# Internship Report



# Lyon 1 Claude Bernard University
# Polytech Lyon
# Academic Year 2024/2025

*Author:*
Baptiste Ridolfi

*Supervisors:*
Prof. Aiko Yakeno
Julien Landel

# Contents

# Chapter 1

# Introduction

## 1.1 Context

One of the most significant challenges in the aerospace industry is to predict the heat flux generated by hypersonic aerodynamic heating during atmospheric re-entry. Currently, the heat shield the most widely used in the space industry is the **ablative heat shield**. For instance, NASA's Orion capsule uses a Boeing-made shield that weighs approximately 1.36 tons, which makes the whole capsule weigh around 9.3 tons, in other terms, 15% of the capsule's global weight [6]. In space missions, weight management is crucial : reducing structural weight means more technical equipment or fuel can be carried. So, engineers constantly seek to reduce it as much as possible. To better understand the behavior of the heat shield in such extreme conditions, space agencies conduct numerical simulations, laboratory experiments, and full-scale experiments. This multi-step approach was notably adopted by NASA during the Apollo program [5].

To reduce dependency on such expensive and complex tests, the Japan Aerospace Exploration Agency (JAXA) developed the High Enthalpy Shock Tunnel (HIEST) in the late 1990s, the largest facility of its kind in Japan [4][15]. HIEST enables researchers to replicate high-enthalpy flow conditions which are the environments of hypersonic flight as long as plasma formation.
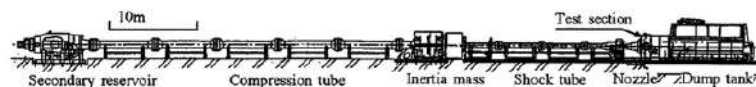
Figure 1.1.1: HIEST facility and schematic explanation[15]

However, most of the experiments have been conducted in wind tunnel and these are not fully

reliable, as noticeable disturbance increases with the Mach number. In such cases, turbulence disturbances lead to a misunderstanding of the heat flux, which affects the accuracy of the results. [7].

## 1.2  Main goals

Dr. Kento Inokuma will conduct testing at the beginning of July in order to obtain many experimental data from a ballistic range to avoid disturbance from wind tunnels. His experiment consists of taking an 8 mm-diameter ball and passing it through a tube at 2 km/s to reproduce the conditions obtained by JAXA and thus, obtain a schlieren photography and heat flux data. These experiments are crucial to evaluate whether the theoretical and numerical results align with physical reality. The main goal of this internship is to develop a numerical prediction method that will primarily be used to compare with the experimental results obtained during the project.

It can be obtained by simulating numerically different results around a vehicle entering the atmosphere. By developing, firstly, simplified computational models such as a cylinder and then a spherical geometry. Those two geometries, depending on whether you are in a two-dimensional or three-dimensional space, are the closest we can get to the Apollo-shaped capsule and, more importantly, the simplest and most affordable solution to conduct real-world experimentation.

This leads to the following problem : **How can we design a simplified model that remains accurate and close to physical reality, while reducing computational time and resource consumption?**

To address this question, my work will be organized into three distinct parts : First, **Numerical Methods** explaining how the computation works, and how the geometry and meshing are chosen and made. Second, **Results Analysis** to discuss the method used and the validity of our results, and finally a detailed discussion on the continuation of my internship and the work to be carried out following the experiment.

Due to the sensitive nature of the research and development involved, this internship is subject to certain confidentiality agreements.

This internship took place in the IFS at Tohoku University in Sendai, in the Aerospace Fluid Laboratory conducted by Prof. Aiko Yakeno. Composed of 6 people, I joined Dr. Kento Inokuma in his research in collaboration with JAXA on the heat transfer around the re-entry capsule.

## 1.3  Presentation of the Host Institution

### 1.3.1  Tohoku University

The laboratory is part of Tohoku University. Created in 1907, it stands as one of the greatest universities of Japan alongside Tokyo and Kyoto University. It is located in Sendai, in Miyagi pre-

fecture. The university is known worldwide for its emphasis on research and innovation, mainly in space and material engineering, science, and technology fields. This makes the university one of the most prestigious in the world, ranking constantly in the top 100 according to QS and Times Higher Education. It has strong and active partnerships with more than a hundred universities worldwide, including one with **Université de Lyon**. Those partnerships offer exchange programs and courses in both English and Japanese, mainly for graduate students, such as the Special Research Student program, which allows me to carry out my internship at this prestigious university.

Tohoku University is world-wide known for its excellence in research, particularly in fluid mechanics, space engineering, materials, and robotics. It can count on strong partnerships with government and industrial agencies (JAXA for example, or Japan Science and Technology Agency (JST)). The university is home to top-tier research facilities, notably **the Institute of Fluid Science** which play a central role in high-level fluid dynamics and aerospace investigations.

### 1.3.2   Institute of Fluid Science (IFS)

The Institute of Fluid Science (IFS) is one of Tohoku University's most efficient and renowned institutions, especially internationally for its work in fluid dynamics, aerospace science, and energy systems. Created in 1974, IFS specializes mainly in fluid dynamics, aerospace science, plasma research, and energy. It is equipped with the most advanced and cutting-edge technology to perform precise experiments numerically with a supercomputer composed of several thousand processors and terabytes of RAM; it is an ultra-powerful computer capable of performing in a few days calculations that would take several months on a standard PC, and also experimentally with wind tunnels, plasma labs, etc. The institute collaborates with major space organizations such as JAXA and NASA for my laboratory. The laboratory hosting me, the Aerospace Fluid Engineering Laboratory, is currently led by Professor Aiko Yakeno and is focused on the experiments introduced before, as well as on aircraft wing optimization and the study of fuselage aerodynamics for specific airplane models. The laboratory operates in a collaborative manner with regular meetings with the main professor that not only allow researchers to share their results with others, receive advice and guidance for their works, but also ensure that individual projects align with the laboratory's direction.[14]

# Chapter 2

# Numerical methods

## 2.1   Numerical conditions

As mentioned before, we are simulating a simplified model of a capsule entering the atmosphere at Mach 6.3. This kind of flow is highly complex and involves several important physical phenomena that must be taken into account to get accurate and realistic results.

To better understand the physical context and to validate our approach, it is relevant to refer to experimental data obtained under similar conditions. JAXA conducted various experiments on an Apollo capsule-shaped using the HIEST[4]. These results, which include surface pressure, heat flux, and flow characteristics with boundary layer trip devices, are accessible in the report titled *Aeroheating measurement of Apollo shaped capsule with boundary layer trip in the free-piston shock tunnel HIEST* [12].

However, more recent values have been obtained by JAXA that are not available in this document. Please refer to the table below for a summary of key physical parameters used in this study, based on those additional unpublished or internal datasets. As I am not certain about the confidentiality status of the exact data, the values presented here are approximate but representative of the conditions actually used.

| $U_\infty$ (m/s) | $P_\infty$ (MPa) | $T_\infty$ (K) | $\rho_\infty$ (kg/m$^3$) | $Re_\infty$ | $H_0$ (MJ/kg) |
|---|---|---|---|---|---|
| 4013 | 0.00751 | 1010 | 0.026 | $2.0 \times 10^6$ | 9 |

Table 2.1: Approximate experimental values used in the following simulations

Firstly, the flow is turbulent mainly due to the high Reynolds number under these conditions. While transition to turbulence typically occurs around $Re = 2 \times 10^5$, flow instabilities such as the von Kármán vortex street can already appear around $Re = 10^4$. In our case, $Re_\infty \gg 2 \times 10^5$, so the flow is clearly **turbulent**. Turbulence introduces chaotic structures in the shock wave, behind the capsule, generating strong gradients of temperature, pressure, and velocity. This increases the computational complexity and often leads to numerical instabilities.

Then, a high gradient of temperature can lead to important heat flux. Heat transfers can occur through three primary mechanisms : radiation, conduction and convection. For the following simulations, we are considering high-speed vehicle entrance in the atmosphere. In this case, radiation and conduction can be neglected alongside convection. Convection can be characterized by

a dimensionless number and a physical quantity.

- **Newton's cooling law :** $Q = h \cdot (T_\infty - T_{wall})$ in $W/m^2$
- **Stanton number :** $St = \frac{h}{\rho \cdot C_p \cdot U}$ or $St = \frac{Nu}{Re \cdot Pr}$

Both results indicate the heat transfer at the capsule's shield. Newton's law gives quantitative values with physical dimensions, making it fundamental in thermodynamics. However, it can't be compared with other flows because it depends directly on velocity and fluid properties.

On the other hand, Stanton number integrates the Reynolds number in its formula. Thus, for the same geometrical model, we can compare different results from different flows without making it "unfair".

We will be using the Stanton number for the rest of our simulation.

## 2.2 OpenFOAM

All simulations in this study were performed using OpenFOAM, an open-source CFD software widely used in both academia and industry. It runs on Linux and is written in C++. OpenFOAM offers many solvers, turbulence models, and numerical schemes, which makes it a flexible tool for different kinds of fluid flow problems.

In this project, two solvers were mainly used: **rhoCentralFoam** and **sonicFoam**. They are both designed for compressible flows and were chosen based on the flow conditions, especially the Mach number, to better capture the physics of high-speed flows. Indeed, sonicFoam is more suited for low Mach number (transient to supersonic flow) condition while rhoCentralFoam is more suited for high mach number(supersonic to hypersonic flows).

## 2.3 Compressible equation in fluid dynamics

In fluid dynamics, we distinguish between compressible and incompressible flows. In this case, since Ma » 1, the flow is considered exclusively compressible. Such flows are governed by three major equations: the continuity equation 2.1, the momentum equation 2.2, and the energy equation 2.3 :

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\mathbf{U}\rho) = 0 \tag{2.1}$$

$$\frac{\partial(\rho \mathbf{U})}{\partial t} + \nabla \cdot [\mathbf{U}(\rho \mathbf{U})] + \nabla p + \nabla \cdot \sigma = 0 \tag{2.2}$$

$$\frac{\partial(\rho E)}{\partial t} + \nabla \cdot [\mathbf{U}(\rho E)] + \nabla \cdot (\mathbf{U}p) + \nabla \cdot (\sigma \cdot \mathbf{U}) + \nabla \cdot \mathbf{j} = 0 \tag{2.3}$$

In the governing equations, $\rho$ denotes the fluid density, $\vec{U}$ the velocity vector, and $p$ the pressure. The total specific energy $E$ is expressed as:

$$E = e + \frac{1}{2}|\vec{U}|^2$$

where $e$ corresponds to the specific enthalpy. The viscous stress tensor, denoted $\boldsymbol{\sigma}$, is defined such that it is positive under compressive states. The dynamic viscosity is denoted by $\mu$, and $\mathbf{I}$ represents the identity tensor. The vector $\vec{j}$ corresponds to the diffusive heat flux, which is related to the thermal conductivity of the fluid.[3]

The resolution of these equations enables stable simulations of most realistic flow situations; however, they have to be implemented carefully to remain realistic. In practice, the compressible Navier–Stokes equations cannot be solved analytically within a reasonable time due to their strong coupling, non-linearity, but more importantly, the complexity of most real fluid flow. Consequently, these equations form the foundation of CFD, which is essential for accurately modeling complex fluid dynamics problems.

