



Escola Superior d'Enginyeries Industrials,
Aeroespacial i Audiovisual de Terrassa

UNIVERSITAT POLITÈCNICA DE CATALUNYA

Simulation of the first stages of a turbofan using OpenFoam

Degree: Aerospace Engineering

Course: Application of Open-Source CFD to Engineering Problems

Delivery date: 09-12-2016

Students:

Herrán Albelda, Fernando

Martínez Viol, Víctor

Morata Carranza, David

Contents

1	Introduction	2
2	Case	3
2.1	Description	3
2.2	Hypotheses	4
3	Mesh generation	5
3.1	blockMesh	5
3.2	Mesh refinement	6
3.2.1	Coarse mesh	7
3.2.2	Standard mesh	7
3.2.3	Dense mesh	7
4	The problem	12
5	Simulation	13

1 Introduction

During the development of the subject of *"Application of Open-Source CFD to engineering problems"* we have obtained knowledges in relation to computational fluid dynamics (CFD), open source and open foam. Initially was presented a list of different possible projects to perform as a group during the course and its subsequent delivery. After agreeing with the teacher, it was decided to choose a project outside this list and related to propulsive systems, specifically a jet engine.

We have found interesting and useful study this engine instead of other aerospace projects such as an airfoil because we are interested in propulsion systems and this project could give extra knowledges on this matter in view of a future TFG related to this subject. Furthermore, this engine is one of the most important in the aerospace industry and it is used in planes such as Airbus A380 or Boeing 747.

2 Case

2.1 Description

The main case of project is based on the analysis of a flow in a turbofan. A turbofan consists in its initial stage of a fan where air enters a high velocity. This fan is a larger compressor than others and split the flow in two, main flow and secondary flow. The secondary flow advances through a conduit but the main flow enters the engine core where it goes through the compressor stage which is divided in two substages, one of low compression and one of high pressure compressor, where pressure increases. After compression stage, flow enters inside combustion chamber where it is mixed with the fuel and it is burned. Gases obtained pass through the turbines and arrive at the nozzle, where the air is accelerated and the thrust is obtained.

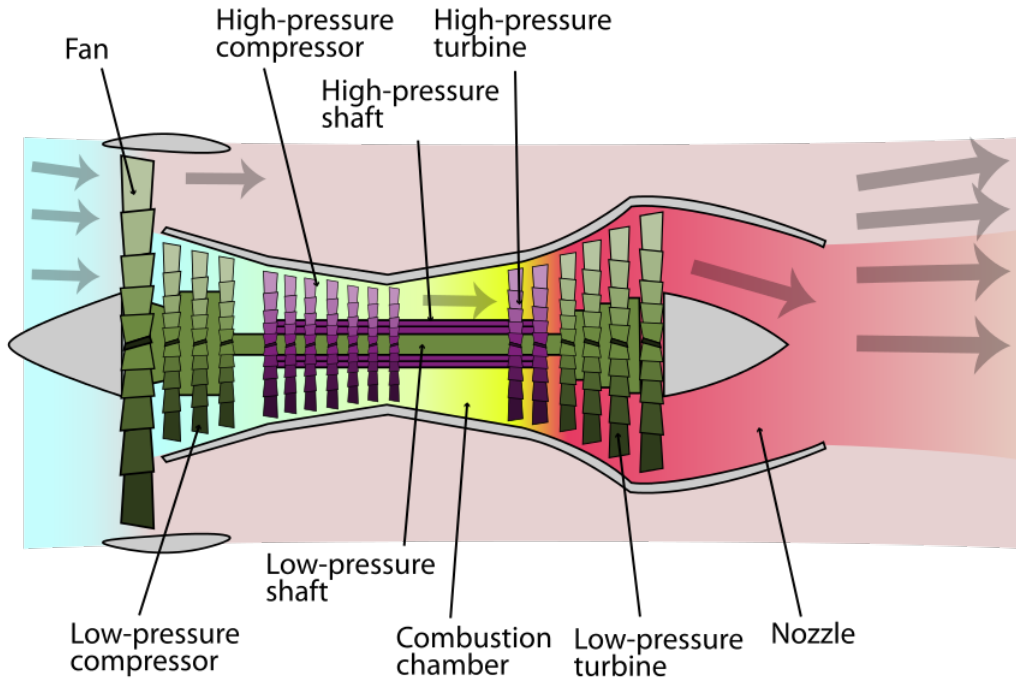


Figure 2.1: First mesh. Toooooo coarse

The study is centered in the behaviour of the flow in the stage of compression, considering a hypotheses and an initial conditions that will

be explained below.

