Simulation of Cold Flow in Cavity-Ramp Combustor

BY

FEDERICO LUGNANI
B.S., Politecnico di Torino, Turin Italy, October 2011

THESIS

Submitted as partial fulfillment of the requirements
for the degree of Master of Science in Mechanical Engineering
in the Graduate College of the
University of Illinois at Chicago, 2013

Chicago, Illinois

Defense Committee:
Farzad Mashayek, Chair and Advisor
W. J. Minkowycz
Pietro Asinari, Politecnico di Torino

To my family

ACKNOWLEDGMENTS

I'd like to express my gratitude towards Dr. Farzad Mashayek, who has been a great academic advisor and has given me the opportunity to work in the Computational Multiphase Transport Laboratory. His patience and constant help and suggestions, together with his accurate review, have made this work possible.

I also want to thank Prof. Pietro Asinari, my second advisor, for helping me in choosing my thesis's subject, and for being always at my disposal for academic and bureaucratic issues during the last twelve months, and Prof. W.J. Minkowycz, for serving on my Defense Committee.

I also thank Prof. Muzio Gola, responsible of TOP-UIC program for Mechanical Engineering, for helping me and my colleagues throughout the duration of the program. His magical interventions and remarkable patience have helped all of us to a succesful academic career. My gratitude is also for Lynn Thomas and Alice Gili, irreplaceable on UIC and PoliTo sides.

I feel the need to thank my colleagues and friends of this semester, Pipi, Fra, Pisolino and Paolino in particular, for sharing with me this experience, for giving me strength and motivation to reach my objectives, and also for the unforgettable fun times. Thanks also to Ste, whose friendship and company have accompanied me for the last year and a half, and to my other friends in Italy, Antonello, Chiara and Mimmo, and in Chicago, Mary, Brayden, Damian and the IFC, Paneb, Dani, Niko, Giorgione, and the others. I want to thank also my labmates, Jon, Hesam and Zia, whose help and company of last year have permitted me to write this thesis in a relaxed and yet stimulating environment.

ACKNOWLEDGMENTS (continued)

Thanks also to my colleagues and friends in Mechanical Engineering in Torino, and especially Paolo, whose knowing and infinite work together have made it possible to overcome the hardest academic semester of college with a smile, and to Fabri, Carpi, Came and the others.

One of my last and greatest thanks goes to my friends in Torino, Fede, Aldo, Cri, Davide, Dani, Riki, Dona, Cenzo, Fra M, Anto, Fra, Remo, Matteo, Bert, Ale L, Mauro, Miki, Gerry, Luca I, Luca N, Mario, Lupo, Maury, Ale G, Mauri M, Giorgia, that Politecnico or High School or other random events have given me the fortune to know, and whose friendship along the years has made me overcome every sort of difficulty I have faced, on both personal and academic planes. My path at college is hopefully over, but I'm sure the time we have shared over last years will make our friendship last forever. You've definitely made a lot to help me become what I am today.

My final thank goes to my family, and in particular to my parents, for the unconditional support and love that gives me strength and ambition to pursue always harder objectives, and to my brother Francy, whose teachings and challenges over the years have always given me determination to improve myself in every field. I really wouldn't be here without you guys.

FL

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

CHAPTER **PAGE**

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES (continued)

FIGURE **PAGE**

LIST OF FIGURES (continued)

FIGURE **PAGE**

LIST OF FIGURES (continued)

FIGURE **PAGE**

LIST OF FIGURES (continued)

FIGURE **PAGE**

NOMENCLATURE

A_s	first parameter for the sponge layer
BFS	backward-facing step
c_p	constant pressure specific heat
CFL	Courant-Friedrich-Levy number
D	domain
D_i	i-th subdomain
Δt	time interval
Δx	grid spacing
e_t	specific total energy
e_Ω	specific total energy in the volume Ω
ER	expansion ratio
$\mathbf{F_f}$	inviscid flux in the x-direction
$\mathbf{F}_\nu$	viscous flux in the x-direction
$\mathbf{F_T}$	thermal flux in the x-direction
$\tilde{\mathbf{F}}$	mapped flux in the X-direction
$\mathbf{G_f}$	inviscid flux in the y-direction
$\mathbf{G}_\nu$	viscous flux in the y-direction

NOMENCLATURE (continued)

$\mathbf{G_T}$	thermal flux in the y-direction
$\tilde{\mathbf{G}}$	mapped flux in the Y-direction
$\mathbf{H_f}$	inviscid flux in the z-direction
$\mathbf{H}_\nu$	viscous flux in the z-direction
$\mathbf{H_T}$	thermal flux in the z-direction
$\tilde{\mathbf{H}}$	mapped flux in the Z-direction
$\mathbf{I}$	identity matrix
$\mathbf{J}$	Jacobian matrix
l_f	reference length
l_i	Lagrangian polynomial
m_Ω	mass contained in the volume Ω
M	flow Mach number
M_f	$v_f/\sqrt{\gamma R T_f}$ reference Mach number
$\mathbf{n}$	normal to the surface
N_s	second parameter for the sponge layer
p	pressure
Pr	$(c_p \rho \nu)/\lambda$ Prandtl number
R	universal gas constant

NOMENCLATURE (continued)

Re_f	$(l_f v_f)/\nu$ reference Reynolds number
$r(W)$	residual
S	control surface
t	time
t_f	reference time
T	temperature
T_f	reference temperature
T_i	Chebyshev polynomial
u	velocity of the fluid in the x-direction
v	velocity of the fluid in the y-direction
$\mathbf{v}$	velocity vector of the fluid
v_f	reference velocity
$\mathbf{v}_\Omega$	velocity vector in the control volume Ω
w	velocity of the fluid in the z-direction
$\mathbf{W}$	vector of conserved quantities
x	spatial coordinate in physical space
X	spatial coordinate in mapped space
y	spatial coordinate in physical space

NOMENCLATURE (continued)

Y	spatial coordinate in mapped space
z	spatial coordinate in physical space
Z	spatial coordinate in mapped space

Greek Symbols

α	thermal diffusivity
δ_{ij}	Dirac delta function
ϵ	secondary viscosity coefficient
γ	c_p/c_v heat capacity ratio
λ	thermal conductivity
μ	dynamic viscosity
μ_b	bulk viscosity
ν	kinematic viscosity
$\mathbf{\Pi}$	stress tensor
$\mathbf{\Pi}_\nu$	viscous stress tensor
ρ	density
ρ_f	reference density
$\sigma(x)$	damping function for the sponge layer
τ_{ij}	components of the stress tensor

NOMENCLATURE (continued)

Ω	control volume
Math Symbols	
$\cup$	union
$\otimes$	external or dyadic product
$\nabla\cdot$	divergence operator
∇	gradient operator for scalars
$\nabla\times$	gradient operator for vectors
$\sum$	summation
$\prod$	multiplication
$\partial a/\partial b$	partial derivative of variable a with respect to variable b
Subscripts	
0	initial
∞	free-stream
f	reference variable
Superscripts	
$\tilde{}$	mapped space variable
g	Gauss point
ggg	Gauss/Gauss/Gauss grid

NOMENCLATURE (continued)

ggl	Gauss/Gauss/Lobatto grid
glg	Gauss/Lobatto/Gauss grid
l	Lobatto point
lgg	Lobatto/Gauss/Gauss grid

SUMMARY

The main objective of this thesis is the analysis of a subsonic laminar cold flow in a dump combustor. This type of combustors have gained increasing attention because of their typical flow recirculation zone after a sudden change in geometry, a feature that makes them particularly suitable for applications where fuel residence time is a major issue. In this sense, dump combustors have interested the scramjet community, because of extremely high flow velocities that imply extremely low time to obtain a satisfactory completion of combustion.

This work adopts the three-dimensional version of Chebyshev Multidomain Spectral Method, because of its high accuracy, its appliability to complex geometries and its convenient parallel implementation. To create an appropriate geometry input file, a mesh generator has been implemented, focusing on the possibility of refining the grid in areas where the flow structure requires it.

To provide validation for the three-dimensional version of the numerical code, after a number of problems characterized by a simple setup, a laminar subsonic flow over a backward-facing step is tested, because of its analogy to a simplified version of the dump combustor. This kind of geometry has received wide attention in the literature, which makes it ideal to implement also a new type of boundary condition. Transmissive boundary conditions have been programmed and tested, believing that such type of boundary is more convenient than those already programmed.

As a final flow geometry, the backward facing step is modified to assume the typical shape of a dump combustor, so that the main interest of this thesis is pursued. The main scope of this

analysis is on the flow features in the cavity and in the proximity of geometrical discontinuities, with different Reynolds numbers, and also with different types of outflow boundary conditions.

CHAPTER 1

INTRODUCTION

1.1 Motivation

Supersonic airbreathing propulsion systems represent a key component of future high-speed transportation. The engines used in these systems typically do not use mechanical parts in order to compress the incoming air, but they rather rely just on their geometry to create a favorable combustion environment. The optimization of this geometry, to ensure minimal losses and maximum completion of combustion, is a challenging task, and many groups are involved in related research activities.

Ramjets and scramjets are subject to extremely unstable flows, characterized by turbulence, shocks, co-existence of non-homogenous physical phases and chemical reactions. All these elements contribute to increase the difficulty of providing measurements to a prohibitive level, so that mainly numerical simulations are employed to model this kind of problems. Recent advances in computer hardware coupled with the possibility of parallel computing, in fact, have greatly enhanced computational fluid dynamics potential.

This work is motivated by the interest in analyzing a supersonic reacting flow in the combustor of a scramjet. With incompressible turbulence still far from being fully understood, modeling turbulence in compressible flows exceeds the aim of this thesis. The present work, in fact, limits its analysis to subsonic laminar flows in the geometry mentioned. Laminar single-

phase flows are chosen for their simplicity, and a subsonic Mach number low enough permits to neglect compressibility effects. Therefore, this thesis aims to model this primitive kind of flows in a geometry representative of a scramjet combustor.

1.2 Ramjet and Scramjet

In order to create thrust, rockets combine liquid fuel with liquid oxygen. Removing the latter, ensuring another way of supplying oxygen, makes it possible to reduce their dimensions or increase their payload. That is precisely the idea on which a ramjet is based. The ramjet, in fact, is a form of airbreathing engine that exploits the forward motion to compress incoming air, without the need of a rotary compressor, and also without the need of an oxygen tank onboard. Because it is an airbreathing engine, it cannot generate thrust at zero speed, i.e. it cannot move an aircraft from a standstill, so that some other propulsion system is needed for this task. The initial compression is responsible of a deceleration of the flow down to subsonic speed, so that combustion takes place in a subsonic flow. Ramjets efficiency reaches a maximum at supersonic speeds around Mach 3, but these propulsion systems are able to operate up to Mach 6. A schematic of this type of engine is shown in figure 1.

A variant of the ramjet is the scramjet (Supersonic Combustion Ramjet), in which even combustion takes place in a supersonic flow. The air flow, in fact, is supersonic throughout the whole extension of the engine, so even through the combustor. Theoretically, the top speed of a scramjet engine is placed between Mach 12 (15,000 km/h) and Mach 24 (29,000 km/h). A schematic of the scramjet engine is represented in figure 2.

The scramjet is constituted of three main parts:

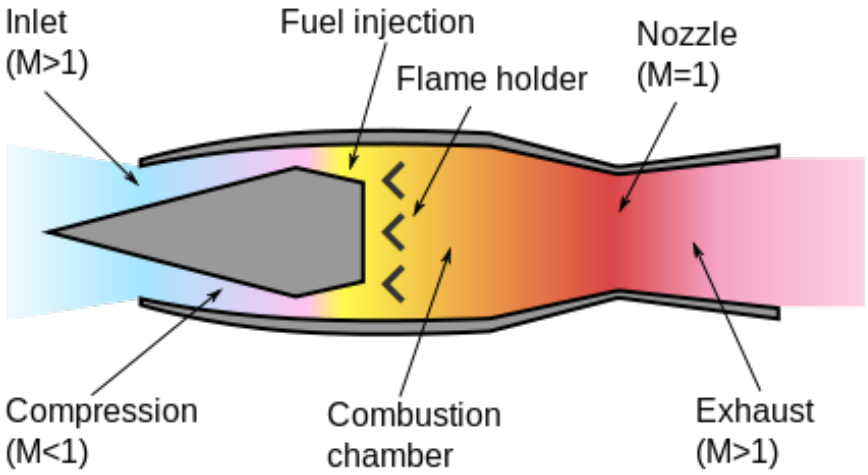


Figure 1: Graphical representation of a ramjet engine[1].

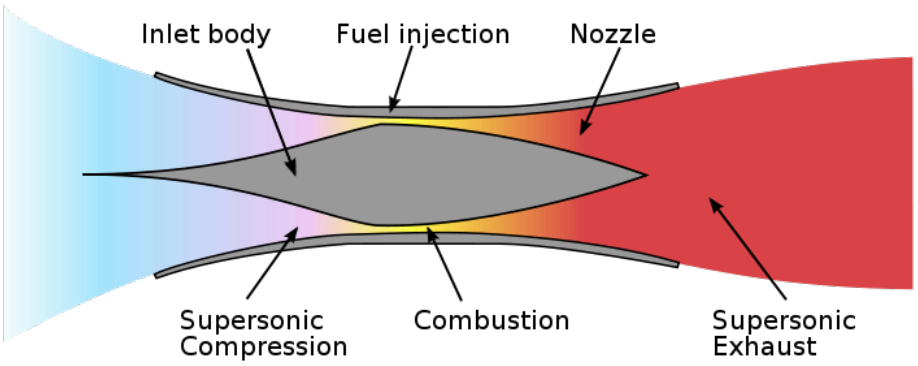


Figure 2: Graphical representation of a scramjet engine[2].

- A converging inlet, responsible of compression and deceleration of incoming air;
- A combustor, where fuel is burned using atmospheric oxygen to produce heat;
- A diverging nozzle, where the supersonic heated airflow is accelerated to produce thrust.

The absence of any rotating parts makes this type of engine easier to manufacture, and reduces its weight, complexity and number of failure points.

Because of the supersonic airflow throughout the scramjet, it has been necessary to choose a particular geometry for the combustor to increase the residence time of the mixture, to have a more complete combustion. The geometry adopted for the combustor involves a cavity, because of its feature of flame-holding device[4]. That is why dump combustors are taken into consideration for ramjets[5]. Such type of combustors, in fact, rely on a large change in the combustor area to perform recirculation, and therefore increase the residence time of the mixture.

1.3 Dump Combustor

This work is motivated by the interest in analyzing the turbulent flow in a dump combustor. The geometry considered in this thesis is taken from Tuttle *et al.*[6], so that the dump combustor is modeled as a backward-facing step followed by a ramp. Its main feature is the sudden expansion of the flow into the combustion cavity, which is responsible for flow separation. In fact, this geometry leads to the development of stable recirculation regions behind the step, increasing the residence time of the fuel and therefore favoring both ignition and a more complete combustion, as it is shown by figure 3. At moderate to high Reynolds numbers, the separated

flow becomes unstable and this culminates into turbulence, which favors the combustion process increasing its volumetric efficiency.

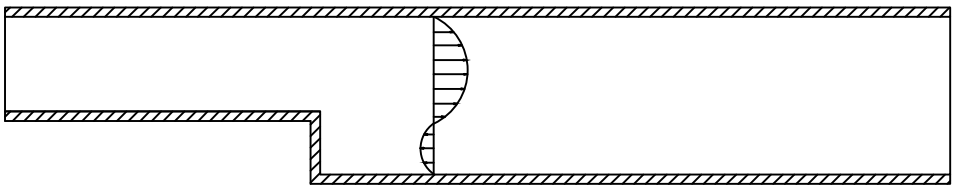


Figure 3: Graphical representation of dump combustor u velocity distribution, no countercurrent shear.

To increase further recirculation and residence time a countercurrent shear may be applied [7] [8], but such a case is not analyzed in this work. This research, in fact, is focused on the properties of the backward-facing step and of the actual dump combustor geometry.

1.4 Backward-Facing Step

The backward-facing step geometry has been widely studied in literature. The reference results for this geometry are those obtained by Armaly *et al.* [3]. Armaly *et al.*'s experimental and numerical data have been referenced by authors like Denham and Patrick [9], Le, Moin and Kim [10], and Lee and Mateescu [11], whose results yielded analogous conclusions. Such experiments have provided useful data for comparison and for the code testing.

As shown by Armaly *et al.*, the flow can be considered two-dimensional only for $Re < 400$ or $Re > 6600$, while for the range of Reynolds numbers in between non-symmetric sidewall

separation occurs, so that flow three-dimensionality must be taken into account. The flow can be considered laminar for Reynolds numbers up to 1200 [12], so that in this region reattachment length varies with Re (it increases as the Reynolds number increases). For Reynolds numbers greater that 6600, the flow can be considered turbulent, and this range is characterized by a fixed reattachment length, independent of the Re assumed.

Research on backward-facing step flow continues together with the advances in computational methods. Williams and Baker [13] investigated a three-dimensional laminar flow over a backward-facing step, for Reynolds numbers in the range $100 < Re < 800$. Their results confirmed the influence of the three-dimensionality of the flow for $Re > 400$ by means of a correct prediction of the reattachment length in this range. Biswas *et al.* [14] carried out three-dimensional simulations for low and moderate Reynolds numbers, i.e. in the range $Re < 400$, to provide informations on mean velocities and streamlines. Park *et al.* [15], instead, studied the separating shear flow behind a backward-facing step for both non-reactive and reactive flow conditions, considering this geometry suitable for a combustor.

In the last few years, growing interest has been shown in the study of the flow over a backward-facing step for higher Reynolds numbers and higher Mach numbers. More and more sophisticated computational methods have opened the doors for the development of particle-tracking algorithms, to be used in the modelling of two-phase turbulent flows. Particle image velocimetry (PIV) has been used by Schroeder *et al.* [16] in order to study the structure of a transitional flow. Chen *et al.* [17], instead, used nano-based planar laser scattering to perform an experimental study of the fine structures generated by a backward-facing step flow in supersonic

flow conditions. Kanchi *et al.*[18], on the other side, performed simulations using a spectral element method and Large Eddy Simulation (LES) for a backward facing step for Reynolds numbers higher than 28,000, and analyzed the effect of a turbulent inflow boundary condition.

In order to ensure stability in case of discontinuities caused by the generation of shock waves, different approaches have been employed in the literature. In particular, Nguyen *et al.*[19] studied different strategies to stabilize solutions in the proximity of discontinuities. Kurganov and Liu[20], instead, proposed a new finite volume method to solve hyperbolic systems of conservation laws, enforcing stability with an adaptive artificial viscosity. Such a method, in fact, increases the viscosity value where the discontinuity is located, so that computations are prevented from being unstable.

1.5 Numerical Scheme

This work is based on Chebyshev multidomain spectral method (CMSM) introduced by Kopriva[21] [22] [23] and implemented, tested and improved by Jacobs[24] for the computation of compressible turbulent two-phase flows. The cases that are considered, however, do not involve two-phase flows and do not exploit any instrument of particle tracking developed in the code, but instead focus on flows induced into turbulence by a sudden change in geometry, such as the backward-facing step.

Simulations are performed by a direct numerical simulation (DNS), and are executed by solving the fully coupled Navier-Stokes system of equations, so that they can provide detailed features of the flow, details that would not be possible using an experimental approach or simulations based on a simplified model.

Although DNS is computationally expensive and limited to low Reynolds numbers and relatively simple flows, its coupling with CMSM, advances in computer hardware and the possibility of parallel computing have opened the door for its application to a wider range of problems. The adaptation of this method to more complex geometries has been analyzed and performed by Jacobs[24], and he drew the conclusion that CMSM offered all the features DNS required, and therefore was the suitable numerical scheme for his work. In this thesis all the simulations are performed using the three-dimensional version of CMSM.

1.6 Contributions and Outline

The main contributions of this thesis are the implementation of a mesh generator and the analysis of a subsonic, single-phase, cold flow in a dump combustor.

The geometry taken into consideration is constituted by a backward-facing step followed by a ramp. A mesh generator based on previous work by Komperda[25] at CMTL has been implemented in Python, to allow the creation of mesh for different shapes, which have been eventually combined into the final channel geometry.

To test the effective functionality of the numerical three-dimensional code developed in the CMTL, based on CMSM, simulations of the flow over a backward-facing step have been performed, and results have been compared to those derived from experiments found in literature.

Two different types of boundary conditions have been imposed at the outlet section. The first type, already available in the code, is based on the specification of the outflow u velocity profile, together with other variables set as their free-stream values. The second type, instead, has been implemented with this work and is represented by the transmissive (or zero-gradient)

boundary condition, that forces each variable to remain constant along the x direction from the last domain section on.

After confirming the code's reliability with the simulations over the backward-facing step, the flow in a dump combustor, obtained by adding a ramp after the step, is then analyzed. In the literature such a geometry has been widely studied for supersonic flows, while little or no research has been conducted for laminar flows, to the best of the author's knowledge. This work, therefore, is also an opportunity to describe in detail laminar incompressible flows in the dump combustor geometry, and provide a comparison between the results obtained imposing two different types of outflow boundary conditions.

In Chapter 2, the governing equations of fluid dynamics used to describe the flow are presented. Chapter 3 is dedicated to the description of the numerical code used for the simulations, and the equations introduced in Chapter 2 are discretized and modeled numerically. In Chapter 4, the mesh generator and its main features are described. Chapter 5 describes in detail boundary conditions used for the simulations, and it presents results for both the backward-facing step and the backward-facing step with the ramp. Finally, Chapter 6 draws conclusions and contains suggestions for future work.

CHAPTER 2

GOVERNING EQUATIONS

2.1 Compressible Navier-Stokes Equations - Dimensional Form

The fundamental governing equations of fluid dynamics are the continuity, momentum and energy equations. They are the mathematical statements of three fundamental physics principles[26]:

1. mass is conserved;
2. $F = ma$, where F indicates the force, m the mass, and a the acceleration (Newton's second law);
3. energy is conserved.

These equations, applied to fluid mechanics, are grouped under the name of Navier-Stokes equations. All of theoretical and computational fluid dynamics is based on these equations. In this section, they will be presented in dimensional form, which of course implies that all the variables are taken with their appropriate dimensional units.

2.1.1 Mass Conservation

The first equation to be taken into consideration is the mass conservation equation, also known as the continuity equation. Its integral form is obtained by referring to a control volume Ω and its delimiting surface S, whose normal in every point is indicated with $\mathbf{n}$. In particular,

referring to an infinitesimal surface element dS and its normal $\mathbf{n}$, as shown in figure 4, it is possible to apply the conservation of mass, and then extend it over the whole domain considered

$$\frac{\partial m_\Omega}{\partial t} = \frac{\partial}{\partial t}\int_\Omega \rho \, d\Omega = -\int_S \rho \mathbf{v} \cdot \mathbf{n} \, dS. \tag{2.1}$$

The surface integral can be transformed into a volume integral by applying Gauss divergence

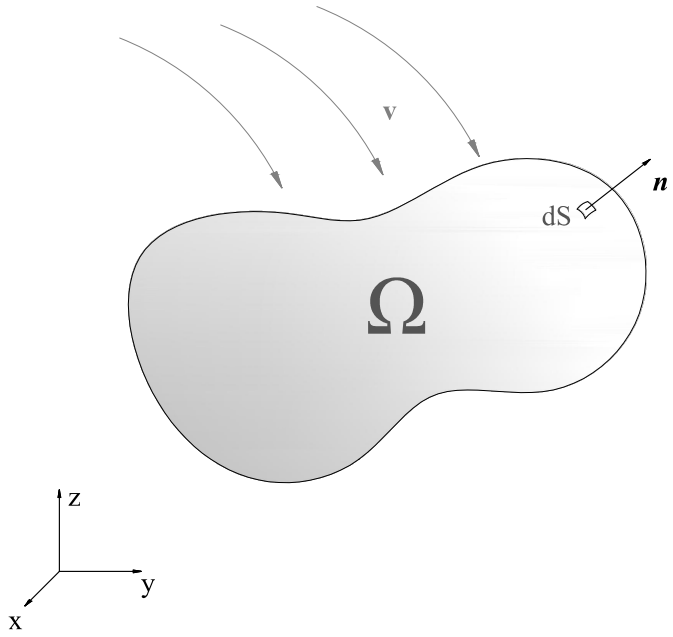


Figure 4: Schematic representation of a generic control volume Ω, an infinitesimal surface element dS and its normal $\mathbf{n}$, immersed in a flow with velocity $\mathbf{v}$.

theorem, and brought for convenience to the left hand side. This procedure leads to a differential coordinate-free expression of mass conservation

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0. \tag{2.2}$$

(Equation 2.2) is suitable for different coordinate systems, since there's just the need to apply a divergence expression consistent with the coordinates chosen. The two most convenient conventions for the purpose of this thesis are the Cartesian coordinate system and the Einstein notation, respectively adopted in the expressions below

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} + \frac{\partial \rho w}{\partial z} = 0, \tag{2.3}$$

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_i}{\partial x_i} = 0, \tag{2.4}$$

where $i = 1, 2, 3$, $x_i = x, y, z$ represents the Cartesian coordinates and $u_i = u, v, w$ represents the Cartesian components of the velocity vector $\mathbf{v}$.

2.1.2 Momentum Conservation

In order to derive an expression for the momentum conservation, a procedure similiar to the former has to be followed. Referring again to a control volume Ω and its delimiting surface S, the expression stating the momentum conservation in integral form is

$$\frac{\partial (m_\Omega \mathbf{v}_\Omega)}{\partial t} = \frac{\partial}{\partial t} \int_\Omega \rho \mathbf{v} \, d\Omega = - \oint_S \mathbf{F}(\rho \mathbf{v}) \cdot \mathbf{n} \, dS + \int_\Omega \mathbf{s}(\rho \mathbf{v}) \, d\Omega. \tag{2.5}$$

The right hand side contains all the forces that act on the volume considered, which are divided into:

- Surface forces, $\mathbf{F}$, which are those forces acting on the external surface S of the volume considered, such as pressure, normal and shear stresses, etc.;
- Body forces, $\mathbf{s}$, which are forces that act on the volume as a whole, such as gravity, centrifugal forces, etc..

Under the assumption of considering a Newtonian fluid, and in the absence of any source term or other body forces (last integral in (Equation 2.5) is then taken equal to zero), which is appropriate for the problems considered in this thesis, the right hand side can be reformulated in a more convenient expression, through the following splitting:

$$\mathbf{F}(\rho\mathbf{v}) = \mathbf{F}_{adv} + \mathbf{F}_{stat} + \mathbf{F}_{dyn},$$

where:

- $\mathbf{F}_{adv}$ is the advection flux, whose formulation is

$$\mathbf{F}_{adv}(\rho\mathbf{v}) = \rho\mathbf{v} \otimes \mathbf{v},$$

  and whose dimensional units are those of a force per unit area, i.e. a mechanical stress;

- $\mathbf{F}_{stat}$ is the hydrostatic flux, given by

$$\mathbf{F}_{stat}(\rho\mathbf{v}) = p\mathbf{I},$$

  where p is the pressure, and $\mathbf{I}$ is the identity matrix;

- $\mathbf{F}_{dyn}$ is the hydrodynamic flux, given by

$$\mathbf{F}_{dyn}(\rho\mathbf{v}) = -\mathbf{\Pi}_{\nu} = -\rho\nu\bigg(\nabla\times\mathbf{v} + (\nabla\times\mathbf{v})^T - \frac{2}{3}\nabla\cdot\mathbf{v}\,\mathbf{I}\bigg),$$

  where ν is the kinematic viscosity of the fluid.

Hydrostatic and hydrodynamic fluxes can be grouped together under the stress tensor $\mathbf{\Pi}$

$$\mathbf{\Pi} = p\mathbf{I} - \mathbf{\Pi}_{\nu}. \tag{2.6}$$

It is now convienent to present a more comfortable reformulation of (Equation 2.5), neglecting the source term, that has been assumed equal to zero

$$\frac{\partial}{\partial t}\int_{\Omega}\rho\mathbf{v}\,d\Omega + \int_{S}(\rho\mathbf{v}\otimes\mathbf{v})\cdot\mathbf{n}\,dS = -\int_{S}\mathbf{\Pi}\cdot\mathbf{n}\,dS, \tag{2.7}$$

or, dividing hydrostatic and hydrodynamic fluxes contributions,

$$\frac{\partial}{\partial t}\int_{\Omega}\rho\mathbf{v}\,d\Omega + \int_{S}(\rho\mathbf{v}\otimes\mathbf{v})\cdot\mathbf{n}\,dS = -\int_{S}p\mathbf{I}\cdot\mathbf{n}\,dS + \int_{S}\mathbf{\Pi}_{\nu}\cdot\mathbf{n}\,dS. \tag{2.8}$$

Applying the mass conservation equation, (Equation 2.8) can be rewritten in coordinate-free vector form by the application of Gauss divergence theorem

$$\frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v}) = -\nabla p + \nabla \cdot \mathbf{\Pi}_{\nu}. \tag{2.9}$$

Since momentum is a vector quantity, it can be either represented along the Cartesian coordinates

$$\frac{\partial \rho u}{\partial t} + \nabla \cdot (\rho u \mathbf{v}) = -\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} + \frac{\partial \tau_{xz}}{\partial z}, \tag{2.10}$$

$$\frac{\partial \rho v}{\partial t} + \nabla \cdot (\rho v \mathbf{v}) = -\frac{\partial p}{\partial y} + \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{yz}}{\partial z}, \tag{2.11}$$

$$\frac{\partial \rho w}{\partial t} + \nabla \cdot (\rho w \mathbf{v}) = -\frac{\partial p}{\partial z} + \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z}, \tag{2.12}$$

or with Einstein notation

$$\rho \frac{\partial u_i}{\partial t} + \nabla \cdot (\rho u_i \mathbf{v}) = -\frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j}, \tag{2.13}$$

where again $i, j = 1, 2, 3$, $u_i = u, v, w$ are the components of the velocity vector $\mathbf{v}$ along the Cartesian coordinates $x_i = x, y, z$, and τ_{ij} are the components of the viscous stress tensor $\mathbf{\Pi}_{\nu}$.