## 2.4    Turbulence models

In computational fluid dynamics (CFD), a wide range of turbulence models are available to simulate fluid behavior under turbulent conditions. However, this comes with a trade-off: a higher resolution of turbulence requires significantly more computational time and resources, whereas lower-fidelity models offer faster results but deviate further from physical accuracy.

In this study, two turbulence modeling approaches are used. Reynolds-Averaged Navier-Stokes (RANS) and Large Eddy Simulation (LES). The RANS model is based on the time-averaged resolution of the Navier-Stokes equations (2.1), (2.3), (2.2) by modeling all the fluctuating components. It is the most commonly used model due to its relatively low computational costs and satisfying results [10]. On the other hand, LES focuses on solving large-scale unsteady turbulent structures by using a low-pass filter in order to resolve directly large eddies and focuses on modeling only the smallest subgrid ones [9].

To close the RANS equations, turbulence models such as the $k$-$\varepsilon$ and $k$-$\omega$ Shear Stress Transport (SST) models are employed to approximate the effects of turbulent fluctuations on the mean flow.

In the $k$-$\varepsilon$ model, the turbulent viscosity is defined as:

$$\mu_t = C_\mu \, \rho \, \frac{k^2}{\varepsilon}$$

where $k$ is the turbulent kinetic energy, $\varepsilon$ is the turbulent dissipation rate, and $C_\mu$ is a model constant (typically 0.09).

By solving the Navier-Stokes equations using this model, we obtain [2] :

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial(\rho k u_i)}{\partial x_i} = \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right] + G_k + G_b - \rho\varepsilon - Y_M + S_k \qquad (2.4)$$

$$\frac{\partial(\rho\varepsilon)}{\partial t} + \frac{\partial(\rho\varepsilon u_i)}{\partial x_i} = \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu}{\sigma_\varepsilon}\right)\frac{\partial\varepsilon}{\partial x_j}\right] + C_{1\varepsilon}\frac{\varepsilon}{k}(G_k + C_{3\varepsilon}G_b) + C_{2\varepsilon}\rho\frac{\varepsilon^2}{k} + S_\varepsilon \qquad (2.5)$$

where

- $G_k$ represents the generation of turbulence kinetic energy due to the mean velocity gradients.

- $G_b$ is the generation of turbulence kinetic energy due to buoyancy.

- $Y_M$ represents the contribution of the fluctuating dilatation in compressible turbulence to the overall dissipation rate.

- $C_{1\varepsilon}, C_{2\varepsilon}$, and $C_{3\varepsilon}$ are constants.

- $\sigma_k$ and $\sigma_\varepsilon$ are the turbulent Prandtl numbers for $k$ and $\varepsilon$, respectively.

- $S_k$ and $S_\varepsilon$ are user-defined source terms.

In the $k$-$\omega$ SST model, the turbulent viscosity is given by:

$$\mu_t = \rho\,\frac{k}{\omega}$$

where $\omega$ is the specific dissipation rate.

After putting k-$\epsilon$ modes into the Navier-Stokes equations 2.1, we obtain the transport equations for this model [11] :

$$\frac{\partial}{\partial t}(\rho k) + \nabla \cdot (\rho\vec{v}k) = \nabla \cdot (\Gamma_k\nabla k) + G_k - Y_k \qquad (2.6)$$

$$\frac{\partial}{\partial t}(\rho\omega) + \nabla \cdot (\rho\vec{v}\omega) = \nabla \cdot (\Gamma_\omega\nabla\omega) + G_\omega - Y_\omega + D_\omega \qquad (2.7)$$

- $\rho$ is fluid density,

- $\vec{v}$ is flow velocity vector,

- $k$ is turbulence kinetic energy,

- $\omega$ is turbulence dissipation,

- $G_k$ and $G_\omega$ are the generation rate of turbulence kinetic energy and turbulence dissipation,

- $\Gamma_k$ and $\Gamma_\omega$ are the diffusion rate of $k$ and $\omega$,

- $Y_k$ and $Y_\omega$ are turbulence generated due to diffusion,

- $D_\omega$ is the orthogonal divergence term.

The $k$-$\varepsilon$ model is well-suited for regions far from walls due to its low computational cost, whereas the $\omega$ model is more accurate near wall boundaries but requires greater computational effort. The SST model combines the advantages of both by using the $\omega$ model in the near-wall region and transitioning to the $\varepsilon$ model in the far field, thereby improving accuracy in shock regions close to the wall as well as in regions farther away, making it the perfect model in our case.[1]

## 2.5  Schemes

To solve the governing equations introduced earlier, OpenFOAM uses various numerical schemes. These schemes approximate time and space derivatives, aiming to balance accuracy, stability, and computational cost.

The main solver used is **rhoCentralFoam**. It is an explicit solver designed for high-speed compressible flows. Based on the central-upwind Kurganov-Tadmor scheme, it is well suited for hypersonic flows and effectively handles shock waves [3].

Time derivatives are usually discretized with a first-order explicit Euler scheme, which is simple and stable only if the time step is small enough. Moreover, spatial derivatives like gradients ($\frac{\partial}{\partial n}$) and Laplacian ($\Delta n$) are mostly discretized using implicit second-order central differencing schemes, providing good accuracy.

Convective terms, especially in turbulence transport equations, require careful treatment to maintain numerical stability. For these, first-order upwind schemes are used, which are more diffusive but ensure robustness.

This mix of explicit and implicit schemes allows OpenFOAM to efficiently simulate compressible turbulent flows while maintaining solution quality.

## 2.6  Computation domain

For the computational domain, several options are available, but one parameter must remain fixed. Based on Dr. Inokuma's experience, an 8 mm diameter sphere will be used. Therefore, in all simulations, we will use a cylinder and a sphere with a diameter of 8 mm.

The sphere's front region is in freestream condition, meaning that this area has non-needed information in the results. Thus, it is unnecessary to model it, as doing so would only increase the

computational cost due to the additional mesh cells that would need to be calculated. For example, when I first started working with OpenFOAM using the prism geometry, I included the front part of the domain in the model, which significantly increased the computation time by several hours. As shown in Figure 2.6.1, the flow in front of the sphere remains uniform and equal to the internal field, confirming that this region has no significant influence on the results.



Figure 2.6.1: First computation realized in my internship

In the upper and lower regions, the domain must be extended far enough to accurately capture the shockwave. In OpenFOAM, boundary conditions can be defined to allow the shockwave to pass through the domain boundaries. Therefore, a compromise must be made between capturing sufficient physical information and minimizing computational resources.

However, for the bottom region, since RANS provides a time-averaged solution, we can apply a symmetry condition to the geometry. As we are modeling a sphere moving through a tunnel, the flow is symmetric with respect to the axis of the sphere's motion, as shown in the figure 2.6.2.
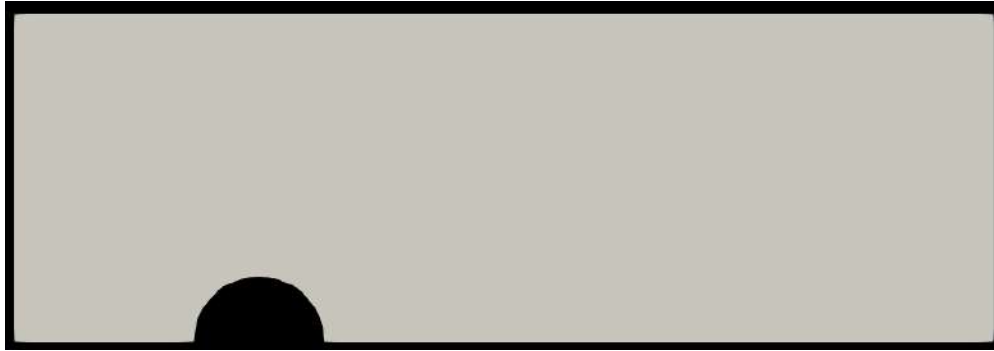


Figure 2.6.2: Symmetry on the cylinder geometry in 2D

The region behind the cylinder or sphere works in conjunction with the upper and lower walls. To capture shockwave information, this region must be long enough — but not excessively so. In fact, to calculate the shockwave angle, only the initial portion of the shockwave is required.

## 2.7   Meshing

Meshing is the most crucial step in the simulation process. High-quality meshing is essential to obtain high-fidelity results. However, generating such meshes often requires significant time and

computational resources. In that case, finding a compromise between mesh quality and computational cost is necessary, and several options are available to achieve this balance.

## 2.7.1 Meshing in OpenFOAM

Firstly, meshing can be performed directly in C++ within OpenFOAM. This approach is well-suited for simple geometries such as squares, triangles, and other basic shapes, and it allows users to avoid relying on external software. However, it is not suitable for complex geometries like aircraft, ships, or spacecraft because you perform at the same time geometry and meshing making it too difficult to do it. Moreover, this method offers limited control over the mesh refinement, such as adapting cell sizes near curved surfaces—for instance, making cells thinner near the surface of a sphere. That is the reason that I avoided this method for the rest of the computation.

## 2.7.2 External software and OpenFOAM

As mentioned earlier, OpenFOAM allows users to import both geometry and meshes from external CFD or CAD software, which significantly enhances its flexibility and capabilities. Two main approaches are possible: the first is to create only the geometry in a CAD software—such as FreeCAD, which I used in my case—and then perform the meshing within OpenFOAM with **SnappyHexMesh**. The second option is to perform both the geometry creation and meshing in an external tool, such as ANSYS Fluent or ICEM CFD (an ANSYS meshing software), before importing the mesh into OpenFOAM using **fluent3DMeshToFoam**.

### 2.7.2.1 SnappyHexMesh

At the beginning of my internship, it took some time to set up my login for the supercomputer, which allows access to many advanced and paid software tools such as Ansys. Thus, I used FreeCAD, an open-source 3D modeler, to create the geometries. After modeling our cylinder, we import it to OpenFOAM with **snappyHexMesh** command.

As we can see in Figure 2.7.1, snappyHexMesh creates rectangular cells for the entire model, and then the user can choose to additionally input thinner cell layers and many other options.
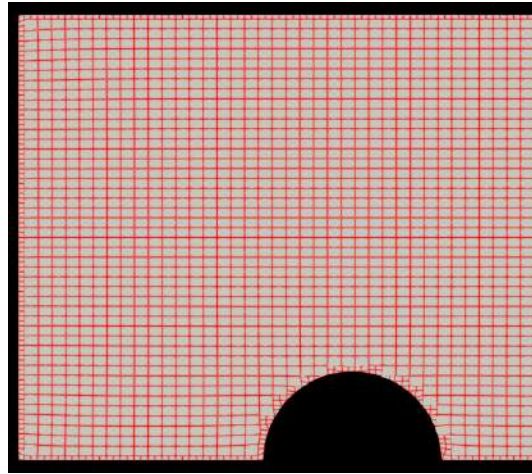
Figure 2.7.1: Mesh using snappyHexMesh

### 2.7.2.2    IcemCFD and fluent3DMeshToFoam

For the rest of the computation until the end of my internship, we will be using IcemCFD to create both geometry and meshing. Icem CFD, which stands for Integrated Computer Engineering and Manufacturing (ICEM) for Computational Fluid Dynamics (CFD) is an advanced mesh generation software primarily used in the fluid dynamics industry.

This software provides full control over the meshing process, offering two main approaches: unstructured (Figure 2.7.3) and structured meshes(Figure 2.7.2) .

