Università degli Studi di Padova

Department of Industrial Engineering DII

Master of Science in Aerospace Engineering

Hypersonic air intake design using multi-objective optimisation based on CFD data

Thesis advisors:

Eng. **Francesco De Vanna**, PhD
Prof. **Ernesto Benini**

Candidate:

Pietro Cecchetto

Academic Year 2021/2022

*La vertigine non è paura di cadere,
ma voglia di volare*
[*Lorenzo Cherubini*]

Ai miei genitori e a mia sorella

Contents

Introduction

In aeronautics, there is a need to reduce consumption and weight in order to save money. Thus, all components are optimised to achieve maximum efficiency at a moderate cost. Optimising the geometry of an aircraft, such as the wings or fuselage, is one of the design objectives. In particular, there is a focus on the geometry of the entire engine body of an aircraft to increase the overall efficiency of the aircraft. There are different types of engines, but almost all of them can be divided into different parts: intake, compression stage, combustion chamber, expansion stage and nozzle. The intake is the first part of the engine. This part is built with the purpose of capturing the air in the best possible way, in terms of speed and angle, to be sent to the internal part of the engine, the compressor. An intake must:

- be light weight and low cost;

- be less invasive as possible regarding the overall vehicle drag;

- compress the air before the compressor with consequent diffusion and slowdown of the flow;

- provide the engine with the adequate mass flow rate, Mach number and reduced distortion;

- have oblique compression trains to reduce energy losses.

Different types of intakes are used nowadays for different purposes. There are extensive studies of subsonic and transonic intakes, because they are more widely used in commercial and military aeronautics. Relatively new is the field of very high speed flow called hypersonic flow. This is typical of rockets, rarely of military aircraft and similar applications.

Only a few geometries are accepted for high flow because there are several phenomena such as shock wake, boundary layer interaction and chemical interaction (these will be discussed in later chapters). The shape of hypersonic inlets always has ramps to generate oblique and normal shocks to have less pressure loss. Ramps are crucial in the design of the inlet. The latter also has a cowl lip, the upper part of the body. This part is important for the resistance generated by the whole intake. It also diverts the flow, and thus defines the mass flow, in and out of the engine.

In this thesys, a basic geometry, known from experimental and computational studies, is combined with an optimisation algorithm for the section of the cowl lip and for the ramps in order to change shape and obtain a better resistance value, pressure ratio or total pressure ratio.

Chapter 1

General concepts of fluid dynamics

1.1 Introduction

Fluid dynamics is an branch of continuum mechanics that studies the behaviour of a fluid, a state of the matter which is continuously deformed when subjected to tangential forces of values even arbitrarily small, typical of gas and liquids.

This chapter aims to define the laws and the equations governing the fluid dynamics. These are fundamental to fully describe the future implications written in this thesys.

1.2 Continuum mechanics

Fluid mechanics can be described with two different reference systems: Lagrangian and Eulerian. In particular, the two views can be described as follows:

- **Lagrangian:** describes a zone, called *material volume $V(t)$*, moving with the fluid such that the same fluid particles are always in the same volume. Its shape changes in time. The frame is integral with the particle;

- **Eulerian:** describes the values of the variables in the various points in the time and in the space. The variation of the variables is described in zone, called *control volume V_0*, fixed in space. The frame is considered external to all the particles.
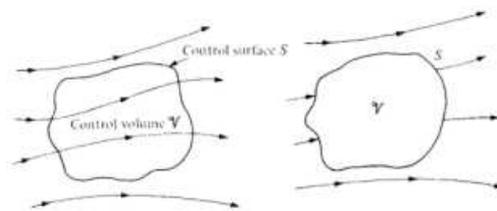
Figure 1.1: Eulerian and Lagrangian view of the same fluid zone

For better understanding, two example can be made: taking into account the pressure of a fluid, it's a field so it's a scalar quantity changing from point to point and in time. It is described in an Eulerian view. Differently, considering the velocity, it's a vectorial quantity depending from the point and the time considered. It considers a particle passing

a point in that moment. It is described in a Lagrangian view.

Before go any further, it is necessary to define some basic relations. Consider $b = b(x, y, z, t)$ a generic scalar or vectorial field, the material derivative respect to the time is so defined:

$$\frac{Db}{Dt} = \frac{\partial b}{\partial t} + \sum_{i=1}^{3} \frac{\partial b}{\partial x_i} \frac{dx_i}{dt} = \frac{\partial b}{\partial t} + \nabla b \cdot \mathbf{v} \tag{1.1}$$

here $\mathbf{v}$ indicates the field of velocity and the second part of the equation is given by the definition of gradient. So, the material or total derivative is defined by sum of the partial derivative respect to time and convective derivative given by the gradient. In another way, the differential operator which calculates in an Eulerian way the derivative in the time of any Lagrangian quantity is called material derived.

Now, for the future use, it necessary to give the definition of extensive and intensive variable. The first is a quantity in a macroscopic system that is proportional to the size of the system (e.g. the mass), the second is a quantity not proportional to the size of the system (e.g pressure). In particular, there is a relation between these two variables:

$$B = \int_V \rho b \, dV \tag{1.2}$$

here B is the extensive variable, b is the corresponding intensive variable and ρ is the density of the system.

Another theorem that will be utilized is the divergence theorem. It is defined as follows:

$$\int_{\partial V} G \cdot \hat{n} \, dS = \int_V \nabla \cdot G \, dV \tag{1.3}$$

here G is a variable at least order one, S is the external surface, ∂V is the boundary of the volume or system and $\hat{n}$ is the normal of the boundary of the volume.

Now, it's time to define the Reynold Transport Theorem (RTT). Given an extensive B and intensive variable b, $V(t)$ as a material volume and a control volume V_0, then a material derivative can be so expressed:

$$\frac{dB}{dt} = \frac{d}{dt} \int_{V(t)} \rho b \, dV(t) = \frac{d}{dt} \int_{V_0} \rho b \, dV_0 + \int_{\partial V_0} \rho b \mathbf{v} \cdot \hat{n} \, dS = \int_{V_0} \left(\frac{d(\rho b)}{dt} + \rho b \nabla \cdot \mathbf{v} \right) dV_0$$

$$\tag{1.4}$$

Here, the core of the RRT is the equality between the second and the third part of the equation. The theorem defines that a variation of an extensive variable in time and space - material derivative - is the sum of a variation of the same variable in the time and of the flux of the same variable through the control volume. The last two parts describes the theorem in a different way using the divergence theorem (equation (1.3)) and combining in a compact way.

1.3 Basic flow equations

Basic flow equations contains the fundamental physics of flows. They are noted as: *mass conservation, balance of the momentum quantity* and *conservation of energy principle*. They are general and so they are applied to all the systems. These equations combined with a constitutive fluid equation resolve the flow of a fluid. They can be expressed in an integral or differential way.

1.3.1 Mass conservation

Starting with the mass conservation equation, given a mass M as extensive variable and so 1 as its intensive value:

$$\frac{d}{dt}M = \frac{d}{dt}\int_{V(t)} \rho dV(t) = 0 \tag{1.5}$$

Applying RTT and the divergence theorem, the final equation is:

$$\int_{V_0} \left(\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v})\right) dV_0 = 0 \tag{1.6}$$

Previous equation is expressed in an integral form, however also a differential form can be define:

$$\boxed{\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0} \tag{1.7}$$

The previous equation has a general validation, however we are interested in steady flow and so another assumption can be made. If the flow is steady, given a generic variable of the flow ψ, then $\partial \psi / \partial t = 0$ and so the mass conservation equation for steady flows becomes:

$$\nabla \cdot (\rho \mathbf{v}) = 0 \tag{1.8}$$

1.3.2 Balance of the momentum quantity

For the second principle of dynamics - the second Newton's law - the variation of the momentum quantity over time in a system is equal to the sum of the external forces acting on the same system:

$$\frac{d\mathbf{Q}}{dt} = \sum \mathbf{F} \tag{1.9}$$

here $\mathbf{Q} = \int_{V(t)} \rho \mathbf{v} dV(t)$ is the momentum quantity. Starting from this basic equation, it is changed to properly fit the fluid dynamics. So, applying the RTT and defining all the forces acting on the system, the equation becomes:

$$\frac{\partial}{\partial t}\int_{V_0} \rho \mathbf{v} dV_0 + \int_{\partial V_0} \rho \mathbf{v}\mathbf{v} \cdot \hat{n} dS = \int_{V_0} d\mathbf{F}_{ev} + \int_{V_0} d\mathbf{F}_{es} \tag{1.10}$$

here, the forces are divided into two contributions:

- $\mathbf{F}_{ev}$: it's the field of forces acting on the whole volume. For simplicity only the gravitational is considered. They are expressed as $\mathbf{F}_{ev} = \rho \mathbf{g} V(t)$. Here $\mathbf{g}$ is the gravity acceleration vector.

- **$\mathbf{F}_{es}$**: it's the field of forces acting on the external surfaces of the volume. They are expressed as $\mathbf{F}_{es} = \mathbf{t}dS$. Here $\mathbf{t}$ is the stress vector.

The stress vector is define by the stress tensor which is a matrix. So

$$\mathbf{t} = \vec{\mathbf{T}} \cdot \hat{n} \longleftrightarrow \vec{\mathbf{T}} = \begin{bmatrix} \sigma_{xx} & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \sigma_{yy} & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \sigma_{zz} \end{bmatrix} \tag{1.11}$$

Substituting the values in equation (1.10) applying the divergence theorem and eliminating the dependence of the system -removing the integral-, the equation becomes:

$$\frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v} \mathbf{v}) = \nabla \cdot \vec{\mathbf{T}} + \rho \mathbf{g} \tag{1.12}$$

that's the balance of the momentum equation in differential form. The stress vector is a field of forces including pressure and friction forces. Moreover, air is a Newtonian fluids -more precisely a Stokes fluid - and so, introducing a constitutive bond, the stress tensor can be define as follows:

$$\vec{\mathbf{T}} = -p\vec{\mathbf{I}} + 2\mu\vec{\mathbf{E}} - \frac{2}{3}\mu(\nabla \cdot \mathbf{v})\vec{\mathbf{I}} \quad \vec{\mathbf{E}} = \frac{\nabla \mathbf{v} + \nabla \mathbf{v}^T}{2} \tag{1.13}$$

here $\vec{\mathbf{I}}$ is the identical matrix, μ is the dynamic viscosity and $\vec{\mathbf{E}}$ is the strain tensor. So, the stress tensor is made up by two terms: pressure and sum of different terms that can be include all as viscous stresses $\vec{\Sigma}$. With these considerations, substituting in equation (1.12) and using mass conservation, a conservative form of momentum balance is defined:

$$\boxed{\rho \frac{D\mathbf{v}}{Dt} = -\nabla p + \nabla \cdot \vec{\Sigma} + \rho \mathbf{g}} \tag{1.14}$$

This is the differential form of momentum balance. With this equation and the mass conservation one, the fluid system is described by 4 equation (1 mass conservation and 3 of momentum balance in the three direction) and 5 unknowns (density, three components of velocity and pressure). The gas state equation $p = \rho RT$ is added but also a new variable T - the temperature - is added in the system. Therefore, the principle of conservation of energy must be used in the system.

1.3.3 Conservation of energy principle

Consider the first principle of thermodynamics:

$$d\dot{\epsilon} = \delta\dot{Q} + \delta\dot{W} \tag{1.15}$$

it expresses the variation in time of the internal extensive energy of the system $d\dot{\epsilon}$ as the sum of variation in time of heat $\delta\dot{Q}$ and work $\delta\dot{W}$. This equation is divided in two parts:

1. The energy of a given system is defined with the sum of the intensive internal energy and the intensive kinetic energy: $\epsilon_s = \int_{V_0} \rho\left(e + \frac{1}{2}\mathbf{v}^2\right)dV_0$

2. the heat is given by the sum of that entered in the volume and transferred by surface conduction: $\dot{Q} = \dot{Q}_v + \dot{Q}_s$

3. the work is given by the sum of volume forces and surface forces: $\dot{W} = \dot{W}_v + \dot{W}_s$

Substituting the previous definition in equation (1.15) and using the RTT, a new equation is given:

$$\frac{d}{dt}\int_{V_0}\rho\Big(e + \frac{1}{2}\mathbf{v}^2\Big) + \int_{\partial V_0}\rho\Big(e + \frac{1}{2}\mathbf{v}^2\Big)\mathbf{v}\cdot\hat{n}dS = \dot{Q}_v + \dot{Q}_s + \dot{W}_v + \dot{W}_s \qquad (1.16)$$

Now, consider only the right part of equation. All the component are defined by an equation:

- $d\dot{Q}_v = \rho Q dV$

- $d\dot{Q}_s = -\mathbf{q}\cdot\hat{n}dS \rightarrow [\text{Fourier law}] : \mathbf{q}{=}K\nabla T$

- $d\dot{W}_v = \mathbf{v}\cdot\rho\mathbf{g}dv$

- $d\dot{W}_s = \mathbf{v}\cdot\mathbf{t}dS = \mathbf{v}\cdot\vec{\mathbf{T}}\cdot\hat{n}dS$

here $\mathbf{q}$ is the thermic flux and K is the thermal conductivity coefficient. Applying the divergence theorem, substituting the variables and with some mathematical steps, the equation of the energy conservation is:

$$\boxed{\rho\frac{D}{Dt}\Big(e + \frac{\mathbf{v}^2}{2}\Big) = -\nabla\cdot(p\mathbf{v}) + \nabla\cdot(\mathbf{v}\cdot\vec{\Sigma}) + \rho Q - \nabla\cdot\mathbf{q} + \mathbf{v}\cdot\rho\mathbf{g}} \qquad (1.17)$$

The previous equation is the last equation that complete the set of Navier-Stokes equations. Evne if they are complete - 6 equations in 6 unknowns - they doesn't admit easy solutions both numerical and analytical. So, let's introduce some hypothesis to simplify the equation (1.17):

- adiabatic flux: $Q = 0$ and $K\nabla T \approx 0$

- max forces are negligible: $\mathbf{g} \approx 0$

- outside the boundary layer [1] the viscous forces are negligible: $\nabla\cdot(\mathbf{v}\cdot\vec{\Sigma}) \approx 0$

- steady flow for pressure: $\dfrac{\partial p}{\partial t} = 0$

Defining enthalpy as $h = e + p/\rho$ and h_0 as total enthalpy, the energy conservation equation becomes:

$$h + \frac{\mathbf{v}^2}{2} = 0 = h_0 \qquad (1.18)$$

here the subscript indicates the *total state* of a variable which is the state at which the variable is carried isoentropically with zero speed conditions.

---

[1] see §1.4

1.4   Boundary layer

The classical Navier-Stokes equations don't have a solution, actually the existence of the solution has not yet been demonstrated. So, these equations are reduced in a simpler set of equations. As previous stated, we considered a Newtonian fluid, in particular a Stokes fluid, and so the set of the new equations becomes simple and solvable.

All the fluxes are viscous and a viscous flux adheres to a wall with its first molecular layers. On the wall, the velocity of the fluid and solid -the wall- coincides: this condition is noted as adhesion condition. The importance of viscous terms in a flux is determined by *Reynolds number*. It is expressed as:

$$Re_L = \frac{uL}{v} \tag{1.19}$$

here $v = \mu/\rho$ is the cinematic viscosity and u the velocity in a preferential direction. Reynolds number is associated always to a characteristic length L which depends from the body considered (i.e. in aerofoils is the chord). Moreover, the Reynolds number defines the ratio between inertial and viscous forces. For values over $Re = 10^5 \div 10^6$, the motion is turbulent. So, for high number Reynolds, the viscous terms are negligible but near the wall the viscosity must be present to ensure the condition of adhesion. Two different zones are present in a flux over a body: an external one where the viscosity is negligible and a zone near the wall where the viscosity is important, the boundary layer.

Given the Navier-Stokes equations, they are described by non-dimensional variables according to a characteristic dimension. The continuity equation, for a incompressible fluid (ρ is a constant) and with gravitational terms negligible, becomes:

$$\nabla' \cdot \mathbf{v}' = 0 \tag{1.20}$$

here $()'$ indicates a non-dimensional variable respect to a characteristic length L. With the same procedure, the equation (1.14) is defined as follows:

$$\frac{D'\mathbf{v}'}{Dt'} = -\nabla'p' + \frac{1}{Re}(\nabla^{2'}\mathbf{v}') \tag{1.21}$$

Last equation represent the idea previously illustrated. So, the flux is divided in two zones:

- **bulk:** is a region sufficiently far from the wall. Here $Re \to \infty$ and the flux is described as if it were inviscid. The only forces are those due to pressure. The set of equations is called Eulero's equations;

- **near wall:** is a region just over the wall where the viscous phenomena are important.

Another important fact is that, due to the adhesion condition on the wall, the velocity decrease from the outer solution. This zone is subjected of a rapidly decrease of momentum. Moreover, this zones is very important for the drag of the object and the flux in it can be laminar or turbulent. However, the velocity showed in Figure 1.2 represents only a specific case.
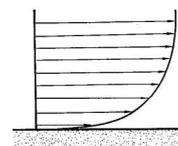
Figure 1.2: Velocity profile
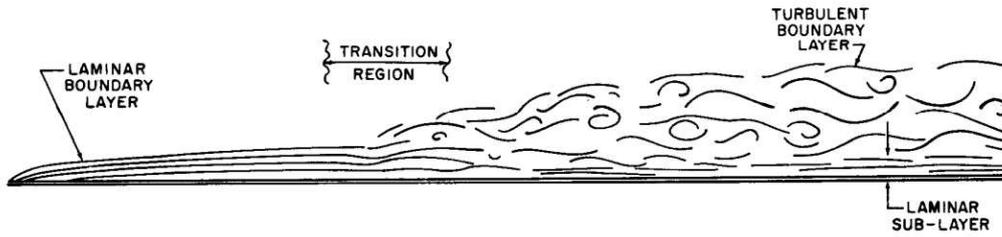
Figure 1.3: Development of boundary layer on a smooth flat plate

Consider a fluid moving over a body as in Figure 1.4 and assume that there is an increase of pressure from point 1 to point 3. As previous stated, the profile velocity near the leading edge of the body takes a curved form with a null value on the surface and a value equal to the external solution in the outer edge. Then, at point 2, the flow separates from the wall because an increase in pressure corresponds to a decrease in velocity. The point of separation occurs where $\partial V/\partial n = 0$. Instead, at the point 3, the flow is reversed and the zone is full of eddies. So, the rapidly increasing of the pressure, caused by the geometry, creates a large wake of recirculating flow downstream of the surface. However, the boundary layer can withstand a small adverse pressure gradient avoiding boundary layer separation. This is achieved by suitable geometries that slowly slow down the flow velocity.
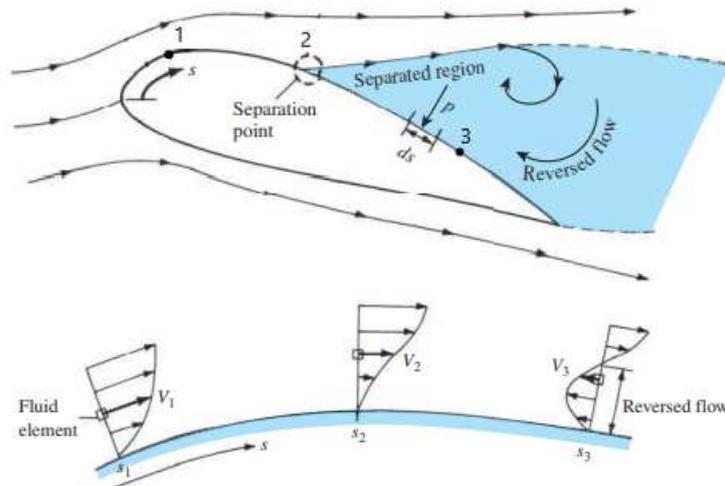
Figure 1.4: Different velocity profiles for different point on the body

The deeply description of all the characteristic of the boundary layer -velocity, height of the layer etc- are too many to be described in this thesis. However, a briefly description is been made and the concepts will be used in the future chapter. More information are presented in [1] and [17].

Chapter 2

Shock waves and high velocity flows dynamic

2.1 Introduction

This chapter provides a view of the high velocity fluxes, the shock waves and their behaviour. These concepts are needed to understand future implications.

2.2 Concepts of thermodynamic

2.2.1 Perfect gas

Considering a perfect gas means a mass of particles where they are far enough to collide between them and generate the intermolecular force. So, a gas with this characteristic is governed by the equation of state:

$$p = \rho R T \tag{2.1}$$

here ρ is the density, p the pressure and R the specif gas constant. Then, the sum of all energies of all the molecules is defined as internal energy e. It is related to a quantity h called enthalpy by the next equation:

$$h = e + pv \tag{2.2}$$

Moreover, for a perfect gas only, internal energy and enthalpy depend on the temperature (in general they depend also on pressure). If their dependence is defined by constant (valid for air for $T < 1000K$), then their relation are so described:

$$
\begin{aligned}
e &= c_p T \\
h &= c_v T
\end{aligned}
\tag{2.3}
$$

here c_p is the specific heats at constant pressure while c_v is the specif heats at constant volume. Furthermore, the two specif heats are related by two equations:

$$
\begin{aligned}
c_p - c_v &= R \\
\gamma = \frac{c_p}{c_v} & \quad (\gamma = 1.4 \ \text{for air})
\end{aligned}
\tag{2.4}
$$

With some basic calculations, two final equation can be defined:

$$c_p = \frac{\gamma R}{\gamma - 1} \ \text{ and } \ c_v = \frac{R}{\gamma - 1} \tag{2.5}$$

2.2.2 Thermodynamic laws

Thermodynamic is governed by different principles. The first of these states that given a fixed mass of gas, δq an incremental amount of heat added to the mass (also called system) and δw the work done on the system, those cause a change of internal energy de as follows:

$$\delta q + \delta w = de \tag{2.6}$$

Three types of process can added heat and do work to the system:

- *Adiabatic:* $\delta q = 0$;

- *Reversible:* no dissipative phenomena occur (no viscosity, thermal conductivity and mass diffusion)

$$\delta w|_{REV} = -pd\left(\frac{1}{\rho}\right) \tag{2.7}$$

- *Isentropic:* both adiabatic and reversible process

The second principle states there is a proper direction of a process. Defining $S(T,p)$ as entropy, then

$$dS = \frac{dq|_{REV}}{T} \longrightarrow S_2 - S_1 \geq \int_1^2 \frac{dq}{T} \tag{2.8}$$

Now, consider a reversible process, combining all the previous equation, it is obtained:

$$de = Tds - pd\left(\frac{1}{\rho}\right) \tag{2.9}$$

From the definition of enthalpy:

$$dh = de + d\left(\frac{p}{\rho}\right) \longrightarrow dh = Tds + \frac{1}{\rho}dp \tag{2.10}$$

Extrapolating the entropy, remembering the equations for a perfect gas (2.3) and (2.1) and integrating the remaining values, the final equation is so defined:

$$S_2 - S_1 = c_p ln\frac{T_2}{T_1} - Rln\frac{p_2}{p_1} = c_v ln\frac{T_2}{T_1} - Rln\frac{\rho_2}{\rho_1} \tag{2.11}$$

Given the previous equations, they can be applied to an isentropic process ($dS = 0$ from equation 2.8), obtaining, by changing position of addends and using the equations (2.4) and (2.5), the following relationship:

$$\frac{p_2}{p_1} = \left(\frac{T_2}{T_1}\right)^{\frac{\gamma}{\gamma-1}} = \left(\frac{\rho_2}{\rho_1}\right)^{\gamma} \tag{2.12}$$

Note that, for aerodynamics problems, isentropic flows are not present in boundary layers and through shock waves. More in dept analysis will be done in future sections.

2.3   High velocity fluxes

Considering a perfect gas, its speed of sound a is a function of temperature only:

$$a = \sqrt{\frac{dp}{d\rho}} = \gamma RT \tag{2.13}$$

Given the previous statement, all flow can be expressed and associated with a variable M called *Mach number*. It defines if the compressibility effects of fluid can be negligible or not. Second case happens only when the flow has velocity near the sound velocity. Moreover, flows with $M < 0.3$ are defined as slow.

$$M = \frac{v}{a} \tag{2.14}$$

Furthermore, the mach number is used to define different categories of flows. They are:

Mach number	Type of flow
< 1	subsonic
~ 1	sonic
> 1	supersonic
≥ 5	hypersonic

Table 2.1: Types of flow depending on Mach number

All these mach number can be easily associated with different cases. A subsonic flow is typical of glider and a supersonic one is reached by military jets. All these types of flows are presents nowadays.

Given the importance of the Mach number, it is used to define the thermodynamic characteristics. All the variable are expressed in their total state. Given an adiabatic, not-viscous and with no gravitational forces flow in stationary motion of a perfect gas, its total thermodynamic variable are so expressed:

$$T_0 = T(1 + \frac{\gamma - 1}{2}M^2) \tag{2.15}$$

$$p_0 = p(1 + \frac{\gamma - 1}{2}M^2)^{\frac{\gamma}{\gamma-1}} \tag{2.16}$$

$$\rho_0 = \rho(1 + \frac{\gamma - 1}{2}M^2)^{\frac{1}{\gamma-1}} \tag{2.17}$$

These are the basic equation used for high velocity flows.

2.4   Shock waves

Subsonic and supersonic flows present some difference regarding the fluid behaviour around a body. In the first one, a disturbance travels upstream to alert the flow of an upcoming object. Then the flow makes the necessary adjustment to overcome the obstacle and so creates space for it in the fluid. In the second one, the flow creates a system of compression and expansion waves to turn itself around the body. This occurs because a

pressure signal travel in a fluid with a velocity equal to the sound velocity, so, in a supersonic flow, it will not be able to change its configuration by communicating upstream the presence of the change.

The waves created by the presence of an object are called *shock waves*. They have a thickness equal to the average free path of molecules and in this space they generate a discontinuity of the flow variables. There can be different types of shock waves: normal and oblique (compression and expansion). All of this types of waves will be discussed in the later sections.