2.1.3 Total Energy Conservation

The third fundamental equation of fluid dynamics states the conservation of total energy, given by the sum of kinetic, potential and internal energy. The specific total energy for the control volume Ω is defined as

$$e_\Omega = \frac{E_t}{m_\Omega} = \frac{1}{m_\Omega} \int_\Omega \rho e_t \, d\Omega \tag{2.14}$$

where E_t is the total energy of the system considered, while e_t is the total energy per unit volume. Following the same procedure for the other two conservation equations considered, it is possible to write the integral formulation for the conservation of total energy

$$\frac{\partial (m_\Omega e_\Omega)}{\partial t} = \frac{\partial}{\partial t} \int_\Omega \rho e_t \, d\Omega = - \oint_S \mathbf{f}(\rho e_t) \cdot \mathbf{n} \, dS \tag{2.15}$$

where $\mathbf{f}(\rho e_t)$ is the total energy flux, which can be split into three terms, similarly to the previous case. In particular

$$\mathbf{f}(\rho e_t) = \mathbf{f}_{adv} + \mathbf{f}_{therm} + \mathbf{f}_{mech},$$

where

- $\mathbf{f}_{adv}$, similarly to the previous case, is the advection flux, whose formulation is

$$\mathbf{f}_{adv}(\rho e_t) = \rho e_t \mathbf{v};$$

- $\mathbf{f}_{therm}$ is the thermal flux, that can be expressed using Fourier's Law as

$$\mathbf{f}_{therm}(\rho e_t) = \mathbf{q}_\alpha = -\lambda \nabla T = -\rho c_p \alpha \nabla T,$$

  where λ is the thermal conductivity $[W/(m\,K)]$, while α is the thermal diffusivity $[m^2/s]$, linked by the relation $\alpha = \lambda/(\rho c_p)$;

- $\mathbf{f}_{mech}$ is the mechanical flux, that can be expressed by means of the stress tensor, namely

$$\mathbf{f}_{mech}(\rho e_t) = \mathbf{\Pi} \cdot \mathbf{v}.$$

A more convenient reformulation of (Equation 2.15) may now be presented:

$$\frac{\partial}{\partial t} \int_\Omega \rho e_t \, d\Omega = -\int_S \rho e_t \mathbf{u} \cdot \mathbf{n} \, dS - \int_S \mathbf{q}_\alpha \cdot \mathbf{n} \, dS - \int_S (p\mathbf{I} - \mathbf{\Pi}_\nu) \cdot \mathbf{v} \, dS. \tag{2.16}$$

In (Equation 2.16) expressions for advective, thermal and mechanical fluxes have been substituted, and the stress tensor has been divided into hydrostatic and hydrodynamic components, through (Equation 2.6). Applying once again the Gauss divergence theorem, the above expression can be transformed into

$$\frac{\partial(\rho e_t)}{\partial t} + \nabla \cdot (\rho e_t \mathbf{v}) = -\nabla \cdot (p\mathbf{v}) - \nabla \cdot \mathbf{q}_\alpha + \nabla \cdot \mathbf{\Pi}_\nu, \tag{2.17}$$

that is the final conservation equation for total energy. This last expression can also be rewritten for Cartesian coordinates, for consistency with what presented for the other conservation equations

$$\frac{\partial(\rho e_t)}{\partial t}+\frac{\partial(\rho e_t u)}{\partial x}+\frac{\partial(\rho e_t v)}{\partial y}+\frac{\partial(\rho e_t w)}{\partial z}=-\frac{\partial(up)}{\partial x}-\frac{\partial(vp)}{\partial y}-\frac{\partial(wp)}{\partial z}+$$

$$+\frac{\partial}{\partial x}\left(\lambda\frac{\partial T}{\partial x}\right)+\frac{\partial}{\partial y}\left(\lambda\frac{\partial T}{\partial y}\right)+\frac{\partial}{\partial z}\left(\lambda\frac{\partial T}{\partial z}\right)+$$

$$+\frac{\partial(u\tau_{xx})}{\partial x}+\frac{\partial(u\tau_{xy})}{\partial y}+\frac{\partial(u\tau_{xz})}{\partial z}+\frac{\partial(v\tau_{xy})}{\partial x}+\frac{\partial(v\tau_{yy})}{\partial y}+$$

$$+\frac{\partial(v\tau_{yz})}{\partial z}+\frac{\partial(w\tau_{xz})}{\partial x}+\frac{\partial(w\tau_{yz})}{\partial y}+\frac{\partial(w\tau_{zz})}{\partial x}. \tag{2.18}$$

The components τ_{ij} of the stress tensor in Cartesian coordinates are defined by the following relations:

$$\tau_{xx}=2\rho\nu\left(\frac{\partial u}{\partial x}-\frac{1}{3}\nabla\cdot\mathbf{v}\right),$$
$$\tau_{yy}=2\rho\nu\left(\frac{\partial v}{\partial y}-\frac{1}{3}\nabla\cdot\mathbf{v}\right),$$
$$\tau_{zz}=2\rho\nu\left(\frac{\partial w}{\partial z}-\frac{1}{3}\nabla\cdot\mathbf{v}\right),$$
$$\tau_{xy}=\tau_{yx}=\rho\nu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right),$$
$$\tau_{yz}=\tau_{zy}=\rho\nu\left(\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}\right),$$
$$\tau_{xz}=\tau_{zx}=\rho\nu\left(\frac{\partial w}{\partial x}+\frac{\partial u}{\partial z}\right).$$

2.1.4 Assumptions

In this work, certain assumptions concerning conservation equations are made, and they are summarized below:

- The fluid is assumed to be Newtonian, which means that viscous stresses arisen from the flow are considered linearly proportional to the local strain rate at every point;
- The heat flux can be modeled using Fourier's law, i.e. the heat flux in each direction is proportional to the directional derivative of temperature along the same direction, and the constant of proportionality is the thermal conductivity λ of the fluid;
- Viscosity, thermal conductivity, thermal diffusivity and specific heats of the fluid are assumed to be constant and independent from the temperature of the fluid;
- The fluid is assumed continuous, so that properties such as density, pressure, temperature and velocity are considered well-defined at infinitesimally small points;
- The fluid has zero bulk viscosity ('Stokes hypothesis'):

$$\mu_b = \epsilon + \frac{2}{3}\mu = 0, \tag{2.19}$$

where μ_b is the bulk viscosity, μ is the dynamic viscosity coefficient and ϵ is a secondary viscosity coefficient. Being $\mu_b = 0$, the two coefficients can be grouped into one, which will be substituted taking also into account that $\mu = \rho\nu$;

- The fluid is assumed to be an ideal gas, so that intermolecular forces are negligible, and it is described by the equation of state

$$p = \rho R T, \tag{2.20}$$

where R $[J/(Kg\,K)]$ is the gas constant. Navier-Stokes equations are coupled with (Equation 2.20), so that the system counts six equations for six unknows: ρ, u, v, w, p and T.

2.2 Compressible Navier-Stokes Equations - Non-Dimensional Form

Even if the equations presented so far give a full theoretical background for describing any fluid in motion, they aren't yet in the most convenient form for computational fluid dynamics analysis. The next step that has to be accomplished is their non-dimensionalization, i.e. the removal of dimensional units associated to physical quantities through an appropriate substitution of variables. To do so, independent variables have to be isolated and scaled with respect to a reference quantity to be determined, while dependent variables are scaled through combination of previously defined reference quantities.

Therefore, all variables are non-dimensionalized by means of

- l_f reference length;
- ρ_f reference density;
- v_f reference velocity;
- T_f reference temperature.

The non-dimensionalization of variables is shown below:

$$x = \frac{x^*}{l_f}, \qquad y = \frac{y^*}{l_f}, \qquad z = \frac{z^*}{l_f},$$
$$u = \frac{u^*}{v_f}, \qquad v = \frac{v^*}{v_f}, \qquad w = \frac{w^*}{v_f},$$
$$\rho = \frac{\rho^*}{\rho_f}, \qquad T = \frac{T^*}{T_f}, \qquad p = \frac{p^*}{\rho_f v_f^2}$$
$$t = \frac{t^* v_f}{l_f}.$$

It is important to remark that the reference time is a dependent variable, since it is expressed as a function of reference length and velocity, i.e. $t_f = l_f/v_f$. For sake of convenience, all dimensional variables have here been highlighted with a $*$ power, so that in the following non-dimensionalized variables can be simply recalled using their traditional symbol.

2.2.1 Mass Conservation

Applying the substitution of variables described in the previous paragraph, equation of conservation of mass becomes

$$\frac{\rho_f}{t_f}\frac{\partial \rho}{\partial t} + \frac{\rho_f v_f}{l_f}\nabla \cdot (\rho \mathbf{v}) = 0, \tag{2.21}$$

with $t_f = l_f/v_f$, that leads to

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0, \tag{2.22}$$

where all the physical quantities involved are now non-dimensionalized with respect to their reference value.

2.2.2 Momentum Conservation

The same procedure is now applied to momentum conservation equation

$$\frac{\rho_f v_f}{t_f}\frac{\partial \rho \mathbf{v}}{\partial t} + \frac{\rho_f v_f^2}{l_f}\nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v}) = -\frac{\rho_f v_f^2}{l_f}\nabla p + \frac{\rho_f \nu v_f}{l_f^2}\nabla \cdot \mathbf{\Pi}_\nu, \tag{2.23}$$

that can be reorganized in the more convenient formulation

$$\frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v}) = -\nabla p + \frac{1}{Re_f}\nabla \cdot \mathbf{\Pi}_\nu. \tag{2.24}$$

2.2.3 Total Energy Conservation

The last equation to non-dimensionalize is the energy conservation equation

$$\frac{\rho_f v_f^2}{t_f}\frac{\partial (\rho e_t)}{\partial t} + \frac{\rho_f v_f^2 v_f}{l_f}\nabla \cdot (\rho e_t \mathbf{v}) = -\frac{\rho_f v_f^2 v_f}{l_f}\nabla \cdot (p\mathbf{v}) - \frac{\rho_f c_p \alpha}{T_f l_f^2}\nabla \cdot \mathbf{q}_\alpha + \frac{\rho_f \nu v_f}{l_f^2}\nabla \cdot \mathbf{\Pi}_\nu, \tag{2.25}$$

that also may be rewritten in a most comfortable way as

$$\frac{\partial (\rho e_t)}{\partial t} + \nabla \cdot (\rho e_t \mathbf{v}) = -\nabla \cdot (p\mathbf{v}) - \frac{1}{(\gamma - 1)M_f^2 \, Pr \, Re_f}\nabla \cdot \mathbf{q}_\alpha + \frac{1}{Re_f}\nabla \cdot \mathbf{\Pi}_\nu. \tag{2.26}$$

In the previous equations, the following expression for total energy has been applied:

$$\rho e_t = \frac{p}{\gamma - 1} + \rho \frac{u^2 + v^2 + w^2}{2}. \tag{2.27}$$

2.2.4 Additional Considerations

In (Equation 2.24) and (Equation 2.26) some non-dimensional numbers have been used to correlate reference and dimensional quantities in a more convenient notation:

- Re_f indicates the reference Reynolds number, a non-dimensional parameter whose definition $Re_f = (l_f v_f)/\nu$ involves kinematic viscosity, reference length and velocity, and gives a measure of inertial (advective) forces as compared to viscous forces;
- M_f indicates the reference Mach number, defined as $M_f = v_f/c_f$, where c_f is the reference speed of sound defined as $c_f = \sqrt{\gamma R T_f}$, and γ is the ratio between constant pressure specific heat and constant volume specific heat. In this work M_f is assumed equal to unity, so that the reference velocity is equal to the reference speed of sound;
- Pr indicates the Prandtl number, given by $Pr = (c_p \rho \nu)/\lambda$, which measures the ratio of viscous and thermal diffusivity.

All previous equations are coupled once again with the equation of state, that for an ideal gas in non-dimensional form reads

$$p = \frac{\rho T}{\gamma M_f^2}. \tag{2.28}$$

Since, as previously said, in this work M_f is taken equal to unity, the equation of state may be rewritten as

$$p = \frac{\rho T}{\gamma}. \tag{2.29}$$

It is important to remark the difference between the reference Mach number M_f and the flow Mach number, indicated with just the symbol M. The former, in fact, is related both to reference velocity and reference speed of sound, while the latter is related to the actual speed of sound. In fact, it is defined as $M = v_f/c^*$, where $c^* = \sqrt{\gamma R T^*}$, where it has to be reminded that dimensional quantities have been highlighted with a $*$ power. The flow Mach number, which does not appear explicitly in the non-dimensional equations, is a local quantity in the flow.

2.3 Matrix Form

Now that all the fundamental equations have been presented in their non-dimensional form, it is convenient to group them into a matrix form, that is the most suitable for a CFD application. Therefore, all conservation equations may be reunited under the following relation:

$$\frac{\partial \mathbf{W}}{\partial t} + \frac{\partial \mathbf{F_f}}{\partial x} + \frac{\partial \mathbf{G_f}}{\partial y} + \frac{\partial \mathbf{H_f}}{\partial z} = \frac{1}{Re_f}\left[\left(\frac{\partial \mathbf{F}_\nu}{\partial x} + \frac{\partial \mathbf{G}_\nu}{\partial y} + \frac{\partial \mathbf{H}_\nu}{\partial z}\right) + \left(\frac{\partial \mathbf{F_T}}{\partial x} + \frac{\partial \mathbf{G_T}}{\partial y} + \frac{\partial \mathbf{H_T}}{\partial z}\right)\right]. \quad (2.30)$$

Vectors constituting (Equation 2.30) can be subdivided in

- Vector of conserved quantities, $\mathbf{W}$

$$\mathbf{W} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho e_t \end{pmatrix};$$

- Vectors that group inviscid (advection and hydrostatic) fluxes along the Cartesian coordinates, $\mathbf{F_f}$, $\mathbf{G_f}$ and $\mathbf{H_f}$

$$\mathbf{F_f} = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ \rho e_t u + pu \end{pmatrix}, \qquad \mathbf{G_f} = \begin{pmatrix} \rho v \\ \rho vu \\ \rho v^2 + p \\ \rho vw \\ \rho e_t v + pv \end{pmatrix}, \qquad \mathbf{H_f} = \begin{pmatrix} \rho w \\ \rho wu \\ \rho wv \\ \rho w^2 + p \\ \rho e_t w + pw \end{pmatrix};$$

- Vectors that group viscous fluxes along the Cartesian coordinates, $\mathbf{F}_\nu$, $\mathbf{G}_\nu$ and $\mathbf{H}_\nu$

$$\mathbf{F}_\nu = \begin{pmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ \tau_{xz} \\ u\tau_{xx} + v\tau_{xy} + w\tau_{xz} \end{pmatrix}, \qquad \mathbf{G}_\nu = \begin{pmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ \tau_{yz} \\ u\tau_{xy} + v\tau_{yy} + w\tau_{yz} \end{pmatrix},$$

$$\mathbf{H}_\nu = \begin{pmatrix} 0 \\ \tau_{xz} \\ \tau_{yz} \\ \tau_{zz} \\ u\tau_{xz} + v\tau_{yz} + w\tau_{zz} \end{pmatrix}.$$

- Vectors that group the influence of temperature, $\mathbf{F_T}$, $\mathbf{G_T}$, $\mathbf{H_T}$

$$\mathbf{F_T} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{(\gamma-1)M_f^2\,Pr}\frac{\partial T}{\partial x} \end{pmatrix}, \quad \mathbf{G_T} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{(\gamma-1)M_f^2\,Pr}\frac{\partial T}{\partial y} \end{pmatrix}, \quad \mathbf{H_T} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{(\gamma-1)M_f^2\,Pr}\frac{\partial T}{\partial z} \end{pmatrix}.$$

The components τ_{ij} of the stress tensor are here defined as

$$\tau_{xx} = 2\left(\frac{\partial u}{\partial x} - \frac{1}{3}\nabla \cdot \mathbf{v}\right),$$
$$\tau_{yy} = 2\left(\frac{\partial v}{\partial y} - \frac{1}{3}\nabla \cdot \mathbf{v}\right),$$
$$\tau_{zz} = 2\left(\frac{\partial w}{\partial z} - \frac{1}{3}\nabla \cdot \mathbf{v}\right),$$
$$\tau_{xy} = \tau_{yx} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x},$$
$$\tau_{yz} = \tau_{zy} = \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y},$$
$$\tau_{xz} = \tau_{zx} = \frac{\partial w}{\partial x} + \frac{\partial u}{\partial z}.$$

In the following chapter, numerical methodology will be described in detail, and also the integration between the code and the equations just presented will be explained.

CHAPTER 3

NUMERICAL METHODOLOGY

3.1 Space Discretization

(Equation 2.30) has to be discretized in space, and the code used employs the staggered-grid Chebyshev multidomain spectral method (CMSM) introduced by Kopriva [21] [22] [23] for two-dimensional problems. Jacobs [24] extended CMSM to three dimensional problems, and this procedure is described in the chapter.

3.1.1 Isoparametric Mapping

If compressible flows are modeled with Fourier spectral methods, severe restrictions have to be taken into account.

A significant alternative to the standard method is a multidomain method, such as CMSM. This kind of methods is based on the subdivision of the computational domain into multiple zones, called subdomains, where the spectral approximation is then applied. Because of its subdivision onto multiple domains, this method is appropriate for the analysis of more complex geometries. These subdomains are non-overlapping regions, so that the whole computational domain D, subdivided into n parts, is given by

$$D = D_1 \cup D_2 \cup ... \cup D_n, \tag{3.1}$$

where D_i indicates the i-th subdomain. These subdomains are quadrilaterals in two dimensions, while hexahedrons in three dimensions.

Chebyshev polynomial is defined on the interval $[-1, 1]$. In order to use this polynomial on a generic subdomain D_i, therefore, it is necessary to map the domain to that interval in all its spatial dimensions. Kopriva's method maps the subdomain to the more convenient interval $[0, 1]$, rather than to the natural interval $[-1, 1]$. This operation is performed with isoparametric mapping, so that spectral accuracy is not influenced, which is very important considering the high approximation order used.

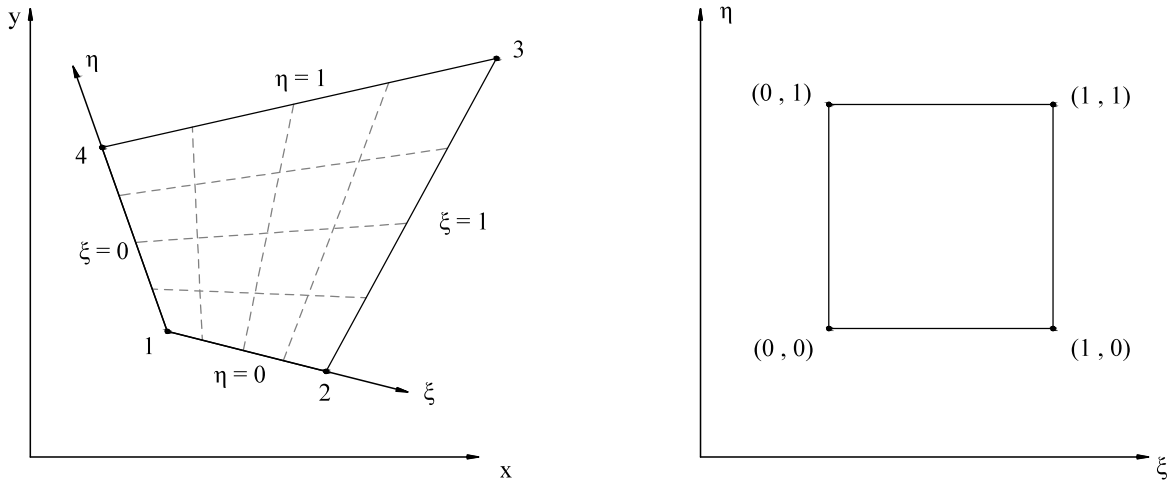


Figure 5: Isoparametric mapping for a two dimensional shape in plane (x, y) to a unit square in plane (ξ, η).

Through isoparametric mapping, every curved element in two or three dimensional physical space can be mapped to a unit square or cube, respectively, allowing the analysis of any complex geometry possible, as shown in figure 5.

Considering the two dimensional case shown in the example, a generic shape in physical space, plane (x, y), can be mapped to a unit square in plane (ξ, η), through the coordinate transformation shown in table I. The bottom-left point of the shape, indicated in figure 5 as 1, is assumed as the origin of the new coordinate system (ξ, η), whereas the sides of the shape are chosen as the lines where the two new coordinates, ξ and η, assume the values of 0 and of 1. Therefore, every corner of the quadrilateral in plane (x, y) is mapped to a vertex of the unit square in plane (ξ, η). In the following, mapping for two and three dimensional cases is analyzed in detail. For the sake of notational convenience, the mapped space will be referred to as (X, Y), instead of as (ξ, η).

TABLE I: ISOPARAMETRIC MAPPING FROM PHYSICAL SPACE (X, Y) TO MAPPED SPACE (ξ, η).

Point	(x, y) coordinates	(ξ, η) coordinates
1	(x_1, y_1)	$(0, 0)$
2	(x_2, y_2)	$(1, 0)$
3	(x_3, y_3)	$(1, 1)$
4	(x_4, y_4)	$(0, 1)$

3.1.1.1 Two-Dimensional Mapping

The parametric curve $g(s)$, with $0 \leq s \leq 1$, is interpolated by the polynomial $P(s)$, which has the same degree as the approximation order within the domain. In particular, the interpolating polynomial P is defined as a linear combination of the fundamental polynomials

$$P(s) = \sum_{j=0}^{N} g_j l_j(s), \tag{3.2}$$

where $l_j(s)$ are the fundamental Lagrangian polynomials, which constitute a base of the polynomials space, and are given by

$$l_j(s) = \prod_{i=0, i \neq j}^{N} \frac{s - x_i}{x_j - x_i}. \tag{3.3}$$

The pecularity of Lagrangian polynomials is that their value is zero in all the nodes except the one they are calculated in, where they assume value 1. In fact, the j-th fundamental Lagrangian polynomial calculated in a generic node reads

$$l_j(x) = \frac{(x - x_1)...(x - x_{j-1})(x - x_{j+1})...(x - x_n)}{(x_j - x_1)...(x_j - x_{j-1})(x_j - x_{j+1})...(x_j - x_n)}, \qquad j = 1, ..., n.$$

If calculated in the node x_j, the polynomial will reduce to

$$l_j(x_j) = \frac{(x_j - x_1)...(x_j - x_{j-1})(x_j - x_{j+1})...(x_j - x_n)}{(x_j - x_1)...(x_j - x_{j-1})(x_j - x_{j+1})...(x_j - x_n)} = 1.$$

It is easy to recognize that, if the Lagrangian polynomial is calculated in a node x_i with $i \neq j$, it has value 0. In other words, Lagrangian polynomials are such that

$$l_j(s) = \delta_{ij} = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } i \neq j. \end{cases} \tag{3.4}$$

The two dimensional mapping is then performed through a linear blending formula [27], applying the equations presented so far, for both x and y directions

$$\begin{aligned} x(X,Y) = \ & (1-Y)P_1(X) + XP_2(Y) + YP_3(X) + (1-X)P_4(Y) + \\ & - x_1(1-X)(1-Y) - x_2X(1-Y) - x_3XY - x_4(1-X)Y. \end{aligned} \tag{3.5}$$

$$\begin{aligned} y(X,Y) = \ & (1-Y)P_1(X) + XP_2(Y) + YP_3(Y) + (1-X)P_4(Y) + \\ & - y_1(1-X)(1-Y) - y_2X(1-Y) - y_3XY - y_4(1-X)Y. \end{aligned} \tag{3.6}$$

In (Equation 3.2) to (Equation 3.6), $P_i(s)$ indicates the polynomial interpolating the curve $g(s)$ referred to the i-th side of the domain, x_i and y_i refer to the number of the corner, while X_i denote the boundary grid points of a particular domain. For convex quadrilaterals, where no inside corner is more than 180 degrees, (Equation 3.2) is regular, which implies the Jacobian of the mapping to the unit region is non-zero. As it can be seen in figure 6, sides and corners of the unit square have been numbered accordingly to what presented in (Equation 3.5) and in (Equation 3.6).

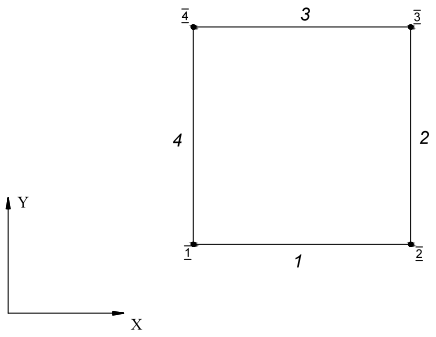


Figure 6: Node and side numbering for a two-dimensional case.

3.1.1.2 Three-Dimensional Mapping

In case of three-dimensional mapping, the procedure to be followed is analogous to that previously applied, with some adjustements to extend it to a three dimensional problem. In particular, aside from the fundamental Lagrange polynomials $P_i(s)$, whose expression is given by (Equation 3.2), it is necessary to introduce the $S_i(s)$ polynomial also

$$S(s,t) = \sum_{i=0}^{N} \sum_{j=0}^{N} g_{ij} l_i(s) l_j(t), \tag{3.7}$$

where $g(s,t)$ interpolates the boundary surfaces at the grid points. The linear blending formula for a three-dimensional case reads [24]

$$\begin{aligned} x(X,Y,Z) = \; & (1-Y)S_1(X,Z) + YS_2(X,Z) + (1-Z)S_3(X,Y) + \\ & + XS_4(Y,Z) + ZS_5(X,Y) + (1-X)S_6(Y,Z) + \\ & - (1-Y)(1-Z)P_1(X) - X(1-Y)P_2(Z) - (1-Y)ZP_3(X) + \\ & - (1-X)(1-Y)P_4(Z) - Y(1-Z)P_5(X) - XYP_6(Z) + \\ & - YZP_7(X) - (1-X)YP_8(Z) - (1-X)(1-Z)P_9(Y) + \\ & - X(1-Z)P_{10}(Y) - XZP_{11}(Y) - (1-X)ZP_{12}(Y) + \\ & + (1-X)(1-Y)(1-Z)x_1 + X(1-Y)(1-Z)x_2 + XY(1-Z)x_3 + \\ & + (1-X)Y(1-Z)x_4 + (1-X)(1-Y)Zx_5 + X(1-Y)(1-Z)x_6 + \\ & + XYZx_7 + (1-X)YZx_8. \end{aligned} \tag{3.8}$$

The subscripts on S (lines 1-2 in (Equation 3.8)), P (lines 3-6) and x (lines 7-9) indicate the surfaces (1 to 6), the sides (1 to 12) and the corners (1 to 8) of the domain, respectively. Equations for $y(X,Y,Z)$ and $z(X,Y,Z)$ can be obtained directly from (Equation 3.8) replacing x_i on the right-hand side with y_i or z_i respectively, therefore their formulation is omitted here. A graphical representation of the elements' numbering is shown in figure 7, where the left representation shows the top, front, and right face numbering, while the right representation highlights the bottom, right and rear faces. The bigger font has been used to indicate the

faces' numbers, while the sides have been indicated with smaller italic numbers, and corners' numbering is underlined and overlined.

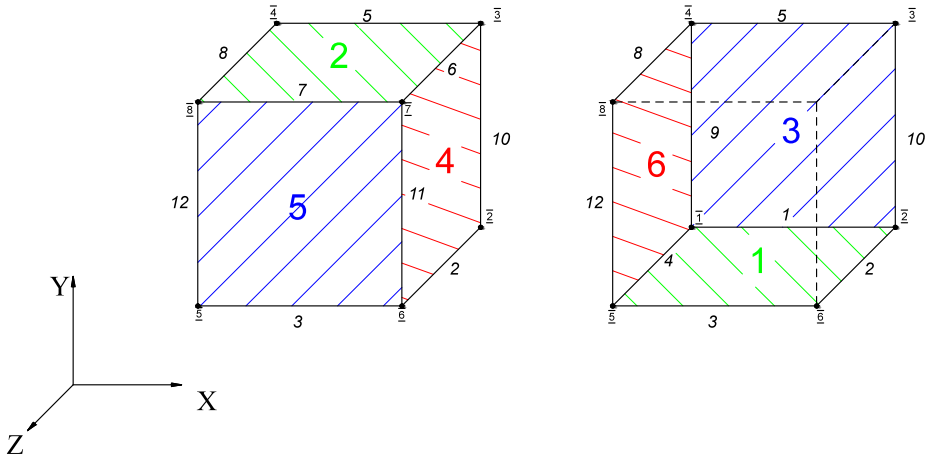


Figure 7: Node, side and surface numbering for a three-dimensional case.

3.1.1.3 Mapped Equations

With the isoparametric mapping just performed, (Equation 2.30) reduces to [28]

$$\frac{\partial \tilde{\mathbf{W}}}{\partial t} + \frac{\partial \tilde{\mathbf{F}}}{\partial X} + \frac{\partial \tilde{\mathbf{G}}}{\partial Y} + \frac{\partial \tilde{\mathbf{H}}}{\partial Z} = 0, \tag{3.9}$$

where $\sim$ indicates a mapped vector. (Equation 3.9) has been obtained using the chain rule, the inverse of the physical space gradient and the following relations:

$$\tilde{\mathbf{W}} = J\mathbf{W}, \tag{3.10}$$

and

$$\tilde{\mathbf{F}} = \frac{\partial X}{\partial x}\mathbf{F} + \frac{\partial X}{\partial y}\mathbf{G} + \frac{\partial X}{\partial x}\mathbf{H}, \tag{3.11}$$

$$\tilde{\mathbf{G}} = \frac{\partial Y}{\partial x}\mathbf{F} + \frac{\partial Y}{\partial y}\mathbf{G} + \frac{\partial Y}{\partial z}\mathbf{H}, \tag{3.12}$$

$$\tilde{\mathbf{H}} = \frac{\partial Z}{\partial x}\mathbf{F} + \frac{\partial Z}{\partial y}\mathbf{G} + \frac{\partial Z}{\partial z}\mathbf{H}. \tag{3.13}$$

The Jacobian of the mapping, J, is defined as follows:

- For a two-dimensional case

$$J(x,y) = \frac{\partial x}{\partial X}\frac{\partial y}{\partial Y} - \frac{\partial x}{\partial Y}\frac{\partial y}{\partial X}; \tag{3.14}$$

- For a three-dimensional case

$$\begin{aligned} J(x,y,z) =& \frac{\partial x}{\partial X}\left(\frac{\partial y}{\partial Y}\frac{\partial z}{\partial Z} - \frac{\partial y}{\partial Z}\frac{\partial z}{\partial Y}\right) - \frac{\partial x}{\partial Y}\left(\frac{\partial y}{\partial X}\frac{\partial z}{\partial Z} - \frac{\partial y}{\partial Z}\frac{\partial z}{\partial X}\right) + \\ &+ \frac{\partial x}{\partial Z}\left(\frac{\partial y}{\partial X}\frac{\partial z}{\partial Y} - \frac{\partial y}{\partial Y}\frac{\partial z}{\partial X}\right). \end{aligned} \tag{3.15}$$

3.1.2 Staggered-Grid Spectral Method

In the following, a staggered-grid spectral method is described in detail. Its description will move from a one-dimensional case, that will serve as basis to explain the basic conscepts of the method, to the complete solution of three-dimensional Navier Stokes equations, following the same procedure explained by Jacobs[24].

3.1.2.1 One Dimensional Case

Considering a one dimensional domain, D, it is divided into n non-overlapping segments, so that

$$D = D_1 \cup D_2 \cup ... \cup D_n.$$

For every subdomain obtained, solution values W are approximated through a truncated Chebyshev expansion

$$W_a(x) = \sum_{i=1}^{n} W_i T_i(x), \tag{3.16}$$

where $T_i(x)$ is the Chebyshev polynomial, defined as

$$T_i(x) = cos[i \cdot cos(x)]. \tag{3.17}$$

(Equation 3.17) represents the trial function. This means that, substituting the approximation given by (Equation 3.16) into the differential equation our system is described by, it doesn't lead to a zero right-hand side, but it appears the so-called residual, indicated with $r(W)$.

A restriction must be placed on $r(W)$. In particular, with the method of weighted residuals, the restriction of $r(W) = 0$ is applied at the points where the solution values are collocated on, which means that the differential equation of interest in now exactly satisfied in those points.

First of all, it is convenient to rewrite (Equation 3.16) with the more convenient Lagrangian polynomials, so that it becomes

$$W_a(x_j) = \sum_{i=1}^{n} W_i l_i(x_j). \tag{3.18}$$

Then, as stated multiple times before, CMSM uses a parametric mapping on the interval $[0, 1]$, which is thought to be more convenient, rather than on the natural interval $[-1, 1]$. Therefore, also the quadrature points have to be mapped on the same unit interval as well.

Two sets of grids are considered:

- Chebyshev-Gauss grid, indicated simply as Gauss grid in the following, whose quadrature points, given as roots of Chebyshev polynomial of degree $n+1$, T_{n+1}, are mapped to $[0, 1]$, and are indicated by

$$X_{j+1/2} = \frac{1}{2}\left[cos\left(\frac{(2j+1)\pi}{2n}\right)\right] \qquad j = 0, ..., n-1. \tag{3.19}$$

  These points are indicated in figure 8 with filled circles;

- Chebyshev-Gauss-Lobatto grid, that will be referred to as Lobatto grid, whose quadrature points are the roots of Chebyshev polynomials of degree n derivative:

$$X_j = \frac{1}{2}\left[cos\left(\frac{j\pi}{n}\right)\right] \qquad j = 0, ..., n. \tag{3.20}$$

  These points are indicated in figure 8 with open circles.

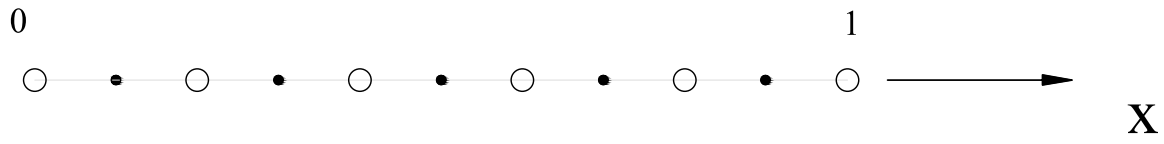


Figure 8: One-dimensional representation of Lobatto and Gauss grids. The filled circles represent Gauss points, while the open circles represent Lobatto points.

