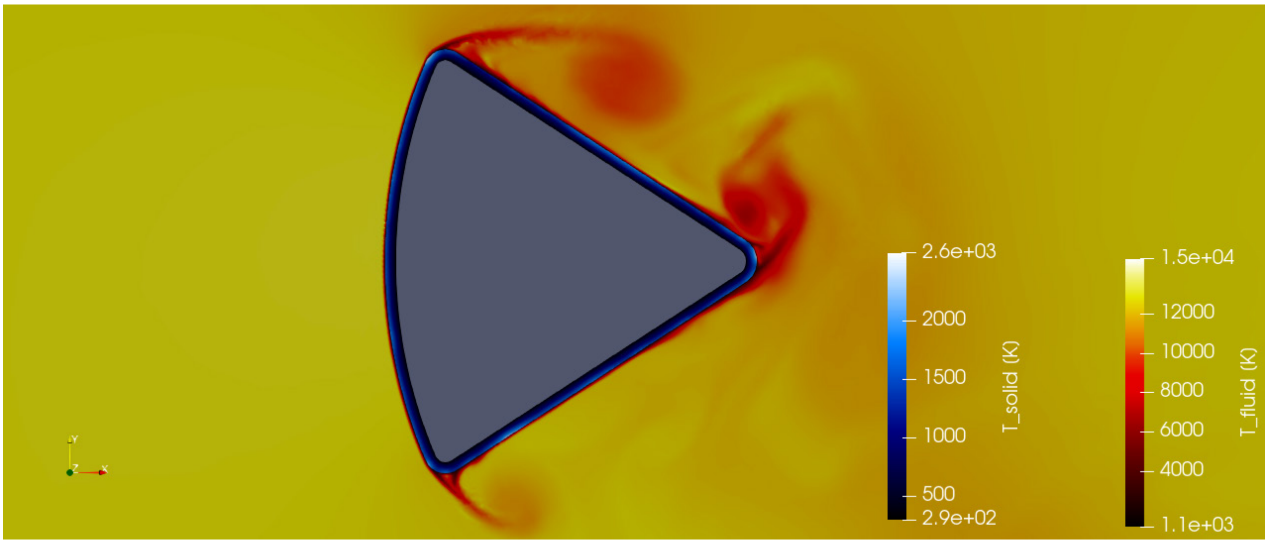


Figure 5.39. Radiative and conductive TPS: Snapshot of the temperature distribution in the solid domain (blue scale) and in the fluid domain (red scale) with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

The temperature along the interface is represented in Figure 5.40. The maximum and minimum recorded temperatures at the interface have a value of 2580 K and 1111 K, respectively. Although there are some differences, this result is almost identical to the case without conduction, which means that the conductive fluxes are small and their effect does not modify the general thermal behaviour.

The material used for the TPS, called TUFROC, can withstand up to 2000 K in operating conditions. In this case, the maximum temperature has exceeded that threshold. However, this analysis has been performed considering steady-state conditions inside the solid domain and, hence, thermal inertia has not been taken into account. Furthermore, the real shock wave surrounding the capsule could lead to the generation of cooler regions around the capsule and the maximum temperature would be even smaller.

Therefore, this result can be considered as a worst case analysis and the real flight conditions should be less severe. However, the real conditions should be simulated and tested to assure the survival of the payload before this approach is used in real missions.

The thermal coupling is shown in detail in Figure 5.41. Here, the effect of radiation is clear, which leads to the generation of a thermal boundary layer and the temperature is reduced one order of magnitude in relation to the reference temperature at the inflow.

The selected CHT approach (see sec. 4.1) assures a perfect match of the temperature distribution at the interface in both domains. On the other hand, the iterative process achieves an equilibrium of the heat fluxes at the interface, which completes the resolution of the CHT problem.

Therefore, the objective of simulating a coupled problem of a reentry capsule, with a radiative and conductive heat shield, in a flow field similar to the real reentry conditions, has been successfully accomplished in this research.

Although more analyses would be required to validate this thermal protection system, the use of thermal radiation as a mechanism to cool a reentry capsule is very effective, even at high Mach numbers and hypersonic conditions. Furthermore, the use of an insulated structure as a TPS, instead of an ablative material, allows to reuse the reentry vehicle and to reduce the cost of future missions.

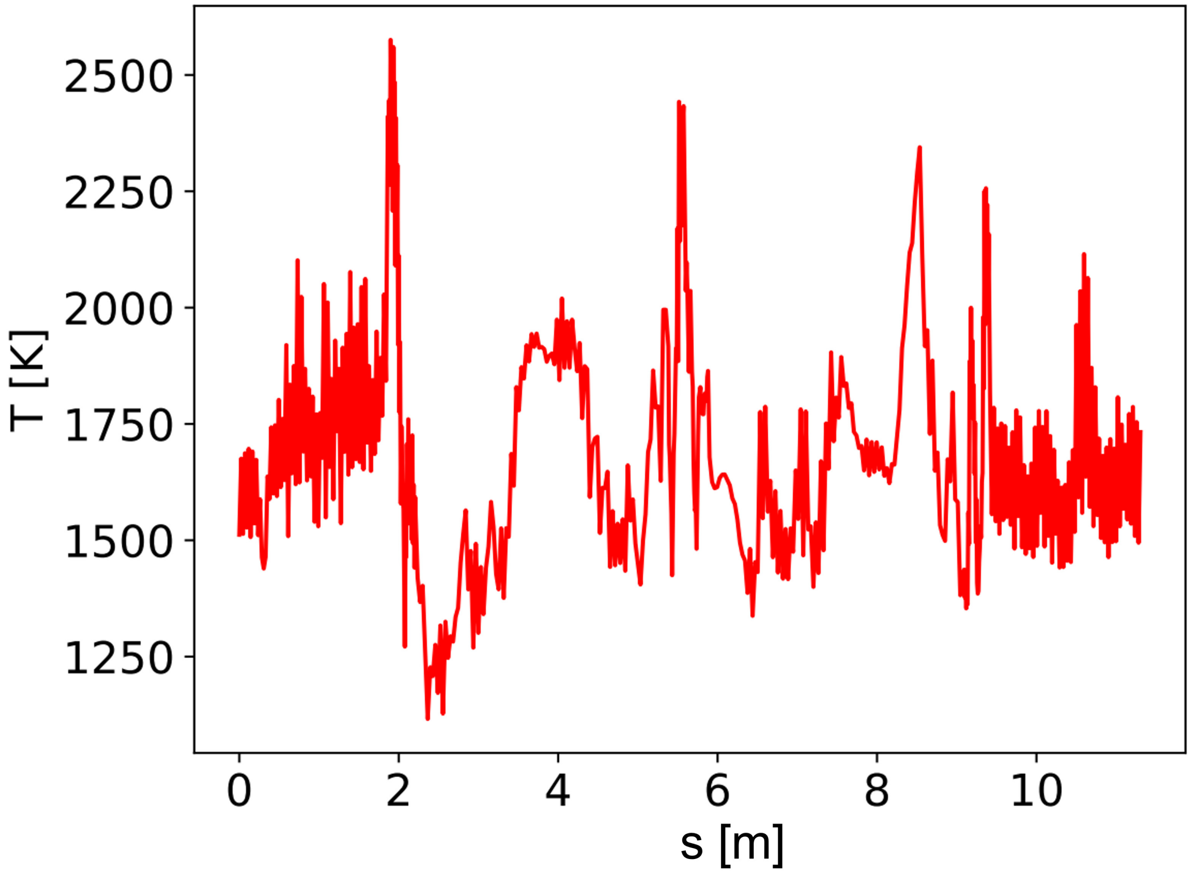


Figure 5.40. Radiative and conductive TPS: Temperature distribution along the fluid-solid interface. Samples recorded clockwise, beginning at the front of the capsule.

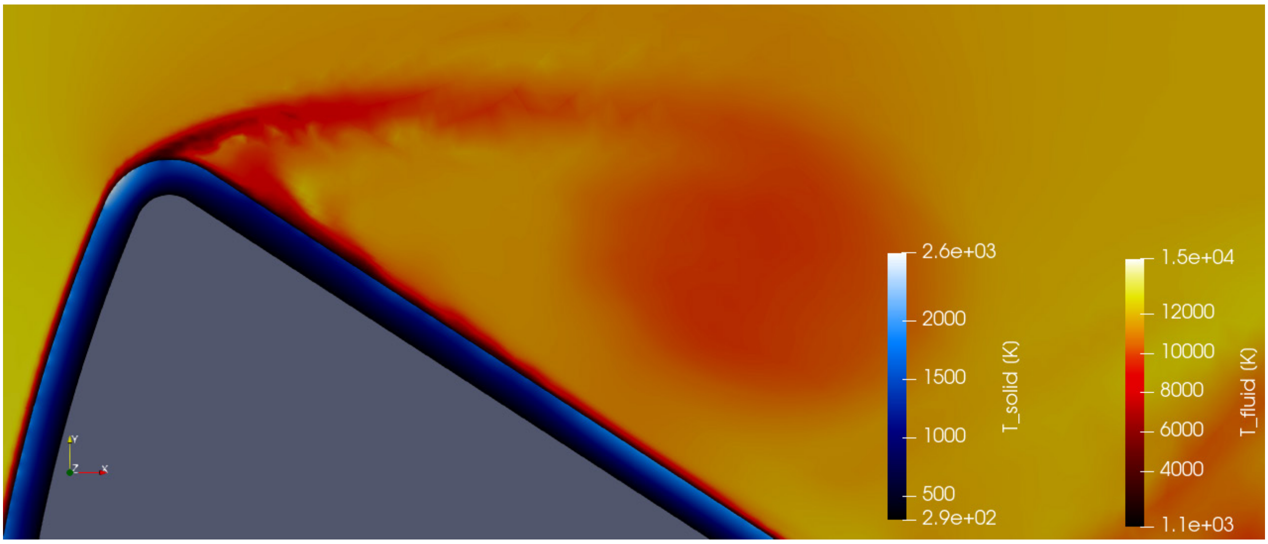


Figure 5.41. Radiative and conductive TPS: Detail of the temperature distribution in the solid domain (blue scale) and in the fluid domain (red scale) with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

Another objective in this research was the use of high-order methods to simulate the air flow surrounding the capsule. High-order methods allow to obtain accurate results with a coarser mesh and, also, the error convergence is faster if the polynomial order is increased than if the degrees of freedom of the mesh are increased. Therefore, in general, for the same threshold error the global computational time is minimum if high-order methods are used.

This approach is represented in Figure 5.42 and 5.43. The first figure shows the mesh used for this simulation and the second one shows the same mesh, but considering virtual elements that represent the accuracy of high-order methods. In this case a polynomial order of $p = 2$ has been used, which is equivalent to a 3rd order method. As it can be appreciated in these figures, the use of high-order methods allows to capture slight details and a rich fluid structure, that would be impossible to be analysed if a low-order method had been used for the same mesh.

Therefore, this objective has been completed successfully, showing the power behind this methodology, that is usually out of reach if a commercial CFD solver is used.