The structured mesh has many advantages. Firstly, it's best suited for simple geometry such as a sphere. Moreover, its regular cell structure improves computational efficiency, allowing faster simulations. And finally, mesh generation is fully controlled by the user.
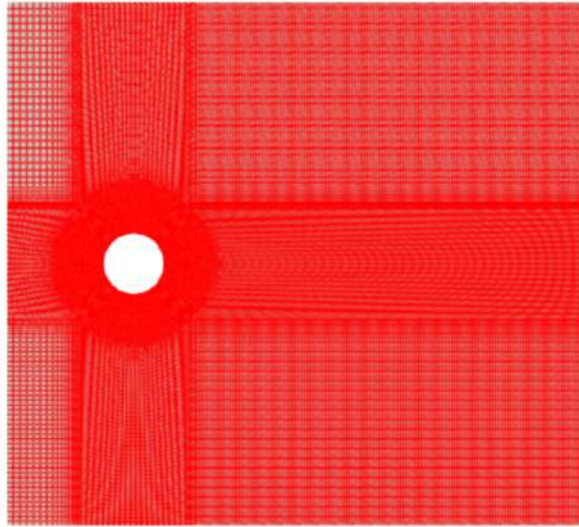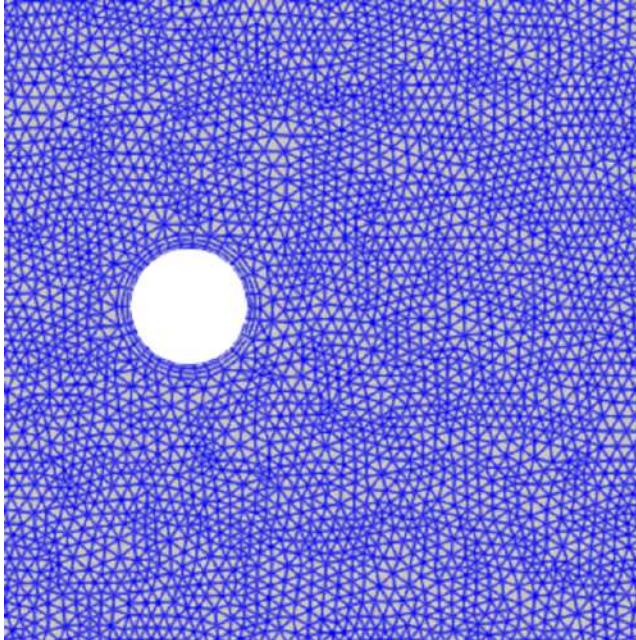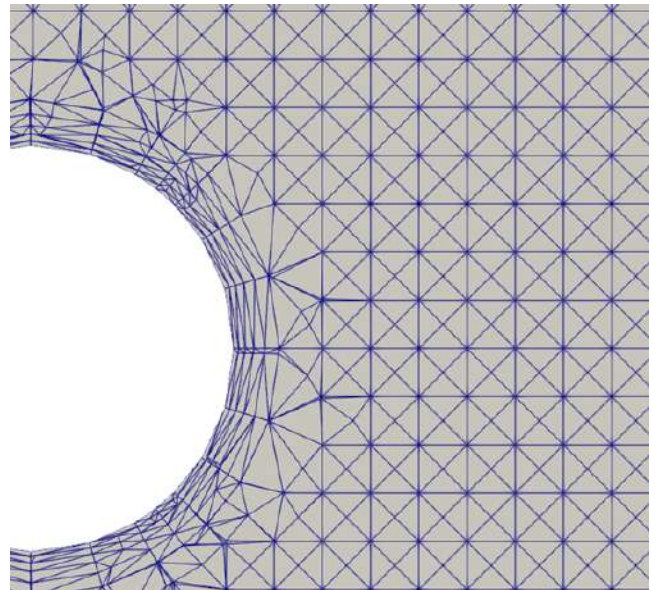
(a) Structured mesh
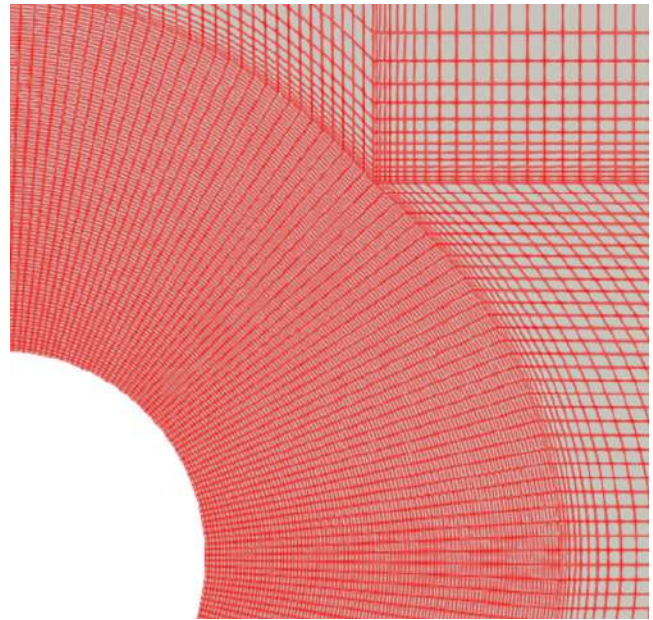
(b) Zoom on structured mesh

Figure 2.7.2: Structured mesh and zoomed view

On the other hand, unstructured mesh allows for better local refinement, around the vehicle in our case or even on the shockwave details.



(a) Unstructured mesh

(b) Zoom on unstructured mesh

Figure 2.7.3: Unstructured mesh and zoomed view

**Control over the mesh is the main reason: placing more cells around the vehicle improves the accuracy of results for heat flux and shockwave formation, while saving**

**time and computational resources in other regions of the domain.**

## 2.8   Boundary conditions

Again OpenFOAM offers a lot of different options, but if we take a closer look at Dr. Inokuma's experiences, we know easily what to input. In total, we need to control 8 parameters :

- Pressure : P

- Velocity : U

- Temperature : T

- Turbulent Kinetic Energy : k

- Turbulent Frequency : $\omega$

- Turbulent Dissipation Rate : $\epsilon$

- Turbulent thermal diffusivity : $\alpha_t$

- Turbulent kinematic viscosity : $\nu_t$

First, P, U and T can be described with the same boundary conditions for every aspect except only one. As the ball is moving at 2 $km/s$, we can consider it nearly instantaneous. Thus, inlet can be defined as **Fixed Values** with the values specified in the tab 2.1.

Then, for the outlet, we have three options. Since no specific information is required at the outlet, we can use **calculated**, **inletOutlet**, or **waveTransmissive** boundary conditions to allow the fluid to exit the domain freely.

For the upper and lower wall, we want to simulate real-life experiments. So, to simulate the tunnel's wall, we input **zeroGradient** for P, T and U. In OpenFOAM, the **zeroGradient** boundary condition imposes a Neumann-type condition where the normal derivative of the field is set to zero at the boundary:

$$\frac{\partial \phi}{\partial n} = 0$$

The left and right walls are defined as **cyclicAMI** boundaries to simulate an infinite domain in the lateral direction.

Finally, since the experiment will last only a few seconds, we can consider the temperature fixed throughout the process. To accurately capture the shockwave formation, we apply the **noSlip** boundary condition on velocity, and **zeroGradient** conditions on pressure and temperature at solid surfaces.

For the other parameters, most of them are set to **calculated** in order to save computation time.

## 2.9    Computation stability

Due to the high velocity, the presence of strong shock waves in hypersonic flows, and the significant temperature difference between the vehicle and the surrounding flow, numerical simulations can easily diverge. To ensure the simulation runs to completion, one key parameter must be carefully controlled.

### 2.9.1    Courant number and step time

One of the indicators to look out for to ensure the computation has a better chance to run is the Courant number. It is defined as follows

$$Co = \frac{U\Delta t}{\Delta x} \tag{2.8}$$

with $\Delta t$ the time step, $U$ the velocity, and $\Delta x$ the characteristic size of cells.

If $Co < 1$, it means that a fluid particle remains within a single cell during one time step. More generally, **if** $Co < 1$, the computation is expected to remain stable.

So if the velocity is around $4000 m.s^{-1}$ and one cell size can be $5e-5\,m$, then we need at least $\Delta t < 10^{-8}\,s$.

## 2.10    Model validation

Since my objective is to identify the most suitable model in terms of computational time and resources, while still achieving realistic and accurate results, three key aspects must be validated throughout the entire simulation process.
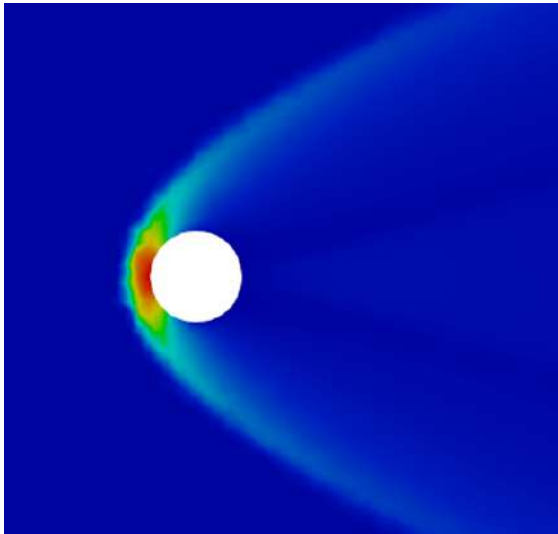
### 2.10.1    Shock wave angle

Measuring the shock wave angle is the simplest and most visually intuitive method for validating a simulation. However, it comes with significant uncertainty: the result heavily depends on the person performing the measurement — as selecting different points can lead to different angles — and on the mesh quality. A coarse mesh increases the uncertainty even further.

Furthermore, in order to compare the shock wave angle in separated shock configurations, a proper reference is required. However, no mathematical formula or abacus currently exists for such cases. As shown in Appendix .1, when the flow deflection angle $\omega$ — defined as the angle between the vehicle and the incoming flow — exceeds **40°** at a Mach number of $Ma = 6$, no single shock angle

solution exists. Instead, the shock wave detaches from the body, resulting in a separated shock configuration.