In the staggered-grid spectral method, solution values W are collocated on the Gauss grid, while fluxes F are collocated at the Lobatto points. It must be underlined that Gauss points, even if are alternated to Lobatto points, do not lie halfway between two consecutive Lobatto points. In addition, while all Gauss points lie inside the interval considered, i.e. if the interval considered is $[a, b]$, Gauss points are such that $a < x_i < b$, whereas Lobatto points include also the two endpoints, which means Lobatto points obey the condition $a \leq x_i \leq b$.

These considerations lead to a new formulation of the equation for the approximation of W, that now reads

$$W^g(X) = \sum_{i=0}^{n-1} W^g_{i+1/2} h_{i+1/2}(X), \tag{3.21}$$

where superscript 'g' indicates the grid considered is the Gauss grid, and $h_{i+1/2}$ is the Lagrange interpolating polynomial on Gauss grid of degree $n-1$, given by

$$h_{i+1/2}(s) = \prod_{j=0, i\neq j}^{n-1} \frac{s - X_{j+1/2}}{X_{i+1/2} - X_{j+1/2}}. \tag{3.22}$$

A similar formulation, but referred to the fluxes, is applied to the fluxes on the Lobatto grid. F, for example, is approximated as follows:

$$F^l(X) = \sum_{i=0}^{n} F^l_i l_i(X), \tag{3.23}$$

with the Lagrangian interpolating polynomial of degree n, $l_i(X)$, expressed as

$$l_i(s) = \prod_{j=0, i\neq j}^{n} \frac{s - X_j}{X_i - X_j}. \tag{3.24}$$

In this case too the grid we are referring to is indicated by a superscript, which reads 'l' for the Lobatto grid.

The procedure of discretizing an equation through CMSM is described referring to the scalar problem, defined on an interval $[a, b]$

$$\frac{dW}{dt} + \frac{\partial F(W)}{\partial x} = 0. \qquad [a, b] \tag{3.25}$$

First of all, the interval $[a, b]$ is subdivided into n non-overlapping intervals, each one of which is mapped to the interval $[0, 1]$. For each of them, (Equation 3.25) may be rewritten as

$$\frac{dW}{dt} + \frac{1}{\frac{\partial x}{\partial X}} \frac{\partial F(W)}{\partial X} = 0. \qquad [0, 1] \tag{3.26}$$

On each of the subdomains just defined, a Gauss and a Lobatto grid are calculated, by means of (Equation 3.19) and (Equation 3.20). With those grids defined, it is possible to approximate W on the Gauss grid with (Equation 3.21), and F_x on the Lobatto grid with (Equation 3.23). Substituting everything in (Equation 3.26), and imposing a $r(W) = 0$, which means imposing a zero residual, yields the following equation:

$$\frac{dW^g_{i+1/2}}{dt} + \frac{1}{\frac{\partial x}{\partial X}} \frac{\partial F^g(X_{i+1/2})}{\partial X} = 0, \qquad i = 0, ..., n-1. \tag{3.27}$$

To determine the flux, F^l, it is necessary to interpolate W^g on the Lobatto points, through the expression

$$W^l(X_i) = \sum_{k=0}^{n-1} W^g_{k+1/2} h_{k+1/2}(X_i) \tag{3.28}$$

Knowing the solution values on the Lobatto grid, it is then possible to determine the flux $F^l(X_i)$ on the same grid. In order to ensure consistency to the method used, the flux must be a continuous and differentiable function throughout the domain, and especially at the interface between two neighboring domains. Since that property is not automatically ensured, C_0 continuity and a patching have to be applied to the fluxes at such interfaces by means of a Riemann solver, that will be briefly described in section (3.1.3). Once this boundary flux is computed, the derivative of the flux is given by

$$\frac{\partial F^g(X_{j+1/2})}{\partial X} = \sum_{i=0}^{n} F^l(X_i) \frac{\partial l_i(X_{j+1/2})}{\partial X}. \tag{3.29}$$

Storing the Lagrangian polynomial derivatives into a matrix D, the partial differential equation is reduced to an ordinary differential equation, ready to be integrated with a method such as the Runge-Kutta method.

3.1.2.2 Three-Dimensional Case

For a three-dimensional case, the same procedure just described is applied, with the necessary adjustements for the higher number of dimensions involved. The subdomains are now isoparametrically mapped to a unit square, so that [29]

$$D_i \longleftrightarrow [0,1] \times [0,1] \times [0,1],$$

where

$$\Omega = D_1 \cup ... \cup D_i \cup ... \cup D_n.$$

Moving from one to three dimensions, also Gauss and Lobatto grids extend in space. In three dimensions, in fact, the grids are the tensor product of the one-dimensional grids of the case previously presented. Therefore, we may subdivide the grids as follows:

- Gauss grid, that includes points defined through the tensor product of (Equation 3.19). This grid is formed by the Gauss/Gauss/Gauss points (GGG), represented by coordinates $(X_{i+1/2}, Y_{j+1/2}, Z_{k+1/2})$, $i, j, k = 0, ..., n-1$, and it is represented with a filled circle in figure 9. On this grid, solution values are approximated, exactly as it happened for the 1D case described in section (3.1.2.1);
- Lobatto grid, that includes three sets of points, defined by tensor product of (Equation 3.19) and (Equation 3.20)
    1. Lobatto/Gauss/Gauss points (LGG), with coordinates $(X_i, Y_{j+1/2}, Z_{k+1/2})$, with i, j, k such that $i = 0, ..., n$ and $j, k = 0, ..., n-1$, denoted in figure 9 with an open square, where the flux $\tilde{F}$ is evaluated;
    2. Gauss/Lobatto/Gauss points (GLG), with coordinates $(X_i, Y_j, Z_{k+1/2})$, with i, j, k such that $j = 0, ..., n$ and $i, k = 0, ..., n-1$, denoted in figure 9 with an open triangle, where the flux $\tilde{G}$ is evaluated;
    3. Gauss/Gauss/Lobatto points (GGL), with coordinates $(X_i, Y_{j+1/2}, Z_k)$, with i, j, k such that $k = 0, ..., n$ and $i, j = 0, ..., n-1$, denoted in figure 9 with an open circle, where the flux $\tilde{H}$ is evaluated.

It has been assumed that the approximation order, n, is the same in each of the three directions.

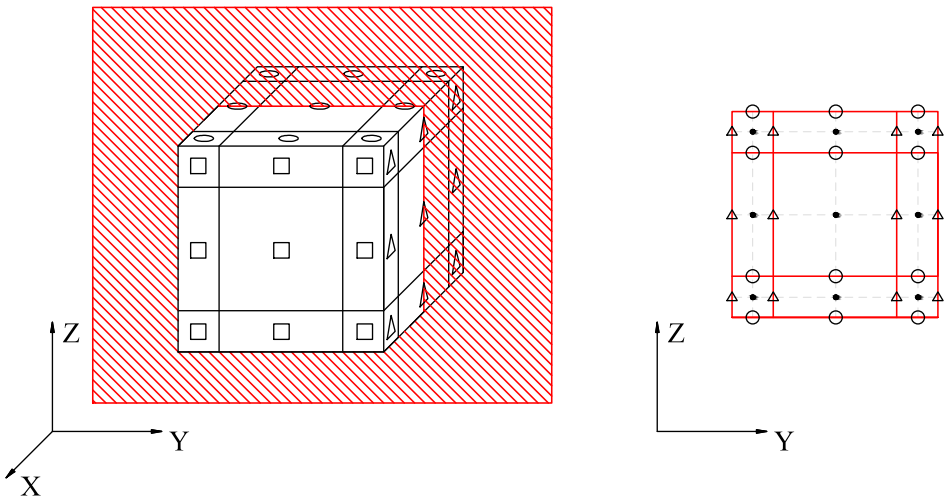


Figure 9: Representation of Lobatto and Gauss grids, three-dimensional representation with sectioning plane on the left, two-dimensional section on the right. The filled circles represent Gauss/Gauss/Gauss points (GGG), the open squares Lobatto/Gauss/Gauss points (LGG), the open triangles Gauss/Lobatto/Gauss points (GLG) and the open circles Gauss/Gauss/Lobatto points (GGL).

Now that all the needed instruments to perform the calculations are defined, a similar procedure to that described for the 1D case is followed. Since Navier-Stokes equations include both non-viscous and viscous contributions, it is convenient to split them in order to approximate the solution. The non-viscous part, that corresponds to Euler equations, is calculated first, through steps similar to those followed in the 1D case.

Solution values W are interpolated to the Lobatto points in all three dimensions (LGG, GLG, GGL grids), but due to Lagrangian polynomials' property, i.e. $l_j(s) = \delta_{ij}$, it is possible to separate these interpolations in every direction independently from the other two. This means

that this three-dimensional interpolation is splitted into three one-dimensional operations, each independent from the other two. Therefore, the expressions used for this operations are

- For the X direction, associated with the LGG grid

$$\mathbf{W}^{lgg}(X_i, Y_{j+1/2}, Z_{k+1/2}) = \sum_{p=0}^{n-1} \mathbf{W}^{ggg}_{p+1/2,q+1/2,r+1/2}\, h_{p+1/2}(X_i); \tag{3.30}$$

- For the y direction, associated with the GLG grid

$$\mathbf{W}^{glg}(X_{i+1/2}, Y_j, Z_{k+1/2}) = \sum_{q=0}^{n-1} \mathbf{W}^{ggg}_{p+1/2,q+1/2,r+1/2}\, h_{q+1/2}(Y_j); \tag{3.31}$$

- For the z direction, associated with the GGL grid

$$\mathbf{W}^{ggl}(X_{i+1/2}, Y_{j+1/2}, Z_k) = \sum_{r=0}^{n-1} \mathbf{W}^{ggg}_{p+1/2,q+1/2,r+1/2}\, h_{r+1/2}(Z_k). \tag{3.32}$$

After the interpolation of the solution values at the Lobatto points, it is then possible to determine $\mathbf{F_f}$, $\mathbf{G_f}$ and $\mathbf{H_f}$ of (Equation 2.30). On the boundary of two different domains, once again, it is necessary to apply a patching in order to ensure continuity and differentiability of the fluxes. Next step is to calculate the mapped fluxes by means of (Equation 3.11) on their respective Lobatto points.

To determine the viscous fluxes, on the contrary, it is first necessary to determine velocity gradients and temperature gradients presented in section (2.3). These gradients' computations require the interpolant of the Lobatto solution values to be continuous at the interface of

neighboring subdomains, in order to ensure the existence of a unique first derivative. This property is obtained through a Dirichlet patching at the interface, that will be briefly presented later on. After the Lobatto interpolants for the solution values are patched, their derivatives are computed at the Gauss points; the gradients obtained are then interpolated back to the Lobatto points.

At this point it is possible to sum viscous and non-viscous fluxes, obtaining therefore the total fluxes at the Lobatto points. After that, the fluxes interpolants are constructed, and they're given by

$$\tilde{\mathbf{F}}(X,Y,Z) = \sum_{p=0}^{n}\sum_{q=0}^{n-1}\sum_{r=0}^{n-1} \tilde{\mathbf{F}}^{lgg}_{p,q+1/2,r+1/2}\, l_p(X)\, h_{q+1/2}(Y)\, h_{r+1/2}(Z);$$
$$\tilde{\mathbf{G}}(X,Y,Z) = \sum_{p=0}^{n-1}\sum_{q=0}^{n}\sum_{r=0}^{n-1} \tilde{\mathbf{G}}^{glg}_{p+1/2,q,r+1/2}\, h_{p+1/2}(X)\, l_q(Y)\, h_{r+1/2}(Z);$$
$$\tilde{\mathbf{H}}(X,Y,Z) = \sum_{p=0}^{n-1}\sum_{q=0}^{n-1}\sum_{r=0}^{n} \tilde{\mathbf{H}}^{ggl}_{p+1/2,q+1/2,r}\, h_{p+1/2}(X)\, h_{q+1/2}(Y)\, l_r(Z).$$

The fluxes just obtained are then differentiated and calculated at the Gauss grid, with

$$\frac{\partial\tilde{\mathbf{F}}(X_{i+1/2},Y_{j+1/2},Z_{k+1/2})}{\partial X} = \sum_{p=0}^{n} \tilde{\mathbf{F}}(X_p,Y_{j+1/2},Z_{k+1/2})\,\frac{\partial l_p(X_{i+1/2})}{\partial X};$$
$$\frac{\partial\tilde{\mathbf{G}}(X_{i+1/2},Y_{j+1/2},Z_{k+1/2})}{\partial Y} = \sum_{q=0}^{n} \tilde{\mathbf{G}}(X_{i+1/2},Y_q,Z_{k+1/2})\,\frac{\partial l_q(Y_{j+1/2})}{\partial Y};$$
$$\frac{\partial\tilde{\mathbf{H}}(X_{i+1/2},Y_{j+1/2},Z_{k+1/2})}{\partial Z} = \sum_{r=0}^{n} \tilde{\mathbf{H}}(X_{i+1/2},Y_{j+1/2},Z_r)\,\frac{\partial l_r(Z_{k+1/2})}{\partial Z}.$$

The final semi-discrete equation at the Gauss grid is given by:

$$\left[\frac{d\tilde{\mathbf{W}}}{dt}\right] + \left[\frac{\partial\tilde{\mathbf{F}}}{\partial X}\right] + \left[\frac{\partial\tilde{\mathbf{G}}}{\partial Y}\right] + \left[\frac{\partial\tilde{\mathbf{H}}}{\partial Z}\right] = 0, \tag{3.33}$$

where each term is calculated on the Gauss grid $(i+1/2, j+1/2, k+1/2)$, and which is then advanced in time with a 4-th order low storage Runge-Kutta scheme.

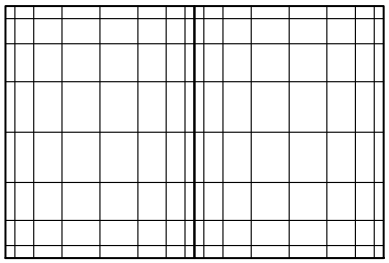
3.1.3 Patching

As previously explained, the Lobatto flux values require C_0 continuity on the whole domain, which implies that this property must be satisfied also on the interface of neighboring domains. In particular, Navier-Stokes equations require two different levels of patching:

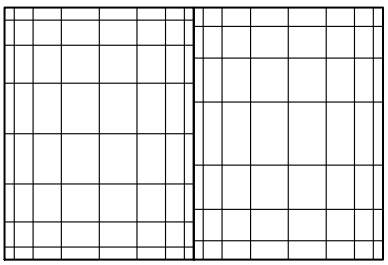
- The first one, a Dirichlet patching, to ensure the uniqueness of the first derivative of $\mathbf{W}$;
- The second one, a Neumann patching, to ensure the uniqueness of the viscous flux derivative.

Both of these patchings are applied at interfaces of two neighboring domains, that are the regions where C_0 continuity is not necessarily satisfied.

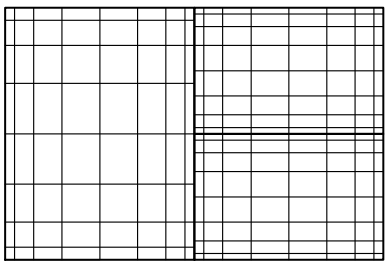
The way Kopriva implemented such patching is through the so-called mortar method. In this method, the Lobatto flux values of two neighboring domains are projected onto a mortar, where the solution values are patched. Doing so, the corner points are excluded in the patching procedure. Even if multiple interface configurations have been programmed, such as conforming, non-conforming (order refinement) or subdivided, all presented in figure 10, for this thesis's purpose only the conforming one will be taken into consideration. As can be seen in figure



(a) Conforming interface.



(b) Non-conforming interface, order refinement.



(c) Subdivision interface.

Figure 10: Types of interface that is possible to encounter.

11, each side of the mortar needs to have projected only one flux component, on which the patching will be later performed. The mortar method, in fact, follows a normal approach, that means the patching is basically a one dimensional operation. An analogue situation would be the shock tube problem, where the gas states on the two sides are known, and the intermediate values are computed through an iterative procedure.

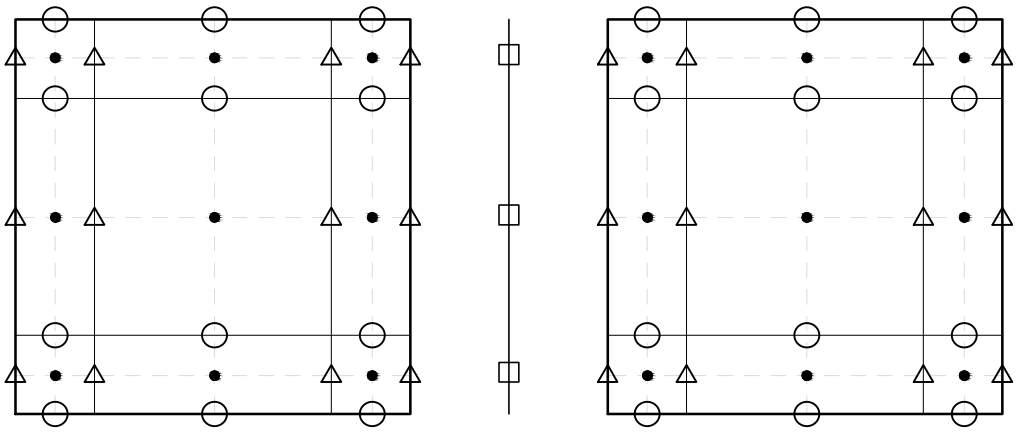


Figure 11: Mortar configuration for a conforming grid.

In the code used to perform the simulations, an approximate Riemann solver introduced by Roe[30] has been programmed, since it is computationally less expensive than the exact solution. Roe's approach is to modify the exact Riemann problem, described by

$$\frac{\partial \mathbf{W}}{\partial t} + \mathbf{J}(\mathbf{W})\left(\frac{\partial \mathbf{W}}{\partial x}\right) = 0, \tag{3.34}$$

where $\mathbf{J}(\mathbf{W})$ is the Jacobian matrix, with an approximated one:

$$\frac{\partial \mathbf{W}}{\partial t} + \mathbf{J}(\mathbf{W^L}, \mathbf{W^R})\left(\frac{\partial \mathbf{W}}{\partial x}\right) = 0, \tag{3.35}$$

where $\mathbf{W^L}$ and $\mathbf{W^R}$ are the solution values on the left domain and on the right domain, respectively. Roe's substitution linearizes (Equation 3.34) into (Equation 3.35), and the approximated Riemann problem obtained is then solved exactly. The most important property of this kind of

Riemann solver is that it ensures conservation across discontinuities, which is very important for shock-capturing schemes (property that anyway will not be analyzed in this work).

Even if Euler equations would not require any further patching, Navier-Stokes equations do, as previously described. This are performed as follows:

1. Dirichlet patching: Lobatto interpolant is patched to ensure a unique first derivative of $\mathbf{W}$ at the Gauss points. A continuous piecewise polynomial $\bar{\mathbf{W}}_{\mathbf{e}}(X, Y, Z)$, where the subscript '$e$' indicates the domain number, that is closer to $\mathbf{W}_{\mathbf{e}}(X, Y, Z)$. This simple approach is convenient because it involves just the domain face values, and its being computationally inexpensive has made it suitable for parallel implementation;

2. Neumann patching: an approach similar to that followed for the Dirichlet patching is adopted, since only viscous stresses and temperature gradients normal to the face have to be patched. Again, an averaging procedure to correct the Lobatto interpolant at the interfaces is followed.

3.1.4 Spatial Integration

Instead of using Chebyshev weights, which have a singularity at the Lobatto endpoints, Kopriva[21] [22] [23] preferred to implement Legendre weights on the Legendre quadrature points, that are interpolated then back to the Gauss points. Such weights are indicated with $w_{i+1/2}$. The integral of a generic variable $\mathbf{U}$ can be determined, for a two-dimensional case as

$$\int_0^1 \int_0^1 \mathbf{U}(x, y)\, dx\, dy \cong \sum_{p=0}^{n-1} w_{p+1/2} \left[\sum_{q=0}^{n-1} w_{q+1/2}\, \mathbf{J}(X_{p+1/2}, Y_{q+1/2}) \mathbf{W}^{gg}(X_{p+1/2}, Y_{q+1/2}) \right], \quad (3.36)$$

and, for a three-dimensional case, as:

$$\int_0^1 \int_0^1 \int_0^1 \mathbf{U}(x,y,z)\,dx\,dy\,dz \cong \sum_{p=0}^{n-1} w_{p+1/2} \left[\sum_{q=0}^{n-1} w_{q+1/2} \times \right.$$
$$\left. \times \left[\sum_{r=0}^{n-1} w_{r+1/2} \mathbf{J}(X_{p+1/2}, Y_{q+1/2}, Z_{r+1/2}) \mathbf{W}^{gg}(X_{p+1/2}, Y_{q+1/2}, Z_{r+1/2}) \right] \right]. \tag{3.37}$$

In both cases **J** is the Jacobian matrix, obtained following what described in section (3.1.1.3).

3.2 Time Integration

The differential equation obtained through spatial discretization is discretized in time with a 4th order Runge-Kutta method. The code had implemented both RK3 and RK4 explicit schemes, presented respectively by Williamson [31] and Carpenter and Kennedy [32]. These methods have been chosen because they're explicit, so that no information from previous timesteps is needed, they're flexible and easily programmable, and the solutions presented by the authors referenced are both low-storage schemes.

The system of equations obtained after space discretization is of the following form:

$$\frac{\partial y}{\partial t} = f(t, y(t)), \tag{3.38}$$

$$y(t_0) = y_0. \tag{3.39}$$

Commonly used Runge-Kutta methods require the definition of a step size $h > 0$, and the solution at a given step is, for a Runge-Kutta method of 4th order:

$$y_{n+1} = y_n + \frac{1}{6}\,h(k_1 + 2k_2 + 2k_3 + k_4), \tag{3.40}$$

$$t_{n+1} = t_n + h, \tag{3.41}$$

for $n = 0, 1, 2, 3$. Coefficients indicated with k_i in (Equation 3.40) are calculated as

$$\begin{aligned} k_1 &= f(t_n, y_n), \\ k_2 &= f\left(t_n + \frac{1}{2}\,h, y_n + \frac{1}{2}\,hk_1\right), \\ k_3 &= f\left(t_n + \frac{1}{2}\,h, y_n + \frac{1}{2}\,hk_2\right), \\ k_4 &= f(t_n + h, y_n + hk_3). \end{aligned}$$

In previous equations, y_{n+1} represents the RK approximation of $y(t_{n+1})$, and is determined by the sum of the present value, y_n, and the weighted average of four increments, each of which involves the size of the interval considered, h, and an estimated slope specified by function f on the right-hand side of the differential equation. A RK4 can be subdivided into four steps, that respond to different operations[33]

1. Euler predictor;
2. Euler corrector;
3. Leapfrog predictor;

4. Milne corrector.

The sequence of these operations constitutes the classical Runge-Kutta method of order 4.

The differential equation resulted from spatial discretization is analogue to (Equation 3.38), where $y = \mathbf{W}$ and $f = \mathbf{F}$. However, the formulation of a Runge-Kutta method of the 4th order has been slightly modified, since it has been adopted a low-storage formulation that does not require the direct specification of the time interval, but that of the total number of steps. The equations governing this method are

$$q_{i+1} = \alpha_{i+1} q_i + \Delta t \mathbf{F}(t_i, w_i),$$

$$w_{i+1} = w_i + \beta q_i,$$

for $i = 1, 2, ..., s$, where s is the number of stages, $w_0 = \mathbf{W}_{n-1}$, $w_n = \mathbf{W}_n$ and $t_i = t_{n-1} + c_i \Delta t$.

3.2.1 Stability and Accuracy

The stability of the scheme is described using the so-called Courant-Friedrich-Levy (CFL) condition, an important stability criterion for hyperbolic equations, whose requirement for stability is [34]

$$CFL = c\frac{\Delta t}{\Delta x} \leq 1. \tag{3.42}$$

In (Equation 3.42), CFL is called the CFL number, Δx is the grid spacing, Δt is the time interval and c is the speed of sound. To examine this criterion's physical significance, we refer to the following second order wave equation:

$$\frac{\partial^2 u}{\partial t^2} = c\frac{\partial^2 u}{\partial x^2}. \tag{3.43}$$

For this equation, the characteristic lines are

- A right running wave, with equation $x = ct$;
- A left running wave, with equation $x = -ct$.

Considering a right running wave through point A (grid point $i-1$) and a left running wave through point C (grid point $i+1$), their intersection point is indicated with P in figure 12. The time interval for which the CFL number is equal to 1 is indicated with Δt_1, and the following condition must be satisfied:

$$CFL = 1 \iff \Delta t = \frac{\Delta x}{c} = \Delta t_1. \tag{3.44}$$

This implies that point P is at a distance equal to Δt_1 from the x axis.

Let's consider a case where $CFL < 1$, so that the stability CFL condition is satisfied. Being $CFL < 1$, it implies $\Delta t_2 < \Delta t_1$, that means the intersection point of the two waves, called Q, is closer to the x axis. Two kinds of domains are now to be distinguished:

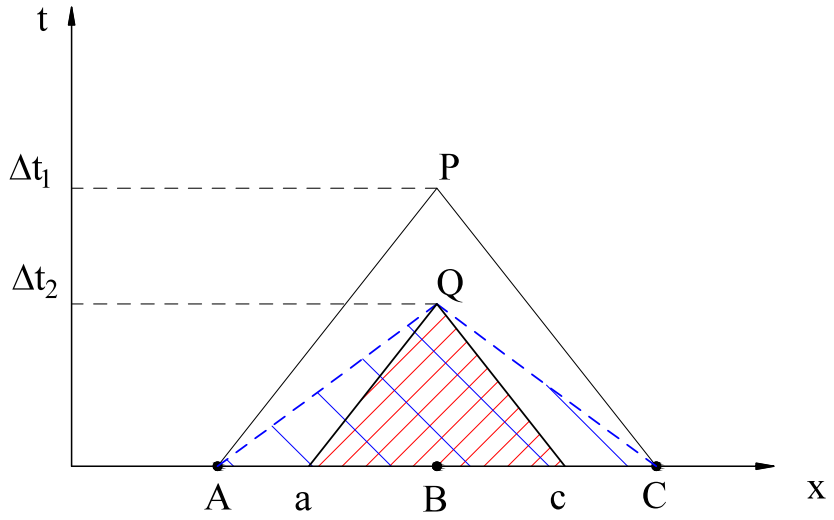


Figure 12: CFL condition representation for a stable case, where the computational domain includes all the analytical domain.

1. The numerical (computational) domain, that involves grid points used to calculate properties in the given point, and that therefore corresponds to the triangle ACQ, whose area has been shaded in blue in figure 12;

2. The analytical domain, that is defined by characteristic lines, and therefore corresponds to the triangle acQ, whose area has been shaded in red in figure 12.

For stability, the computational domain must include all of the analytical domain, such as in figure 12.

If it is chosen a CFL number greater than 1, as shown in figure 13, the waves' intersection point would be farther from the x axis then before, that corresponds to a time interval $\Delta t_3 > \Delta t_1$. As a consequence, the analytical domain would be defined by the triangle efR. In

this case, the analytical domain is not completely included in the computational domain, and therefore the CFL condition is not satisfied and the behaviour is not stable.

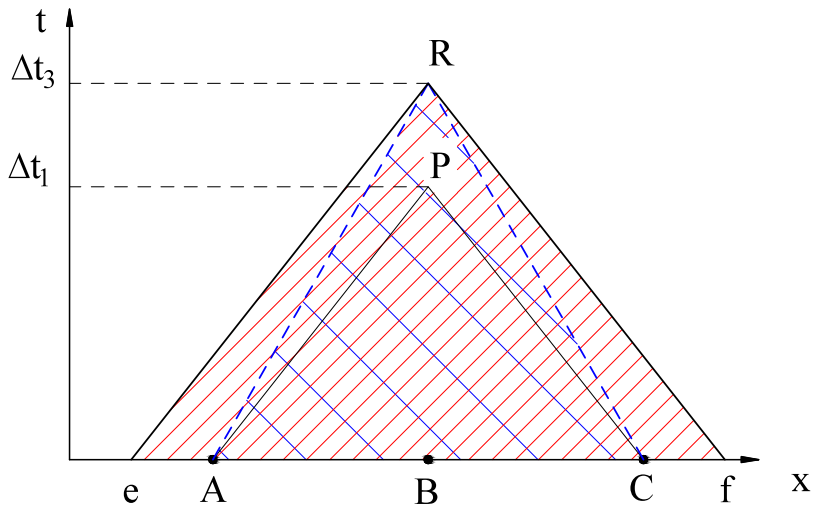


Figure 13: CFL condition representation for an unstable case, where the computational domain does not include all the analytical domain.

The question of accuracy can also be examined through figure 12. Considering this stable case, properties at point Q depend theoretically just on the points in the red-shaded area, that correspond to the analytical domain. However, even if points A and C are outside the analytical domain, and theoretically should not matter, the numerical calculation of properties at point Q actually relies on informations taken from A and C. This situation becomes even worse when $CFL \ll 1$ ($\Delta t_2 \ll \Delta t_1$). In this case, in fact, even if the calculations are stable, the

large mismatch between the domain dependence of Q and the location of the numerical data to calculate properties at Q can be responsible of quite inaccurate results.

Therefore, the CFL number must be less or equal than 1 for stabilty, but its accuracy is as greater as closer to 1 it is chosen.

Clarified what is the CFL condition and its physical and numerical meaning, it is now possible to consider its influence on the three-dimensional problem this thesis is about. In this case, there are three velocity components to take into account, and also the grid is not necessarily equally-spaced. For this reasons, the CFL relation is modified as

$$\Delta t = \frac{CFL\Delta x_{min}}{a}, \tag{3.45}$$

where Δx_{min} is a minimum grid spacing, corresponding to a minimum value of the Lobatto endpoints over the whole domain, and a is a characteristic maximum wave speed, that can be expressed as $a = \sqrt{u^2 + v^2 + w^2} + c$ and has to be calculated on all the points. On mapped coordinates, (Equation 3.45) becomes

$$\Delta t = \frac{CFL}{max\left(\frac{U}{\Delta X} + \frac{V}{\Delta y} + \frac{W}{\Delta Z}\right)}, \tag{3.46}$$

where U, V and W are the wave velocities along X, Y and Z respectively, expressed as:

$$U = (u+a)\frac{\partial X}{\partial x} + (v+a)\frac{\partial X}{\partial y} + (w+a)\frac{\partial X}{\partial z}, \tag{3.47}$$

$$V = (u+a)\frac{\partial Y}{\partial x} + (v+a)\frac{\partial Y}{\partial y} + (w+a)\frac{\partial Y}{\partial z}, \tag{3.48}$$

$$W = (u+a)\frac{\partial Z}{\partial x} + (v+a)\frac{\partial Z}{\partial y} + (w+a)\frac{\partial Z}{\partial z}. \tag{3.49}$$

For these velocities, the maximum has to be determined over the whole domain, and is then applied in (Equation 3.46).

In the following chapter, the mesh generator is described.

CHAPTER 4

MESH GENERATOR

A primary objective of this thesis, as previously stated, is the implementation of a mesh generator, in order to model three-dimensional geometries in a format suitable to the numerical code used. As of now, the code is able to deal with elements that are rectangles or skewed convex quadrilaterals for a two-dimensional case, and parallelepipeds or hexahedrons in three dimensions.

The mesh generator, based on the code developed by J. Komperda at CMTL[25], is written in Python, a relatively new programming language that has many advantages concerning productivity, usability and readability, when compared to classical and older ones such as FORTRAN and C. In fact, its syntax allows programmers to express concepts in a lower number of code lines than it would be possible in other languages.

The format to adopt for the '$.mesh'$ file generated is the simplemesh format, that presents nodes, elements and boundary conditions for the shape in this order. The mesh generator has been divided into two files, one responsible to generate the two-dimensional geometry on the $x - y$ plane and assign it boundary conditions, and the other responsible to optionally extrude this two-dimensional shape in the z direction, and write the '$.mesh'$ file. In the following, both files are presented to give an exhaustive description of the mesh generator.

4.1 Two-Dimensional Shape

The file responsible for the two-dimensional shape generation will be referred to as $2D.py$. Being the domain still limited to two dimensions, the elements built will be rectangles or skewed quadrilaterals. For the current version of the code employed, in fact, no triangles were allowed, but just polygons with four sides.