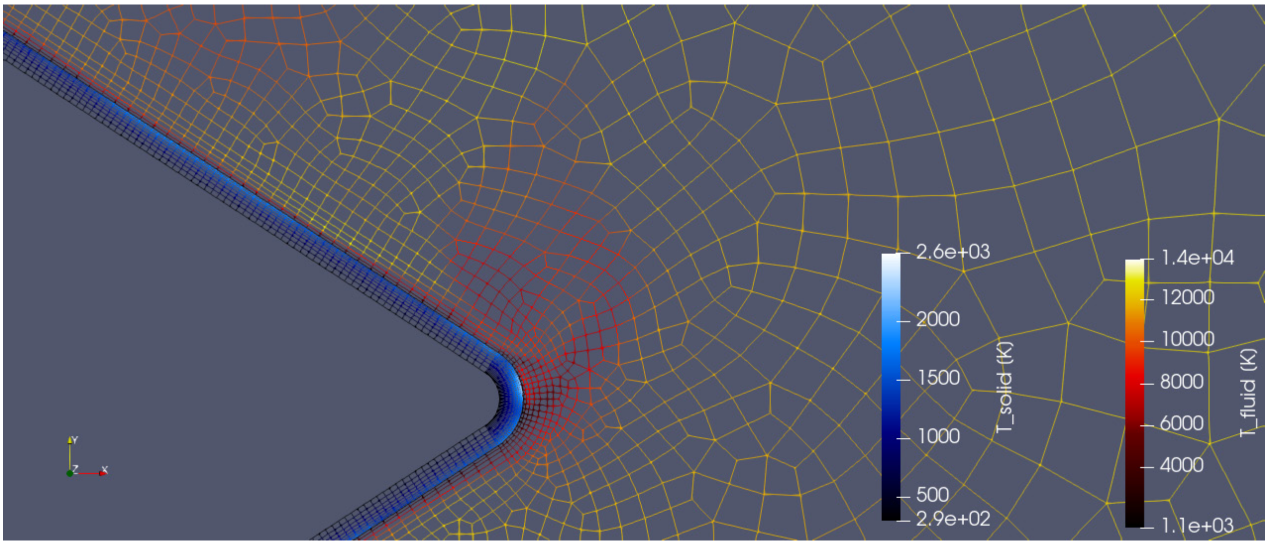


Figure 5.42. Radiative and conductive TPS: Detail of the temperature distribution over the mesh in the solid domain (blue scale) and in the fluid domain (red scale) with $p = 2$, for the inflow conditions after the shock wave and $AoA = -18°$.

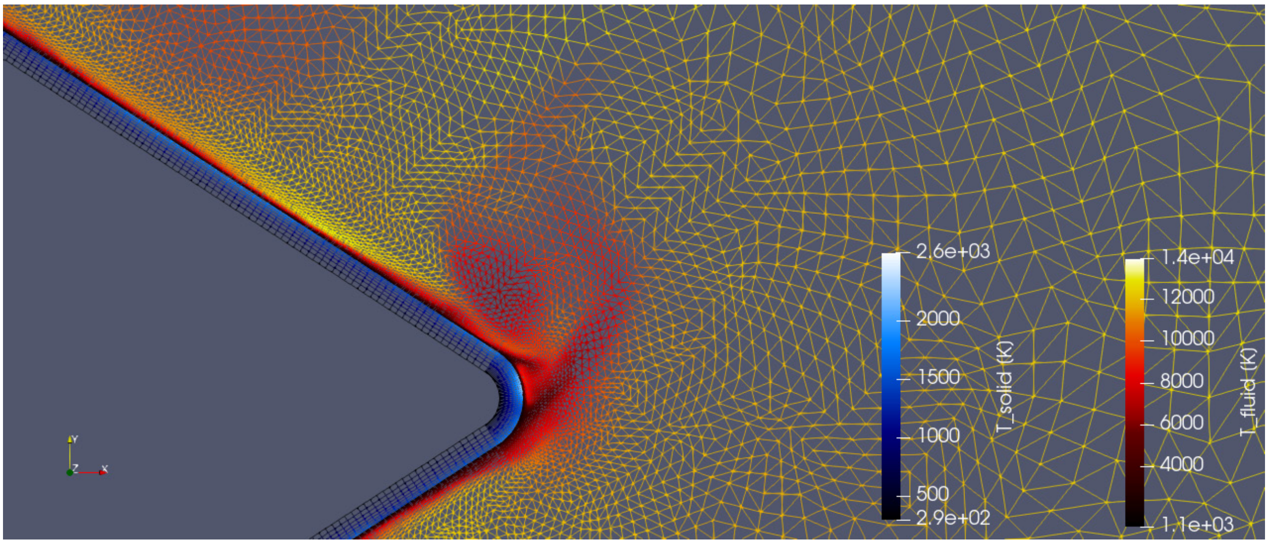


Figure 5.43. Radiative and conductive TPS: Detail of the temperature distribution over the mesh (including sub-elements for $p = 2$) in the solid domain (blue scale) and in the fluid domain (red scale), for the inflow conditions after the shock wave and $AoA = -18°$.

Finally, it is important to analyse the temperature distribution inside the TPS of the capsule in order to decide if the payload can withstand the thermal loads during the descent phase and to check whether the isothermal BC in the inside of the capsule is an appropriate model to define the thermal behaviour of the reentry vehicle.

The thermal distribution in a cross section of the capsule is schematically represented in Figure 5.44. The temperature variation within the TPS is almost linear at each point, from the external wall temperature, $T_{w,ext}$, which is equal to the temperature of the fluid at the interface, to the internal wall temperature, $T_{w,int}$, which has been considered as constant.

The isothermal BC used for this analysis will be valid if, during the whole reentry phase, the temporal variation of the temperatures $T_{w,int}$ and T_{int} is negligible. This statement will be correct if the conductive heat fluxes over the internal wall are small.

The temporal evolution of the internal temperature will be computed taking into account the thermal inertia of the payload and the mean characteristic time for the reentry phase.

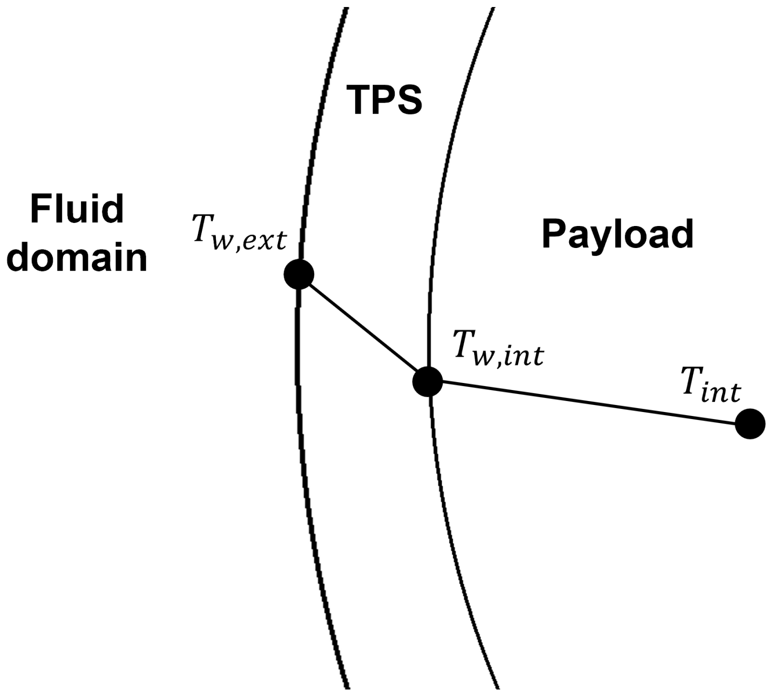


Figure 5.44. Temperature profile within the TPS of the capsule.

The heat flux at the internal interface, between the TPS and the payload area, can be estimated establishing a continuity of the conductive fluxes at the interface, through the following expression:

$$q_{int} = \frac{k_{TPS}}{t_{TPS}}(T_{w,ext} - T_{w,int}) = h_{int}(T_{w,int} - T_{int}), \tag{67}$$

where $t_{TPS} = 0.1\,\mathrm{m}$ is the thickness of the TPS, $k_{TPS} = 0.1\,\mathrm{W/(m \cdot K)}$ is the thermal conductivity of the TPS and h_{int} is the mean heat transfer coefficient of the inside.

Once the complete CHT problem has been solved, these fluxes can be computed from the previous expression. The resulting fluxes along the internal interface are represented in Figure 5.45. As it is shown in this figure, the mean heat flux along the internal interface has a value of $q_{int} = 1383.6\,\mathrm{W/m^2}$. This value should remain almost constant during the reentry if the inside of the capsule is considered as isothermal. This result also shows that the conductive heat fluxes inside the TPS are smaller than the radiative heat fluxes, whose maximum is two orders of magnitude higher; and, hence, these conductive fluxes are not essential to have a reliable model of the flow field surrounding the reentry capsule. Therefore, the resolution of the complete CHT problem is only required to obtain the temperature distribution within the TPS of the capsule.

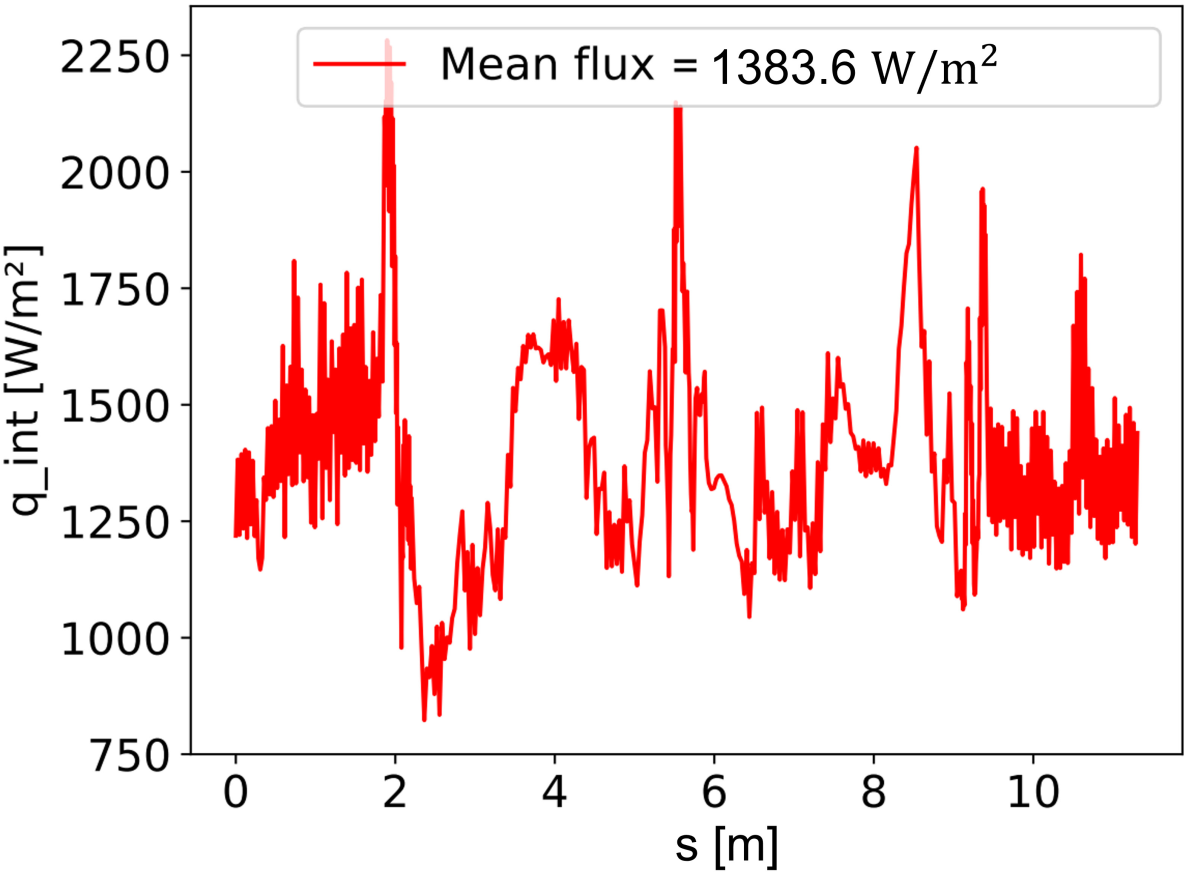


Figure 5.45. Radiative and conductive TPS: Conductive fluxes along the internal interface of the TPS. Samples recorded clockwise, beginning at the front of the capsule.

