UNIVERSIDADE DA BEIRA INTERIOR
Engenharia

Hydrodynamic Study of a Lander Platform for Subsea Operations

(Versão final após defesa)

José Jerónimo da Silva Gonçalves

Dissertação para obtenção do Grau de Mestre em
Engenharia Aeronáutica
(Ciclo de estudos integrado)

Orientador: Prof. Doutor Miguel Ângelo Rodrigues Silvestre
Co-orientador: Engenheiro Tiago Correia Bartolomeu

Covilhã, dezembro de 2019

Acknowledgments

To all those who have accompanied me on this long journey in search of knowledge and wisdom, I express my most sincere gratitude. From my childhood educators to my university professors, I am sincerely grateful for all that they have taught me.

Firstly, I would like to thank my supervisor at UBI (Universidade da Beira Interior), Professor Miguel Silvestre for the time spent helping me, for the knowledge he has given me and for being always present, without him this dissertation would not be possible. I am genuinely grateful to CEiiA (Center of Engineering and Product Development) for the opportunity they gave me and all the support and conditions provided.

I am particularly grateful to my supervisor at CEiiA, Tiago Bartolomeu. You have consistently supported me and raised my head on the most difficult days. Thanks not only for your help, but also for your advice and friendly word. I couldn't have had a better mentor. Further, this gratitude also includes CEiiA's team: Marcos Moreira, Miguel Machado and Oleh Tkachuk.

I could not forget those who in these last months worked alongside me. André, Mariana, and especially, Daniel for the support and contagious positive energy they gave me. Moreover, thanks to all my friends who accompanied me and somehow contributed: Filipe, Jeremias, Juan, Tiago, Marcos, Pedro, Luis, António, Sílvia, José, Daniel, Maria, Jorge, Francisca, David, Martim, Inês, Filipa, Rui, Gonçalo, Gustavo, Monte, Francisco and Flávio.

Finally, I thank my family and my girlfriend for all the patience, courage and unconditional love they gave me. To my parents, Maria and Jerónimo, I am eternally grateful that they are always there, letting me follow my dreams even though it may sometimes be painful to get away from them. To my brothers, Fátima, Mara, Sara, Laura and Francisco, thank you for everything. And to my girlfriend, Maria João, nothing I can say will express what you've done for me these months. You're the best!

Dedication

Dedication to my uncles João and Manuel, who were like a father to me.

Resumo

O estudo da sismografia no fundo dos oceanos é de extrema importância para prevenção de tsunamis. Estima-se que cerca de 500000 sismos são detetados anualmente, sendo que na maioria dos casos o epicentro está localizado nos oceanos. Deste modo, a sua deteção precoce seria uma mais valia para evitar, quer perdas humanas, quer perdas materiais. Atualmente utilizam-se sismómetros no fundo dos oceanos (OBS) como instrumentos de deteção sísmica. O preço destes instrumentos influencia bastante a sua precisão, o que poderá fazer com que sejam detetados diferentes intervalos de frequência. Um dos problemas que estes aparelhos apresentam é o ruído gerado pelo escoamento. Ruído este que, dependendo da sua frequência pode corromper o sinal transmitido. Por isso, é crucial estudar, hidrodinamicamente, a geometria que menos ruído gera, a fim de evitar alarmes indesejados.

O presente trabalho tem como principal objetivo analisar, através do software OpenFOAM (Open-source Field Operation And Manipulation), o comportamento do escoamento em torno de um cilindro, de forma a poder identificar, filtrar e/ou eliminar as frequências indesejadas induzidas pelo escoamento. A velocidade máxima considerada para este estudo é 0.08 [m/s], que aliada às características do fluido e da geometria que se pretende estudar, resultou num número de Reynolds de 80000. Para este estudo, foram utilizadas as equações de Unsteady Reynolds-Averaged Navier-Stokes (URANS) com o modelo de turbulência κ-ω Shear Stress Transport (SST).

Foram analisadas diferentes razões de aspeto do OBS para se compreender qual era a variação da frequência induzida pelo escoamento. Foi observado que a frequência induzida pela escoamento decresce com a diminuição da razão de aspeto da geometria. Em seguida, analisou-se a configuração conceptual utilizando a escala 1:1, da qual se obteve uma frequência máxima de 0.0057 [Hz]. Com o objetivo de minimizar o ruido gerado pelo escoamento, foram sugeridas duas geometrias alternativas: um cilindro com uma calota esférica no topo e um cilindro com três cintas espirais. No final, foi conseguido reduzir em 60% a frequência induzida pelo escoamento, com o cilindro com uma calota esférica no topo.

Com a modificação sugerida, é possivel obter resultados que não comprometem o correto funcionamento do sismógrafo. Na impossibilidade de eliminar por completo o ruído, pode ser desenvolvido um filtro eletrónico de forma a que estas frequências indesejadas sejam identificadas e excluídas da amostra.

Palavras-chave

Sismómetros de fundo do oceano, OpenFOAM, frequência, Dinâmica de Fluidos Computacional (CFD), carenagem.

Abstract

The seismography study in the oceans is extremely important for tsunami prevention. It is estimated that annually are detected over 500000 and, in most cases, the epicenter is at the ocean. Thus, its premature detection would be an asset to avoid both human and material losses. Nowadays, Ocean Bottom Seismometers (OBS) are used as instruments of seismic detection. The price of these instruments greatly influences their accuracy, which may cause different frequency ranges to be detected. One of the main issues of these instruments is the noise generated by the flow. This noise, depending on its frequency, can corrupt the signal transmitted. For this reason, it is crucial to study, hydrodynamically, the geometry that generates less noise, in order to avoid undesired alarms.

This project has the objective of analysing, through OpenFOAM (Open-source Field Operation And Manipulation) software, the flow behavior around a cylinder, to identify, filter and/or eliminate the undesired frequencies induced by the flow. The maximum velocity considered for this study is 0.08 [m/s], which combined to the flow characteristics and the geometry intended to study, resulted in a Reynolds number of 80000. For this study, the equations utilized were Unsteady Reynolds-Averaged Navier-Stokes (URANS) with the turbulence model κ-ω Shear Stress Transport (SST).

In order to understand which was the flow-induced frequency variation, different OBS aspect ratios were analysed. It was observed that the induced frequency would decrease with the reduction of the aspect ratio. Following this, it was analysed with a scale $1{:}1$ the conceptual configuration, which the result indicated a maximum frequency of 0.0057 [Hz]. Therefore, in order to minimize the noise generated by the flow, two alternative geometries were suggested: a cylinder with a spherical hubcap and a cylinder with three helical strakes. In the end, it was possible to achieve a 60% reduction in the induced frequency, with the cylinder with a spherical hubcap configuration.

Along with the suggested modification, it is possible to obtain results that do not compromise the correct functioning of the seismograph. Failing to eliminate completely the noise, an electronic filter should be developed, so that these unwanted frequencies could be identified and excluded from the sample.

Keywords

Ocean Bottom Seismometer, OpenFOAM, frequency, Computational Fluids Dynamics (CFD), shell configuration.

Contents

List of Figures

List of Tables

Acronyms List

ASV	Autonomous Surface Vehicle
AUV	Autonomous Underwater Vehicle
BOBO	BOttom BOundary
CEiiA	Center of Engineering and Product Development
CFD	Computational Fluid Dynamics
CFL	Courant-Friedrichs-Lewyor
DNS	Direct Numerical Solution
DOBO	Deep Ocean Benthic Observatory
DoF	Degrees of Freedom
EMEPC	Estrutura de Missão para a Extensão da Plataforma Continental
FFT	Fast Fourier Transform
FVM	Finite Volume Method
IDL	Instituto Dom Luiz
ISIT	Intensified Silicon Intensifier Target
LES	Large Eddy Simulations
OBJ	Wavefront Object
OBS	Ocean Bottom Seismometer
OpenFOAM	Open-source Field Operation And Manipulation
RANS	Reynolds Averaged Navier-Stokes
ROV	Remotely Operated Vehicle
SST	Shear-Stress Transport
STL	Stereolithography
TWERC	Tsunami Warning and Early Response system of Cyprus
UBI	Universidade da Beira Interior
URANS	Unsteady Reynolds Averaged Navier-Stokes
US	United States
UUV	Unmanned Underwater Vehicle
V&V	Verification and Validation

Nomenclature

Δy	First layer thickness	$[m]$
∇T	Temperature gradient	$[K/s]$
$\overline{C_D}$	Mean drag coefficient	$[-]$
$\overline{D}$	Mean drag	$[N]$
$\widehat{D}$	Oscillating drag amplitude	$[N]$
$\widehat{L}$	Oscillating lift amplitude	$[N]$
C_D	Drag coefficient	$[-]$
C_L	Lift coefficient	$[-]$
Co	Courant number	$[-]$
D	Drag	$[N]$
d	Diameter	$[m]$
e	Internal energy per unit mass	$[J/kg]$
f_v	Vortex shedding frequency	$[Hz]$
l	Characteristic linear dimension	$[m]$
H	Cylinder height	$[m]$
k	Coefficient of thermal conductivity	$[W/(m.K)]$
L	Lift	$[N]$
nu	Fluid kinematic viscosity in OpenFOAM	$[m^2/s]$
nut	Kinematic eddy viscosity in OpenFOAM	$[m^2/s]$
p	Pressure	$[Pa]$
Re	Reynolds number	$[-]$
S_E	Source of energy	$[-]$
S_M	Source factor	$[-]$
St	Strouhal number	$[-]$
t	Time	$[s]$
T_v	Vortex shedding period	$[s^{-1}]$
U	Velocity	$[m/s]$
u_t	Friction velocity	$[m/s]$

w	Vector of work associated with each control volume face	$[-]$
y^+	Non-dimensional distance to the wall	$[-]$

Greek Letters

κ	Turbulent kinematic energy	$[m^2/s^2]$
μ	Absolute viscosity	$[kg/(m \cdot s)]$
ν	Kinematic viscosity	$[m^2/s]$
ω	Specific dissipation rate	$[s^{-1}]$
ω_v	Vortex shedding frequency	$[rad/s]$
ϕ_v	Phase angle	$[rad]$
ρ	Density	$[kg/m^3]$
τ	Shear stress	$[N/m^2]$
τ_w	Wall shear stress	$[kg/(m \cdot s)]$
ε	Turbulent dissipation rate	$[m^2/s^3]$

Chapter 1

Introduction

1.1 Motivation

Oceans are the blood and essence of planet Earth and all humankind. They flow 97% of the planet's water and extend over three quarters of the planet. These are responsible for the production of more than half of the oxygen present in the atmosphere and absorb the most carbon dioxide. Oceans are a fundamental piece of the Man's existence on Earth, even though over 80% of this vast underwater kingdom remains unexplored, unmapped and unobserved [18] [19].

It is estimated that around 500000 earthquakes are detected annually due to natural causes, it is in the oceans that most epicenters are located. Among which, the strongest can originate tsunamis, bringing huge catastrophes to mankind. The most recent and catastrophic case that Portugal has witnessed, was the Lisbon earthquake in 1755, followed by a tsunami that destroyed most of the city. So, in order to better understand this unknown area and improve human life, it is important to explore it as much as possible without jeopardizing marine biodiversity.

Over the past decades, underwater operations in deep sea have been increasingly a topic of interest in the marine community, in which vehicles such as ROVs (Remotely Operated Vehicles) or AUVs (Autonomous Underwater Vehicles) are often used as tools for current operations at full ocean depth. However, a support ship is always needed to carry out missions with these vehicles, which imply associated operational costs.

When long term operations are desired, lander platforms are the most economic suitable option. These stable platforms are, autonomous and unmanned, oceanography research instruments to be deployed independently on the seabed over extended periods [20]. Ocean Bottom Seismometer are a good example of these platforms and, it is a receiver for seismic signals that can be heard near the seabed [21]. The major challenge when developing this type of instrument is to minimize the interference of the currents occurring at these depths, with the focus to receive a clean signal by the lander [22].

Exploration of the ocean floor can bring both great business opportunities and new scientific discoveries. Therefore, it is of great interest to be able to obtain information about it at low cost. Thus, it is relevant to use lander platforms for this purpose.

The "Continental Shelf Extension Mission Framework" (EMEPC) has worked since 2005, so that in May 2009 it delivered a proposal for the extension of the continental shelf as shown in Figure 1.1. Once accepted, Portugal will have a continental shelf of about 3900000 kilometers, where 97% will consist of ocean [1].
For this reason, it is of great importance for Portugal to improve the level of ocean floor research, either for the prevention of natural catastrophes or for scientific research. Therefore,

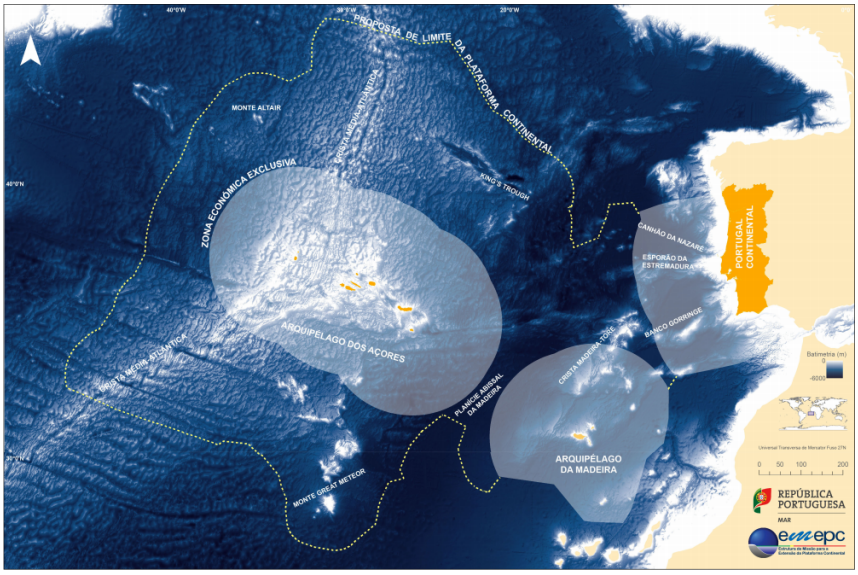


Figure 1.1: Continental platform extension [1].

the use of landers is a great choice given its low cost and the possibility of having long term missions.

1.2 Purpose and Contribution

The dissertation was a challenge proposed by CEiiA, together with partnerships with UBI and IDL (Instituto Dom Luiz). The work developed in the dissertation will be made in parallel with the current DUNE project.

Currently, a first version of the OBS is already established (Figure 1.2). In the first place, a hydrodynamic analysis will be made to the current configuration. Afterwards, modifications will be suggested to improve hydrodynamically and reduce the noise induced by the flow.

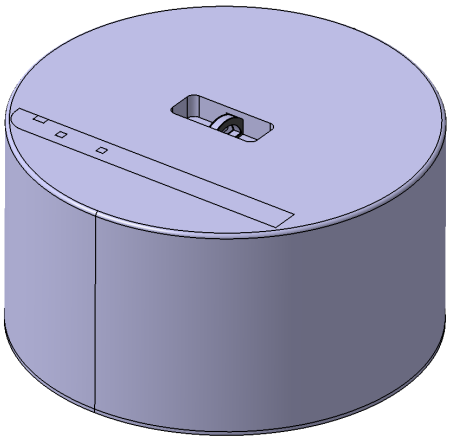


Figure 1.2: DUNE shell.

As OBS is a recent technology, there are still many improvements available. One of which is the reduction of signal interference due to flow currents in the mission depths. The main purpose

of this thesis is to reduce this interference in the signal of the OBS.

In order to evaluate this signal noise, OpenFOAM software will be used to hydrodynamically analyse the OBS. The use of this software is intended to validate it for unsteady ocean floor flows. So, in the future, one can have greater flexibility, control and experience in similar analyses.

1.3 Research and Objectives

The main focus of the dissertation is to identify, reduce or eliminate the noise in the OBS signal, from the interference of currents in its geometry. Thereby, the dissertation author outlined his goals with those of CEiiA and its partners, where a set of objectives were established. Those objectives include:

- Code validation for unsteady analysis;
- Hydrodynamic study of different geometries;
- Most suitable configuration selection.

Since errors are associated to numerical simulations, validation is an essential step to have confidence in the results. Hereafter, different configurations can be analised to select the best one according to the results and customer requirements.

1.4 Thesis Outline

The dissertation is subdivided into 5 chapters in a logical way and easy to understand order. A short description of each chapter is given below:

Chapter 1 begins with the author's motivation, the purpose and contribution of the dissertation, and finally its research and objectives.

Chapter 2 presents the literature review, where an introduction to aquatic landers is presented sand also about flow around cylindrical bodies, relevant theory, and experimental studies developed by other authors.

Chapter 3 describes the implementation of the CFD methodology. At the end of the chapter a mesh independence study is described, as well as the validation of the code by comparison with similar studies.

In chapter 4 the case study is outlined, where requirements and assumptions made are described. The results of the different configurations are presented and compared.

Finally, in the last chapter, chapter 5, is resumed the conclusions, difficulties and also future work.

Chapter 2

Literature Review

This chapter covers the review of all the topics considered relevant to the case study in question, such as UUV's (Unmanned Underwater Vehicle) history, evolution of OBSs and the influence of its design on its performance, as well as a theoretical introduction to all the principles used. In this chapter, some examples of other analyses carried out by other authors will be presented. To conclude, a theoretical introduction to the methods and models will be given.

2.1 Sea Technologies

Once the dissertation is related to Ocean Bottom Seismometers, a detailed study of all sea technologies is not relevant. Still, in order to contextualize, it is important to understand the classification of OBS's within the sea technologies. As can be seen in Figure 2.1, sea technologies can be divided into two main categories: Deep Sea Technologies, where there are manned and unmanned vehicles; and Surface Sea Technologies: which, like the previous one, can also be manned or unmanned.

Surface vehicles, as the name implies, are vehicles that operate at the surface of the oceans. Depending on the type of mission, they may have different configurations, power and accuracy. In recent years the technology of surface vehicles has passed through Autonomous Surface Vehicles (ASVs), since it has greater safety, autonomy and cost efficiency when compared to manned surface vehicles.