The different files of engine to can realise the study, the nacelle, the fan, the combustion chamber, the different stages of compressor and turbine and the nozzle, have been downloaded from the website GrabCad, specifically the files of a "*High Bypass Turbofan Jet Engine*".

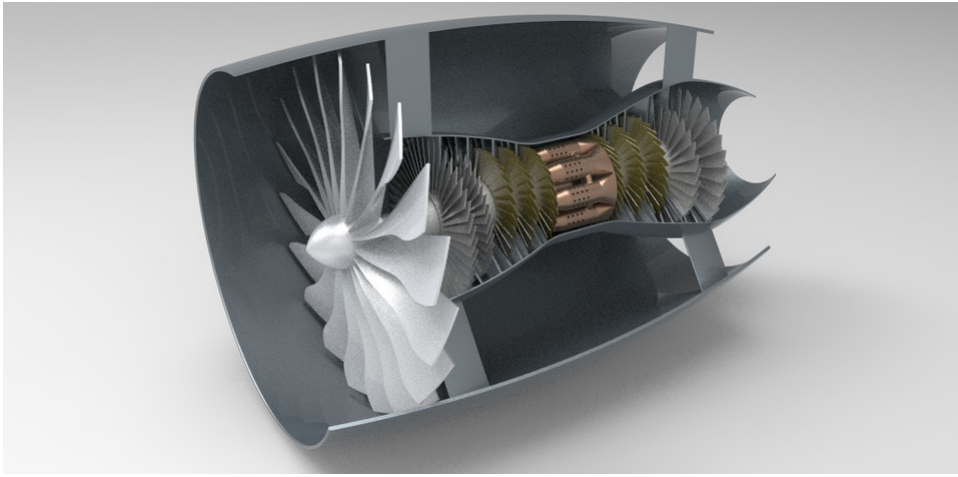


Figure 2.2: First mesh. Toooooo coarse

2.2 Hypotheses

- Tridimensional flow
- Viscous flow
- Compressible flow
- Newtonian flow
- Stationary flow

3 Mesh generation

3.1 blockMesh

The definition of the *blockMeshDict* is the first part that needs to be defined in order to simulate the case. Only the first two stages of the turbofan engine will be simulated, so we have to make sure that the *blockMesh* includes them.

The domain of the mesh is a $0.58 \times 2.3 \times 2.3m$ rectangular prism and the vertices are as follows:

```
vertices
(
(0.77 0 0)
(1.35 0 0)
(1.35 2.3 0)
(0.77 2.3 0)
(0.77 0 2.3)
(1.35 0 2.3)
(1.35 2.3 2.3)
(0.77 2.3 2.3)
)
```

Once the boundaries of the *blockMesh* are defined, the number of cells that it will have has to be set. It has to be taken into account that a very dense mesh will not be efficient when simulating the case. On the other hand, a very coarse mesh will not be efficient either because additional divisions will have to be set when generating the refined mesh with *snappyHexMesh*. Thus, a compromise solution between a mesh with a very high number of cells and a very low number of cells has to be attained.

This basic mesh has been divided every $0.05m$. It means that we have done 12 divisions in the x direction, 46 on the y direction and 46 more on the z direction. Obviously, the vertex numeration has been kept the same as the one that comes as default in every tutorial case; so, it was easier to

define the inlet face of the *blockMesh* as well as the outlet face that will be used to define the boundary conditions.

The log obtained when the mesh has been generated is as follows:

```
Writing polyMesh
-----
Mesh Information
-----
boundingBox:  (0.77 0 0) (1.35 2.3 2.3)
nPoints:  28717
nCells:   25392
nFaces:   79396
nInternalFaces: 72956
-----
Patches
-----
patch 0 (start: 72956 size: 2208) name: frontAndBack
patch 1 (start: 75164 size: 2116) name: inlet
patch 2 (start: 77280 size: 2116) name: outlet

End
```

3.2 Mesh refinement

To refine the mesh, the *snappyHexMesh* utility, included in OpenFoam, which is used and several parameters have to be modified in order to obtain a dense mesh that is suitable for the simulation of this complex geometry. Additionally, it has to be considered that a particular geometry is has a relative velocity; this is, the rotor is rotating, and so the first and second stages of the Low Pressure Compressor of the turbofan engine, while the nacelle and the combustor are static.