This mean conductive heat flux is absorbed by the inside of the capsule during the whole reentry phase. Therefore, if the payload has enough thermal inertia, the increase in temperature will be negligible and, hence, it can be considered as isothermal, validating the whole analysis.

The temporal evolution of the temperature is given by the energy balance (11), considering the mean flux calculated at the interface:

$$V \rho c \frac{\mathrm{d}T}{\mathrm{d}t} = S q_{int}, \tag{68}$$

where $V = 13.4\,\mathrm{m}^3$ is the volume, $S = 32\,\mathrm{m}^2$ is the internal surface, ρ is the mean density and c the thermal capacity of the payload.

The value of the density will be $\rho = 414.7\,\mathrm{kg/m^3}$, taking into account the original volume and mass of the Apollo command module. There is not data available to accurately estimate the thermal capacity; therefore, a mean value of $c = 1000\,\mathrm{J/(kg \cdot K)}$ will be used, considering the air, the crew and the electronic and mechanical devices.

From the previous equation, the average temperature increase can be estimated:

$$\Delta T = \frac{S}{V}\frac{q_{int}}{\rho c}\Delta t = 3.98\,^{\circ}\mathrm{C}, \tag{69}$$

with $\Delta t = 500\,\mathrm{s}$, as the characteristic time for the critical phase (from a thermal point of view) of the reentry.

Therefore, during the descent, the inside of the capsule will have its temperature increased only by $4\,^{\circ}\mathrm{C}$, which is completely negligible in relation to the high temperatures recorded in the TPS.

In conclusion, this result has shown that the isothermal BC for the inside of the capsule is appropriate and it can be used to model the aerothermodynamic behaviour of the reentry vehicle with a high degree of accuracy and reliability.

5.6 Shock capturing

The last analysis of this research, as a complement to all the previous results, is focused on the shock wave generated in the hypersonic flow.

Until now, the inflow conditions have been computed after the shock wave, considered as a normal shock wave. However, in this analysis the inflow conditions will be computed directly from the real data of the descent trajectory (see Figure 3.6 in sec. 3.2), as it was explained in sec. 3.2.2, where the new inflow conditions are compiled in Table 3.3.

The shock capturing functionality has been recently included in the CFD solver HORSES3D and it is not robust enough yet to solve the complete conjugate heat transfer problem and assure the convergence in a reasonable time. In fact, this analysis has been the first time this solver has been used to solve a shock wave with a hypersonic Mach number.

Therefore, to simplify this problem, the capsule will be considered as adiabatic because the main interest of this problem is the resulting shock wave. Furthermore, the shock capturing module, described in sec. 3.1.3, has been used to improve the stability of this problem.

The mesh used to solve the Navier-Stokes equations has been modified to improve the accuracy near the region where the shock wave appears, which has been located through a preliminary analysis with the original mesh. In this case, an angle of attack of 0 ° has been used to make the mesh refinement process easier.

The final mesh that has been used to capture the shock is represented in Figure 5.46, where the original boundary layer around the capsule has been preserved. However, the region where the shock is expected is formed by smaller elements than in previous analyses. Furthermore, the interface between elements has been oriented to simulate the shape of the shock wave and, hence, the full potential of Discontiuous Galerking can be used.

This simulation has been run with a $p = 2$ polynomial order. However, in this case it is not possible to assure that the error follows a 3rd order convergence rate, because of the discontinuity in the flow field, which might reduce the effective order of the method locally.

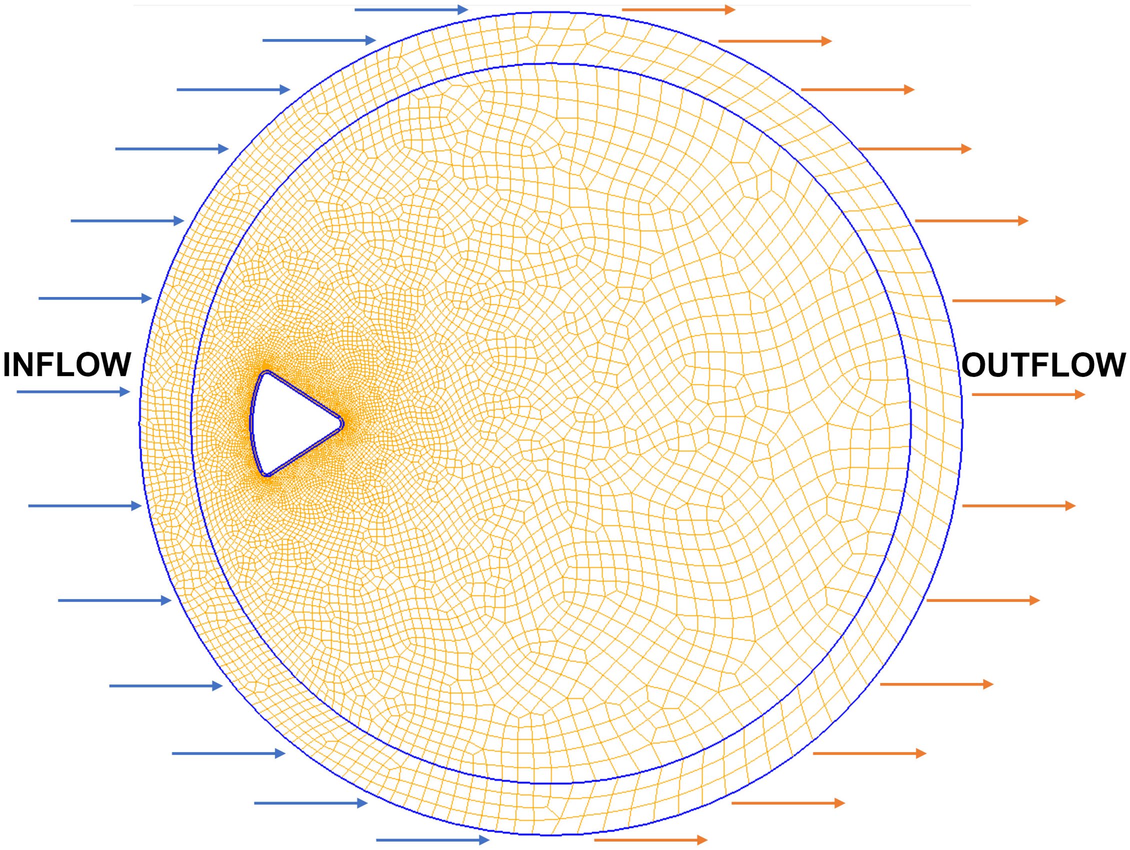


Figure 5.46. Mesh for the fluid domain around the reentry capsule, used for the shock capturing analysis.

The resulting Mach number is represented in Figures 5.47 and 5.48, where the effect of the shock wave is clear, effectively reducing the Mach number in a small region. The blunt shape of the capsule generates a *detached shock wave*, which is similar to a normal shock wave near the front of the body. Furthermore, the proposed methodology has allowed to accurately capture the shock, showing the real behaviour of the flow field surrounding the reentry vehicle.

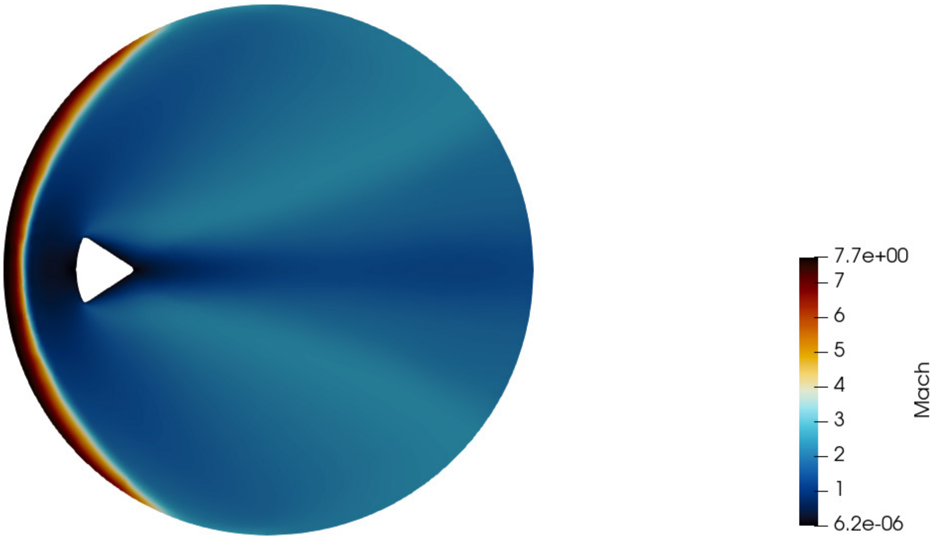


Figure 5.47. Shock capturing: Snapshot of the Mach number distribution in the fluid domain with $p = 2$, for the real inflow conditions before the shock wave and $AoA = 0°$.

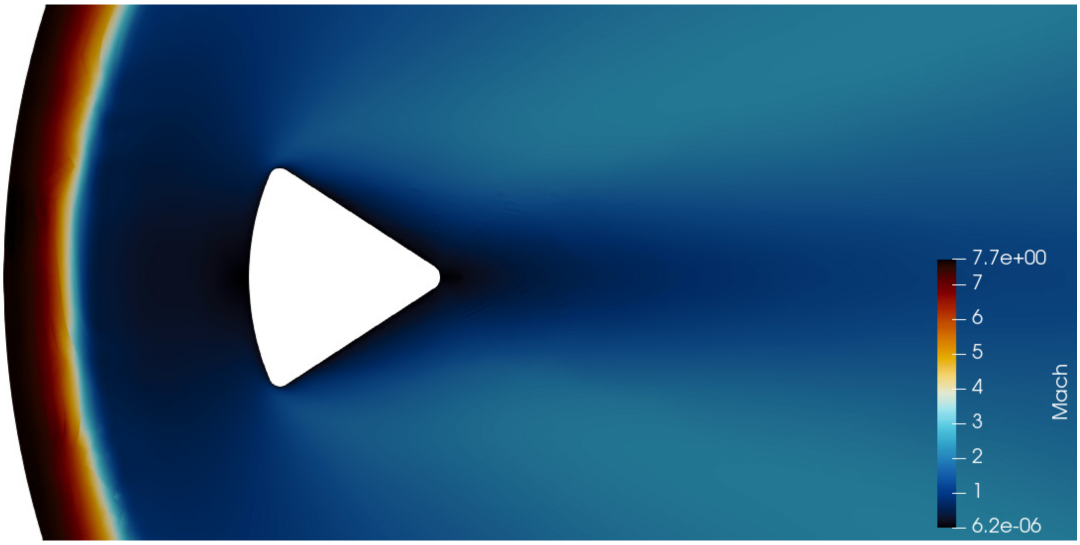


Figure 5.48. Shock capturing: Detailed view of the Mach number distribution near the shock wave with $p = 2$, for the real inflow conditions before the shock wave and $AoA = 0°$.