The mesh generation moves from elemental blocks, i.e. the basic parts a more complex shape can be subdivided into. Considering the example of figure 14, it is possible to notice how an example shape is subdivided into its elementary blocks, each of which is directly described in the mesh generator.

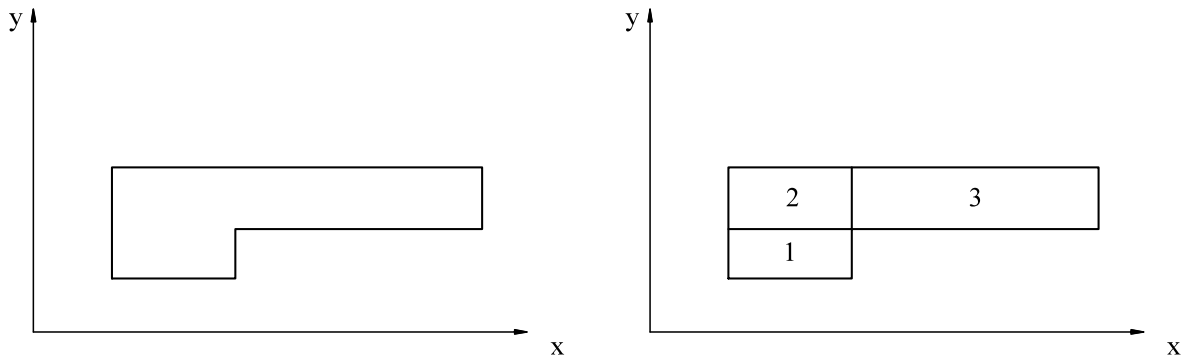


Figure 14: Example shape representing a forward-facing step (left), and subdivision into its elementary blocks (right).

File $2D.py$ is divided into different classes, each corresponding to a particular shape. Those can be grouped into three main categories:

1. Category 'rectangle', that includes all the quadrilaterals with four 90 angles, i.e. rectangles and squares;
2. Category 'genshape', that includes all the other convex quadrilaterals, i.e. those excluded from the first category. Even if rectangles and square are just a particular case of the shape described in this group, it has been considered convenient to separate them in a stand-alone group;
3. Category 'unstructshape', that is responsible of the combination of shapes declared in the previous two groups.

The peculiarity of Python is that it involves a syntax that is easy to read and to understand, and therefore the mesh generator has been implemented to make the user comfortable from its very first usage.

Categories 1 and 2 represent the real core of the mesh generator. The informations that should be specified in order to create a two-dimensional mesh are

- The contour of the shape. This can be described by specifying bottom left corner coordinates and sides' lengths in case of a rectangle, or the four corner points coordinates in case of a generic shape. An example of generic quadrilateral created by the mesh generator is shown in figure 15;

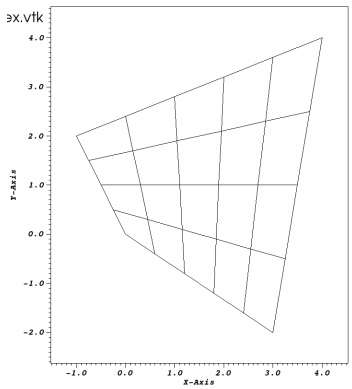
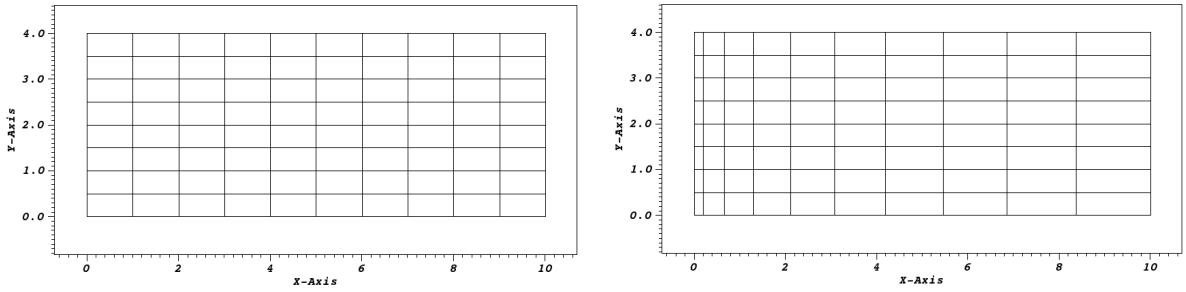


Figure 15: Example quadrilateral defined specifying its corner points coordinates.

- The number of points the mesh should have along the x-direction (npx) and along the y-direction (npy). Considering that also the corner points are included as points of the mesh, the interior points to generate for each side of the shape are $(npx-2)$ and $(npy-2)$, for sides oriented along x-axis and y-axis, respectively;
- The nodes' distribution, i.e. the way the grid is packed in the shape. Due to the different accuracy requirements of different points, in fact, the mesh may be built more refined near one or two sides, near one, two or four corners. For that reason, it has been made possible to specify some parameters that allow the grid to be differently refined, accordingly to user's necessities. An example is shown in figure 16(b). If none of these parameters is specified, the grid is assumed to be equally spaced, as shown in figure 16(a). Knowing how the grid is spaced along the four sides, through linear interpolation is possible to calculate all interior points' coordinates;



(a) Rectangle with uniform grid distribution.

(b) Rectangle with non-uniform grid distribution.

Figure 16: Types of grid distributions implemented in the mesh generator.

- The elements in which the elementary shape is subdivided, i.e. the connections between the nodes already calculated. Dealing with quadrilaterals, it is well known how the interior points have to be connected, since every row has npx points, and there is a total of npy rows. The algorithm implemented for creating connections is responsible to number all nodes of the shape, count the elements and assign to each element the four nodes that delimitate it;

- The boundary conditions on the shapes' sides, i.e. what each side of the shape will be faced to in the physical problem. Such boundary conditions are specified with a number, each corresponding to a different type of boundary (e.g. inflow boundary condition, outflow boundary condition, wall boundary condition, etc.). Since it is necessary to impose such conditions just on the sides of the shape, the elements whose one side lies on the border of the shape are numbered accordingly to the boundary they will be assigned to, and accordingly to the face they face to, accordingly to what presented in sections (3.1.1.1)

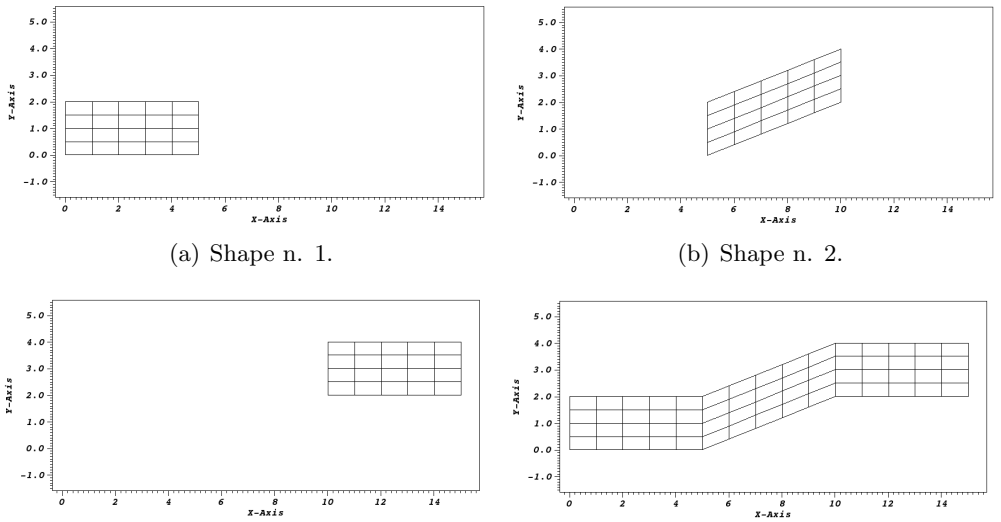
and (3.1.1.2). Since just an elementary shape is being defined, it may happen that one of its sides will result interior to the final mesh. In this case, the number 0 is imposed, and the same will happen to the side of the other shape that will combine with the one described in the example.

Once all the elementary shapes have been defined, they are summed together by means of category 'unstructshape'. This class is responsible to manage all the features previously defined, and combine them with as less computational effort as possible. In particular, the addition of two blocks can be summarized in the following steps:

1. The nodes of the two shapes are combined, and roundoff errors due to limited machine precision are corrected, allowing almost coincident nodes to become one out of two;
2. The old connections' lists are updated, without recalculating them for each element. The elements' numbers are also updated. This algorithm saves a lot of computational time with respect to the one previously used to perform a similar operation;
3. The old boundary conditions' lists are updated as well, accordingly to the new element numbering.

Updating nodes, elements, and boundary conditions, the original blocks are now melted into a unique shape. In case of multiple shape additions, they are performed in serial, where the result of the addition of the first two shapes is then combined with the third one, and so on and so forth, as shown in figure 17. After all the two-dimensional quadrilaterals have been

combined together, the resulting shape is ready to enter file $3D.py$, in order to be formatted for a two-dimensional case, or to be converted into a three-dimensional mesh.



(a) Shape n. 1.

(b) Shape n. 2.

(c) Shape n. 3.

(d) Shape n. 4, result of the serial addition of the three shapes previously defined.

Figure 17: Example of the routine implemented in class 'unstructshape'.

4.2 Three-Dimensional Shape

File $3D.py$ is divided into two parts, that perform two separate operations.

The first part, optional, is responsible to convert the two-dimensional shape received in input to a three dimensional one, extruding it into the z-direction. In order to perform this operation, it is necessary to specify the desired depth, the number of points along the z-axis

(npz), their distribution (if uniform mesh is not desired) and boundary conditions for front and rear faces.

The mesh generator, basically, repeats the shape received in input on the (x, y) plane at different z coordinates, generating then a three-dimensional distribution of nodes. Such nodes are then grouped into three dimensional elements (hexahedrons), that are the elementary blocks constituting the shape. Even boundary conditions on the sides are repeated along the z axis, while new boundary conditions previously specified are assigned to front and rear face. The output result of such operations is shown in figure 18, where the two-dimensional shape given in input is the same represented in figure 17.

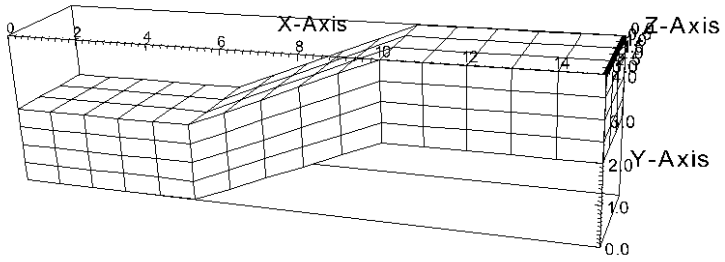


Figure 18: Example of extrusion of a two-dimensional shape in the third dimension, to generate a three-dimensional mesh.

Once the first part has terminated its work (whether it has been called), the mesh generator deals with a two- or three-dimensional mesh. The second part of file $3D.py$ formats the mesh constructed into a .mesh file that results appropriate for the code's purpose. Once also the second part is done, the .mesh file is ready to be employed for the physical simulation, and therefore the mesh generator has terminated its work.

CHAPTER 5

NUMERICAL SIMULATIONS

The logic the code follows to perform numerical analysis of physical problems has been explained in the previous chapter, so that this one deals directly with the numerical simulations performed. The first section of this chapter is dedicated to the introduction and description of boundary conditions used to describe the cases that will be analyzed. The second part of the chapter focuses on the flow over a backward-facing step, where the numerical results obtained are compared to those experimental found in literature. Finally, the last part of the chapter examines the flow over a backward-facing step with a ramp added after the step, so that the inlet area equals the outlet area. This particular kind of shape is chosen because of the interest in modeling the flow in a dump combustor, where the backward-facing step is adopted to create a recirculating region that increases the residence time of the fuel.

5.1 Boundary Conditions

This section describes the boundary conditions that are imposed at test sections' boundaries to perform the simulations. Their implementation in the code works exactly as the mortar patching presented in section (3.1.3), with the difference that only one side of the mortar is known (the side inside the computational domain), while the other is unknown and needs to be specified (the side outside the computational domain, i.e. there's the need to specify an appropriate boundary condition).

Essentially, this work deals with four different kinds of boundaries:

- Inflow boundary;
- Outflow boundary;
- Wall boundary;
- Periodic boundary.

For each one of the types elencated, appropriate boundary conditions are introduced in the following.

5.1.1 Inflow Boundary Conditions

As Jacobs stated in his work[24], 'It has been shown that for a well-posed solution in three-space dimensions, five flow variables need to be specified at an inflow boundary, and four at an outflow boundary'. The inflow boundary conditions employed in this work are constituted of the specification of

- u, velocity profile along the inlet section;
- ρ, density value, equal to free stream density ρ_∞;
- p, pressure value, equal to free stream pressure p_∞;
- v, velocity along y axis, equal to 0 (fully developed flow);
- w, velocity along z axis, equal to 0 (fully developed flow).

These five variables are specified at the inlet section, so that the problem is well posed.

The values specified are imposed as boundary values, so that the Riemann problem is solved on the mortars at the inlet section as specified in section (3.1.3), by means of the Roe [30] solver and (Equation 3.35).

The u velocity profile chosen is representative of the fully developed flow condition, so that the ingoing flow is representative of channel flow at steady state condition from the very beginning of the simulation. This profile is not specified at all nodes, but its values are described at different y coordinates. To assign u to a node, a linear interpolation is made between the two points of the profile that have the closest coordinate to the node computed, for every node in the inflow section.

5.1.2 Outflow Boundary Conditions

A possibility for specification of outflow boundary conditions is to refer to the same type of conditions described in section (5.1.1). If the outlet channel is sufficiently long, in fact, it can be assumed that the flow has enough space to fully develop again. Therefore, in this case too, the boundary condition is based on a u velocity profile specified at the exit section, that is once more reflective of a fully developed flow condition. The variables specified at the exit, then, are

- u, velocity profile, fully developed;
- ρ, density;
- p, pressure;
- v, velocity along y axis, set to 0;
- w, velocity along z axis, set to 0.

Again these variables make the problem well posed at the outlet section. Once more, the outflow boundary condition is imposed with the same logic of the inflow boundary condition described in section (5.1.1), so that the problem is solved on the mortars with (Equation 3.35). Density and pressure at the outlet are considered equal to their respective free stream values, assuming that after a sufficient test section length the flow returns to its free stream conditions. To make sure this condition is satisfied, an outgoing channel long enough need to be specified for the computational domains analyzed. As for the inflow boundary conditions of section (5.1.1), the values of u are interpolated from the profile to the nodes of the computational grid.

There's also another type of outflow boundary conditions used in this work, known as transmissive (or zero-gradient) boundary conditions. Their logic is to specify some variables' first derivatives at the outlet section, so that no exact values need to be imposed. Considering an outlet channel long enough, in fact, it is possible to assume that flow exiting the computational domain is in a regime of fully developed flow. Therefore, instead of specifying variables' exact values at the exit, the outflow condition can be summarized by the following equations:

$$\frac{\partial u}{\partial x} = 0;$$
$$\frac{\partial \rho}{\partial x} = 0;$$
$$\frac{\partial p}{\partial x} = 0.$$

Since the flow at the outlet section is fully developed, u, ρ and p derivatives with respect to x are set equal to 0, that means this variables don't change anymore in the x direction. In addition, also v and w are imposed equal to 0, because of the regime of fully developed flow achieved. Basically, for each time step the values on mortar side outside the computational domain are set equal to those on the other side of the mortar, inside the test section. This boundary condition is responsible to 'transmit' informations from the domain to the boundary, where these information are imposed as boundary values.

5.1.3 Wall Boundary Conditions

The third type of boundary conditions is represented by a no-slip isothermal wall boundary condition. Temperature T_w is imposed, and the walls operate as thermostats, so that $T_w = const$ during the whole simulation. This condition could be employed to study the effects of heating/cooling applied to the flow, but the main purpose of this work is different from that. Therefore, wall temperature is chosen accordingly to flow conditions, so that $T_w = T_\infty$, i.e. the temperature of the wall is equal to the free stream temperature of the flow. As a consequence, the wall is responsible nor to heat neither to cool the flow, but is just used as a physical interface where no heat exchange is performed.

Another feature of the wall boundary condition is its no-slip condition, that is achieved imposing a flow on the opposite side of the wall with a velocity equal in magnitude but opposed in sign. As a results, at steady state, the flow closer to the wall is expected to decrease its u velocity down to 0 and to reach a temperature $T = T_w$.

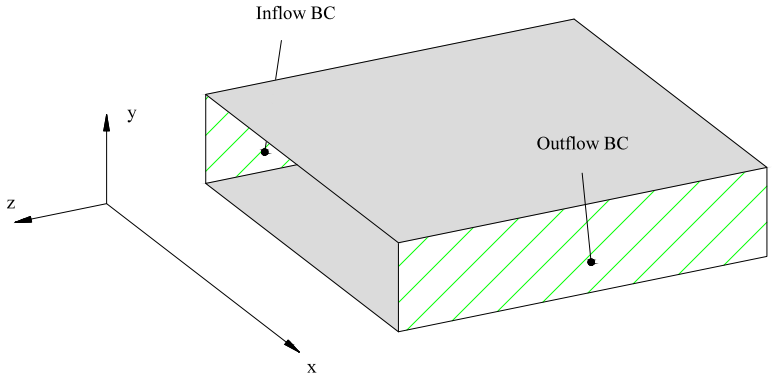
5.1.4 Periodic Boundary Conditions

The last type of boundary conditions to describe is constituted by the so called periodic boundary conditions. They are used to simulate a large system by modelling just a small part of it, but far from its edge.

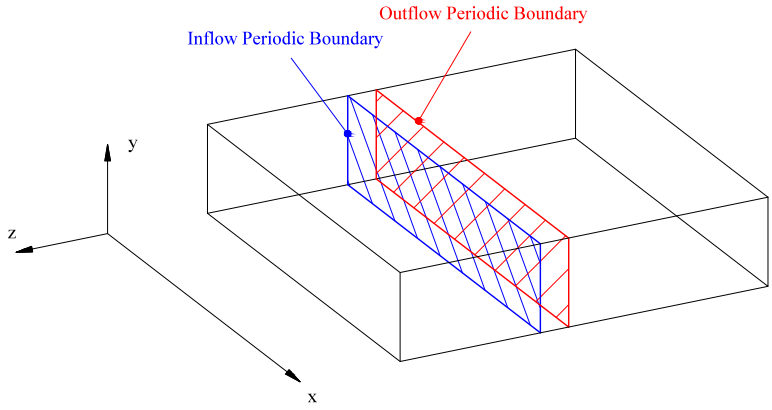
In the cases considered in the following, a two-dimensional geometry is repeated along the z direction, to generate a three-dimensional mesh. It is exactly along this direction that periodic boundary conditions are perfectly applied. They're based on the fact that the flux of all flow variables leaving the outlet periodic (or cyclic) boundary is set equal to the flux entering the inlet cyclic boundary. In other terms, the values of each variable just upstream and downstream of the periodic inlet plane are set equal to the values of each variable just upstream and downstream of the periodic outlet plane.

As shown in figure 19, a wide geometry 19(a) can be modeled isolating just a small section of it 19(b), and analyzing it as if it was the whole model 19(c). Values obtained for such a section are then repeated in space along the whole extension of the original model.

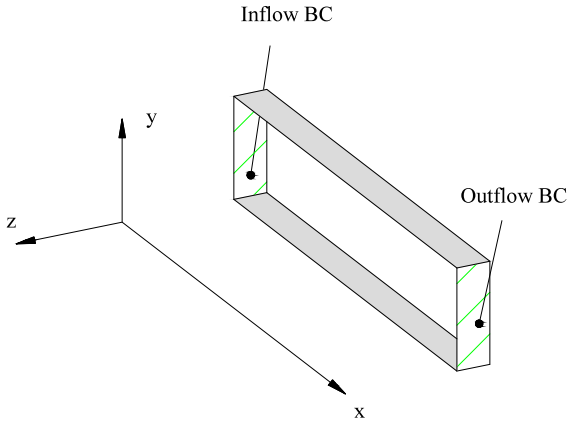
Such a boundary condition is particularly suitable for the z direction, since the flow analyzed are two-dimensional, and therefore each cross-section on plane $x - y$ is repeated equal to itself along z axis.



(a) Geometry of the problem.



(b) Isolation of a section of the original geometry, far from its edges.



(c) Initial problem is simplified, since only the represented geometry has to be computed.

Figure 19: Graphical representation of periodic boundary conditions.

5.2 Flow over a Backward-Facing Step

Described all the boundary conditions that have been employed, this section will focus on the analysis of the flow over a backward-facing step (BFS), that constitutes a simplified model of the dump combustor that is going to be analyzed later.

The general features for a backward-facing step are shown in figure 20. They may be summarized as

- Boundary layer development: it occurs along the inlet channel;
- Boundary layer separation: because of a discontinuity in the geometry of the problem, the boundary layer separates from the wall and turns into a curved shear layer, that will reattach to the bottom wall downstream of the step;
- Reattachment length: it's the distance between the step and the reattachment point on the bottom wall of the separated boundary layer. It has been proved to be dependent on the Reynolds number, especially for the range $400 < Re < 3000$;
- Recirculation zone: the main recirculation zone is enclosed between the step, the bottom wall and the reattaching boundary layer, and this zone is present for every Re. Above $Re \cong 400$ the flow is proven to become three dimensional, and also other recirculation zones develop in the channel, and on both sides of it;
- Recovery to fully developed wall flow: for the cases analyzed in the following, the flow starts to fully develop again after the primary recirculation zone, since the Reynolds numbers adopted guarantee no other recirculation zone is formed.

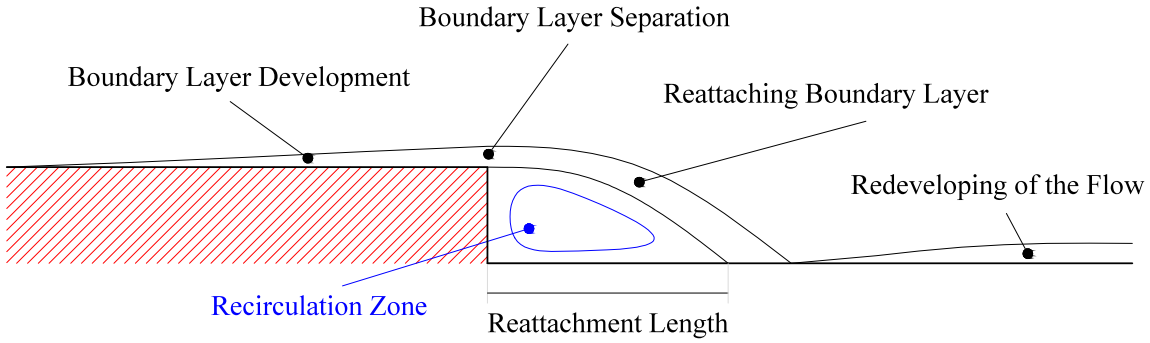


Figure 20: Principal features of the backward-facing step flow, for relatively low Re and for a channel with open top.

The reference experiments for the simulations performed are those of Armaly *et al.*[3]. Although Ref.[3] presented results for a wide Reynolds number range, i.e. $70 < Re < 8000$, this work is going to concentrate only on the region where the flow is proved to be two-dimensional, that means for Reynolds numbers lower than approximately 400. The analysis in Ref.[3] shows that different flow regimes are characterized by typical reattachment length variations with Reynolds number, and has also been the first to report other regions of separated flows downstream of the step and of both sides of the test section.

5.2.1 Computational Model

The geometry of the backward-facing step is shown in the schematic of figure 21, and dimensions adopted by Armaly *et al.*[3] have been reported in table II.

For computational purposes, inlet and outlet lengths specified by Armaly *et al.* have been shortened, because the inlet channel has been proven to be sufficiently long to allow the flow to be fully developed at the step, while the outlet channel length ensures that outlet conditions do

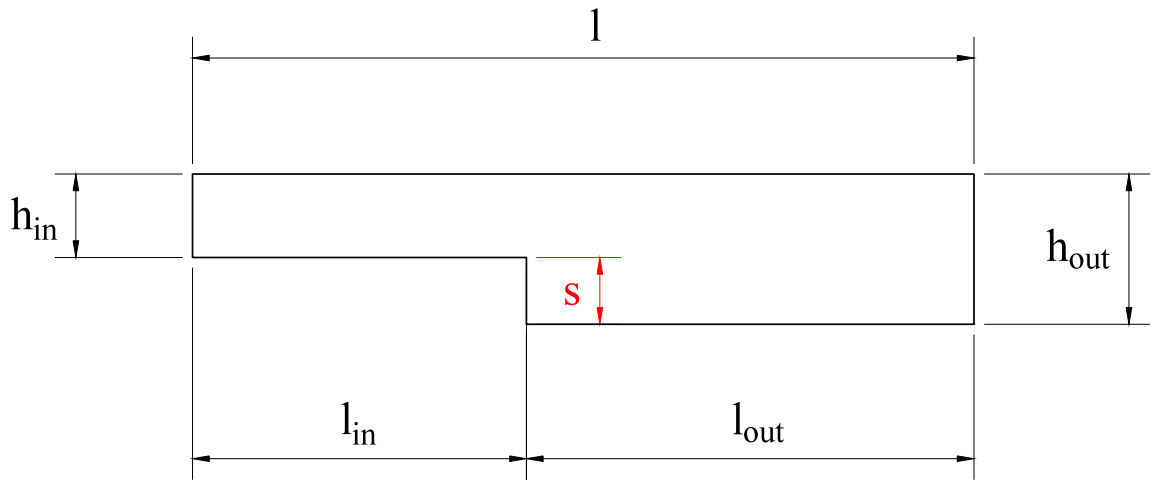


Figure 21: Geometrical dimensions in a backward-facing step case.

TABLE II: DIMENSIONS OF THE BACKWARD-FACING STEP IN ARMALY'S SETUP.

Element	**Symbol**	**Value [mm]**
Inlet channel length	l_{in}	200
Outlet channel length	l_{out}	500
Total length	l	700
Inlet channel height	h_{in}	5.2
Outlet channel height	h_{out}	10.1
Step height	s	4.9

not affect the flow in the vicinity of the step. In addition, all non-dimensional lengths specified have been normalized with the step height.

The step, whose height is 4.9 mm in Armaly's setup, is then assumed to be 1.0. This leads to an inlet channel height of 1.0612, and an outlet channel height of 2.0612. For the inlet channel, furthermore, a length of 5.0 unit lengths is chosen, while 20.0 have been assigned to the outlet channel, leading to a total length of 25.0 for the test section. Since the code used must deal with a three-dimensional geometry, a small depth of 0.05 is specified. However, for the range of Reynolds numbers chosen, the flow will be two-dimensional, and the test section where results will be analyzed is the middle one.

Geometrical dimensions defined so far are summarized in table III, where dimensional and non-dimensional lengths for the computational model are exposed.

TABLE III: DIMENSIONS OF THE BACKWARD-FACING STEP FOR NUMERICAL SIMULATIONS' SETUP.

Element	**Symbol**	**Value [mm]**	**Non-dimensional Value**
Inlet channel length	l_{in}	24.5	5.0
Outlet channel length	l_{out}	98	20.0
Total length	l	122.5	25.0
Inlet channel height	h_{in}	5.2	1.0612
Outlet channel height	h_{out}	10.1	2.0612
Step height	s	4.9	1.0

For its comparison between experimental and computational results, Armaly *et al.* used a grid of $N_x \times N_y = 45 \times 45$. In this work, sixth order polynomials have been used with 40, 16 and 1 domains in the x, y and z directions, respectively. For such a polynomial order, Lobatto points are respectively 320, 128 and 8 in each direction.

The grid is not uniform, but packed where more accuracy is needed. In particular, a power law is chosen in the x direction for the inlet channel, approaching the step, with a last length of 0.1, as well as for the outlet channel, moving away from the step, with a first length of 0.1. In the y direction, instead, something different is imposed, since a finer grid is required near the walls and the step corner, where higher gradients are expected. Therefore, a double power distribution is generated for the mesh, with initial and last length of 0.06, and with a larger mesh going towards the middle of the channel. An example of the $x - y$ plane section view of the mesh used is shown in figure 22. Since just one element is created in depth along the z direction, no particular distribution is needed along the z axis.

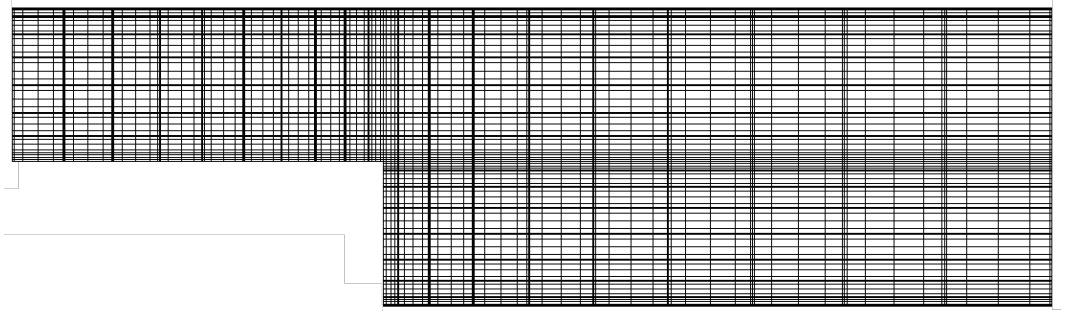
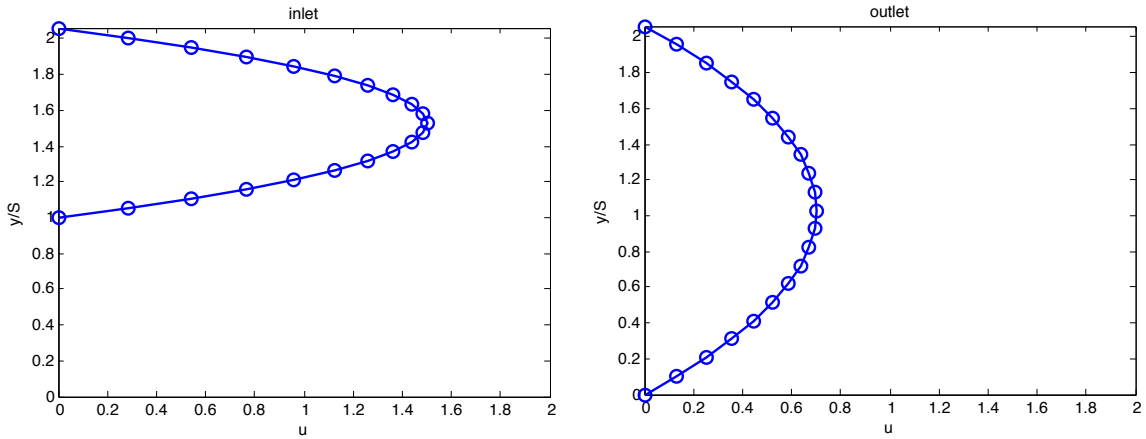


Figure 22: Example of mesh refinement for the backward-facing step, $x - y$ plane section.

The boundary condition at $x = -5.0$, i.e. at the inlet section, is an inflow boundary condition with a fully developed u velocity profile specified, as shown in figure 23(a). This profile is kept for 5 non-dimensional length units, and it corresponds also the profile at $x = 0$, i.e. at the step. For each of the different Reynolds numbers chosen, the inlet profile is calculated assuming a maximum velocity u_{max} derived from Armaly *et al.*[3], and considering then a mean value $\bar{u}_\infty$ such that

$$\bar{u}_\infty = \frac{2}{3} u_{max}. \tag{5.1}$$

Assuming a fully developed u velocity profile for the inlet, both v_∞ and w_∞ are equal to 0. The density and the pressure at the inlet section are $\rho_\infty = 1.0$ and $p_\infty = 4.464$, respectively.



(a) Inlet velocity profile, at $x = -5.0$.

(b) Outlet velocity profile, at $x = 20.0$.

Figure 23: Examples of the u velocity profiles imposed as inflow and outflow boundary conditions for a backward-facing step problem.

At $x = 20.0$ two different types of outflow boundary condition are specified, as described in section (5.1.2).