For example, setting aside data accuracy, let us focus solely on the shockwave angle.
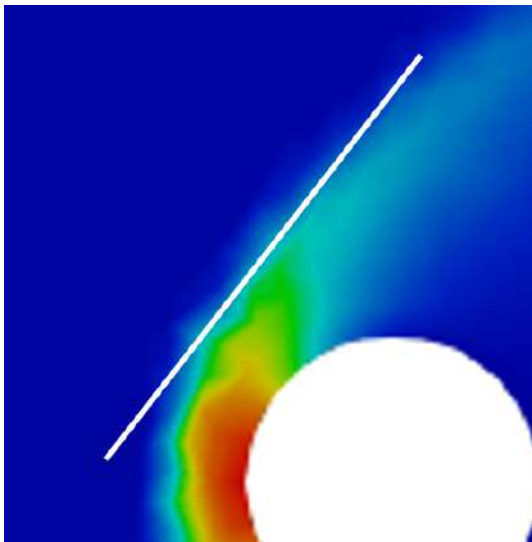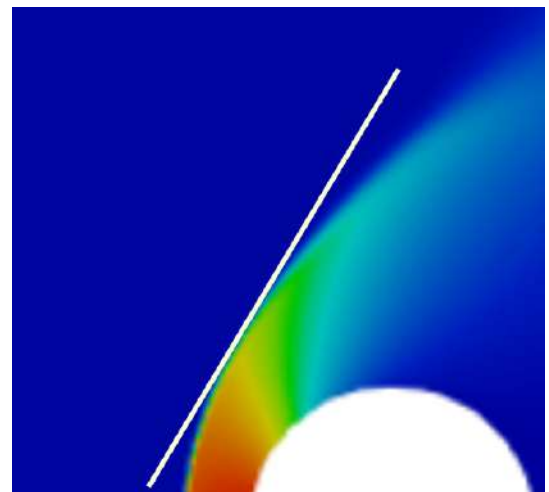


(a) Bad mesh quality
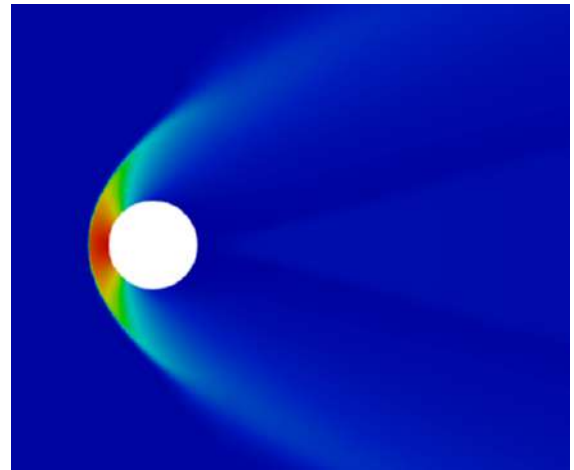
(b) Good mesh quality

Figure 2.10.1: Comparison between poor and fine mesh quality

In figure 2.10.1, it's difficult to have the same shock wave angle even if we pick the same angle. The discontinuity of the mesh can lead to misreading.



(a) Bad mesh quality

(b) Good mesh quality

Figure 2.10.2: Comparison between faulty and improved mesh

Firstly, setting aside the fact that the point at which the angle is taken is not in the same place,

we can see that this is difficult to have the angle tangent at the point we are looking. Using simple trigonometry formulas $\theta = atan(\frac{y}{x})$, we obtain :
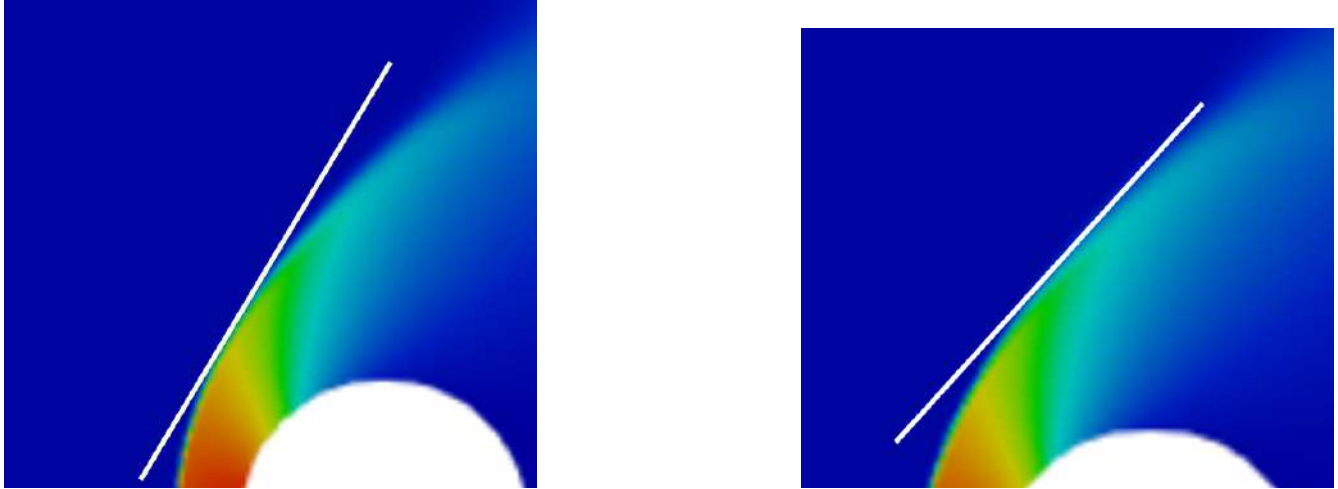
- (a) : 52.04203917 °

- (b) : 59.0362434 °



Figure 2.10.3: Two different versions of corrected mesh

In this figure, we can see that it is difficult to select the exact same point each time, which leads to inconsistent shock wave angle measurements.

- (a) : 59.0362434 °

- (b) : 47.9050555 °

In conclusion, while this method provides a visual estimation of the shock wave angle, it lacks sufficient reliability to serve as the sole validation criterion. Therefore, an additional validation approach is required. Furthermore, we can't use this method unless an experience in the same condition have been led and published.

## 2.10.2 Rankine–Hugoniot equations

The most reliable and efficient method for estimating flow conditions behind a shock wave is to use the Rankine–Hugoniot equations [8]. These equations allow us to calculate key variables such as the pressure and temperature immediately after the shock.

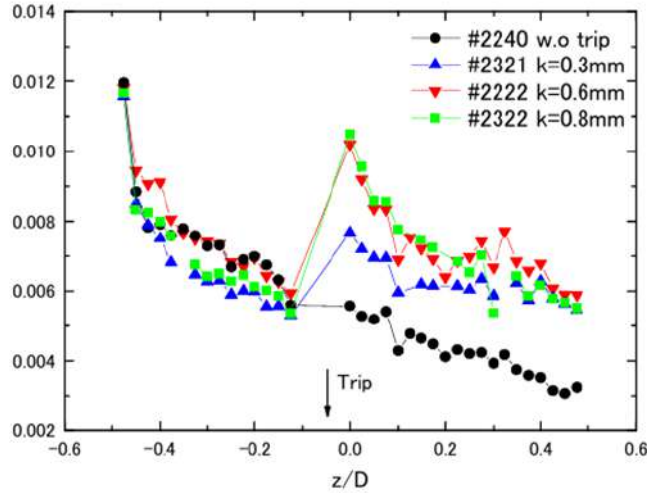$$\frac{p_2}{p_1} = 1 + \frac{2\gamma}{\gamma + 1}\left(M_1^2 - 1\right) \tag{2.9}$$

Figure 2.10.4: Stanton number along the heat shield at 9 MJ/kg (extracted from [12]).

$$\frac{T_2}{T_1} = \frac{\left(2\gamma M_1^2 - (\gamma - 1)\right)\left((\gamma - 1)M_1^2 + 2\right)}{(\gamma + 1)^2 M_1^2} \tag{2.10}$$

In these equations 2.9 and 2.10, $p_1$, $T_1$, $\gamma = 1.4$, and $M_1 = 6.3$. Using these values, $p_2$ and $T_2$ can be computed and directly compared with the results obtained from the simulation. We obtain :

- $$\frac{p_2}{p_1} = 46.138333 \tag{2.11}$$

- $$\frac{T_2}{T_1} = 8.658445 \tag{2.12}$$

### 2.10.3  Dr. Tanno's paper

Another method for validating the simulation results is to compare them with experimental data from Tanno's study. In this paper, the authors present experimental results in graphical form based on real conditions in the HIEST shock tunnel. As shown in the graph 2.10.4 (adapted from Tanno et al., 2013) [12], for a specific total enthalpy of 9 MJ/kg, the measured **Stanton number** lies between 0.005 and 0.010. If the simulation results fall within this range, they can be considered realistic and consistent with experimental observations.

## 2.11  Conclusion

This study highlights the importance of mesh quality, turbulence modeling, and boundary conditions in achieving stable and realistic hypersonic flow simulations. With validation methods now

established, we can proceed to analyze the simulation results in detail.

# Limitations of the computation

Due to the high velocity of the flow and the presence of strong shock waves, very small time steps were required to maintain numerical stability and keep the Courant number below 1. Combined with the high-resolution meshes necessary to accurately capture the shock wave structure and thermal effects, this led to a significant increase in computational cost.

The computational domain is divided among the available processor cores; the more cores are used, the faster the simulation runs. However, during most of the internship, computational resources (both cores and memory) were limited. Full access to the maximum number of processor cores was only granted in the final week.

As a result, simulation runtime often extended over several days, and only a limited number of cases reached full convergence. Therefore, the analysis in the next section focuses on a selection of representative results.

# Chapter 3

# Results analysis

To begin with, let us take a closer look at the geometry under consideration. Since Dr. Inokuma's experience models the re-entry vehicle as a sphere, we can decompose the problem into several simplified models. First, we can approximate it as a 2D cylinder to assess how closely this model represents reality. We can then analyze two cases: one assuming symmetry and the other without symmetry.

## 3.1 Cylinder in 2D

### 3.1.1 Symmetry case

#### 3.1.1.1 Computation domain

First of all, we assumed symmetry in the model to reduce computation time, expecting it would still yield satisfactory results. The geometry was created using FreeCAD and then exported to OpenFOAM via snappyHexMesh, resulting in the following mesh:
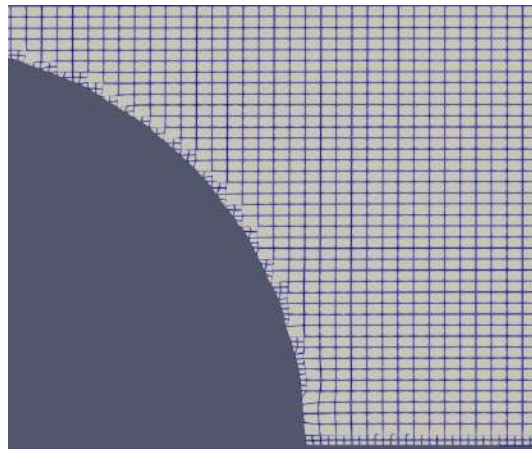


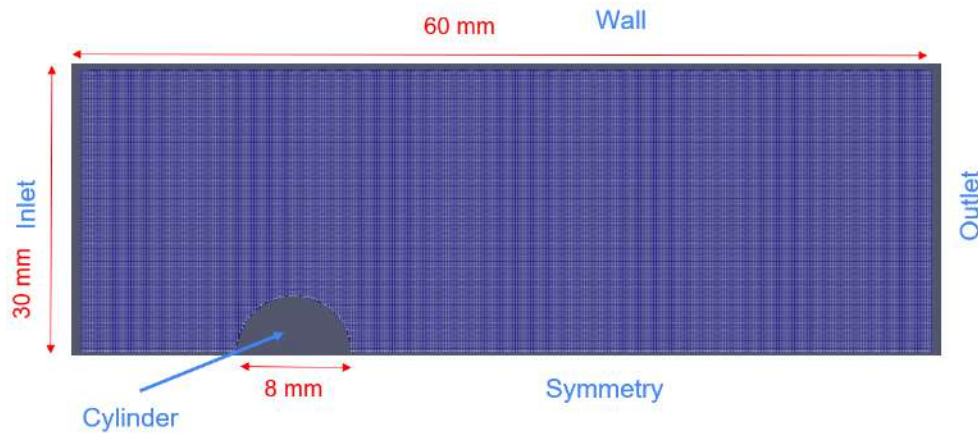Figure 3.1.1: Symmetry profile's mesh - 75 433 cells

Figure 3.1.2: Model's dimension

We can first note that the mesh layer quality is insufficient. Since flow separation occurs during the formation of the shock wave, the mesh layers may need to extend beyond the shock formation region to properly capture the physics.