2.4.1 Normal shock

Normal shock waves are normal to the flow and causes a sudden deceleration; static pressure, temperature, density, and entropy rise; Mach number and total pressure drop across the wave. Considering a control volume near a normal shock and indicating as 1 and 2 the flow variables before and after the shock wave, the fundamental equation of fluid dynamics are applied to the zones 1 and 2.
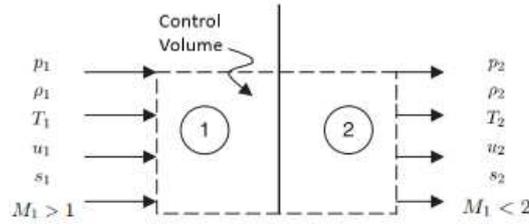
Figure 2.1: Control volume for a normal shock

The continuity equation of a steady flow through the control volume demands:

$$\rho_1 u_1 = \rho_2 u_2 \tag{2.18}$$

The momentum equation in the direction $\hat{n}$ normal to the shock is:

$$p_1 + \rho_1 u_1^2 = p_2 + \rho_2 u_2^2 \tag{2.19}$$

This equation defines the conservation of impulse $I = p + \rho u^2$ across the shock. Finally, the energy equation for an adiabatic flow with no work becomes:

$$h_1 + \frac{u_1^2}{2} = h_2 + \frac{u_2^2}{2} \tag{2.20}$$

So a system of three equations is created in four unknowns -ρ_2, u_2, T_2, p_2. Two new equations are added and so the final system becomes:

$$\begin{cases} \rho_1 u_1 = \rho_2 u_2 \\ p_1 + \rho_1 u_1^2 = p_2 + \rho_2 u_2^2 \\ h_1 + \dfrac{u_1^2}{2} = h_2 + \dfrac{u_2^2}{2} \\ h_2 = cp T_2 \\ \rho_2 = \dfrac{p_2}{RT_2} \end{cases} \tag{2.21}$$

The problem expressed in (2.21) is resolvable and it defines the normal shock equations which correlate the variables before and after the shock:

$$\frac{M_2}{M_1} = \frac{1}{M_1}\sqrt{\frac{1 + \frac{\gamma+1}{2}M_1^2}{\gamma M_1^2 - \frac{\gamma-1}{2}}} \tag{2.22a}$$

$$\frac{\rho_2}{\rho_1} = \frac{(\gamma+1)M_1^2}{2 + (\gamma-1)M_1^2} \tag{2.22b}$$

$$\frac{p_2}{p_1} = 1 + \frac{2\gamma}{\gamma+1}(M_1^2 - 1) \tag{2.22c}$$

$$\frac{T_2}{T_1} = \left[\frac{2\gamma}{\gamma+1}(M_1^2 - 1)\right]\frac{2 + (\gamma-1)M_1^2}{(\gamma+1)M_1^2} \tag{2.22d}$$

As the reader can view, all the relations are expressed in function of M_1 and γ. So, noted the variables before the shock - Mach number included[1] - the variables after the shock are easily obtained by equations (2.22). Others two notes can be done:

- Shock waves are presents only in supersonic flows: considering the second principle of thermodynamics ($S_2 - S_1 > 0$), the equation (2.11) and substituting the variables with the ones obtained in equations (2.22), then it is obtained that $M > 1$;

- $M2 < 1$: This is easily verifiable using equation (2.22a). Moreover, the flow is subsonic after the shock only for normal shock wave.

2.4.2 Oblique shock

Consider the oblique shock wave sketched in Figure 2.2: β is the wave angle, θ is the flow turning angle, u is the normal velocity component and w is the tangential velocity component. Moreover, all the variable are defined as 1 if they are in the zone before the shock or 2 if they are present after the shock.
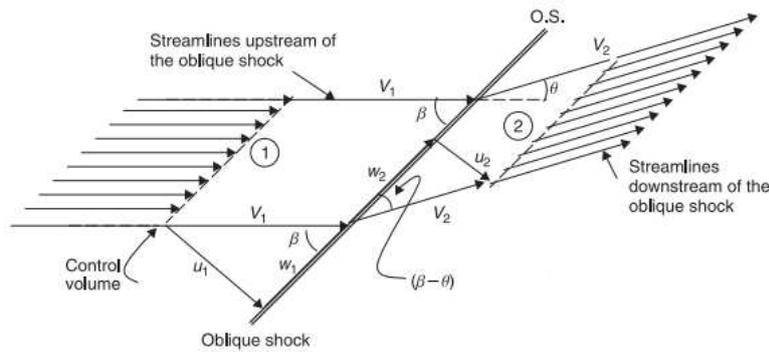
Figure 2.2: Oblique shock

As done before with normal shock, to obtain the variables after the shock is necessary to resolve the continuity, momentum and energy equation. So, indicating with $\hat{n}$ and $\hat{t}$ the normal and tangential direction, the system becomes:

---

[1]Note: for air $\gamma = 1.4$

$$\begin{cases} \rho_1 u_1 = \rho_2 u_2 \\ p_1 + \rho_1 u_1^2 = p_2 + \rho_2 u_2^2 & \text{momentum in } \hat{n} \text{ direction} \\ w_1 = w_2 & \text{momentum in } \hat{t} \text{ direction} \\ h_1 + \dfrac{u_1^2}{2} = h_2 + \dfrac{u_2^2}{2} \\ h_2 = cpT_2 \\ \rho_2 = \dfrac{p_2}{RT_2} \end{cases} \tag{2.23}$$

The system is similar to that one of normal shock wave. The only difference is the conservation of the momentum in the tangential direction. Moreover, the conservation equations normal to the shock take on the exact form of the normal shock equations that were solved earlier to arrive at the shock jump conditions, so an oblique shock can be treated as a normal one using the normal component of the flow to the oblique wave:

$$M_{1n} = M_1 sin\beta \tag{2.24}$$

So, the equations which defines the variables after the shock are:

$$M_{2n}^2 = \frac{1 + \frac{\gamma+1}{2}M_{1n}^2}{\gamma M_{1n}^2 - \frac{\gamma-1}{2}} \tag{2.25a}$$

$$\frac{\rho_2}{\rho_1} = \frac{(\gamma+1)M_{1n}^2}{2 + (\gamma-1)M_{1n}^2} \tag{2.25b}$$

$$\frac{p_2}{p_1} = 1 + \frac{2\gamma}{\gamma+1}(M_{1n}^2 - 1) \tag{2.25c}$$

$$\frac{T_2}{T_1} = \left[\frac{2\gamma}{\gamma+1}(M_{1n}^2 - 1)\right]\frac{2 + (\gamma-1)M_{1n}^2}{(\gamma+1)M_{1n}^2} \tag{2.25d}$$

here the downstream Mach number is defined as:

$$M_2 = \frac{M_{2n}}{sin(\beta - \theta)} \tag{2.26}$$

Also on the oblique shock equations can be done some considerations:

- unlike the normal shock ones, the oblique shock equations depend on M_{1n}, β and γ;

- normal shock waves are a particular case of oblique shock where $\beta = \pi/2$

The variable θ is necessary to evaluate the Mach number after the shock wave, as seen from equation (2.26). So a new relation, called $\theta - \beta - M$, is introduced:

$$\tan\theta = 2\cot\beta \frac{M_1^2 \sin^2\beta - 1}{M_1^2(\gamma + \cos(2\beta)) + 2} \tag{2.27}$$

So, the equations (2.25) with equation (2.27) give the fluid characteristics after the shock (i.e. temperature, density, pressure and Mach number).

The $\theta - \beta - M$ relation is resolved in a graphical way because the equation is implicit and non-linear. So, the chart is presented in Figure 2.3.
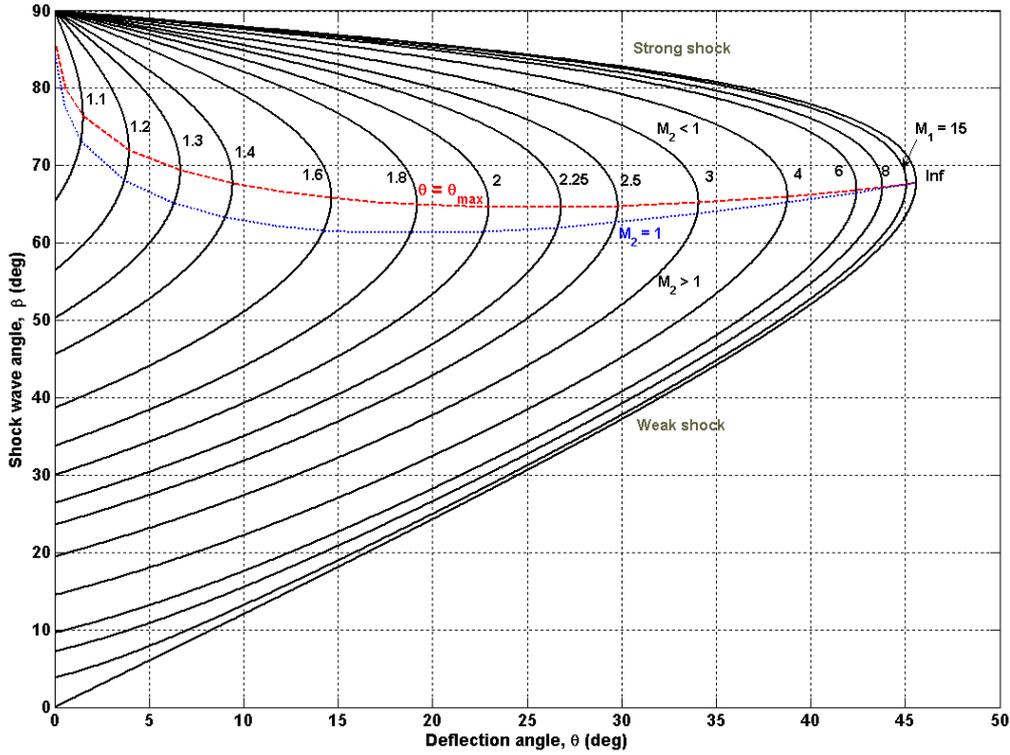
Figure 2.3: $\theta - \beta - M$ relation

From the Figure 2.3 some considerations can be made:

- The equation (2.27) has results if $M_1 > 1$. From chart, the reader can view that there aren't curve for Mach number under the unity. This further affirms that oblique shock only occurs under supersonic flow conditions;

- Consider the blue line. It is the locus of points where $M_2 = 1$. Moreover, the points under this line represent the flow with $M_2 > 1$ and so after the shock there is a supersonic flow, instead the points over the blue line represent the flow with $M_2 < 1$ and so after the shock there is a subsonic flow. The blue line is also called *weak-strong line*;

- Consider the red line. It is the locus of points where $\theta = \theta_{max}$. They are the only points which, given M_1 and θ, the solution of equation (2.27) is only one;

- Given θ, there are two possible cases:

  1. if $\theta < \theta_{max}$ then there can be two types of shocks: strong and weak shock. Nature and experimental data have proven that weak shocks are more likely to occur. So the flow after the shock has an angle $\beta_{2w} < \beta_{2s}$;

  2. if $\theta > \theta_{max}$ there are no solutions. In this case, a locally normal shock wave is formed detached from the point of discontinuity of the geometry. The shock formed is a strong one. An example view is given in Figure 2.5.
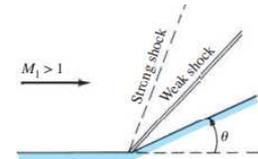
Figure 2.4: Weak and strong shocks
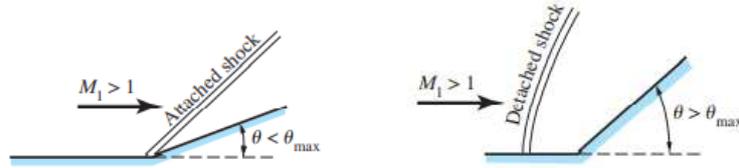
Figure 2.5: Attached and detached shock

In order to summarise and give a methodology for the study, the steps for calculating the characteristics of the fluid after oblique shock are then described:

1. Noted the variable befor the shock -T, ρ, p, M- and the geometric angle of the obstacle θ, then the flow angle is evaluted with the equation (2.27). In particular, the chart of Figure 2.3 is used;

2. Normal component M_{1n} of the flow is evaluated with equation (2.24);

3. M_{2n} is calculated with equation (2.25a) and other flow variables after the shock are evaluated with equations (2.25);

4. Finally, M_2 is derived by equation (2.26).

2.4.3 Expansion shock

Expansion shock waves occur when a supersonic flow is turned away from itself. Expansion Mach waves are capable of only turning and accelereting the flow incrementally. They present a zone in which there is an expansion fan, a region where can be visualized an infinite number of Mach waves. The fan is bounded in a region by forward and rearward Mach lines. Moreover, it is formed by continuous succession of Mach wave, each isentropic, so the whole expansion fan is considered isentropic. Similar to the oblique shock waves, expansion ones are formed only in supersonic flows however an expansion wave is isentropic while an oblique wave always experiences an entropy increase.
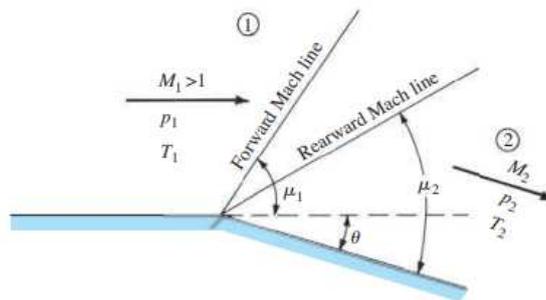
Figure 2.6: Expansion shock wave

To obtain the equation of this type of shock, it is necessary to consider the Figure 2.7. The relation between turning angle and speed is:

$$\frac{V + dV}{\sin(\frac{\pi}{2} - \mu)} = \frac{V}{\sin(\frac{\pi}{2} - \mu - d\theta)} \tag{2.28}$$
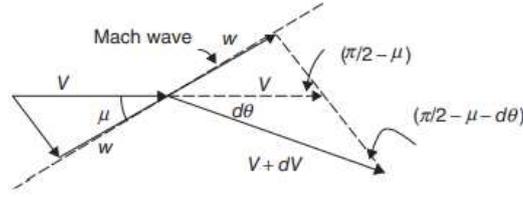
Figure 2.7: Geometrical construction for the changes across a Mach wave

Let us simplify the sines, isolate the velocity terms, apply small angle approximation to $\sin\theta$ and $\cos\theta$ and use Taylor expansion with truncation at the linear term ($1/(1\pm\epsilon) \approx 1 \mp \epsilon$), such as:

$$1 + \frac{dV}{V} \approx 1 + d\theta \cdot \tan\mu \tag{2.29}$$
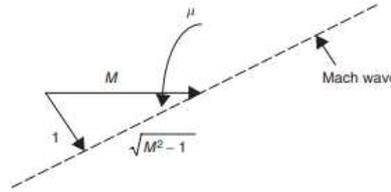
Figure 2.8: Mach triangle

Consider the triangle in Figure 2.8:

$$\tan\mu = \frac{1}{\sqrt{M^2 - 1}} \tag{2.30}$$

Substituting the equation (2.30) in (2.29), it is obtained a relation between the turning angle and the incremental acceleration:

$$d\theta = \sqrt{M^2 - 1}\frac{dV}{V} \tag{2.31}$$

We want to express the previous relation in function only of the Mach number. So, remembering that velocity is a function of Mach and temperature, then, the ratio of the incremental velocity can be espessed as:

$$\frac{dV}{V} = \frac{dM}{M} + \frac{1}{2}\frac{dT}{T} \tag{2.32}$$

Noted the equation (2.15) for temperature and considered T_0, the total state, as a constant across the supersonic waves, then the logarithmic derivative of temperature is:

$$\frac{dT}{T} = \frac{-(\gamma - 1)MdM}{1 + \dfrac{\gamma - 1}{2}M^2} \tag{2.33}$$

Substituting equation (2.33) in (2.32) and all again in (2.31) and integrating, the final equation for the turning angle of the flow becomes:

$$\theta = \int_{M_1}^{M_2} \frac{\sqrt{M^2 - 1}}{1 + \frac{\gamma-1}{2}M^2}\frac{dM}{M} \tag{2.34}$$

23

This integral for an arbitrary Mach number is defined by the Prandtl-Meyer function as follow:

$$\nu(M) = \sqrt{\frac{\gamma+1}{\gamma-1}} \arctan \sqrt{\frac{\gamma+1}{\gamma-1}(M^2-1)} - \arctan \sqrt{M^2-1} \qquad (2.35)$$

So, the deflection of the flow can be finally expressed as:

$$\theta = \nu(M_2) - \nu(M_1) \qquad (2.36)$$

Some considerations can be made for the equation (2.35):

- for $M = 1$ then $\nu(M) = 0$;

- for $M \to \infty$ then $\nu = \dfrac{\pi}{2}\sqrt{\dfrac{\gamma+1}{\gamma-1}} - 1 \approx 130.45°$ and so the flow has a maximum expansion angle even if the geometry allows more space.

To summarise, as done for the compression waves, the calculation process for the variables after expansion is also described here. Noted the variables of the flow before the expansion wave, the procedure can be described as follows:

1. evaluation of $\nu(M_1)$ from equation (2.35);

2. noted the deflection angle θ, evaluation of $\nu(M_2)$ from equation (2.36);

3. calculation of M_2 from equation (2.35);

4. since the expansion wave are isentropic, equations (2.15),(2.16) and (2.17) are used to obtain the values of the variables after the shock.

To speed up the process, it is advisable to use tables in the literature (see [1]) which provide known values for equation (2.35), thus avoiding analytical calculation.

2.4.4   Shock interactions and reflections

The analysis of shock waves, in particular the oblique ones, is made with interaction between the supersonic flow with a geometric discontinuity which generates the oblique shock. However, this situation is particular and not very present in nature because the flow is channelled between more surfaces and so the oblique shock wave can not extend indefinitely. It will impact with another surface or an additional shock wave. These phenomena are important in the design and analysis of supersonic airplanes, missiles, rocket engines, etc. So, a qualitative discussion of shock wave interactions and reflection is made.
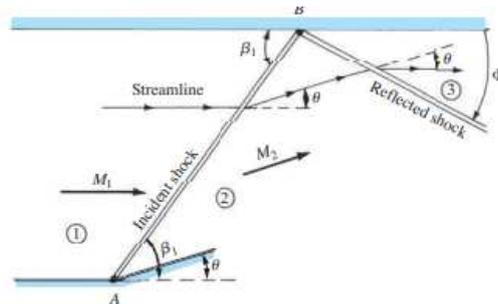
Figure 2.9: Reflection of a shock wave in a channel

For the first case, consider a section of a channel as shown in Figure 2.9. Here, the geometry discontinuity A creates the compression shock wave and the free stream flow is deviated by an angle equal to θ. The oblique shock wave could travel to infinity but it impinge at point B the upper surface of the channel. The flow in region 2 must be bent downward with an angle equal to θ in order to maintain a flow tangent to the upper wall. This happens with a second shock, called reflected shock. The properties of region 2 and 3 are determined by the equation used for the oblique shock relation.
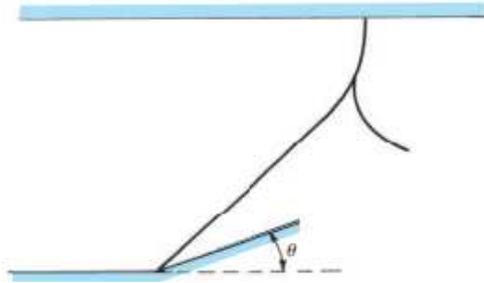
Figure 2.10: Delta reflection of a shock wave in a channel

In nature, but also with the validation of computational fluid dynamics and with experimental tests, the reflection of the shock as explained in Figure 2.9 is very rare. A realistic view is shown in Figure 2.10. The oblique shock wave, generated by the discontinuity, becomes curved near the upper wall condition. Here, locally a normal shock wave is created. This happens to allow the flow after the shock to becomes parallel to the wall behind the shock intersection. Moreover, a curved shock spreads from the intersection and propagates downstream. This type of interaction of shock waves is called λ, due to the shape similarity with the Greek letter.
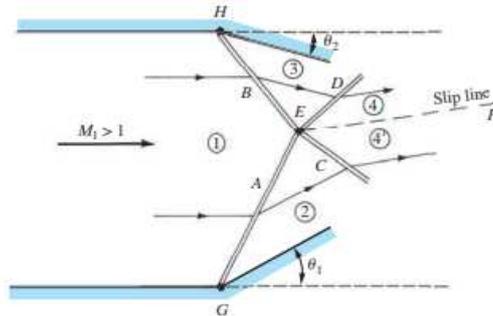
Figure 2.11: Intersection of shock waves in a channel

Another case is the interaction between different oblique shocks generated by different geometry discontinuity, as shown in Figure 2.11. The two oblique shocks are generated by points A and B and the properties downstream, in region 2 and 3, are calculable with the equation for oblique shock waves. Then, the two waves propagate in two different direction until they intersect at the point E. Here, both are reflected and their path continue with another wave: A becomes D and B becomes C. The properties in region 4 and 4′ are not calculable with analytical equations but only with computational fluid dynamics or experimental tests. However, the different properties of the two regions are divided by a so called *slip line*. Across it, the pressure is constant ($p_4 = p_{4'}$) and the direction of the velocity is the same ($\hat{v}_4 = \hat{v}_{4'}$). All other properties are different.
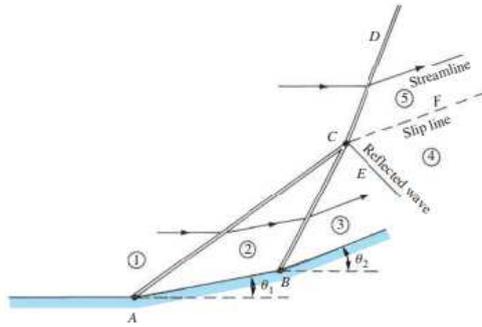
Figure 2.12: Union of shock waves

An additional case is represented in Figure 2.12. Two oblique shock wave are generated by two different geometry discontinuity on the same wall. The properties of the flow in the zones 2 and 3 are easily evaluated with the relation for oblique shock waves. In this case, the intersection of the two shocks occurs at the point C. Here, they merge and propagate with a new stronger shock represented by line CD and a weaker one described by CE line. Last one is necessary to adjust the direction of the flow in order to be tangential to θ_2. Again, the downstream properties of zone 4 and 5 are different but they are divide by the slip line with the same characteristic described for the previous case.

All of these cases are only some of the most commonly situations happening in nature but with high frequency. Being only a qualitative explanation of the phenomena, the discussion stops here.

As discussed earlier, shock waves can interact with the boundary layer. This situation frequently happens and it very complex. Consider a inviscid flow, the shock wave reflection can be visualized as in Figure 2.9. There is discontinuous pressure increase at point B, a combination of incident and reflected shock waves. So, in this point, there is a very infinitely large adverse pressure gradient. As discussed in Chapter 1.4, if the fluid is viscous, and so there is the boundary layer, then the adverse pressure gradient would cause the boundary layer to detach.
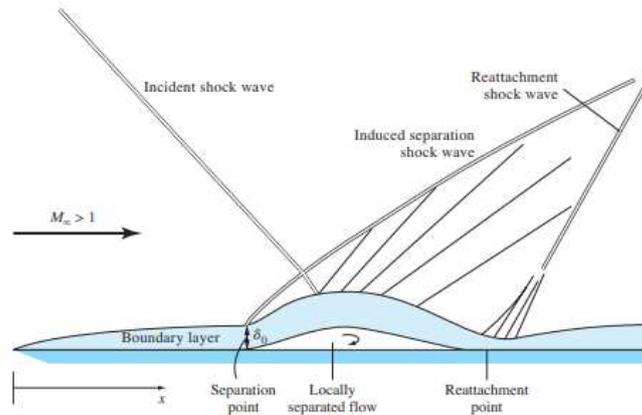
Figure 2.13: Boundary layer interaction

Consider for example the scheme of Figure 2.13. The wall presents a growing boundary layer where the flow is subsonic near the wall to reach the supersonic condition near the

outer edge. An incident shock wave impacts on the boundary layer with an adverse pressure gradient that acts on the boundary layer and generates detachment. Here, the flow presents a recirculation bubble in which there are vortices structures. Last ones generates a bump of the boundary layer. Moreover, the flow after the incident shock wave is induced to turn into itself, due to the condition imposed by the new shape of the downstream boundary layer. This bending of the flow induce a second shock wave, identified as the induced separation shock wave. Later, the boundary layer turns to the surface and reattaches itself on the latter. The external flow follows the turn of the boundary layer turning away from itself and this generates expansion waves. Finally, in the reattachment point, the external flow turns into itself causing the generation of the final shock wave called reattachment shock wave.

2.5 Hypersonic flows

Hypersonic flows have some differences with the supersonic ones. Consider a flat plate or a tapered body. Using the theory previously discussed, noted M_∞ and θ, the angle described by the shock wave is evaluated using the $\theta - \beta - M$ relation. It is noted that the $\beta \approx \theta$, so, in an inviscid fluid, for very high Mach number, the oblique shock wave are almost tangential to the geometry. In hypersonic flows, the viscous terms are important. In nature, the boundary layer after the shock are very thin for two reasons:

- the boundary-layer thickness in slender body is approxximately proportional to M_∞^2;

- hypersonic vehicles generally fly at high altitudes where the density is low and so the Reynolds number is also low.
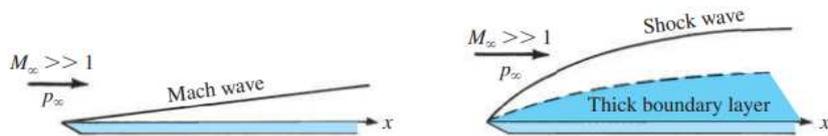
Figure 2.14: Inviscid flow (left) and viscous flow (right)

As shown in Figure 2.14, the thick boundary layers create a bend of the shock wave which becomes very strong in the leading edge of the flat plate.

Also high-temperature effects are very important in hypersonic flow. At high temperature, the gas is not ideal but real, so the temperature cannot be calculated with the usual shock wave theory. Moreover, the chemical dissociation reactions of oxygen and nitrogen take part in the energy balance. So, kinetic energy is transformed into chemical and thermal energy reducing the final temperature values after the shock.

The oblique shock relations previously discussed are the starting point to obtain approximate and simplified version for high Mach numbers. So, consider a classic oblique shock wave, the equations (2.25) and substitute $M_{1n} = M_1 \sin\beta$. For hypersonic flows $M_1 \to \infty$ and so $M_1^2 \sin^2\beta$. Finally, the equations for hypersonic flow, with 1 and 2 the zones before and after the shock, are so described:

$$\frac{p_2}{p_1} = \frac{2\gamma}{\gamma+1}M_1^2\sin^2\beta \to \infty \tag{2.37a}$$

$$\frac{\rho_2}{\rho_1} = \frac{(\gamma+1)M_1^2 sin^2\beta}{(\gamma-1)M_1^2\sin^2\beta+2} \approx \frac{\gamma+1}{\gamma-1} = \begin{cases} 6 & \text{for } \gamma = 1.4 \\ \infty & \text{for } \gamma \to 1 \end{cases} \tag{2.37b}$$

$$\frac{T_2}{T_1} = \frac{2\gamma(\gamma-1)M_1^2 sin^2\beta}{(\gamma+1)^2} = \begin{cases} \infty & \text{for } \gamma > 1 \\ 1 & \text{for } \gamma \to 1 \end{cases} \tag{2.37c}$$

From the $\theta - \beta - M$ relation in equation (2.27), in the hypersonic range, it becomes:

$$\tan\theta = \frac{2}{\tan\beta}\left[\frac{\sin^2\beta-1}{\gamma+\cos 2\beta+2}\right] \approx \frac{2}{\tan\beta}\frac{\sin^2\beta}{\gamma+\cos 2\beta} \tag{2.38}$$

Consider a slender body, then $\theta, \beta << 1$, so:

$$\beta = \frac{\gamma+1}{2}\theta = \begin{cases} 1.2\theta & \text{for } \gamma = 1.4 \\ \theta & \text{for } \gamma \to 1 \end{cases} \tag{2.39}$$

The previous equations are only used for theoretical analysis. The use of the exact oblique shock equation are always recommended.