FALTA EXPLICAR SURFACE EXTRACT

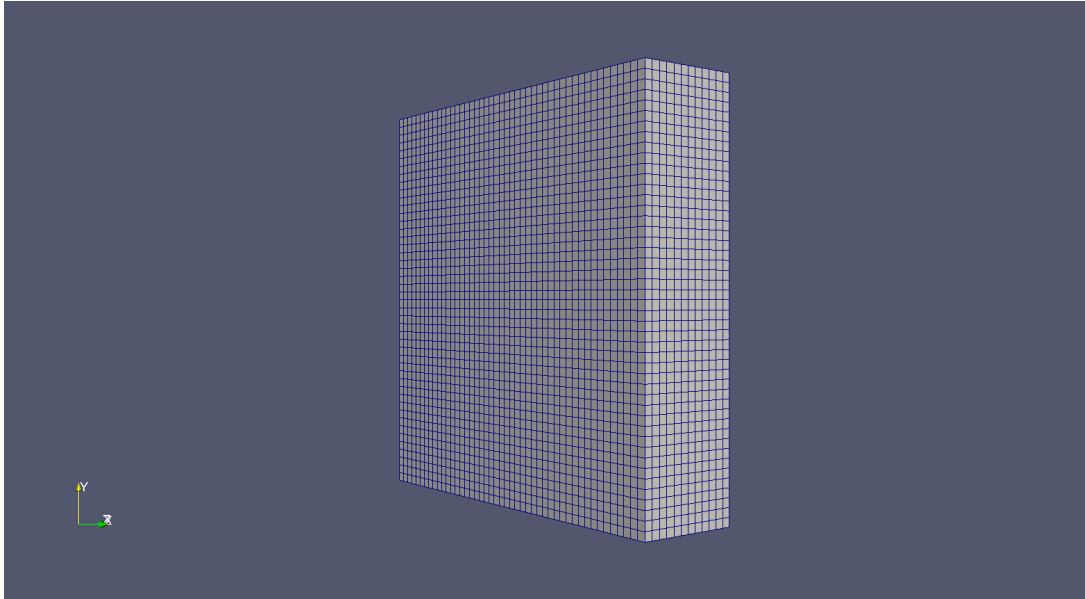


Figure 3.1: blockMesh

A comparison between the coarse mesh and the refined mesh generated is presented below.

3.2.1 Coarse mesh

The number of control volumes in the mesh is 40.