Note that the model was built in 3D because OpenFOAM does not handle purely 2D simulations well. To simulate a 2D case, we used only **one** cell along the z-axis.

#### 3.1.1.2 Computation results



Figure 3.1.3: Pressure profile with scale bar

Figure 3.1.4: Temperature profile with scale bar

According to the Rankine–Hugoniot equations, the temperature observed is in the range of the theoretical value 2.10. However, the observed pressure profile is approximately twice as high as the theoretical value calculated 2.9. The mesh quality remains good according to the quality of images we obtained.



Figure 3.1.5: Reflexion on the top wall

As shown in Figure 3.1.5, a reflection occurs at the end of the shock waves. This discontinuity could be the cause of the observed pressure increase, especially if such a reflection also occurs during the initial formation of the shock waves.

Nevertheless, we also conducted the same computation using a coarser mesh and a change in boundary conditions as backups in case the primary simulation failed.

However, the results obtained with this lower-resolution mesh were the same.

As a conclusion, we decided that the symmetry wasn't good enough to be a viable model.

## 3.1.2 Full model case

For the full model case, the computations took up to three days due to the fine mesh and the increased number of cells. Two meshes were considered: a structured mesh with local refinement around the cylinder to capture more detailed information on the heat flux, and an unstructured mesh to provide global insights across the entire computational domain.

### 3.1.2.1 Computation domain

We considered the same geometry for both cases :
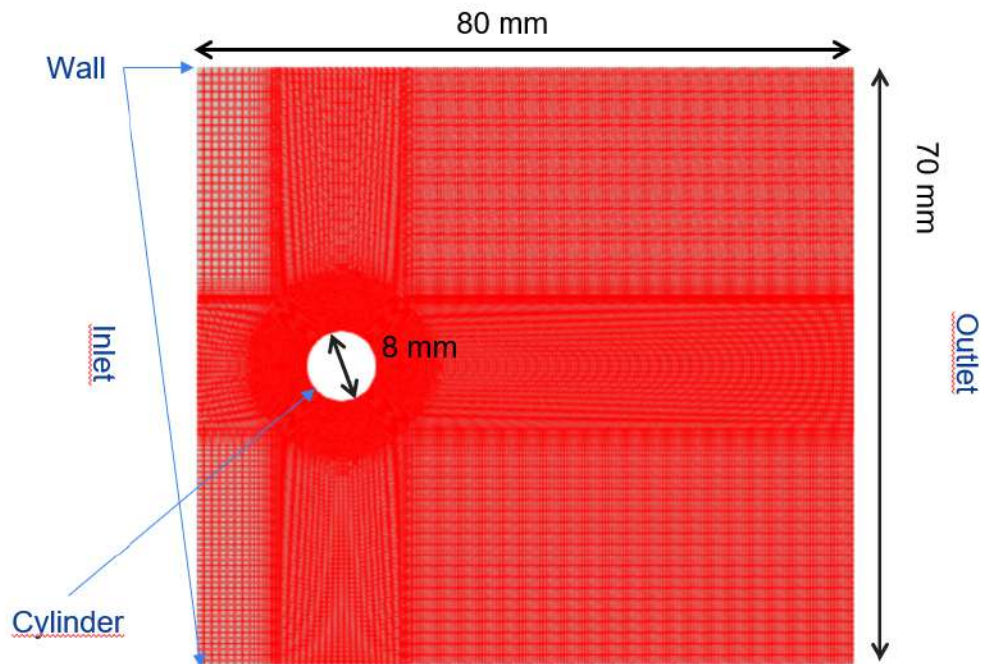


Figure 3.1.6: Full model geometry

The boundary conditions are the same as in the symmetric case, and a reflection was initially observed. Therefore, we changed the boundary conditions to **waveTransmissive** in order to prevent such reflections.

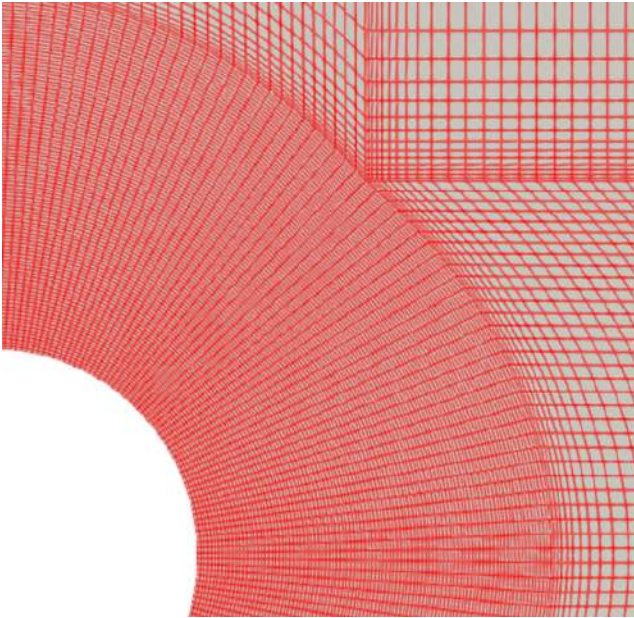The following section provides a more detailed look at the meshing.

## Structured mesh



Figure 3.1.7: Structured mesh refinement near the wall

| Property | Value |
|---|---|
| No. of cells | 383,054 |
| Cylinder number of layers | 3 |
| Cylinder growth ratio | 1.2 |
| External growth ratio | 2 |

Table 3.1: Mesh parameters

Table 3.2: Mesh characteristics

First, the refinement around the cylinder provides more precise values for the heat flux. For example, larger cells correspond to values averaged over 0.2 mm, which reduces local accuracy. Since the results need to be highly accurate, controlling the growth ratio of the mesh is essential.

Then, the structured mesh allows us to use only one cell along the z-axis, which is not possible with the unstructured mesh, as we will see now.
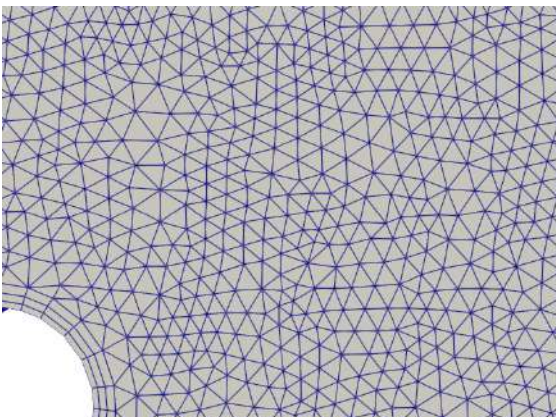
## Unstructured mesh



Figure 3.1.8: Unstructured mesh near the cylinder

| Property | Value |
|---|---|
| No. of cells | 1,561,690 |
| Cylinder growth ratio | 1.2 |
| Cylinder cell size | 2e-4 m |
| External growth ratio | 2 |

Table 3.3: Mesh parameters

The first observation is that the layer cell size is relatively large, but it may still be sufficient to capture the shock wave. As shown in Table 3.3, there is a significant increase in the number of cells. This is due to the unstructured mesh, which we could not modify. As a result, the output is displayed in 3D, even though the results remain effectively two-dimensional.

#### 3.1.2.2    Computation results

**Unstructured mesh**

Before running the computation, we can already observe in the figure below that additional refinement is needed. The round shape of the geometry is still visible in 2D, which would not be the case if the mesh were fine enough.
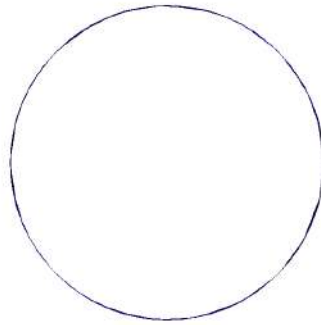


Figure 3.1.9: Pictures of cylinder in XY plan

However, the rest of the mesh appears to be fine enough to produce meaningful results. Let's proceed with the computation.
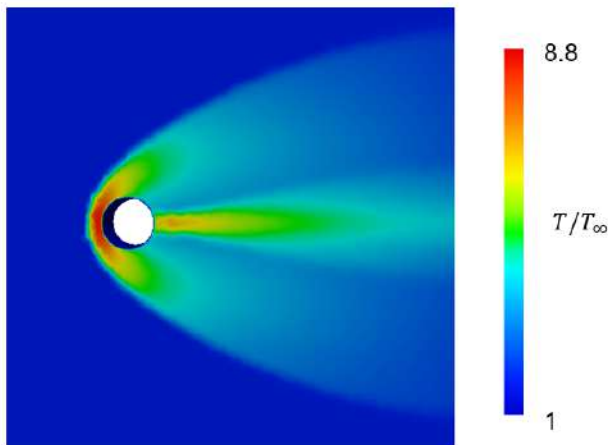


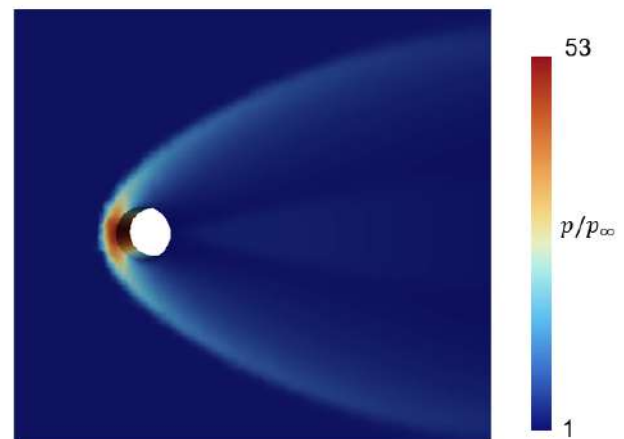Figure 3.1.10: Temperature profile along the cylinder



Figure 3.1.11: Pressure profile along the cylinder

When compared to the Rankine–Hugoniot theoretical values (Equations 2.9 and 2.10), the temperature profile matches well, while the pressure profile shows a 10% error. Nevertheless, the results remain usable and can still be analyzed.

### 3.1.2.3   Stanton number

We can expect the Stanton number to be the highest at the stagnation point. Indeed, at the stagnation point, the velocity drops to zero — it goes from the inlet velocity which is 4013 $m.s^{-1}$ to zero $m.s^{-1}$, thereby losing all of its kinetic energy. This energy is then converted into thermal energy through friction, significantly increasing the temperature. The heat flux can thus be described by Fourier's law: $q = -k\frac{\partial T}{\partial n}$. The location where the temperature gradient is the highest corresponds to the point where heat transfer is the highest.