Chapter 3

Hypersonic air intake

3.1 Introduction

Air intake is the first component of the engine aircraft in contact with air. The requirements of the intakes depend on the mission of the aircraft, but in general they are designed to follow some general requirements:

- provide spatially an temporally smooth flow in to the engine with minimum distortion;

- compress the flow;

- minimize shock and viscous losses to improve the drag coefficient;

- light weight and low cost manufacture;

- independence of performances by operating at different incidences;

- tolerate back pressure caused by heat addition;

- provide particle separator;

- allow for engine thrust growth;

There are some conflicts in the requirements, so reaching an optimal compromise solution is not an easy task.

Inlets are different in shape and behaviour depending on the operating conditions they are working on. So, two categories are defined: subsonic and supersonic/hypersonic inlet. This chapter aims to give a first and general description of intakes, in particular those operating at high Mach number. Concerning the objectives of this thesis, an overview is made for hypersonic intakes.

3.2 Hypersonic inlet characteristics

All supersonic and hypersonic inlets must slow down the flow to compatible velocities for the compressor or chamber of combustion for the ramjet. From the isoentropic gasdynamics:

$$\frac{dA}{A} = -\frac{d\mathbf{v}}{\mathbf{v}} \cdot (1 - M^2) \qquad (3.1)$$

29

For a supersonic intake, the equation (3.1) gives a convergent–divergent (C–D) duct capable of decelerating a supersonic flow to a subsonic one reversibly and adiabatically.
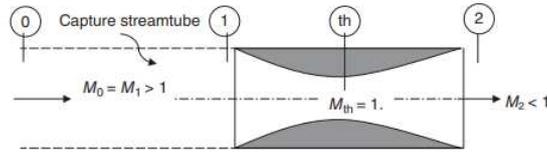
Figure 3.1: Convergent-divergent supersonic intake

From the theory, an area can be estimated to have a desirable Mach number at the exit of the intake. However, this is not a reachable solution because supersonic intakes have two major issues:

1. *"unstarted" problem*: the incapability to reach the design conditions;

2. *buzz*: a shock wave is alternately swallowed and regurgitated by the inlet generating interaction with boundary layers and flow instability

This two phenomena and their resolution are briefly explain in the next sections.

3.2.1 "Unstarted" problem

The inlet flow starts at a very low speed, in the subsonic regime. Here, the area capturing the air is larger than that of the throat and the velocity in the throat never reaches the speed of sound. Subsequently, the speed increases, remaining in the subsonic regime, and the throat has a unit Mach number. This is where the first bottleneck occurs. As the velocity increases again, close to the transonic regime, the capture area is smaller than the starting section of the inlet, an outflow occurs and therefore part of the mass flow does not enter the inlet. In the supersonic regime, a shock wave is generated before the inlet. This phenomenon also creates mass flow spillage and induces high pressure and resistance losses. The process can be visualized in Figure 3.2.
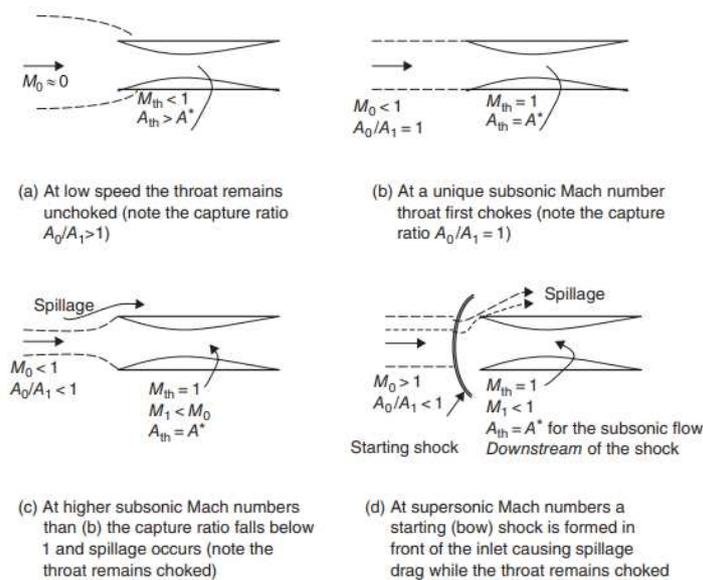
Figure 3.2: Flow characteristics of an isentropic C–D inlet from low speed to supersonic conditions

In order to work in the design condition, the shock wave needs to be swallowed by the intake. Without explaining the whole procedure in detail, it is known that in converging sections the shock wave is not stable and it is therefore swallowed up, whereas in diverging sections the shock wave is positioned at a point in the section that depends on the value of the backpressure. The latter will place the shock at the throat for a maximum total pressure recovery.

The swallowing of the shock wave can be done in different way:

- **Overspeed**: the freestream reaches a Mach number greater than design one, so the starting shock is at the inlet lip and the process of swallowing of the shock starts;

- **Kantriowitz-Donaldson intake**: a larger throat area is established with respect to isentropic throat. The starting shock is formed in front of the intake and then it is swallowed and backpressure adjusted;

- **Variable throat**: follow the same principle of K-D intake. Starting shock is generated at Mach design, the throat is opened and the shock reach the convergent section.
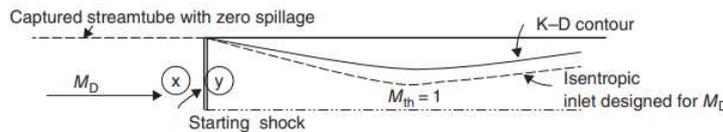
Figure 3.3: A K–D inlet at design Mach number

3.2.2 Buzz

The second major problem of supersonic/hypersonic intakes is the buzz instability. It is an unsteady cyclic phenomenon resulting from the interaction between shock waves and boundary layer which generates separation of the flow. The shock wave has a back-and-forth oscillatory motion that causes high resistance to spillage and distortion of the fluid flow. The entire process can be schematized as in Figure 3.4.
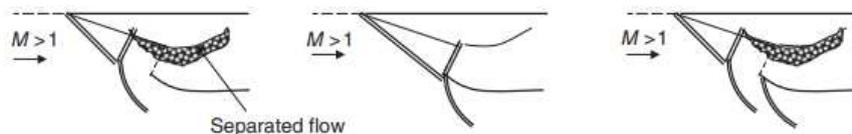
Figure 3.4: A Buzz instability cycle

3.3 Mixed-Compression Inlets

For high speed flow, as in hypersonic range, the mixed-compression intakes are most used. They are very complex but follow the rule that more ramps generate weaker shocks and so a minor loss of pressure. Moreover, they allow internal shock reflection for better performance reducing the curvature of the cowl lip. They can be operation up to Mach 5 with high total pressure recovery. However, they present several problems:

- the present of internal throat is source of starting problem. It is mitigated using variable throat geometry;

- external compression surfaces indicates possible buzz instability. Boundary layer suction and throat bleed are built to overcome the instability;

- separation of boundary layer and distortion due to the interaction with shock waves;

- terminal shock position. It is regulated by the two pitot tube and the changes on backpressure.
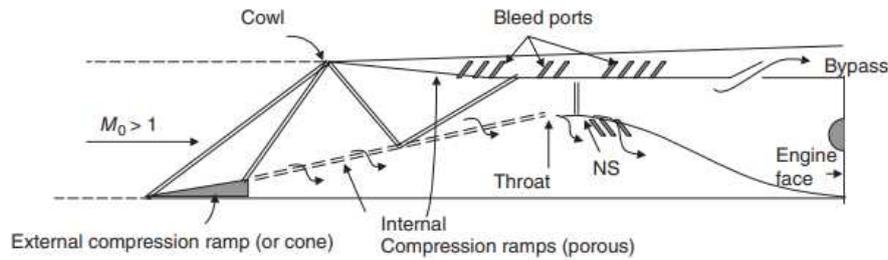
Figure 3.5: A mixed compression inlet with external and internal shocks

The design of the hypersonic inlet is related to a Mach number. Obviously, the intake meets the flow also at different velocities. If the velocity is under the design value, the shocks, produced by the presence of the ramps, converge before the cowl lip generating spillage, high drag and low mass flow. Instead, a velocity over the design one pushes the shocks in the inlet with possibilities of generating flow detachment and distortion.
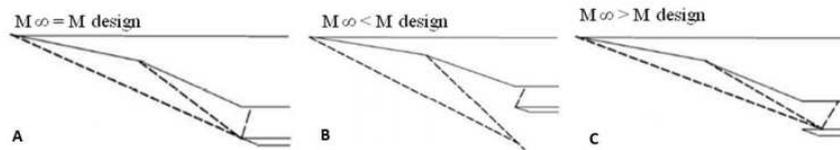
Figure 3.6: Hypersonic intake desing and off-design conditions

Hypersonic intake are currently under studies to mitigate all the problems previously discussed to improve the overall inlet performance. For example, NASA built an hypersonic aircraft running a scramjet engine (Figure 3.7a) with an inlet similar to the one previously discussed. Similar inlet will be discussed in this thesis with a view of improving his performance.

(a) X-43 side view

(b) Inlet scheme

Figure 3.7: X43-A hypersonic aircraft

Chapter 4

Basic concepts of Computational Fluid Dynamics

4.1 Introduction

The aim of this chapter is to provide to the reader the basic concepts of Computation Fluid Dynamics. Here, it will explain how the theoretical formulas are applied to a modern computer and how the latter computes the variables and parameters of the fluid flow.

4.2 Elements of computational fluid dynamics

4.2.1 Navier-Stokes equation

The Computational Fluid Dynamics (CFD) is that branch of fluid dynamics that aims to solve problems related to fluid motion through the use of calculation tools. It is used to perform the validation of the theoretical models or to avoid the use of wind gallery to perform fluid analysis for a new experimental model. It reduces production cost of new components of an aircraft through the use of calculation tools for various tests.

In particular, the CFD resolves numerically the set of differential Navier-Stokes' equations. From equation (1.7), (1.14) and (1.17):

$$\begin{cases} \dfrac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \\[2mm] \rho \dfrac{D\mathbf{v}}{Dt} = -\nabla p + \nabla \cdot \vec{\Sigma} + \rho \mathbf{g} \\[2mm] \rho \dfrac{D}{Dt}\left(e + \dfrac{\mathbf{v}^2}{2}\right) = -\nabla \cdot (p\mathbf{v}) + \nabla \cdot (\mathbf{v} \cdot \vec{\Sigma}) + \rho Q - \nabla \cdot \mathbf{q} + \mathbf{v} \cdot \rho \mathbf{g} \end{cases} \tag{4.1}$$

This is a system of 5 equations in 7 unknown. It is closed by other two equations: $\rho = \rho(p, T)$ as the constitutive fluid model ($p/\rho = RT$ for ideal gas) and $e = e(p, T)$ as internal energy model ($e = c_v T$ for ideal gas).

Navier-Stokes' equation are a set of partial differential equations. This kind of equations needs the initial state of the system to be resolved. This initial condition is called Cauchy condition. It is necessary but it is not enough to resolve the differential equations. The conditions on the boundary of the spatial domain are also defined. Given $\phi = \phi(x, t)$

as a variable that can evolve in space and time, the spatial condition can be expressed in three ways:

- **Dirichlet's condition**: variable has an imposed value changing only in time

$$\phi(x_a, t) = \phi_a(t) \tag{4.2}$$

- **Neumann's condition**: derivative on the boundary has a imposed value

$$\frac{\partial \phi(x_a, t)}{\partial x} = \frac{\phi_a(t)}{\partial x} \tag{4.3}$$

- **Robin's condition**: linear combination of Dirichlet and Nuemann's conditions

$$\phi(x_a, t) = \alpha \phi_a(t) + \beta \frac{\partial \phi_a(t)}{\partial x} \quad \alpha, \beta \in \mathbb{R} \tag{4.4}$$

Given the initial and the boundary conditions, the partial differential equations can be solved.

4.2.2 Mesh

In order to evaluate the partial differential equation in the domain, it is necessary to define a computational grid, called mesh. This grid is a set of points and elements that describe in a discrete way the fluid domain.

The continue solution of the flow becomes a discrete solution because it is defined only in the points of the grids. There is not only a spatial grid but the flow is also temporally discretized. If the fluid changes is behavior over time, it is called *unsteady*. While the fluid is static over time, it is called *steady* and the discretization of the time is not necessary.

The finite difference method defines a derivative of a function through incremental ratios truncated to a certain order of magnitude. Defining a field variable $\phi(x, t)$ with $x \in [0; L_x]$ $\forall t > t_0$, here L_x is the length of the domain and it is divided by a set of points $x_i = (i-1)L_x/n_x = (i-1)\Delta x$ where n_x is the number of the points. Using nodal terms of variable field $\phi(x_i, t)$, the derivative in the i-th point can be written as follow:

$$\phi_{i+1} = \phi_i + \frac{d\phi_i}{dx}\Delta x + \frac{1}{2}\frac{d^2\phi_i}{dx^2}\Delta x^2 + \frac{1}{6}\frac{d^3\phi_i}{dx^3}\Delta x^3 + \mathcal{O}(\Delta x^4) \tag{4.5}$$

The derivative can be truncated to the first order and extrapolated from the previous equation, it becomes:

$$\phi_{i+1} = \phi_i + \frac{d\phi_i}{dx}\Delta x + \mathcal{O}(\Delta x^2)$$
$$\frac{d\phi_i}{dx} = \frac{\phi_{i+1} - \phi_i}{\Delta x} + \mathcal{O}(\Delta x^2) \tag{4.6}$$

The derivative in equation 4.6 is called *downwind* because it is obtained using i-th and (i+1)-th points of the grids. Similarly, an *upwind* derivative can be expressed as follow:

$$\phi_{i-1} = \phi_i - \frac{d\phi_i}{dx}\Delta x + \frac{1}{2}\frac{d^2\phi_i}{dx^2}\Delta x^2 - \frac{1}{6}\frac{d^3\phi_i}{dx^3}\Delta x^3 + \mathcal{O}(\Delta x^4) \tag{4.7}$$

$$\phi_{i-1} = \phi_i - \frac{d\phi_i}{dx}\Delta x + \mathcal{O}(\Delta x^2)$$

$$\frac{d\phi_i}{dx} = \frac{\phi_i - \phi_{i-1}}{\Delta x} + \mathcal{O}(\Delta x^2) \tag{4.8}$$

Another kind of derivative can be obtained by the sum of 4.6 and 4.7 and, with a truncation of the third order, the expression becomes:

$$\frac{d\phi_i}{dx} = \frac{\phi_{i+1} - \phi_{i-1}}{2\Delta x} + \mathcal{O}(\Delta x^2) \tag{4.9}$$

This derivative is called *central* because it used (i-1)-th and (i-1)-th points. These derivatives can be visualized in Figure 4.1.
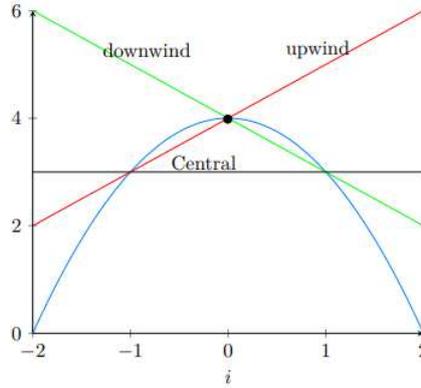
Figure 4.1: Different kind of derivatives of a function (blue)

With the same logic, also the second and third order derivative can be written and their central,upwind and downwind derivatives can be obtained starting from the following description with a process similar to the previous one.

$$\phi_{i+1} = \phi_i + \frac{d\phi_i}{dx}\Delta x + \frac{1}{2}\frac{d^2\phi_i}{dx^2}\Delta x^2 + \mathcal{O}(\Delta x^3) \tag{4.10}$$

$$\phi_{i+1} = \phi_i + \frac{d\phi_i}{dx}\Delta x + \frac{1}{2}\frac{d^2\phi_i}{dx^2}\Delta x^2 + \frac{1}{6}\frac{d^3\phi_i}{dx^3}\Delta x^3 + \mathcal{O}(\Delta x^4) \tag{4.11}$$

The more the derivative order is higher, more higher is its accuracy. However, more points need to be used and so the computational time can increase very rapidly. Usually, the commercial software don't go over third order.

4.2.3 Convergence and stability

The numerical solutions given by CFD software need to be validated. They can give some results but way different from experimental and theoretical ones. In particular, there are two problems that affect the results of the simulations: the error between the numerical and correct solution and numerical error of the calculator.

Regarding the first one, given ϕ^n as the numerical solution and $\phi(t_n)$ the right solution, then a method has convergence if:

$$\lim_{\Delta t \to 0} \phi^n = \phi(t_n) \tag{4.12}$$

or, in a better way:

$$\lim_{\Delta t \to 0} e^n = 0 \qquad (4.13)$$

here $e^n = \phi^n - \phi(t_n)$ is the numerical error at step n.

A method has convergence means that the process of discretization happened in the right way and the passage between a continue domain to a discrete one is maintained. Also, convergence is determined by residual: it measures the local imbalance of a conserved variable in each control volume. Therefore, every cell in the model will have its own residual value for each of the equations being solved [2].

Moreover, due to the fact that a calculator has a finite arithmetic (i.e. 10E-14 in double precision), the calculator introduces an error of approximation. The stability represents the insensitivity of the method to this error. A method is stable if it produces an infinitesimal succession of solution ϕ^n for $n \to \infty$, so:

$$\lim_{n \to \infty} \phi^n = 0 \qquad (4.14)$$

So the method doesn't produce divergence due to the accumulation of perturbation errors of the finite representation of the numbers in the calculator.

4.2.4 Reynolds Average Navier-Stokes equations

Navier-Stokes equations governs any fluid motion, but their intrinsic non-linearity is responsible for critical problems, e.g. turbulence, multiscale nature of turbulent flows. The computational fluid dynamics (CFD) describes the fluid flow with different methods. These methods describes in different way the turbulence, the main source of inaccuracy for CFD codes. Three kinds of equations can be used:

- **Direct Navier-Stokes (DNS):** solves directly the governing equations of fluid flows without the use of any modelling assumption. It requires to solve the extensive range of temporal and spatial scales of a turbulent flow, from very large to very small. Its computational time is very high (Re^3);

- **Large Eddy Simulation (LES):** simulates the large scales of turbulence but models the smallest one. Less accuracy and computational time of DNS.

- **Reynolds Averaged Navier-Stokes (RANS)** evaluates the fluid flow taking into account only the mean values. It has the minimum computational time.
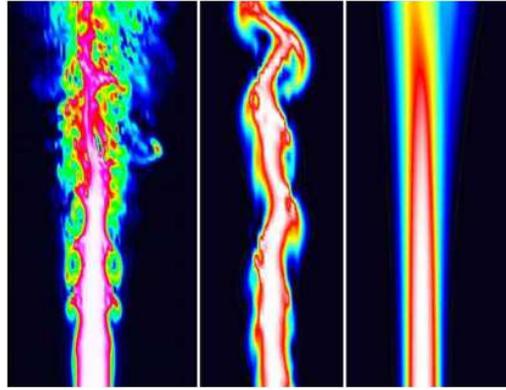
Figure 4.2: DNS (left), LES (middle) and RANS (right) predictions of a turbulent jet [14]

All fluid phenomena are governed by Navier-Stokes equations. Everything is deterministic: given the initial conditions (ICs) and boundary conditions (BCs), the future evolution of the system is prescribed. Yet, due to very small disturbances, even not measurable, of ICs and BCs, the solutions are very different: this phenomenom is called deterministic chaos.

A generic quantity of a flow can be divided in two terms with different characteristics:

1. **instantaneous:**

   - with deterministic chaos, strongly depending on BCs and ICs;
   - no symmetry in the system;
   - high variation between high and low scale[1];
   - intensity of gradients increases with Reynolds.

2. **medium:**

   - determinism;
   - recover of the symmetry of the system;
   - smooth variation on the scale of the object;
   - similarity with laminar fluxes

So, a generic quantity, for example the velocity, can be described as:

$$u(x,t) = U(x,t) + u'(x,t) \tag{4.15}$$

here $u(x,t)$ is the instantaneous velocity, $U(x,t)$ is the mean velocity and $u'(x,t)$ are the velocity fluctuations.

Moreover, the Reynolds average is an ensemble average operator that applies to different components of a flow:

$$U(x,t) = \langle u(x,t) \rangle = \frac{1}{N} \sum_{h=1}^{n} u_h(x,t) \tag{4.16}$$

---

[1]Turbulent flows are multiscale phenomena: vortical structures (eddies) span for 6-8 order of magnitude in size

Equation (4.15) is called Reynolds decomposition. If the problem has steady boundary condition (in mean), the Reynolds average velocity does not depend on time so the Reynolds average coincides with time average (ergodicity).

Now, let's consider the incompressible Navier-Stoke's equation (density $\rho = costant$):

$$\begin{cases} \nabla \cdot u = 0 \\ \dfrac{\partial u}{\partial t} + u \cdot \nabla u = -\dfrac{\nabla p}{\rho} + \nu \nabla^2 u \end{cases} \tag{4.17}$$

First, let's introduce the Reynolds decomposition into the equations but remember some rules:

- Being the Reynolds average a linear operator, it commutes with all the derivative operators that are linear;

- The Reynolds average of a fluctuating quantity is zero: $< x' >= 0$;

- The Reynolds average of an averaged quantity is: $< X >= X$

Then, apply the Reynolds average to the equations. Starting with continuity, expressed in tensor notation:

$$\left\langle \frac{\partial (U_i + u_i')}{\partial x_i} \right\rangle = \left\langle \frac{\partial U_i}{\partial x_i} \right\rangle + \left\langle \frac{\partial u_i'}{x_i} \right\rangle = \frac{\partial \langle U_i \rangle}{\partial x_i} + \frac{\partial \langle u_i' \rangle}{\partial x_i} = \frac{\partial U_i}{\partial x_i} = 0 \tag{4.18}$$

$$\frac{\partial U_i}{\partial x_i} = 0 \rightarrow \nabla \cdot U = 0 \tag{4.19}$$

$$\frac{\partial u_i'}{\partial x_i} = 0 \rightarrow \nabla \cdot u' = 0 \tag{4.20}$$

Equation (4.19) defines the mean velocity field is solenoidal while equation (4.20) defines the fluctuating velocity field is solenoidal. Applying the same procedure to the momentum equations, they become:

$$\rho \left(\frac{\partial U_i}{\partial t} + U_j \frac{\partial U_i}{\partial x_j} \right) = \frac{\partial}{\partial x_j} \left[\mu \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) - P\delta_{i,j} - \rho < u_i' u_j' > \right] \tag{4.21}$$

So a new system of equation is generated: 4 equations (3 given by the components of the momentum and 1 given by the continuity) in 10 unknowns (P, U, V, W and the Reynolds stresses: $< u'u' > < u'v' > < u'w' > < v'v' > < w'w' > < v'w' >$). The Reynolds stress can be thought as originated by an extra viscosity, the so-called turbulent viscosity or eddy viscosity, that accounts for the effects of the fluctuating fields (unsolved) on the mean fields. Cause the problem can't be closed, due to these new variables, turbulence models are needed. These models have a set of equations capable to resolve the turbulence. There are different ones and some of them can be visualized in Table 4.1.

n° equations	Turbulence models
1	Spalart-Allmarass
2	k-ϵ
2	kω-SST
4	Transition-SST
7	Reynolds stress model

Table 4.1: Different turbulence models

In this thesys only three method will be used: Spalart-Allmaras (SA), kω-SST and Transition-SST.

Starting from SA, it solves a modeled transport equation for the kinematic eddy (turbulent) viscosity. Due to the fact it only use one equation, it has inability to rapidly accommodate changes in length scale. The kω-SST is a hybrid model between k-ϵ and standard k-ω. The first one is used is the zone full of turbulence while the second one is used in near wall zones because it describes better the viscous strains. Last one, the Transition-SST model is based on the coupling of the SST $k - \omega$ transport equations with two other transport equations, one for the intermittency and one for the transition onset criteria, in terms of momentum-thickness Reynolds number. For a detailed description of the models, the reader is referred to Pope [17].

To describe appropriately the flow near the wall, the near wall Reynolds number is set for every model as $y_w^+ \leq 1$. It is determined as:

$$y_w^+ = \frac{y_w}{\delta_v} \tag{4.22}$$

Here, the y_w indicates the high of the first cell and δ_v is the wall local viscous length. As noted from (4.22), the high of the first cell near the wall is fundamental to achieve the correct y_w^+. So, it is obtained as follows:

$$\delta_v = \frac{\mu_w}{\rho_w u_\tau} \tag{4.23}$$

here, ρ_w is the wall density, μ_w the wall viscosity and u_τ is the friction velocity. Moreover

$$u_\tau = \sqrt{\frac{\tau_w}{\rho_w}} \tag{4.24}$$

here, τ_w is the wall shear stress. Furthermore,

$$\tau_w = \frac{1}{2} C_f \rho v_\infty^2 \tag{4.25}$$

The skin friction coefficient, for a flat plane, is defined as:

$$C_f = 0.026 Re_l^{1/7} \tag{4.26}$$

where Re_l is the Reynolds number based on a characteristic length l and it is described as:

$$Re_l = \frac{\rho v_\infty l}{\mu} \tag{4.27}$$

Noted that $y_w^+ \simeq 1$ and noted the characteristic Reynold number, the high of the first cell is determined from equation (4.22) and the quality of boundary layer resolution near wall is guaranteed.

Chapter 5

Computational setup and numerical methodology

5.1 Introduction

This chapter describes the setup of the computational models, the differences between each of them and the motivation of every change with respect to the experimental model taken into account. While there are some geometry differences, these don't change the accuracy of the solutions but they have to be taken into account to reproduce the results that can be compared to the experimental ones. Finally, it will be chosen a mesh refinement to use during the model optimization.