3.2.2 Standard mesh

3.2.3 Dense mesh

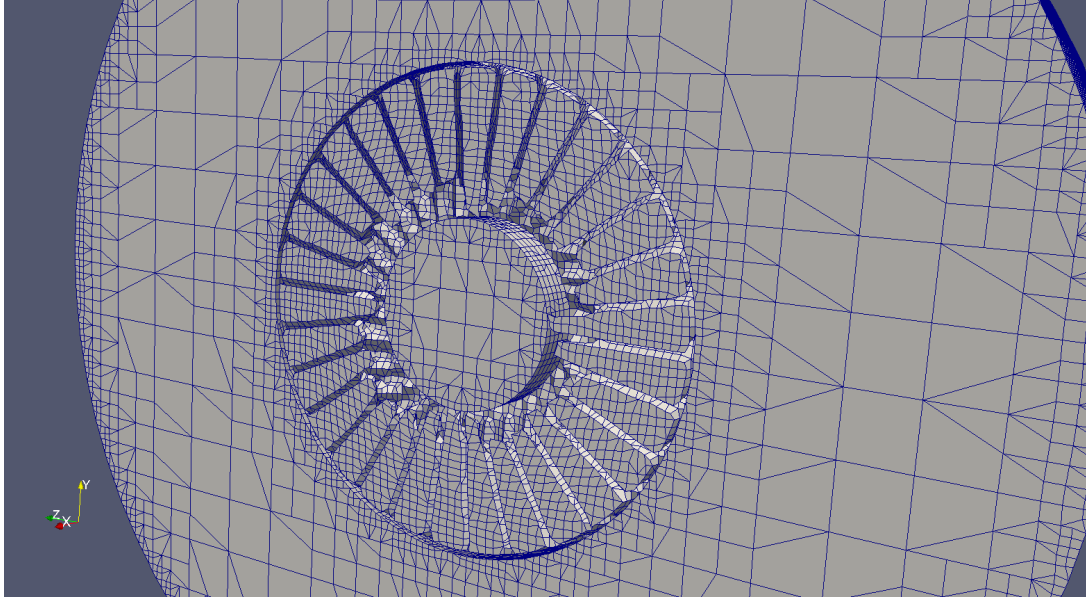


Figure 3.2: Detail of the coarse mesh

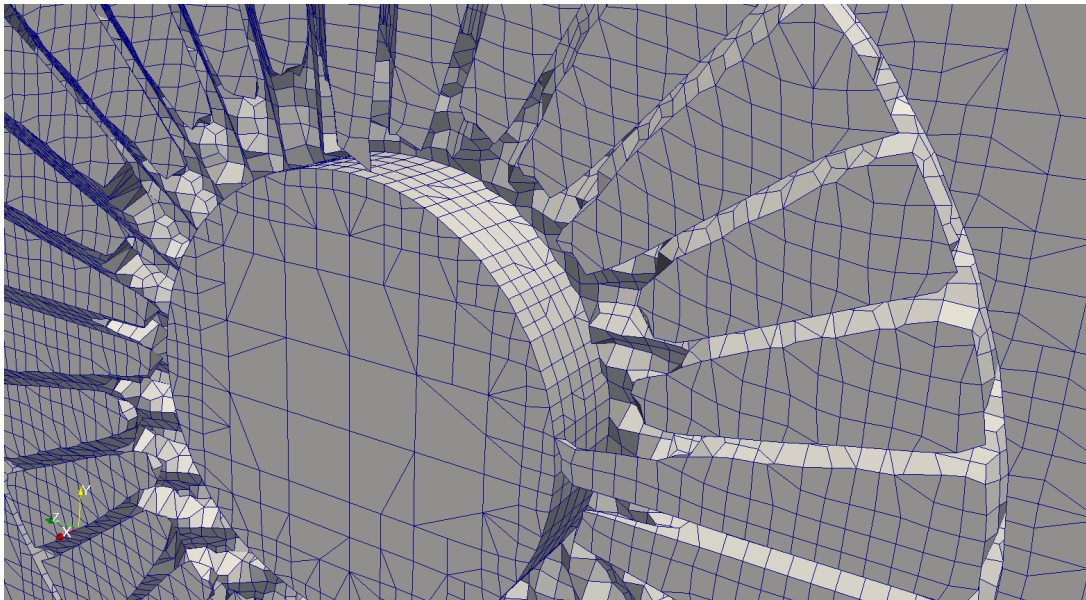


Figure 3.3: Detail of the coarse mesh