The temperature distribution is shown in Figures 5.49 and 5.50. In this case, the temperature gradient is located in a very thin layer, where the temperature is increased more than one order of magnitude. It is interesting to notice that the after-shock region, in a wide range around the capsule, has a very similar temperature, which may justify the use of the conditions after the shock for all the previous analyses.

The maximum temperature has a value 9086 K, on the frontal surface of the capsule, as it was expected from the previous results. However, there is also a considerable hot region at the rear of the capsule, which is a low velocity region and most of the energy is thermal energy, instead of kinetic energy.

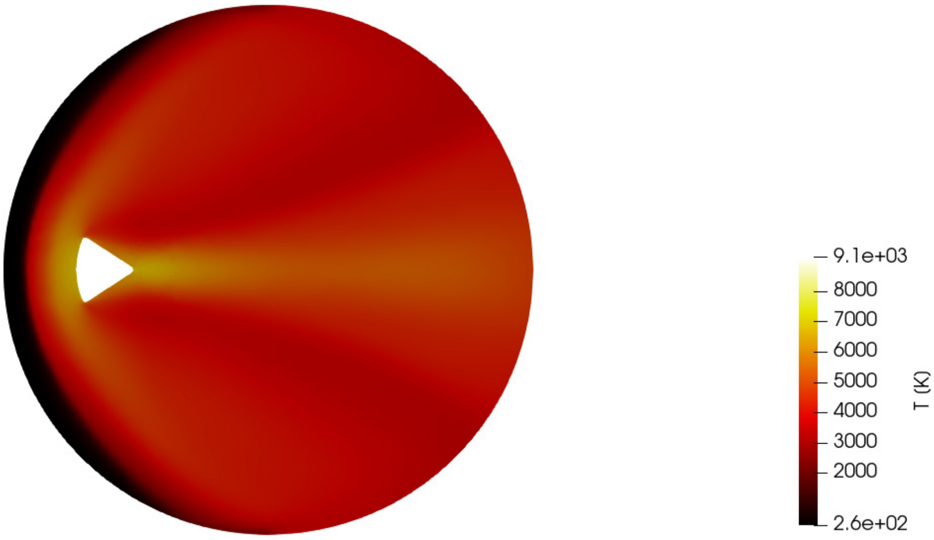


Figure 5.49. Shock capturing: Snapshot of the temperature distribution in the fluid domain with $p = 2$, for the real inflow conditions before the shock wave and $AoA = 0°$.

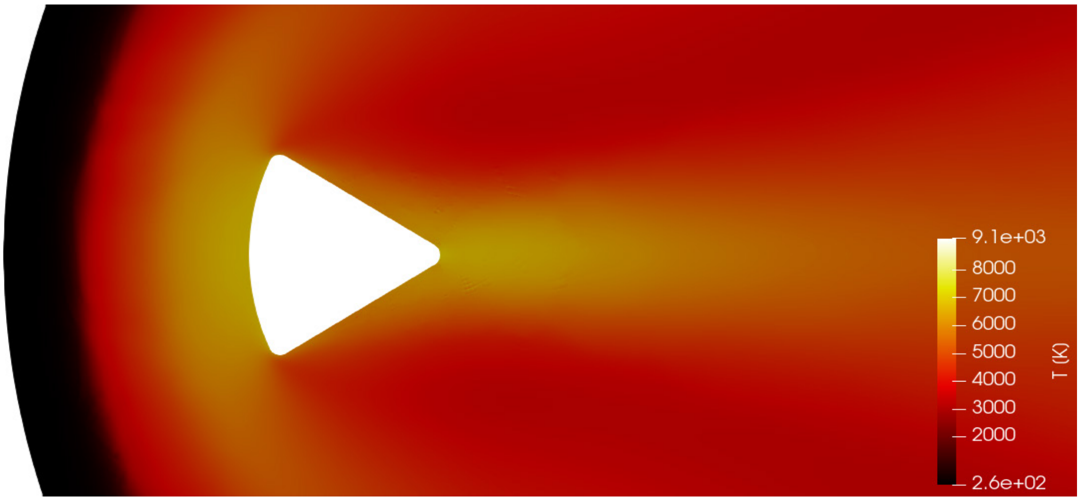


Figure 5.50. Shock capturing: Detailed view of the temperature distribution near the shock wave with $p = 2$, for the real inflow conditions before the shock wave and $AoA = 0°$.

The pressure has experienced an important change through the discontinuity, whose value has been increased almost two orders of magnitude, as it is shown in Figures 5.51 and 5.52. The proposed methodology has allowed to constrain the shock to a very thin region, preserving the nature of the discontinuity. This accuracy has been achieved through the use of the DG discretisation and the correct orientation of the elements of the mesh near the shock, as it is represented in Figure 5.53. As it can be observed, most of the boundaries beween elements are located exactly over the discontinuity, taking advantage of the features offered by the Discontinuous Galerkin solver.

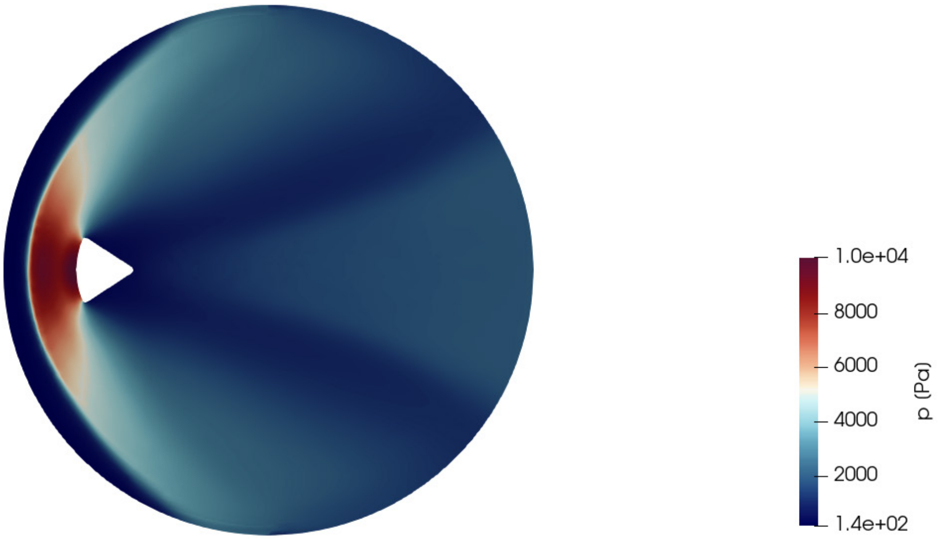


Figure 5.51. Shock capturing: Snapshot of the pressure distribution in the fluid domain with $p = 2$, for the real inflow conditions before the shock wave and $AoA = 0°$.

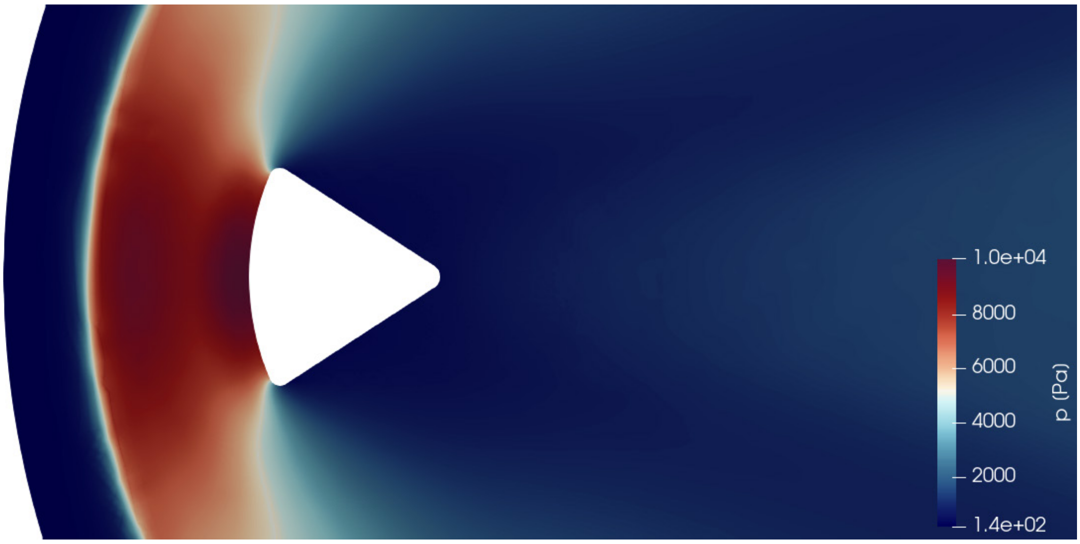


Figure 5.52. Shock capturing: Detailed view of the pressure distribution near the shock wave with $p = 2$, for the real inflow conditions before the shock wave and $AoA = 0°$.

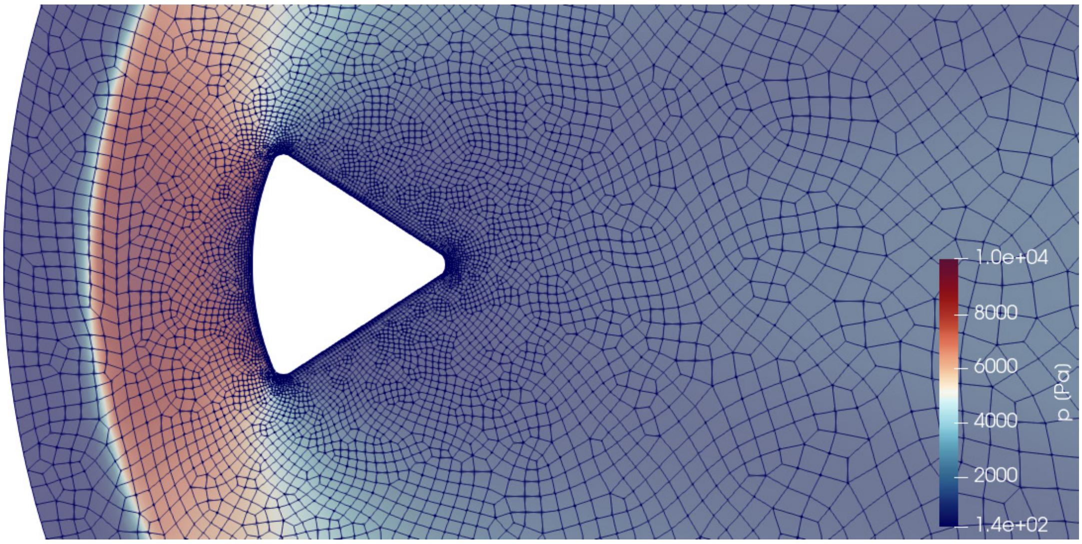


Figure 5.53. Shock capturing: Detailed view of the pressure distribution and the mesh near the shock wave with $p = 2$, for the real inflow conditions before the shock wave and $AoA = 0°$.

As the temperature is more uniform after the shock, the density, which is represented in Figure 5.54, shows a behaviour very similar to that of the pressure. An interesting feature of this result is that the minimum density is located at the rear of the capsule, instead of upstream the shock wave. This effect is generated by the low pressure region at the rear, together with the high temperature present in that zone, as the three variables (density, pressure and temperature) are linked through the equation of state (27).