5.2 Geometry and test description

The geometry is divided in two parts: a bottom one which defines the two ramps and the bottom part of the inner channel; and an upper one which defines the external ramp (cowl lip), the upper part of the inner channel and the external surface. The baseline geometry presents a chord $L_0 = 150 \ mm$ and an angle of 10° for the first ramp and 22° for the second ramp respect to the horizontal axis and a channel located at 21 mm height.
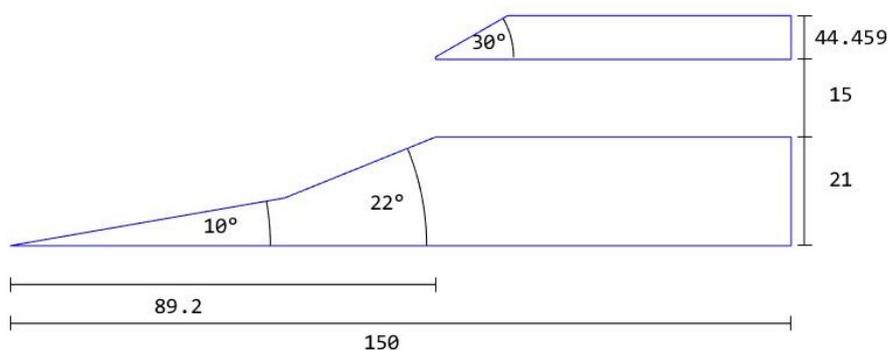
Figure 5.1: Baseline geometry. All the measures are taken in mm

It is noted that the geometry presents some differences in the cowl lip section respect to the experimental one. Here, in order to obtain a normal shock wave, there is a 0.4 mm segment between the upper part of the inner channel and the cowl lip. This segment is connected by a arc of circle to the external ramp, as showed in Figure 5.2. Furthermore,

the cowl lip section starts at the inlet (start of the channel) and its ramp has a 30° angle. The height of this section is 44.459 *mm*.
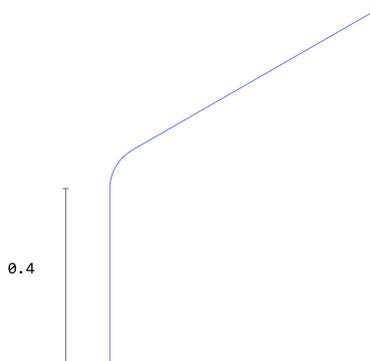
Figure 5.2: Magnification of the cowl lip measured in mm

Despite the changes, the computational geometry can be easily associated the experimental one. The really important factor is to provide a cowl lip ramp with 30° angle. The presence of a finite 'vertex' of the cowl lip provides acceptable variations with respect the experimental results, as will be proven later.

Experimental tests were conducted in the High Supersonic Tunnel at the University of Manchester. The experimental setup, provided by [3], was conducted with precise values of fluid as showed in Table 5.1. The intake presents three pressure sensors to capture unsteady static pressure. The air is stored in a tank and then a valve regulates its pressure. The heaters are used to provide the correct Reynolds number.

Parameters	Symbol	
Freestream Mach	M_∞	5
Total temperature	T_0	390 K
Freestream total pressure	$p_{0\infty}$	820 kPa
Freestream velocity	v_∞	806 $m \cdot s^{-1}$
Reynolds number per unit length	Re_l	$2.3 \times 10^6 \ m^{-1}$

Table 5.1: Freestream experimental parameters

5.3 Software environment

The software ANSYS$^©$ with its package FLUENT is used to analyze the fluid flow. Ansys is a library of different software used for different kind of analyses, such as structural one for example. Ansys provide a common space to be able to interact between different software and to transfer data from different packages. Fluent is the software used for fluid dynamics analysis. It is a well known software in the CFD community for its reliability and so it is largely use. It resolves the fluid equation using the finite volume method [19].

The user has to define to the software the type of geometry (2D or 3D), the number of cores to use in parallel and the working directory. Then, the program starts its GUI (Graphical User Interface) which is divided in different section: in the left side is present

the setting tree that is also present in a different form in the upper side of the software, in the right side there is a graphical windows in which the user can plot different variables and in the bottom side there is the command windows, a box in which the user, through a specific language, can manipulate the simulation. The latter allows the user to work in TUI (Tangible User Interface) with scripts without the need to open the software and setup the simulation step by step manually.
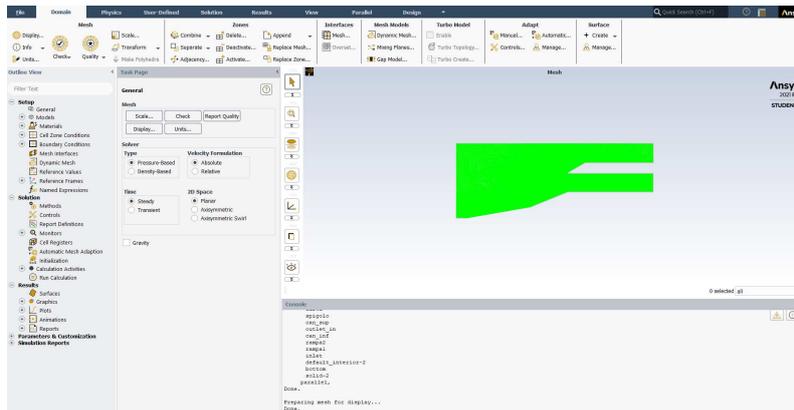
Figure 5.3: Example screen of Fluent

SALOME is an open source platform providing different packages that can be used for different analysis. This software presents two different package that will be use: geometry and mesh. First one creates 2D and 3D geometry of different shape, second one generates computational grids with a wide variety of settings (e.g. structured mesh, local sizes, quad elements etc.).

It presents a graphical interface divided in different parts: a upper section that provides the different packages and the setting of the package in use, a left section where all the changes of the users can be visualized (e.g. in geometry, points with its coordinates are displayed), a middle section where the user can visualize all the graphical objects (e.g. whole body or a section in geometry) and a bottom section in which the user can provide to the software the settings in Python language.
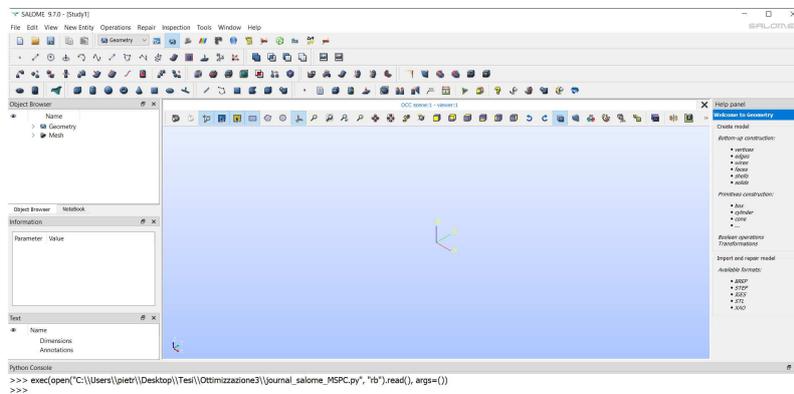
Figure 5.4: Example screen of Salome

MATLAB$^{©}$ is the least software used. It will be use to provide scripts to the previous software, to generate the optimisation process and to display graphically the results obtained by Fluent.

5.4 Numerical settings

5.4.1 Computational fluid domain

Given the geometry of experimental intake, it is necessary to provide a computational fluid domain in order to confine the simulation case. So, in the interest to evaluate the results respect to the computational case given by [6], it is created a similar domain: $L_x \times L_y = (157.5 \times 60)\ mm$.
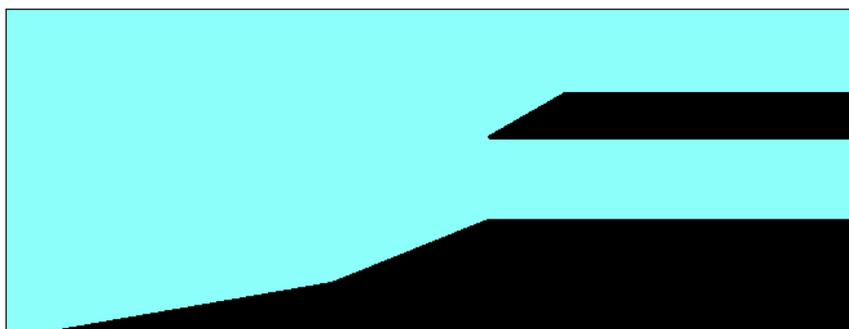
Figure 5.5: Computational fluid domain

The whole body is positioned 7.5 mm from the inlet of the domain to give room to the fluid to evolve itself. The upper boundary is located at 60 mm to provide enough space to the fluid to evolve over the external surface. The right boundary is divided in two zones: an upper one over the intake external surface and one inside the channel; always to compare the results with [6]. In addition, for future use, a surface is defined at the beginning of the canal that joins the lower and upper surfaces of the canal (it is not a material surface); it will be called inlet. The curves of the computational domain and their name are written in Table 5.2.

Geometric element	Coding
Freestream inlet	FREE_INLET
Upper boundary	TOP
Bottom boundary	BOTTOM
Inlet channel	INLET
External outlet	EX_OUTLET
Inner outlet	IN_OUTLET

Table 5.2: Coding of computational domain

5.4.2 Mesh settings

Grids are very important to have a high accuracy in the results and a similarity with the experimental ones. Grids or meshes are divided in two macro categories: structured meshes with rectangular elements and unstructured meshes with triangular elements. First one are the best way to define a grid because they are built following the fluid flow. However, they usually need a support topology to define a proper grid for complex geometry

not having only straight curves.

While the geometry could be divided by rectangular elements, the mesh of the body is defined by triangular ones with a view to using a similar grid in the optimization process because triangular meshes are very flexible . Indeed, the future geometry are not known at priori and so a structured mesh could generate errors with different geometry from the baseline one.

The mesh is not build with equal triangles over the whole computational domain. Through different local sizes, the triangles near the wall are more tiny than the ones far from the walls. In this zone the cell density increase because there is a rapidly change of variables in the flow. These variations can be captured by using more cells in those locations. Furthermore, there is a presence of different inflation layers, increasing from coarse to fine, to enhance resolution near the wall and to give an appropriate value of y_w^+. The aim is to have $y_w^+ < 1$ in all cases to ensure a proper resolution near wall also for all the geometry during the optimization process.

All the simulations are calculate with a number of inflation layer $n > 36$, increasing from coarsest mesh to finest one, and a total thickness of all layers imposes at $2.5e-03\ mm$. With this settings, the recommend threshold for wall-resolved RANS simulation $y+ < 1$ is verified in almost every point of the geometry and so the high resolution near the wall is well respected. This is not true for some points that generate the arc between the normal edge of the cowl lip and the external ramp and for the bottom point of the normal wall that define the cowl lip, as the reader can see in Figure 5.6. Due to the geometry, these points are not so well describe by the grid. Although the mesh is not optimally described in these zones, the $y+$ values are acceptable cause they are just above the unit.

(a) Corner

(b) Arc
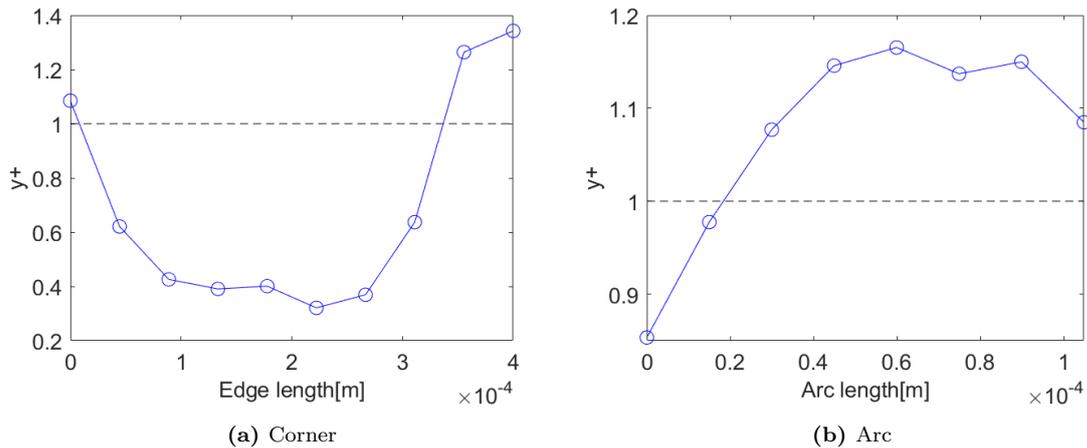
Figure 5.6: $y+$ values on normal wall and arc of the cowl lip section

Three different meshes are used to evaluate the mesh independence of the model. This analysis aim to evaluate the most appropriate grid level with respect to computational time, accuracy and independence of the solution. Three different levels of grid are used: a refinement of the mesh is present due to the decrease of the values of local size and maximum size from the coarse mesh to the finest one.

	Coarse	Medium	Fine
Max Size [m]	0.001	0.00068	0.00042
Growth Rate	0.0076	0.0046	0.0045
Ramp 1 [m]	0.0004	0.00015	0.00015
Ramp 2 [m]	0.00025	0.0001	0.0001
Channel Bottom [m]	—	0.0004	—
Channel Upper [m]	—	0.0004	—
External Ramp [m]	—	0.0005	—
Viscous Layers	36	36	38

Table 5.3: Settings for different kind of mesh

5.4.3   Boundary conditions

Boundary conditions are necessary to execute a simulation. They (see Figure 5.7 for a full view) are so defined:

- *pressure far-field* (Top and free_inlet):

    - $M_\infty = c_\infty/u_\infty = 5$ here the c_∞ is the undisturbed speed of sound and u_∞ is the free-stream velocity;
    - $p_{tot\infty} = 820\,kPa$ so it is imposed a $p_\infty = 1549.83\,Pa$;
    - $T_{tot\infty} = 390\,K$ so it is imposed a $T_\infty = 65\,K$.

- *pressure outlet* (Ex_outlet):

    - *Backflow Direction Specification Method*: From Neighboring Cell
    - *Backflow Pressure Specification*: Static Pressure
    - *Backflow Total Temperature*: 390 K

- *pressure inlet* (In_outlet):

    - *Gauge Total Pressure*: 308000 Pa
    - *Supersonic/Initial Gauge Pressure*: 15400 Pa
    - *Total Temperature*: 390 K

- *symmetry* (Bottom): the fluid is symmetric respect to horizontal axis, thus this condition can be summarized as follow:

    - zero normal velocity at a symmetry plane;
    - zero normal gradients of all variables at a symmetry plane.
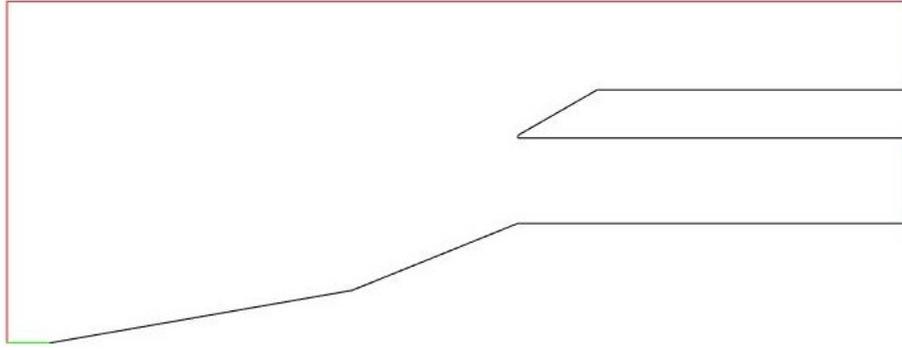
Figure 5.7: Fluid domain and boundary conditions divided by colors: Pressure far field in red, all the walls in black, symmetry in green, pressure inlet in cyan and pressure outlet in blue

Pressure inlet is the condition imposed in the inner outlet, the flow exit of the channel. A previous simulation with the canonical condition 'pressure outlet' was set but the residuals didn't converge under values of 1e-05. However, the results of the simulation are used to define the gauge pressure for the pressure inlet condition. So, the flow is helped in developing within the channel and the residues fall to acceptable values.

In the top boundary, the pressure far field condition is used. As there is little distance between the external surface and the top boundary, a symmetry condition should be more suitable. However, there is a production of shock waves reflected from the upper limit which do not reflect the results of the experiment. So, a pressure far field condition is chosen for this case.

Note that the inflow must have a free-stream Reynolds number set to $Re_{L0} = 2.3 \cdot 10^6 \ m^{-1}$, where $Re_{L0} = \rho_\infty u_\infty / \mu_\infty$, so the viscosity is equal to $\mu_\infty = 4.37621E - 06$. It is imposed also a turbulence intensity of 1% at the inflow. This computational setup replicates the experimental setup of [3].

5.4.4 Model setup

Numerical scheme coupled is used in the simulation: it works coupling the solution of the equations of the model. Also, Least Squares Cell-Based approach is employed for gradients' reconstruction, that is the way in which the derivatives of the various quantities are estimated. The series expansion of the various quantities are truncated to a certain order of magnitude chosen by the user. In this case, the Navier-Stokes equations' spatial components are computed using 1st-order Upwind scheme.

Regarding the last point, the usual procedure is to start the simulation with a 1st-order scheme, to reach a first solution with a good convergence and then to restart the simulation with a 2nd-order scheme and the result of the previous analysis as a starting point. This procedure allows more accuracy and more numerical stability. This is not true for this case: the simulation doesn't reach convergence and so the results are not acceptable. In order to have a solution for a 2nd-order Upwind scheme, it's necessary that the whole geometry is immersed in the flow, thus it can be implemented only in different kind of meshes, where there is enough space over, under and behind the whole body so the residuals can go down and thus the solution can converge. So, the 1st-order scheme was accepted even if its results are not optimal in computational term.
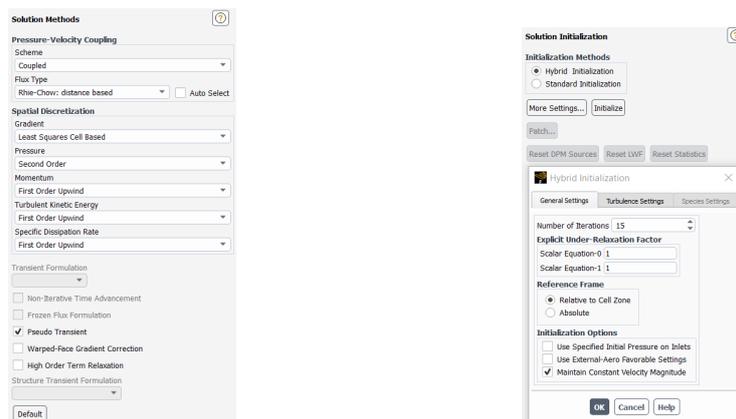
Figure 5.8: Numerical scheme (left) and setup initialization (right) of fluent

As previous stated, the convergence is a important factor to evaluate the results obtained from the simulation. Convergence is the difference between the (i+1)-th solution and the previous one. When this difference is under a determineted value, the simulation stops and the results are accepted. Usually, the convergence is set when residuals go down under 1E-05 but, due to the fact that it is used a 1st-order scheme, the convergence is set to 1E-08.

Before starting the simulation, it is necessary to initialize it by providing the first attempt values. A hybrid initialization has been used for this calculation model. It solves Laplace's equation to determine the velocity and pressure fields. All other variables will be automatically patched based on domain averaged values or a particular interpolation recipe (more informations are provided in [2]).

The simulations run the compressible RANS system of equations and three models are applied to the three different meshes. The one-equation Spalart-Allmaras (SA) model is used as the fastest model available, also the two-equations k-ω Shear Stress Transport (KWSST) model and the four-equations Transition SST (TSST) model are used for all the meshes. The first one is used because it develops the flow faster than the other two models, the third one is more precise than the others and the second one is a compromise between accuracy and time.

5.5 Mesh sensitivity analysis

In this section the results for every mesh are displayed in order to analyze a possible mesh sensitivity. All the data are so presented:

1. mesh visualization with magnification on the critical zones;

2. $y+$ values for the ramps and the upper and bottom surface of the channel;

3. C_P and C_f on the ramps and the bottom surface of the channel;

4. local, medium and SET pressure and drag coefficient on the ramps.

The first and the second point are compared with those obtained from [6] and [5] to verify the correctness of the simulation. Fourth point shows the values expected from the SET (see chapter 2.4.3) and the values produced by the models.

5.5.1   Coarse mesh

Figure 5.9: Overview of coarse mesh

(a) Ramps and bottom channel

(b) Upper channel
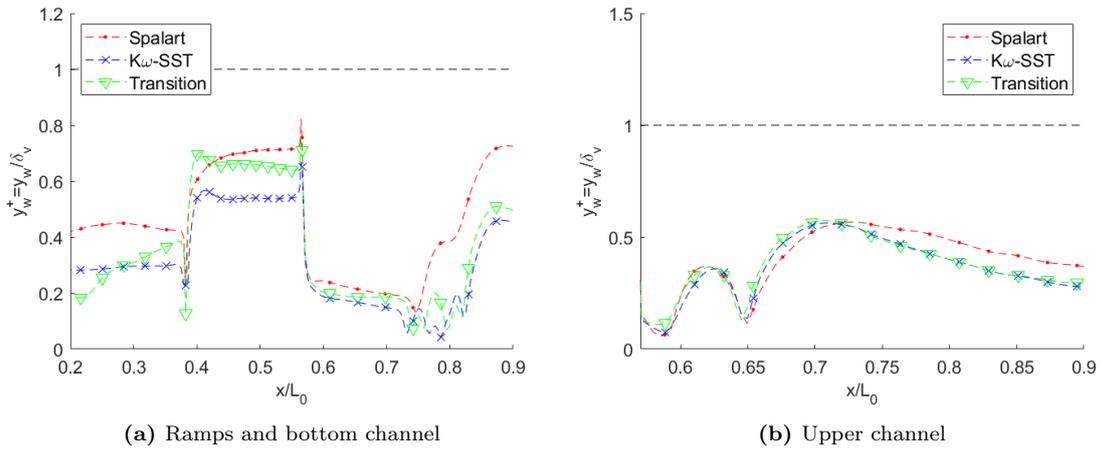
Figure 5.10: $y+$ values with different turbulence models

(a) Skin friction coefficient

(b) Pressure coefficient
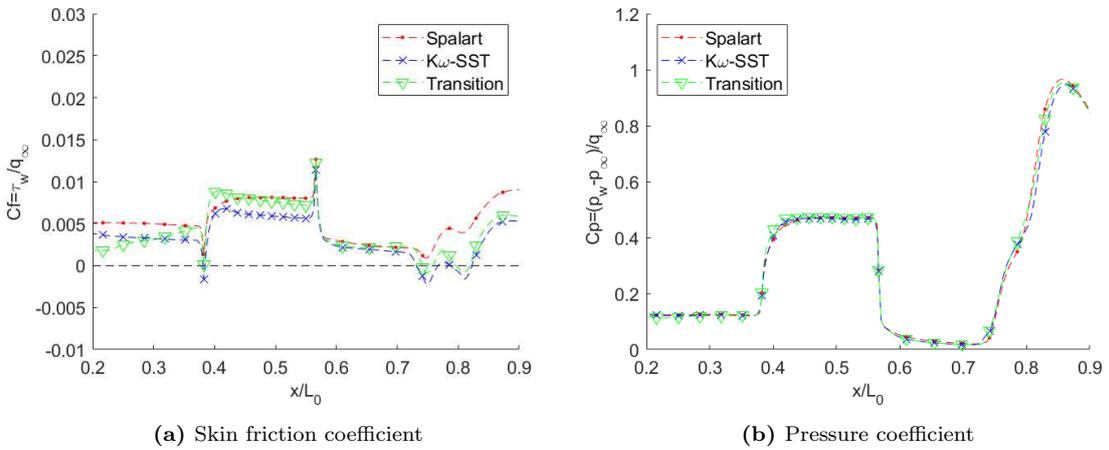
Figure 5.11: C_f and C_P values with different turbulence models on ramps and bottom channel

49

(a) First ramp

(b) Second ramp
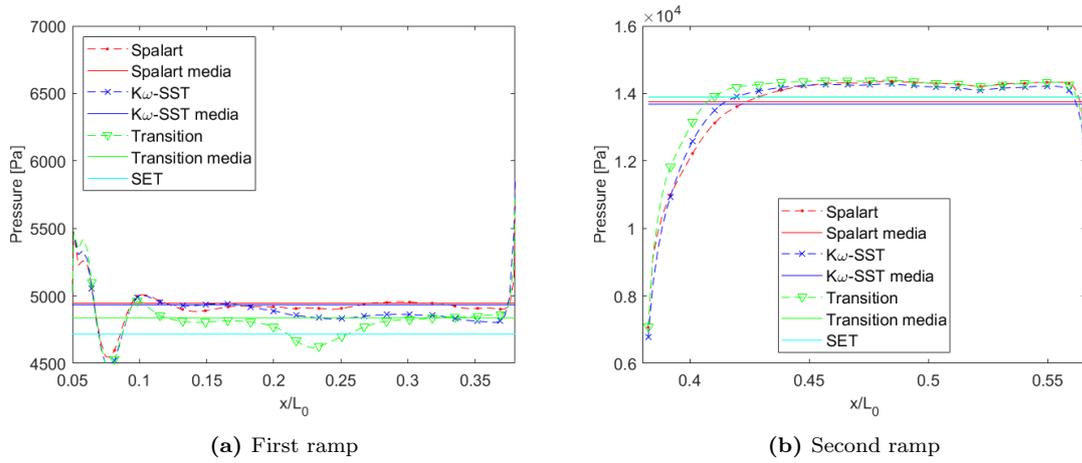
Figure 5.12: Local, medium and SET pressure for different turbulence models

Turbulence models	Ramps	Fluent drag [N]	Average fluent drag [N]	SET drag [N]	Fluent Cd	Average Fluent Cd	SET Cd
Spalart-Allmaras	1	45.992	45.861	43.858	0.011926	0.011281	0.010788
	2	160.03	160.68	162.62	0.039364	0.039524	0.040001
kω-SST	1	45.549	45.5	43.858	0.011204	0.011192	0.010788
	2	159.73	160.41	162.62	0.039291	0.039456	0.040001
Transition SST	1	44.95	44.881	43.858	0.011057	0.01104	0.010788
	2	162.39	163.07	162.62	0.039944	0.040111	0.040001

Table 5.4: Local, medium and SET drag and drag coefficient for different turbulence models