OpenFOAM only allows users to compute the wall heat flux directly. However, it is possible to extract the necessary data as `.csv` files and use the following formula to compute the Stanton number :

$$St \approx \frac{q}{\rho_\infty H_0 U_\infty} \tag{3.1}$$

with : 2.1

- q : Heat flux

- $\rho_\infty$ : Freestream density
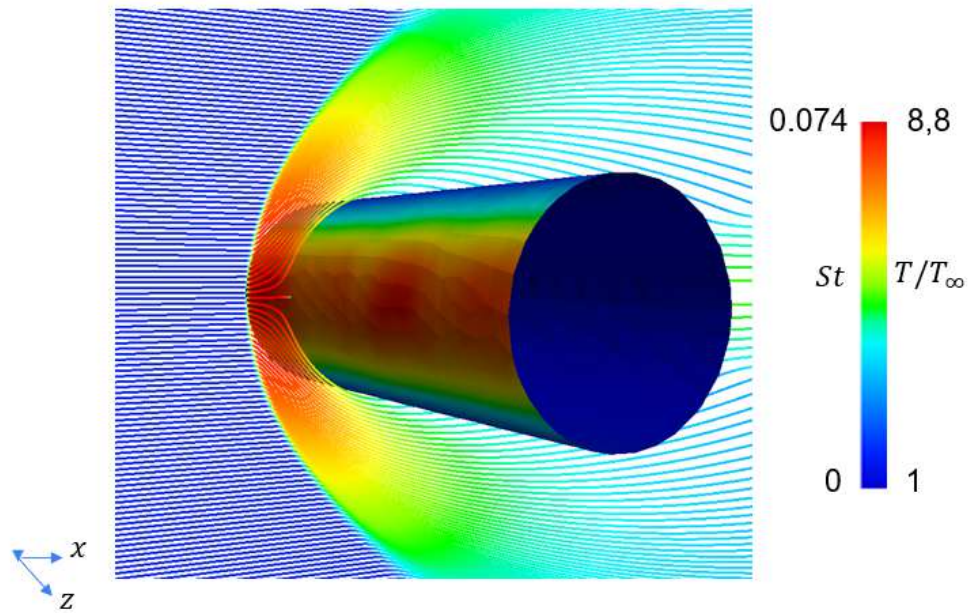
- $H_0$ : Free enthalpy

Thus, we obtain :

Figure 3.1.12: Streamlines represent the temperature profile and the cylinder shows the Stanton number

First, according to Fourier's law, the results appear consistent, as the highest values are located at the stagnation point. However, a discontinuity can be observed along the $z$-axis in the Stanton number distribution, which may be attributed to the poor mesh quality.

To conclude, although this mesh was of poor quality, it still yielded acceptable results. However, it led to a significant loss of computational resources and time due to unnecessary cells along the $z$-axis. As a result, this model was discarded.

**Structured mesh**

For the structured mesh, in order to improve precision in the near-wall region and to assess whether there are significant differences, we performed two simulations: one using the $k$–$\epsilon$ model and another using the $k$–$\omega$ SST model (see Section 2.4).
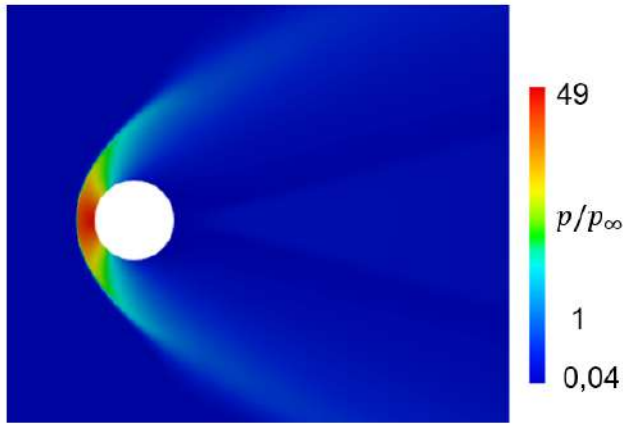
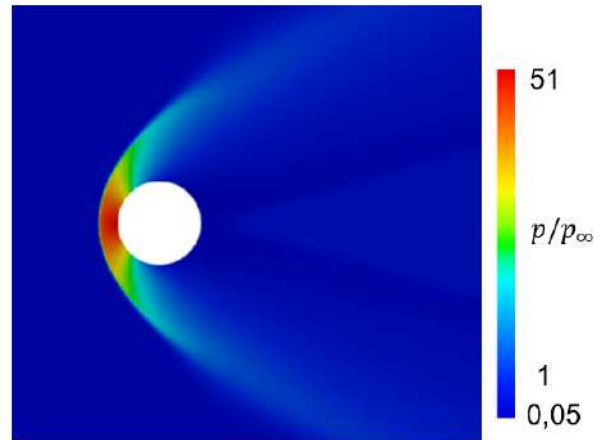Figure 3.1.13: Pressure with the $k$–$\omega$ SST model



Figure 3.1.14: Pressure with the $k$–$\epsilon$ model

Firstly, we observe high precision in the near-wall region due to thinner cells in both simulations. A closer examination reveals that both models produce similar results in terms of range. However, when compared to the theoretical pressure values (see Equation 2.9), the $k$-$\omega$ SST model proves to be more accurate, making it the better choice for this case.

This conclusion can also be drawn from the temperature distribution, as shown below. 3.1.16 3.1.15
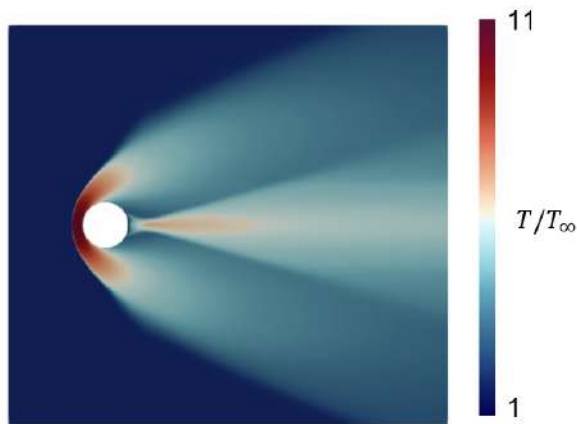


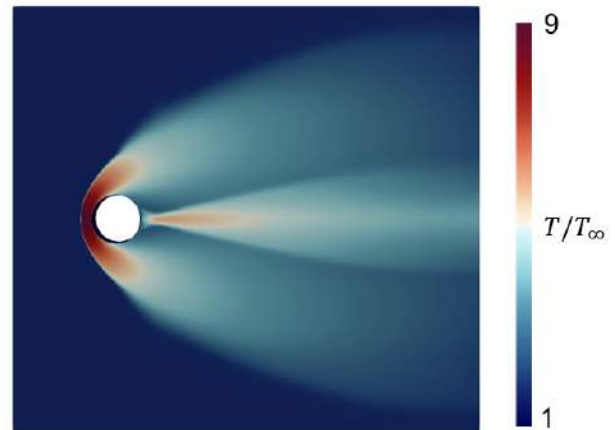Figure 3.1.15: Temperature field using the $k$–$\epsilon$ model



Figure 3.1.16: Temperature field using the $k$–$\omega$ SST model

The $k$–$\epsilon$ model shows a 21% error, which means that the results cannot be considered reliable due to the excessively high margin of error.

### 3.1.2.4    Stanton number

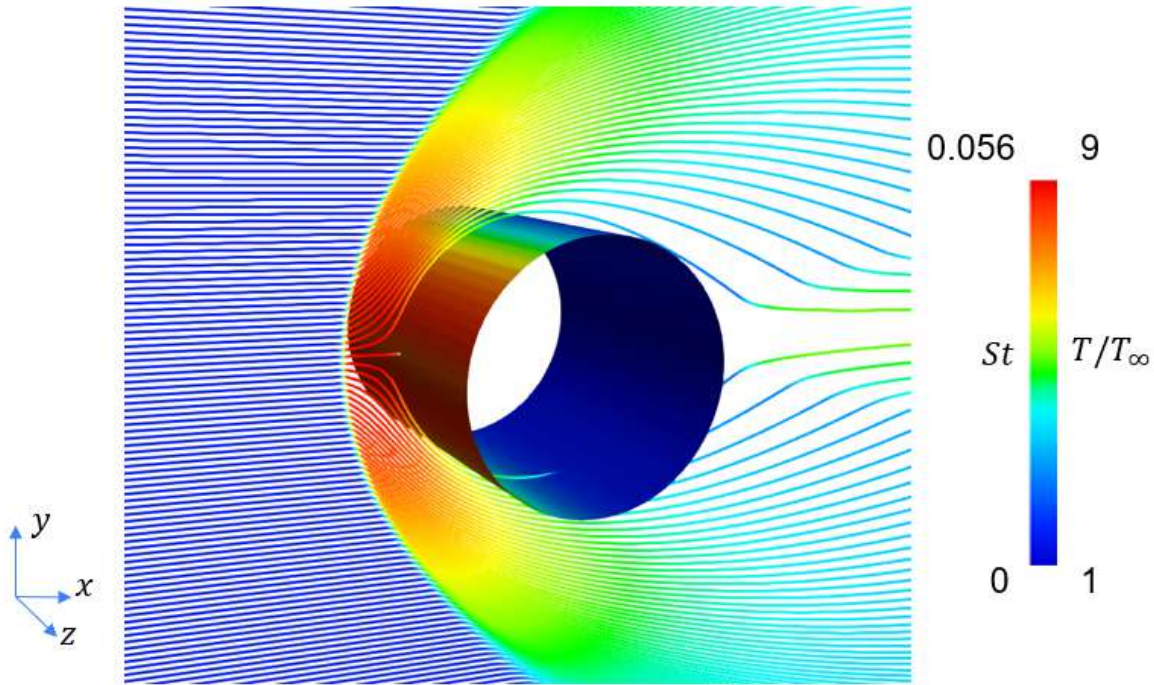After exporting data and calculate it, we obtain this graph :

Figure 3.1.17: Streamlines represent the temperature profile and the cylinder shows the Stanton number

The Stanton number behaves as expected according to the graph: it reaches its maximum at the stagnation point and drops to zero at the rear of the cylinder due to flow separation. However, when compared with Dr. Tanno's work, specially Figure 2.10.4—we observe that the computed Stanton number does not fall within the range of experimental values. There is a 506% error between both numerical and experimental values, indicating that the cylinder model might not be the best option.

In summary, the unstructured mesh led to unnecessary computational costs and irregularities in the results, particularly along the $z$-axis, despite producing globally acceptable values. On the other hand, the structured mesh provided highly satisfactory results, especially in the near-wall region. However, the inaccuracy observed in the Stanton number may be attributed to the two-dimensional nature of the simulation. Indeed, as shown in three-dimensional simulations, the shock wave tends to be sharper, which decreases the local heat flux around the 3D model – in our case, a sphere. [13].

## 3.2 Sphere in 3D model

### 3.2.1 Computation domain

Before creating the geometry, it is useful to consider how to reduce computational cost by eliminating unnecessary regions. Instead of computing a full rectangular prism, we can design a geometry that concentrates cells where they are most needed. In the three-dimensional case, the flow is expected to form a Mach cone. Therefore, we chose a cylindrical shape for the internal mesh, which offers a more efficient meshing strategy without losing any information in the wake region.