Hereinafter, a detailed description of Deep-Sea Benthic Landers will be given, once they are the vehicles of interest to this work. These can be subdivided according to the configuration of the vehicle. In the present work the focus goes to the axisymmetrical closed-frame lander, since the axisymmetric structure causes the flow to have the same behavior in the body regardless of the flow direction.

Depending on its mission, an OBS can be classified as [20]:

- Micro tripod open-frame landers;
- Macro open-frame landers;
- Axisymmetric closed-frame lander;
- Flat shape landers.

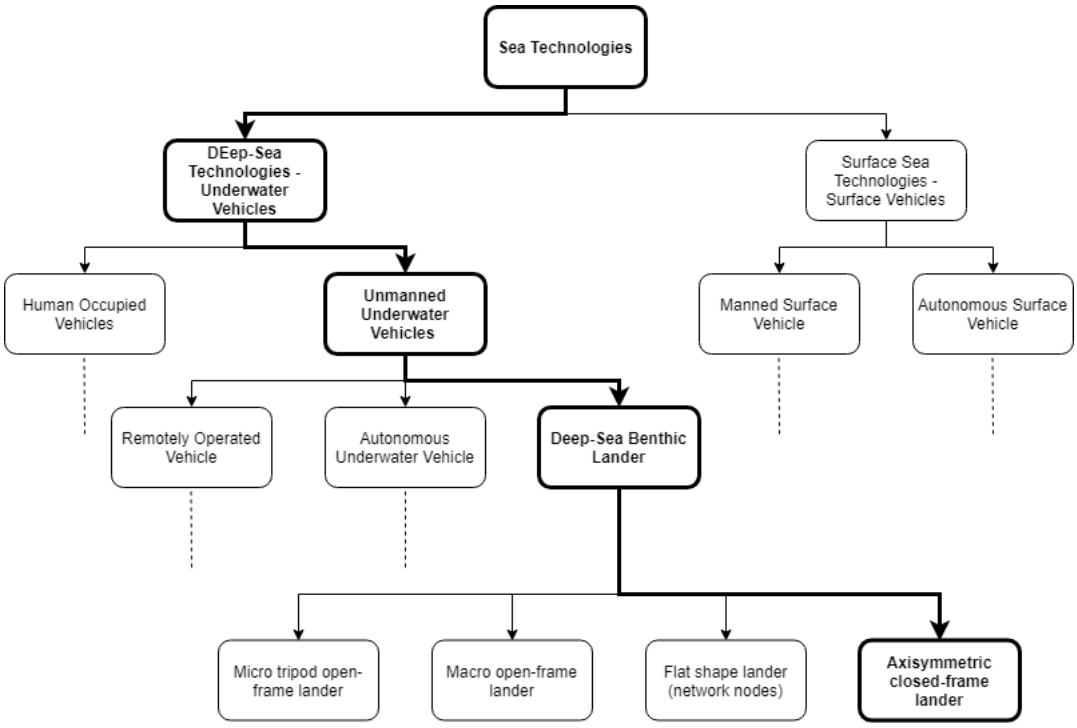


Figure 2.1: Deep-Sea Technologies - Underwater Vehicles.

2.1.1 Deep-Sea Benthic Landers

Deep-Sea Benthic Lander is a general term for an autonomous and unmanned oceanography research vehicle, which descends by free-fall to the ocean floor (without remote assistance), and operates independently on the ocean floor. The platforms have the purpose of collecting physical, biological and chemical variables over a predefined period of time. They operate independently of the surface for periods of a few days, e.g. biological studies, up to a few years, e.g. physical oceanography studies. After completing its mission, the lander releases the ballast weight by an acoustic signal transmitted from the surface or by a pre-programmed device. The vehicle comes back to the surface by merit of its positive buoyancy [23] [24].

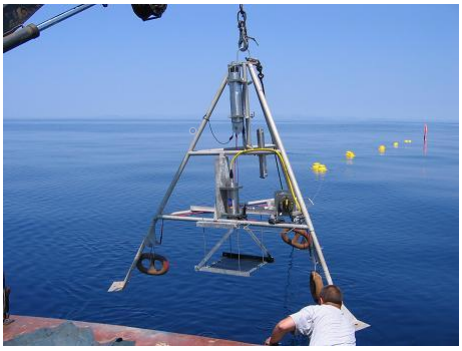
Reference [25] [26] present the different possible benthic landers configurations for their different missions.

2.1.1.1 Micro tripod open-frame landers

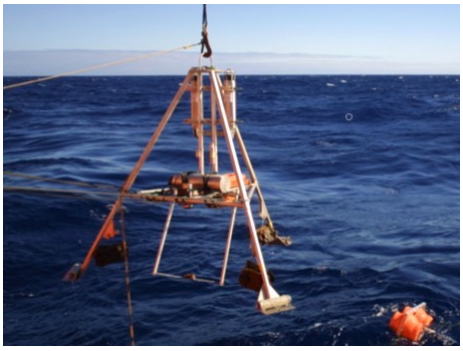
Open frame tripod shape is the simplest lander configuration, which can be equipped with different type of sensors, capturing images of deep sea creatures or analysing water quality.

An example is the ISIT (Intensified Silicon Intensifier Target) lander, in which the purpose is to research stimulated and spontaneous bioluminescence emissions in the water column and benthic boundary layer to depths around 1000 [m], as well as fauna attracted to artificial food falls (Figure 2.2a) [27].

Another example is HADAL-Lander A (Figure 2.2b), known as Alfie, which is a free fall baited lander equipped with a high resolution video camera. This lander attracts the local fauna with the bait that is attached to it and then film the bait-attending fauna [28].



(a) ISIT lander, adapted from [20].

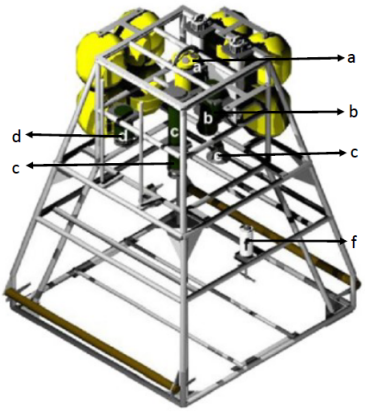


(b) Hadal-Lander A [28].

Figure 2.2: Micro tripod open-frame landers.

2.1.1.2 Macro open-frame landers

DOBO (Figure 2.3a), the Deep Ocean Benthic Observatory, lander was designed for larger payloads with the objective of observing the biological response due to deep-ocean physical parameters. This lander was designed to operate for long periods, exceeding six months [29].



(a) DOBO lander. It is equipped with an acoustic Doppler current profiler (a), camera (b), battery (c), controller (d), flash system (e), current meter (f), adapted from [30].

(b) BOBO lander, adapted from [31].

Figure 2.3: Macro open-frame landers.

Netherlands Institute of Sea Research developed the BOBO (BOttom BOundary) Observatory (Figure 2.3b), a four-meter tall lander for deep sea missions. Its structure was designed to remain on the seabed for long periods, sometimes more than one year. This lander is adaptable to several configurations, it can measure near-bottom salinity, currents speeds, currents directions, temperature and it is equipped with cameras [32] [33].

2.1.1.3 Axisymmetric closed-frame lander

The mission purpose of each lander directly influences its mechanical shape. The open-frames landers are used in order to minimize the effects of currents from the sea and allow the water to flow through the instruments and sensors.

One of the advantages of the closed-frame configuration is its hydrodynamic shape that allows to reduce the noise caused by flow on the lander (Figure 2.4).

Figure 2.4: Z3000 lander, adapted from [2].

Z3000, developed by Magseis Fairfield is an axisymmetric lander [2]. It is autonomous, lightweight, easily released and retrieved, and it can go to depths down to 3000 $[m]$. Its principal mission is seismic analysis at the bottom of the ocean [2].

2.1.1.4 Flat shape landers

The last configuration to be described is the flat shape lander. Similarly to the previous one, the flat shape structure rests on the bottom with reduced exposed drag, and uses its hydrodynamic effect to act as free descending forces that anchor the lander.

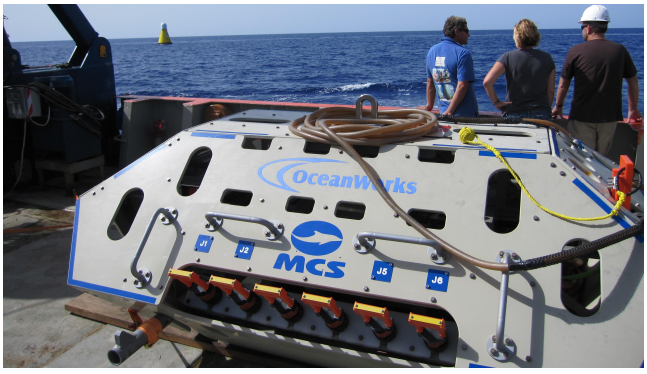


Figure 2.5: TWERC lander, adapted from [3].

In Figure 2.5 one can see a flat shape system, called TWERK (Tsunami Warning and Early Response system of Cyprus), which is one of the offshore communications backbone. This lander belongs to the OceanWorks International's seafloor network systems [34].

2.1.2 Ocean Bottom Seismometer - History, Main Applications and Design

Ocean Bottom Seismometer is traced back to the 30's of the twentieth century. The first instruments were employed in military studies, but were abandoned when new procedures concerning near-surface sources and receivers were implemented. During the late 1950's and early 1960's, the United States (US) government has sparked a new life for OBS technology. The main purpose of the seismometers was to detect and identify clandestine nuclear explosions [35].

The ”Vela Uniform project” was one of the programs that made the most advances in OBS technology, requiring a uniform distribution of monitoring station around the planet, which conducted to the need for ocean seismic stations. Since then, the use of OBS’s for passive earthquake recording or active experiments has become very common for offshore scientific purpose, as well as gas and oil reservoirs research, and tsunami forecast [36].

Nowadays, OBSs are developed with different shapes and sizes depending on their mission: from short-period instruments for active source experiments, to large OBSs operating independently for one year or more. Technology has evolved since the release of the first OBS, but the principle of the system remains the same. It is a platform equipped with one or more sensors, and a disposable compartment, where the ballast weight is placed, that descends and lands on the ocean floor to record autonomously man-made or natural seismic signals. Completed the mission time, or after receiving an acoustic signal, the disposable compartment is released and the OBS ascends to the ocean surface using its positive buoyancy [36]. Note that the OBS structure can be any of the configurations discussed in subsection 2.1.1.

According to the evolution of the technology, OBSs aims mainly on two applications:

- Research: is the main application nowadays, whether exploring the unknown ocean floor, seismic analysis for natural causes or even monitoring the location of whales;
- Defense: used to identify man-made seismic signals, which may include clandestine nuclear explosions.

Ocean Bottom Seismometers, as mentioned above, are released on the seafloor, in a very hostile environment where ambient pressure is high, ambient temperature is low and water salinity can cause corrosion of metal parts. In order to obtain credible data, the seismometer must be well laid on the seabed, withstand the currents at these depths and reduce the flow interference on the seismometer as much as possible. The instruments must be mechanically robust and tightly constructed to withstand high acceleration, shocks while handled on board the ship, release, recovery and the impact on the seabed [36] [35].

Figure 2.6: Disposal of OBSs on a ship, adapted from [4].

For reasons such as high transport costs, long release and recovery distances, the same transport is used to deploy and collect multiple devices, as depicted in Figure 2.6. Therefore, these instruments should be compact due to limited space on board the ship and/or for ease of launch and recovery. Instruments are often disassembled for ease transportation. The assembly of these type of devices should be fast and easy to perform, avoiding ambiguities, because the longer it takes, the higher the cost. Based on these requirements, there is no OBS project

considered ideal to satisfy all conditions. Thereby, each project has been designed aiming its specific mission and may have different designs and sizes (Figures 2.2 2.3 2.4 2.5 2.7) [36].

Figure 2.7: OBS deployment, launch and recovery phases, adapted from [5] [6].

2.2 Flow around Cylindrical Structures

Since OBSs are typically closed-frame axisymetrical, one has to consider that, when exposed to a fluid flow, it may be subjected to excitations or vibrations. These vibrations, known as flow induced vibrations, can lead the structure to fatigue or to the induction of undesired frequencies during its mission. Thus, during the product/structure development, it is necessary to take into account these limitations.

2.2.1 Flow Regime

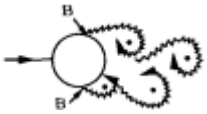
The Reynolds number (Re) is a dimensionless number used in fluid mechanics for the calculation of the flow regime (which vary from laminar to turbulent) of a given fluid, whose flow can be internal or external. By definition, the Reynolds number is a relation between orders of magnitude of inertia and viscous forces, which can be expressed by Equation 2.1:

$$Re = \frac{l \times U}{\nu} \tag{2.1}$$

where l is the characteristic linear dimension, U is the flow velocity and ν is the kinematic viscosity of the fluid.

Different flow regimes are shown in Table 2.1 and are achieved as the result of enormous changes in the Reynolds number. As the Reynolds number increases, it causes a separation of the flow in the wake region of the cylinder, which is known as vortex or eddies. Very low Reynolds number does not induce separation of the flow, but with the increase of Re, it initiates the separation of the flow, and thus becoming unstable. This leads to the phenomenon called vortex shedding at a given frequency, as summarized in Table 2.1 [10].

Table 2.1: Regimes of flow around a smooth, circular cylinder in steady current, adapted from [10].

Figures	Description	Reynolds Range
	No separation Creeping flow	$Re < 5$
	A fixed pair of symmetric vortices	$5 < Re < 40$
	Laminar vortex street	$40 < Re < 200$
	Transition to turbulence in the wake	$200 < Re < 300$
	Wake completely turbulent. A. Laminar boundary layer separation	$300 < Re < 3 \times 10^5$ Subcritical
	A. Laminar boundary layer separation B. Turbulent boundary layer separation, but boundary layer laminar	$3 \times 10^5 < Re < 3.5 \times 10^5$ Critical (lower transition)
	B. Turbulent boundary layer separation: the boundary layer partly laminar partly turbulent	$3.5 \times 10^5 < Re < 1.5 \times 10^6$ Supercritical
	C. Boundary layer completely turbulent at one side	$1.5 \times 10^6 < Re < 4 \times 10^6$ Upper transition
	C. Boundary layer completely turbulent at both sides	$4 \times 10^5 < Re$ Transcritical

2.2.2 Vortex Shedding

One of the first individuals to describe the vortex shedding was Leonardo da Vinci, by drawing sketches of vortices formation in the backflow of blunt bodies, as showed in Figure 2.8. This phenomenon is the most important feature of the flow regimes, which is common to all the flow regimes where Re is greater than 40 (Table 2.1) [10]. The boundary layer over a cylinder surface, for these values of Re, will separate due to the adverse pressure gradient introduced by the divergent geometry of the flow surroundings at the rear side of the cylinder.

Figure 2.8: Sketch from Leonardo da Vinci's notebooks [7].

Vortices or vortex are fluid swirling or rotation, that typically characterize a turbulent fluid flow [10]. In Figure 2.9, it is possible to observe a vortex, normally denominated wingtip vortex.

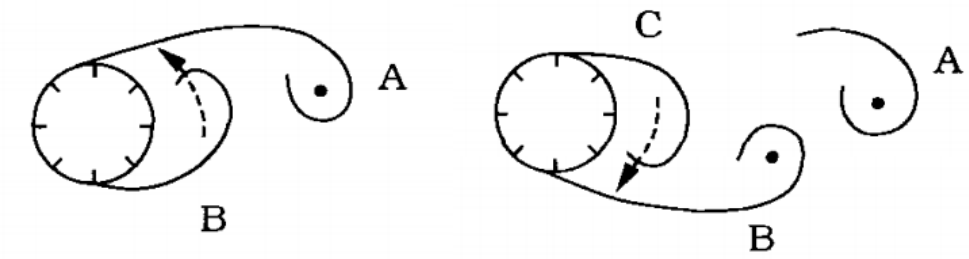
Figure 2.9: Formation of a tip vortex in the wing [8].

Vortex shedding has become a fundamental topic in fluid mechanics, since Strouhal's determination of the shedding frequencies in 1878, and stability analysis of the Von Karman vortex street in 1911 [37]. Ever since, many researches deeply studied the phenomenon of vortex formation and shedding. When the flow crosses a blunt body, natural vortex shedding or vortex street appear, as can be seen in Figure 2.10. It has been noted that such flows normally have three specific types of flow instabilities, which are: separated shear layer instability, boundary layer instability and Karman vortex instability [38].

Figure 2.10: Vortex shedding evolving into a vortex street [9].

Vortex shedding is determined by the viscosity of the fluid passing over a blunt body, that is a property of the fluid and not the flow [10].

As can be observed in Figure 2.11a, the larger vortex (vortex A) naturally becomes strong enough to draw the opposing vortex (vortex B) through the wake. The vorticity in the vortex A is clockwise while vortex B is counterclockwise. The advance of vorticity of vortex B, with signal opposite to vortex A, will cut the additional vorticity supplied to vortex A from its boundary layer. At this moment the vortex A splits and becomes free. Then it is pulled away by the flow. After the shedding of vortex A, a new vortex will be formed on the same side of the cylinder, namely vortex C (Figure 2.11b). Here, the vortex B will unwind in the same role as the vortex A, i.e. it will grow in size and strength so as to attract the vortex C through the wake (Figure 2.11b). This will lead to the vortex B shedding. This mechanism will continue each time a new vortex is shed on one side of the cylinder, where the shedding will continue to occur in an alternate mode between the two sides of the cylinder.



(a) Vortex B is being drawn across the wake, before vortex A detachment.

(b) Vortex C is being drawn across the wake, before vortex B detachment.

Figure 2.11: Mechanism of vortex shedding [10].

As result, the wake has a semblance of a vortex street as seen in Figure 2.10 [10].