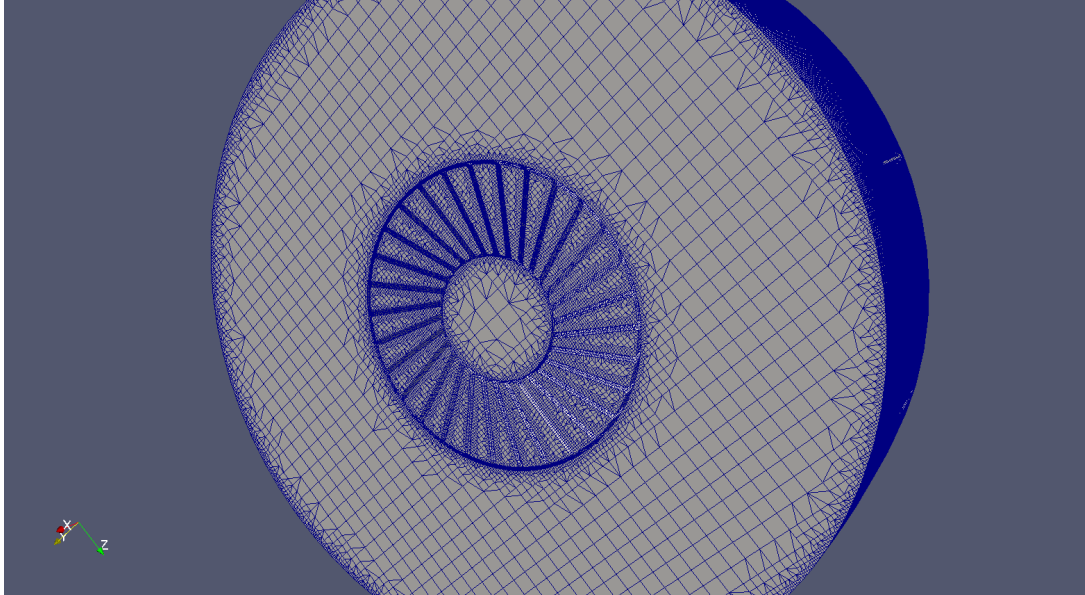


Figure 3.4: Detail of the standard mesh

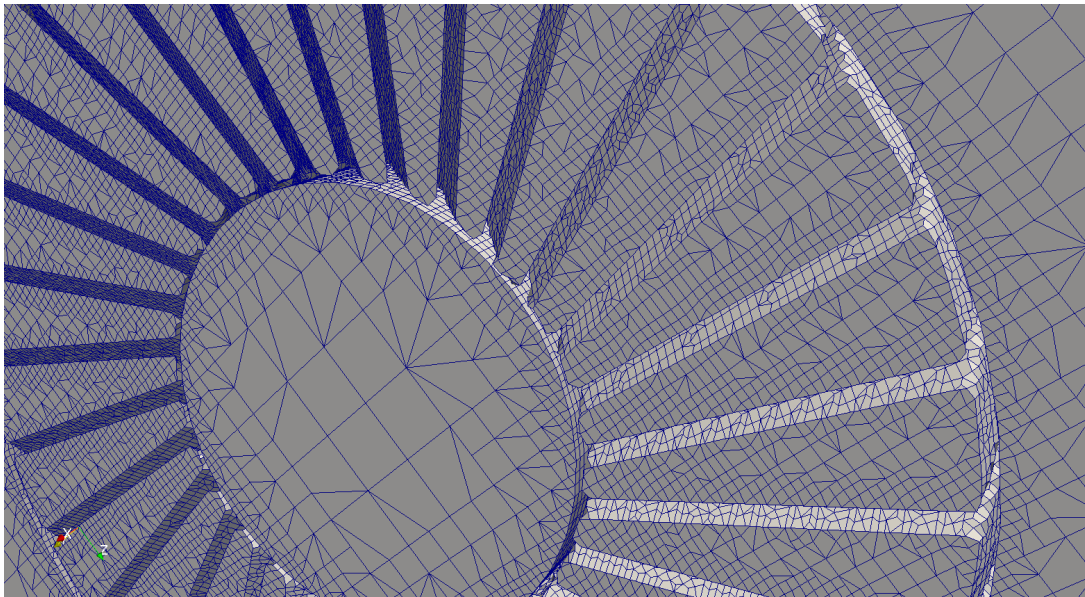


Figure 3.5: Detail of the standard mesh

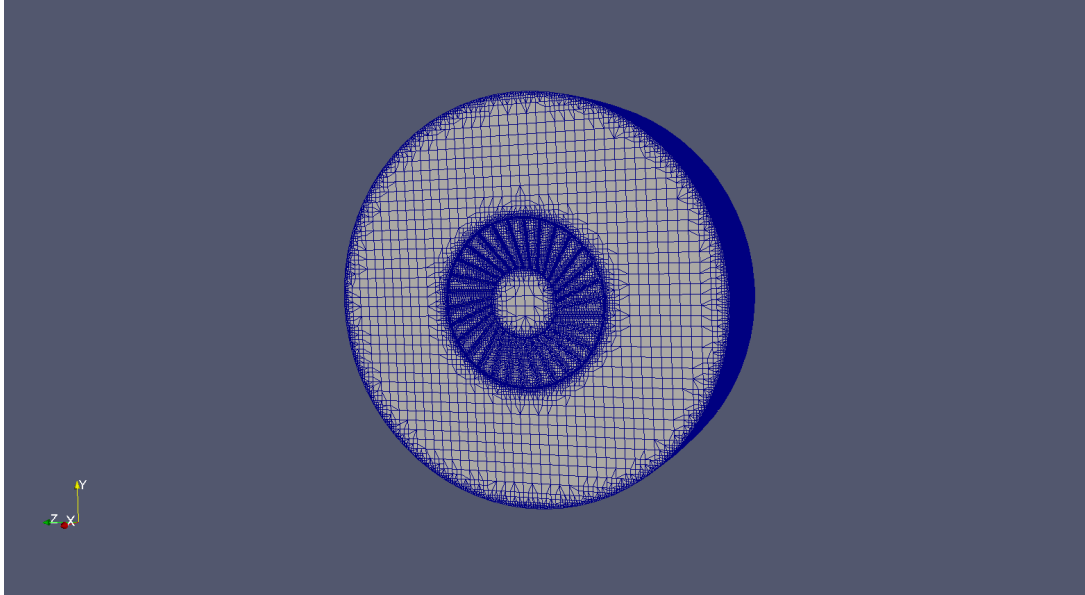


Figure 3.6: Detail of the coarse mesh