After creating the mesh on ICEM CFD, we obtain :
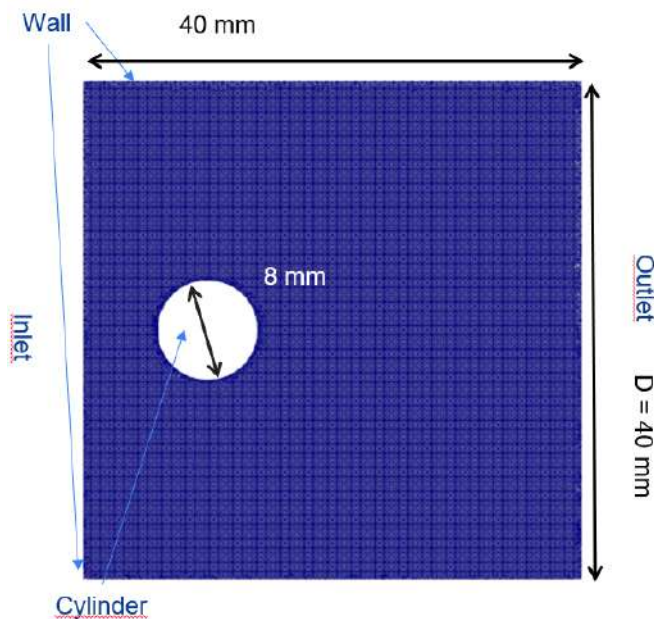


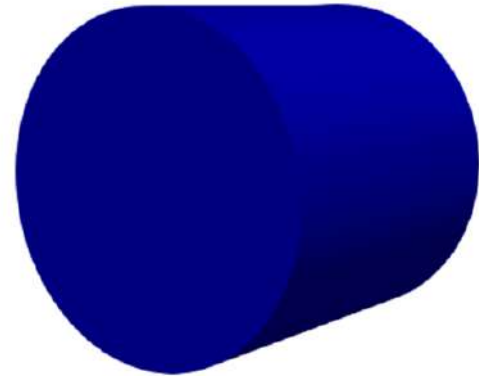Figure 3.2.1: Slice cut on Y-axis to see domain dimensions used for the meshing



Figure 3.2.2: Internal cylindrical mesh shape

Which gives those informations

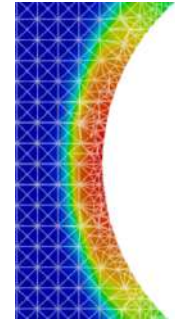| Property | Value |
|---|---|
| No. of cells | 10,955,448 |
| Sphere number of layers | 5 |
| Cylinder cell size | 2e-6 m |
| External growth ratio | 2 |

Table 3.4: Meshing information



Figure 3.2.3: Zoom on spherical mesh refinement

A mesh containing over 11 million cells implies a significant computational cost, but, at that time, such resolution is necessary to capture as much physical detail as possible. Moreover, we can see in Figure 3.2.3 that all the shockwave is covered by layer refinement. By now, we could think that this is enough to have information.

At the beginning of the computation, I received authorization to use 1,000 processor cores, which considerably accelerated the simulation. Nevertheless, the entire computation still required four days to complete.

### 3.2.2 Sphere simulation

The RANS model is in $k$-$\omega$ SST and the boundary condition for the wall is still **waveTransmissive**.
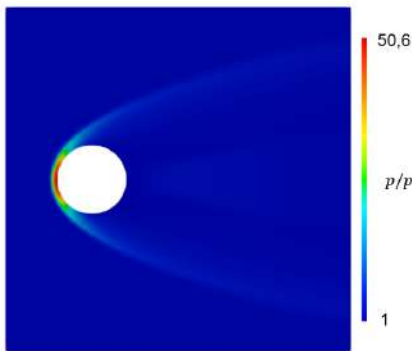


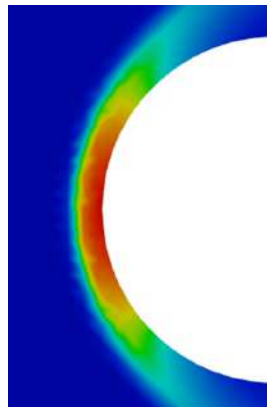Figure 3.2.4: Pressure field around the sphere



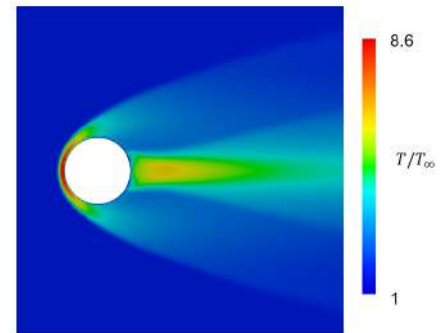Figure 3.2.5: Temperature field (zoomed view)



Figure 3.2.6: Temperature field around the sphere

Firstly, according to the Rankine–Hugoniot results 2.9, the pressure field shows an 8% error, indicating a noticeable difference. However, the temperature field aligns well with the theoretical

predictions, suggesting that the model may still be considered valid.

Secondly, in Figure 3.2.5, a discontinuity can be observed along the shockwave, which may be caused by the mesh quality. Since the mesh is composed of tetrahedral elements, such artifacts are to be expected. The main concern is that this discontinuity may also appear in the heat flux distribution, potentially affecting the accuracy of the results.

Finally, the cylindrical geometry appears to be a suitable computational domain, as it preserves the essential flow features without information loss. In particular, the shock wave adopts a bullet-like shape 3.2.7, which fits well within this domain. However, as we will see later, there are alternative domain shapes that may offer even better performance.
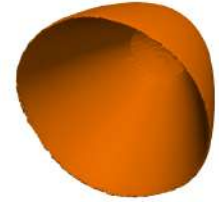


Figure 3.2.7: Shock wave formation

### 3.2.3   Stanton number

After exporting the heat flux into Excel, we obtain :
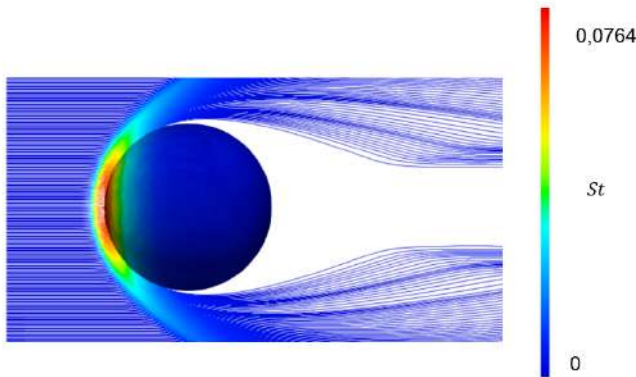


(a) Stanton number distribution (top view)
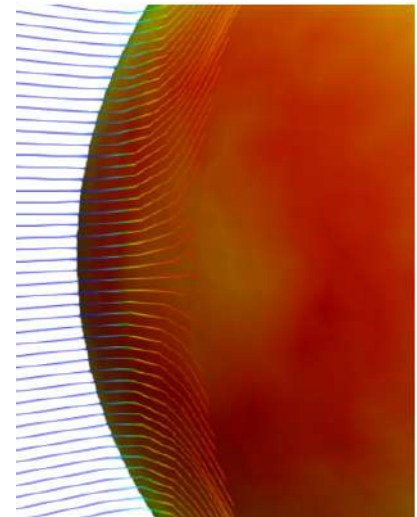


(b) Stanton number distribution (side view)

In Figure 3.2.8a, a noticeable change in the shock wave shape can be observed: the flow separation is more pronounced in the spherical model. However, the results remain poorly defined, with a discrepancy of up to 700%. Consequently, the model cannot be considered sufficiently relevant when compared to experimental results, as shown in Figure 2.10.4.

# Chapter 4

# Conclusion and Perspectives

The comparison between structured and unstructured meshes, as well as between cylindrical and spherical geometries, has revealed the critical importance of mesh quality and domain design in high-speed flow simulations.

Although some configurations—particularly the cylinder with a structured mesh, which provided the best results among all the simulations conducted—yielded acceptable outcomes, they are not sufficient to conclude this study. The 2D modeling of shock waves, combined with the significant discrepancies observed in the Stanton number between theoretical and numerical results, adds complexity to the analysis. Simulating a reentry vehicle using a sphere or a cylinder in a ballistic configuration remains challenging when relying solely on numerical outputs. However, the computational cost of such simulations is considerably lower than that of full-scale experimental campaigns on actual reentry capsules, making them a valuable and accessible alternative for preliminary analysis.

**Future Work and Article Perspective**

For the remainder of my internship, the workflow is already taking shape. Dr. Inokuma's experiment begins in less than two weeks, giving me a one-month window to obtain reliable simulation results for comparison. According to both Prof. Yakeno and Dr. Inokuma, it may take some time to optimize the mesh without compromising important flow features. For instance, Figure 3.2.4 suggests that the overall mesh quality is sufficiently fine; however, as shown in Figure 3.2.3, further refinement is still needed around the vehicle. This could constitute the next step in the computation process. Moreover, Figure 3.2.7 illustrates that the shock wave around a sphere adopts a bullet-like shape. Consequently, a bullet-shaped computational domain might be the most appropriate geometry for this study.

Another direction being considered for the spherical model is to **intentionally introduce roughness**. Until now, all simulations have been conducted on ideal, perfectly smooth bodies moving at Mach 6.3. However, Dr. Inokuma has suggested adding surface roughness to the sphere in order to increase turbulence and achieve more realistic results, since in practice, manufacturers cannot produce a perfectly smooth and symmetrical sphere.

To further improve realism, computations using **Large Eddy Simulation (LES)** turbulence models should be performed. While time-averaged models (such as RANS) are useful for saving

computational resources, resolving the eddies in the wake is essential for accurately assessing drag.

According to Prof. Yakeno, if the optimal geometry can be identified before the end of the internship, both the spherical results and a newly proposed shape—referred to as a **streamlined body**—may be used to develop a scientific publication.

Indeed, the drag behind a body can be expressed as:

$$C = C_p + C_f \tag{4.1}$$

where $C_p$ represents the **pressure drag** and $C_f$ the **friction drag**. Pressure drag primarily results from flow separation, which is significant around bluff bodies such as spheres, but almost nonexistent for streamlined shapes.

The idea is to **compare both geometries** and subtract their drag contributions in order to develop a method for estimating pressure drag in a ballistic range.

# Nomenclature

$\boldsymbol{\sigma}$      Viscous stress tensor

$\Delta t$      Time step

$\Delta x$      Characteristic size of cells

$\Gamma_\omega$      Diffusion rate of $\omega$

$\Gamma_k$      Diffusion rate of $k$

$\mu$      Dynamic viscosity

$\mu_t$      Turbulent viscosity

$\omega$      Specific dissipation rate

$\rho$      Fluid density

$\rho_\infty$      Freestream density

$\sigma_\varepsilon$      Turbulent Prandtl number for $\varepsilon$

$\sigma_k$      Turbulent Prandtl number for $k$

$\varepsilon$      Turbulent dissipation rate

$\vec{j}$      Diffusive heat flux vector

$\vec{U}$      Velocity vector

$C$      Total drag coefficient

$C_{1\varepsilon}$      Model constant

$C_{2\varepsilon}$      Model constant

$C_{3\varepsilon}$      Model constant

$C_\mu$      Model constant in $k - \varepsilon$ model

$C_f$      Friction drag coefficient

$C_p$      Pressure drag coefficient

$Co$      Courant number

$D_\omega$      Orthogonal divergence term

$E$        Total specific energy

$e$        Specific enthalpy

$G_b$        Generation of turbulence kinetic energy due to buoyancy

$G_k$        Generation of turbulence kinetic energy due to mean velocity gradients

$h$        Heat transfer coefficient

$H_0$        Total enthalpy

$k$        Turbulent kinetic energy

$Nu$        Nusselt number

$p$        Pressure

$P_\infty$        Freestream pressure

$Pr$        Prandtl number

$q$        Heat flux

$Re$        Reynolds number

$Re_\infty$        Reynolds number

$S_\varepsilon$        User-defined source term for $\varepsilon$

$S_k$        User-defined source term for $k$

$St$        Stanton number

$T_\infty$        Freestream temperature

$T_\text{wall}$        Wall temperature

$U_\infty$        Freestream velocity

$Y_\omega$        Turbulence generated due to diffusion for $\omega$

$Y_M$        Contribution of fluctuating dilatation in compressible turbulence

# Abstract

This internship was carried out at the Institute of Fluid Science (IFS) at Tohoku University. It focused on the numerical modeling of hypersonic flow around re-entry vehicles, specifically the Apollo capsule, in order to predict heat fluxes. The goal was to support experimental research in collaboration with JAXA by developing reliable CFD simulations using OpenFOAM and other CFD software such as ANSYS. Simplified geometries, in 2D (cylinder) and 3D (sphere), were

studied to first create a simplified model of the Apollo capsule and eventually enable less expensive but still realistic experiments. These simulations were performed using compressible solvers and turbulence models (RANS, SST k-$\omega$).