2.2.2.1 Von Karman Vortex Street

A Von Karman vortex street is the designation given to an oscillating type of vortices. When a fluid flows across two dimensional blunt bodies, vortices are produced and shed in an alternating mode on the two sides of the body [39]. Given the symmetry of the body, this phenomenon will initially have a symmetrical behavior, but will eventually change into a classical alternating pattern, as the Reynolds number increases. Figure 2.12 illustrates a common Von Karman vortex street, which has this name after the studies in the field by Theodore Von Karman.

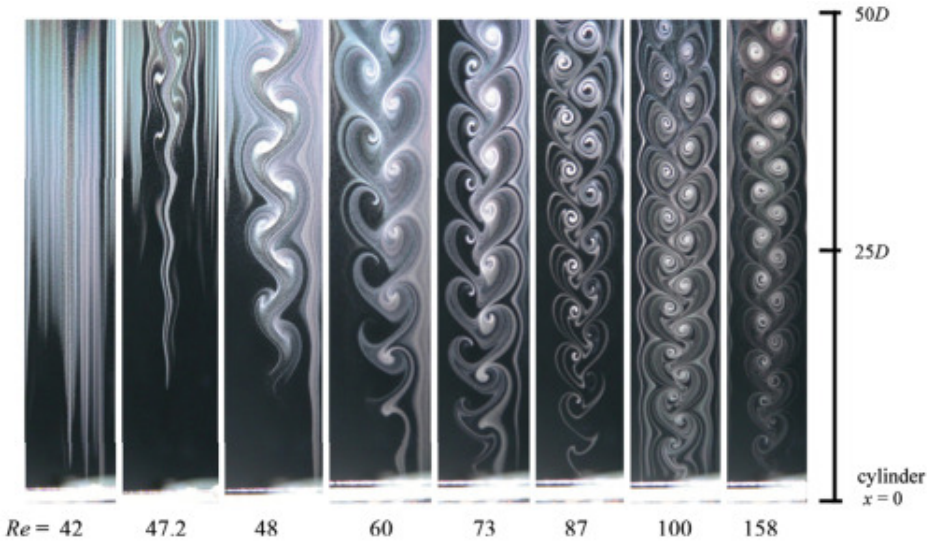


Figure 2.12: Von Karman Vortex Street at increasing Reynolds number [11].

The Von Karman vortex street is the most typical fluid dynamics example of natural instability in the transition regime (from laminar to turbulent).

2.2.2.2 Vortex Shedding Frequency

A flow creates vortices alternately as it passes through the body, where it can be found the so-call vortex-shedding frequency vortex. Established with the cylinder diameter d and the flow velocity U, on dimensional grounds, the vortex shedding frequency can be seen as a function of the Reynolds number:

$$St = St(Re) \tag{2.2}$$

or also:

$$St = \frac{f_v \times l}{U} \tag{2.3}$$

in which f_v is the vortex-shedding frequency. This dimensionless normalized shedding frequency, namely St, is called the Strouhal number. Figure 2.13 illustrates the relationship between Re and St. The vortex-shedding frequency becomes constant at lock-in, *"defined as the local synchronization between the vortex shedding frequency and the cross-flow structural vibration frequency"* [40], and the value of the Strouhal number is changing with the free-stream velocity. The separation delay of the boundary causes the large increase in St in the supercritical region [10].

2.2.3 Horseshoe Vortex

Two dimensional fluid flow around cylindrical geometries have been substantially investigated in the past. However, when a cylinder mounted on a flat plate is exposed into a fluid flow, the

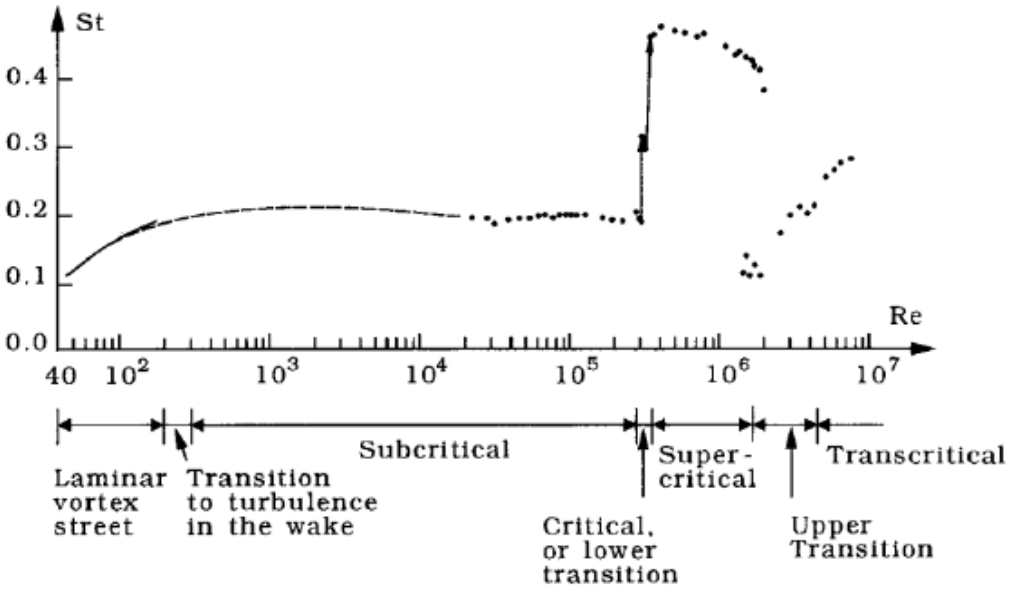


Figure 2.13: Strouhal number for smooth circular cylinder, adapted from [10].

culminating flow field is three dimensional and it presents other phenomena that are not visible in two dimensions. As a result of the adverse pressure gradient generated by the existence of the cylinder, the boundary layer over the cylinder goes through a three dimensional separation. The separated shear layer then curls to form a vortex across the cylinder base, as can be seen in Figure 2.14 [41].

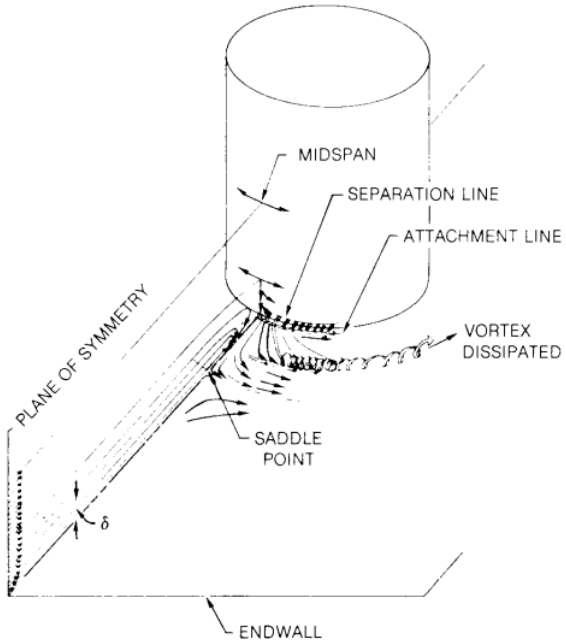


Figure 2.14: Horseshoe vortex formation, adapted from [42].

The end of this vortex is wiped out downstream and, when analysed in a perpendicular view to the flow movement, it has a characteristic similar to a horseshoe, hence its name, horseshoe vortex. Examples of this vortex shedding can be seen around the bases of bridge piers set in a river bed and at the junction of airplaine wings and fuselage [41].

2.2.4 Drag and Lift Forces

As explained in subsection 2.2.1, the regime of flow across a circular cylinder varies in function of the Reynolds number (Table 2.1). The pressure distribution around the cylinder changes periodically as the shedding process takes place. The result is a periodic variation of the force components on the cylinder. Generally, the force in flow direction is called drag (D), while the force normal to flow is called lift (L). The drag force is an unavoidable consequence of a fluid moving through a body and it changes periodically in time, due to the vortex shedding, oscillating around mean drag. Drag force can be subdivided into two components:

- Pressure drag: dependent on the shape of the body and depends on the flow separation point;
- Skin friction: dependent on the viscous friction between the fluid and surface of the object the flow is surrounding.

Similarly, the lift force arises, fluctuating at the vortex shedding frequency, and usually begins with a zero mean. It is generated perpendicular to the direction of the fluid, as it moves throughout the stationary body.

Drag and lift forces are given by the following equations (according to time):

$$D(t) = \overline{D} + \widehat{D}\sin(2\omega_v t + \phi_v) \tag{2.4}$$

$$L(t) = \widehat{L}\sin(\omega_v t + \phi_v) \tag{2.5}$$

where $\widehat{D}$ and $\widehat{L}$ are oscillating drag and lift amplitude respectively, and $\overline{D}$ is the mean drag. The phase angle between the vortex shedding and the oscillating forces is defined by ϕ_v, and the ω_v represents the vortex shedding frequency, defined by Equation 2.6:

$$\omega_v = \frac{2\pi}{f_v} \tag{2.6}$$

Sumer [10] described an experiment accomplished by Drescher in 1956 [43], where Drescher was able to trace the lift and drag forces from the calculated pressure distribution, as it is illustrated in Figure 2.15. As shown in Figure 2.15, drag and lift forces are influenced by the vortex shedding frequency, where the lift force has the same period and the drag force has half of the vortex shedding period.

In the Figure 2.15, C_L and C_D are the dimensionless parameters for lift and drag forces respectively, and can be obtained as:

$$C_L = \frac{L}{\rho\frac{1}{2}HdU^2} \tag{2.7}$$

$$C_D = \frac{D}{\rho \frac{1}{2} H d U^2} \tag{2.8}$$

where ρ, H are the fluid density and cylinder height respectively [10].

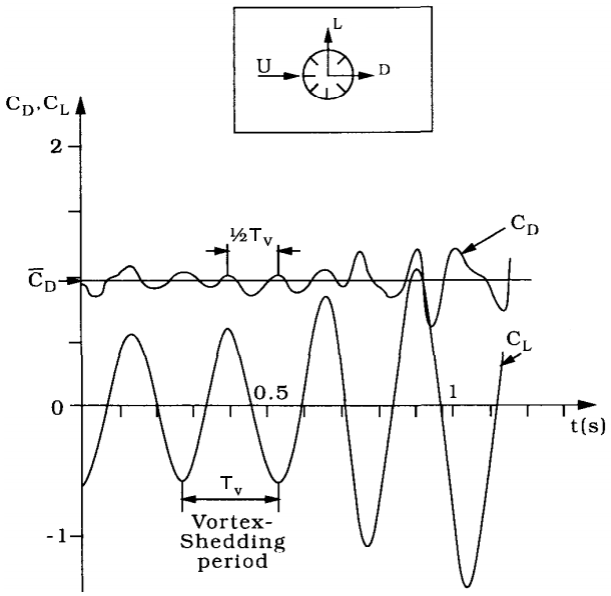


Figure 2.15: Lift and drag forces traces, adapted from [10].

In Figure 2.16, it can be seen the so-called standard drag curve that illustrates the main features in the drag of a circular cylinder. The drag coefficient lies approximately around $1.0 - 1.2$ until it reaches the critical region, in a Reynolds Number range near 2×10^5. Therefore, the instability causes the flow to pass the boundary layer from laminar to turbulent. Is possible to observe by the standard drag curve, in Figure 2.16, the drag coefficient behavior for an infinite circular cylinder for the Reynolds range presented [12].

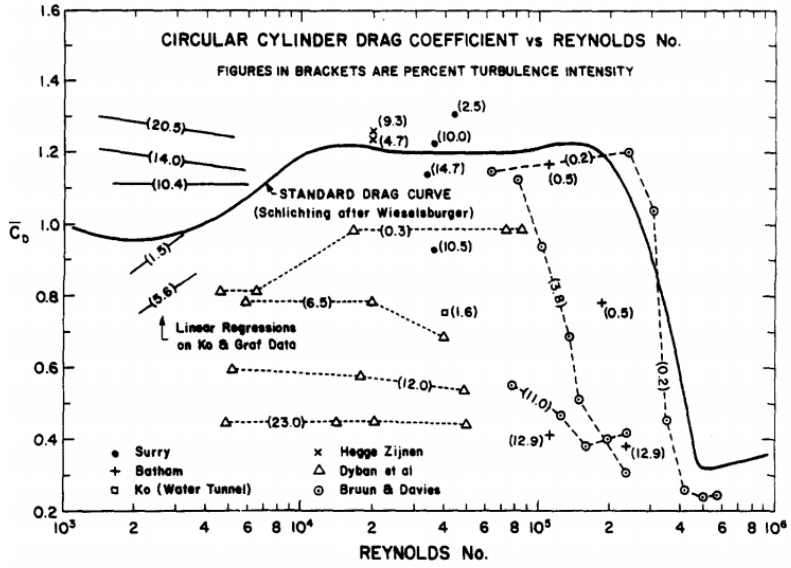


Figure 2.16: Reynolds number vs circular cylinder drag coefficient as obtained by several researchers, adapted from [12].

2.3 Computational Fluid Dynamics

Computational Fluid Dynamics is a numerical modeling of fluid mechanics or simply a computer based simulation to solve and analyse problems associated with heat transfer, fluid flow and related phenomena such as chemical reactions. CFD techniques provide nowadays a vast range of applications for industrial and non-industrial areas, such as scientific studies. CFD is for these reasons a powerful technique, regarding the simulation of fluid flows.

During the 1960's, CFD started to be introduced in the aerospace industry, being integrated into the design, research, development and manufacturing of aircraft and jet engines.

In CFD simulations, unlike the model testing facility or experimental laboratory, there is no requirement for a huge facility. Some of the advantages of using CFD compared to experimental approaches are presented[44][13]:

- Almost always faster;
- Significantly cheaper (20%-40%);
- Real scale models are typically used;
- Ability to evaluate a system under dangerous conditions, e.g. reliability study and accident investigation;
- High detail level of results provided.

Henceforth, information regarding CFD and related fundamentals will be described.

2.4 Governing Equations of Viscous Flow

Derived from its molecular structure, fluids are substances that do not offer resistance to external shear forces: even the minimum force causes a deformation of a fluid particle. Both types of fluids, liquids and gases, can be considered a continuous substance in most cases of interest and obey the same laws of motion, despite of the existence of a distinction between them [44].

The fluid flow is generated by the action of external applied forces. They can be divided into two types of force: body forces, e.g. forces induced by rotation and gravity; and surface forces, e.g. pressure and shear force created by a movement of a rigid wall relative to the fluid [44].

Although all fluids behave similarly during the action of forces, their macroscopic properties are considerably different. These properties should be known if the fluid movement is to be studied, being density and viscosity the most important properties of the fluid [44].

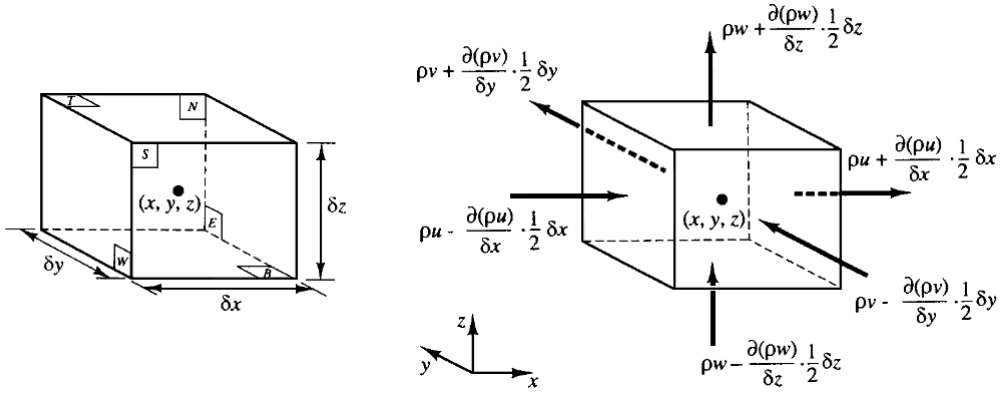
The flow velocity affects its properties in a number of ways. At very low speed, the inertia of the fluid can be neglected and a creeping flow can be considered. There are many other phenomena that affect the fluid flows, including the density variations, giving rise to buoyancy and temperature variations that lead to heat transfer [44].

Since the fluid behaves according to the laws of physics, the conservation laws can be derived by considering a given volume of infinitesimally small control (Figure 2.17a) and its extensive properties, such as energy, mass and momentum [44]. Thereby, three governing equations of the fluid flow can be obtained, in which these equations represent mathematical statements of the laws of conservation of physics:

- Continuity equation or Mass conservation equation;
- Momentum conservation equation (Newton's second law);
- Energy conservation equation (First law of thermodynamics).[13].

2.4.1 Mass Conservation

The mass conservation theory states that the mass will remain constant, in a closed system, over the time. Consequently, the flow that are leaving the element are given a negative sign, while the ones that are directed into the element get a positive sign, as they produce an increase of mass in the element (Figure 2.17b).



(a) Fluid element. (b) Mass flow transfers of fluid element.

Figure 2.17: Fluid element for conservation laws, adapted from [13].

Following all steps of the deduction in [44] and [13], Equation 2.9 can be achieved:

$$\frac{\partial \rho}{\partial t} + div(\rho \overrightarrow{U}) = 0 \tag{2.9}$$

Equation 2.9 is for an unsteady, three-dimensional mass conservation at a volume in a compressible fluid. In Equation 2.9, the first term represents the rate of change in time of density, i.e. mass per volume unit, while the second term describes the net flow of mass out of the element through its boundaries and is called the convective term, where $\overrightarrow{U}$ is fluid velocity vector (u, v, w).