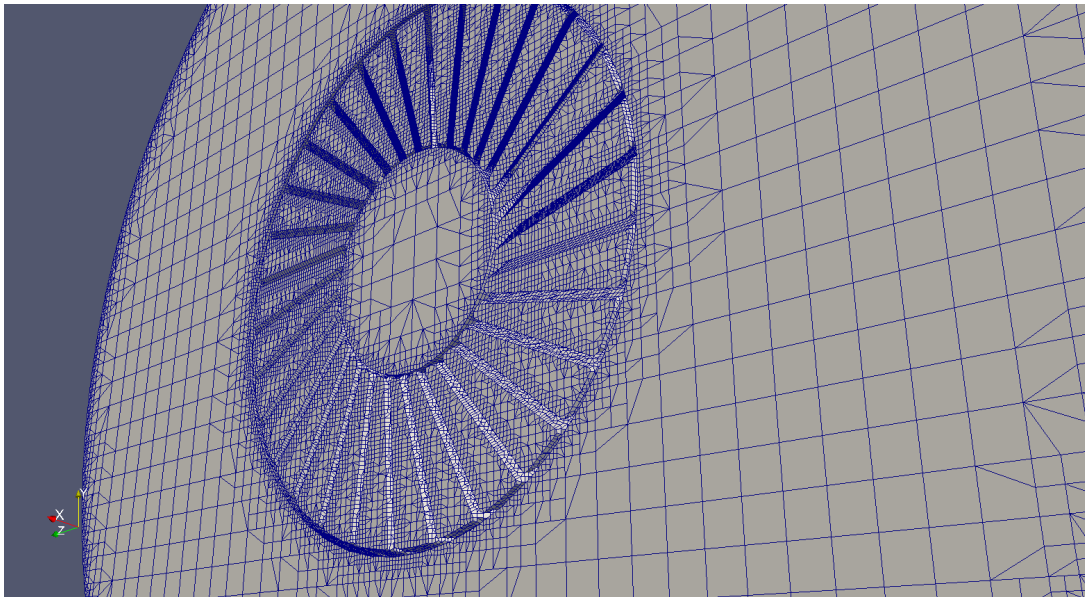


Figure 3.7: Detail of the dense mesh

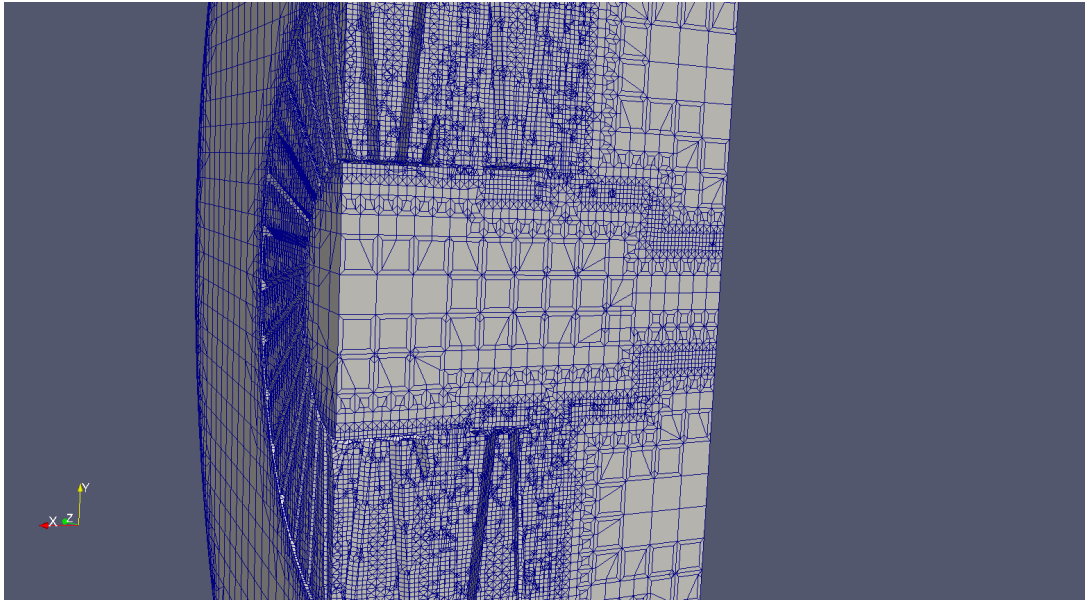


Figure 3.8: Detail of the dense mesh

4 The problem

EMPTY

5 Simulation

EMPTY