The first kind of outflow boundary condition is an outflow boundary condition, based on the specification of an outflow u velocity profile. Considering that the outlet section is at a sufficient distance from the step, for the range of Reynolds numbers considered, a u velocity profile for a fully developed flow is imposed (shown in figure 23(b)), where the maximum value is once again derived from Armaly *et al.*[3], and the mean value is calculated with (Equation 5.1). This time too, v and w are set equal to 0, while density and pressure are imposed as $\rho = 1.0$, to satisfy mass conservation, and $p = p_\infty = 4.464$, considering that the flow has returned to its free stream state. All these values are imposed as boundary values and are used in the solution of the Riemann problem on the mortar.

The second kind of outflow boundary condition used is represented by the so-called transmissive (or zero-gradient boundary condition). In particular, since the outlet section is assumed to be far enough from the step to allow the flow to reach a fully developed condition again, a zero-gradient for the variables involved is imposed, so that

$$\frac{\partial u}{\partial x} = 0; \qquad \frac{\partial \rho}{\partial x} = 0;$$
$$\frac{\partial T}{\partial x} = 0; \qquad \frac{\partial p}{\partial x} = 0.$$

This means that all variables are assumed to be constant with respect to x, starting with outlet section (that is the last section of the computational domain). In addition, since it is assumed the flow to be fully developed at this section, again v and w are set equal to 0.

Channel walls are characterized by a no-slip boundary condition, and they are isothermal with a temperature equal to the free stream temperature, given by

$$T_\infty = \frac{p_\infty \gamma}{\rho_\infty}. \tag{5.2}$$

Free stream temperature is set so that the Mach number for all the simulations performed is 0.4, so that it is low enough to be compared with incompressible data, and large enough to allow large timestep size. A schematic of the boundary conditions applied in a cross section along the $x - y$ plane is shown in figure 24.

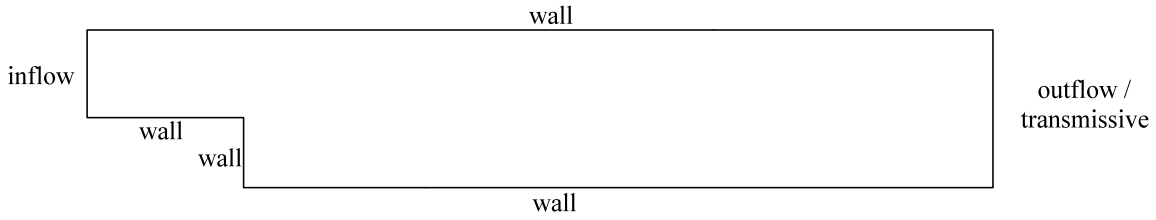


Figure 24: Boundary conditions applied for the backward-facing step, section along the z-direction.

Expansion ratio is defined as outlet channel height over inlet channel height, and therefore is given by:

$$ER = \frac{h_{out}}{h_{in}} = 1.942.$$

The hydraulic diameter is chosen accordingly to what presented by Armaly *et al.*, so that it is assumed equal to twice the inlet channel height, i.e. $D_h = 2h_{in} = 2.1224$. The Reynolds number is then defined as:

$$Re = \frac{\bar{u} D_h}{\nu}, \tag{5.3}$$

and is varied in the range $100 < Re < 400$. Additional reference flow parameters are presented in table IV.

TABLE IV: REFERENCE VALUES USED IN THE BACKWARD-FACING STEP STUDY.

Property	**Symbol**	**Value**	**Dimensional Unit**
Density	ρ_f	1.225	kg/m^3
Dynamic Viscosity	μ	1.789×10^{-5}	$kg/(m \cdot s)$
Kinematic Viscosity	ν	1.461×10^{-5}	m^2/s
Expansion Ratio	ER	1.942	-

Simulations are run in parallel on 80 processors on the MIE Cluster at UIC, for non-dimensional time units, with a CFL number of 0.9 to be on the safe side, so approximately 40,000 timesteps are required for every flow through to be computed.

5.2.2 Comparison with Experiments

In the following, the comparison between simulations' results and Armaly *et al.*'s experiments is performed, for the range of Re specified previously.

5.2.2.1 Reattachment length

Many authors have analyzed in depth the laminar flow region for the backward-facing step problem. In particular, the following comparison involves results presented, besides of Armaly *et al.*, also by Denham and Patrick [9]. As stated by Armaly *et al.* in their work, the reattachment length depends not only on the Reynolds number, but also on many other variables, such as the expansion ratio, the slope at the wall of the inlet velocity profile, and so on. A comparison of simulations' results with those obtained by Armaly *et al.* and Denham and Patrick is presented here, to prove the good agreement of simulations with results of both experiments.

Simulations for Reynolds numbers of 70, 100, 150 and 250 were performed in order to compare the numerically predicted reattachment lengths with those experimentally measured in literature. The two different types of boundary conditions predicted slightly different reattachment lengths, with a difference of approximately 4% more for the transmissive case. The overall distribution, however, remarks quite accurately the references chosen.

Results are presented in figure 25, where the Reynolds number Re of the case is on the horizontal axis, and its reattachment length normalized with the step, x_R/s, is shown on the

vertical axis. In addition to simulations' results, also the reference data are plotted. In the following representation, 'Sim.-Spec. Prof.' indicates simulations performed with u profile specified at the outlet section, while 'Sim.-Transmissive' indicates simulations performed with transmissive boundary conditions.

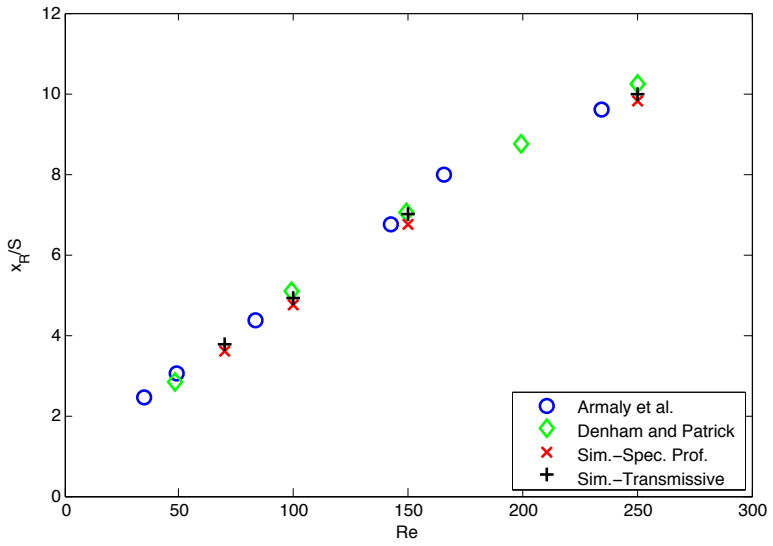


Figure 25: Comparison of reattachment lenghts from different authors as a function of the Reynolds number.

Considering the general trend in figure 25, it appears that results from transmissive boundary conditions are closer to the experimental data. Nevertheless, even results obtained with imposed u velocity profile reproduce a similar distribution, but in this case reattachment length is sligthly underestimated.

5.2.2.2 Velocity profile

This section refers to the comparison of the variation of the u velocity profile along the x direction. Two cases are run in particular, for $Re = 100$ and $Re = 389$, since it has been considered useful to perform a comparison with Armaly's experimental results. For both types of boundary conditions, u values along the test section were close enough so that only one of them is represented in the next figures.

Comparison for $Re = 100$ is presented in figure 26, that shows good agreement between simulations and literature results.

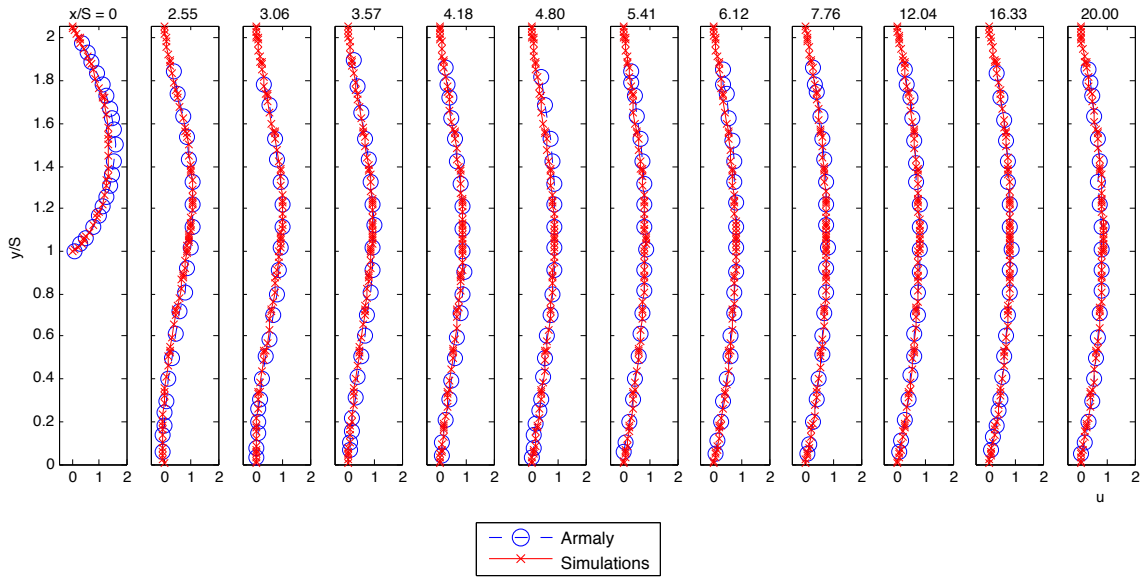


Figure 26: Comparison between Armaly's experimental data and simulations' results for u velocity, with $Re = 100$.

A similar comparison for $Re = 389$ is shown in figure 27. In this case too, good agreement between simulations and Armaly *et al.*'s experiments is achieved.

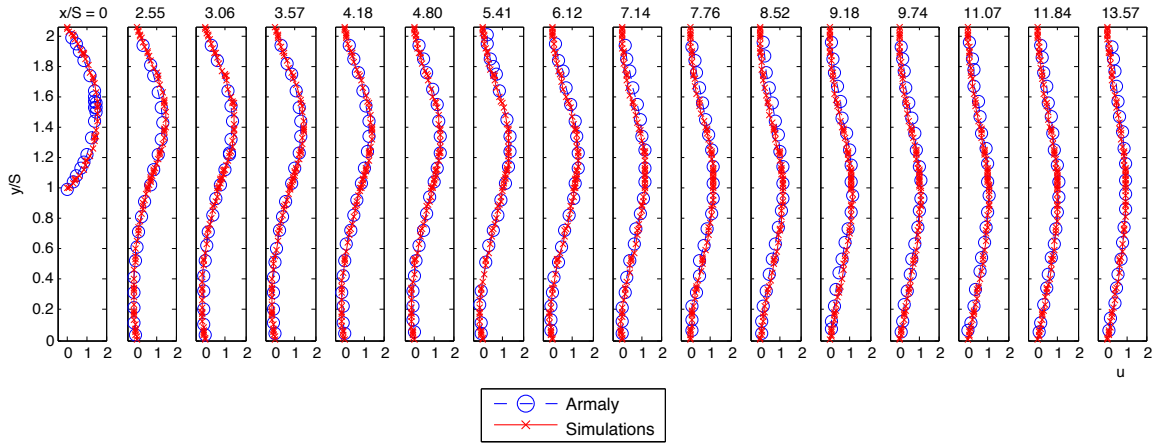
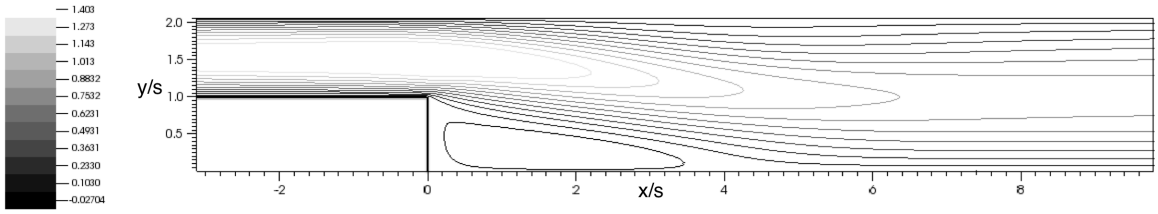


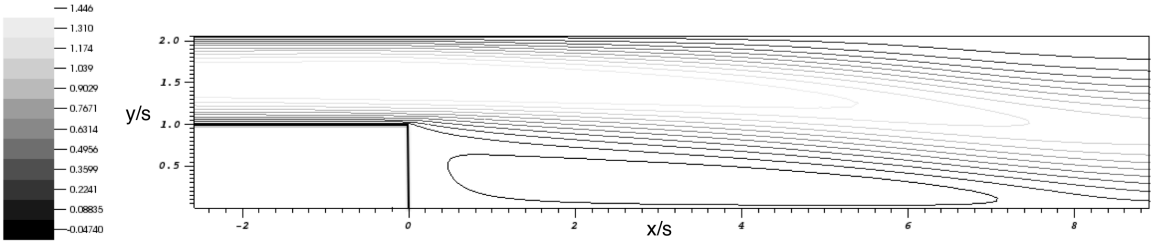
Figure 27: Comparison between Armaly's experimental data and simulations' results for u velocity, with $Re = 389$.

In both cases specified inlet u velocity profile is analogue to the profile found at the step, with a slight difference in maximum u value. In the rest of the test section, the distributions extrapolated from both simulations are in good agreement with those presented in the paper by Armaly *et al.*[3].

Velocity distributions and their contours, for the two cases compared with Armaly's results, are presented in figures 28 and 29.

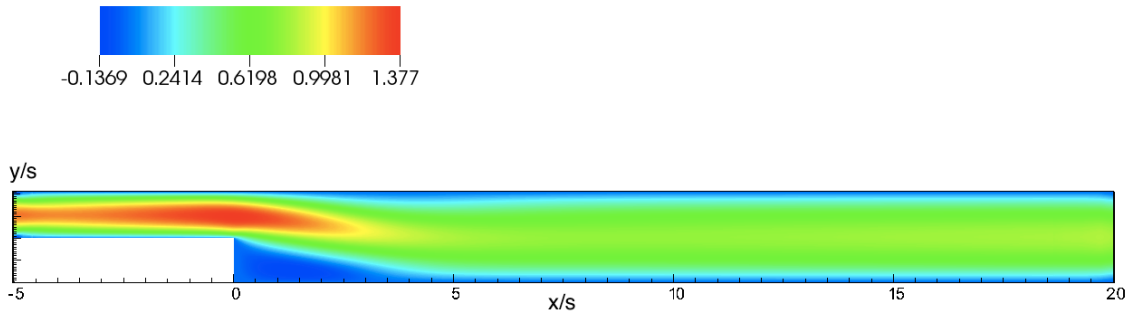


(a) u velocity contours for $Re = 100$.

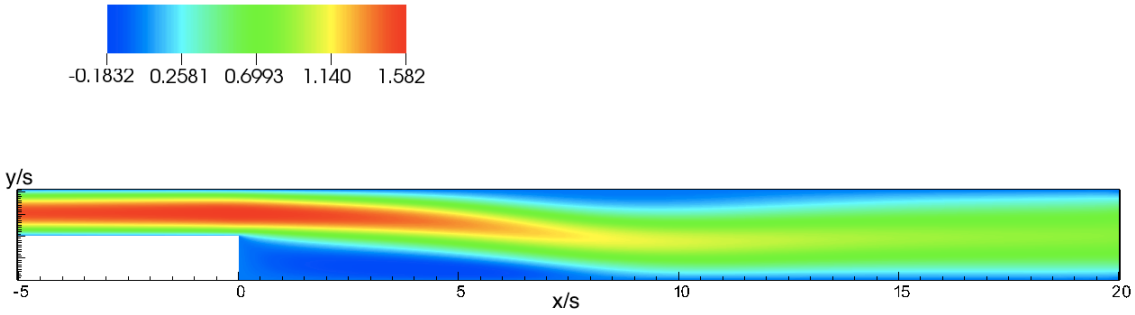


(b) u velocity contours for $Re = 389$.

Figure 28: Contours for u velocity in the vicinity of the step for the cases compared with Armaly *et al.*[3].



(a) u velocity distribution for $Re = 100$.

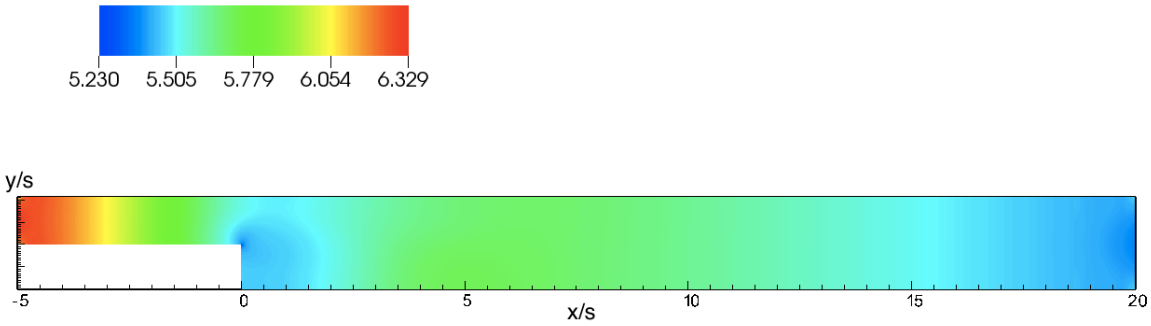


(b) u velocity distribution for $Re = 389$.

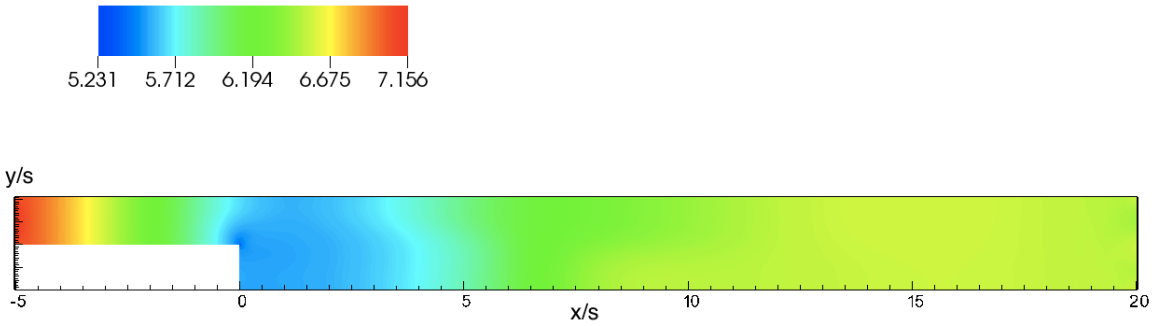
Figure 29: Distribution for u velocity in the whole test section for the cases compared with Armaly *et al.*[3].

5.2.2.3 Pressure Field

The pressure drop along the test section is about 0.6 non-dimensional units for $Re = 100$, and of 1.0 non-dimensional units for $Re = 389$, consistent with the fact of having a higher pressure drop when the Reynolds number of the flow is higher. Pressure distribution along the backward-facing step geometry for these two configurations is shown in figure 30.



(a) Pressure distribution for $Re = 100$, with imposed velocity profiles at inlet and outlet sections.



(b) Pressure distribution for $Re = 389$, with imposed velocity profiles at inlet and outlet sections.

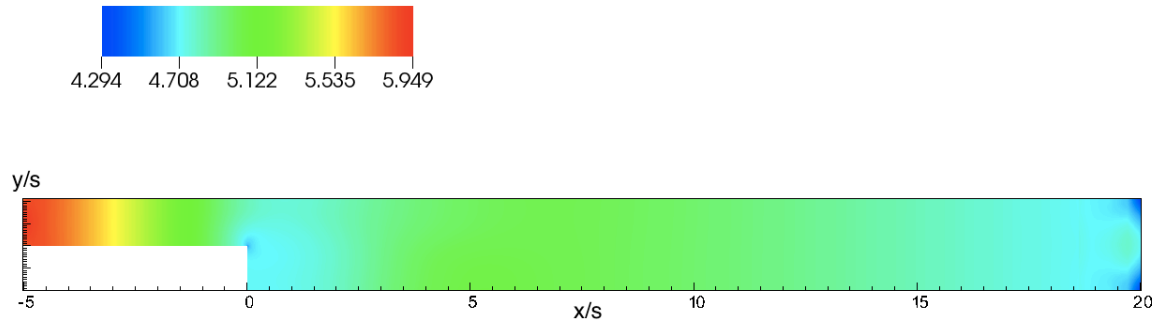
Figure 30: Distribution for pressure in the whole test section, for imposed velocity profiles at inlet and outlet test sections.

From comparison of the two images in figure 30, it is possible to notice how the case for the lower Reynolds number has a pressure drop in a relatively small area around the step corner, and this area broadens with increasing Re. For $Re = 389$, the decrease in pressure involves a wide area that is extended approximately from $x/s = -0.5$ to $x/s = 4.0$.

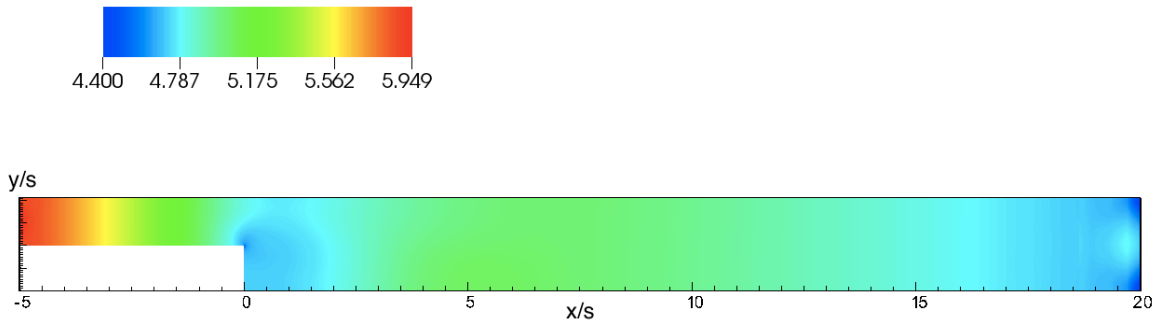
It is also interesting to compare the pressure distribution of the imposed velocity profiles case, with the transmissive boundary conditions case, shown in figure 31(a). Due to computational imperfections, in fact, the last sections before the outlet are characterized by an inaccurate pressure distribution. A possible solution to this problem is represented by the application of a 'sponge layer', responsible to reduce numerical instabilites in the vicinity of the outlet section. Its implementation is discussed for the case of the backward-facing step with the ramp, in section (5.3.2.2): The rest of the test section, on the contrary, has a pressure distribution that is comparable to that previously obtained, exception given by the peak pressure that reaches a value slightly lower. The difference in color rendering in figure 31(a) is due to the lower pressure reached near the outlet section, value that should be neglected as a result of numerical issues. Imposing a limit to the minimum pressure to be represented on the plot, it is possible to notice how pressure distribution with transmissive boundary conditions is similar to that obtained imposing u velocity profiles. This comparison is shown in figure 31(b) and 31(c).

5.2.3 Conclusions

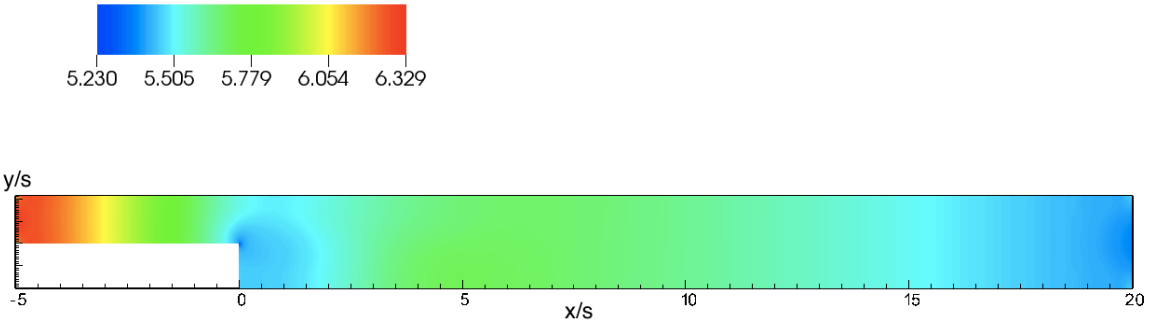
The simulations over the backward-facing step have shown accuracy with respect to what presented in literature. In particular, prediction of reattachment lengths for different Reynolds numbers, in laminar flow regime, is accurate within an error range of nearly 5%. Profiles of u



(a) Pressure distribution for $Re = 100$, with transmissive boundary conditions.



(b) Pressure distribution for $Re = 100$, with transmissive boundary conditions, and limited minimum pressure representation.



(c) Pressure distribution for $Re = 100$, with imposed velocity profiles at inlet and outlet sections.

Figure 31: Pressure distribution for $Re = 100$.

velocity for different Re along the test section vary accordingly to those presented by Armaly *et al.*[3], for both imposed velocity profiles and transmissive boundary conditions. Finally, the comparison of the two types of boundary conditions chosen shows analogies in their results, even if further testing and development is needed.

In summary, results obtained from the simulations are compatible to those presented in the literature. Therefore, the code employed is validated, and it will be used in the next section to analyze a similar geometry, where a ramp is added some dimensional units after the step, in order to have an outlet section with the same area of the inlet.

5.3 Flow over a Backward-Facing Step with a Ramp

While the previous part of this chapter focused on the flow over a backward-facing step, now this geometry is slightly modified to make it closer to a problem of widespread interest. The BFS geometry is altered adding a ramp after the step, so that a cavity is generated. After this cavity, the test section reduces its height so that outlet channel has the same height of the inlet channel.

This particular shape has been adopted to reproduce the actual geometry of a cavity combustor. Such type of combustors are employed in very different areas, ranging from ground-based power generation to aircraft engines. As already explained for the backward-facing step, but still valid for this case, a cavity adjacent to a flow is responsible of the formation of a recirculation zone. In this particular zone, the fluid has a longer residence time when compared to the bulk flow of the channel, so that this zone is ideal to provide heat for combustion. It is exactly because of their promise to be flame-holding devices that cavities have drawn the attention of the scramjet (Supersonic Combustion Ramjet) community [4], even if more research is needed to optimize its dynamic, i.e. maximize flame-holding capability and minimize losses.

Even if the geometry analyzed is interesting for its supersonic applications, further implementation of the code would be needed to obtain results for such cases. Therefore, this thesis is going to focus just on subsonic flows, where the bulk flow is in laminar regime. For such a kind of flow, the general features for the geometry analyzed, shown in figure 32, are

- Boundary layer development: along the inlet channel, flow is developed until it reaches the fully developed flow condition;

- Boundary layer separation: at the step corner, the sudden change in geometry is responsible for the separation of the boundary layer from the wall, which turns into a curved shear layer;
- Reattachment zone: instead of a well-defined reattachment length, used in the case of a simple backward-facing step, it is necessary to refer to a wider reattachment zone, since the separated boundary layer is not able to reattach on the bottom of the cavity (except for extremely low Re, or for sufficiently long cavities, cases that are not analyzed in this thesis);
- Recirculation zone: it is located adjacent to the cavity walls, and it moves closer to the ramp with increasing Reynolds number;
- Recovery to fully developed flow: after the corner formed by the ramp wall and the outlet channel bottom wall, the flow starts to redevelop to eventually reach its steady state fully developed condition.

In contrast to the backward-facing step case, where many relevant articles were found in the literature, for the geometry considered in the following not many studies based on its coupling with a subsonic flow have been conducted. In fact, subsonic flows are analyzed mainly for rectangular cavities, where their aspect ratio (length-to-depth) is changed, leading to the interpretation of the different experimental and/or numerical results obtained. The particular shape this thesis describes has, to the best of the author's knowledge, never been analyzed for subsonic flows with low Reynolds numbers, due to its wide application in supersonic conditions.

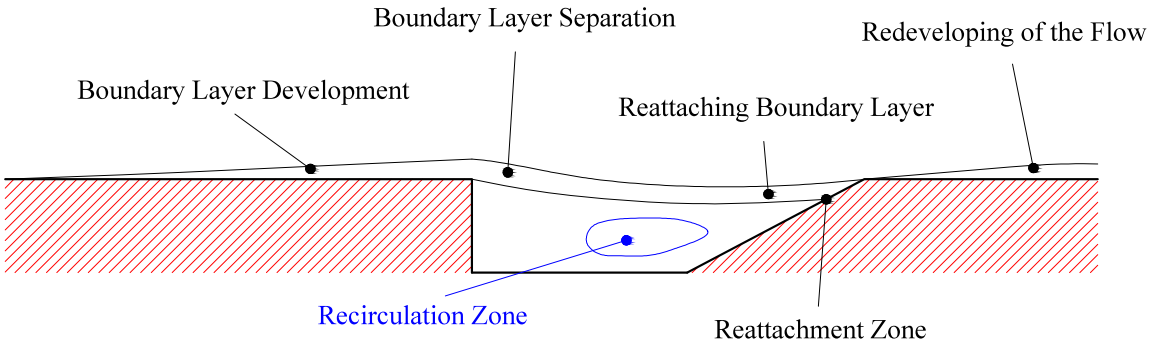


Figure 32: Principal features of the backward-facing step with ramp flow, for relatively low Re and for a channel with open top.

5.3.1 Computational Model

The problem geometry is shown in the schematic presented in figure 33, and dimensions adopted are taken from Tuttle *et al.*[6] about the investigation of a supersonic high-speed cavity. These dimensions are reported in table V. Lengths for inlet channel and outlet channel are not specified in the literature reference (indicated as *n.s.* in table V), therefore they are going to be assumed accordingly to the simulations' needs, i.e. long enough to let the flow fully develop before and after the cavity. This means that the channels, especially the outlet one, have to be chosen long enough to let the flow reach a fully developed condition, in order to allow the imposition of a fully developed u velocity profile as outflow boundary condition. In addition, even if Tuttle *et al.* have analyzed a geometry where the bottom wall of the channel is inclined by 2.5°, generating a diverging channel, such an inclination is neglected here, so that the inlet channel and the outlet channel are of the same height.

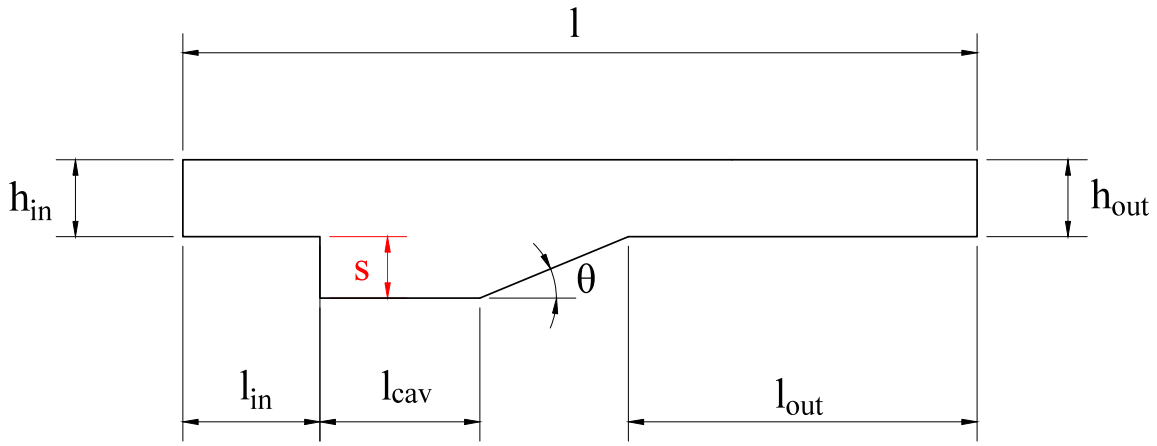


Figure 33: Geometrical dimensions for the case of the backward-facing step associated with the ramp.

TABLE V: DIMENSIONS OF THE BACKWARD-FACING STEP AND THE RAMP IN TUTTLE *ET AL.*'S SETUP.

Element	**Symbol**	**Value [cm]**
Inlet channel length	l_{in}	*n.s.*
Cavity length	l_{cav}	4.65
Outlet channel length	l_{out}	*n.s.*
Total length	l	*n.s.*
Inlet channel height	h_{in}	5.08
Outlet channel height	h_{out}	5.08
Cavity height	s	1.65
Theta	θ	22.5°

Tuttle *et al.*[6] provided a schematic of their setup, where it is clearly shown that their channel is approximately 90.0 cm long. Considering that the flow analyzed by this work is subsonic, and that no inclined bottom wall has been adopted (the diverging channel presented by Tuttle[6] makes the supersonic flow accelerate along its length), shorter lengths have been specified with respect to those that is possible to extrapolate from the drawing.