In the case of an incompressible fluid, i.e. a liquid, the density is constant, so the Equation 2.9 becomes:

$$div(\overrightarrow{U}) = 0 \tag{2.10}$$

2.4.2 Momentum Conservation

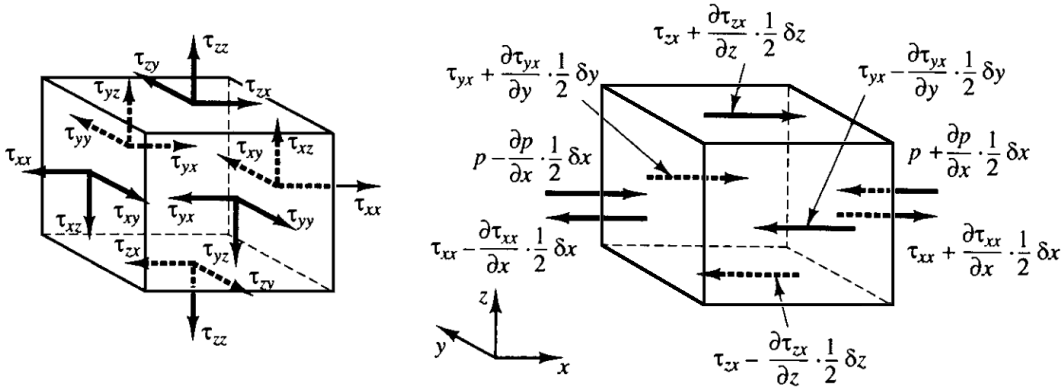
Based on Newton's second law, the momentum conservation equation depicts that the sum of forces in a fluid particle must be equal to the rate of change of momentum of the particle. Differentiating the rate of momentum change per unit volume of a fluid particle for x, y and z direction [13], comes:

$$x - direction : \rho \frac{\partial \overrightarrow{u}}{\partial t} \tag{2.11}$$

$$y - direction : \rho \frac{\partial \overrightarrow{v}}{\partial t} \tag{2.12}$$

$$z - direction : \rho \frac{\partial \overrightarrow{w}}{\partial t} \tag{2.13}$$

In Figure 2.18a, the stress state of a fluid element is defined in terms of pressure, where the normal stress is denoted by p, and the nine viscous stress components are denoted by τ. The subscripts i and j in τ_{ij} imply that the stress component operates in the y-direction on a surface normal to the x-direction.



(a) Stress components of fluid element. (b) Stress components in the x-direction.

Figure 2.18: Stress components, adapted from [13].

Firstly, the components in the x-direction of the pressure force and stress components are considered, as showed in Figure 2.18b, and replicating in the y and z directions, one can arrive at the final moment equations (again following the steps of the deductions done in [44] and [13]).

$$x - direction : \rho \frac{Du}{Dt} = \frac{\partial(-p + \tau_{xx})}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + S_{Mx} \tag{2.14}$$

$$y - direction : \rho \frac{Dv}{Dt} = \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial(-p + \tau_{yy})}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} + S_{My} \tag{2.15}$$

$$z - direction : \rho\frac{Dw}{Dt} = \frac{\partial\tau_{xz}}{\partial x} + \frac{\partial\tau_{yz}}{\partial y} + \frac{\partial(-p+\tau_{zz})}{\partial z} + S_{Mz} \tag{2.16}$$

where source S_M describes the contribution of the body forces on the total forces per unit volume of the fluid.

2.4.3 Energy Conservation

Finally, from the first law of thermodynamics comes the last conservation equation, the energy conservation equation. This equation states that: *"the rate of change of energy of a fluid particle is equal to the rate of heat addition to the fluid particle plus the rate of work done on the particle"*[13]. The energy conservation equation can be presented as:

$$\rho\frac{D(e+\frac{1}{2}(u^2+v^2+w^2))}{Dt} = div(k\nabla T) - div(w) + S_E \tag{2.17}$$

where k is the coefficient of thermal conductivity, e is the internal energy per unit mass, ∇T the temperature gradient, w is the vector of work associated with each control volume face and S_E is a source of energy per unit volume per unit time [13].

2.5 General Transport Equation

The Navier-Stokes equations can be used to derive the transport equation of viscous and incompressible fluids. Having a Newtonian fluid, where stress versus strain rate curve is linear, the conservation of momentum equations become the Navier-Stokes equations, and can be defined for x, y and z directions as:

$$\rho\frac{Du}{Dt} = -\frac{\partial p}{\partial x} + \frac{\partial}{\partial x}[2\mu\frac{\partial u}{\partial x} + \eta div\overrightarrow{U}] + \frac{\partial}{\partial y}[\mu(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x})] + \frac{\partial}{\partial z}[\mu(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x})] + S_{Mx} \tag{2.18}$$

$$\rho\frac{Dv}{Dt} = -\frac{\partial p}{\partial y} + \frac{\partial}{\partial x}[\mu\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}] + \frac{\partial}{\partial y}[2\mu\frac{\partial v}{\partial y} + \eta div\overrightarrow{U}] + \frac{\partial}{\partial z}[\mu(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y})] + S_{My} \tag{2.19}$$

$$\rho\frac{Dw}{Dt} = -\frac{\partial p}{\partial z} + \frac{\partial}{\partial x}[\mu\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}] + \frac{\partial}{\partial y}[\mu(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y})] + \frac{\partial}{\partial z}[2\mu\frac{\partial w}{\partial z} + \eta div\overrightarrow{U}] + S_{Mz} \tag{2.20}$$

where μ and η are two constants of proportionality: μ is the (first) dynamic viscosity, relating stresses to linear deformation and η is the second viscosity, relating stresses to the volumetric deformation.
Therefore, the general transport equation is modulated as:

$$\frac{\partial(\rho\phi)}{\partial t} + div(\rho\phi\overrightarrow{U}) = div(\Gamma grad\phi) + S_\phi \tag{2.21}$$

Where ϕ expresses the value of a property per unit mass. This equation has several transport processes. The first term on the left side is the rate of change term, or normally denominated as unsteady term, while the second term is the convective transport. On the right side of the equation, firstly the diffusion transport term is presented, and finally the source term [13].

2.6 Modeling Approaches for Turbulence

In engineering, most of the flows are turbulent. Thereby, it is important to study turbulent flow regime not only in a theoretical way, but also with a practical approach. The turbulent flow is characterized by a three-dimensional, fluctuating and highly unsteady flow, or what can be called chaotic, with high vorticity levels and a wide range of length scales. Turbulent flow is a time-dependent process, being difficult to get the solution of the transport equation for this kind of flow [45] [46].Thus, these complex equations need to be solved numerically using computational methods.

A few methods can be used to predict turbulent flow: Direct Numerical Solution (DNS), Large Eddy Simulations (LES) or Reynolds Averaged Navier-Stokes (RANS) [44] [13].

DNS is the most accurate approach to turbulent flow, which solves the Navier-Stokes equations without averaging. In this kind of simulation all the turbulent motions in the flow are solved. The DNS results give a more detailed flow information compared to the other two methods. These results sometimes have more information than an engineer may need, due to the high time consumed. Considering the fact of DNS being a fundamental tool in turbulence, its computational cost is very high, even in low Re flows. This method becomes inefficient unless a powerful computer is used [44].

On the other hand, LES only solves the time dependence of the larger eddies, once they contain most of the energy, and models, time averaging, the small ones. Namely, LES solves spatially averaged RANS equations, which apply a low-pass filter. LES only solves large eddies. Consequently, LES is less computationally demanding than DNS and it is normally preferred when RANS do not give reliable values [44].

RANS equations are used, giving approximate time-averaged solutions to the Navier-Stokes equations. Depending if the flow behavior is in the time domain or not, there are two possible approaches. The first one is flow with statistic steady trend, while the second is assuming all variables as the sum of the time mean value, and a fluctuation about that value. These fluctuations, the Reynolds stresses, arise as additional unknowns that force more equations to model RANS turbulence.

However, when the flow has unsteady behavior, time averaging can not be used, and the ensemble averaging approach has to be used. This process is known as Reynolds averaging when applied to the Navier-Stokes equations.

RANS are broadly used, relative to DNS and LES, since less computational power, in some orders of magnitude, is required [44].

These three methods, described above, are summarized in Figure 2.19.

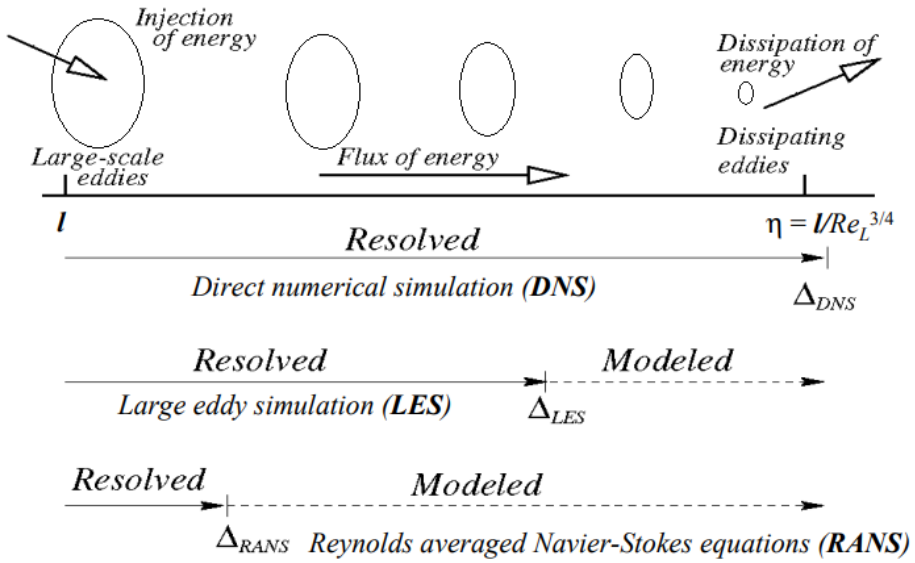


Figure 2.19: Prediction methods, adapted from [14].

2.7 Turbulence Models

Since it is being utilized the software OpenFOAM, it will be represented the turbulence models related to this tool. Thus, some of these models provided by OpenFOAM are:

- Spalart-Allmaras model (one-equation model);
- κ-ε models (two-equation model);
  - ⋆ Standard κ-ε model;
  - ⋆ Realizable κ-ε model;
  - ⋆ Renormalization-group κ-ε model.
- κ-ω models (two-equation model);
  - ⋆ Standard κ-ω model;
  - ⋆ Shear Stress Transport κ-ω model.
- Large Eddy Simulation Model;
  - ⋆ Smagorinsky model.

Others models can be found in [47] and [48]. It is necessary to refer that "There is not yet a single, practical turbulence model that can reliably predict all turbulent flows with sufficient accuracy." [47]. Therefore, the choice of the turbulence model should be done depending on computational effort and purpose. Following the ideas concluded from literature review, three models will be described: standard κ-ε, standard κ-ω and κ-ω SST models. More detailed information about turbulence models is found in [49].

2.7.1 Standard κ-ε model

Standard κ-ε is a two-equation model which solves two transport equations, and models the Reynolds Stresses using the Eddy Viscosity approach [13]. It is a semi-empirical model and, by being robust and fairly accurate, is the most used model for industrial applications by engineers [47] [48].

$$\frac{\partial \rho\varepsilon}{\partial t} + \frac{\partial}{x_j}(\rho u_j \varepsilon - (\mu + \frac{\mu_\tau}{\sigma_\varepsilon})\frac{\partial \varepsilon}{\partial x_j}) = c_{\varepsilon 1}\frac{\varepsilon}{\kappa}\tau_{tij}S_{ij} - c_{\varepsilon 2}f_2\rho\frac{\varepsilon^2}{\kappa} + \phi_\varepsilon \tag{2.22}$$

$$\frac{\partial \rho\kappa}{\partial t} + \frac{\partial}{x_j}(\rho u_j \frac{\partial \kappa}{\partial x_j} - (\mu + \frac{\mu_\tau}{\sigma_\kappa})\frac{\partial \kappa}{\partial x_j}) = \tau_{tij}S_{ij} - \rho\varepsilon + \phi_\kappa \tag{2.23}$$

The model takes into account the effects of convection and diffusion in turbulent flows. The equation 2.22 computes the rate of dissipation of turbulent kinetic energy per mass unit, ε, whereas the equation 2.23 computes the turbulent kinetic energy of the flow, κ [13] [49].

This model is only valid for fully turbulent flows, since it presents several limitations such as:

- it assumes the flow as fukky turbulent;
- it ignores the effects of molecular viscosity [48];
- wall functions must be used at near-wall regions, due to ε [47] [48].

2.7.2 Standard κ-ω model

The Standard κ-ω model is a widely tested two-equations eddy viscosity model. This model was developed in parallel with Standard κ-ε model as an alternative to define the eddy viscosity function. Convective transport equations are solved for the turbulent kinetic energy, κ, and its specific dissipation, ω (Equation 2.24 and 2.25)[49].

$$\frac{\partial \rho\kappa}{\partial t} + \frac{\partial}{x_j}(\rho u_j\ \kappa - (\mu + \sigma\mu_\tau)\frac{\partial \kappa}{\partial x_j}) = \tau_{tij}S_{ij} - \beta\rho\omega\kappa \tag{2.24}$$

$$\frac{\partial \rho\omega}{\partial t} + \frac{\partial}{x_j}(\rho u_j\ \omega - (\mu + \sigma\mu_\tau)\frac{\partial \omega}{\partial x_j}) = \alpha\frac{\omega}{\kappa}\tau_{tij}S_{ij} - \beta\rho\omega^2 \tag{2.25}$$

Compared to the Standard κ-ε model, the Standard κ-ω model proved to have superior numerical stability, especially in the viscous sublayer near the wall. As this model has high ω values in the wall region, it does not require explicit wall-damping functions like the Standard κ-ε model and other two equation models [49]. This model is robust and accurate for a large range of boundary layer flows with pressure gradient[48] [47].

2.7.3 SST κ-ω model

The SST κ-ω model merge multiple desirable elements of both previous two-equations models [49]: "The SST κ-ω model uses a blending function to gradually transition from the Standard κ-ω model near wall to a high Reynolds number version of the κ-ε model in the outer portion of the boundary layer." [47].

In order to combine the Standard κ-ω and the Standard κ-ε model, the latter must be transformed into a κ-ω formulation. The differences of this formulation for Standard κ-ω are that an additional cross-diffusion term appears in the ω equation and the modeling constants differ[49]. The transport equation for the turbulent kinetic energy remains the same, while the transport equation for the specific dissipation of turbulence becomes Equation 2.26:

$$\frac{\partial \rho\omega}{\partial t} + \frac{\partial}{x_j}(\rho u_j\, \omega - (\mu + \sigma\mu_\tau)\frac{\partial \omega}{\partial x_j}) = P_\omega - \beta\rho\omega^2 + 2(1 - F_1)\frac{\rho\sigma_{\omega 2}\partial\kappa\partial\omega}{\omega\partial x_j\partial x_j} \tag{2.26}$$

Regarding the Standard κ-ω, it includes a modified turbulent viscosity formulation, which accounts for the transport effects of the main turbulent shear stress [47] [48].

For more information on equations 2.22, 2.23, 2.24, 2.25 and 2.26, as well as the respective constants of each see reference [49].

2.8 State-of-the-Art

Wang, Zhou and Mi [50] experimentally investigated the effect of aspect ratio on Drag and Strouhal number of a wall-mounted circular cylinder in subcritical and critical regime. The studies were performed at different aspect ratios between 2.65-7 with a range of Reynolds number of 1.24×10^4 to 1.73×10^5 and free-stream longitudinal turbulent intensity was about 0.2%. The authors concluded that both drag coefficient and Strouhal number for the wall-mounted finite length cylinder are lower than their counterparts in a $2D$ cylinder, and that they reduce monotonously as aspect ratio decreases.

Okamoto and Yagita [51] performed an experimental study of the flow around a circular cylinder of finite length placed normal to the plane surface. The range of free stream velocity varies from 2.56 to 22.6 $[m/s]$, with a range of Reynolds number of 6.5×10^5 to 3.89×10^6. The study was performed by varying the cylinder aspect ratio from 1 to 12.5 in a wind tunnel with dimensions of $250 \times 250 \times 1800$ $[mm]$ with turbulent intensity of 0.45%. The results display a decrease in shedding vortex frequency with decreasing aspect ratio.

Kawamura, Hiwada, Hibino, Mabuchi and Kumada [52] conducted flow visualization experiments around a finite circular cylinder on a flat plate, in order to study the three dimensional effects of the flow. The study was carried with different aspect ratio (from 1 to 8) in a wind tunnel having as dimensions $300 \times 300 \times 2000$ $[mm]$. The diameter of these cylinders was 20 $[mm]$ and the main flow velocity was 24 $[m/s]$, which gives as Reynolds number 3.2×10^4. The results show an increase of the Strouhal number as the aspect ratio grows.

Sakamoto and Arie [53] carried out an experiment in order to study vortex shedding in a circular cylinder placed vertically in a turbulent boundary layer. All Strouhal number measurements were performed in a $400 \times 400 \times 4000$ [mm] wind tunnel. At the maximum free-stream velocity that was employed (20 [m/s]), the free-stream turbulence level was about 0.2%. The results illustrate a drop in Strouhal number as the aspect ratio decreases.

Moreover, Sakamoto and Oiwake [54] also investigated the fluctuating force generated by the shedding vortices of a circular cylinder placed vertically in a turbulent Boundary Layer. The experiment was designed in a $600 \times 600 \times 5400$ [mm] wind tunnel with 0.3% free stream turbulence at a flow rate of 30 [m/s]. The results lead to an increase in the magnitude of fluctuating lift with increasing cylinder aspect ratio.

Okamoto and Sunabashiri [55] executed an experimental study in order to analyse the vortex shedding from a circular cylinder of finite height located on a Ground Plane. For this study, they used a wind tunnel with a working section of $500 \times 500 \times 2000$ [mm] and a Reynolds number between 2.5×10^4 and 4.7×10^4 with a maximum of free-stream turbulence level of 0.4%. The results show a decrease in Strouhal number as the aspect ratio decreases to the value of $H/d = 4$. Reducing cylinder aspect ratio further ($H/d < 4$) Strouhal number stars to increase with decreasing aspect ratio. The authors conclude that for $H/d = 1$ and 2 the vortices on both sides of the circular cylinder oscillate symmetrically, while for larger aspect ratios ($H/d = 4-7$) these oscillations change to antisymmetrical.