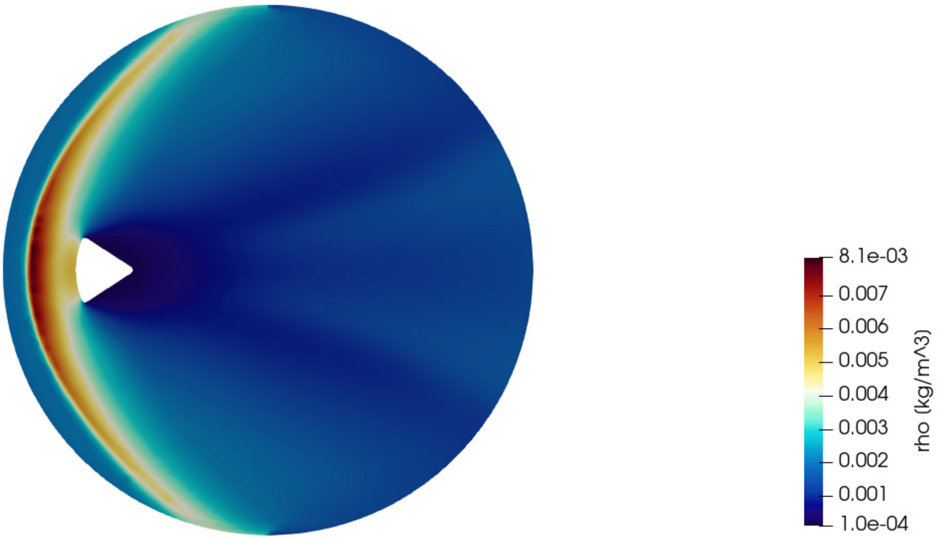


Figure 5.54. Shock capturing: Snapshot of the density distribution in the fluid domain with $p = 2$, for the real inflow conditions before the shock wave and $AoA = 0°$.

Finally, the velocity field is represented in Figures 5.55 and 5.56, where a perfect symmetry can be observed due to the angle of attack is 0 ° in this analysis. When the fluid surrounds the capsule, the velocity is increased near the curvilinear edges at both sides, which is consistent with the decrease of the pressure near those regions.

The vertical velocity downstream, following the y axis, is reduced as the air flow is farther from the capsule. On the other hand, the maximum horizontal velocity, following the x axis, is located downstream, where the reduction in temperature has increased the kinetic energy, surpassing the original velocity upstream the shock wave.

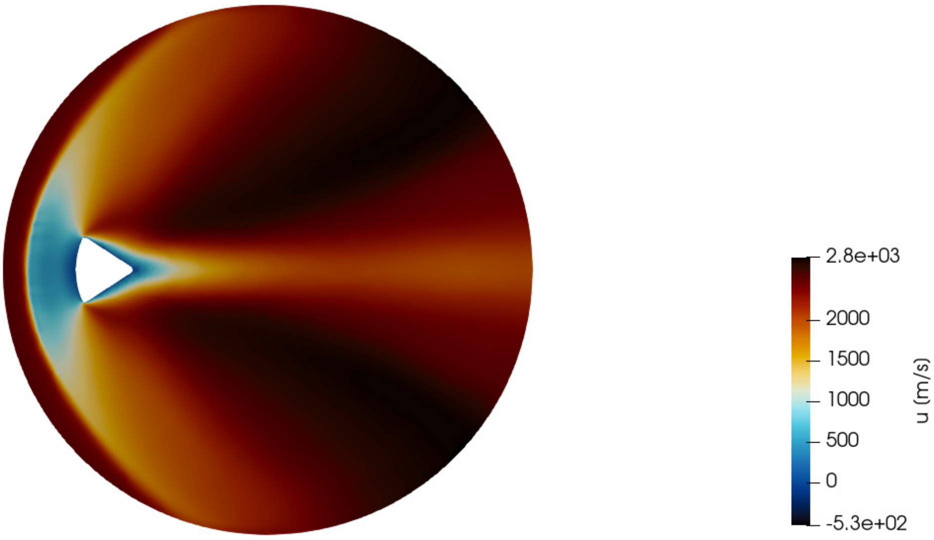


Figure 5.55. Shock capturing: Snapshot of the velocity field, following the x axis, in the fluid domain with $p = 2$, for the real inflow conditions before the shock wave and $AoA = 0°$.

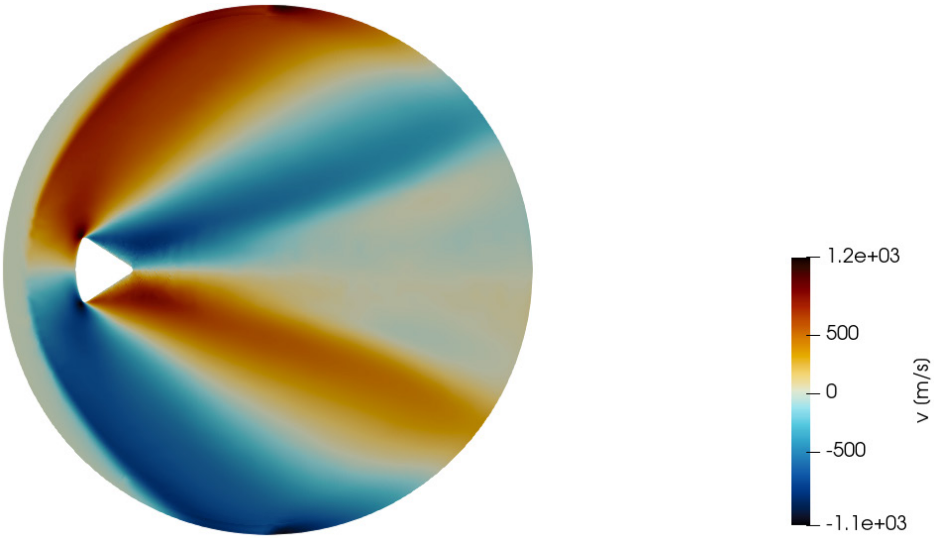


Figure 5.56. Shock capturing: Snapshot of the velocity field, following the y axis, in the fluid domain with $p = 2$, for the real inflow conditions before the shock wave and $AoA = 0°$.

This last analysis has allowed to understand the effect of a real shock wave on the hypersonic flow surrounding a reentry capsule. The possibility of capturing shock waves with hypersonic Mach numbers will improve the reliability of future analyses, while a better understanding of the reentry conditions can be acquired.

These results are similar to other works, present in the scientific literature, regarding shock capturing through a Discontinuos Galerkin discretisation, as it is the case of [50], where a velocity distribution is represented for a flow field at $M = 5$ and $Re = 10^6$. This result shows a high fidelity compared to the Mach distribution obtained in this research, even though the reentry capsule and the inflow conditions are different in both cases. Nevertheless, the general behaviour of the flow field follows a similar trend in both analyses.

However, it is important to mention that the solution within the trail of the capsule must be considered with caution. The artificial viscosity used as a stabilisation mechanism for the shock capturing analysis might have made this region smoother than the real situation. Furthermore, for this analysis the turbulence model has not been considered, because of a lack of time to run the whole 3D simulation, which requires a tough and long process to achieve a stable solution (see sec. 9).

Although there are some features that can be improved to model shock capturing, this functionality has been satisfactorily tested with the solver HORSES3D, and the objective regarding the model of shock waves in hypersonic conditions has been successfully completed.

6 Conclusions

In this work, several hypersonic flows have been simulated to model the aerodynamic and thermal behaviour of a reentry capsule during the descent phase. In particular, open-source solvers have been used with this aim, which have been previously validated through the resolution of some test cases. The different simulations that have been performed have allowed to successfully fulfil the objectives that had been established for this research.

First, the real inflow conditions have been estimated based on the trajectory of the Apollo Command Module during its reentry in the AS-202 mission. The ISA model has allowed to compute the atmospheric conditions, known the altitude for a particular instant of the reentry. Furthermore, two different inflow conditions have been obtained: the real conditions, extracted from the trajectory, to capture the detached shock wave at hypersonic Mach numbers, and the after-shock conditions, considering a normal shock wave, to simplify the Conjugate Heat Transfer analysis between fluid and solid.

The aerodynamic problem has been simulated through the CFD solver HORSES3D, which provides a wide set of available functionalities to model the flow field surrounding the capsule. A turbulence model has been used to perform a Large Eddy Simulation of the 3D flow field, which has allowed to include the influence of small scales on the aerodynamic problem. Furthermore, high-order methods have been present in every analysis, resulting in accurate results with a coarser mesh, which has significantly reduced the global simulation time compared to the time that is required to achieve the same error through low-order solvers.

The heat equation within the solid domain has been solved through the FEM solver MFEM. This solver has been linked with HORSES3D through a programming interface, developed in Python, which has allowed to solve simultaneously the aerodynamic and the thermal problem in a common framework. Furthermore, the correct functioning of this interface has proved the possibility of exchanging information between applications written in different programming languages. This fact has shown the flexibility of open-source software for the resolution of complex aerothermodynamic problems.

The meshing process has been performed through the application Gmsh, which is compatible with both solvers. Gmsh allows to create 2D and 3D meshes over arbitrary geometries and, furthermore, it is possible to define curvilinear elements up to a selected order, which is essential to perform high-order simulations.

Several analyses have been performed to simulate the hypersonic flow surrounding the reentry capsule, in increasing degree of complexity. First, a reentry capsule, modelled as an adiabatic BC, has been simulated, which has allowed to understand the general behaviour of the turbulent flow field. However, in this case a maximum temperature of 13495 K is reached at the front of the capsule. This temperature is huge and a Thermal Protection System is required to protect the inside of the vehicle against the thermal loads induced by the fluid.

The next analysis was performed considering an isothermal capsule at 2000 K. In this case, a thermal boundary layer appears and some eddies transport the cool fluid downstream, reducing the temperature of the flow. However, the Mach number distribution does not show a significant variation in relation to the previous case. This analysis has been very useful, as it will be used as a restarting point for the analysis with a radiative TPS, improving the stability of the thermal problem.

Once the aerodynamic problem has been solved, the effect of the TPS was included through 3 different CHT analyses, considering conduction, radiation and both to solve the thermal problem. The use of a conductive TPS only allows to insulate the inside of the reentry vehicle. However, the maximum temperature over the capsule has a value of 13089 K, which can not be withstood by any known material. Therefore, the effect of thermal radiation has to be considered to cool the capsule down.

The use of a radiative TPS has allowed to reduce the temperature over the capsule, with a maximum value of 2610 K, which is a more reasonable value, specially taking into account that a quasi steady-state analysis has been performed. The radiative effect of the Sun and the Earth is negligible, as its value is several orders of magnitude below the heat flux radiated by the capsule. The addition of conduction within the TPS does not modify the aerodynamic or thermal behaviour of the problem. However, it has proven that the inside of the capsule is expected to has its temperature increased only by 4 °C during the reentry and, hence, it can be considered as isothermal without an important lack of precision.

Finally, the shock wave generated by the reentry capsule has been simulated at a Mach number of $M = 7.66$ and the real atmospheric conditions. In this case, the shock capturing functionality provided by HORSES3D has been used to improve the convergence in presence of discontinuities. The resulting shock wave has been accurately modelled through this approach, but the solution within the trail might not be reliable enough due to the effect of the artificial viscosity that has been used to stabilise the problem.

7 Future research

The analysis of a reentry capsule immersed in a hypersonic flow has been simulated from different points of view, showing several approaches to address the aerothermodynamic problem. However, some assumptions have been made to simplify the original problem, focusing only in those aspects that are more relevant for the present analysis. Therefore, there are several options to improve the present research in a future work.

The ideal gas assumption was already implemented within the CFD solver, which has allowed to solve the equations without modifying the source code. However, this assumption might not be valid for very high temperatures, when the flow behaves as chemically reactive. Therefore, the definition of a new set of equations in HORSES3D would be desirable in a future work.