Special attention was given to mesh quality, boundary conditions, and model validation through the Rankine–Hugoniot relations and experimental data from Tanno et al. Despite a higher computational cost, the structured mesh around the cylinder provided the most consistent results. The 3D simulation, on the other hand, showed discontinuities due to the mesh and some deviations in heat flux prediction. This work highlights the trade-off between accuracy and computational cost and sets the stage for future simulations using LES models and optimized geometries.

**Keywords:** Hypersonic flow, CFD simulation, Heat flux prediction, OpenFOAM, Re-entry vehicle, Turbulence, Shockwave.

## Résumé

Ce stage a été réalisé à l'Institute of Fluid Science (IFS) de l'université de Tohoku. Il porte sur la modélisation numérique de l'écoulement hypersonique autour de véhicules de rentrée atmosphérique, et plus précisément de la capsule du programme Apollo, afin de prédire les flux thermiques. L'objectif était de soutenir les recherches expérimentales en collaboration avec la JAXA en développant des simulations CFD fiables à l'aide d'OpenFOAM et de logiciels de CFD tels que ANSYS. Des géométries simplifiées, en 2D (cylindre) et en 3D (sphère), ont été étudiées afin de, premièrement, réaliser un modèle simplifié de la capsule Apollo puis finalement de pouvoir réaliser des expériences moins coûteuses mais tout autant fidèles à la réalité. Ces simulations ont été réalisées avec des solveurs compressibles et des modèles de turbulence (RANS, SST k-$\omega$).

Une attention particulière a été portée à la qualité du maillage, aux conditions aux limites et à la validation des modèles via les relations de Rankine–Hugoniot et les données expérimentales de Tanno et al. Malgré un coût de calcul plus élevé, le maillage structuré autour du cylindre a fourni les résultats les plus cohérents. La simulation 3D, quant à elle, a révélé des discontinuités liées au maillage et des écarts dans la prédiction du flux thermique. Ce travail met en évidence le compromis entre précision et ressources, et prépare le terrain pour des simulations futures utilisant les modèles LES et des géométries optimisées.

**Mots-clés :** Écoulement hypersonique, Simulation CFD, Prédiction de flux thermique, OpenFOAM, Véhicule de rentrée spatiale, Turbulence, Onde de choc.

# Bibliography

[1] Dendy Adanta, I. M. Rizwanul Fattah, and Nura Muaz Muhammad. *Comparison of Standard k-ε and SST k-ω Turbulence Models for Breastshot Waterwheel Simulation.* 2020.

[2] Dendy Adanta, I. M. Rizwanul Fattah, and Nura Musa Muhammad. "COMPARISON OF STANDARD k-epsilon AND SST k-omega TURBULENCE MODEL FOR BREASTSHOT WATERWHEEL SIMULATION". In: *Journal of Mechanical Science and Engineering* 7.2 (Oct. 2020), pp. 039–044. DOI: 10.36706/jmse.v7i2.44. URL: https://jmse.ejournal.unsri.ac.id/index.php/jmse/article/view/44.

[3] Canteros, Maria Laura and Polanský, Jiří. "Review and comparison of two OpenFOAM® solvers: rhoCentralFoam and sonicFoam". In: *EPJ Web of Conf.* 299 (2024), p. 01005. DOI: 10.1051/epjconf/202429901005. URL: https://doi.org/10.1051/epjconf/202429901005.

[4] Japan Aerospace Exploration Agency (JAXA). *About the High Enthalpy Shock Tunnel "HI-EST".* https://www.jaxa.jp/about/centers/kspc/pdf/kspc_facilities_03.pdf. Source in Japanese. n.d.

[5] D.B. Lee, J.J. Bertin, and W.D. Goodrich. *Heat-transfer rate and pressure measurements obtained during Apollo orbital entries.* Tech. rep. TN D-6028. NASA, 1970.

[6] NASA. *Orion Reference Guide.* Tech. rep. Version 111022. NASA, Feb. 2023. URL: https://www.nasa.gov/wp-content/uploads/2023/02/orion-reference-guide-111022.pdf.

[7] F. Kevin Owen and Andrew K. Owen. "Measurement and assessment of wind tunnel flow quality". In: *Progress in Aerospace Sciences* 44.5 (2008), pp. 315–348. ISSN: 0376-0421. DOI: https://doi.org/10.1016/j.paerosci.2008.04.002. URL: https://www.sciencedirect.com/science/article/pii/S0376042108000286.

[8] Akihiro Sasoh. *Compressible Fluid Dynamics and Shock Waves.* See page 69. Singapore: Springer, 2020. ISBN: 9789811505034.

[9] M. Shaaban, M. El Morsy, and M. Ramadan. *LES simulation of turbulent flow over wavy surface using OpenFOAM.* 2020. URL: https://www.akademiabaru.com/submit/index.php/cfdl/article/view/2267/1249.

[10] Tsan-Hsing Shih et al. *A new k- eddy viscosity model for high Reynolds number turbulent flows–Model development and validation.* 1995. URL: https://api.istex.fr/ark:/67375/6H6-5KTMCLC2-P/fulltext.pdf?sid=clickandread.

[11] Junjie Sun et al. "Aerodynamic Thermal Simulation and Heat Flux Distribution Study of Mechanical Expansion Reentry Vehicle". In: *Aerospace* 10.3 (2023). ISSN: 2226-4310. DOI: 10.3390/aerospace10030310. URL: https://www.mdpi.com/2226-4310/10/3/310.

[12] Hideyuki Tanno et al. "Aeroheating measurement of Apollo shaped capsule with boundary layer trip in the free-piston shock tunnel HIEST". In: *52nd Aerospace Sciences Meeting.* DOI: 10.2514/6.2014-0434. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2014-0434. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2014-0434.

[13]   Tayfun E. Tezduyar, Masayoshi Senga, and Darby Vicker. "Computation of Inviscid Supersonic Flows Around Cylinders and Spheres with the SUPG Formulation and YZ Shock-Capturing". In: *Computational Mechanics* 38.4 (2006), pp. 469–481. ISSN: 1432-0924. DOI: 10.1007/s00466-005-0025-6. URL: https://doi.org/10.1007/s00466-005-0025-6.

[14]   Institute of Fluid Science Tohoku University. *Institute of Fluid Science*. Official website. 2025. URL: https://www.ifs.tohoku.ac.jp/eng/.

[15]    And. "HIEST(lt;gt; 2: 21)". In: 61.9 (2013), pp. 309–315. DOI: 10.14822/kjsass.61.9_309.
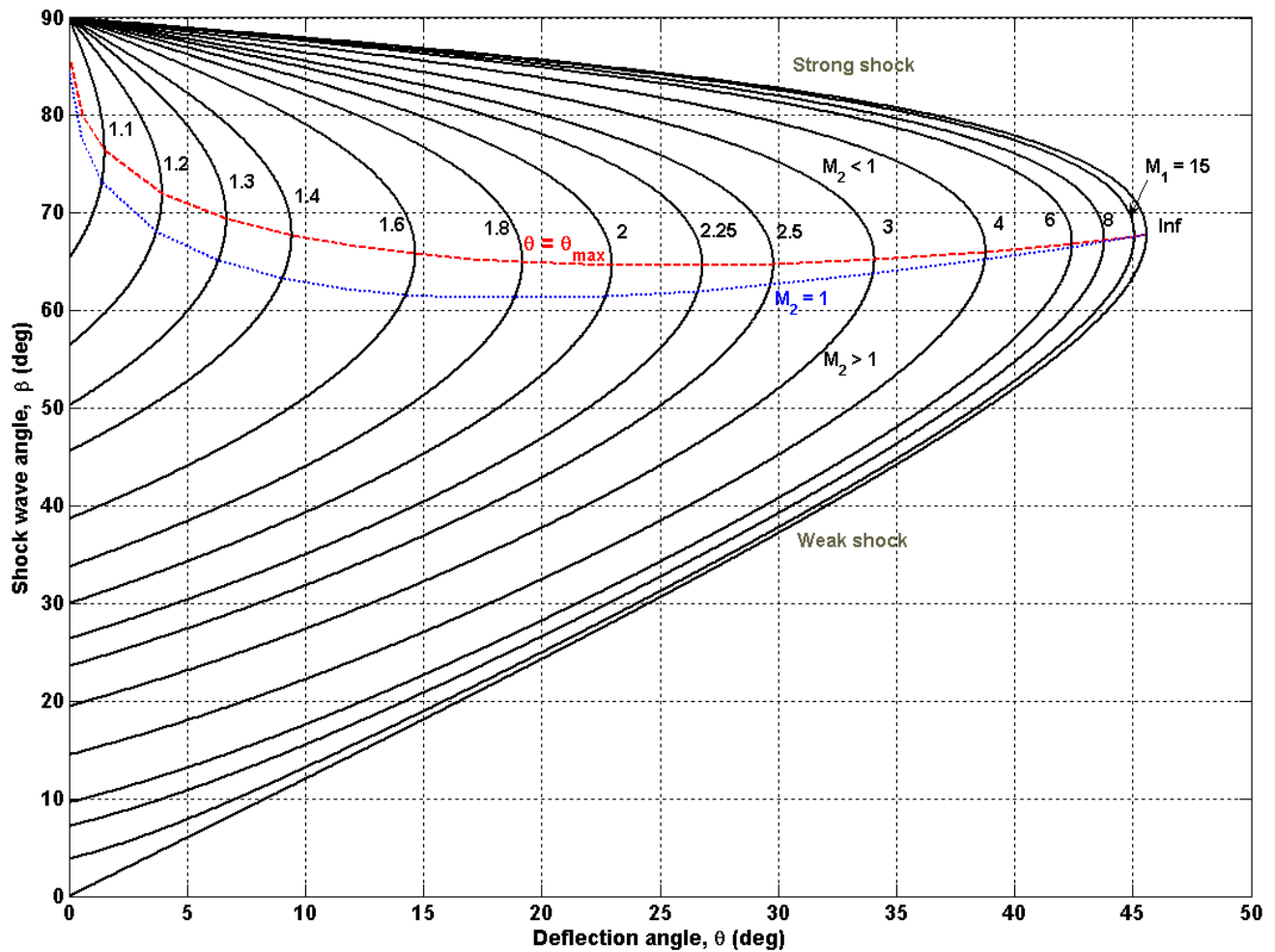
# Appendices

# .1  Shockwave angle measurement



Figure .1.1: Shock wave angle abacus