Park and Lee [56] studied the free end effects on near wake flow structure behind a circular cylinder with finite height. For this study three aspect ratios were studied ($H/d = 6, 10, 13$) in the subsonic wind tunnel at a Reynolds number of 20000. These studies had a free-stream turbulence intensity of not more than 0.08% for a velocity of 10 [m/s]. The results of this investigation demonstrate that vortex shedding frequency decreases with decreasing cylinder aspect ratio. The authors have found that the vortex formation region and periodic vortex shedding disappears very close to the cylinder top free end. The cause would be the descent of twin vortices from the free end, which interacts with regular shed vortices on both sides of the cylinder.

Zhang [57] made a comparison between different turbulence models in an unsteady flow around a finite circular cylinder numerically using OpenFOAM software. The study was conducted around a cylinder with an aspect ratio equal to one unit where ten turbulence models were studied. Among the RANS turbulence models considered, the SST κ-ω model presented the best performance, being close to the values obtained numerically by the LES models. With this study the author was in a position to validate that this model is highly recommended for adverse pressure gradient scenarios and separating flow around a finite cylinder.

Chapter 3

CFD Methodology

Implementation, Verification and Validation (V&V) must be considered as key steps to be successful in CFD, being the last two, V&V, the main means to evaluate the reliability and accuracy of a numerical simulation. This chapter describes the methodology used to validate the code. Hereinafter, the validation geometry used and the considered methods will be presented. Finally, a comparison between the results obtained for the validation geometry and those present in the literature is presented.

3.1 CFD workflow

CFD codes are performed around numerical algorithms, which are capable of solving fluid flow problems. In order to have access to the power of solution, commercial CFD packages include graphical interfaces for analyse problem data and examine results, e.g. give input values, define boundaries conditions and others. In contrast, open-source software, most of the times, comes without a graphical interface. For this reason, it is less user friendly, but without the heavy cost of commercials. However, any CFD code contains in the implementation phase three main stages:

- Pre-processing;
- Processing;
- Post-processing.

These three stages are briefly described in the Figure 3.1.

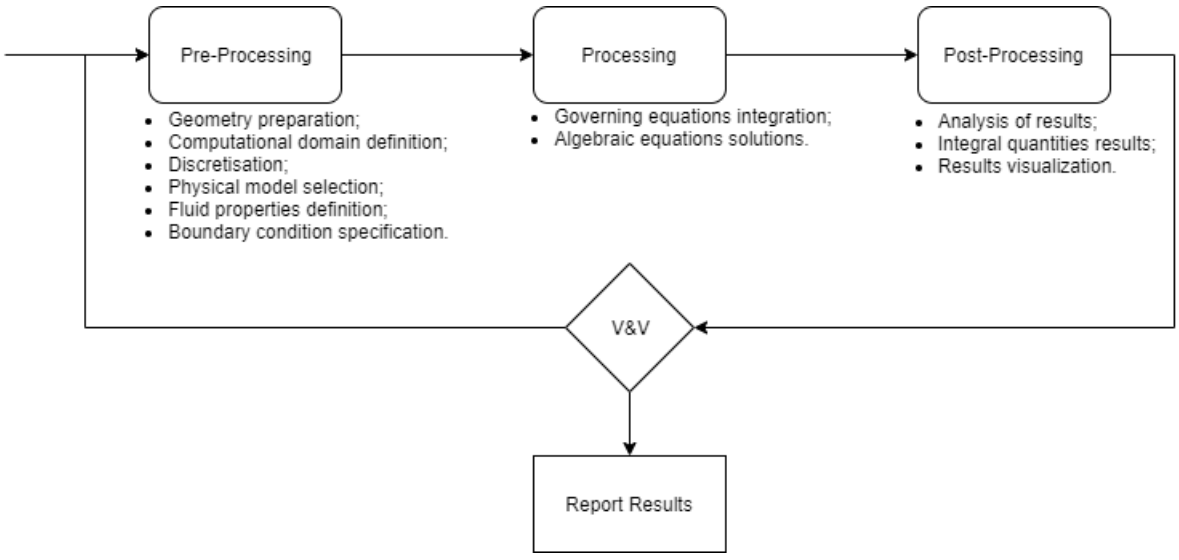


Figure 3.1: CFD work scheme [15].

Once these three stages are concluded, results should be evaluated. During this evaluation, two sub-stages are performed: verification and validation. The main goal of verification is to check:

- Results homogeneity;
- Numerical error.

Finally, in validation stage the results obtained will be compared with experimental or even with reliable numerical data. This will secure that the computational simulations agree with physical reality [58].

3.2 OpenFOAM

This thesis presents the validation of unsteady flows using OpenFOAM software, an open-source and freeware CFD toolbox. Open-source Field Operation And Manipulation is, in short, a set of libraries (approximately one hundred C++ libraries), providing utilities and solvers, previously compiled, useful for the user to solve CFD problems [16].

OpenFOAM was developed with the purpose of sharing information between users. Consistent with this philosophy, it enables to reuse code and it is available for all users to view, review and modify the original code. Thereby, and as the software has a large community of users who contribute to the faster code development, OpenFOAM is expanding a lot in scientific research [59].

So, the main advantages of this software are: the code is open, is customizable, has a supportive community and, last but not least, it is a free software (which is a huge advantage over commercial software). Thus, the use of OpenFOAM as a CFD tool has been increasing over the years [59].

3.2.1 Model Setup

The Finite Volume Method is used as basis for OpenFOAM simulations [13]. This method keeps the fluid dynamic quantities in a single node for each mesh element. In order to avoid unrealistic behavior by storing velocity and pressure at exactly the same point, the code enforces interpolation schemes to separate these values between the center of each face and the center of the cell [16].

OpenFOAM comes with some tutorials, where it has basic cases. A case typically has three folders inside, namely *0, constant* and *system* (Figure 3.2). In directory *0*, the boundary conditions for each simulation are delineated, and it is where the user includes a serie of documents that characterize the physics of the problem, such as: velocity, pressure and turbulent quantities, κ, $omega$ and nut (eddy kinematic viscosity). The *constant* directory includes the aspects that remain constant throughout the simulation, e.g. the mesh (*polyMesh*), the transport models (*transportProperties*) and the turbulence model (*turbulenceProperties*). At last, in the *system* folder, the simulation is described by the user. Here, the simulation can be controlled by editing at least three files: *controlDict*, *fvSolutions* and *fvSchemes*. In these three files the user defines how the equations are discretized and which numerical algorithm is to be use to solve the present system of equations [59].

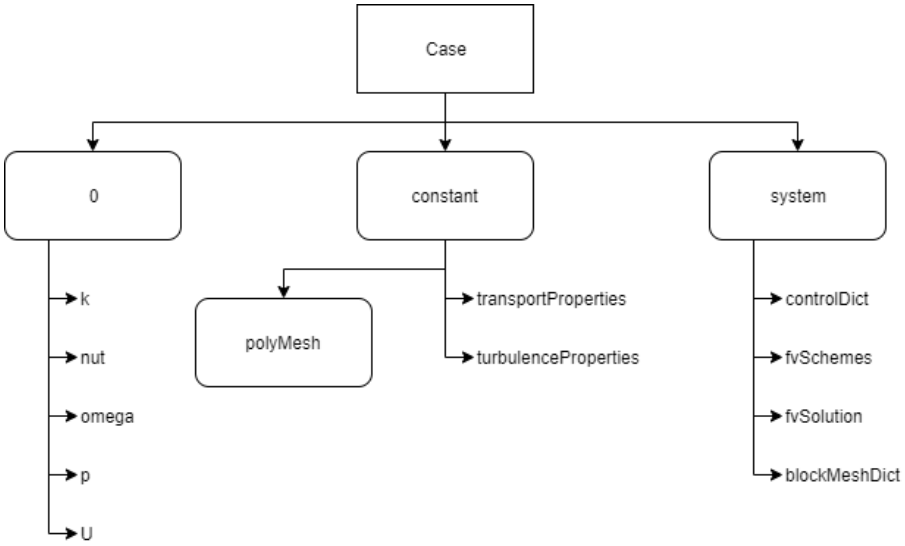


Figure 3.2: Case configuration.

A different folder system was adopted for the efficiency of the whole process, resulting in Figure 3.3:

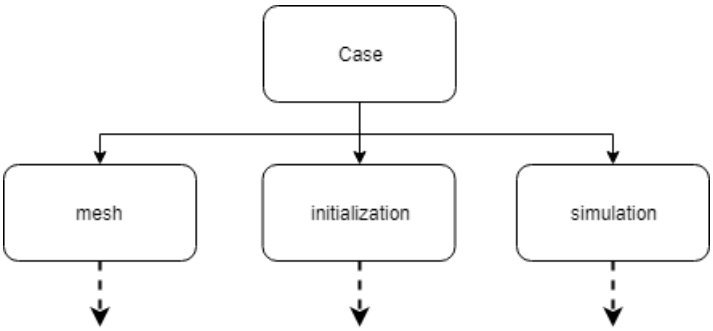


Figure 3.3: Case configuration.

Firstly, the control volume is discretized in the *mesh* folder. This sets up a folder, *polyMesh*, inside the directory constant. This folder (*polyMesh*) is archived to the folder *initialization/constant* and the folder *simulation/constant*. After control volume discretization, an initialization has been made to the case, using a quick steady state simulation. This simulation is performed in the *initialization* folder, and helps the transient simulation to take much less time to achieve results. Finally, the mesh is saved inside *simulation* folder and the steady-state initialization is used to begin the transient simulation.

The process detailed above is summarized is summarized in Figures 3.4. In Figure 3.2 to 3.4, folders are identified as boxes, while other representations are text files. All the text files represented in Figure 3.4 are fully described in Appendix A.1 and the entire process will be further detailed in section 3.4.

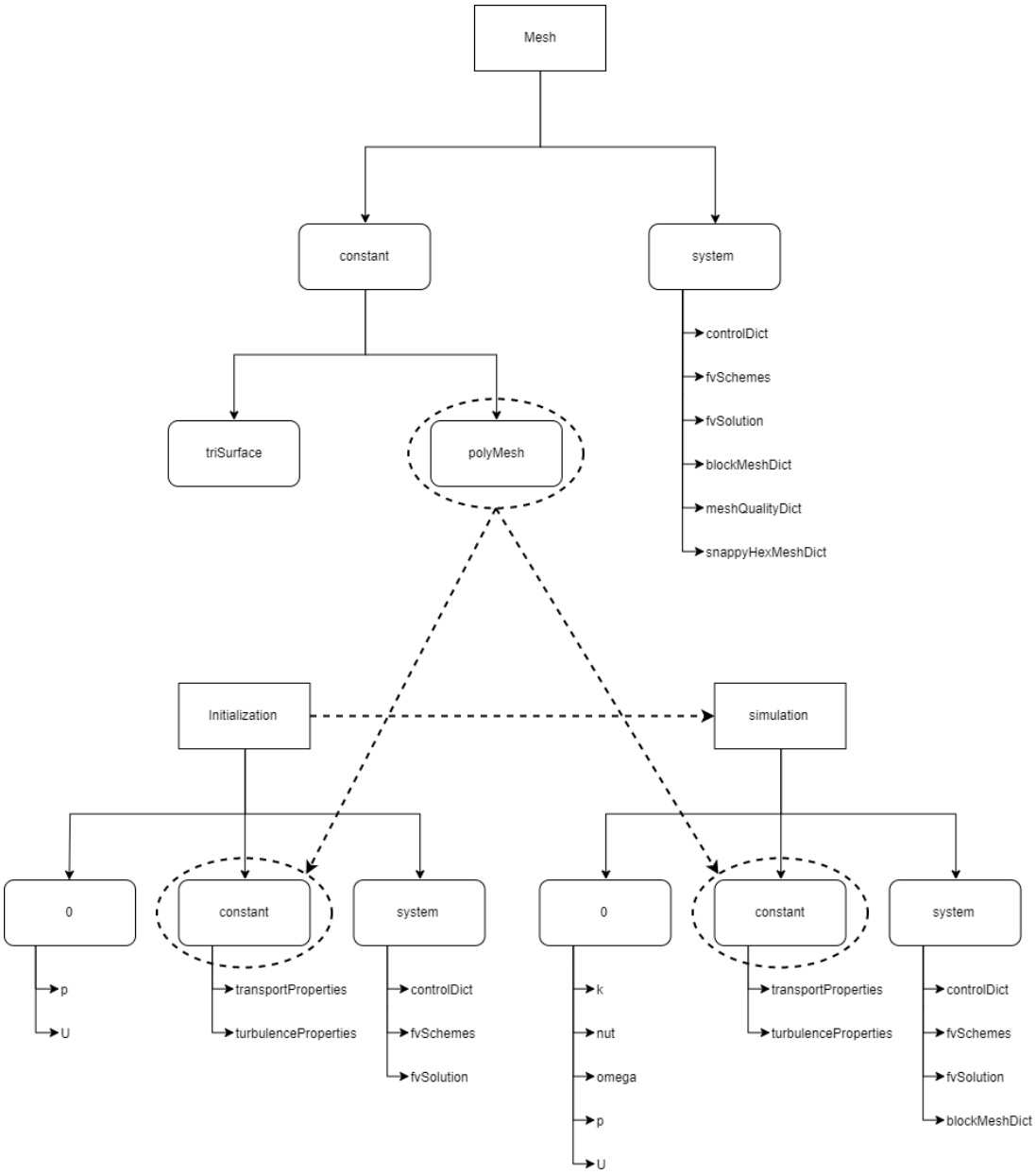


Figure 3.4: Case configuration.

3.3 Numerical Validation Setup

The code validation methodology can be seen in Figure 3.5. Primarily, the geometry to be analysed is drawn and the computational domain defined. The next step is to create the mesh and validate it through a mesh independence study. Afterward, the setup model is prepared and the simulation can be started. In the end, the results obtained are compared with experimental studies presented in Section 2.8. If there is a good agreement between both, one can draw some conclusions, otherwise changes must be made to the setup model. This procedure is repeated until the appropriate setup model is found.

3.3.1 Geometry Validation and Flow Domain Definition

By comparison with the OBS, validation geometry has been simplified to a cylindrical configuration. To perform code validation, four aspect ratios will be used ($\frac{H}{d} = 2, 3, 5$ and 10). The geometry was designed using CATIA V5R20 [60] and a tessellation has been given to the geometry (depicted in Figure 3.6). After geometry drawing, the STL (Stereolithography) is stored in the *mesh/constant/triSurface* directory.

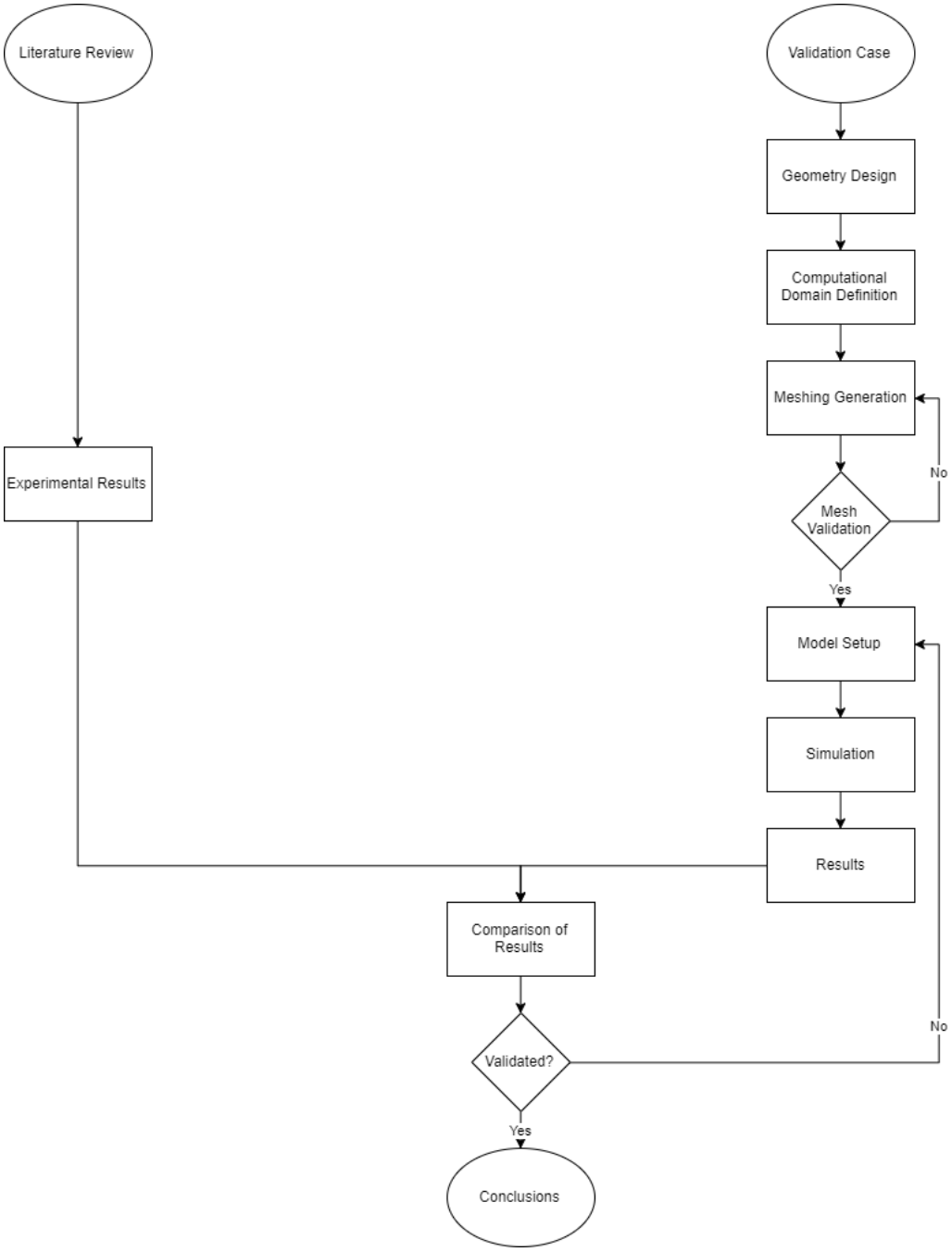


Figure 3.5: Validation process.