In the process of non-dimensionalization, a step height of 0.65 is imposed. With such a choice, the cavity length becomes 1.815, and the inlet and outlet channel height results of 2.0. The inclination θ of the ramp is mantained consistent with the reference, so that an angle of 22.5° is assumed. Since, as stated earlier in this section, inlet and outlet channel lengths have not been specified by Tuttle *et al.*, it has been chosen 3.0 as non-dimensional length for the inlet channel, and a total length of the test section of 13.0. Finally, a depth of 1.0 is specified along the z direction. The range of Reynolds numbers chosen, i.e. $100 < Re < 400$, ensures that the flow remains laminar and two-dimensional, and therefore results are going to be analyzed in the middle section along z axis.

All dimensions described so far are presented in table VI, together with the conversion among dimensional and non-dimensional units.

Tuttle *et. al.* used particle image velocimetry to characterize flow properties. Such method consists in using sufficiently small tracer particles in the fluid, so that they are assumed to closely follow the flow dynamics, and register their motion to calculate speed and direction of the flow being studied.

TABLE VI: DIMENSIONS OF THE BACKWARD-FACING STEP WITH THE RAMP FOR NUMERICAL SIMULATIONS' SETUP.

Element	**Symbol**	**Value [cm]**	**Non-dimensional Value**
Inlet channel length	l_{in}	7.62	3.0
Cavity length	l_{cav}	4.65	1.815
Outlet channel length	l_{out}	16.805	6.616
Total length	l	33.02	13.0
Inlet channel height	h_{in}	5.08	2.0
Outlet channel height	h_{out}	5.08	2.0
Step height	s	1.65	0.65
Theta	θ	22.5°	-

In this work, a sixth order polynomial has been used with 56, 16 and 1 domains in the x, y and z directions, respectively. For such a polinomial order, Lobatto points are respectively 448, 128 and 8 in each direction. A grid with this features has shown adequate grid independence, with its quadrilateral mesh constituted of 52224 nodes for each $x-y$ plane section, that correspond to a total of 417792 nodes for the whole mesh.

For lower polynomial orders and a lower number of elements, in fact, results have been underestimated, especially in the cavity. Two grids with less elements and a lower polynomial order have been tested, with 27648 nodes for each $x-y$ plane section (165888 total), and with 35280 nodes for each $x-y$ plane section (246960 total), respectively. For a higher polynomial order and a higher number of elements, instead, no significant increase in accuracy has been

achieved. The grid tested was constituted by 66096 nodes for each $x - y$ plane section (594864 total).

The grid is not uniform, and it is made finer in the points of interest of the flow, and in particular

- Near the step corner;
- Near the top and the bottom walls of the channel;
- Near the cavity walls;
- In the shear layer.

A power law is chosen for the nodes distribution in the inlet channel along the x direction, approaching the step corner, with a last length of 0.15 non-dimensional units. The cavity has also a power law distribution, with a first length of 1.0. On the ramp, on the contrary, the x distribution of points is assumed uniform, since the mesh has been made fine enough so that no further refinement is needed. The outlet channel also has a power distribution, with a first length of 0.15. In the y direction, instead, a finer grid is required near top and bottom wall of channel and cavity. Therefore, a double power law is specified, with 0.05 set both as last and as first length for the top part of the channel (inlet channel, outlet channel, and the zone in between), and 0.04 as last and as first length for the bottom part (cavity and ramp). An example of how the mesh is constructed on the $x - y$ plane is shown in figure 34. In this case too, just one element is created along the z direction, so that no custom distribution of points is made along that axis.

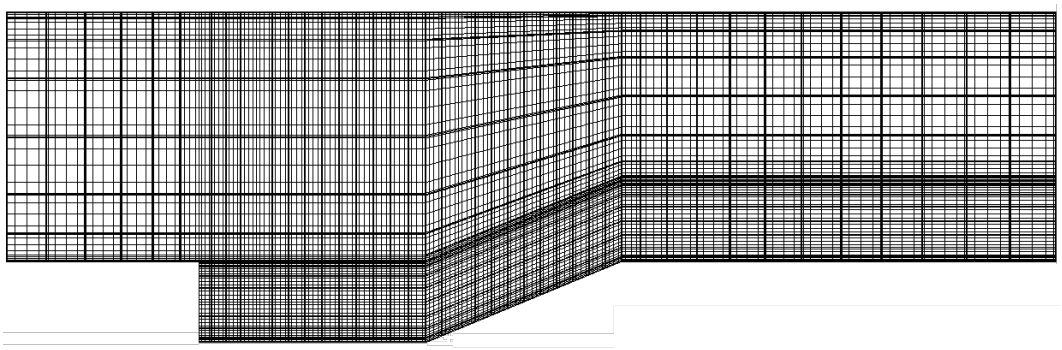


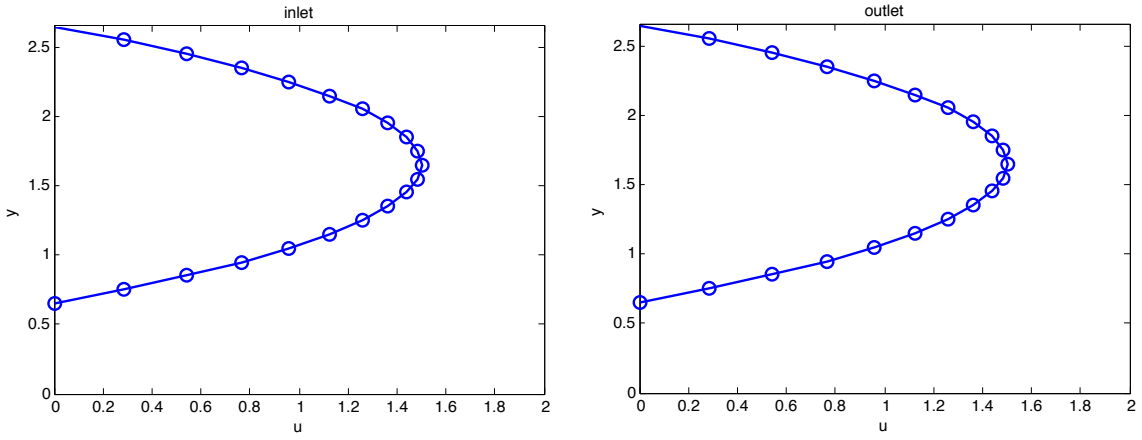
Figure 34: Example of mesh refinement for the backward-facing step with the ramp, $x - y$ plane section.

The boundary condition at $x = -3.0$, at the inlet section, is an inflow boundary condition with a fully developed u velocity profile specified. As done for the standalone backward-facing step, it is chosen a profile where the maximum value for the velocity in the x direction is $u_{max} = 1.5$, so that the mean velocity at the entrance is calculated with (Equation 5.1), and it is $\bar{u} = 1.0$. Once more, since the inflow profile is fully developed, $v = w = 0$. Density and pressure at the inlet section are, respectively, $\rho = 1.0$ and $p = 4.464$.

At $x = 10.0$, in correspondance of the outlet section, two different outflow boundary conditions are tested.

The first kind of boundary conditions is again based on the specification of outflow boundary values for the u velocity profile at the outlet section. Being the outlet section far enough from the cavity, the flow is assumed capable to reach its steady state along the outflow channel, and therefore the profile corresponds to a condition of fully developed flow. In addition, considering that the outlet section has exaclty the same geometrical dimensions of the inlet section, the

same velocity profile used for the inflow condition is specified as mortar values, as suggested in figure 35.



(a) Inlet velocity profile, at $x = -3.0$.

(b) Outlet velocity profile, at $x = 10.0$.

Figure 35: Examples of the u velocity profiles imposed as inflow and outflow boundary conditions for a backward-facing step coupled with a ramp problem.

In addition, values on the mortar for v and w are set equal to 0, and density and pressure are imposed to assume again their atmospheric values, i.e. $\rho = 1.0$ and $p = 4.464$, that is physical considering the outlet channel is long enough to leave the flow the time to fully develop again.

The second kind of outflow boundary conditions employed is constituted by transmissive boundary conditions. Equations describing these boundary conditions can be found in sections (5.1.2) and (5.2.1), therefore they are repeated here. Derivatives along the x direction of u, ρ

and p are set to zero at the outlet section, that implies also temperature T does not vary along the x direction. In addition, it is also imposed $v = w = 0$, to make the problem well defined.

Channel walls are characterized by a no-slip boundary condition, and they are set isothermal with a temperature $T_w = T_{atm}$ given by (Equation 5.2). Temperature T_{atm} is chosen so that $Ma = 0.4$ for the flows computed, low enough to consider them nearly incompressible. A schematic of the boundary conditions applied in a cross section along the $x - y$ plane is shown in figure 36.

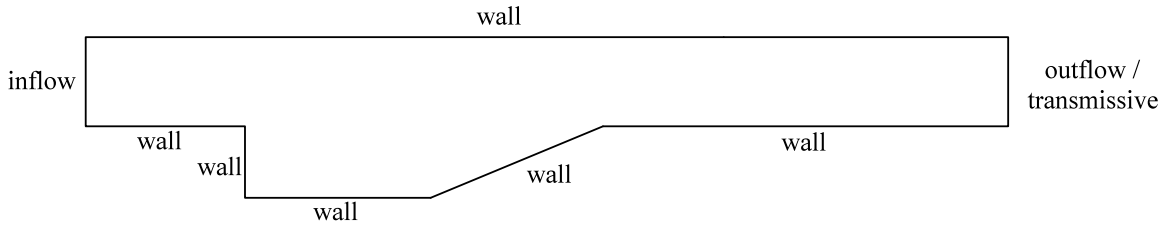


Figure 36: Boundary conditions applied for the backward-facing step with the ramp, section along the z-direction.

It is possible to define two characteristic parameters to describe the flow:

- ER_1, that is the expansion ratio between the channel height at the cavity and the inlet channel height, i.e.

$$ER_1 = \frac{h_{in} + s}{h_{in}} = 1 + \frac{s}{h_{in}} = 1.325; \tag{5.4}$$

- CR_2, that is the compression ratio between the channel height at the cavity and the outlet channel height, i.e.

$$CR_2 = \frac{h_{in} + s}{h_{out}} = 1.325. \tag{5.5}$$

With the outlet channel and the inlet channel being of the same height, $ER_1 = CR_2$. The overall test section expansion ratio is given by

$$ER = \frac{ER_1}{CR_2} = 1.0, \tag{5.6}$$

and is equal to unity because of the same geometric characterstics of inlet and outlet channels. The hydraulic diameter is chosen in accordance to the case of the backward-facing step previously studied, so that $D_h = 2h_{in} = 4.0$. The Reynolds number is once more defined through (Equation 5.3):

$$Re = \frac{\bar{u}\, D_h}{\nu},$$

where ν is the kinematic viscosity of air. In this study, Re is varied in the range $100 < Re < 400$. The value of ν and those of other parameters involved are briefly summarized in table VII.

The simulations are run in parallel on 80 processors, currently available on the MIE Cluster at UIC. The CFL number has been assumed equal to 0.9 in case of the specified u profile BCs, while it has been reduced to 0.4 in case of transmissive boundary conditions, in order to reduce instability. Approximately 55,000 timesteps are needed per flow through, with a computational time of nearly 2 hours.

TABLE VII: REFERENCE VALUES USED IN THE BACKWARD-FACING STEP WITH RAMP STUDY.

Property	**Symbol**	**Value**	**Dimensional Unit**
Density	ρ_f	1.225	kg/m^3
Dynamic Viscosity	μ	1.789×10^{-5}	$kg/(m \cdot s)$
Kinematic Viscosity	ν	1.461×10^{-5}	m^2/s
Expansion Ratio	ER	1.0	-

5.3.2 Results

This section presents results obtained for the geometry considered, for both types of outflow boundary conditions, and in the Reynolds number range specified earlier in this chapter. In particular, reattachment zone, u velocity distribution and pressure distribution are analyzed.

5.3.2.1 Reattachment Zone

The present section compares results obtained for the reattachment zone with the different kinds of boundary conditions imposed. As described in section (5.3), because of the geometric characteristics of the cavity, it is not possible to define a reattachment length as it has been done for the backward-facing step case. As clearly explained by Ben-Yakar and Hanson [4], in the case of a supersonic flow, cavities can be divided into two categories depending on their length-to-depth ratio (that following this thesis's notation is indicated as l_{cav}/s):

- Open cavities, for a length-to-depth ratio $l_{cav}/s < 7 - 10$. The cavity is called 'open' because the shear layer reattaches to the back face of the cavity (the ramp in the case considered);
- Closed cavities, for a length-to-depth ratio $l_{cav}/s > 10 - 13$. In this case, the cavity is called 'closed' because the shear layer reattaches to the lower wall.

The critical length-to-depth ratio, i.e. the l_{cav}/s at which transition between the two regimes occurs, depends also on the flow Mach number, on the boundary layer thickness at the step, and on the cavity width (for three-dimensional flows). The case considered is characterized by a length-to-depth ratio $l_{cav}/s = 1.815/0.65 = 2.792$, so that it corresponds to an open cavity.

Ensured that the shear layer reattaches on the back face of the cavity, where the ramp is situated, it is then possible to plot the variation of reattachment point (described by its y coordinate) with the Reynolds number for the cases considered. As it can be deducted from figure 37 trend is not linear with increasing Reynolds number. In particular, for the lower range of Reynolds numbers the simulations with specified outflow u velocity give a slightly lower value of reattachment point y_R, while for the highest Re used the lowest y_R prediction is made by transmissive boundary conditions. In all the cases considered the difference between the reattachment points predicted with the two types of boundary conditions remains in the order of 1-2% of the value predicted.

5.3.2.2 Velocity Profile

This section compares the different u velocity profiles obtained with increasing Reynolds numbers, ranging from a minimum of $Re = 100$ to a maximum of $Re = 389$. Such a comparison

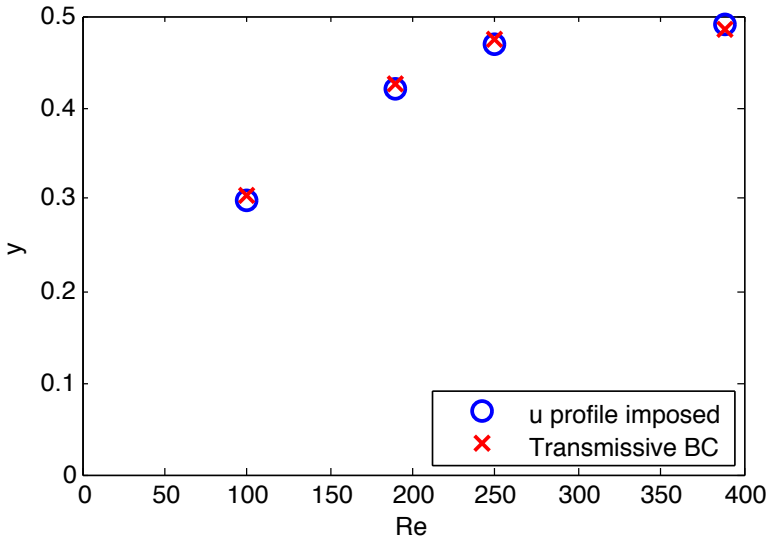
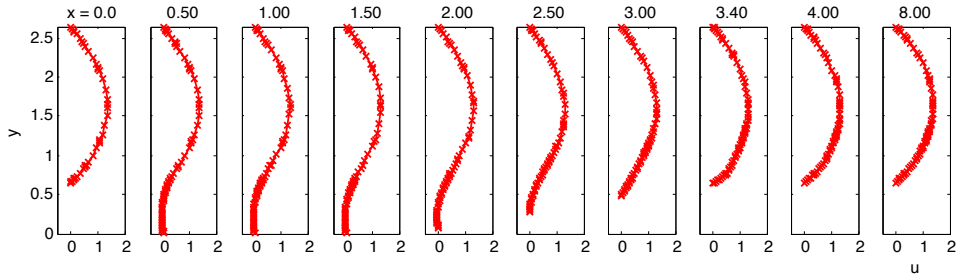


Figure 37: Comparison of boundary layer reattachment zones for the backward-facing step with ramp, for different types of outflow boundary conditions.

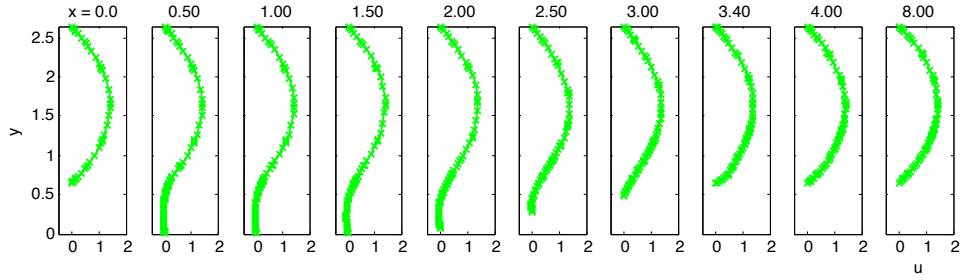
is shown in figure 38. Only the u profile for imposed u boundary conditions is plotted, since transmissive boundary conditions are examined later with a plot of the complete test section. As the Reynolds number increases, a higher maximum velocity is registered for the upper part of the test section, and also a higher value of u (in modulus) is obtained in the cavity, showing a more effective recirculation in the cavity.

In addition, as shown in figure 39, as the Reynolds number increases the center of the recirculation zone tends to move farther from the step and closer to the ramp.

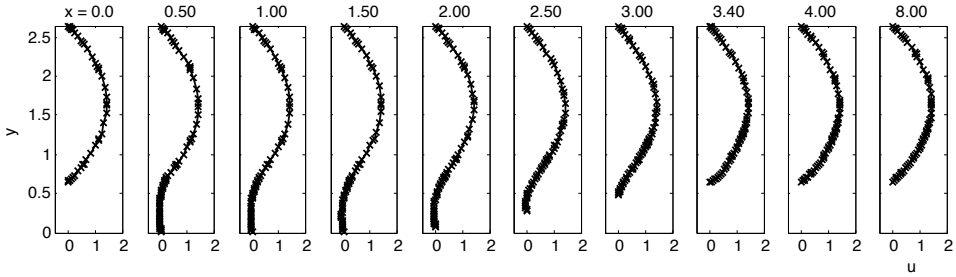
Finally, figures 40 and 41 show the u velocity distribution along the whole test section for imposed u velocity profile and transmissive outflow boundary conditions, respectively. Distributions for the lowest and the highest Reynolds numbers tested have been represented, since



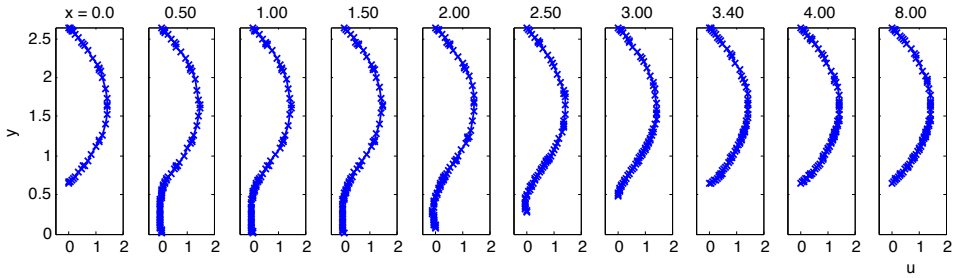
(a) $Re = 100$.



(b) $Re = 190$.

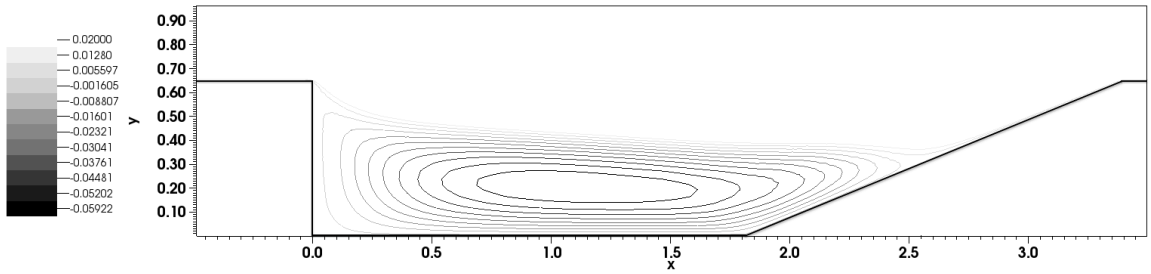


(c) $Re = 250$.

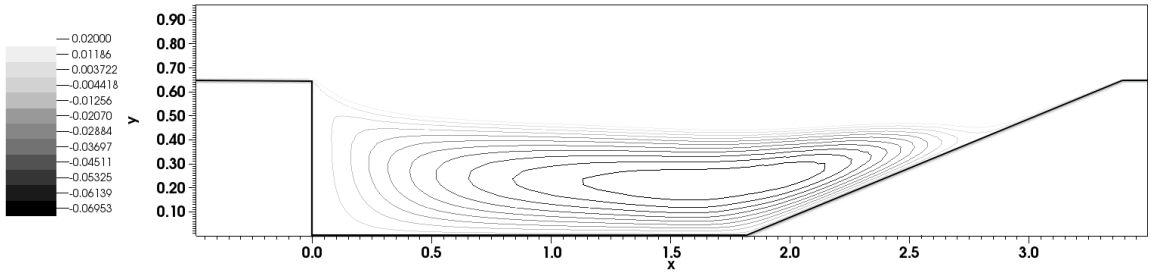


(d) $Re = 389$.

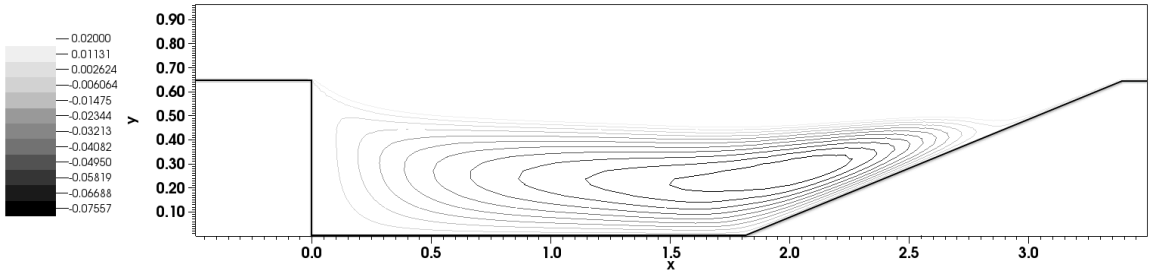
Figure 38: Graphical representation of u velocity distribution along vertical cross sections, for different x coordinates, for the backward-facing step with ramp.



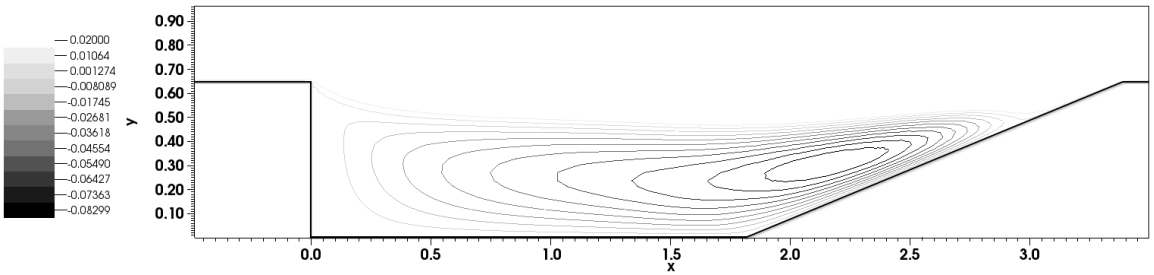
(a) $Re = 100$.



(b) $Re = 190$.



(c) $Re = 250$.



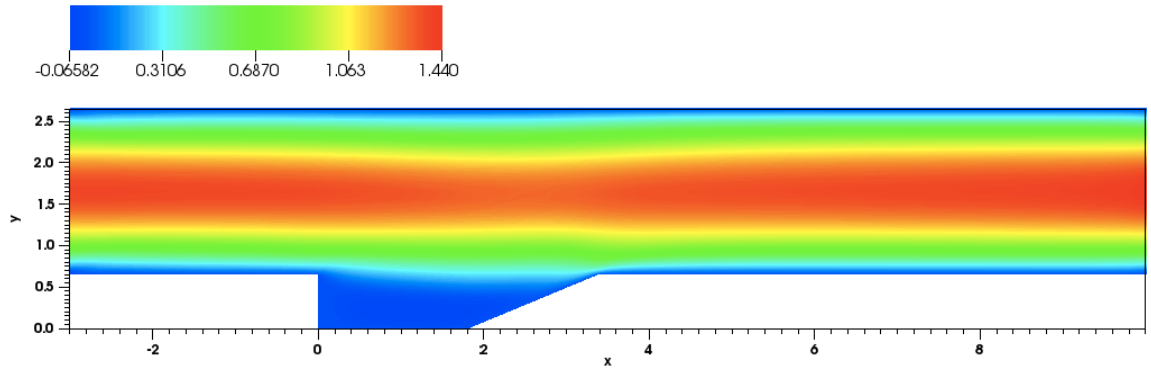
(d) $Re = 389$.

Figure 39: Graphical representation of main recirculation zone for increasing Reynolds numbers, for the backward-facing step with ramp, contours for u velocity.

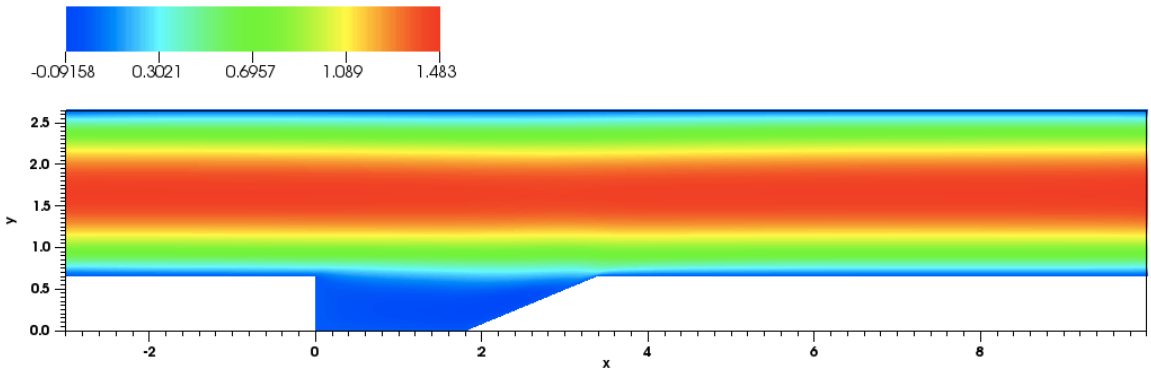
the extreme cases are considered the most meaningful to be analyzed for the flow. The plots for transmissive boundary have been forced within the same limits of those obtained for u profile specified, so that it becomes easier to compare results coming from both types of simulation.

In particular, it is possible to notice how transmissive boundary conditions overestimate the value of u, especially in the channel centre, when compared to the specified u velocity profile boundary condition. Comparing the values assumed by the u velocity along the whole test section, at a height $y = 1.65$ (i.e. exactly at the middle height of the inlet/outlet channel) and in the middle section, as shown in figure 42, it is possible to notice how the case where transmissive boundary conditions have been imposed shows a steep instability at the exit, that leads to a propagation of numerical errors that involve all the rest of the test section, errors that are responsible of non physical results.

To limit this phenomenon and obtain physical results, it is necessary to reduce as much as possible the numerical issues at the exit. A possible solution is the implementation of a buffer domain, to be applied before the outlet section, in order to smooth down such instability. This approach has been succesfully proposed and adopted by Pruett *et al.*[35], but with the inconvenience that the viscous terms of the basic equations must be manipulated in the buffer region. This work refers, therefore, to a more convenient approach, suggested by Guo *et al.*[36] and adopted also by Adams[37]. This solution introduces a sponge layer at the outflow boundary, which typically is one boundary layer thickness long. Since in this thesis the major instability has been identified within the last elements in the x direction, the sponge layer length is

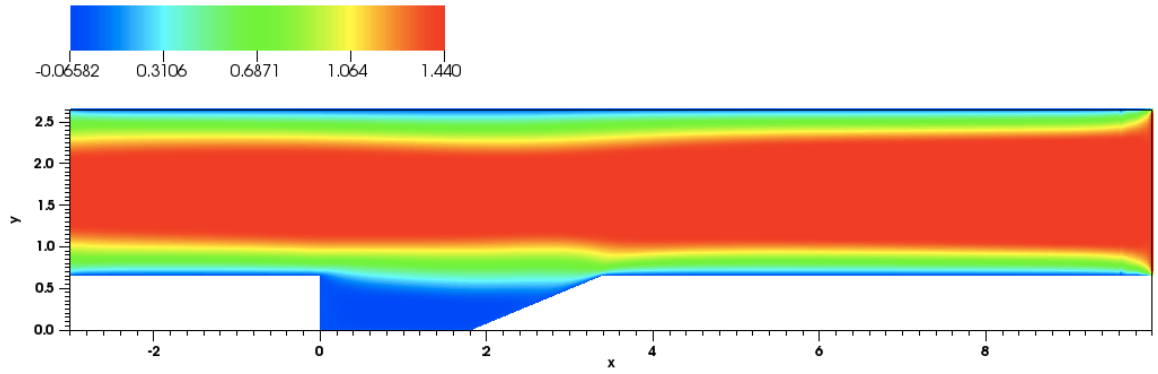


(a) $Re = 100$.

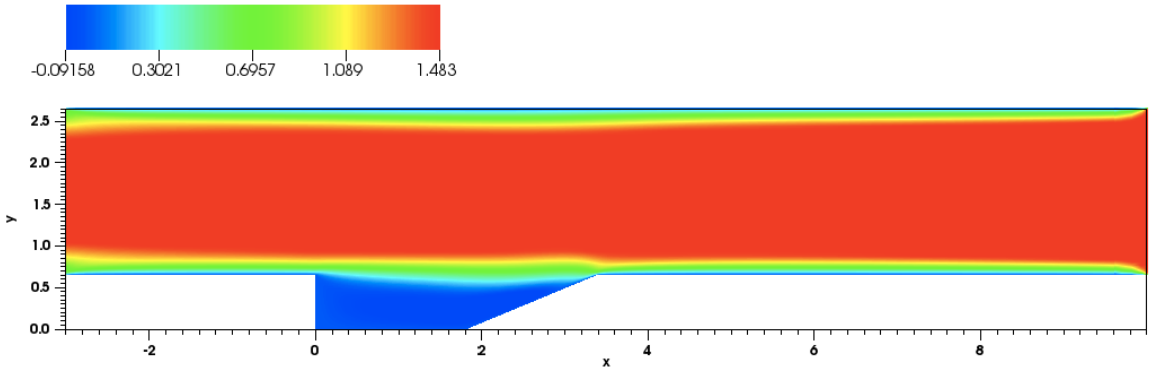


(b) $Re = 389$.

Figure 40: Distribution for u velocity in the whole test section for the backward-facing step with ramp, imposed u profile at the outlet section.



(a) $Re = 100$.



(b) $Re = 389$.

Figure 41: Distribution for u velocity in the whole test section for the backward-facing step with ramp, transmissive outflow boundary conditions.