Furthermore, the shock wave and the CHT problem have been computed separately, considering the after-shock conditions in the second case. This approach is a simplification of the real problem and it is possible that different results could be obtained if both problems were solved simultaneously. Therefore, a future research could be oriented towards the improvement of stability, in order to solve the complete problem in a single simulation. Also, it would be desirable to run the simulation with different meshes to estimate the accuracy of the results and the error convergence rate in presence of a shock wave.

The simulation of turbulence requires the definition of a three-dimensional problem. In this case, a 2D mesh was extruded and periodic boundary conditions were applied to simulate the 3D behaviour. However, the real behaviour of the flow field for a real 3D capsule could differ from this analysis, specially for the shock capturing model, as the shape of the shock and the distance from the capsule to the shock might be different. The 3D simulation of the hypersonic flow field, including the shock wave and the CHT problem, coupling the fluid with a 3D capsule, is a very challenging problem, but it would provide a better fidelity in relation to the real situation.

Finally, an accurate thermal model of the inside of the capsule would be desirable to be taken into account in a future research. Although the isothermal assumption for the inside has proven to be reasonable, a complete FEM model of the different devices, materials and equipment would provide a better understanding of the thermal behaviour inside the reentry vehicle during the descent.

8 Annex 1: Programming interfaces

The resolution of *Conjugate Heat Transfer* problems requires to solve simultaneously the fluid and solid equations in order to compute the thermal coupling between both domains. Unless a monolithic approach is used (see sec. 4), two different solvers are needed. These solvers may use different discretisation approaches and methodologies or even different programming languages. Therefore, a *programming interface* is required to perform the coupling in a common framework.

In this case, the FEM solver for the solid domain, MFEM [17], is an open-source solver that has been developed in C++. However, the CFD solver for the fluid domain, HORSES3D [11], which is also an open-source programme, developed at ETSIAE-UPM, has been written in Fortran 2003. Hence, an interface must be developed to link these two different worlds with a single objective.

MFEM already provides a Python wrapper, created by its developers, which allows to call from Python every functionality of this solver. Therefore, Python will be the selected language to develop the interface. If HORSES3D could be linked with Python, then both solvers could be coupled through the same interface.

There are two different approaches to link Fortran and Python:

1. **Python as main**: The Fortran library is compiled and transformed to a Python module. The Python interface has a complete control over the Fortran module. This method allows to keep the original performance of the Fortran programme.

2. **Fortran as main**: The Fortran programme has a complete control and the Python library is called only when needed. Variables are shared through a global dictionary instead of through the function's arguments. The performance is reduced proportionally to the number of calls to the Python library.

Each one of these approaches has its advantages and disadvantages and the use of one or other approach strongly depends on the particular situation and the flexibility of both solvers.

Python as main

This is usually the desired approach, because the Python programme acts as main and it facilitates the post-processing tasks. Furthermore, it would allow to have control over both solvers from the same main programme, which simplifies the coupling process.

The key behind this methodology is to compile the Fortran programme, while creating intermediate interfaces which allow to call each function and subroutine of the Fortran module as if it was a Python function. In order to transform the compiled Fortran module and to create the intermediate interfaces, the library F2PY [60] is used.

Although this library allows to transform Fortran programmes to Python functions, the Fortran code has to be written in a very specific way in order to make it work properly and, hence, it is not useful in practice. Therefore, we developed a *Fortran parser* in a previous work in order to overcome this problem. The parser, called *fparser* [61], is a Python-based open-source library with a simple and intuitive GUI which automatically rewrites the content of any Fortran module in order to make it compatible with F2PY. This way, the previous problem is not a constraint any more and the Fortran module can be called from Python easily. However, the resulting transformed Fortran module, after F2PY has been used, is difficult to be used and some experience is required in this field to extract the full potential of this approach. Therefore, the fparser's GUI also creates a simpler Python interface which links the main Python programme with Fortran, so the user only has to interact with usual Python functions. This methodology is shown schematically in Figure 8.1.

The main advantage of this approach is the *performance* of the transformed Fortran module. Several tests have demonstrated that the computational efficiency of the Fortran programme is not reduced when transformed to Python functions. This fact is specially interesting for those applications which require a large computational time (e.g. CFD).

However, the conjunction between F2PY and fparser has some limitations that may be an impediment for some applications:

1. The Fortran programme must be formed by hierarchical modules, which may contain functions and subroutines.
2. Object-oriented programming can be used only in low-level modules, which can not be accessible from Python.

3. Fortran procedures can be used to use a Python function as an input to a Fortran function. However, concatenated procedures are not implemented in fparser.
4. When compiling the Fortran programme, most of the flags can not be selected by the user due to they have to be automatically tuned to assure a correct compatibility with F2PY.

These limitations, which are not a critical problem in many cases, entail a significant constraint for the present research because of the particular characteristics of the CFD solver, HORSES3D:

1. It is highly object-oriented at all levels.
2. It has been designed to be used as an executable programme and, hence, the main file is not a Fortran module.
3. There are several *Makefiles* linked together to dynamically compile different functionalities of the solver. It can not be compiled in a single step.

These features are incompatible with the previously described approach. Therefore, this methodology will not be used as a coupling tool between both solvers.

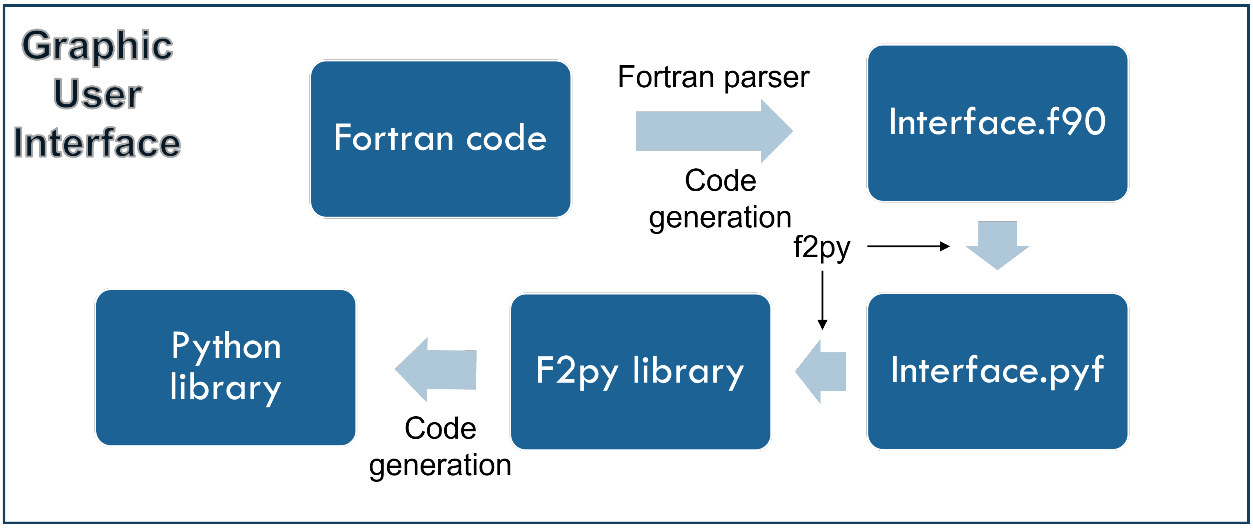


Figure 8.1. Schematics of the proposed methodology: F2PY-fparser.

Fortran as main

As an alternative to the previous approach, there is another way of linking a Fortran and a Python programme. In this case, the Fortran programme acts as main and, therefore, it will call the Python library when needed. The only way to do this is using *compiled Python code*. Although Python is not designed to be compiled, there are some libraries which allow to use this functionality. In this case CFFI [62] will be used to compile Python in a C-style way.

Once every Python function has been compiled, it is necessary to create some Fortran interfaces, using the *iso_ c_ binding* functionality (included within Fortran 2003 compilers), to communicate with the C-Python code. These interfaces will be added to the Makefile and they will be linked to the main Fortran code during the compilation.

The main problem of this approach is that it is necessary to create a new Fortran interface with *iso_ c_ binding* each time a new Python function is defined. This would make arduous the task of creating new Python functionalities and the setup of the coupling process would require a large amount of time. Therefore, it is important to avoid the creation of too many interfaces, but keeping the same functionality.

The equilibrium between effort and number of interfaces has been achieved by Noah Brenowitz [19], who provides a Github repository with only the essential Fortran interfaces. These interfaces allow to call three predefined Python functions:

1. **Set state**: Updates the value of a shared variable between Fortran and Python.
2. **Get state**: Recovers the value of a shared variable between Fortran and Python.
3. **Call function**: Calls an arbitrary Python function from Fortran. It only works if the Python function is installed as a library within a Python environment.

The key behind this idea is a global Python dictionary called *state*, which has been defined at the interfaces to be accessible from Fortran and Python simultaneously. The command *set_ state* updates a variable of this dictionary and *get_ state* recovers the value of a variable. These variables are always arrays up to three-order tensors.

The command *call_function* allows Fortran to gain access to installed Python libraries. However, no input arguments can be sent through this command. Therefore, every Python function which needs to be called from Fortran must have a single input argument: the global dictionary *state*. This way, the input variables can be updated from Fortran and then, when the Python function is called, it has access to all the predefined shared variables.

This three functions available in Fortran are enough to share any variable and to call any Python function. The scheme of this procedure is represented graphically in Figure 8.2.

Although this approach is restrictive from the point of view of the Python functions, it gives a complete freedom to the Fortran programme and no requirements are established over this language. In the case of the solver HORSES3D, it is essential to have a lot of flexibility in the Fortran side, while the Python side can be defined as desired. Therefore, in this case, this is the best possible approach and it is the one that has been used in the whole research.

However, it is important to mention that the use of this approach deteriorates the computational performance of the programme. Therefore, the Python library has to be designed to be called a few times only, because each additional call is expensive due to the user defined Python functions are not compiled. However, the flexibility provided by this methodology makes it a good trade-off, even though the required time to run the simulations will increase.

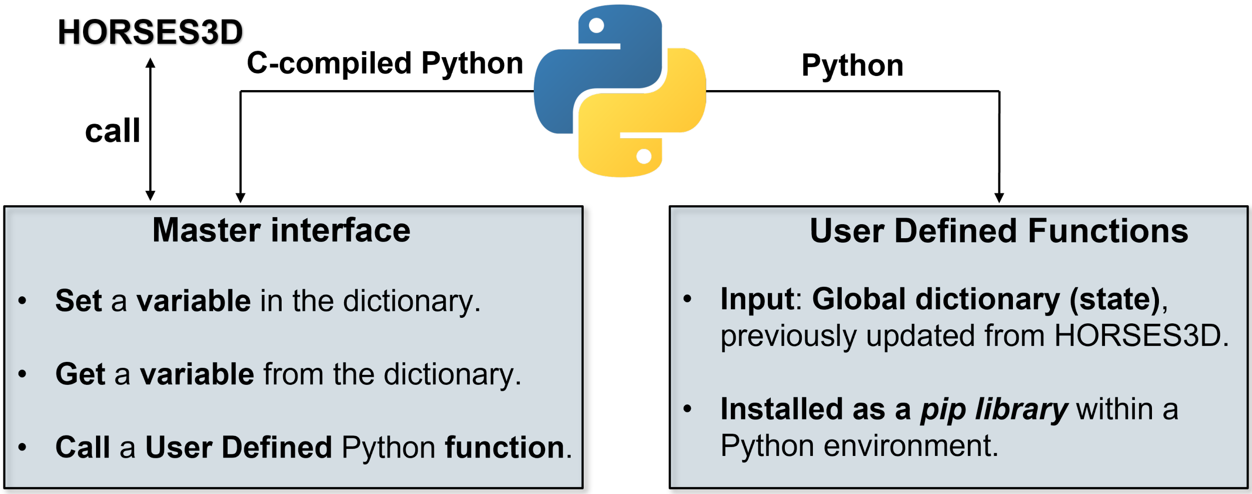


Figure 8.2. Schematics of the proposed methodology: Python interface with HORSES3D as main.