The flow domain is defined by a rectangular box made from *blockMesh* command (Figure 3.7). The control volume will cover the fluid surrounding the geometry. The boundaries (inlet, outlet, right, left, bot and top) of the domain will be used later to define the boundary conditions.

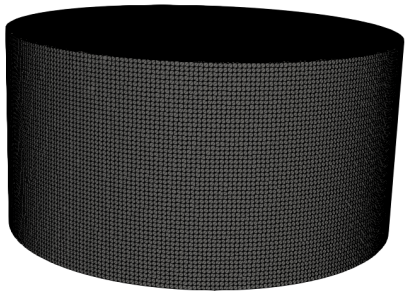


Figure 3.6: Cylinder base refinement.

The reference axis system was fixed on the center of the cylinder, as can be seen in Figure 3.7, where the x axis represents the direction of the fluid in the control volume.

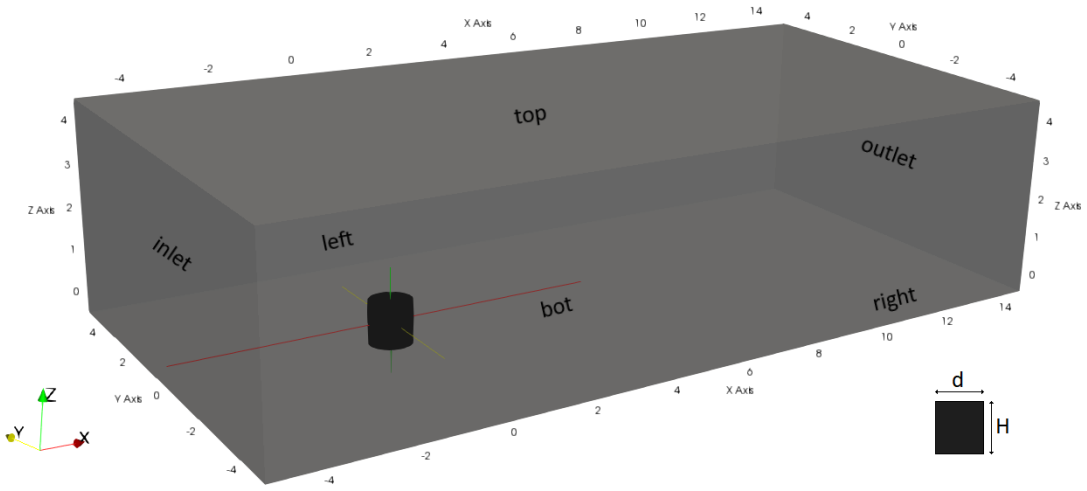


Figure 3.7: Control volume representation.

The domain's dimensions, directly related to the geometry, were defined accordingly the International Towing Tank Conference in Table 3.2. Such dimensions should not be too large due to the computational effort, but neither can be too small so as not to lose physical phenomena that are occurring around fluid domain [61].

Table 3.1: Flow domain dimensions.

Boundary	Distance [m]
inlet	$5d$
outlet	$15d$
right	$5d$
left	$5d$
bot	$-$
top	$5H$

3.3.2 Mesh Generation

In order to analyse the geometry using numerical methods, the domain should be discretized, or subdivided, into finite volumes. It will assume that the flow properties are spatially constant, which limits the resolution of flow properties gradients to the control volume discretization resolution.

The number of elements into which the domain is subdivided is directly connected with computational effort. So, a trade-off must be done in order to minimize computational effort without losing accuracy.

For this purpose, two OpenFOAM tools were used, *blockMesh* and *snappyHexMesh*, for mesh generation. The first one discretize only the control volume, while the second one relates the refinement given to control volume with the considered geometry. This latter step gives a greater refinement over the desired zones, as well as control over inflation layers generation. Below a brief explanation of these tools is presented.

3.3.2.1 blockMesh

The *blockMesh* tool creates the control volume and decompose it into $3D$ hexahedral blocks. The mesh is generated through the *blockMeshDict* file located in *mesh*/*system*/*blockMeshDict*. The *blockMesh* tool reads the *blockMeshDict* file, generating the *polyMesh* folder, where it can be found all the mesh data (points, faces, cells and borders) [16].

3.3.2.2 snappyHexMesh

The *snappyHexMesh* tool automatically generates a $3D$ mesh containing, mostly, hexahedral and split-hexahedral blocks from triangulated surface geometries, typically in STL or Wavefront Object (OBJ) format (which, as mentioned before, are located at *mesh*/*constant*/*triSurface*). All steps of this tool are outlined in the *snappyHexMeshDict* file, which is in *mesh*/*system*/*snappyHexMeshDict*. This command can only be used after the base mesh has already been created [16]. The *snappyHexMeshDict* file is subdivided into three steps:

- 1^{st} **step: castellatedMeshControls:** In this sub-directory cell division is performed according to user specifications. Entries for this sub-directory are in Appendix A.1 in the *snappyHexMeshDict* file. Once the feature and surface splitting is complete, the cell removal operation ca be initiated. For this operation it is necessary to identify a point (locationInMesh), which will decide whether if the mesh is kept inside or outside that region/geometry. Lastly, a refinement is made in regions specified by the user, where it can be squares, rectangles or spheres. Here, refinements zones are created according to what is intend to study, thus using different levels of refinement depending on the phenomena to be studied (e.g. refinement level 1 means that the base mesh cell is subdivided into 4 of the same size and so on). This process is illustrated in Figure 3.8;

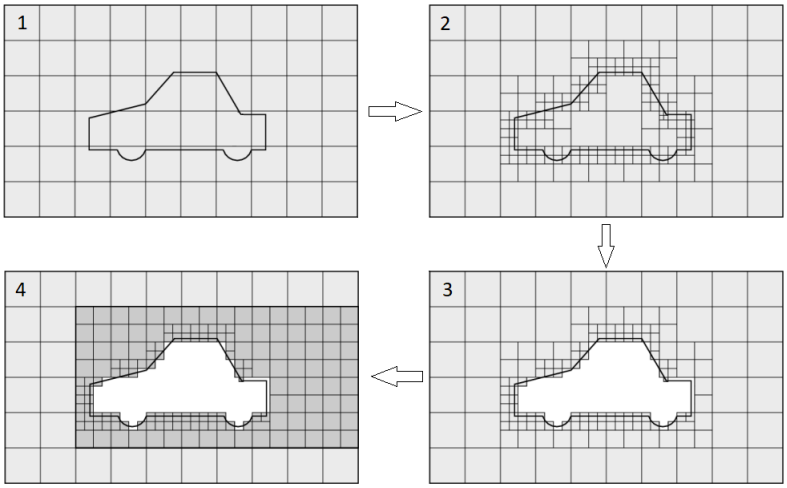


Figure 3.8: CastellatedMeshControls process, adapted from [16].

- 2^{nd} **step: snapControls:** In this sub-directory, cell vertex points will be relocated onto surface geometry to eliminate the jagged castellated surface from the mesh, as depicted in Figure 3.9;

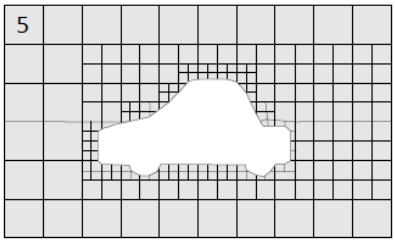


Figure 3.9: Surface snapping, adapted from [16].

- 3^{rd} **step: addLayersControls:** After snapping step, the mesh may contain some irregular cells along the control geometry, so this third step is optional. Additional layers of hexahedral cells aligned with the geometry surface are introduced in this step (Figure 3.10). This step is essential when studying turbulent flows, as it is of interest to study the boundary layer most effectively and correctly. After the mesh has been changed, its quality is controlled by the file *meshQualityDict* present in the *system* folder[16].

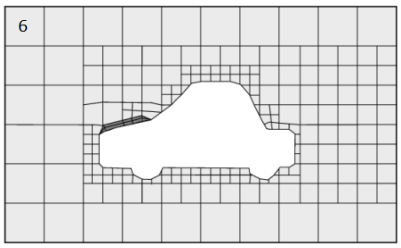


Figure 3.10: Layer addition, adapted from [16].

When the *snappyHexMesh* command is completed, the mesh is generated (Figure 3.11), and the *polyMesh* folder can be moved for both initialization and simulation constant directory (Figure 3.4).

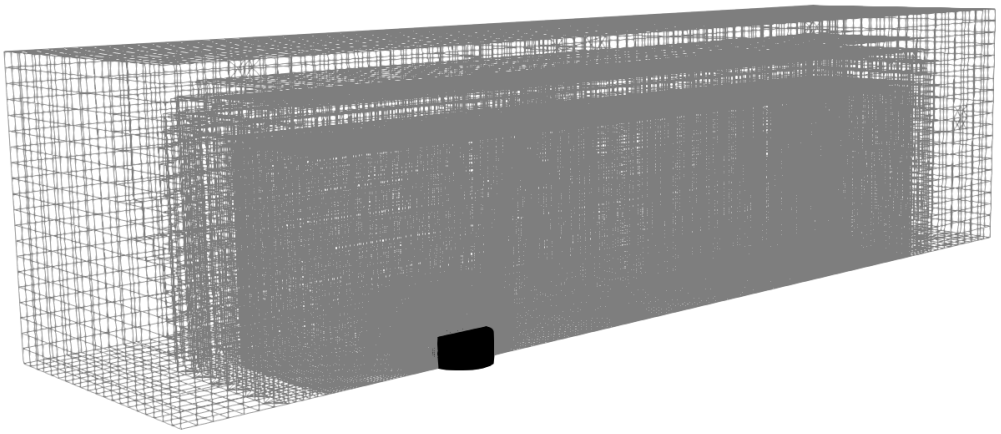


Figure 3.11: Mesh wireframe.

3.3.3 Physical Model Setup

Based on literature review, in subsections 2.6 and 2.7, and comparing with the state-of-the-art, RANS equations, with the turbulent model SST κ-ω, were the most suitable compared to the others. The choice was made considering a trade-off between computational power and accuracy in the results obtained. Zhang [57] concluded that among the eight different RANS turbulence models that he studied, the SST κ-ω model was the one that obtained the best results, being even close to the results obtained with LES. This last one, was not select due to the computational cost it needed.

In OpenFOAM, the modeling approaches for turbulence are introduced to the file present in *initialization/constant/turbulenceProperties* and *simulation/constant/turbulenceProperties* (Appendix A.1). With the addition of this turbulent model, it is necessary to add other empirical equations for κ, ω and nut.

For the transport model, the Newtonian model was selected, where it was assumed that kinematic viscosity is constant. Viscosity is specified in *initialization/constant/transportProperties* or *simulation/constant/transportProperties* in Appendix A.1.

3.3.3.1 Near-Wall Influence

There are certain parameters to be aware of to better understand the phenomena near the walls, such as:

$$y^+ = \frac{\triangle y \times u_\tau}{\nu} \tag{3.1}$$

where,

$$u_\tau = \sqrt{\frac{\tau_w}{\rho}} \tag{3.2}$$

In equations 3.1 and 3.2, u_τ is the friction velocity, $\triangle y$ is the first layer thickness, y^+ is a di-

mensionless measure of wall distance and τ_w is wall shear stress.

With regarding to boundary layer treatment, y^+ plays a crucial role. The subdivision of the near wall region into a turbulent boundary layer can be described as (Figure 3.12)[62]:

- Viscous sublayer region ($y^+ < 5$): Wall shear is dominated by viscous stress and the velocity profile is presumed to be laminar;
- Buffer region ($5 < y^+ < 30$): where both viscous and turbulent shear dominate;
- Log-law region ($30 < y^+ < 300$): turbulent shear prevails;
- Outer region ($y^+ > 300$): where the flow no longer has the wall effect.

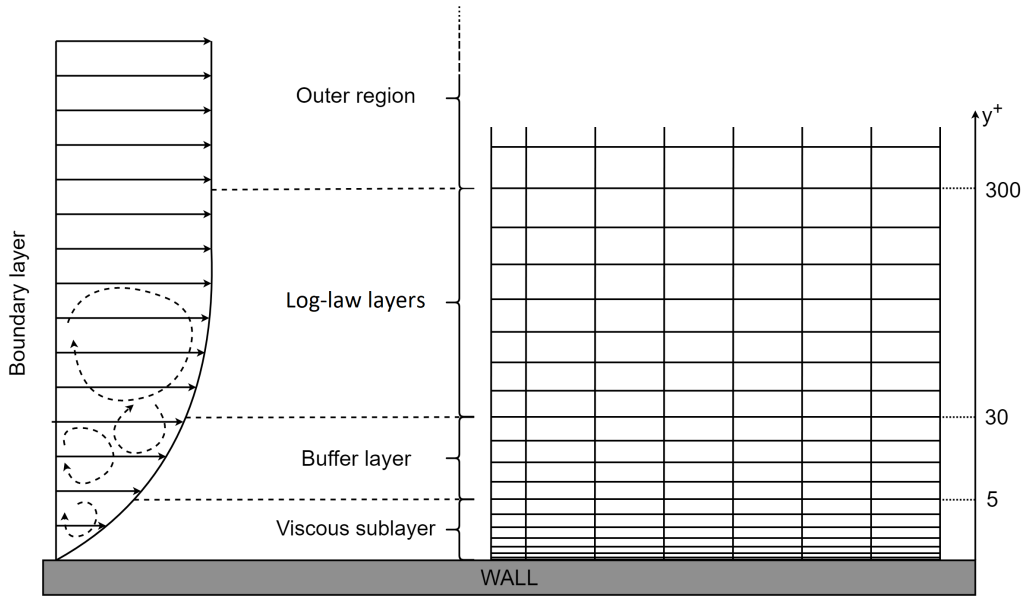


Figure 3.12: Boundary layer subdivision.

3.3.3.2 Wall Function Method

For wall integration in turbulence models, using SST κ-ω, it is essential that the first computational cell above the wall be located below the viscous sublayer region ($y^+ < 5$) [63]. The wall function introduction enables to satisfy the fluid physics near the wall without the need of a high computational power. This consists in two major advantages: it saves storage and computing time. Implementation of wall functions must be implemented without significant loss of accuracy [63] [64].

3.3.4 Boundary Conditions

Boundary conditions have to be as realistic as possible, since they represent all the physics involved in the problem. The simulation is started with these conditions, and once unrealistic, the simulation loss credibility. This information is located in *initialization/0* and *simulation/0* (Appendix A.1) and are summarized in the Table 3.2:

Table 3.2: Simulation boundary conditions.

	k	nut	omega	p	U
inlet	fixedValue	calculated	fixedValue	zeroGradient	fixedValue
outlet	zeroGradient	calculated	zeroGradient	fixedValue	zeroGradient
right	zeroGradient	zeroGradient	zeroGradient	zeroGradient	zeroGradient
left	zeroGradient	zeroGradient	zeroGradient	zeroGradient	zeroGradient
top	zeroGradient	zeroGradient	zeroGradient	zeroGradient	zeroGradient
bot	kqRWallFunction	calculated	omegaWallFunction	zeroGradient	fixedValue
cylinder	kqRWallFunction	calculated	omegaWallFunction	zeroGradient	fixedValue

3.3.5 Temporal Discretization

Solver selection is considered a crucial step before performing the analysis, and once an unsteady incompressible flow is presented, two OpenFOAM solvers stand out, pisofoam and pimplefoam. The choice of the solver, pimplefoam, was based on the state-of-the-art and the fact that this solver converges with stability for longer time steps. In order to predict the maximum time step value, it was follow general CFD guidelines, where it is advised that the time step should not exceed $\frac{1}{100}\frac{d}{U}$ [61]. This time step value does not guarantee convergence, as the mesh may not support such a high time step. Consequently, OpenFOAM provides a function called Courant Number, that will control the time step in order to obtain a temporal accuracy and numerical stability.

3.3.5.1 Courant-Friedrichs-Lewyor Condition

According to Laney: "The full numerical domain of dependence must contain the physical domain of dependence." This is called Courant-Friedrichs-Lewyor (CFL) condition (Figure 3.13). CFL condition can not be transgressed in order to not lose information that may affect the true solution [17].

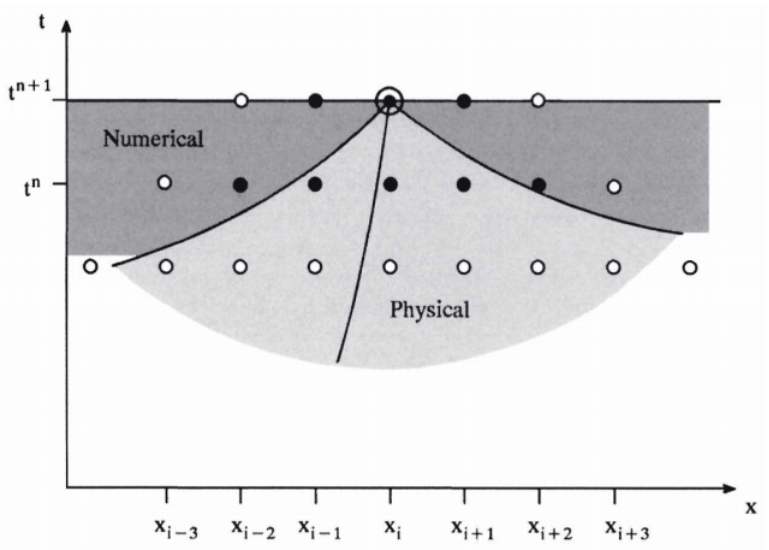


Figure 3.13: Numerical vs. physical domain of dependence, adapted from [17].

With the purpose to fulfill this condition, it must be ensured that the Courant number (Co) is less than one in the whole domain. For a cell Co is defined as:

$$Co = \frac{\delta t |U|}{\delta x} \tag{3.3}$$

where δt is the time step, $|U|$ is the velocity magnitude through that cell and δx is the cell size in the velocity direction.