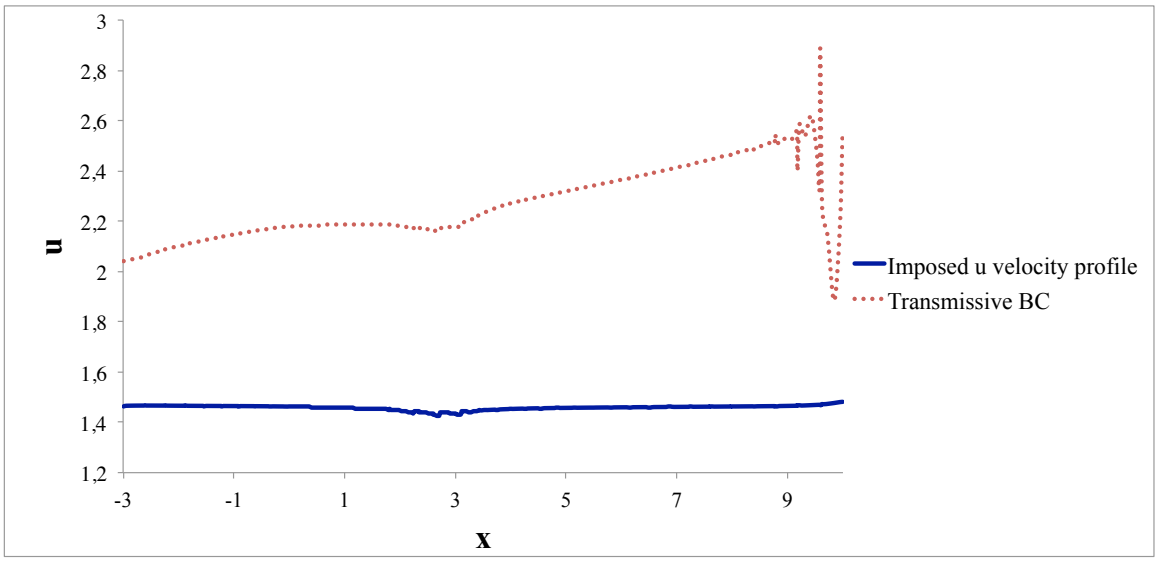


Figure 42: Comparison of u velocity along the whole test section between imposed u BC and transmissive BC for $Re = 389$, at a height $y = 1.65$, and in the middle section.

assumed equal to the x length of the last domain. The region where the sponge layer is applied is highlighted in figure 43.

With the sponge layer adopted, a vector quantity is added to the right-hand side of the Navier-Stokes equations, defined as

$$\mathbf{Z} = -\sigma(x)(\mathbf{W} - \mathbf{W_0}), \tag{5.7}$$

where $\mathbf{W}$ indicates the vector of conservative variables, and $\mathbf{W_0}$ indicates their values for a reference steady flow. In the case considered, the vector $\mathbf{W_0}$ is defined referring to the condition

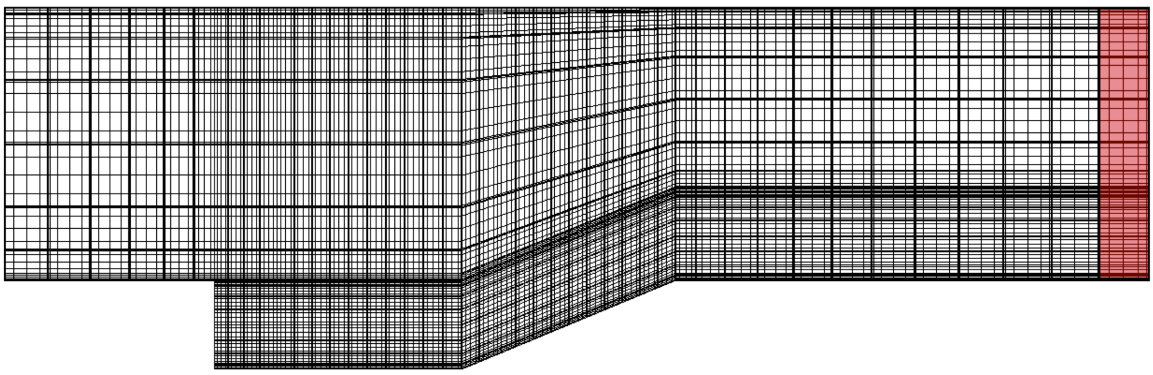


Figure 43: Test section mesh, where the region highlighted represent the zone where the sponge layer is applied.

of fully developed channel flow. The function $\sigma(x)$ is responsible for damping instabilities in the sponge layer, and it is defined as

$$\sigma(x) = \begin{cases} 0, & \text{outside the sponge region,} \\ A_s(N_s+1)(N_s+2)\frac{(x-x_s)^{N_s}(L_x-x)}{(L_x-x_s)^{N_s+2}}, & \text{inside the sponge region.} \end{cases} \tag{5.8}$$

In (Equation 5.8), it is possible to notice how the damping function $\sigma(x)$ is a function of only the x coordinate. x_s and L_x indicate the endpoints of the sponge layer region, i.e. x_s is the x coordinate at which the sponge layer zone begins, and L_x the x coordinate at which the sponge layer ends (that corresponds also to the x coordinate of the last points of the domain). The sponge region is therefore represented by the points such that $x_s < x < L_x$. A_s and N_s are two parameters, tuned accordingly to the damping wanted. In this case, they are chosen as their typical values, i.e. $A_s = 4$ and $N_s = 3$.

The sponge layer has been implemented in the code, and it is considered as a sink by Navier-Stokes equations. Results obtained for the velocity distribution in the test section for $Re = 100$ and $Re = 389$ are presented in figures 44(a) and 44(a), respectively.

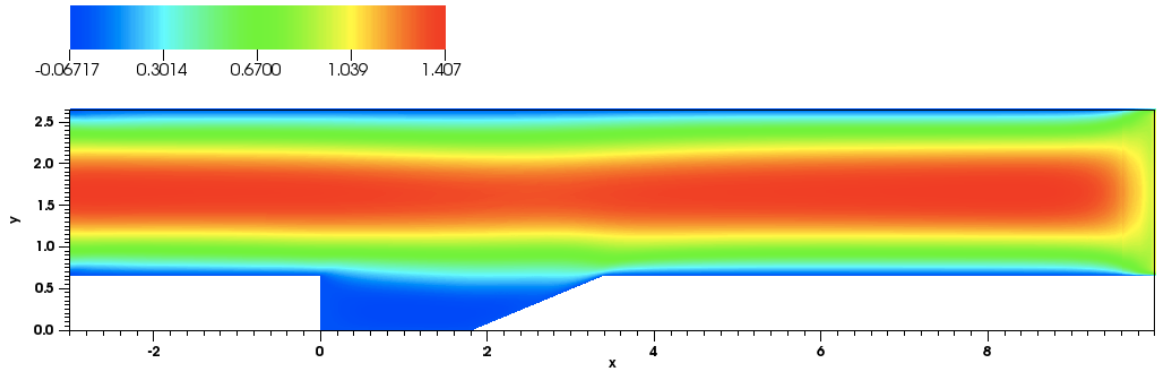
In addition, it is possible to plot the evolution of the u velocity value along the same line considered in the case of figure 42. This comparison is shown in figure 45.

From figure 45 it is possible to notice that the sponge layer has effectively damped the oscillations near the outlet section, but most of all has made possible good agreement of the results between the two types of boundary conditions. Although the damping near the outlet still needs some adjustements, achievable tuning the parameters in (Equation 5.8), results in the rest of the test section are in good agreement with those obtained with imposed u velocity profile as outflow boundary condition. For this reason, it is necessary to neglect results in the sponge layer zone, as they do not correspond to a physical phenomenon, but rather to numerical instability, that has been damped but not completely eliminated.

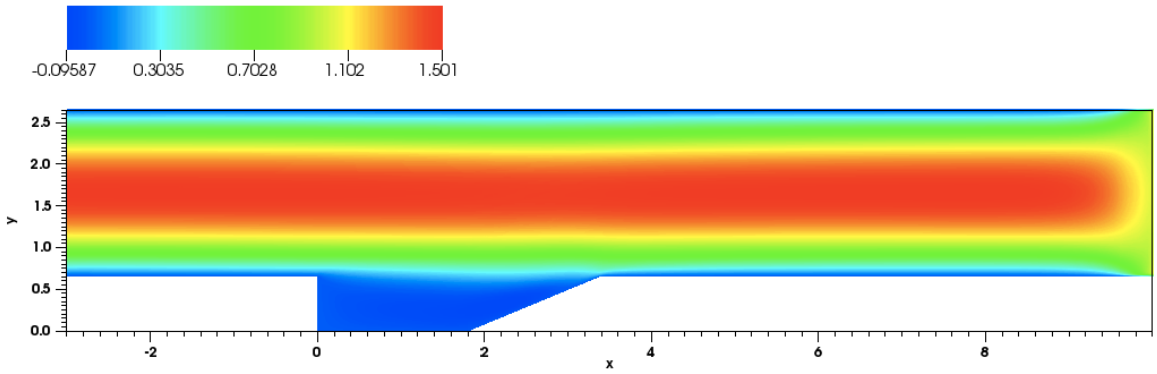
5.3.2.3 Pressure Field

The pressure drop from the inlet to the outlet section is about 0.4 non-dimensional units for $Re = 100$, and of 1.0 non-dimensional unit for $Re = 389$, again consistent with higher pressure drop in case of higher Reynolds number, as it happened for the backward-facing step. Pressure distribution for specified u velocity profile at the outlet section is shown in figure 46, for the lowest and the highest Reynolds numbers tested.

As it happened in the case of u velocity distribution, also for the pressure p the oscillations encountered near the outlet by transmissive boundary conditions are responsible for wrong



(a) $Re = 100$.



(b) $Re = 389$.

Figure 44: Distribution for u velocity in the whole test section for the backward-facing step with ramp, transmissive outflow boundary conditions with sponge layer at the outflow boundary.

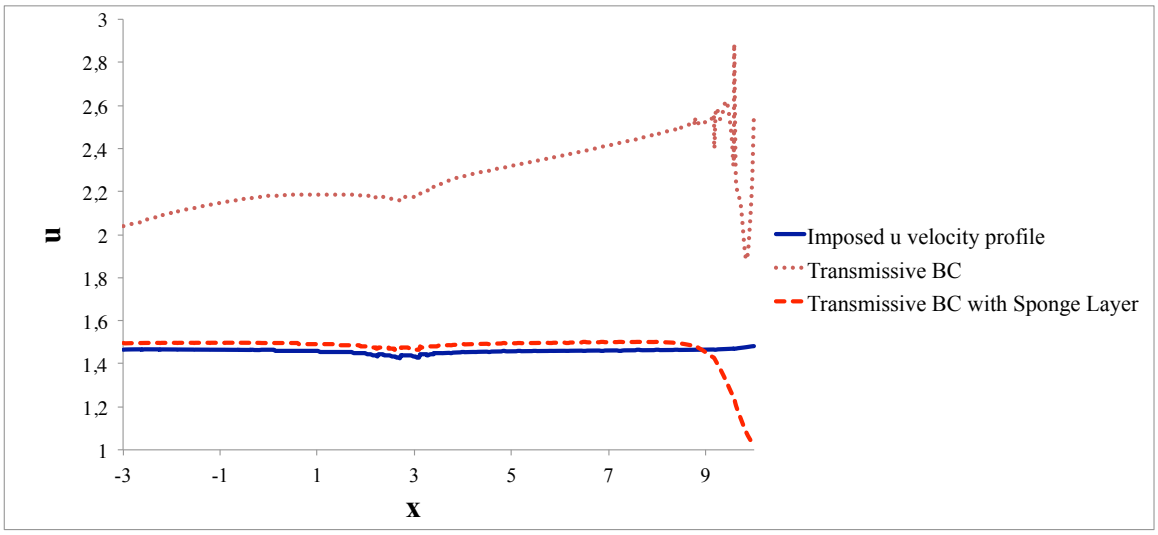
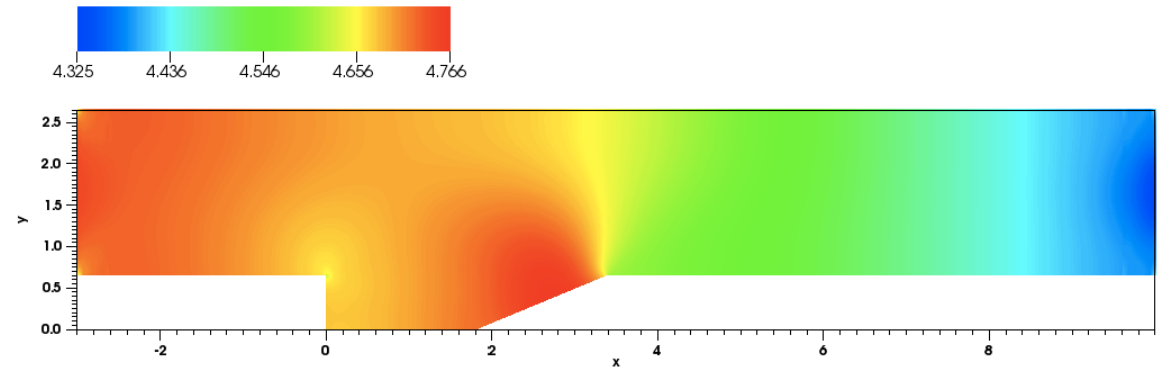


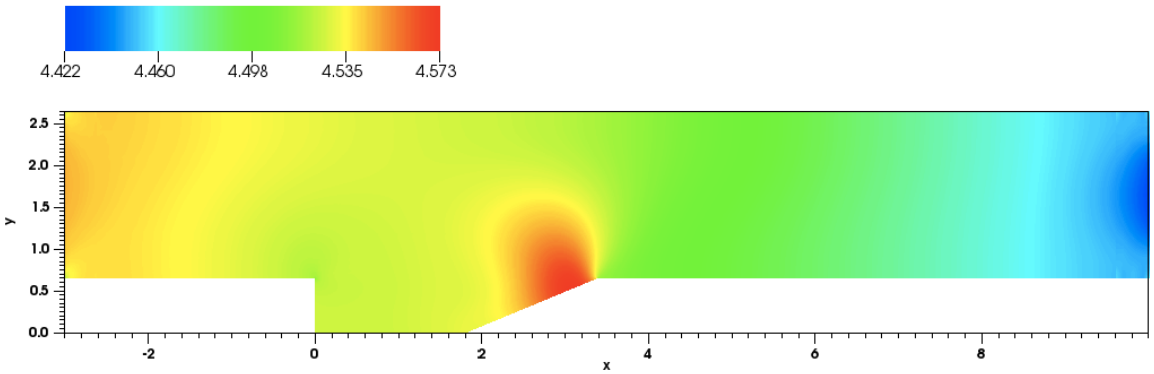
Figure 45: Comparison of u velocity along the whole test section between imposed u BC, transmissive BC and transmissive BC with sponge layer for $Re = 389$, at a height $y = 1.65$, and in the middle section.

results in the rest of the test section. With the implementation and the application of the sponge layer, whose features are explained in section (5.3.2.2), pressure distribution becomes in good agreement with that obtained in the case of u velocity imposed boundary condition. Pressure distribution in the case of transmissive boundary condition with sponge layer is presented in figure 47. The comparison of results obtained for the three different cases are shown in figures 48 and 49.

Pressure distribution, with the application of the sponge layer, is in good agreement with that obtained imposing the u velocity profile as outflow boundary condition. In the region where the sponge layer is applied, some further damping in needed, especially tuning parameters A_s and N_s of the damping function. In the rest of the section, instead, the values obtained are

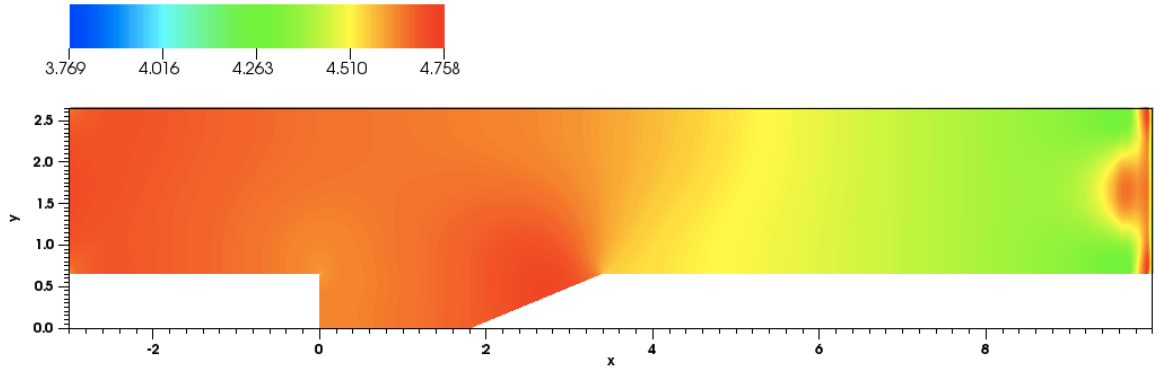


(a) $Re = 100$.

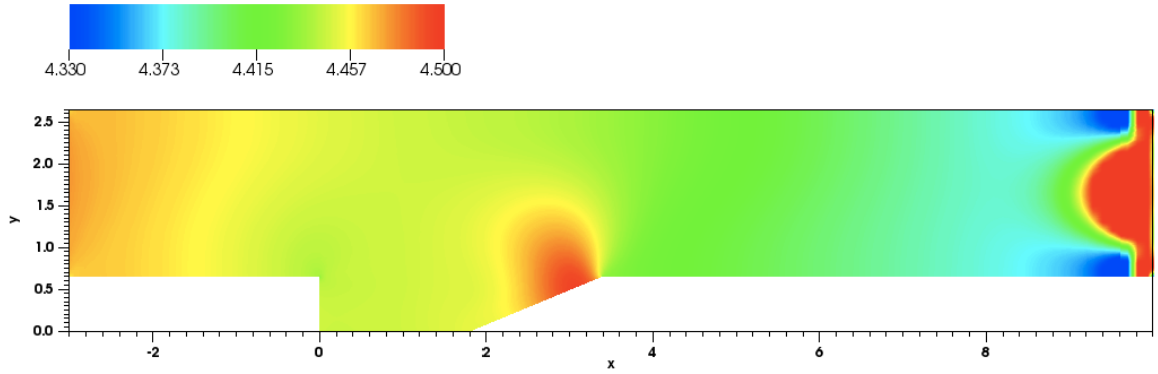


(b) $Re = 389$.

Figure 46: Distribution for pressure p in the whole test section for the backward-facing step with ramp, u profile specified at the outlet.



(a) $Re = 100$.



(b) $Re = 389$.

Figure 47: Distribution for pressure p in the whole test section for the backward-facing step with ramp, transmissive boundary condition with sponge layer applied in the last domain.

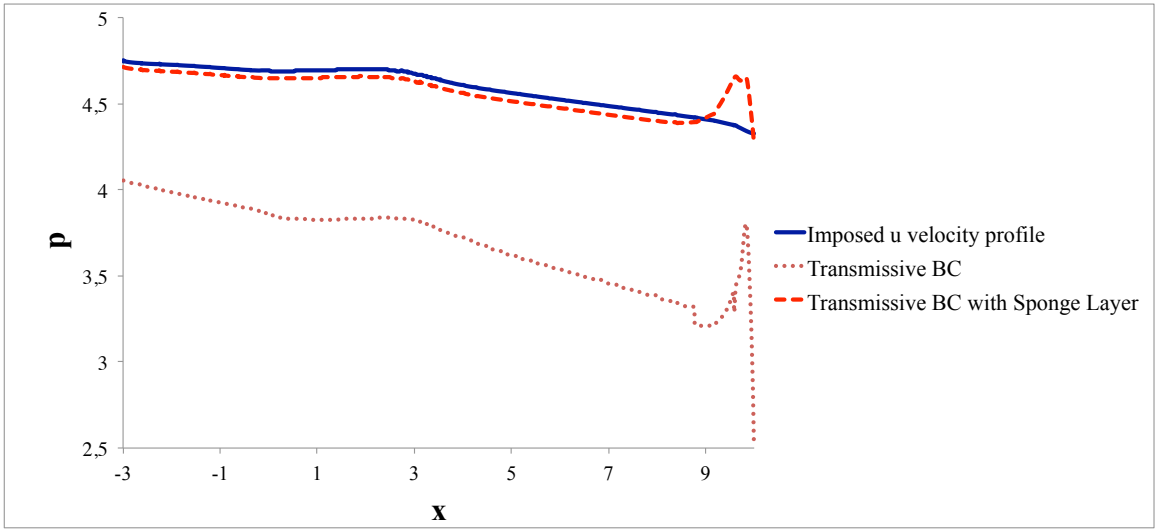


Figure 48: Comparison of pressure along the whole test section between imposed u BC, transmissive BC and transmissive BC with sponge layer for $Re = 100$, at a height $y = 1.65$, and in the middle section.

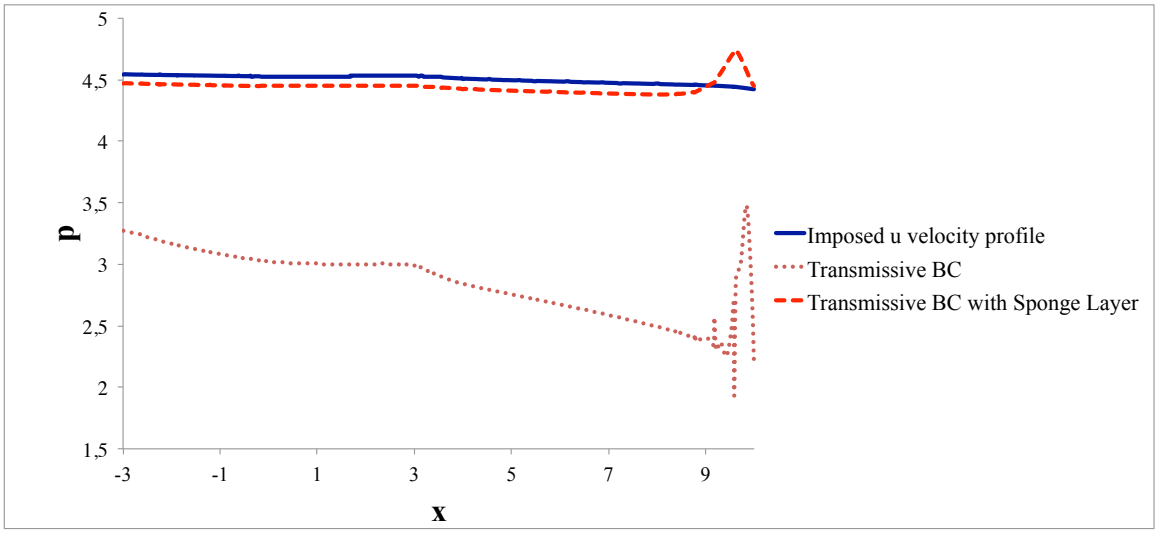


Figure 49: Comparison of pressure along the whole test section between imposed u BC, transmissive BC and transmissive BC with sponge layer for $Re = 389$, at a height $y = 1.65$, and in the middle section.

consistent with those obtained applying the other type of boundary condition. If the sponge layer was not applied, as it can be seen from lines representative of standalone transmissive boundary conditions in figures 48 and 49, the values obtained throughout the whole test section would have been unphysical, because of the propagation of numerical errors from the outlet section to the rest of the domain.

The comparison of pressure between the two types of boundary conditions imposed is extended to two more heights of the channel, i.e. $y = 0.68$ and $y = 1.0$, as represented in figures 50 and 51.

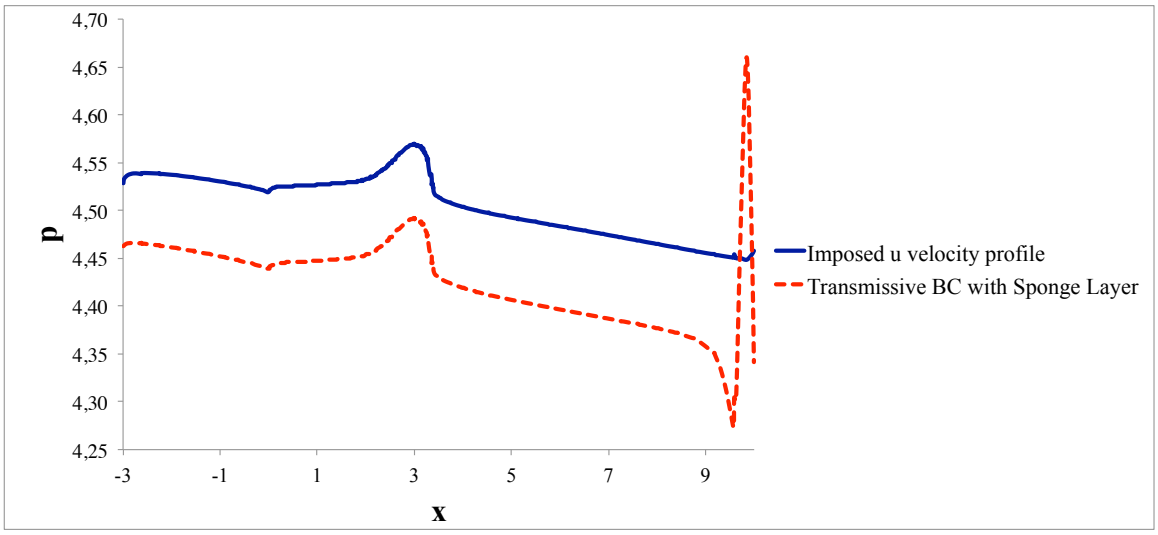


Figure 50: Comparison of pressure along the whole test section between imposed u BC and transmissive BC with sponge layer for $Re = 389$, at a height $y = 0.68$, and in the middle section.

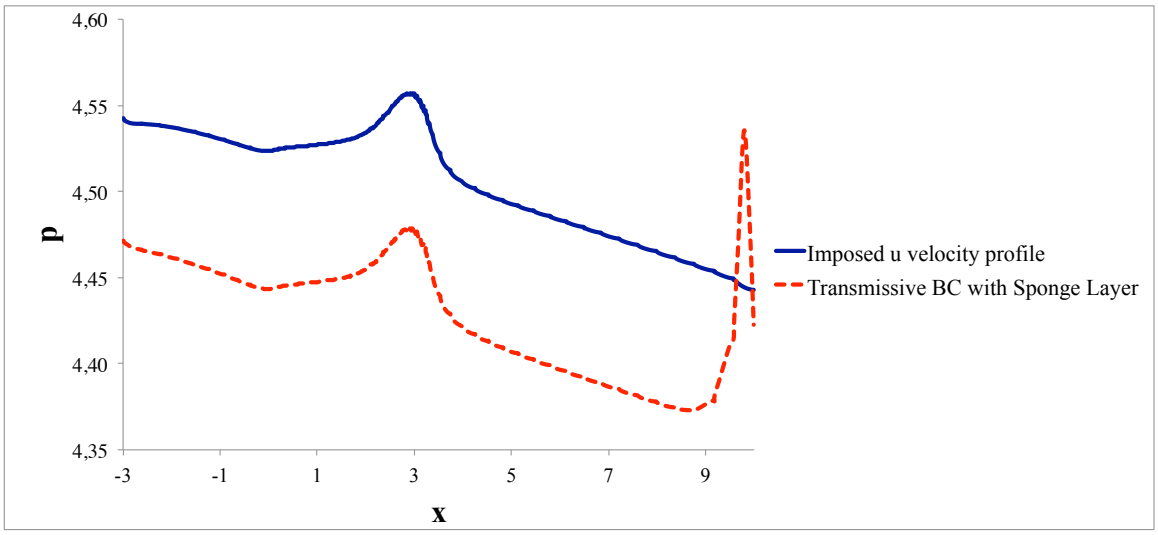


Figure 51: Comparison of pressure along the whole test section between imposed u BC and transmissive BC with sponge layer for $Re = 389$, at a height $y = 1.0$, and in the middle section.

As shown in figures 50 and 51, good agreement is achieved for pressure distribution along the whole test section at different heights in the channel, with a difference between the two types of boundary conditions remaining in the range of 2% of the pressure value, except for the last part of the outlet channel, where the sponge layer is situated. In this region, in fact, oscillations do not give accurate results in predicting the pressure value.

5.3.3 Conclusions

The simulations over the backward-facing step with the ramp have shown consistency with what theoretically expected for the different Reynolds numbers adopted.

Increasing the Reynolds number, in fact, the u velocity clearly suggests that the recirculation phenomenon is emphasized. Comparing the flow structure in the cavity within the range of

Reynolds number chosen, it is possible to notice that the centre of the main recirculation zone moves towards the ramp as Re increases. Increasing it further, i.e. for Reynolds such that $200 < Re < 400$, the centre of the main recirculation zone moves further downstream, so that it follows the ramp geometry. This phenomenon is also proven with the analysis of the reattachment point y_R as a function of the Reynolds number adopted.

Results for transmissive boundary conditions have initially been in great disaccordance with those obtained imposing the u velocity profile at the outlet section. This disaccordance has been attributed to the numerical errors transmissive boundary conditions encounter near the outlet section, that cause huge instability in this section of the domain. Such errors, during the simulations, have propagated throughout the whole test section, leading to results in disaccordance with those previously obtained. To reduce and fix this instability a 'sponge layer' has been implemented, whose aims are to prevent the outflow boundary condition to become unstable and to block the propagation of the error generated to the rest of the domain. This sponge region is based on the model proposed by Guo *et al.*[36] and adopted also by Adams[37], which considers the addition of a vector to Navier-Stokes equations to limit oscillations near the outlet section.

Simulations have been performed also with the application of this sponge layer, and results obtained have shown good agreement with those associated with imposed u boundary condition. The zone where the sponge layer is applied still presents oscillations, but the in the rest of the domain results reproduce well those previously generated.

The distribution of pressure too is consistent with the expectations. When the Reynolds number increases, it is possible to notice a steeper gradient in pressure at the ramp corner, which is concentrated into a smaller zone. Results before the application of the sponge layer are once again affected by the propagation of numerical errors from the outlet section. After the sponge layer has been imposed in the last domain, it has been possible to identify good agreement with the results obtained for the u velocity profile imposed boundary condition. The zone where the sponge layer is applied does not provide accurate results for pressure, but the damping it performs makes it possible to achieve accurate predictions in the rest of the domain.

Although the results obtained throughout the domain are accurate, they still present some oscillations towards the outlet section. It is possible to try to reduce these inaccuracies increasing the length of the sponge layer, or modifying the two parameters involved in the formulation of the sponge layer, i.e. A_s and N_s. With such modifications, the sponge layer could be tuned more appropriately to the case considered, and therefore results obtained, even in the regions closer to the outlet section, would be more accurate.

5.4 Sponge Layer Tuning

In order to obtain consistent results using transmissive outflow boundary condition it has been necessary to implement a sponge layer, responsible of damping down instabilities in the vicinity of the outlet section.The formulation of the damping function and the sponge layer characteristics have been described in section (5.3.2.2).

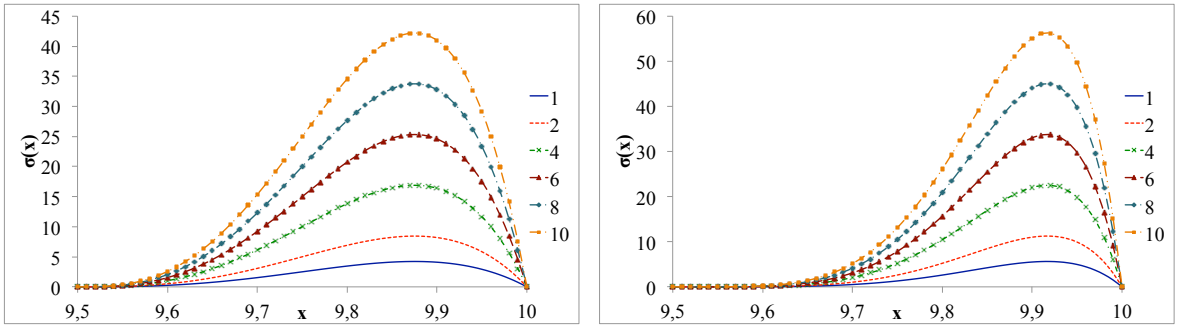
This section's purpose is to tune appropriately the damping function, in order to reduce instabilities even further, and obtain results even more accurate. With the previous choice of

parameters, i.e. $A_s = 4$ and $N_s = 3$, both of which coincides with the typical choice in the literature[36], instabilities at the outlet have been reduced, but not eliminated. This section aims to fix them as much as possible, considering different values for these two parameters, and even considering a possible extension of the sponge layer to a wider domain.

5.4.1 Regulation of Parameters

First of all, it is necessary to understand the influence of A_s and N_s on the damping function. In order to do that, while one of the two parameters is kept fixed, the other is varied, and values assumed by the damping function in the region where the sponge layer is applied are plotted.