50
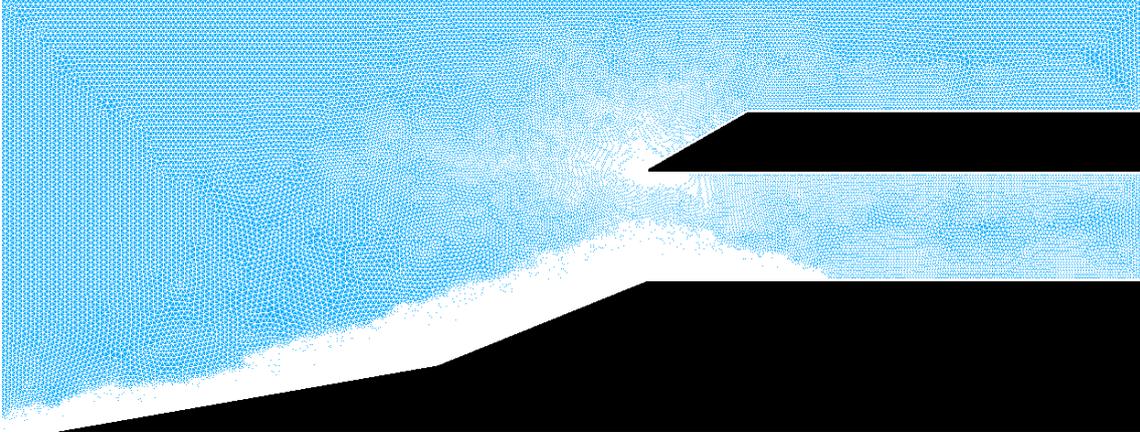
5.5.2 Medium mesh

Figure 5.13: Overview of medium mesh

(a) Ramps and bottom channel

(b) Upper channel
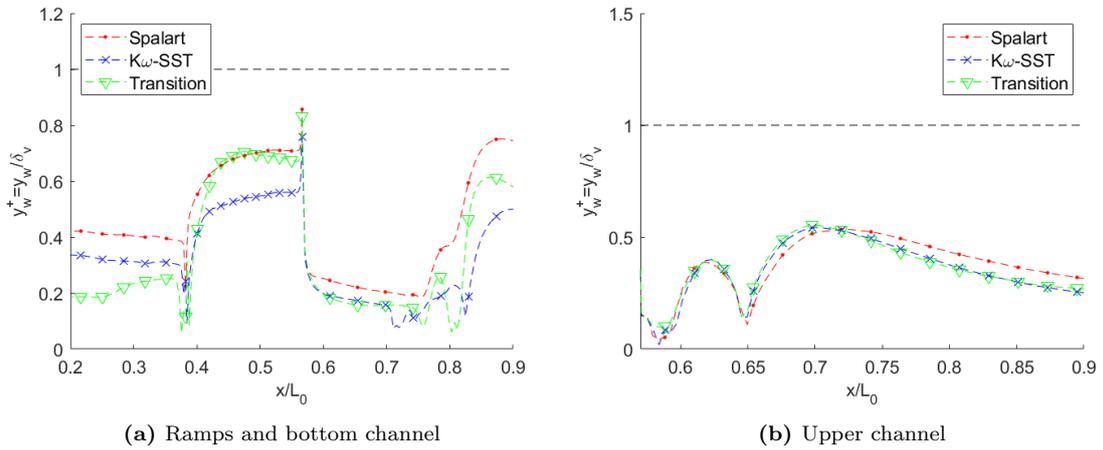
Figure 5.14: $y+$ values with different turbulence models

(a) Skin friction coefficient

(b) Pressure coefficient
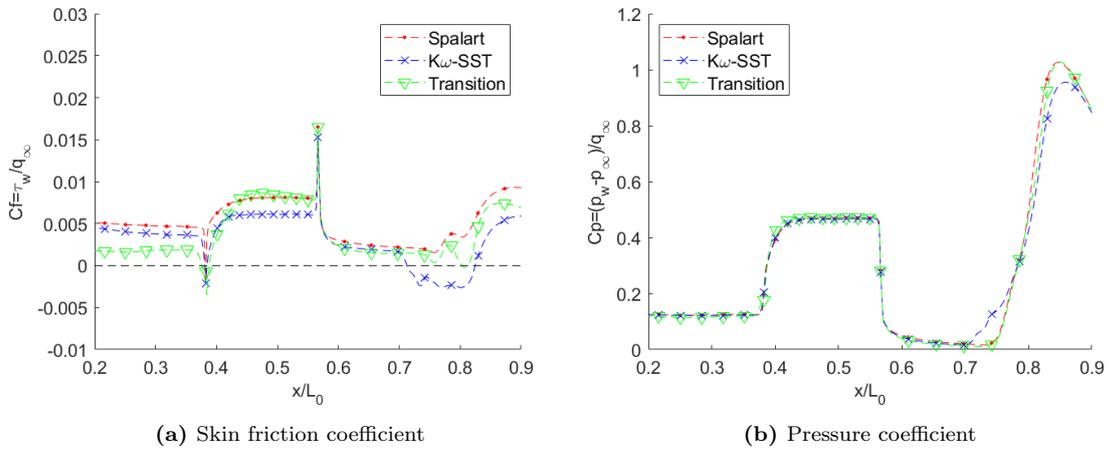
Figure 5.15: C_f and C_P values with different turbulence models on ramps and bottom channel

(a) First ramp

(b) Second ramp
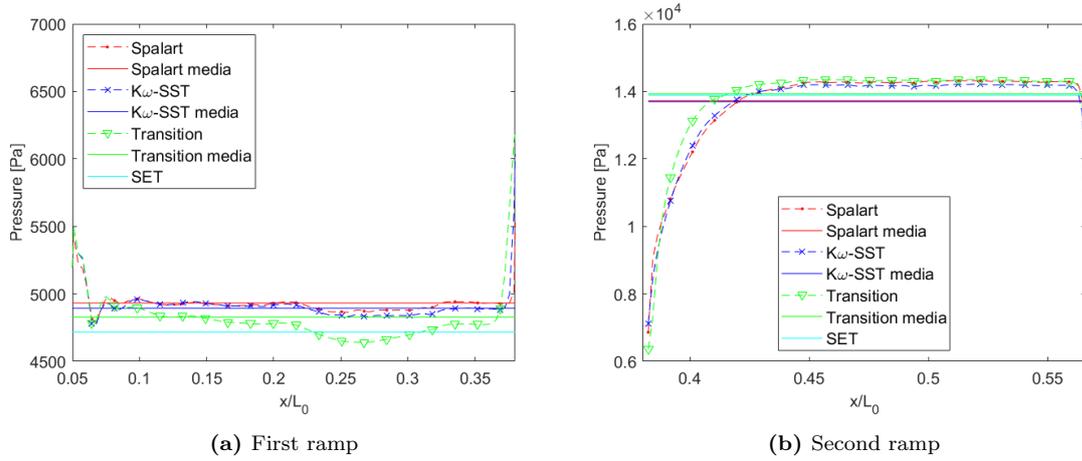
Figure 5.16: Local, medium and SET pressure for different turbulence models

Turbulence models	Ramps	Fluent drag [N]	Average fluent drag [N]	SET drag [N]	Fluent Cd	Average Fluent Cd	SET Cd
Spalart-Allmaras	1	45.956	45.861	43.858	0.011304	0.011281	0.010788
	2	159.97	160.68	162.62	0.039348	0.039524	0.040001
kω-SST	1	45.852	45.5	43.858	0.011278	0.011192	0.010788
	2	159.19	160.41	162.62	0.039158	0.039456	0.040001
Transition SST	1	44.936	44.881	43.858	0.011053	0.01104	0.010788
	2	161.56	163.07	162.62	0.03974	0.040111	0.040001

Table 5.5: Local, medium and SET drag and drag coefficient for different turbulence models

5.5.3 Fine mesh
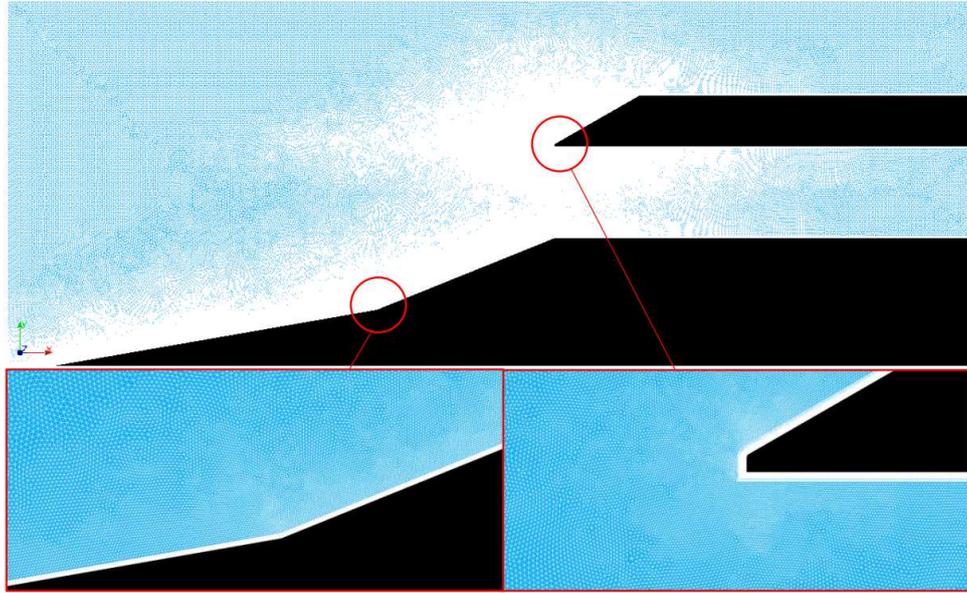
Figure 5.17: Fine mesh in critical zone

(a) Ramps and bottom channel

(b) Upper channel
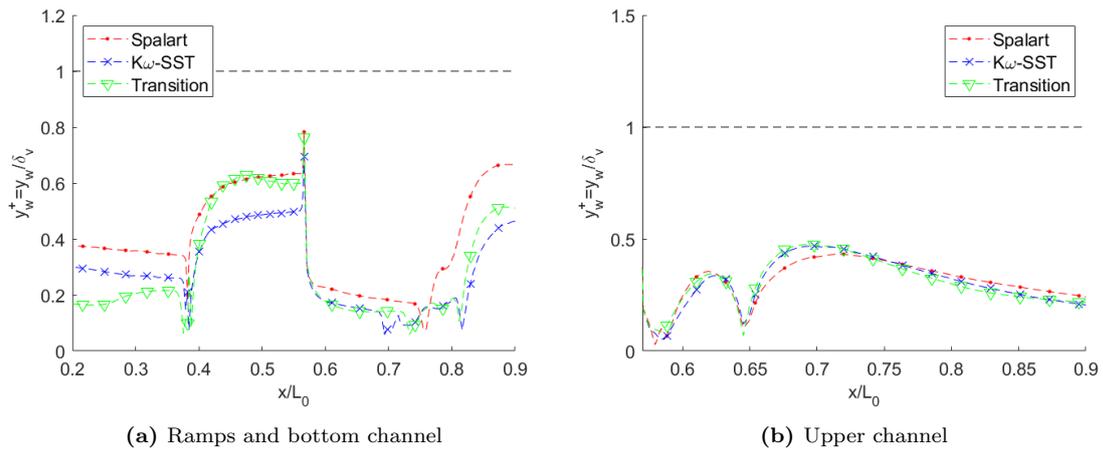
Figure 5.18: $y+$ values with different turbulence models

(a) Skin friction coefficient

(b) Pressure coefficient
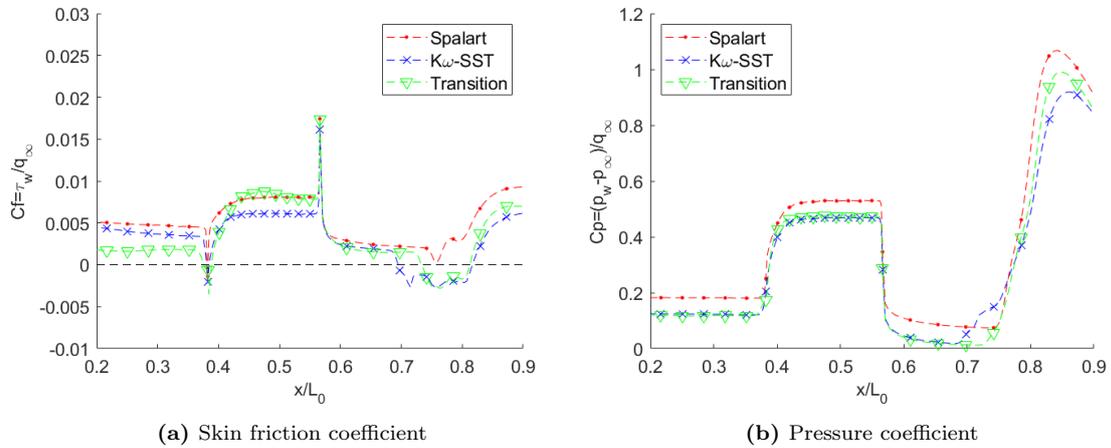
Figure 5.19: C_f and C_P values with different turbulence models on ramps and bottom channel

(a) First ramp

(b) Second ramp
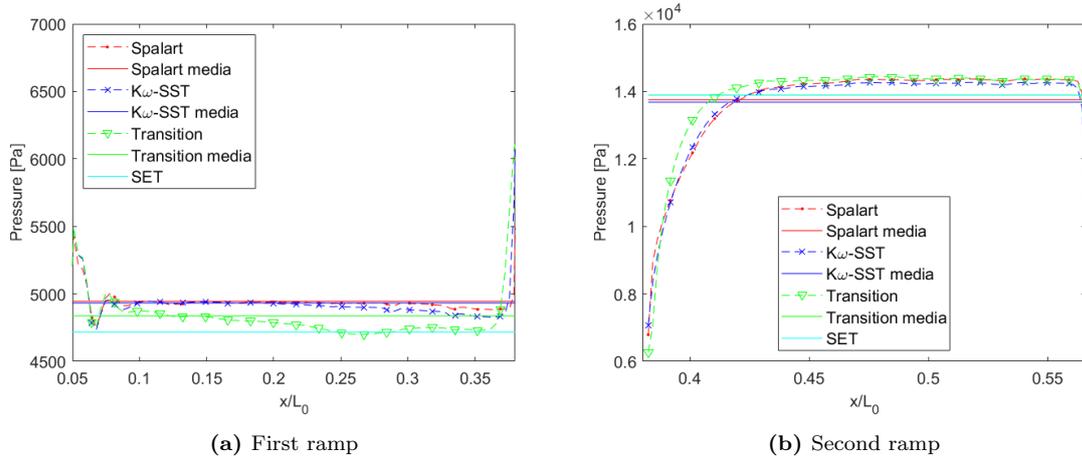
Figure 5.20: Local, medium and SET pressure for different turbulence models

Turbulence models	Ramps	Fluent drag [N]	Average fluent drag [N]	SET drag [N]	Fluent Cd	Average Fluent Cd	SET Cd
Spalart-Allmaras	1	46.042	45.968	43.858	0.011325	0.011307	0.010788
	2	160.37	161.03	162.62	0.039447	0.03961	0.040001
kω-SST	1	45.945	45.862	43.858	0.011302	0.011281	0.010788
	2	159.53	160.18	162.62	0.039242	0.039401	0.040001
Transition SST	1	45.028	44.973	43.858	0.011076	0.011062	0.010788
	2	161.96	162.68	162.62	0.03984	0.040015	0.040001

Table 5.6: Local, medium and SET drag and drag coefficient for different turbulence models

5.5.4 Model validation and baseline analysis

First of all, the drag coefficient $C_D = D/(q_\infty h)$ with $q_\infty = 0.5\rho_\infty v_\infty^2$ undisturbed dynamic pressure and h the intake channel height, and the flow rate scaled to coarse mass $\mathring{G}^* = \dot{m}/\dot{m}_{coarse}$ with $\dot{m} = \rho u h$ are parameterized with respect to the turbulence model for all the three grids.
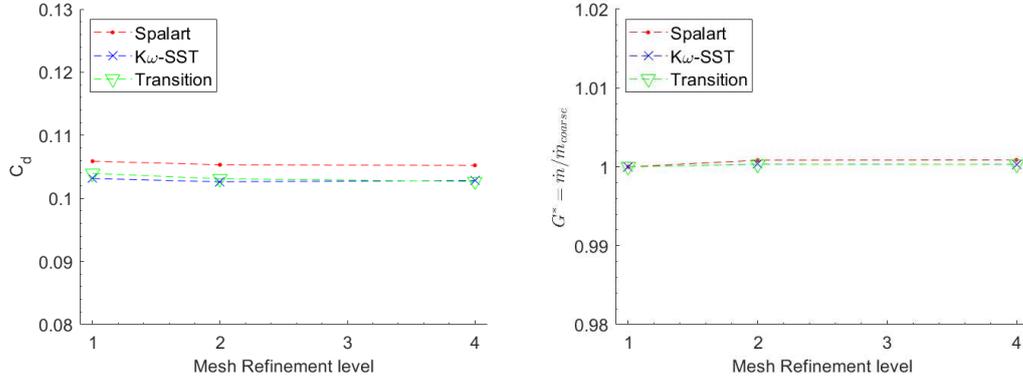
Figure 5.21: Drag Coefficient and Mass flow for all turbulence models and for every grid refinement

As expected, Spalart-Allmaras model generated a different value of C_D with respect to the other two models. This is due to the use of a singular equation to describe the entire fluid flow that causes a coarse description of the flow. There is not a clear mesh dependency since the drag is constant in all the grids. Furthermore, it is noted that the flow mass rate is well captured in all simulations since there aren't differences in either the turbulence models or the mesh refinement. Finally, from the results of all analysis reported in Figure 5.21, there is not a clear mesh dependency.

The inflation layers are sufficient to describe well the flow near walls. All $y+$ are under the requested value of 1 for all the meshes and the models. The trends are very similar and independent through all the grids.

Spalart-Allmaras doesn't display any recirculation bubble for any mesh: neither between the ramps or in the starting section of the channel the skin friction assumes negative values as can be seen in the three Figure 5.11.a, 5.15.a and 5.19.a. The bubble between the two ramps is very tiny in coarse mesh for KWSST and TSST turbulence models, while the other bubble becomes larger with the increasing of the mesh refinement. Furthermore, its shape is always bigger for the KWSST model. The trends is similar through all the meshes.

As the reader can see in Figure 5.11.b, the trends of the three models are overlapping in the coarse grid. In the medium one, there are different values in opening section of the channel ($x = 0.11 \div 0.13$ of Figure 5.15.b) only for the KWSST model while, in the finest one in Figure 5.19.b, the Splarat-Allmaras model generates larger value respect to the other two models. All the three figure present similar values but it is noted that the coarse mesh presents equal values for all the models and so this is an index that the mesh considered is not optimal but too sparse. Note that the bubble is difficult to represent especially for flows described with RANS technique

The pressures are evaluated on first and second ramp because, due to their geometry and to the fluid flow, they follow closely the SET theory. As stated in chapter 2.4.3, this

theory provides angle of the shock, Mach number, pressure and temperature of the flow after the shock while it is known the angle of the geometry generating the shock and the input variables (temperature, pressure and Mach). The pressures of all the meshes and all the models are used to calculate their average values to be compared with the value from SET theory. TSST turbulence model present a large variation along the first ramp respect to the others two models. It is noted that the SET theory provides, in all cases, the minor value respect to the turbulence models. Regarding the second ramp, the situation is the opposite of the first ramp: TSST present the higher values while the SET's ones are higher than those of the turbulence models. For both ramps, there is not a large variation of trends of all the models through the meshes. Noted the pressure values on the two ramps, it is generated a script providing drag and its coefficient along the two ramps. The values presented in Table 5.4,5.6 and 5.5 are evaluated using local and average values of the three turbulence models and they are compared with the theory values.

There is not a clear mesh dependency from the consideration made so far. A more in-depth analysis of how the flow is acting is necessary to evaluate the appropriate grid in terms of calculation time and solution accuracy. So, Mach field of the three grid with the turbulence models are displayed. As the reader can view from Figure 5.22, the mach field is different from the two sparsest mesh and the finest one. The latter presents the stream lines similar to those tested by [3]: there is a normal shock in the start of the channel, phenomenon that the other two grids cannot catch. Moreover, the finest mesh presents a recirculation bubble thinner than the others two and the intersection of the normal shock with the two shocks defined by the two ramps is well defined. So, it has been chosen the mesh with 300k elements because provide variables fields similar to the experiment and the computational time it is not high.

(a) Spalart-Allmarass coarse
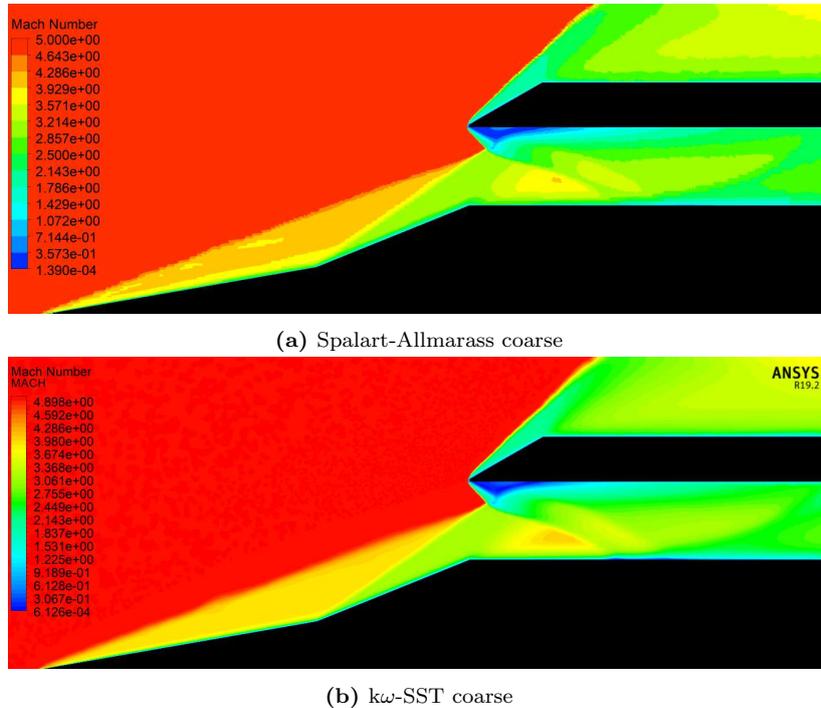
(b) kω-SST coarse

Figure 5.22: Different models for different meshes
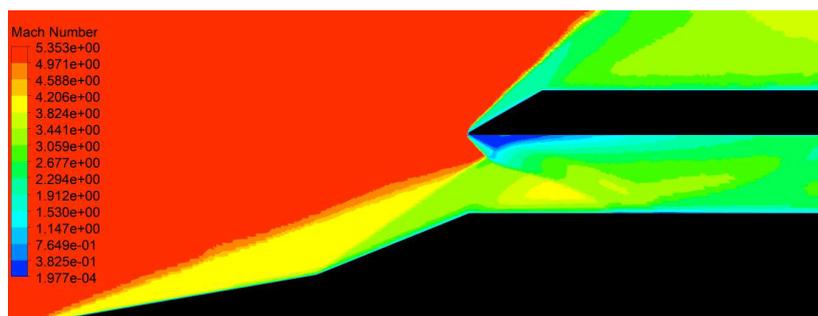
(c) Transition-SST coarse
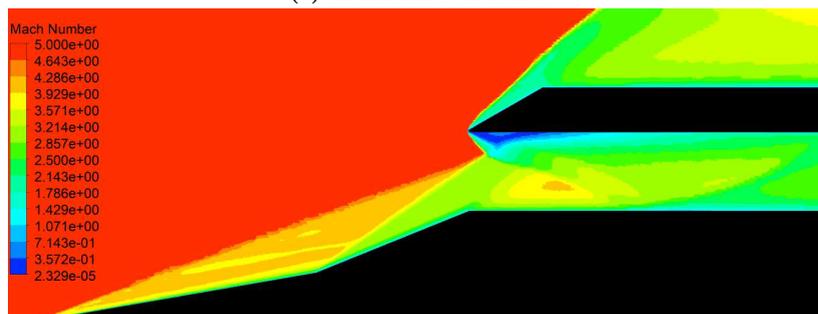
(d) Spalart-Allmarass medium
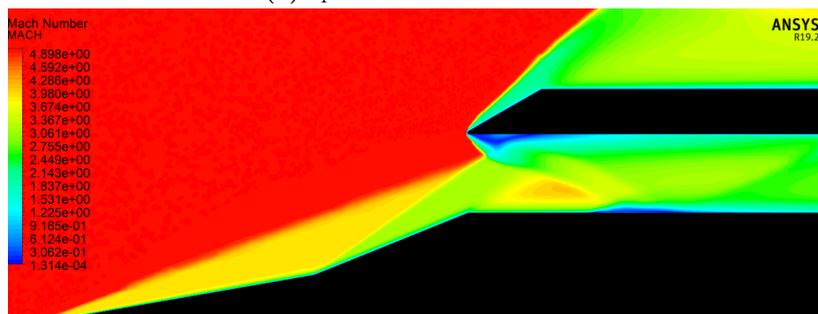
(e) kω-SST medium
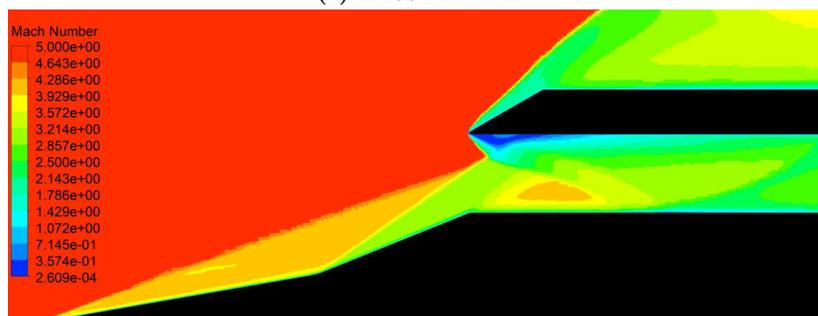
(f) Transition-SST medium
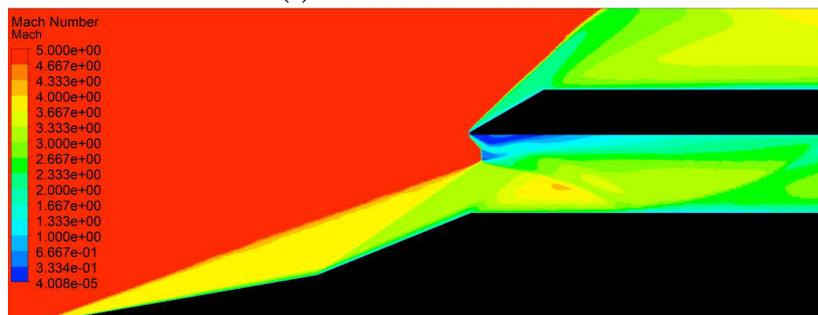
(g) Spalart-Allmarass fine

Figure 5.22: Different models for different meshes (cont.)