The time step is controlled, in the *controlDict* file, by limiting the Co to a fixed value. In most of the unsteady analyses, the first time-steps are not trustworthy, so it can be used a higher value on the Co. In the sample interval, Co must be bellow the value of one unit. This way, it will be faster to converge without losing information that affects the solution.

The CFL condition is necessary when a sample is to be taken, but not enough to have stability. For a stable simulation, a good combination of schemes, solvers and time-step have to be selected. These files (*fvSchemes*, *fvSolutions* and *controlDict*) are in Appendix A.1, where it can be seen which schemes and solutions were selected in order to obtain a stable and accurate simulation.

3.4 Simulation

As outlined in subsection 3.2.1, the case study is divided into three steps: meshing; initialization and simulation. The commands used in each step are in Table 3.3, 3.4 and 3.5. All commands explanation can be found in [16]. OpenFOAM version 5.x was used to perform the simulations.

Table 3.3: Meshing commands.

Order	Command
1	foamCleanTutorials
2	blockMesh
3	surfaceFeatureExtract
4	snappyHexMesh -overwrite
5	checkMesh -allGeometry -allTopology

Table 3.4: Initialization commands.

Order	Command
1	foamCleanTutorials
2	rm -rf 0
3	cp -r 0_org 0
4	cp -r ../mesh/contant/polyMesh constant
5	potentialFoam -initialiseUBCs -noFunctionObjects

Residuals can be visualized while the simulation is running, as well as check force components behavior, or even, perform simulation changes. The simulation convergence was controlled by the generation of two documents,*forcesgnuplot* and *Residuals* (Appendix B.1), which allowed to follow simulation convergence. .

Table 3.5: Simulation commands.

Order	Command
1	foamCleanTutorials
2	rm -rf 0
3	cp -r 0_org 0
4	cp -r ../mesh/contant/polyMesh constant
5	mapFields ../initialization -consistent -noFunctionObjects -mapMethod cellPointInterpolate -sourceTime latestTime
6	renumberMesh -overwrite
7	decomposePar
8	foamJob -parallel -screen pimpleFoam

3.5 Mesh Independence Study

The study of mesh independence is fundamental in a CFD work, since it is possible to find the best trade-off between mesh quality and computational effort. For this study, the infinite cylinder was considered, for a lower computational cost/effort. The study is presented in Figure 3.14 and summarized in Table 3.6, where it is compared to the theoretical value (represented by the solid line) of $2D$ cylinder (infinite cylinder) for the same range of Reynolds number [65].

	Cells	Strouhal number
Theoretical 2D cylinder [48]	-	0.200000
Present study of a 2D cylinder	227064	0.248312
	244728	0.240930
	252468	0.287973
	498712	0.244260
	706300	0.209441
	1004668	0.198044
	1639764	0.209807

Table 3.6: Mesh convergence results.

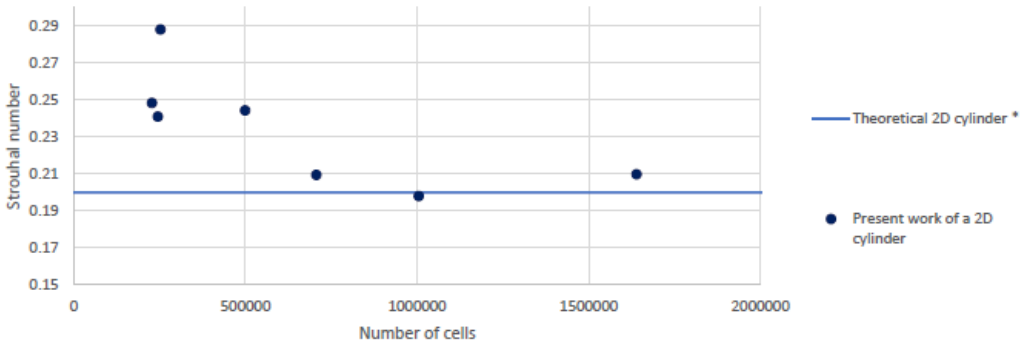


Figure 3.14: Mesh convergence results.

As can be seen from Figure 3.14, there are a few simulations that approach the theoretical value. The mesh with around 700000 cells will be chosen, as the objective of this study is to have a lower computational effort with small errors.

The purpose of this work, as described in Chapter 1, is to find the frequency induced by the flow in the geometry to be studied. As the simulations were elaborated in time regime, a problem arises: to change from time regime to frequency regime. For this, the FFT (Fast Fourier Trans-

form) is followed in order to move to frequency regime and thus achieve the Strouhal number associated to the simulation, with the help of Equation 2.3. An example of the whole process can be found in Appendix C.1.

3.6 Numerical Validation

In order to validate the code used, four different aspect ratios were analysed, $\frac{H}{d} = 2, 3, 5$ and 10, with the aim of validate the code. The results were compared with the results obtained in the state-of-the-art, (Table 3.7) described in Section 2.8, as illustrated in Figure 3.15.

Table 3.7: Authors' references.

Kawamura (1984) [52]	Okamoto (1992) [55]	Okamoto (1973) [51]	Park (2000) [56]
Sakamoto (1983) [53]	Sakamoto (1984) [54]	Wang (2012) [50]	Zdravkovich [66]

The results of the present work, illustrated in Figure 3.15, exhibit a maximum error of 15% when compared to Zdravkovich's study [66].

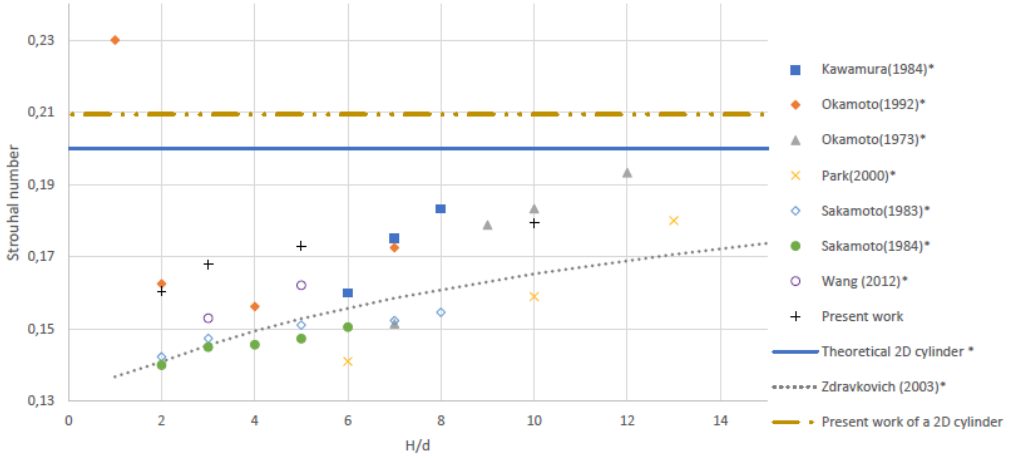


Figure 3.15: Validation results.

Chapter 4

Case Study: OBS Shell

In this chapter, assumptions and requirements adopted for this study will be described, and the OBS conceptual configuration will be presented. Therefore, analysis made to the conceptual configuration and respective results will be here discussed. Afterwards, new concepts will be presented, resulting from modifications to the conceptual configuration, aiming to reduce the frequency induced in the analysed geometry, and additionally the results are compared with the conceptual geometry. In the end, the configuration is selected according to the results, as illustrated in Figure 4.1.

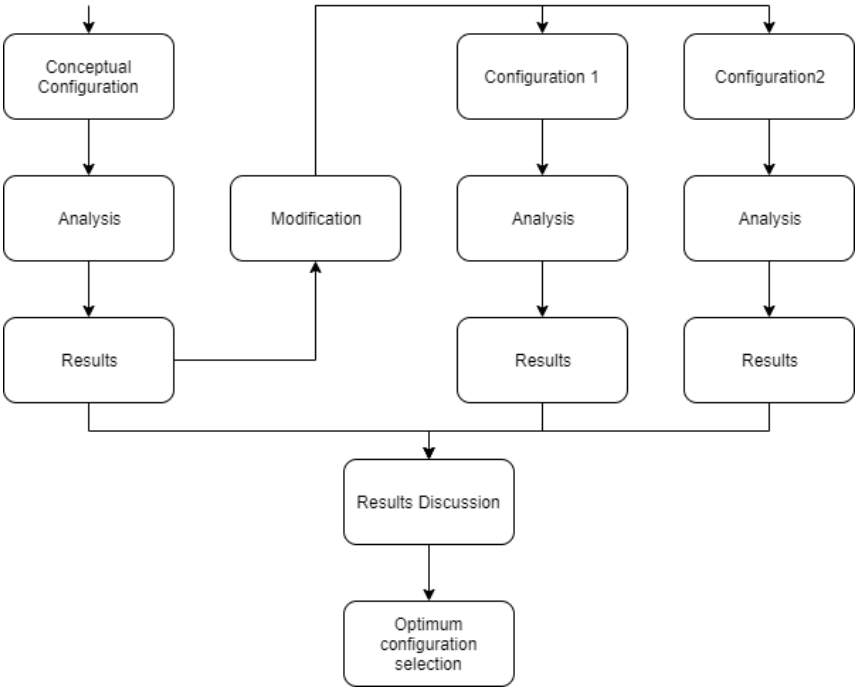


Figure 4.1: Case study configuration.

Considering the operational requirements given by DUNE project, which were considered from the point of view of conceptual design as requirements for the OBS, the main goal of this system is to operate without flow interference (noise), or as little interference as possible. Thereby, when frequency is identified, if it is not possible to remove this interference, it must be ensured that it is not interfering with the results measured seismic frequency interval. The requirements are summarized in Table 4.1.
During this study, some assumptions were made, such as considering:

- the geometries were stationary during the analysis, i.e. zero Degrees of Freedom (DoF) ;
- the flow regime as unsteady;
- the fluid as fresh water in the liquid state at $20°C$, and remaining its properties constant throughout the simulation (Table 4.2) [67].

Table 4.1: Mission requirement.

Requirements	
U_{max}	0.08 [m/s]
Frequency range from OBS	$0.01 - 40.00$ [Hz]
OBS diameter	1.00 [m]
OBS height	0.50 [m]

Table 4.2: Fluid properties.

Density	998.2 [kg/m^3]
Kinematic viscosity	1.0×10^{-6} [m^2/s]

4.1 OBS Shell Conceptual Design

In order to simplify these analyses, all small cavities of the the OBS shell, e.g, cavities for antennas, sensors or eyelets, were suppressed. These simplifications were necessary, due to the computational effort of the unsteady simulations. The OBS shell was, then, reduced to a cylinder with aspect ratio of 0.5, as can be seen in Figure 4.2.

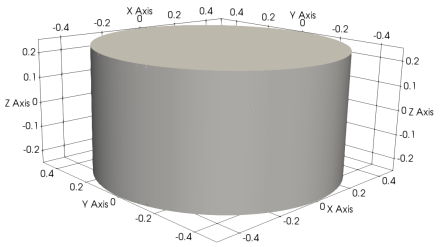


Figure 4.2: OBS shell.

4.1.1 Results

Once the geometry was prepared, the required input parameters were set to start the simulation. This step was carried out according to the validated model in Chapter 3, as it is described in Figure 4.3, where the Lift Coefficient (C_L) is shown in function of the time domain. From this figure, it is possible to observe that, initially, the C_L has a random behavior and, after several time-steps, it starts to have an oscillatory attitude. This behavior becomes continuous throughout time.
Considering the oscillatory behavior, the results were analysed and the frequency was measured as outlined in Appendix C.1. The final outcome is illustrated in Figure 4.4, where results in frequency domain can be seen (Frequency vs. Magnitude). Since the OBS working frequency is 0.01 to 40 [Hz], and given that the highest frequency found was 0.0057 [Hz], it could be concluded that it is outside the OBS working range. However, in order to minimize this value or even try to eliminate it, some modifications are suggested.

Figure 4.5 represents the velocity streamlines around the conceptual configuration, where it is possible to notice the different phenomena taking place. Three different types of vortices were detected. Firstly, ahead of the geometry, the floor presence generates a horseshoe vortex,

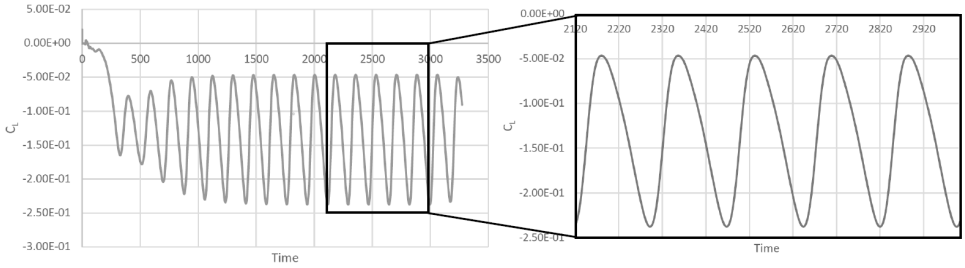


Figure 4.3: C_L as a function of time of conceptual design.

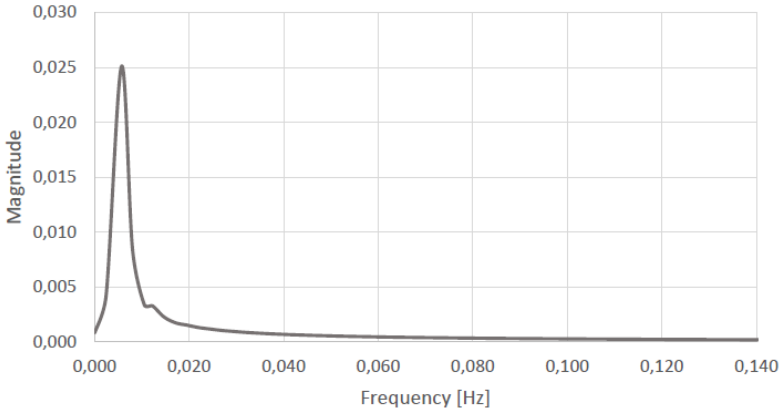


Figure 4.4: FFT of conceptual design.

which extends across the wake. This vortex is common in all walled flows and has a horseshoe-like feature, thus giving rise to its name (Figure 4.5). At the top, the flow separates from the cylinder and reattaches further, creating a new vortex. This vortex has more influence on the results as the aspect ratio of the cylinder decreases. At last, behind the cylinder, two symmetrical vortices are formed. These are the most common vortices in flows around cylindrical structures and the ones that most influence have in the results.

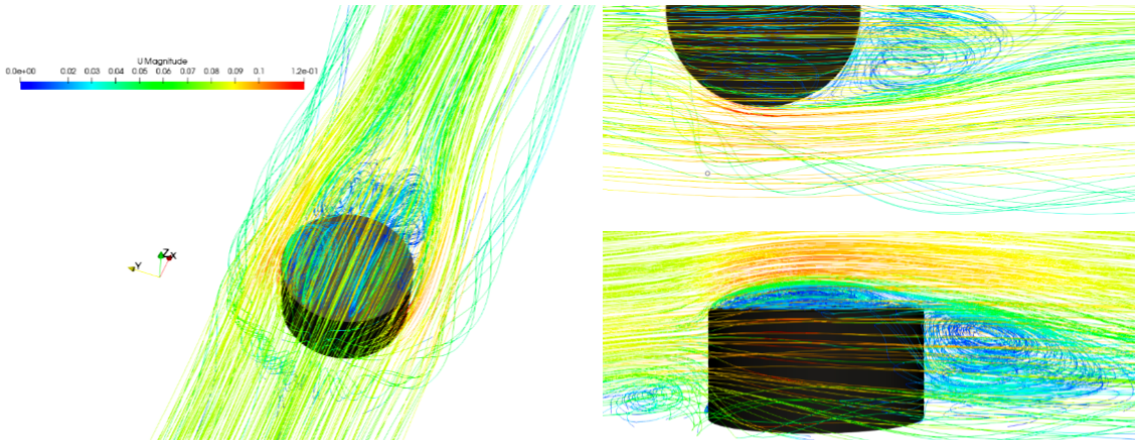


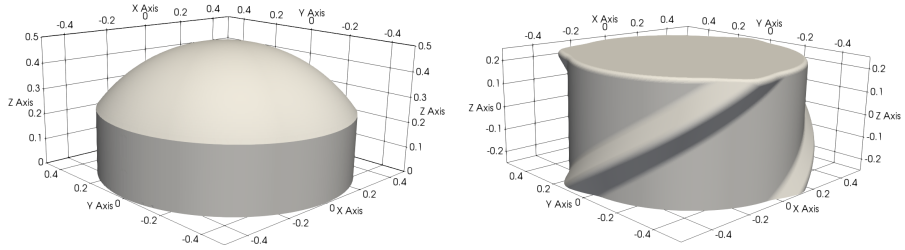
Figure 4.5: Three dimensional flow around conceptual design. Close view of horseshoe vortex.

From this analysis, some modifications were suggested in order to improve the results of the OBS shell configuration. The goal was to eliminate the free end effect and the wake created by the flow after passing through the body. In this regard, two geometries were considered, one more focused to reducing the free end effect, and another more focused to reducing the wake generated by the flow. These improvements have been proposed in these two phenomena, since

the third, the effect of the ground, is not possible to eliminate. The only way to reduce the effect of the ground was by changing the circular shape. This was not permitted due to design requirements and due to the implications it would cause, such as induced different frequencies depending on the flow direction.

4.2 OBS Shell Geometries

Due to design constraints at both operation and logistics levels, general dimensions previously defined must be respected. Thereby, in Figure 4.6, the two design improvement suggestions are presented. In the first case, Figure 4.6a, the purpose was to remove the disturbances caused by free-end exposed to the flow. To solve this issue, a spherical hubcap was placed on the top of the cylinder. Improvements are predicted, mostly on the top of the configuration, since no recirculation region is expected with this modification. Thus, it will be possible to reduce the free end effect in the results and reduce the wake length. In the second case, Figure 4.6b, the focus was to reduce the vortices observed in the cylinder wake and, consequently, the associated frequency, by aplying three helical strakes in the original geometry. This concept is also used in [68], where improvements were achieved in the results.