In figure 52, N_s has been assumed equal to 3 (figure 52(a)) and equal to 5 (figure 52(b)), while A_s is varied in the range $1 < A_s < 10$, and its value is indicated in the legend. It

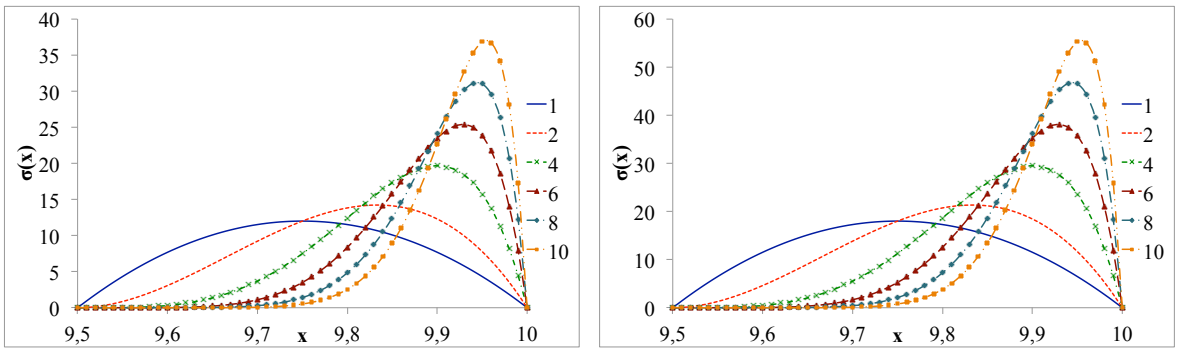


(a) $N_s = 3$. (b) $N_s = 5$.

Figure 52: Variation of the damping function keeping N_s fixed, and varying A_s in the range $1 < A_s < 10$.

is possible to notice how increasing A_s, the maximum of the damping function increases, but remaining on the same x coordinate. The curve in figure 52(a) corresponding to $A_s = 4$ is the damping function that has been used in sections (5.3.2.2) and (5.3.2.3) for the analysis of the backward-facing step with ramp flow. Comparing figures 52(a) and 52(b), it is evident that the maximum damping value achieved by $\sigma(x)$ increases both with A_s and N_s, since curves in figure 52(b) have a higher maximum, if compared with their respective ones in figure 52(a).

An analogous procedure can be applied for the analysis of the effect of the variation of N_s on the damping function values. In this case, A_s is assumed first equal to 4 (figure 53(a)), and then equal to 6 (figure 53(b)). It is evident that increasing N_s and keeping A_s constant,



(a) $A_s = 4$. (b) $A_s = 6$.

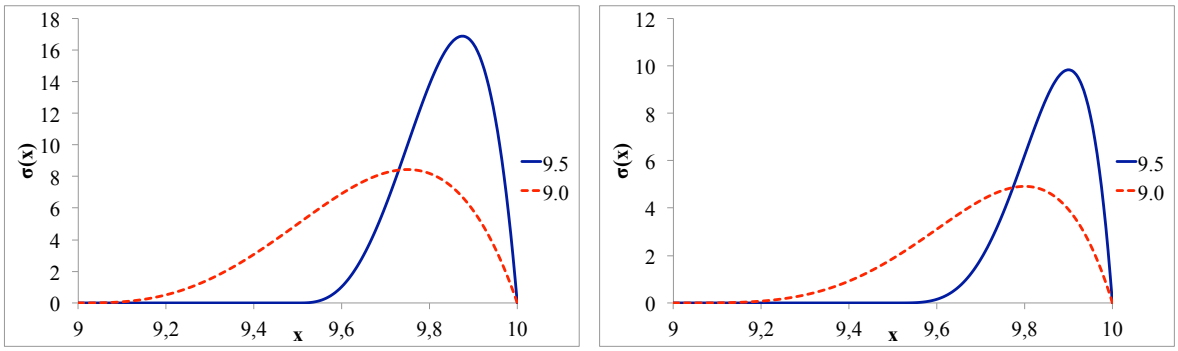
Figure 53: Variation of the damping function keeping A_s fixed, and varying N_s in the range $1 < N_s < 10$.

the maximum of the curve tends to move towards the right endpoint of the interval where the sponge layer is applied. In addition, the maximum does not keep a constant value as N_s increases, but it increases as well, since the curve keeps the same subtended area. As already underlined for figure 52, for a same A_s the curve with the higher N_s presents a higher maximum, situated in correspondance of the same x coordinate.

5.4.2 Extension of the Sponge Layer

Another possibility of optimization is represented by the extension of the zone where the sponge layer is applied. For the simulations performed in sections (5.3.2.2) and (5.3.2.3), the sponge layer has been applied just in the last domain before the outlet section. As explained by Guo *et al.*[36], in fact, the sponge layer usually is assumed as thick as one boundary layer.

This section examines the effect of extending the sponge layer also to a previous subdomain for the case of the backward-facing step with the ramp, so that x_s is assumed equal to 9.0. The effects of this extension can be seen in figure 54, where the damping function has been plotted with reference to the previous sponge layer extension (i.e. $x_s = 9.5$), for two different couples A_s, N_s. The legend indicates the x_s the curve corresponds to. As it emerges from figure 54, a wider region to apply the sponge layer lowers its damping effect, for the same choice of A_s and N_s. In particular, having doubled the region where the sponge layer is imposed, the maximum value of the damping function assume a value that is close to half of that it assumed for a smaller region. In addition, enlarging the operating zone of the sponge layer, the maximum damping is situated in correspondance of a x coordinate closer to the center of the interval.



(a) $A_s = 4$, $N_s = 3$.

(b) $A_s = 2$, $N_s = 4$.

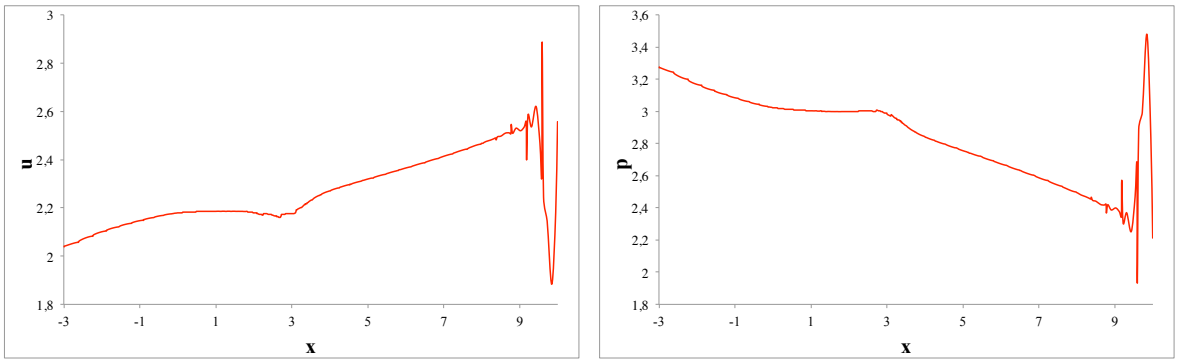
Figure 54: Variation of the damping function increasing the sponge layer extension, keeping A_s and N_s fixed.

Now that all the possible parameters to regulate the sponge layer have been presented, and their different effects analyzed, it is possible to proceed with their optimization for the case of the backward-facing step with the ramp.

5.4.3 Optimization

This section is dedicated to different choices of A_s, N_s and x_s in order to achieve the best possible results for the case of the backward-facing step with the ramp. As it can be seen from figure 55, the greatest oscillations are achieved at a distance of approximately 12.5 non-dimensional units from the inlet section, i.e. at $x = 9.5$. It is exactly at this point that the sponge layer is wanted to damp oscillations as much as possible, so the tuning of parameters is performed to have a maximum damping in the vicinity of $x = 9.5$.

In particular, three choices for the three parameters involved have been considered convenient:



(a) u velocity. (b) Pressure.

Figure 55: Greatest oscillations for the case of backward facing step with ramp with transmissive boundary conditions, $Re = 389$, in the middle section along z axis, and in the middle height of the channel.

1. $x_s = 9.0$, $A_s = 4$, $N_s = 1$;

2. $x_s = 9.0$, $A_s = 4$, $N_s = 3$;

3. $x_s = 9.0$, $A_s = 6$, $N_s = 3$.

All of the three cases considered involve a sponge layer zone that is double as thick as before, since $x_s = 9.0$. The damping functions for these cases are shown in figure 56, where the legend indicates A_s, N_s for each curve. The curve represented in black indicates the damping function used previously, when just the last domain has been reserved for the sponge layer, with $x_s = 9.5$, $A_s = 4$ and $N_s = 3$.

Simulations have been run with the same geometry and boundary conditions specified in section (5.3), with the implementation of the sponge layer responding to the curves just shown in figure 56. In figure 57 is represented a graphical comparison for u velocity along the whole

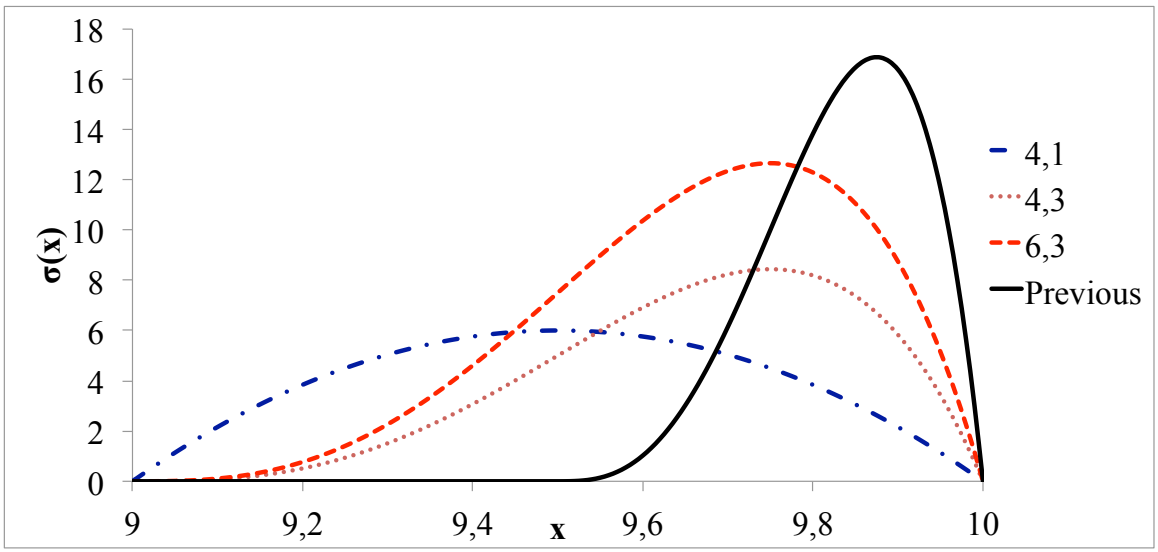


Figure 56: Comparison of the damping functions $\sigma(x)$ for the different cases considered.

test section, between u velocity profile specified boundary condition, transmissive boundary condition with sponge layer described in section (5.3.2.2), and transmissive boundary condition with sponge layer described by the four cases just presented. Figure 58, instead, present an analogous comparison for the pressure.

The case $A_s = 4$, $N_s = 1$, represented in the figures with a red line, appears as the most convenient choice for tuning the boundary layer. With this choice, in fact, the damping effect of the sponge layer has its higher effect, although the region where the values obtained are not physical is broadened. A different result is obtained choosing $N_s = 3$, where values for u velocity and pressure are almost independent of the A_s chosen. The maximum damping effect,

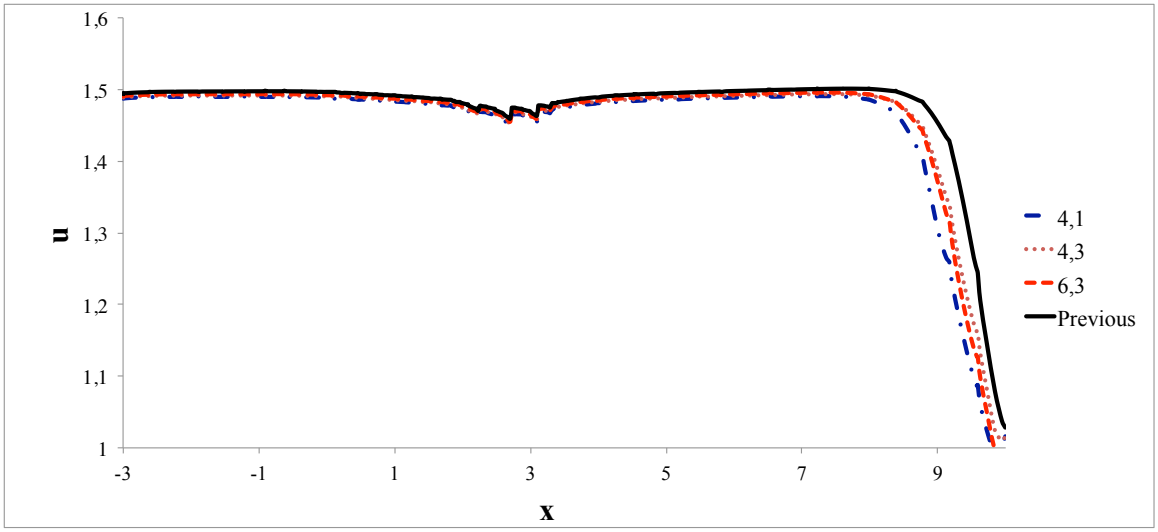


Figure 57: Comparison of u velocity along the whole test section for transmissive BC with sponge layer, with different parameters, for $Re = 389$, at a height $y = 1.65$, and in the middle section.

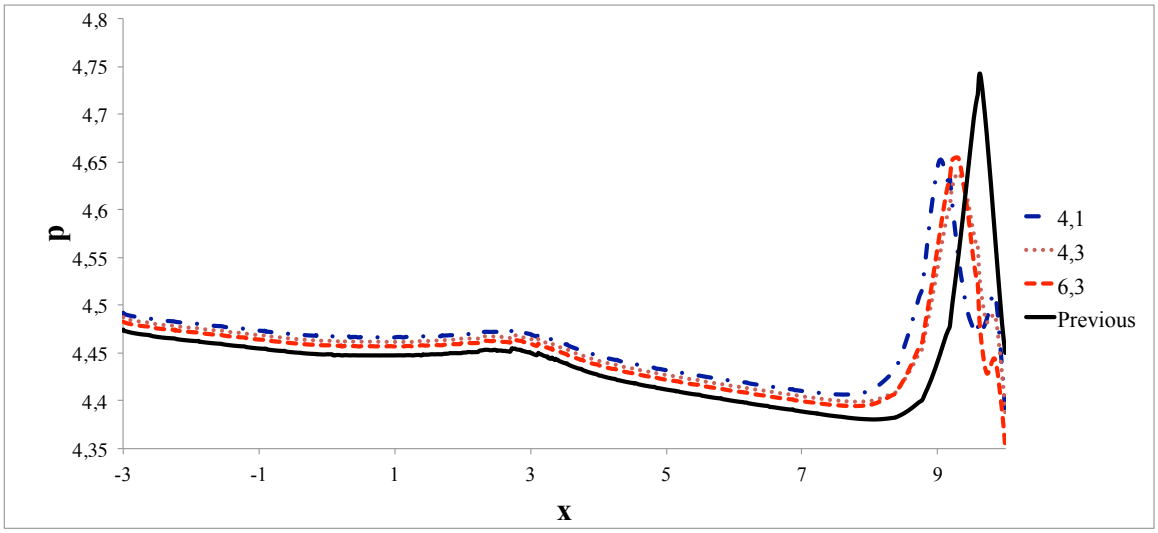


Figure 58: Comparison of pressure value along the whole test section for transmissive BC with sponge layer, with different parameters, for $Re = 389$, at a height $y = 1.65$, and in the middle section.

in fact, is situated at a same x coordinate, as suggested in figure 56, so that u and p have a similar trend in both cases, with respect of their x coordinate.

If the sponge layer had been restricted to just the last domain, as represented by the black line in figures 57 and 58, a higher damping would have been imposed in a section closer to the end of the domain. Such damping leads to a slight underestimation of u and p values, when compared to the other cases considered, but causes a smaller zone where the results obtained are not physical, because of the influence of numerical errors at the outlet section. These errors, in fact, could have not been completely eliminated, but they have been contained in order to achieve accurate results in the rest of the test section.

CHAPTER 6

CONCLUSIONS

This chapter summarizes all the results achieved in this thesis. It has been considered suitable to divide the chapter in four parts. The first section is dedicated to the mesh generator, the second part discusses mesh and code validation with comparison between simulations and literature references, the third part groups results obtained for the backward-facing step with ramp geometry, and the last section is reserved for suggestions on future work.

6.1 Mesh Generator

One of the main achievements of this thesis is the implementation of a mesh generator to be used with the computational code for simulations.

The mesh generator is based on the first version implemented by J. Komperda [25] at CMTL at UIC, that was able to generate squares, rectangles, and shapes obtained by their combination, with a uniform grid. When combining different shapes, however, the basic program would have had to recalculate all connections between points, constituting a loss in terms of computational time.

With further implementation, the mesh generator has been made able to deal with any two-dimensional convex quadrilateral, where the word 'convex' is here used to indicate quadrilaterals where each of the interior corners is in the range $0° < \theta < 180°$. All routines have been modified and the possibility to create a uniform or a non-uniform grid has been implemented.

In the last case, power distribution and symmetric power distribution have been programmed. This improvements have made possible to obtain a grid refinement where the flow structure is more complex, and a wider grid where it is more simple. As a results of that, the usage of the computational power available is optimized, in the sense that a finer grid uses more computational resources, in order to describe accurately a complex flow structure, while a wider grid allows the allocation of lower computational power for regions where the flow structure is simpler.

The specification of boundary conditions has also been integrated into the mesh generator. This method of specifying conditions on shapes' boundaries has been adopted because of its straightforwardness. For each two-dimensional shape, in fact, four numbers in sequence indicate what type of boundary is each side faced to, starting from the bottom side and proceeding counter-clockwise. Such a simple approach has been chosen because particularly suitable for the combination of different shapes.

Besides the introduction of new features in the mesh generator, existing routines have been optimized as well. In particular, the algorithm used to calculate connections for a shape resulting from additions of different shapes has also been modified, so that information are stored and just updated at every shape combination, rather than recalculated each time. This improvement has permitted an important saving of computational time during the usage of the mesh generator.

The two-dimensional mesh generator has been extended to a third dimension, where the two-dimensional shape generated previously is extruded into the z direction. In this direction too, it has been implemented the possibility to create a uniform or a non-uniform grid.

To ensure compatibility between the mesh generator and the numerical code, a final routine has been implemented in order to store information (i.e grid points, connections and boundary conditions) in the correct format.

6.2 Backward-Facing Step

In order to provide validation of the numerical code used, and to test the good operation of the .mesh file created by the mesh generator, simulations' results over a backward-facing step have been compared with experimental results found in literature [3], using the same setup.

Two types of outflow boundary condition are imposed, for a range of Reynolds numbers $100 < Re < 400$, and a Mach number equal to 0.4. A first set of simulations is performed specifying outflow profiles for various variables at the outlet section, feature that was already included in the numerical code adpoted. Another set of simulation has been run using transmissive boundary conditions at the outlet, implemented specifically with this thesis.

Predictions for reattachment length have shown good agreement with experimental data taken from Armaly *et al.* [3], and with those from Denham and Patrick [9]. For the range of Reynolds numbers considered, in fact, it is possible to notice a linear dependence between the reattachment length and the Reynolds number. Simulations with transmissive boundary conditions generate results closer to those found in literature.

Profiles for u velocity at different cross-sections along the test section length have been plotted and compared to experimental data from Armaly *et al.* for $Re = 100$ and $Re = 389$. In both cases, good agreement is achieved, sign that the development of the flow predicted by the code is consistent with the references.

Pressure distribution is analogous for both types of boundary condition adopted. With the specified u velocity profile, however, a higher value of pressure along the test section is observed. Transmissive boundary conditions, on the other side, the value of p predicted is lower in the whole test section, and this difference is associated with the numerical issues transmissive boundary conditions generate in the proximity of the outlet section.

In summary, results obtained with simulations are consistent with experimental data taken from Ref.[3] and[9], especially when a u velocity profile has been imposed as outflow boundary condition, so that the code used is validated.

6.3 Backward-Facing Step with Ramp

Having validated the code with the geometry of the backward-facing step, such a geometry is modified with the addition of a ramp after the step, in order to reproduce the geometry of a dump combustor. Geometrical dimensions are based on those taken from Tuttle *et al.*[6], but a shorter outlet channel is chosen. In addition, no inclination is imposed to the lower wall, so that the outlet channel results with the same height as the inlet channel.

The geometry adopted has been widely analyzed in the literature, because of its application in the scramjet engine. Despite that, most of the experiments and simulations have focused just on supersonic flows, with little or no interest in the analysis of subsonic laminar flows in such a geometry. This work, therefore, has aimed to model a subsonic laminar flow in a dump combustor, in a range of Reynolds numbers $100 < Re < 400$, and with a Mach number equal to 0.4.

In this case too simulations have been performed using the two types of boundary conditions applied previously. When the u velocity profile has been specified at the outlet section, because of the same geometrical characteristics of inlet and outlet channels, the same profile has been specified at both inlet and outlet.

Comparison of reattachment points for increasing Reynolds numbers shows, for this geometry as well, a slightly higher prediction made by transmissive boundary conditions. With both types of boundary condition considered the difference obtained between the values of the reattachment point is in the order of approximately 1-2% of the value predicted. For this reason, it is possible to ignore the differences between the values predicted, and assert that the simulations show good agreement, independently of the outflow boundary condition imposed.

The flow structure in the cavity has been studied for increasing Reynolds numbers, showing how the recirculation phenomenon is emphasized as Re increases. Furthermore, this recirculation zone tends to be centered more and more downstream, i.e. closer to the ramp, when the Reynolds number raises. For a Re sufficiently high, the main recirculation phenomenon is centered on the ramp, rather than on the cavity bottom wall.

In this case, transmissive boundary conditions have not predicted accurate results. Numerical insabilities at the outlet section, associated with the process of projecting flux values to the mortars, have been responsible of a propagation of errors that has involved all the rest of the test section, leading to non-physical results for all the variables considered. Therefore, it has been necessary to implement a way to keep solution values near the outlet section within a certain range. A sponge layer has been programmed, responsible to damp down oscillations in

the vicinity of the outlet section, with a formulation analogous to that introduced by Guo *et al.*[36]. With the specification of a damping function, in fact, oscillations are damped down in a region in the vicinity of the outlet section, with a higher damping as greater is the difference between the conserved quantity measured in the point and the reference conserved quantity imposed.

With the application of the sponge layer in the last domain before the outlet section, results obtained are in very good agreement with those achieved with u velocity profile specified boundary condition. The area of the domain closer to the outlet section, instead, has been still affected by oscillations, that couldn't have been completely eliminated.

The pressure field in the test section shows a steeper variation towards the ramp corner for higher Reynolds numbers, while for lower Re the change in pressure in this zone becomes less sharp. The comparison between the results obtained with the two types of outflow boundary condition gives a good agreement after the sponge layer has been applied. In this case too, oscillations near the outlet section have been damped, but not completely eliminated. In the rest of the test section, however, an accurate distribution of pressure is achieved.

In addition, it has been tried to optimize the buffer domain for the higher Reynolds number used. The effects of variation of the parameters involved in the definition of the damping function have been studied, together with the possibilty to extend the sponge layer to a wider domain, and everything has been tuned to achieve the configurations considered most convenient for the case in exam. Even with this tuning, however, it has not been possible to eliminate completely oscillations near the outlet section, but they have been reduced further.

6.4 Future Work

Future extensions of this work can be made regarding either the mesh generator or the code. With regards to the mesh generator, other ways of distributing points along the shapes sides may be implemented, and a Graphical User Interface (GUI) may be written to make the program more user-friendly. A further implementation of the mesh generator, also, could be its extension to curved geometries, and to three-dimensional elements with variable depth. With regards to the code, more testing and development is needed on the transmissive boundary condition. With the implementation of the sponge layer, results obtained with this type of boundary conditions have shown good agreement in the majority of the test section, but oscillations at the exit are still present. Such oscillations need to be investigated and reduced further, so that the consistency of results is ensured throughout the whole test section.

CITED LITERATURE

1. Emoscopes, Wolfkeeper, and Cryonic07: Ramjet operation. Retrieved April 12, 2013, from http://en.wikipedia.org/wiki/Ramjet.
2. Emoscopes and Luke490: Scramjet operation. Retrieved April 12, 2013, from http://en.wikipedia.org/wiki/Scramjet.
3. Armaly, B. F., Durst, F., Pereira, J. C. F., and Schoenung, B.: Experimental and theoretical investigation of backward-facing step flow. J. Fluid Mech., 127:473–496, 1983.
4. Ben-Yakar, A. and Hanson, R. K.: Cavity Flame-Holders for Ignition and Flame Stabilization in Scramjets: An Overview. Journal of Propulsion and Power, 17(4):169–877, 2001.
5. Sivasegaram, S. and Whitelaw, J. H.: Combustion Oscillations in a Dump Combustor with a Constricted Exit. J. Mech. Eng. Sc., 202:205–210, 1988.
6. Tuttle, S. G., Carter, C. D., and Hsu, K. Y.: Particle Image Velocimetry in an Isothermal and Exothermic High-Speed Cavity. 50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, AIAA 2012-0330, Nashville, Tennessee, 09 - 12 January 2012.
7. Sengupta, K., Russell, K., and Mashayek, F.: Step Geometry and Countercurrent Effects in Dump Combustor, Part 1: Cold Flow. AIAA Journal, 45:2033, 2007.
8. Mashayek, F., Sengupta, K., and Russell, K.: Step Geometry and Countercurrent Effects in Dump Combustor: Reacting Flow. Numerical Heat Transfer, Part A: Applications, 54:569–580, 2008.
9. Denham, M. K. and Patrick, M. A.: Laminar flow over a downstream-facing step in a two-dimensional flow channel. Trans. Inst. Chem. Engrs., 52:361, 1974.
10. Le, H., Moin, P., and Kim, J.: Direct numerical simulation of turbulent flow over a backward-facing step. J. Fluid Mech., 330:349–374, 1997.
11. Lee, T. and Mateescu, D.: Experimental and numerical investigation of 2-D backward-facing step flow. Journal of Fluids and Structures, 12:703–716, 1998.
12. Lima, R. C., Andrade, C. R., and Zaparoli, E. R.: Numerical study of three recirculation zones in the unilateral sudden expansion flow. International Communications in Heat and Mass Transfer, 35(9):1053–1060, 2008.
13. Williams, P. T. and Baker, A. J.: Numerical Simulations of a Laminar Flow over a 3D Backward-Facing Step. Int. J. Numer. Meth. Fluids, 24:1159–1183, 1997.
14. Biswas, G., Breuer, M., and Durst, F.: Backward-Facing Step Flows for Various Expansion Ratios at Low and Moderate Reynolds Numbers. J. Fluids Eng., 126:362–374, 2004.

CITED LITERATURE (continued)

15. Park, S., Wachsman, A., Yi, T., Wee, D., Annaswamy, A. M., and Ghoniem, A. F.: A Backward-facing Step Combustor: Reduced-Order Modeling and Control. Proceedings of the 42nd IEEE, Conference on Decision and Control:2328–2333, Maui, Hawaii USA, December 2003.
16. Schroeder, A., Schanz, D., Heine, B., and Dierksheide, U.: Investigation of Transitional Flow Structures Downstream of a Backward-Facing-Step by Using 2D-2C- and High Resolution 3D-3C- Tomo- PIV. In New Results in Numerical and Experimental Fluid Mechanics VIII, volume 121, pages 219–226. Springer, 2013.
17. Chen, Z., Yi, S., He, L., Tian, L., and Zhu, Y.: An experimental study on fine structures of supersonic laminar/turbulent flow over a backward-facing step based on NPLS. Chinese Science Bulletin, 57(6):584–590, 2012.
18. Kanchi, H., Sengupta, K., and Mashayek, F.: Effect of turbulent inflow boundary condition in LES of flow over a backward-facing step using spectral element method. International Journal of Heat and Mass Transfer, 62:782 – 793, 2013.
19. Nguyen, V.-T., Nguyen, H. H., Price, M. A., and Tan, J. K.: Shock capturing schemes with local mesh adaptation for high speed compressible flows on three dimensional unstructured grids. Computers and Fluids, 70:126 – 135, 2012.
20. Kurganov, A. and Liu, Y.: New adaptive artificial viscosity method for hyperbolic systems of conservation laws. J. Comp. Phys., 231:8114 – 8132, 2012.
21. Kopriva, D. A. and Kolias, J. H.: A conservative staggered-grid Chebyshev multidomain method for compressible flows. J. Comp. Phys., (125):244–261, 1996.
22. Kopriva, D. A.: A conservative staggered-grid Chebyshev multidomain method for compressible flows. II. A semi-structured method. J. Comp. Phys., (128):475–488, 1996.
23. Kopriva, D. A.: A staggered-grid multidomain spectral method for the compressible Navier-Stokes equations. J. Comp. Phys., (244):142–158, 1998.
24. Jacobs, G. B.: Numerical simulation of two-phase turbulent compressible flows with a multidomain spectral method. Chicago, Illinois, 2003.
25. Komperda, J.: SimpleMAC. Retrieved February 13, 2013, from https://github.com/jonkomperda/SimpleMAC.
26. Anderson, J. D.: Computational Fluid Dynamics. The Basics with Applications.. McGraw-Hill, 1995.
27. Gordon, W. J. and Hall, C. A.: Construction of curvilinear co-ordinate systems and applications to mesh generation. Int. J. Num. Meth. Eng., (7):461–477, 1973.
28. Fletcher, C. A. J.: Computational Techniques for Fluid Dynamics 1. Fundamental and General Techniques. Springer, 1988.
29. Sengupta, K.: Direct and Large-eddy Simulation of Compressible Flows with Spectral/hp Element Methods. Chicago, Illinois, 2009.

CITED LITERATURE (continued)

30. Roe, P. L.: Approximate Riemann Solvers, parameter vectors, and difference schemes. J. Comp. Phys., (43):357–372, 1981.
31. Williamson, J.: Low-storage Runge-Kutta schemes. J. Comp. Phys., (35):48–56, 1980.
32. Carpenter, M. H. and Kennedy, C. A.: A fourth order 2N-storage Runge-Kutta scheme. NASA TM 109112, June 1994.
33. Lomax, H., Pulliam, T. H., and Zingg, D. W.: Fundamentals of Computational Fluid Dynamics. Springer, 2004.
34. Wendt, J.: Computational Fluid Dynamics - An Introduction. Springer, 2009.
35. Pruett, C. D., Zang, T. A., Chang, C. L., and Carpenter, M. H.: Spatial Direct Numerical Simulation of High-Speed Boundary-Layer Flows. Part I: Algorithmic Considerations and Validation. Theoret. Comput. Fluid Dynamics, 7:49–76, 1993.
36. Guo, Y., Adams, N. A., and Kleiser, L.: A comparison study of an improved temporal DNS and spatial DNS of compressible boundary layer transition. AIAA J., 34:683–690, 1996.
37. Adams, N. A.: Direct Numerical Simulation of Turbulent Compression Ramp Flow. Theoret. Comput. Fluid Dynamics, 12:109–129, 1998.
38. Nieuwstadt, F. T. M.: Turbulentie, inleiding in de theorie en toepassing van turbulente stromingen. Utrecht, The Netherlands, 1992.
39. Ferziger, J. H. and Peric, M.: Computational Methods for Fluid Dynamics - 3rd Edition. Springer, 2001.
40. Tannehill, J. C., Anderson, D. A., and Pletcher, R. H.: Computational Fluid Mechanics and Heat Transfer - Second Edition. Taylor and Francis, 1997.
41. Fletcher, C. A. J.: Computational Techniques for Fluid Dynamics 2. Specific Techniques for Different Flow Categories. Springer, 1988.
42. Brusa, E., Delprete, C., and Gay, P.: Tutorato di Calcolo Numerico. Consorzio Nettuno, Politecnico di Torino, 2001.

VITA

NAME:	Federico Lugnani
EDUCATION:	M.Sc., Mechanical Engineering, Politecnico di Torino, Italy, 2013
	M.Sc., Mechanical Engineering, University of Illinois at Chicago, 2013
	B.Sc., Mechanical Engineering, Politecnico di Torino, Italy, 2011
HONORS:	'Roberto Rocca Education Program' scholarship, 2012
	TOP-UIC scholarship, 2012