57

(h) kω-SST fine

(i) Transition-SST fine
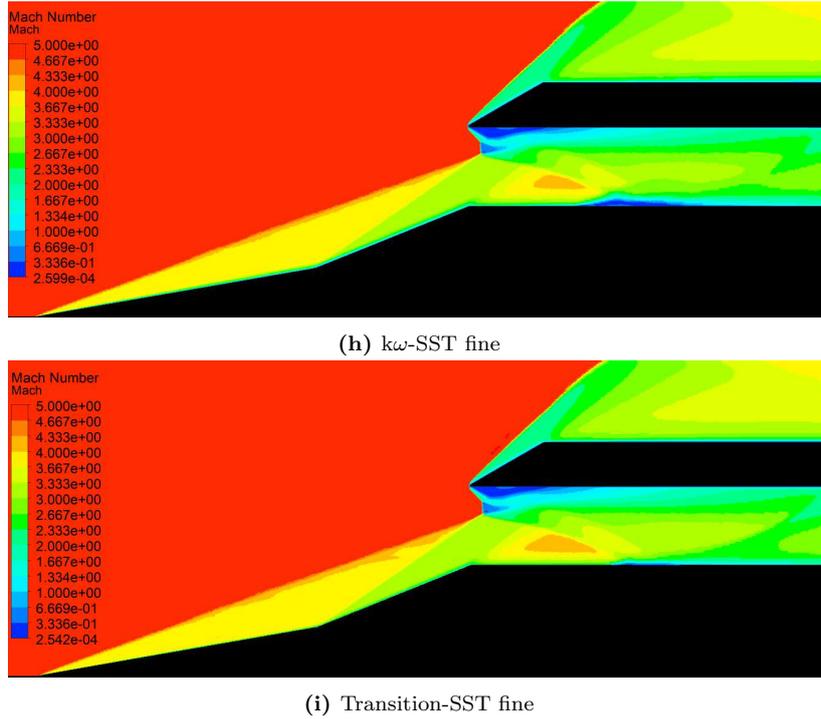
Figure 5.22: Different models for different meshes (cont.)

Moreover, there are a sufficient number of points to resolve the thermal and velocity gradients near the wall but it is chosen to use the finest grid because it's not known how many layers are required to have near the wall to have an high resolution and also the optimization procedure could generate different wall conditions.

Despite some slight differences, all the models provide same trend and values confined in a very narrow band. This shows a independence to the numerical models but a dependency to RANS numerical technique.

Futhermore, mach and density analysis can be made. As show by the results in Figure 5.23, the flow between the three different models is almost similar: the major differences are the position of the bubble, its shape and its magnitude. This bubble is located differently from the model of [6] and its magnitude is lower also in kω-SST model, the model that have the largest one. This flow recirculation is generated by the fan at the entrance of the channel and the interaction between the shocks, generated by the lower ramps and the cowl lips, and the boundary layers. A second recirculation bubble is located between the two ramps due to the interaction between shock and boundary layer. Its magnitude and shape are very tiny so it can be only appreciate through numerical values in figure 5.19: in particular, it is noted that the bubble is located at $x \simeq 0.06$ where C_f assumes negative values.

Regarding the model choice, despite the flow is well captured from all the turbulence model, the $k\omega - SST$ model predict better a sudden expansion at the channel inlet and so well represent the recirculation bubble in magnitude and shape respect to the experimental flow [3]. Also, this model offers a good computational time due to the fact that it use only 2 equations.

(a) Spalart-Allmaras density field

(b) Spalart-Allmaras Mach field

(c) k$\omega - SST$ density field

(d) k$\omega - SST$ Mach field

(e) Transition SST density field

(f) Transition SST Mach field
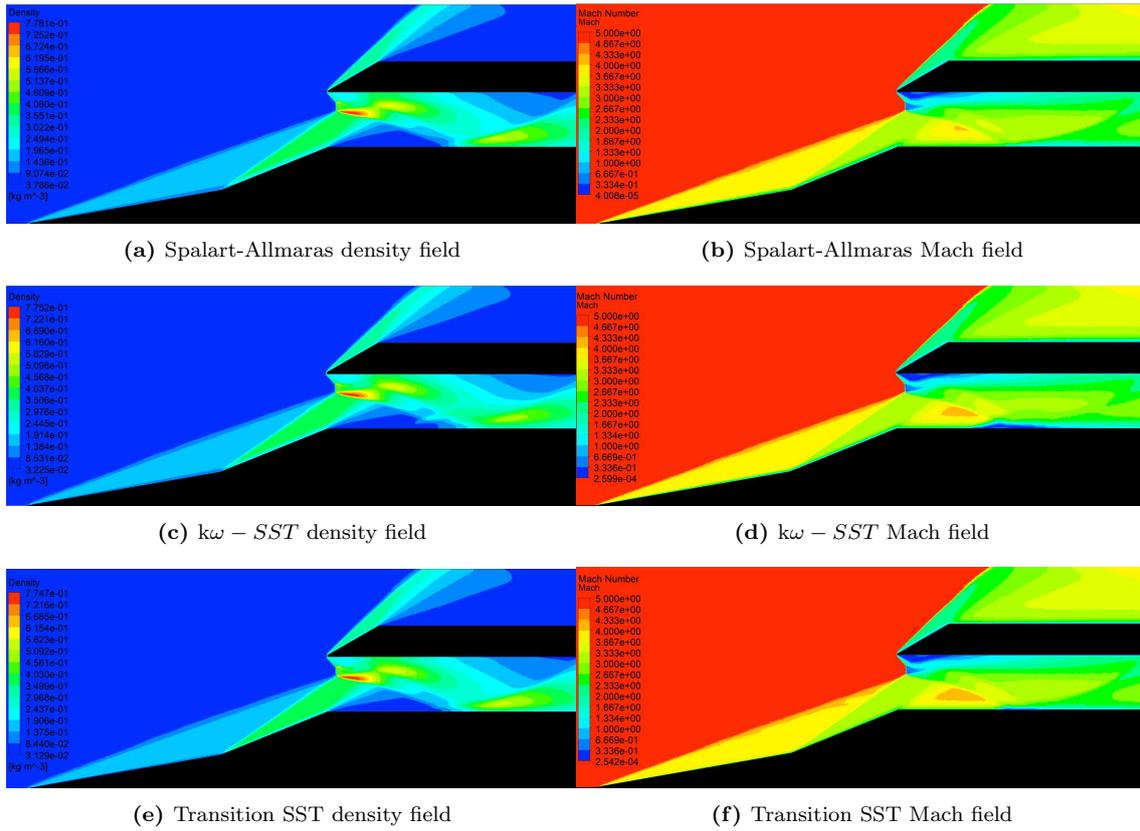
Figure 5.23: Density and Mach fields for different turbulence models for 300k mesh

In conclusion, the optimisation process will define grids with setting similar to the one that we previous called as fine and the flow will be calculated using the two-equation model $k\omega - SST$.

Chapter 6

Optimisation concept and procedure of hypersonic intake

6.1 Introduction

Noted the baseline geometry and its flow characteristics, the cowl lip section is modified to provide a better flow in terms of drag and pressures of the entire intake. A fluid dynamics optimisation procedure is executed. This chapter provides elements of optimisation and its application on the study case.

6.2 Concept of optimisation

6.2.1 Basic notions

Before explaining the definition of optimisation and its procedure, it is necessary to define some basic elements of mathematical analysis:

- **Maximum and minimum of a function**: Let $f : A \subset \mathbb{R}^n \longrightarrow \mathbb{R}$ be a function, with A closed and limited set of $\mathbb{R}^n$. $X_0 \in A$ is a maximum or a minimum for f if:

$$f(x_0) > f(x) \ \ \forall x \in A \ \ \text{and} \ \ f(x_0) = \max\{f(x)\} \tag{6.1}$$

$$f(x_0) < f(x) \ \ \forall x \in A \ \ \text{and} \ \ f(x_0) = \min\{f(x)\} \tag{6.2}$$

  Futhermore:

$$\max\{f(x)\} = \min\{-f(x)\} \tag{6.3}$$

- **Optimisation problem**: Let f be a function, then an optimisation problem defines x_0 values such that $f(x_0)$ is maximum or minimum in its definition set. Here, f is called *objective function*, x is called *vector of decision variables* and x_0 defines the optimal variables. So, a optimisation algorithm is a sequence of different operations necessary to solve an optimization problem.

- **Constrained and unconstrained problems**: if the optimisation problem resolves the maximum or the minimum of the function in the whole closed and limited set A, it's called unconstrained problems. If the optimisation problem researches the maximum or the minimum of the function in a region defined by a set of constrains k described by the functions $g_i \leq 0$ for $i = 1, k$ of the set A, it is called constrain problem.

- **Mono and multi objective problem**: if an optimisation problem regards only one objective function f, it is called mono-objective problem; if an optimisation problem regards more than one objective function f, it is called multi-objective problem.

The previous points express the basic notions necessary to understand the optimization process used in this thesys. It is therefore understood that an optimization problem is a series of operations used to derive the minimum or maximum of an objective function. However, in engineering applications, the values to be obtained are constrained in a set defined by the physical models of the problem under consideration, hence we speak of constrained problem. Moreover, the present case, as will be seen below, considers the minimization of different functions, so we speak of multi-objective problem. Putting together the notions just listed, we can give the following definition:

Let $\mathbf{F}(x) = (f_1(x), \ldots, f_m(x))$ be the vector function that summarizes m scalar objective functions of an optimization problem and let $\mathbf{G}(\mathbf{x}) = (g_1(x), \ldots, g_k(x))$ be the vector function that summarizes k constrain conditions; the multi-objective optimisation problem is so defined:

$$\min = \begin{cases} F(x) \\ G(x) \leq 0 \end{cases} \tag{6.4}$$

However, for a multi-objective function there might be a case where an optimal solution x_1 for the function f_1 is not optimal for the function f_2. In 'mathematical' terms, the problem cab be summarized as follow:

$$f_1(x_1) < f_1(x_2) \text{ while } f_2(x_1) \leq f_2(x_2) \tag{6.5}$$

To avoid this problem, it is defined the Pareto's set. It is the set of solution resulting not-worst than the others in relation to objectives set. A more complete definition can be given as follows: let x_1 and x_2 be two possible solutions for minimal conditions, x_1 dominates x_2 ($x_1 \preceq x_2$) if, called $f_1, \ldots, f_m$ objective of the problem, $\forall j = 1, \ldots, m$

$$f_j(x_1) \leq f_j(x_2) \text{ and } \exists k \in \{1, \ldots, m\} \text{ such that } f_k(x_1) < f_k(x_2) \tag{6.6}$$

The image of the set of dominant solutions according to $\mathbf{F}$ is also known as Pareto's front.

6.2.2 Genetic algorithm

Genetic algorithms are some of the most commonly used algorithms for optimizations. They allow you not to have to manipulate the objective functions and to obtain solutions that can be compatible with the constraints imposed. In addition, they are able not only to generate different solutions but classifying them into better and worse and discarding the latter. On the surface such solutions may seem random but contain a search logic able to make them efficient and little expensive in terms of computational time.

The genetic algorithm is a method for solving optimization problems that is based on natural selection. For a initial set of solutions (or individuals) called initial population, which are random generated, a new set of solutions is created and they are called generations. The latter are generated each step by the parents, the individuals randomly chosen

by the previous generation. However, the numerical recombination aren't totally random but they follow the Darwin's evolutionary theory: the best individuals survive while the worst are discarded. In particular, The genetic algorithm uses three main types of rules at each step to create the next generation from the current population: selection rules, crossover rules and mutation rules.

Before setting out the three rules, it is necessary to define the initial population: it is a set possible solution at the iteration zero, randomly generated and composed by m individuals. So, it is created a matrix where the rows are the individuals and the columns are the dimension of the problem. The matrix can be define as follows:

$$P_0 = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ \vdots & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ x_{m1} & x_{m_2} & \cdots & x_{mn} \end{bmatrix}_0 \tag{6.7}$$

The selection rule creates the couples from the initial population which will generate the future new set of individuals or generations. This rule can use different method to provide the couples, but the most famous one are the roulette method and the tournament method. Both provide the calculation of objective functions for each individual: $f(x_1) \dots, f(x_m)$ called fitness values. Noted the fitness values, the two method use different ways to obtain the couples.

1. *Selection rules*: select the individuals, called parents, that contribute to the population at the next generation. Two method are used:

   - **Roulette method**: to each fitness values is allocated a probability:

     $$p_i = \frac{f(x_i)}{\sum_{i=1}^{m} f(x_i)} \tag{6.8}$$

     Each individual has an higher probability to be chosen by the algorithm if it fitness value is better than others (e.g. if algorithm is searching for the minimum, the fitness with the minimum value has the higher probability in the set). The algorithm picks two individuals to generate a couple.

   - **Tournament method**: it is chosen two random fitness values, they are compared between them and it is chosen the better one that it will be the first individual of the couple. The second one is chosen in a similar way.

2. *Crossover rule*: combine two parents to form children for the next generation. The children are a combination of their parents. It is defined a breaking point: after it the structure of the individual is changed and it is no longer similar to the structure of first parent but it has the structure of the second one. The higher the breaking points, the greater the algorithm's ability to search for new solutions. The breaking point is random but the numbers of breaking points used is usually from one to three.

3. *Mutation rule*: apply random changes to individual parents to form children. This random changes affect usually a small number of individuals ($3\% \div 5\%$)

After the application of this three rules a new set of individuals are generated and their fitness is calculated. Several simulations have led to affirm that the optimal conditions
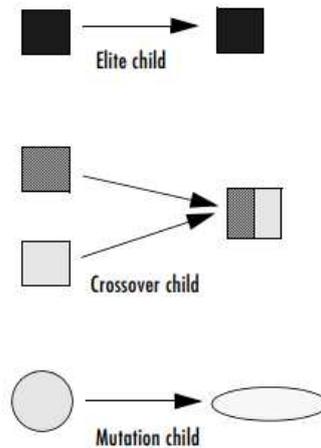
Figure 6.1: Children created bu the three methods

through these rules are reached after a hundred generations. For multi-objective problems, the solutions are ranked respect to the Pareto's front and those which are in the Pareto's front have greater possibility of being chosen.

6.3 Optimisation of an intake

Based of what was discussed earlier, for the optimisation problem is necessary to define the independent set of variables or decision variables $(x_1, \ldots, x_n)$ and the objective function $(f_1, \ldots, f_n)$.

Usually, in optimisation process regarding the fluid diynamics, the independent variables are geometric parameters. In this case, the geometry is an hypersonic intake and so the geometric parameter are the length and the shape of the walls. The objective is to find a geometry with better fluid flow, thus it develops a train of shock waves with minor dissipation and minor detachment of the flow from walls. Moreover, the new intake must have minor drag respect to the baseline geometry

The algorithm tries to minimize the objective function. The latter are evaluated in the baseline geometry and in all new ones created by the algorithm. Their objective functions are compared between themselves and finally a ultimate geometry is generated. This geometry will have the better performance across all objective functions.

To better understand, in the following sections, it will define the parametrisation of the geometry, necessary to define the independent variables, and the definition of the optimisation objectives, necessary to define the objective functions.

6.3.1 Geometry parameters

The aeronautical intakes are often symmetric or axisymmetric respect to an axis. The center of an engine in based on the axis and the whole body (compressor, turbine, nacelle etc.) are built around it. So, for an initial study, their flow can be analyzed through a 2D section. Usually, supersonic and hypersonic sections are divided in two parts: a bottom one that generate the shock waves and a upper one that keeps the shock waves outside

and reduce the drag of the nacelle.

Bezier splines are used to generate the geometry. It uses some points called control points to define a curve. More control points are used, more complex spline can be reconstructed. These points aren't part of the spline, as a normal spline generated by interpolation, but they control the curve at distance. Bezier spline are define with the following equation:

$$x(t) = \sum_{i=0}^{n} \binom{i}{n} (1-t)^{n-i} t^i x_i \tag{6.9}$$

here $x(t) = \{x(t), y(t)\}^T$ denotes the spline coordinates, $x_{i=1}^n = \{x_i, y_i\}_{i=1}^n$ is the spline control points set, $t \in [0:1]$ is the curve parameter and n is the spline order. This spline uses the start and ending points equal to the baseline geometry, while the control points are defines through the previous equation. Better explanation are given in the following sections.

6.3.2 Optimisation objectives

To define the objective function necessary to the algorithm, all the optimisation problems need to visualize the critical problems that the designer would like to improve. An analysis to a general intake is done and then a more specific one for supersonic and hypersonic intake is performed. Given a general view, then the designer can evaluate the corresponding parameters to maximize or minimize.

Generally an intake must be light weight to cost less in terms of fuel consumption and aerodynamic resistance. Also, an intake must provide the engine with adequate mass flow rate, proper Mach number and uniform flow to generate less vortices. They cause mechanical losses, rotating stall instability and high vibration of the pallets of the first stage of the compressor causing high-cycle fatigue. The nacelle generates additional drag due to the difference of pressure between the internal and external flow. They are not the only aspects that could be aim of the design but they are the most important ones.

As previously stated, the supersonic intake also has starting phenomenon problem: the normal shock instability inside a converging ducks. Moreover, there is a constant motion of the shock causing a difference of performance over time. In the specific case of hypersonic intake, concepts to optimize are so described:

- intake must be as less as invasive as possible regarding the overall vehicle drag;

- intake must produce an intense flow compression, with consequent diffusion and slowdown of the stream to the compressor;

- intake with compression trains are preferred to reduce energy losses related to the formation of the shocks.

These concepts are translated in the following objective functions to minimize:

$$f_1 = C_d \quad f_2 = \frac{p_0}{p_1} \quad f_3 = \frac{p_0^o}{p_1^o} \tag{6.10}$$

here f_1 is the drag coefficient, f_2 is the inverse of the Static Pressure Ratio (SPR), the ratio between the area-weighted pressure in the start and in the exit of the channel, and

f_3 is the inverse of the Total Pressure Ratio (TPR), the ratio between the area-weighted total pressure in the start and in the exit of the channel. In order to have only acceptable physical solution, some constrains are used:

- $f_1 \leq 10 \cdot C_d|_{baseline}$

- $f_2 < 1$

- residuals $< 1E - 03$

All solutions outside these ranges are marked with very high values (1000). This allows the research to go to more promising geometry for future generations.

6.3.3 Optimisation loop

An optimisation loop is a set of rules, divided in blocks, that need to communicate between themselves in an iterative way to optimise the geometric parameters for objective functions. When loop terminates, due to the reach of the maximum number of iterations, one or more optimized geometries are obtained. In this thesis, the loop presents communication between various software. A schematic of the loop can be viewed in Figure 6.2.
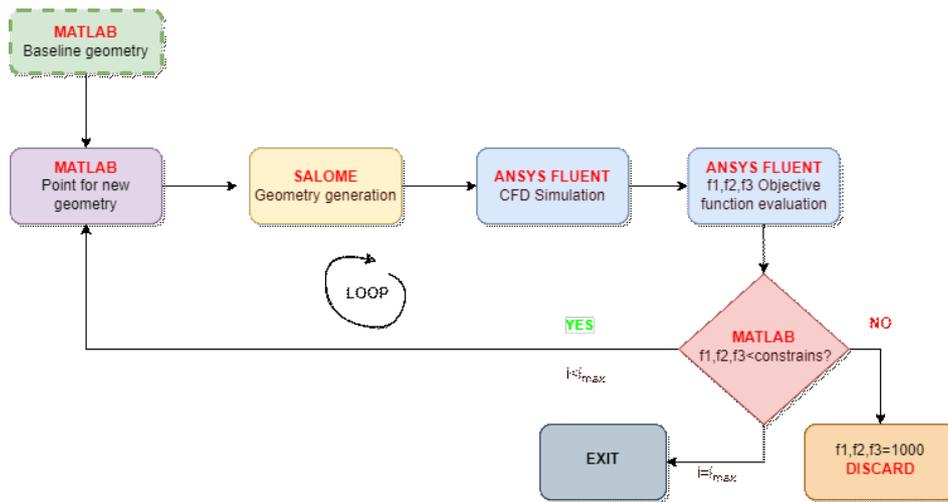
Figure 6.2: Schematic of the optimisation loop

The procedure of optimisation consists of the following steps:

1. A baseline geometry is needed. It is given by experimental data and design process. It is outside the loop because is unique and it represents the starting point.

2. Noted the baseline geometry, a Matlab script generates different points of the cowl lip and the ramps, the parts to be optimized, in the way mentioned in section 6.3.1. A second software, called Salome, creates the geometry and the mesh from the points given by Matlab. This mesh is sent to Fluent. It applies the boundary conditions and all the settings necessary to the CFD simulations.

3. At the end of the simulation, Fluent export the f_1, f_2 and f_3 values to Matlab. The latter compares the values with the constrains. Three case are possible: the objective functions don't respect the constrain (a values of 1000 is given to them), the iteration is the last one and so the loop is terminated or the objective functions respect the constrains.

66

4. If the objective function are acceptable and the iteration is minor than the maximum one imposed by the user, than the genetic algorithm generates different point to simulate and the loop restart from point 2.

This is a conceptual scheme, to deepen in the subject is necessary to define new sections.

6.4 Optimisation of an experimental hypersonic intake

It is necessary to define the control points of the baseline geometry. The latter and its point are generated in a script outside the loop. Noted the points defining the baseline geometry, the script computes an optimisation procedure and it gives the control points used in the loop. For a better explanation, look in the next section.

6.4.1 Baseline with spline reconstruction

As stated before, the control points of the Bezier curve describing the geometry are needed to define the initial population of the main loop scripts. From linear algebra, given a generic curve then it can be described by a linear combination of Bezier curves. Moreover, it is define a unique Bezier curve with n control points and with a minimum Euclidean distance from the generic curve.

So, the script searches for the set of control points which define the Bezier curve with the minimum distance from the baseline geometry. The script is applied to two different section of the intake: the cowl lip and the two ramps. This and the future codes are splitted in different parts to explain better all the steps. The description of the code used to apply the Bezier theorem to the cowl lip section is given below. The ramp section has the same structure.

```
1  outnum = 30;
2  nVar = 6;
3  nInd = 200;
4  nGen = 200;
5  ch = 13.856;
6
7  folder = 'baseline';
8  G = importdata('CoordinateCowlLipBaseline.txt');
9  COWLLIP = [G]
```

Code 6.1: Geometry parameters

Input parameter are defined. In particular: *outnum* are the number of points used to describe the baseline geometry of the cowl lip; *nVar* indicates the numbers of the coordinates (x,y) of control points that the script will extrapolate; *nInd* and *nGen* are the genetic input parameters; *ch* represents the chord of the cowl lip; from line 7 to 9 the script receives the points coordinates from the baseline geometry. These lines give the Figure 6.3.
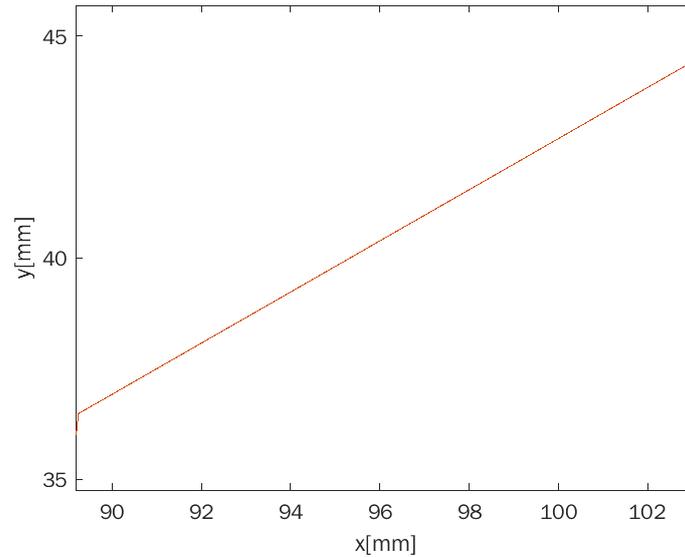
Figure 6.3: Cowl lip geometry

Noted the input parameters, the optimisation script need also the constrains definition (as seen in Section 6.2.2). These are divided in lower (lb) and upper bound (ub), the minimum and maximum values of the variable. The chord is divided in equal parts in number equal to the control points and the boundary are so defined: along the x axis equal segments are used and along the y axis the points can may vary from 34 to 46.

```
10   vert=COWLLIP(:,2);
11   orizz=COWLLIP(:,1);
12   discret=ch/((nVar/2)+1);
13
14   lbx=zeros(1,nVar/2);
15   ubx=zeros(1,nVar/2);
16
17   for  k=1:nVar/2;
18       lbx(k)=orizz(1,1)+(0.5+(k-1))*discret;
19       ubx(k)=orizz(1,1)+(1.5+(k-1))*discret;
20   end
21
22   for  k=1:nVar/2;
23       lby(k)=34;
24   end
25
26   for  k=1:nVar/2;
27       uby(k)=46;
28   end
29
30   lb = [lbx lby]
31   ub = [ubx uby]
```

Code 6.2: Constrains definition

An initial population is initialized and generated with random components inside the constrains. For the optimisation the multi-objective NSGA-II algorithm uses the function called *distance* to obtain the control points and the spline with the minimal distance. Note that the *decvar* is a auxiliary variable, so it stores the controls points. Finally, the points are saved in matrix form and they are used to generate the new curve with the *bezier*

68

function. It is noted that the first and the last points of the spline are the same of the baseline geometry because they will be connect with the rest of the intake geometry.

```
32  initPop = zeros(nInd, nVar);
33  for k=1:nInd
34      initPop(k,:) = rand(1,nVar).*(ub-lb)+lb
35  end
36
37  options = gaoptimset('InitialPopulation',initPop,'Generations',nGen,'
        PopulationSize',nInd)
38  pcCL = ga(@(decvar) distance(decvar,outnum,COWLLIP),nVar,[],[],[],[],lb
        ,ub,[],options);
39  pcCL = reshape(pcCL,[nVar/2 2]);
40  save('ProfiloBaseline\CLpoints.mat');
41  save('pcCL.mat', 'pcCL');
42  save('COWLLIP.mat', 'COWLLIP');
43
44  curveS = bezier([RAMP(1,:);pcCL;RAMP(end,:)],outnum)
```

Code 6.3: Genetic component

In Figure 6.4 the reader can view the baseline curve, the Bezier curve overlapping and the control points generating this last curve. The boundary are limited by the constrains. The control points near the center of each segment and the precise reconstruction performed by the Bezier curve validate the procedure.
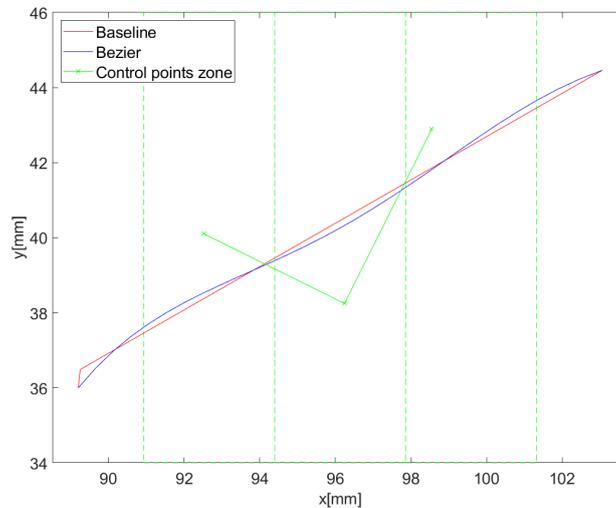
Figure 6.4: Baseline geometry and Bezier curve of cowl lip

A similar process in used to define the Bezier's curve and the corresponding control points describing the two ramps. Only variations are the coordinate points given to the code, the boundary and the chord value ($ch = 81.7$). The results are given in Figure 6.14b.
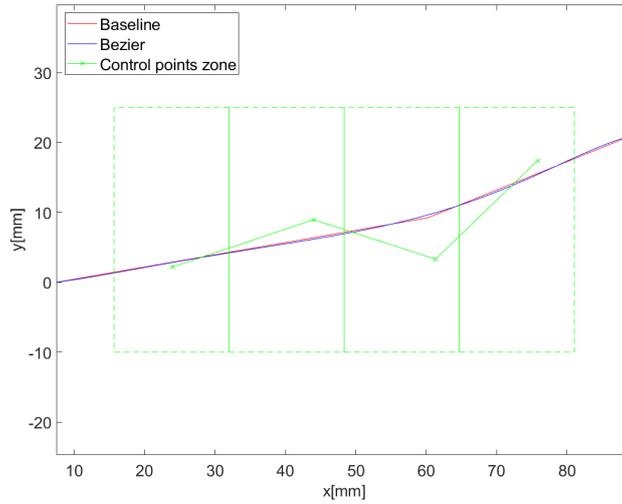
Figure 6.5: Baseline geometry and Bezier curve of ramps

6.4.2 Optimisation loop procedure

In this section will be described all the scripts present in the loop. Optimisation is carried out first for the cowl lip only and then simultaneously for the two ramps. The principle and method are the same but for a better understanding we continue by describing the code for total optimisation of intake. The loop is made by different codes:

- *OptimizationLoop*: the main script ruling in the loop;

- *fitness*: this script allows to exchange data between Matlab, ANSYS Fluent and SALOME. Moreover, it extrapolate the f_1, f_2 and f_3 objective functions Fluent data and save or discard them;

- *journal_salome*: this script creates a journal with the commands (in Python language) readable by SALOME, containing geometry and mesh parameters;

- *journal_fluent*: this script creates a journal with the commands readable by ANSYS Fluent, containing the settings for the fluid flow simulations.