9 Annex 2: Issues and challenging problems

The use of non-commercial software to solve aerodynamic (e.g. HORSES3D) and thermal (e.g. MFEM) problems is a challenging task. Although these applications offer a greater flexibility and provide several functionalities which allow to obtain accurate solutions, through the use of high-order methods for the different analyses considered in this research, the use of these tools and the creation of programming interfaces is far from being a trivial issue.

On the other hand, commercial CFD software (e.g. FLUENT) is prepared to work almost always, due to its incredible robustness. However, this advantage is a double-edged sword:

1. It is very difficult to implement generic high-order methods which can properly work for every possible scenario. Hence, commercial applications usually use low-order methods, which reduces the accuracy of the solution.
2. If the programme is very robust, it is possible to obtain solutions for ill-posed problems or simulations without any physical meaning due to the input parameters may be incorrectly configured.

Therefore, commercial software must be used with caution, specially for inexperienced users. This work has been developed to show some interesting functionalities that are possible through the use of non-commercial tools, which may provide excellent solutions once it is clear how to use them correctly.

In order to explain the main difficulties that have appeared during these months, a brief summary of the most challenging issues is provided below, focusing on the *boundary conditions*, the *coupling* between Fortran and Python, the *time step and CFL number*, the *thermal radiation* and the *shock capturing*. All these problems have been solved successfully and it is expected that the following information can be useful for future users of HORSES3D and MFEM.

Boundary conditions

In HORSES3D, there are three possible thermal boundary conditions: adiabatic, isothermal and user-defined. Adiabatic and isothermal are the most common and they can be used with some simple flags. However, user-defined boundary conditions are hard to be implemented correctly.

These BC are separated between Dirichlet and Neumann BC. These two possible conditions are called from two different subroutines: *UserDefinedState1* and *UserDefinedNeumann1*, respectively. In each one of these conditions, the nodes at the boundary are called iteratively.

There are two main problems in this approach:

1. Each node at the boundaries (including the physical nodes of the mesh plus additional nodes created by the solver depending on the selected polynomial order) is called randomly, due to the parallelisation and multiprocessing infrastructure. Although this approach is not a problem for simple BC (e.g. $T_{BC} = f(x, y, z)$), it is a real problem when information exchange is required between different solvers and, also, when the implementation of a spatial filter is needed over some variables.

   Furthermore, updating the information through the Python interface for each node is computationally expensive. Therefore, the only possible solution is to save the coordinates and temperature values for each node at the boundary in a *static array*[2]. Then, the whole array can be sent to Python, the values can be sorted (based on the coordinates) and the filters can be applied before the solid equations are solved.

2. Dirichlet and Neumann BC are defined in different subroutines. Hence, before MFEM is called to solve the solid, the temperature distribution has to be updated from one subroutine; but then, the heat fluxes have to be recovered from another subroutine. This process requires a synchronisation between both subroutines.

   Due to the input arguments of each subroutine should not be modified, this synchronisation has to be performed based on the current iteration of the CFD solver. The heat fluxes have to be recovered one iteration after the temperature array has been updated and the solid heat equation has been solved.

In MFEM, Dirichlet BC are easily implemented. However, the problem is, again, the coupling process. Once the temperature distribution has been solved within the solid domain, it is necessary to interpolate this distribution to compute the gradient along the contour, due to that is the input for the fluid domain as a Neumann BC.

The first problem is that it is hard to interpolate a scalar field over an arbitrary curvilinear geometry. In this case, the interpolation is only performed correctly if the nodes at the

---

[2] The term *static* is used in C programming language to define local variables which keep their last value every time a subroutine is called. In Fortran, the term is *save* variables.

boundary of the fluid domain are the same that the nodes at the boundary of the solid domain. This is always true only if the polynomial order is one. If not, each solver computes additional nodes between the physical nodes of the mesh and the resulting nodes can be completely different in both solvers. Therefore, the solution is to use a $p = 2$ polynomial order in the fluid, which is programmed to create the additional nodes in the middle of the real nodes. Then, the solid's mesh can be redefined to include all these intermediate points and assure a correct interpolation. It is also important to use *Gauss-Lobatto* nodes to assure that every physical node of the mesh has been taken into account.

The second problem in MFEM is that, even though the solution can be computed for a polynomial order up to $p = 9$, the solution array only contains information regarding the physical nodes of the mesh. Therefore, most of the high-order resolution is lost after the interpolation and, hence, it is convenient to use a $p = 1$ order in this case.

Fortran-Python coupling

Another issue that it is worth mentioning is related to the exchange of multi-dimensional arrays between Fortran and Python. The selected approach to create the interface has been to use the Fortran programme as main (see sec. 8). In this case, the Python functions are compiled in a C-style way to perform the link with Fortran. The problem is that arrays in C follow a *row-major order*, while in Fortran they follow a *column-major order*. This means that C arrays are stored in rows and Fortran arrays are stored in columns.

Therefore, when an array is shared through the interface, the memory positions associated to a Fortran array represent the transposed array in C and Python. As no error is raised in this situation, it is essential to check carefully that the transposition has been performed in every array before they are shared between different programming languages.

On the other hand, if the selected approach had been to use the Python programme as main, this issue would not be a problem, as it is internally handled by the library F2PY. However, it is recommended to use the function *numpy.asfortranarray*, when defining an array in Python, to improve the performance of the coupling process.

Different programming languages may define arrays in different ways and, hence, it is important to check this issue the first time the coupling is performed to avoid errors in future analyses.

Time step and CFL

In HORSES3D it is possible to define the time step explicitly or through the CFL number. In general, the use of the CFL number is desirable as the time step can be automatically adapted as a function of the mesh.

However, to perform the coupling between two solvers it is very important to define the time step explicitly. One reason for this is that it is useful to know the time between one coupling operation and the next one. Known the time step, the number of iterations between couplings can be modified as a function of the problem characteristic time, in order to improve the stability.

The second reason is related to the synchronisation of the subroutines where the BC are defined. The time is the only input variable of these subroutines that can be used as a measure for the number of iterations, which is required to assure a correct synchronisation. Therefore, the time step must be constant and known to use this methodology.

Thermal radiation

The thermal protection system known as *insulated structure* (see sec. 1.5) has a double function: it radiates most of the heat to space, cooling the interior of the reentry vehicle, and it insulates the capsule, due to its low thermal conductivity, which reduces the conductive heat fluxes.

Without taking into account the effect of thermal radiation, the CHT problem can be solved easily through the coupling of fluid and solid at the interface between both domains. In this case, the conductive heat fluxes at the interface are the same for the fluid and the solid.

However, when the effect of radiation is considered, the conductive heat fluxes at the interface are different for the solid and fluid domain. Following the selected CHT approach (see sec. 4.1), the fluid heat fluxes can be computed as follows:

$$q_{fluid} = q_{solid} - q_{rad}, \tag{70}$$

where the conductive heat fluxes q_{fluid} and q_{solid} are already projected following the normal vector at each node of the interface. The radiative heat flux, q_{rad}, is lost because the energy is radiated to space and the atmosphere has been considered to be transparent to thermal radiation.

When the radiative flux is considered, the temperature around the capsule is around 13000 K, due to the effect of the shock wave. Therefore, the radiative flux reaches a value of $q_{rad} = 1.5\,\mathrm{GW/m^2}$, which is a huge value. This flux generates a numerical divergence of the CFD solver, due to its incredibly high value, and the time step required to stabilise the solution would be too small to be computed in a reasonable time.

The methodology to solve this issue is based on the simulation of several radiative cases, modifying in each case different parameters such as the mesh refinement, the time step, the boundary conditions, the spatial/temporal filter used to stabilise the CHT problem or the restarting point of the simulation, and checking whether these variations improve the performance and the numerical stability.

Finally, the solution for this problem is to solve only the fluid, considering an isothermal capsule at a smaller temperature, $T = 2000\,\mathrm{K}$. Then, a new simulation is performed, restarting from the previous solution. This way, the initial radiative flux is smaller and it is easier to converge the problem. However, the first iterations are critical and any fluctuation may cause a divergence of the simulation.

Hence, the approach followed in this work has been to run a simulation, restarting from the isothermal solution, but considering the radiative heat fluxes only (the conduction inside the solid has been discounted). The conductive heat fluxes inside the solid, even though they are small, generate fluctuations that destabilise the solution. Considering only the radiation, the evolution of the temperature in time is smoother and the convergence can be achieved.

Once this analysis has been completed successfully and the capsule has already been cooled through the effect of thermal radiation, it is possible to include again the conductive heat fluxes without putting the convergence of the solution in danger.

In conclusion, numerical stability problems are usually complex and each particular case may require a different solution. Therefore, it is important to define a methodology to tackle the problem and to increase the complexity progressively. This way, it is easier to identify a possible problem and to remove its root before it can generate a greater problem in a future analysis.

Shock capturing

The appropriate modelling of shock waves is a challenging task for any CFD solver, because of the presence of a discontinuity within the flow field. Even using a Discontinuous Galerkin discretisation, which can handle discontinuities, and taking advantage of the shock capturing module, which improves the stability of the problem, it is important to follow some steps to avoid a numerical divergence during the simulation.

The first key aspect is to progressively increase the Mach number (and the Reynolds number) from a lower value until the final hypersonic Mach number is reached. In this case, the Mach number has been increased from $M = 2$ to $M = 7.66$, running several simulations. Each simulation has been restarted from the final solution of the previous one. This way, the severity of the shock wave changes little by little, making the convergence process easier.

The second key aspect that has to be taking into account is related to the use of the shock capturing module of HORSES3D (see sec. 3.1.3). The sensor thresholds s_1 and s_2, which determine the region where the artificial viscosity is applied, have to be tuned, through a trial and error process, to assure that this viscosity is only localised over the shock. Furthermore, the value of the artificial viscosity in each region has to be defined to avoid a divergence of the algorithm, but preserving the physical meaning of the problem and a reasonable time step.

References

[1] E. H. Hirschel, C. Weiland, Selected Aerothermodynamic Design Problems of Hypersonic Flight Vehicles, 1st Edition, Springer, Berlin, Heidelberg, 2009. `doi:https://doi.org/10.1007/978-3-540-89974-7`.

[2] The Royal Aeronautical Society, Equations for calculation of International Standard Atmosphere and associated off-standard atmospheres (november 2008).

[3] A. P. Shafeeque, CFD analysis on an atmospheric re-entry module, International Research Journal of Engineering and Technology (IRJET) 04. URL `https://www.academia.edu/31538604/IRJET_CFD_ANALYSIS_ON_AN_ATMOSPHERIC_RE_ENTRY_MODULE_SHAFEEQUE_A_P`

[4] O. Uyanna, H. Najafi, Thermal protection systems for space vehicles: A review on technology development, current challenges and future prospects, Acta Astronáutica 176 (2020) 341–356.

[5] Space thermal control team, IDR/UPM, Passive thermal control: Coatings and surface finishes (2021).

[6] D. Huergo Perea, Conducción de calor y simulación numérica, Transferencia de calor y control térmico, MUSE, IDR.