Starting from the mail loop script, the results evaluated from the previous scripts are loaded and defined as *pcCL* and *pcR* variables. As the previous code, it is necessary to define the input parameters for the GA algorithm. Here, *nVar_opt* indicates the number of variable to be optimized, *nInd_opt* indicates the number of individuals for generation and *nGen_opt* indicates the number of generations. Each individual occupies $\sim 10 \div 15$ minutes of time to complete the entire loop, moreover, the number of loops that can be run is given in principle by multiplying the number of individuals by the number of generations (350 in this case). The individuals and the generations number can be arbitrarily define. Their values are defined in order to not have excessive computational time.

Two global variables are defined: *datab* and *counter*. Being global variables means that they can be used not only in the main loop but also in the all scripts presented in the optimisation process. Moreover, the *counter* variable is used to define the different saving folders while the *datab* variable is a structural matrix containing *decvar* and *f* matrix.

The first one includes the decision variables (the coordinates of the points), the latter one includes the values of the objective functions.

```
1  clear variables;
2  close all;
3  clc;
4
5  load ('pcCL.mat', 'pcCL');
6  load ('pcR.mat', 'pcR');
7
8  nVar_opt = numel(pcCL)+numel(pcR);
9  nInd_opt = 5;
10 nGen_opt = 70;
11
12 global counter datab
13 counter = 0;
14
15 datab.decvar=zeros(0, nVar_opt);
16 datab.f=zeros(0,3);
```

Code 6.4: *OptimizationLoop* parameters

The constrains defines a zone where the control points can be evaluated and used by the GA algorithm. The zone regarding the cowl lip section, can be visualized in Figure 6.7. The constrains of the new control points are defined by changing the ones obtained to reconstruct the baseline geometry with a Bezier spline. The zone is slightly unbalanced: the future geometry will have the cowl lip bigger and pointed towards the free-stream inlet than the baseline one. This choice is due to the fact that, for some points, SALOME generates a degenerate geometry: it creates two edges due to twisting of the Bezier curve (in conflict with one edge necessary for Fluent as will be seen later. See Figure 6.6) or it can't generate any mesh.
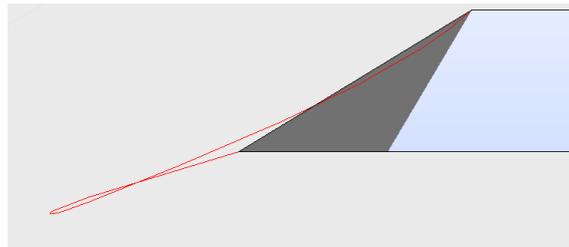
Figure 6.6: Example of error in geometry reconstruction

The control points of the two ramps are also unbalanced. As for the variation along the x-axis, it depends on the point considered. It is symmetrical: the first point has a larger variation along the positive direction than the negative one. Conversely, the last point has a larger variation along the negative side. This type of symmetrical variation is done to obtain points that are not so close to the beginning or end of the ramps, to avoid the generation of high inclinations in these areas and to ensure a smooth transition. The y-coordinates change more in a positive direction to obtain positive bumps. An equal change in height does not generate this type of curve but only a straight line or too small bumps.

The difference in height and length (ch_x and ch_y) between the start and the end of the ramps are used to determinate the boundary of the control points.

```matlab
17  ch_x=81.7;
18  ch_y=21;
19
20  %——Cowl Lip——%
21  for k=1:numel(pcCL(:,2))
22      lbxCL(k)=pcCL(k,1)*0.525;
23      lbyCL(k)=pcCL(k,2)*0.74;
24  end
25
26  for k=1:numel(pcCL(:,2))
27      ubxCL(k)=pcCL(k,1)*0.95;
28      ubyCL(k)=pcCL(k,2)*1.22;
29  end
30
31  %——Ramps——%
32  for k=1:numel(pcR(:,2))
33      lbxR(k)=pcR(k,1)-ch_x/6;
34      lbyR(k)=pcR(k,2)-(ch_y/4);
35  end
36
37  for k=1:numel(pcR(:,2))
38      ubxR(k)=pcR(k,1)+ch_x/6;
39      ubyR(k)=pcR(k,2)+(ch_y/2);
40  end
```

Code 6.5: *OptimizationLoop* constrains
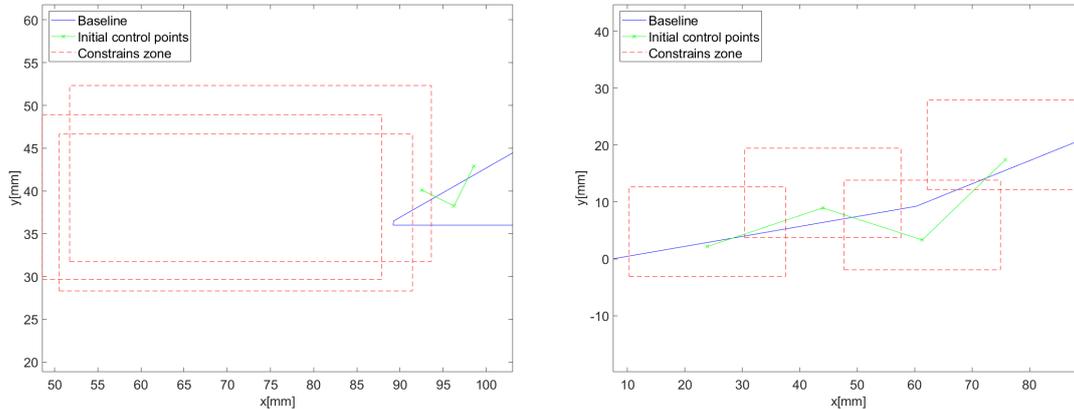
Figure 6.7: Constrain zones for the cowl lip (left) and the two ramps (right)

An initial population of control points of five individuals is given by random generation limited by the boundary previously defined. Then, the mean value of ordinates is taken and then new coordinates near the mean value are chosen. This process permits to create only streamlined geometries; the round ones do not allow compression between channel input and output, as they are not considered by the optimisation process. Then the ordinates are sorted and they are associated with the correspondents x-values[1]. Finally,

---

[1]lines from 67 to 71 are only some of the lines that are repeated with different indices

optimisations starts with the GA algorithm. Here, the *fitness* function is the function to be optimized.

```matlab
42  %Initialization of the Population
43  pcR = reshape(pcR,[numel(pcR), 1]);
44  pcCL = reshape(pcCL,[numel(pcCL), 1]);
45  initPop_opt = zeros(nInd_opt,nVar_opt);
46  initPop_opt(nInd_opt-4,:)=[pcR pcCL];
47      for k=nInd_opt-4:nInd_opt
48          initPop_opt(k,:)=rand(1,nVar_opt).*(ub_opt-lb_opt)+lb_opt;
49      end
50
51  for i=1:5
52      media(i)=(initPop_opt(i,4)+initPop_opt(i,5)+initPop_opt(i,6))/3;
53  end
54
55  %Mean and sorted
56  for k=4:6
57      for i=1:5
58          if abs(initPop_opt(k)-media(i))>1 && initPop_opt(i,k)>media(i)
59          initPop_opt(i,k)=media(i)+1;
60          end
61          if abs(initPop_opt(i,k)-media(i))>1 && initPop_opt(i,k)<media(i
              )
62          initPop_opt(i,k)=media(i)-1;
63          end
64      end
65  end
66
67  row1=initPop_opt(1,1:3);
68  Row1=initPop_opt(1,4:6);
69  [Row1,sortIdx] = sort(Row1);
70  row1=row1(sortIdx);
71  initPop_opt1=[row Row]
72  initPop_opt2=initPop_opt(1:5,7:14);
73  initPop_opt=[initPop_opt1,initPop_opt2];
74
75  options = gaoptimset('InitialPopulation',initPop_opt,'Generations',
        nGen_opt,'PopulationSize',nInd_opt);
76  pc_opt = gamultiobj(@fitness,nVar_opt,[],[],[],[],lb_opt,ub_opt,options
        );
```

Code 6.6: Optimisation setup

Stopping the description of the main script, now the *fitness* script is going to be analyzed. This function uses the global variable *decvar* as an input, then it checks if there are duplication and finally it generates a folder dynamically. It gives the path of the two software used (SALOME and ANSYS Fluent), the path of the mesh that will be used by Fluent for the simulation and the path of the journals. The latter are written by others two scripts and they represents the commands that the two software will use during their simulation. Finally, Salome starts and generates the geometry and the mesh. Then, it closed and ANSYS Fluent starts, takes the mesh saved from SALOME and computes the fluid flow analysis. Some reports, figures and data are saved in the folders previously created.

```matlab
1  function f = fitness(decvar)
2
```

```matlab
3   global counter datab;
4   counter = counter +1;
5
6   datab.decvar(counter,:)=decvar;
7   id = find(ismember(datab.decvar(1:counter-1,:), decvar, 'rows'),1);
8
9   if isempty(id)
10  counters = sprintf('%d',counter);
11
12      if ~exist (counters,'dir')
13          mkdir (counters);
14      end
15
16      path_salome = 'C:\SALOME-9.7.0';
17      path_fluent = 'C:\Program Files\ANSYS Inc\ANSYS Student\v212\fluent
            \ntbin\win64';
18      mesh = 'C:\Users\pietr\Desktop\Tesi\Ottimizzazione3\%s\Mesh-export.
            unv', counter ;
19
20      %Journal
21      journal_salome_MSPC(counter,datab.decvar);
22      journal_fluent_MSPC(counter);
23
24      %Start Salome e Fluent
25      cd(path_salome);
26      !run_salome.bat -g C:\Users\pietr\Desktop\Tesi\Ottimizzazione3\
            journal_salome_MSPC.py
27
28       pause(120)
29       !kill_salome.bat
30
31      cd(path_fluent);
32      !fluent.exe 2ddp -t4 -gu -driver null -i C:\Users\pietr\Desktop\
            Tesi\Ottimizzazione3\journal_fluent_MSPC.jou
33
34      ind=sprintf('C:/Users/pietr/Desktop/Tesi/Ottimizzazione3/%s/',
            counters);
35      cd(ind);
```

Code 6.7: *fitness* parameters

Given the data from ANSYS Fluent, they are read by the function and they are allocated in the objective function called $f(1), f(2)$ and $f(3)$. Here the constrains of drag, pressure ratio and residual are applied. Finally, the data are saved in the matrix f.

```matlab
36      %Constrains on f(1),f(2) and f(3)
37      f(1) = drag;
38      f(2) = ipratio;
39      f(3) = iptotratio;
40
41      if drag >0.2
42          f(1) = 1000;
43          f(2) = 1000;
44          f(3) = 1000;
45      end
46
47      if pratio <1
```

```
48          f (1) = 1000;
49          f (2) = 1000;
50          f (3) = 1000;
51       end
52
53       if  residuals >0.001
54          f (1) = 1000;
55          f (2) = 1000;
56          f (3) = 1000;
57       end
58
59       datab.f(counter,:)=[f(1) f(2) f(3)];
60
61    else
62       f=datab.f(id,:);
63    end
64    cd('C:\Users\pietr\Desktop\Tesi\Ottimizzazione3');
65    end
```

Code 6.8: *fitness* objective functions

For better understanding, the *fitness* function can be summarized as follows:

1. dynamic creation of folders to save data;

2. opening of Salome, generation of geometry and corresponding mesh through reading the journal;

3. opening of ANSYS Fluent, simulating fluid flow with the setting read in the journal, evaluation and saving of interesting variables;

4. objective functions evaluation and saving.

Returning to the main loop, here the control points defining the different Bezier curves and the corresponding objective functions are saved.

```
32   save('decvar.mat',  '−struct','datab','decvar');
33   save('fitness.mat','−struct','datab','f');
```

Code 6.9: Saving objective functions and control points

6.5 Results

The present section reports the optimisation results and their discussion. They are divide in two subsection: cowl lip optimisation and total optimisation.

6.5.1 Cowl lip optimisation

The Pareto front in showed in Figure 6.8. It reports the drag coefficient value, the static pressure ratio and the total pressure ratio of all the geometries derived from the optimisation analysis. These geometries are obtained by a set of Bezier control points created each time by the optimisation process previously discussed. The Pareto front is a three-dimensional surface generated by the optimised values. This shape proves that the three designing goals are in conflict with each other, indeed if only one objective function were a function of another, then the Pareto front would collapse into a curve; at one point if all the objective functions were related to each other.
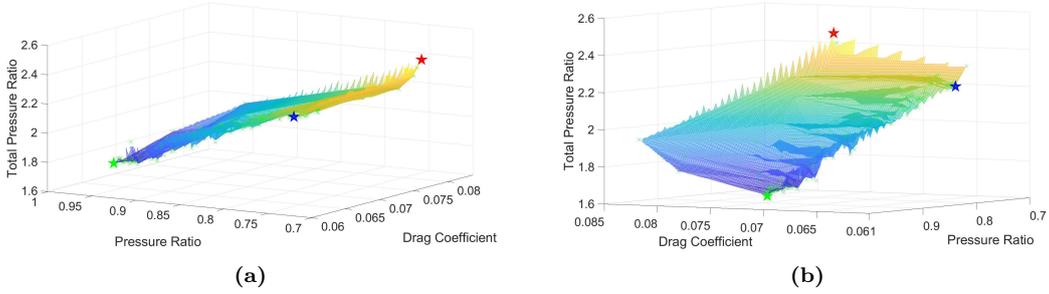
Figure 6.8: Pareto surface with highlighted optimal solutions: C_d (★), SPR (★), TPR (★)

In Figure 6.8, the red, green and blue stars mark the points representing the three optimal geometries for which the drag coefficient, SPR and TPR take on optimal values. Table 6.1 shows the values for each geometry and compares them to the baseline by representing the percentage change.

Case	C_d	SPR	TPR	$\frac{C_d - C_d^*}{C_d^*}[\%]$	$\frac{SPR - SPR^*}{SPR^*}[\%]$	$\frac{TPR - TPR^*}{TPR^*}[\%]$
Baseline	0.096	1.32	0.37	− − −	− − −	− − −
C_d	0.064	1.33	0.45	- 33.3	+ 0.76	+ 21.6
SPR	0.078	1.40	0.41	- 18.7	+ 6.06	+ 10.8
TRP	0.070	1.00	0.60	- 27.1	- 24.2	+ 62.2

Table 6.1: Baseline $(\cdot)^*$ and optimised geometries parameters

The three optimised geometries are represented in Figure 6.9 with the baseline geometry to get a better view of the changes in geometry. Concerning the first one (Fig. 6.9a), it represent the geometry optimised for the drag coefficient. The cowl lip is pointed with a very small radius of curvature at the tip of the point of maximum extension. The section is not symmetrical, as in the lower part of the cowl lip, the profile takes on a slight curve that accompanies the flow inside the canal

Concerning the geometry obtained by the SPR optimum design (Fig. 6.9b), the cowl lip has an upward direction and the radius of curvature is wider than the previously discussed geometry, resulting in a more rounded tip. In addition, there is an inward swelling

in the lower part of the cowl lip. The upper part is also humped, but this is less pronounced than in the lower part.

For the geometry obtained for optimal TPR (Fig. 6.9c), the geometric variations are similar to the case of optimal C_d. However, this geometry presents a slightly downward biased direction, as can be more clearly seen in Figure 6.10. In addition, the profile is more rounded than the geometry for optimal C_d.

(a) C_d optimum   **(b)** SPR optimum

(c) TPR optimum   **(d)** Baseline geometry
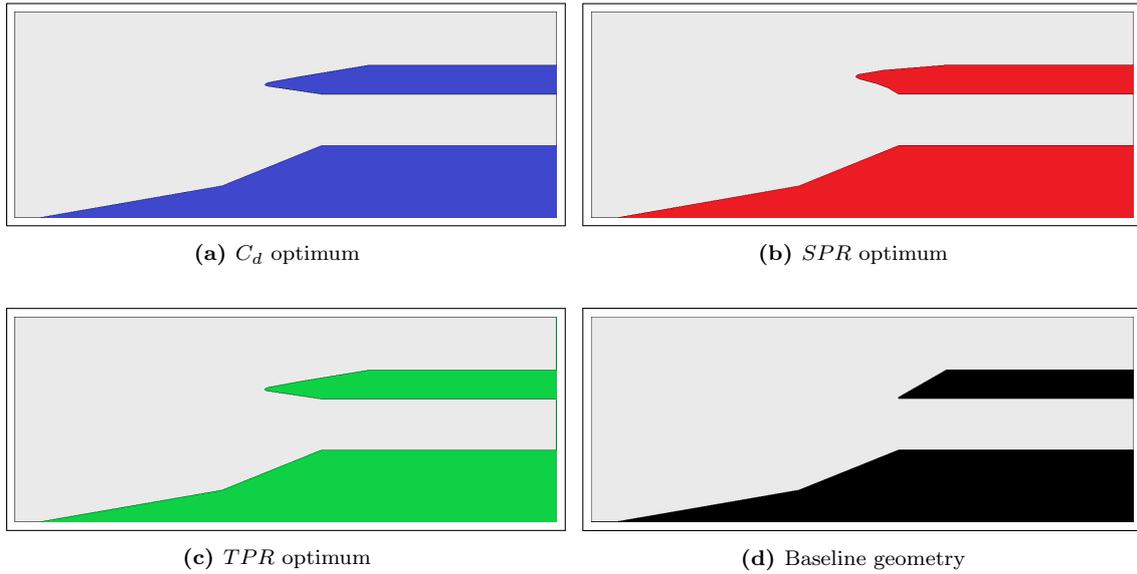
Figure 6.9: Optimal design geometries optimised for three different cases and the baseline geometry
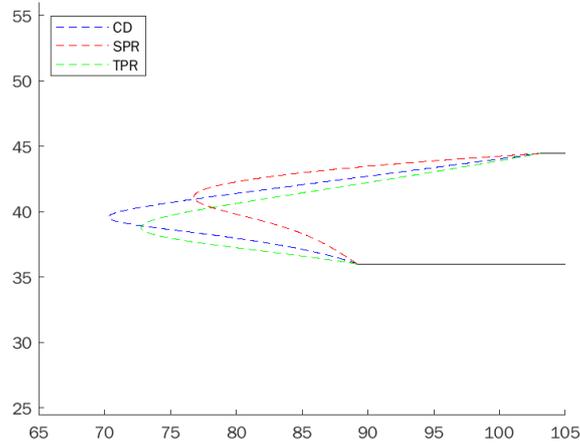
Figure 6.10: Geometries optimised for different variables

Different geometries lead to different flow behaviours, so a comparison of Mach and density fields for the optimal solutions is discussed. Before all, these fields are represented is Figure 6.11.

(a) C_d optimum Mach

(b) C_d optimum density

(c) SPR optimum Mach

(d) SPR optimum density

(e) TPR optimum Mach

(f) TPR optimum density
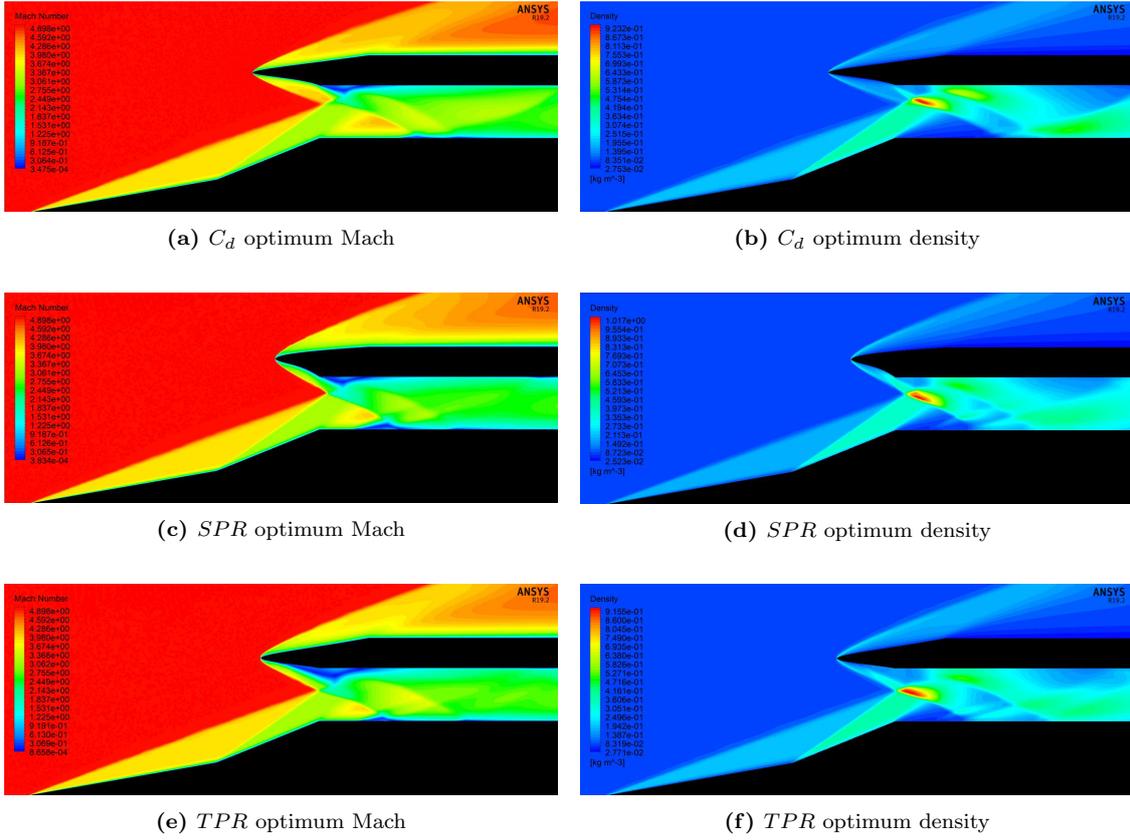
Figure 6.11: Mach and density field for the optimal solutions

Starting from the C_d-minimum configuration, the normal shock wave at the starting of the channel is missing due to the pointed geometry. The latter generates an oblique shock meeting the others oblique shock generated by the two ramps in the triple point. The latter is not well define as the baseline geometry, but it is highly mitigated. The recirculation bubble in the lower surface is almost non-existent while the one in the upper wall is reduced. The frontal wave system has been optimized to avoid detachment of the flow. This behaviour is typical for the reduction of the drag coefficient. The rarefaction of the density field in the portion of the lower wall is still visible. The compression of the flow is still present but it is not right after the triple point. The expansion fan in the channel is similar to the one in the baseline geometry. Regarding the external surface, the flow is more homogeneous with the freestream flow while in the baseline the flow was highly compressed.

Concerning the SPR geometry, the Mach and the density field well visualized the triple point. Differently form the baseline geometry, the triple point is generated by the intersection and union of the oblique shocks of the two ramps with the oblique shock of the cowl lip. The cowl lip presents a curved lower wall with a precompression role of the flow without destabilizing the boundary layer. All the shocks merged in one point and the compression is not dispersed as the previous geometry. The compression is all present in one zone, indeed the second bubble of compression is not present in the channel. The detachment of the flow on the lower wall of the channel and the expansion fan are still present. The latter has minor intensity respect to the baseline geometry. In the channel the flow results more homogeneous. Regarding the external section, the flow suffers a

slightly compression.

The TPR geometry is similar to the C_d one. Also, here the oblique shocks intersect themselves in the triple point creating an initial compression. The latter is more intense that the one presented in the C_d optimum geometry. Moreover, the recirculation bubble in the upper wall of the channel is extensive and its shape goes well before the initial section. Also, the lower recirculation bubble is slighty more intensive. In the channel, the second high density zone disappears as in the SPR geometry. In addition, the flow presents two expansion fans responsible of the two minor recirculation bubble in the lower wall of the channel. Concerning the external section, the flow is compressed for a large section, similarly to the C_d geometry. Then an expansion occurs when the surface becomes flat.

6.5.2 Total optimisation

The optimisation process is applied to both the ramps and the upper lip at the same time. The same consideration of the Pareto surface as in the previous section can also be made for this case. Thus, three points corresponding to C_d, SPR and TPR are obtained.
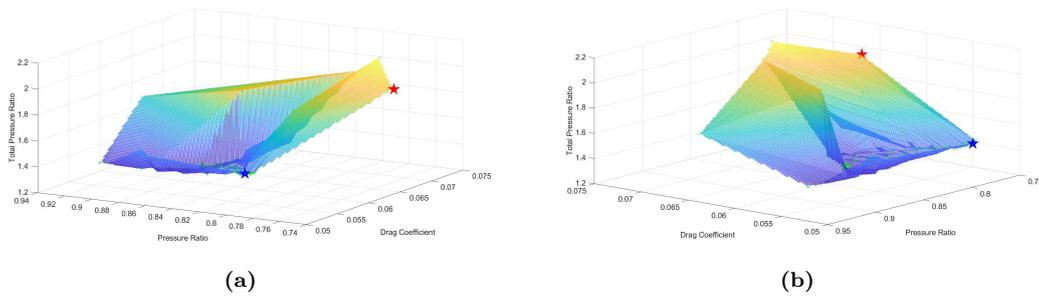
Figure 6.12: Pareto surface with highlighted optimal solutions: C_d (★), SPR (★), TPR (★)

The values of the optimised curves obtained with the Bezier formulation are presented in Table 6.2. The new geometric configurations generate different values which are compared with the basic ones.

Case	C_d	SPR	TPR	$\frac{C_d - C_d^*}{C_d^*}[\%]$	$\frac{SPR - SPR^*}{SPR^*}[\%]$	$\frac{TPR - TPR^*}{TPR^*}[\%]$
Baseline	0.096	1.32	0.37	$-\,-\,-$	$-\,-\,-$	$-\,-\,-$
C_d	0.051	1.27	0.66	- 46.9	- 3.79	+ 78.4
SPR	0.067	1.33	0.51	- 3.02	+ 0.76	+ 37.8
TPR	0.058	1.17	0.74	- 39.6	- 11.4	+ 100

Table 6.2: Baseline $(\cdot)^*$ and optimised geometries parameters

The three optimised geometries are represented in Figure 6.13. Considering the one optimised for the drag coefficient (Figure 6.13a), it has a pointed lip that is not symmetrical but slightly upward facing. Regarding the two ramps: the first one has a rounded shape, second one is starting before the analogue of the baseline. It has slightly minor inclination.

The optimised geometry of the static pressure ratio (figure 6.13b) has a rounded profile with a high radius of curvature. While the upper surface of the cowl lip is connected to the outer surface with a slightly inclined line, the inner surface is hollowed out internally with a rounded wall to better compress the flow. The cowl lip is smaller than the previous

one discussed. The two ramps are both slightly rounded. They are similar and have less camber than the other two optimised geometries. Between the two ramps, a flat area is generated. These details are more visible in the Figure 6.14b.

The TPR geometry is similar to the C_d one but with small changes. Regarding the cowl lip, it has a more rounded tip and is more pointed upwards. The two ramps are almost identical with the ones of the C_d geometry, but the first one is slightly more rounded.

The particulars of the three geometries are more visible in the Figure 6.14a and 6.14b.

(a) C_d optimum

(b) SPR optimum

(c) TPR optimum

(d) Baseline geometry
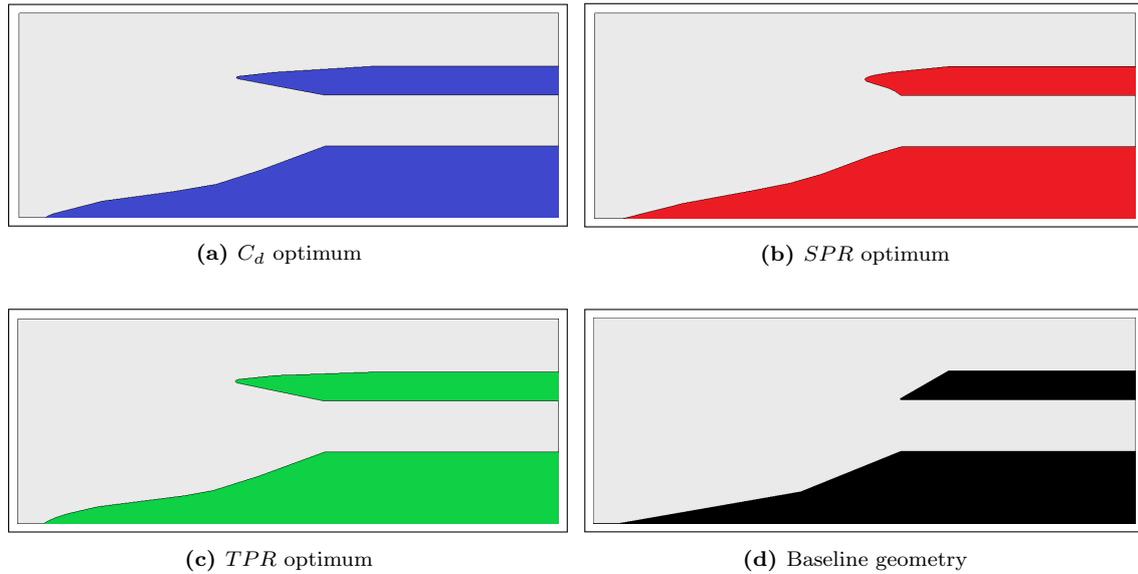
Figure 6.13: Optimal design geometries optimised for three different cases and the baseline geometry

(a) Cowl Lip optimised
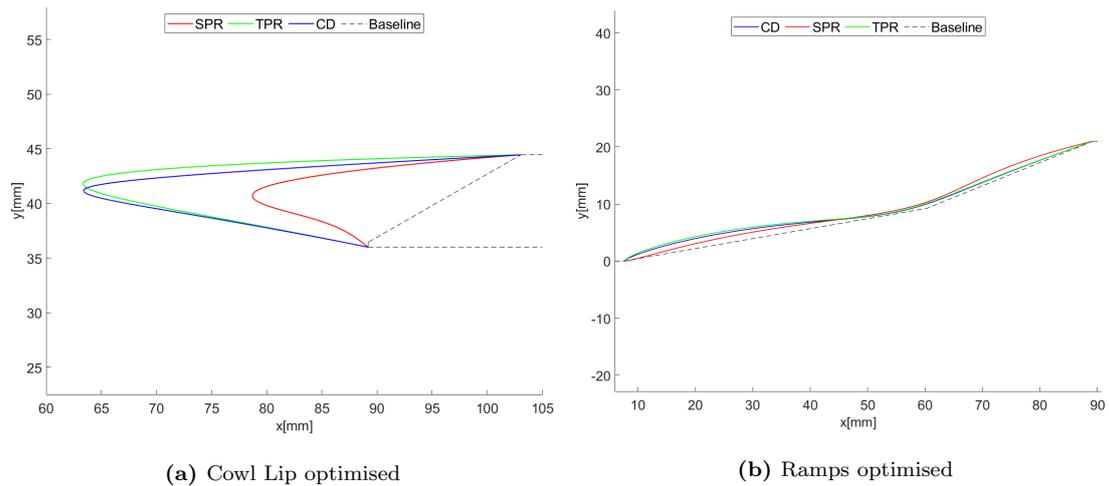
(b) Ramps optimised

Figure 6.14: Cowl lip and ramps optimised for the different objective function

Mach and density field are visualized in Figure 6.15. The description of the flow behaviour through the different geometries is discussed in the next section.

(a) C_d optimum Mach

(b) C_d optimum density

(c) SPR optimum Mach

(d) SPR optimum density

(e) TPR optimum Mach

(f) TPR optimum density
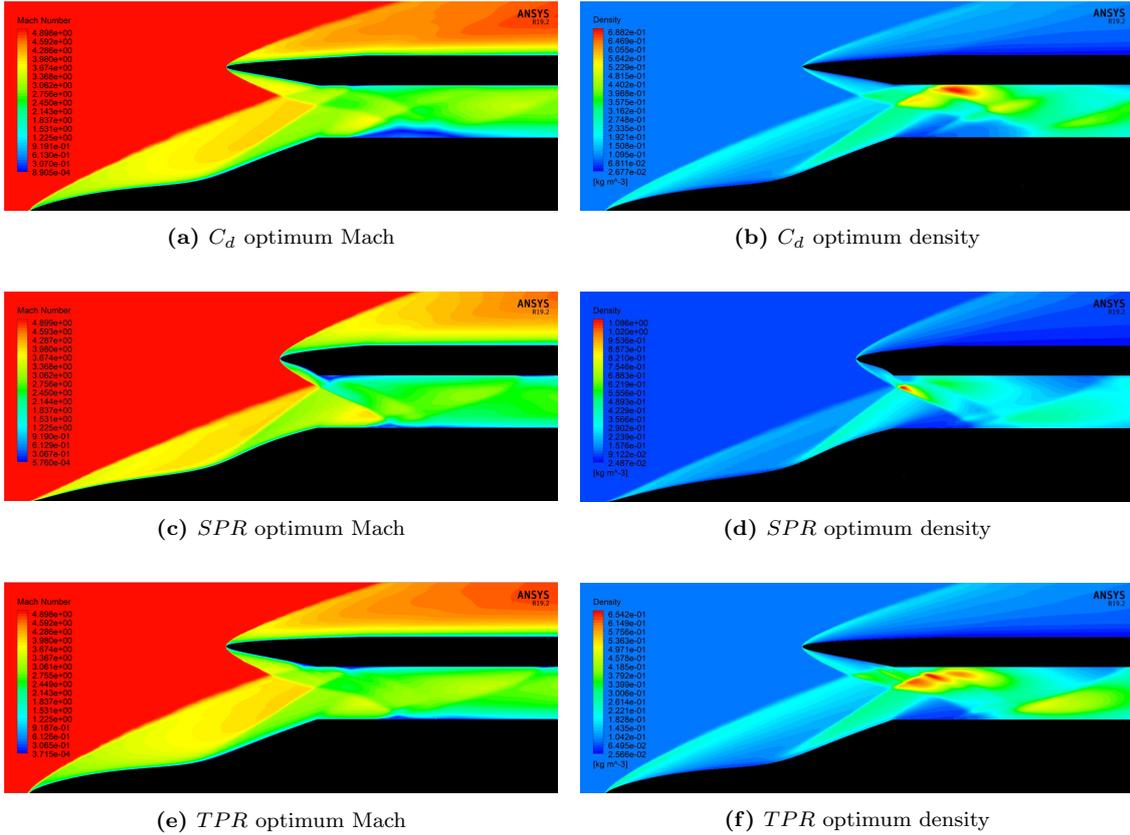
Figure 6.15: Mach and density field for the optimal solutions

For the optimal Mach field C_d, there is the presence of two shock waves that give compression to the flow. These shock waves are generated initially by a bulge, which replaces the first ramp, and by the second ramp. The shock waves do not converge at a single point, but intersect with the shock wave generated by the cowl lip at two different points. The reflections of the shock wave are visible in the upper surface of the channel. The Mach field shows incipient separations in the lower surface of the channel due to the interaction between the shock wave and the boundary layer. Although the recirculation bubble is larger than in the other two cases, it occurs in a single area, unlike the other geometries which present a continuous separation in both canal walls. The formation of the triple point and the strong oblique shock resulting from the coalescence of several shock waves are highly attenuated resulting in the displacement of the compression bubble deeper into the channel. The frontal shock system induces maximum compression at the inner surface of the roof.

With regard to the SPR optimum, the ramps are replaced by two different bulges: a larger one for the first ramp and a smaller one for the second. The cowl lip is less massive than in the other two optimised geometries, allowing the creation of the triple point at the beginning of the channel which generates a recirculation zone on the upper surface of the channel. The front shock system induces a strong oblique shock affecting the lower surface of the canal. In addition, an expansion fan is generated by the expansion given by the geometry of the second ramp and the channel entrance. The interaction between wave and boundary layer generates a small recirculation bubble. With regard to the density field, the triple point creates a small compression bubble. Flow reflections are more visible.

The Mach field of TPR shows some similarity to that of C_d. The geometry of the ramps and the cowl lip are very similar, however a change in direction of the cowl lip generates different flow behaviours. Starting from the Mach field, the triple point is attenuated and the two shock waves, created by the optimised ramp, intersect the one generated by the cowl lip at two different points. Thus, the shock present at the beginning of the channel is mitigated. There are three different recirculation bubbles, two in the upper surface of the channel, caused by the interaction between the two shocks and the boundary layer, and another generated by the same principle in the lower surface of the channel. The upper bubbles are not present in the C_d geometry, while the lower channel bubble is less intense than in the C_d case. There are not as many shock wave reflections. Regarding the density field, the second part of the optimised ramp causes the flow to enter the channel at an angle until it hits the upper surface of the channel. Here, several compression bubbles are present. In addition, a second slight compression is present at the end of the channel due to the impact of the flow with the bottom surface.

An analysis of the pressure and friction coefficient can be very helpful to evaluate the different mechanism of the optimised geometry. A comparison between the coefficients for the baseline and the optimised geometry is reported in Figure 6.16. While the pressure coefficient provides a quantitative description of the flow compression, the friction coefficient identifies the separation zones.

(a) Pressure coefficient

(b) Friction coefficient
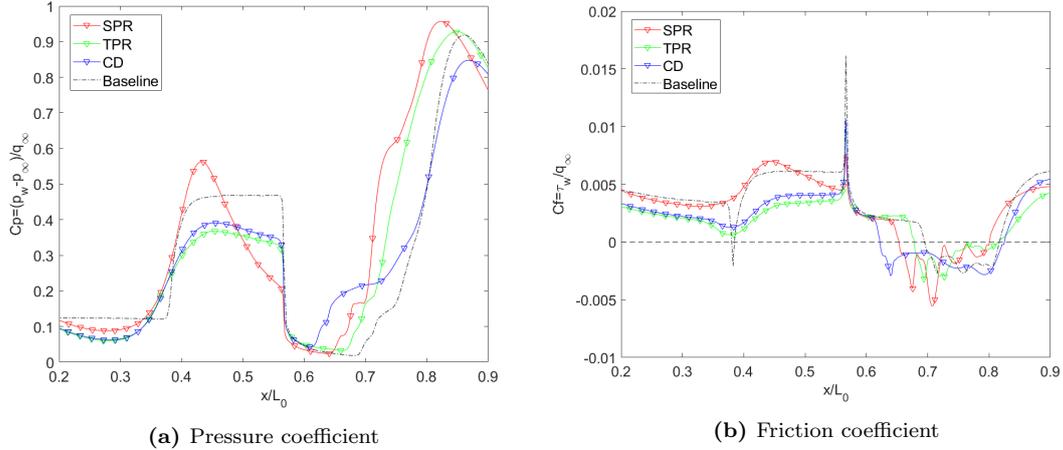
Figure 6.16: C_P and C_f comparison between the baseline and the optimal design solutions

As far as the SPR geometry is concerned, it presents a decrease in pressure due to the first bulge of the ramps, then a pressure peak caused by the flat area between the first and second bulge and the shock train. Finally, there is a rapid decrease in pressure due to the presence of the last bulge. The geometries TPR and C_d have a very similar trend. First, a pressure decrease is present at the ramp bulge, then the flow experiences a compression due to the second shock wave and a slight pressure decrease due to the presence of a small bulge as the second ramp.

All geometries and baseline have an expansion shock at the beginning of the channel with a subsequent sudden drop in pressure. The latter causes a recirculation bubble with different shapes and intensities governed by the intensity of the previous shock. High-pressure/density values govern the region immediately before the channel, causing the flow separation. For all geometries, the friction coefficient does not reach the peak of that

of the baseline. Furthermore, there is no bubble between the two ramps, as in the base case, for all optimised geometries.

6.5.3 Comparison between the two optimisation

The results of the six geometry obtained by the optimisation process are reported in table 6.3. While an optimisation of both the ramps and cowl lip gives better results for drag coefficient and total pressure ratio, in the case of static pressure ratio the optimisation of the only cowl lip is preferred. The changes on the ramps improve significantly the results of the optimisation, especially in TPR case.

Optimisation Case	$\frac{C_d - C_d^*}{C_d^*}[\%]$	$\frac{SPR - SPR^*}{SPR^*}[\%]$	$\frac{TPR - TPR^*}{TPR^*}[\%]$
Only Cowl Lip	- 33.3	+ 6.06	+ 62.2
Ramps and Cowl Lip	- 46.9	+ 0.76	+ 100

Table 6.3: Parameters comparison between optimised geometries and baseline $(\cdot)^*$

Regarding the optimised geometries, as visible in Figure 6.17, the TPR and C_d obtained in the total optimisation provided a thinner cowl lip than the optimisation of the cowl lip alone. The geometries of SPR are very similar, the shape is almost identical but with differences in the area occupied. The similarity of all shapes also verifies the optimisation process and the ability of all scripts to be flexible and therefore the possibility of being used for different shapes and problems.
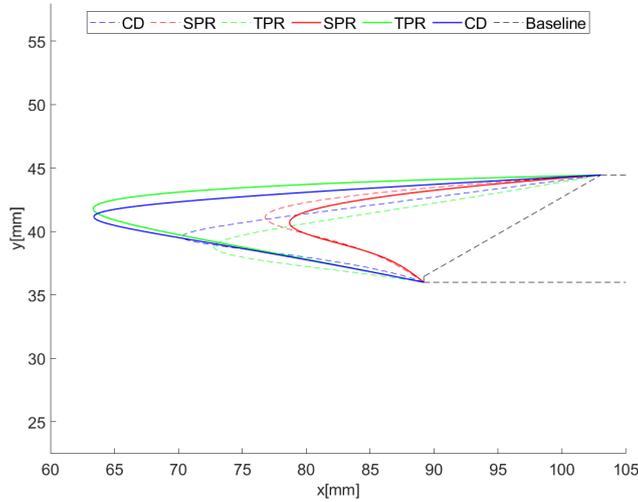
Figure 6.17: Comparison of different cowl lips optimised with (——) and without (- - -) the ramps

Chapter 7

Conclusion

The present study provides a numerical optimisation strategy for automatic design of the cowl lip and ramps of hypersonic inlets. A Reynolds-Averaged Navier-Stokes system of equations is used to validate the model in terms of global an local quantities respect to the experimental and literature results. Grid resolution and turbulence model sensitivity is performed revealing the minimum mesh size of 300k and the $k - \omega\ SST$ was the best settings for a good compromise between calculation time and accuracy. In particular, this turbulence model well resolved the entire flow including the wall dynamics.

After the validation of the mesh and the model, the RANS equations are used to re-solve different geometries generated with the Genetic Algorithm. The new inlet shapes are determined with the usage of different goals: the minimisation of the drag coefficient and the maximisation of static and total compression ratios. The first goal is obtained with a shape with minor interference possible to the overall inlet, while the others two objectives are obtained with the disappearance of the normal shock in favour to the gradual compression generated by different oblique shocks.

Future works could test the same geometry with a high fidelity CFD strategies as LES and DNS, to well represent the flow of the inlet. Moreover, future test could use the optimisation strategy presented in this thesis to generate new geometry with larger boundary constrains. Finally, more works could study the behaviour of unsteady flow of the shapes obtained in this optimisation process to analyze the buzz phenomenum never discussed in this thesis.

Ringraziamenti

Vorrei partire ringraziando l'ingegnere De Vanna Francesco e relatore di tale tesi. Nonostante i fitti impegni ha sempre dimostrato elevata disponibilità e professionalità. La passione nella fluidodinamica computazionale espressa nel suo corso di insegnamento mi ha appassionato a mia volta a tale disciplina facendomi scoprire un mondo nel quale spero di intraprendere la mia futura carriera lavorativa. Ulteriore ringraziamento va anche all'ingegnere Danilo Bof, che con grande pazienza mi ha aiutato diverse volte a uscire da problemi che sembravano insormontabili.

Mamma, papà e Giulia, voi mi avete sostenuto in questo lungo e tortuoso percorso spronandomi a migliorare sempre più. Spero di avervi resi orgogliosi e vi ringrazierò sempre per l'affetto che continuate a darmi ogni giorno.

Cari nonni Giovanni e Franca, voi mi avete viziato con molto cibo e tante risate. Mi avete cresciuto insegnandomi il valore del lavoro ma anche che la nonna ha sempre ragione! Siete una coppia fantastica e divertentissima. Quello che sono oggi è anche merito vostro.

Nonna Gigia non sei potuta essere qui oggi a differenza di tre anni fa. Sei stata la nonna delle patatine fritte, dell'uovo sbattuto e delle partite a pallone. Mi manca venire a casa tua per controllarti e fare gli stupidi insieme. Ti ringrazio per i momenti di vita che mi hai regalato.

Nonno Natale, sono passati tanti anni ma non mi sono mai dimenticato di te. Sei stato il mio eroe. Ti ricordo altissimo, severo ma super buono. Ti sarò sempre grato per quello che hai fatto insieme a me e spero che da lassù ti possa rendere orgoglioso ogni giorno. Mi manchi.

Per gli amici di sempre Pier e Riccardo. Nonostante siamo sempre lontani, vi ringrazio per l'immenso sostegno che mi avete dato e per essere stati partecipi in ogni momento della mia vita, bello o brutto che sia. Vi ringrazio perchè siete due veri amici.

Coinquilini e vicine, abbiamo passato diverso tempo insieme e mi avete piano piano conosciuto. Siete stati un gruppo bellissimo con cui vivere e condividere ogni momento giornaliero. Mi mancherete e grazie per questi anni di spensieratezza.

Agli amici universitari: Luca, Lorenzo, Davide, Giulia, Giuliana e Sara. Ci siamo divisi con gli studi magistrali e con la fine di questi ma ci tengo a ringraziarvi per la bellissima amicizia che si è creata tra noi e continua ad esserci.

Ci tengo a ringraziare anche tutti i parenti. Siete una famiglia bellissima che mi ha sostenuto ma anche insegnato molto. Ringrazio tutti cugini e cugine poco seri che mi ritrovo ma anche tutti gli zii e zie che non sono da meno.

Denise, eccoci qua per la seconda volta. Ne abbiamo passate di tutti i colori, sopratutto nell'ultimo anno. Ti voglio ringraziare perchè tu più di tutti mi hai sostenuto e visto crescere in tutti questi anni. Ti ringrazio per la forza che mi dai ogni giorno e per essere un punto fisso nella mia vita.

Bibliography

[1] John D. Anderson. *Fundamentals of aerodynamics - Sixth Edition*. New-York: McGraw-Hill Education, 2017.

[2] Inc. ANSYS. *ANSYS Fluent User's Guide*. 2013.

[3] Federico Berto et al. "Time-Accurate Experimental Investigation of Hypersonic Inlet Buzz at Mach 5". In: *AIAA Journal* 58.5 (2020), pp. 2197–2205. DOI: 10.2514/1.J058764.

[4] Salsa S. Bramanti M. Pagani C.D. *Analisi matematica 1*. Bologna: Zannichelli editore S.p.A, 2011.

[5] Francesco De Vanna et al. "Hypersonic intake design using a CFD data-driven multi-objective optimisation strategy". In: *ASCEND 2021*. DOI: 10.2514/6.2021-4199.

[6] Francesco De Vanna et al. "Large-Eddy Simulations of the Unsteady Behavior of a Hypersonic Intake at Mach 5". In: *AIAA Journal* 59.10 (2021), pp. 3859–3872. DOI: 10.2514/1.J060160.

[7] K. Deb et al. "A fast and elitist multiobjective genetic algorithm: NSGA-II". In: *IEEE Transactions on Evolutionary Computation* 6.2 (2002), pp. 182–197. DOI: 10.1109/4235.996017.

[8] Benini E. *Propulsione Aerea*. Padova: Cleup, 2005.

[9] Saeed Farokhi. *Aircraft Propulsion*. John Wiley & Sons Ltd, 2014.

[10] J.H. Ferziger and M. Peric. *Computational methods for fluid dynamics*. Milan: Springer, 2002.

[11] H. H. Jr. Hurt. *Aerodynamics for naval aviators*. Federal Aviation Administration: Direction of commander, naval air systems command, 1965.

[12] Abdullah Konak, David W. Coit, and Alice E. Smith. "Multi-objective optimization using genetic algorithms: A tutorial". In: *Reliability Engineering & System Safety* 91.9 (2006), pp. 992–1007. DOI: https://doi.org/10.1016/j.ress.2005.11.018.

[13] M A Kotov et al. "Preliminary experimental assessment of supersonic airflow behavior over ExoMars and X–43 inlet models using multiple flow regime shock tube". In: *Journal of Physics: Conference Series* 1009 (Apr. 2018), p. 012038. DOI: 10.1088/1742-6596/1009/1/012038.

[14] Adrian Maries et al. "Interactive Exploration of Stress Tensors Used in Computational Turbulent Combustion". In: Berlin, Heidelberg: Springer, 2012. DOI: 10.1007/978-3-642-27343-8_7.

[15] Laurie A. Marshall, Griffin P. Corpening, and Robert T. Sherrill. "A Chief Engineer's View of the NASA X-43A Scramjet Flight Test". In: 2005. DOI: 10.2514/6.2005-3332.

[16] Àlex Navó and Josep M. Bergada. "Aerodynamic Study of the NASA's X-43A Hypersonic Aircraft". In: *Applied Sciences* 10.22 (2020). ISSN: 2076-3417. DOI: 10.3390/app10228211.

[17] Stephen B. Pope. *Turbulent Flows.* New-York: Cambridge University Press, 2013.

[18] Inc. The MathWorks. *Genetic Algorithm and Direct Search Toolbox For Use with MATLAB, User's Guide.* 2004.

[19] H K Versteeg and Malalasekera W. *An introduction to computational fluid dynamics, The finite volume method.* London: Longman and Scientific Technical, 1995.

[20] Sun Yijie and Shen Gongzhang. "Improved NSGA-II Multi-objective Genetic Algorithm Based on Hybridization-encouraged Mechanism". In: *Chinese Journal of Aeronautics* 21.6 (2008), pp. 540–549. DOI: https://doi.org/10.1016/S1000-9361(08)60172-7.