[7] F. Barbosa, E. Zaparoli, C. Andrade, Unified approach for conjugate heat-transfer analysis of high speed air flow through a water-cooled nozzle, The Aeronautical Journal 120 (2016) 355–373. `doi:10.1017/aer.2015.15`.

[8] J. M. Torrico, A high-order discontinuous Galerkin multiphase flow solver for industrial applications, Ph.D. thesis, ETSIAE-UPM (2020).

[9] A. Mauricio, Efficient space and time solution techniques for high-order discontinuous Galerkin discretizations of the 3D compressible Navier-Stokes equations, Ph.D. thesis, ETSIAE-UPM (2019).

[10] A. M. Rueda-Ramírez, J. Manzanero, E. Ferrer, G. Rubio, E. Valero, A p-multigrid strategy with anisotropic p-adaptation based on truncation errors for high-order discontinuous Galerkin methods, Journal of Computational Physics 378 (2019) 209–233. `doi:https://doi.org/10.1016/j.jcp.2018.11.009`. URL `https://www.sciencedirect.com/science/article/pii/S0021999118307277`

[11] E. Ferrer, G. Rubio, G. Ntoukas, W. Laskowski, O. A. Mariño, S. Colombo, A. Mateo-Gabín, F. M. de Lara, D. Huergo, J. Manzanero, A. M. Rueda-Ramírez, D. A. Kopriva, E. Valero, HORSES3D: a high-order discontinuous Galerkin solver for flow simulations and multi-physics applications (2022). `doi:10.48550/ARXIV.2206.09733`. URL `https://arxiv.org/abs/2206.09733`

[12] P. Wang, Y. Li, Z. Zou, W. Zhang, Conjugate heat transfer investigation of cooled turbine using the preconditioned density-based algorithm, Propulsion and Power Research 2 (1) (2013) 56–69. `doi:https://doi.org/10.1016/j.jppr.2012.10.004`. URL `https://www.sciencedirect.com/science/article/pii/S2212540X12000053`

[13] T. Verstraete, S. Scholl, Stability analysis of partitioned methods for predicting conjugate heat transfer, International Journal of Heat and Mass Transfer 101 (2016) 852–869. `doi:http://dx.doi.org/10.1016/j.ijheatmasstransfer.2016.05.041`.

[14] N. Bhandari, Planetary exploration: Scientific importance and future prospects, CURRENT SCIENCE 94.

[15] N. E. Leggett, Requirements for manned space flights to mars: Should NASA skip returning to the Moon?, Review of U.S. Human Spaceflight Plans Committee (2009) 4.

[16] J. Muylaert, W. Berry, Aerothermodynamics for space vehicles – ESA's activities and the challenges, ESA bulletin 96 (1998) 10.

[17] MFEM, Finite element discretization library (2022). URL `https://mfem.org/`

[18] H. Huynh, Z. Wang, P. Vincent, High-order methods for computational fluid dynamics: A brief review of compact differential formulations on unstructured grids, Computers Fluids 98 (2014) 209–220, 12th USNCCM mini-symposium of High-Order Methods for Computational Fluid Dynamics - A special issue dedicated to the 80th birthday of Professor Antony Jameson. `doi:https://doi.org/10.1016/j.compfluid.2013.12.007`. URL `https://www.sciencedirect.com/science/article/pii/S0045793013004829`

[19] N. Brenowitz, Call Python from Fortran (2022). URL `https://github.com/nbren12/call_py_fort`

[20] Gmsh, A three-dimensional finite element mesh generator with built-in pre- and post-processing facilities (2022). URL `https://gmsh.info/`

[21] W. Niehaus, Vehicles, heat shield concepts and materials for reentry, in: Defense Documentation Center for Scientific and Technical Documentation, Alexandria, Virginia, 1963.

[22] D. Paul, C. Clay, B. Harber, H. Croop, D. Glass, S. Scotti, Extreme environment structures, in: Structures Technology for Future Aerospace Systems, AIAA, 2000.

[23] H. Rivers, D. Glass, Advances in hot structures development, in: Thermal Protection Systems and Hot Structures Proceedings of the 5th European Workshop, European Space Agency, 2006.

[24] G. D'Aelio, J. Parker, Ablative plastics, Marcek Dekker.

[25] J. R. Olds, Results of a rocket based combined cycle SSTO design using parametric MDO methods, in: MSAE Technical Paper Series, Aerospace Atlantic Conference and Expo, Dayton, Ohio, 1994.

[26] A. M. Cary Jr., J. N. Hefner, An investigation of film-cooling effectiveness and skin friction in hypersonic turbulent flow, in: AIA 4th Fluid and Plasma Dynamics Conference, Palo Alto, California, 1971.

[27] D. E. Glass, Ceramic matric (CMC) thermal protetcion systems (TPS) and hot structures for hypersonic vehicles, AIAA.

[28] D. Greuel, A. Herbertz, O. J. Haidn, M. Ortelt, H. Hald, Transpiration cooling applied to C/C liners of cryogenic liquid rocket engines, in: 40th AIAA/ASME/ SAE/ASEE/JPC Conference and Exhibit., Fort Lauderdale, Florida, 2004.

[29] ESA-ESTEC, Noordwijk, The Netherlands , ECSS-E-ST-10-04C (november 2008).

[30] W. L. Ko, L. Gong, R. D. Quinn, Reentry thermal analysis of a generic crew exploration vehicle structure, National Aeronautics and Space Administration.

[31] Isidoro Martínez academic web-site (April 2022).
URL `http://webserver.dmt.upm.es/~isidoro/`

[32] B. Laub, New TPS materials for aerocapture, AIP Conference Proceedings 608 (2002) 337. `doi:https://doi.org/10.1063/1.1449742`.

[33] S. M. Johnson, Coatings and surface treatments for reusable entry systems, NASA Ames Research Center (2016) 25.
URL `https://ntrs.nasa.gov/citations/20160003291`

[34] J. R. Howell, The Monte Carlo method in radiative heat transfer, Journal of Heat Transfer 120.

[35] A. Vreman, An eddy-viscosity subgrid-scale model for turbulent shear flow: Algebraic theory and applications, Physics of fluids 16 (10) (2004) 3670–3681.

[36] B. Cockburn, G. E. Karniadakis, C. W. Shu, The development of discontinuous Galerkin methods, in: Computational Science and Engineering, Vol. 11, Springer, Berlin, Heidelberg, 2000. `doi:https://doi.org/10.1007/978-3-642-59721-3_1`.

[37] D. A. Kopriva, Implementing spectral methods for partial differential equations: Algorithms for scientists and engineers, Springer Science & Business Media.

[38] U. Piomelli, Large-eddy simulation: achievements and challenges, Progress in Aerospace Sciences 35 (1999) 335–362.

[39] S. T. Johansen, J. Wu, W. Shyy, Filter-based unsteady RANS computations, International Journal of Heat and Fluid Flow 25 (2004) 10–21.

[40] P. Manna, D. Chakraborty, Numerical investigation of conjugate heat transfer problems.

[41] B. John, P. Senthilkumar, S. Sadasivan, Applied and theoretical aspects of conjugate heat transfer analysis: A review, Arch Computat Methods Eng 26 (2019) 475–489. `doi:https://doi.org/10.1007/s11831-018-9252-9`.

[42] A. R. Crowell, B. A. Miller, J. J. McNamara, Robust and efficient treatment of temperature feedback in fluid–thermal–structural analysis, AIAA 52 (2014) 2395–2413. `doi:https://doi.org/10.2514/1.J052820`.

[43] P. Dechaumphai, A. R. Wieting, E. A. Thornton, Flow-thermal-structural study of aerodynamically heated leading edges, J. Spacecraft 26 (4) (1989) 201–209. `doi:https://doi.org/10.2514/3.26055`.

[44] H. Guiqing, Z. Weijiang, Two dimensional coupled flow-thermal structural numerical simulation, Acta Aerodyn Sin 1 (2000) 019.

[45] N. Hosters, M. Klaus, M. Behr, H.-G. Reimerdes, Application of a partitioned field approach to transient aerothermal problems in rocket nozzles, Computers & Fluids 88 (2013) 795–803.

[46] Q. Liu, E. Luke, P. Cinnella, Coupling heat transfer and fluid flow solvers for multidisciplinary simulations, Journal of Thermophysics and Heat Transfer 19 (4) (2005) 417–427.

[47] A. R. Crowell, B. A. Miller, J. J. McNamara, Loosely coupled time-marching of fluid–thermal–structural interactions, in: 54th AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics, and materials conference, Boston, MA, 2013.

[48] R. K. Peetala, Conjugate heat transfer analysis in hypersonic applications, Ph.D. thesis, Indian institute of technology Guwahati (2014).

[49] X. Zhao, Z. Sun, L. Tang, G. Zheng, Coupled flow-thermal-structural analysis of hypersonic aerodynamically heated cylindrical leading edge, Engineering Applications of Computational Fluid Mechanics 5 (2) (2011) 170–179.

[50] R. Ranjan, L. Catabriga, Y. Feng, High-order spectral/hp-based solver for compressible Navier–Stokes equations, AIAA`doi:{https://doi.org/10.2514/1.J060704}`.

[51] N. M. Roy, A. P. Shafeeque, Hypersonic flow analysis on an atmospheric re-entry module, International Journal of Engineering Research and General Science 3 (5).

[52] M. J. Wright, D. K. Prabhu, E. R. Martínez, Analysis of Apollo Command Module afterbody heating part I: AS-202, Journal of thermophysics and heat transfer 20 (1).

[53] A. Martin, L. C. Scalabrin, I. D. Boyd, High performance modeling of atmospheric re-entry vehicles, Journal of Physics: Conference Series 341.

[54] A. Fedele, R. Gardi, G. Pezzella, Aerothermodynamics and thermal design for on-ground and in-fight testing of a deployable heat shield capsule, CEAS Space Journa 12 (2020) 411–428. `doi:https://doi.org/10.1007/s12567-020-00312-w`.

[55] A. Appar, R. Kumar, S. K. Naspoori, Conjugate flow-thermal analysis of a hypersonic reentry vehicle in the rarefied flow regime, Physics of fluids 34. `doi:https://doi.org/10.1063/5.0082783`.

[56] M. Paipuri, C. Tiago, S. Fernández-Méndez, Coupling of continuous and hybridizable discontinuous Galerkin methods: Application to conjugate heat transfer problem, Journal of Scientific Computing 78 (2019) 321–350. `doi:https://doi.org/10.1007/s10915-018-0769-8`.

[57] M.-P. Errera, R. Moretti, J. Mayeur, M. Gelain, L. Tessé, J.-M. Lamet, E. Laroche, A numerical predictive model for conjugate heat transfer with radiation, International Journal of Heat and Mass Transfer 160. `doi:https://doi.org/10.1016/j.ijheatmasstransfer.2020.120155`.

[58] D. Huergo, Anomaly detection in time series for space applications, IDR-UPM. URL `https://github.com/Dhueper/TimeSeries-AnomalyDetection`

[59] A. Luikov, Conjugate convective heat transfer problems, International Journal of Heat and Mass Transfer 17 (2) (1974) 257–265.

[60] Numpy, F2PY user guide and reference manual (2022).
URL `https://numpy.org/doc/stable/f2py/index.html`

[61] D. Huergo Perea, N. Martínez Figueira, M. Ramiro Aguirre, P. Romero Ramos, Fortran to Python wrapper (2020).
URL
`https://github.com/mramagu/Fortran-to-Python-Wrapper/releases/tag/1.0`

[62] CFFI (2022).
URL `https://cffi.readthedocs.io/en/latest/`