(a) First OBS shell iteration, cylinder with spherical hubcap.

(b) Second OBS shell iteration, cylinder with three helical strakes.

Figure 4.6: OBS shell geometries.

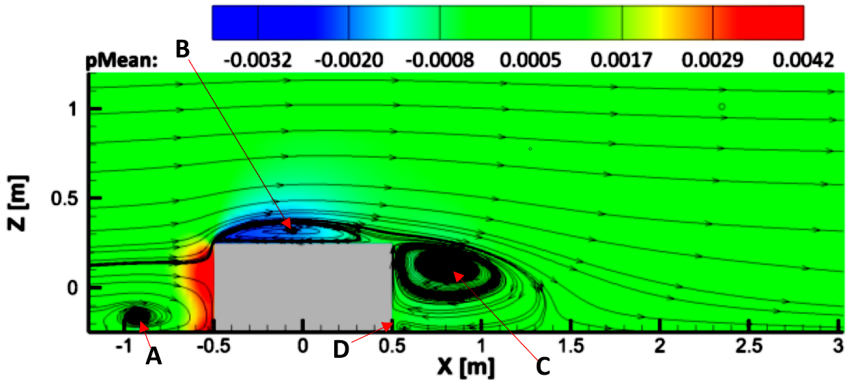
4.2.1 Results

Both geometries followed the same procedure as applied to the conceptual geometry, and as it is shown in Chapter 3. The cylinder with a spherical hubcap design, illustrated in Figure 4.6a, allowed to reduce the frequency induced by flow. This reduction reflected improvements around 60% in the induced frequency (from 0.0057 [Hz] to 0.0023 [Hz]). On the other hand, the cylinder with three helical strakes, presented in Figure 4.6b, showed no improvement in results, contrary to expected, since the flow-induced frequency has increased. The induced frequency increased 300% compared to the original geometry (from 0.0057 [Hz] to 0.0230 [Hz]). This frequency increment may be associated with the fact that the helical strakes were added to a coin-like cylinder (cylinder with low aspect-ratio), or even to the different dimensions given to helical strakes, when compared to those given in study [68]. Thereby, the helical strakes do not have a good performance when it comes to cylinders with low aspect ratios. Therefore, the results may be directly related to these dimensions, where better results would be expected for larger aspect ratios.

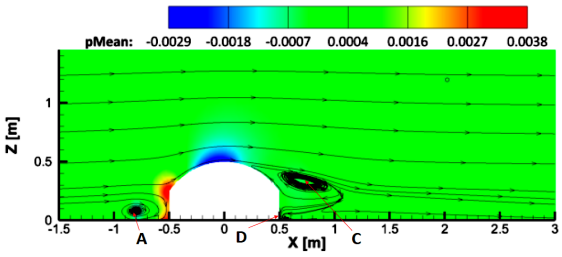
Consequently, just the cylinder configuration with the spherical hubcap was suitable as a improvement for the conceptual design. In Appendix D.1 it is possible to notice different three-dimensional perspectives of the mean flow for both geometries, where it is possible to identify all the phenomena caused by the flow.

4.3 Discussion

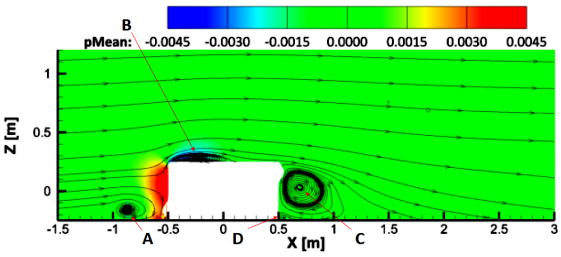
In this section, the mean pressure field with velocity streamlines for different planes is presented. In Figure 4.7 it is possible to identify the phenomena when the flow passes through the body. In general, it is possible to observe the effect of the floor presence, when the flow passes through any of the three geometries. This effect results in the creation of a vortex horseshoe, denoted by the letter A in Figure 4.7. In Figure 4.7a, an abrupt variation of geometry generates a flow separation from the leading edge of the cylinder top and a reattachment a bit further, denoted by the letter B. Consequently, an isolated recirculation region is formed. This recirculation region is also noted in the cylinder with helical strakes, but not with the same strength. The modification to a spherical hubcap worked well in the elimination of this recirculation region, since there is no separation of the flow until the rear of the configuration. In the near wake behind the three geometries two symmetrical vortices coexist, one clockwise (caused by the downwash flow and denoted by letter C) and another anti-clockwise (induced by a very weak upwash flow close to the bottom wall and denoted by letter D). At last, it is of significant important the flow field above the free end, once the curvature of the streamlines within this region is related to the angle of the downwash flow behind the configuration.



(a) Cylinder shape.



(b) Spherical hubcap.

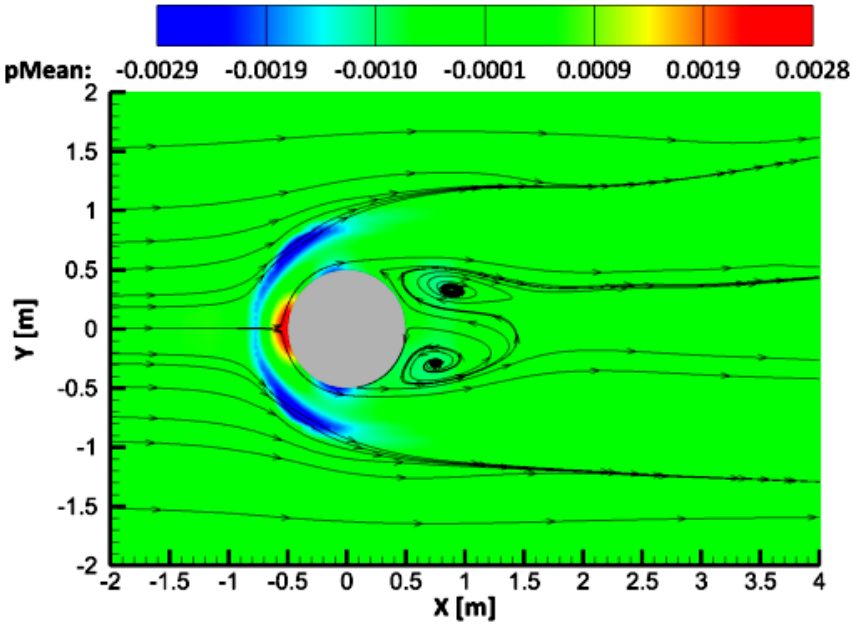


(c) Cylinder with helical strakes.

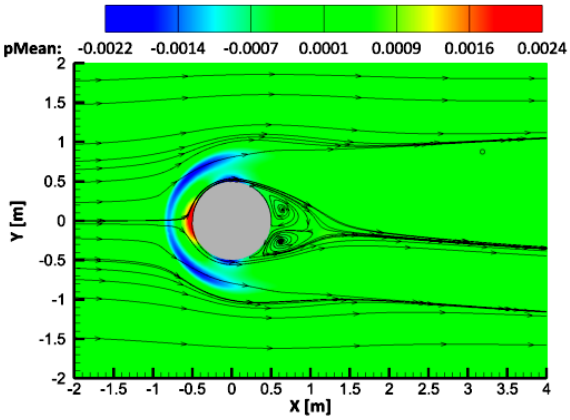
Figure 4.7: Relation of the mean streamlines and pressure contours on the plane $Y = 0$.

In Figure 4.8, another perspective from the ground effect is witnessed for the three configurations, where it can be observed a Z-plane view equal to 20% of the cylinder height from the

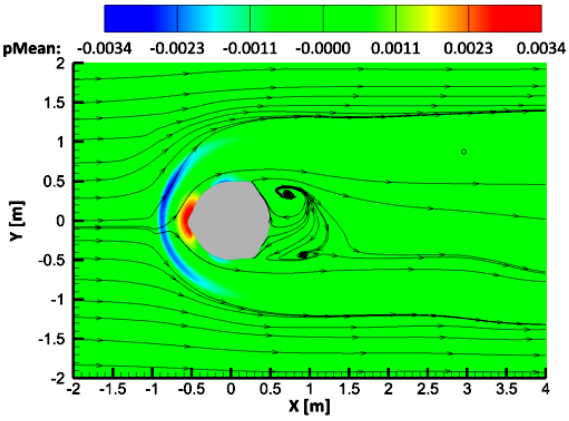
bottom. The ground effect makes the horseshoe vortex to extend along the wake. Improvements are noted in the spherical hubcap, as the flow separates than the other two, making the flow to reattached in the wake. As the flow crosses the body, two symmetrical vortices appear.



(a) Cylinder shape.



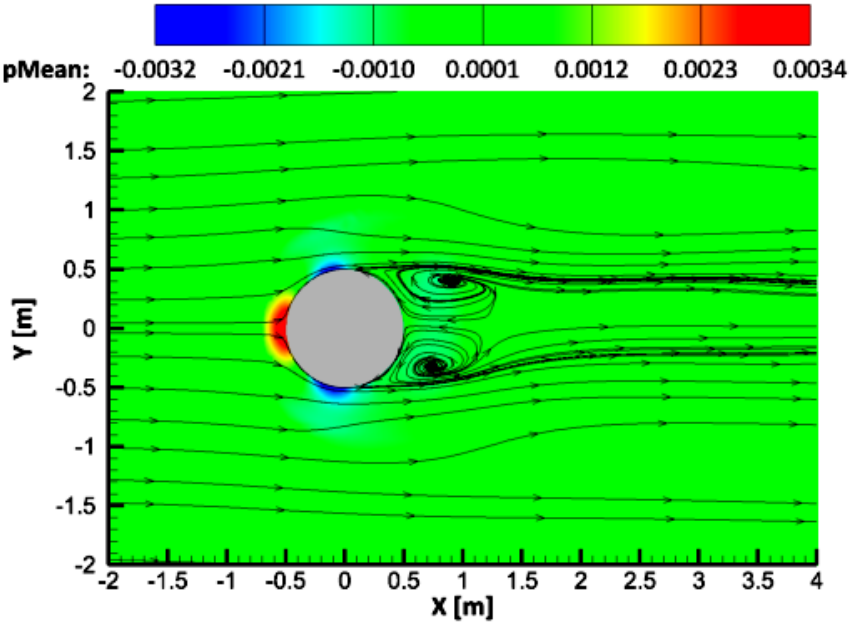
(b) Spherical hubcap.



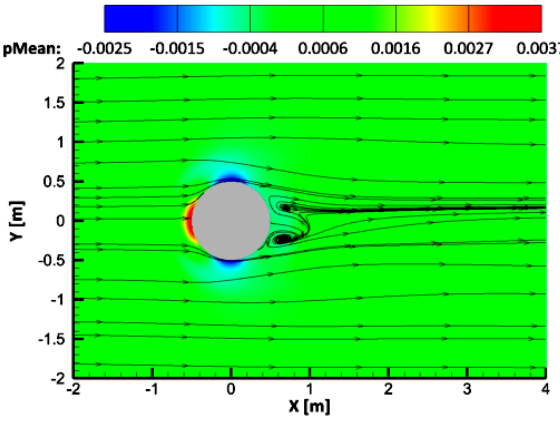
(c) Cylinder with helical strakes.

Figure 4.8: Relation of the mean streamlines and pressure contours on the $Z = 20\%H$ from the bottom plane.

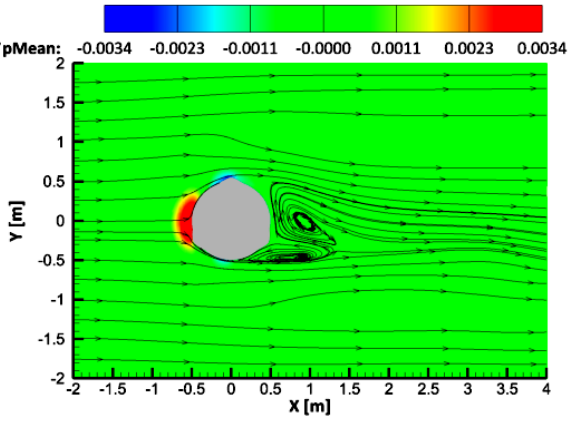
Lastly, the mid height plane (Z equal to 50% of the cylinder height from the bottom) is analysed in Figure 4.9. It is noticed that the horseshoe vortex is almost imperceptible at this height. On the other hand, the obstruction effect of the geometry generates a strong positive/negative pressure region before/behind the geometry. Thus, two symmetrical vortices are well developed behind, where once more the spherical hubcap has the latter separation.



(a) Cylinder shape.



(b) Spherical hubcap.



(c) Cylinder with helical strakes.

Figure 4.9: Relation of the mean streamlines and pressure contours on the $Z = 50\%H$ from the bottom plane.

Chapter 5

Conclusions

This chapter summarizes all the choices/conclusions that have been made throughout the work. During this project, CFD unsteady analyses were user to determine how the flow-induced frequency varies in different geometries. In order to achieve this goal a free/open-source software, OpenFOAM, was used. In the first place, studies found in the literature regarding to the flows around the cylindrical bodies were analysed, as well as, the flow-induced frequency influence on the cylinder aspect ratio.

An investigation was made into OpenFOAM software, where, through tutorials and forums, the necessary knowledge to use the software was acquired. After this investigation, a mesh independence study was performed with the SST κ-ω turbulence model, where this model proved to be suitable for this type of studies. After this, a code validation was performed, where different aspect ratios were studied and the results were compared with the experimental data available in the literature, showing a good agreement.

Then, following the general CFD guidelines, case study analyses were carried out. Thus, conceptual shell geometry of the OBS was analysed, and several phenomena were observed. Among these phenomena, the principal were a recirculatory region in the free end of the cylinder, vortices were noted in the cylinders wake and, lastly, the presence of the floor generated horseshoe vortex. Two different geometries were suggested in order to solve two of these three phenomena observed, since design requirements do not allow changes in order to avoid the horseshoe vortex. The first one, cylinder with spherical hubcap, has a main goal of eliminating the recirculatory region, while the second, cylinder with three helical strakes has the purpose of reduce the vortices near the rear of the cylinder.

5.1 Difficulties

Since OpenFOAM was initially unknown by the author of the dissertation, many challenges were encountered throughout the work and many hours were spent solving them. Although the fact that OpenFOAM has many users, a large database and tutorials, adapting to this software was not an easy task.

The main challenges experienced were the study of mesh independence and the code validation. In the first case, being a key step in CFD, the process to achieve a good mesh can be very long and exhausting. This is due to the fact that there are many parameters to define, which can be handled in different ways to obtain the mesh convergence/independence. Regarding to the code validation, many other factors appear that may influence the results of the analysis. Since the choice of the solver, to choosing the most advisable schemes and solutions, the user has many possible combinations, with stability and accuracy varying between them. Meaning that, several weeks or even months are required for basic software knowledge.

Since the present regime is unsteady and subcritical, the numerical forecast of the flow is very unpredictable, increasing the difficulty of numerical analysis. The numerical simulations were dependent on the available computational effort and, being in unsteady regime, each simulation took several days.

5.2 Results

Firstly, the mesh independence was carried out, where SST κ-ω turbulence model with around 700000 elements was the mesh with the better trade-off between computational effort and accuracy. The mesh was then adapted to the case study.

In order to perform the code validation, the results were compared with theoretical and experimental data available in the literature, changing the aspect ratio of the cylinder. Regarding the numerical validation study, this has a behavior that is common to almost all other studies: a decrease in Strouhal number with decreasing aspect ratio. This decrease, according to Zdravkovich, happens due to the free end downwash flow [66]. Although there are several errors associated with this comparison, such as errors associated with scale, Reynolds number similarity or even numerical predictions, results demonstrated good agreement with the literature.

Analysing the concept geometry, it was concluded that the flow was inducing a frequency in the OBS body, causing it to wobble at a frequency of 0.0057 [Hz]. Thus, in order to present improvements, either in the recirculation region or in the vortices present in the wake of the conceptual configuration, two alternative design solutions were proposed. The first concept that was proposed, was a cylinder with a spherical hubcap at the top, once there was a need to eliminate the recirculatory region from the top of the conceptual configuration. This modification in the geometry allowed to achieve a frequency reduction of 60%. This improvement was only possible due to the modification of the flow around geometry, thus making the design more hydrodynamic. It was observed that, with this modification, the flow separates later from the geometry, relatively to the conceptual geometry and, consequently, causes the flow to be more streamlined at the wake of the cylinder. Regarding to the second geometry, a cylinder with three helical strakes, purposed to reduce the vortices generated near the wake of the cylinder, this had a worse behavior than the conceptual one. In this case, the frequency induced by the flow increased 300% (0.023 [Hz]). When the geometry was proposed, this was not the expected result, but it can be explained once the examples observed in the literature were related to geometries with bigger aspect ratios.

With these analyses, it was possible to observe all the expected phenomena in a flow around a body mounted in a wall. It was possible to see the vortices generated by the free-end flow separation and reattachment, the vortices generated at the rear of the body, as well as, the creation of the horseshoe vortex, because of the floor presence.

Therefore, comparing both the results of the analyses and the requirements initially given, consequently, the best choice found was the cylinder with a spherical hubcap at the top.

5.3 Future Work

The work carried out during this dissertation can be further continued in several aspects. Future work is suggested as follows:

- A more comprehensive study should be carried out to the concept shell design. This study intends to evaluate the presence of all the details and its behavior in the presence of the flow;
- The new geometries purposed in this work should also be reanalysed. In the view of the author, firstly, different heights for the spherical hubcap should be assessed;
- New geometries could be designed and evaluated to minimize the frequency induced by the flow;
- Different velocities can be studied, even if higher than the previous measured by the OBS, with the aim of developing a filtering model to be applied to the data collected by the OBS;
- The performance of this study may be compared, regarding to the computational effort vs quality results, by using different combinations of schemes, solutions or even solvers;
- Lastly, a hydrodynamic stability analysis shall be performed concerning the ascent and descent phases of the OBS mission, i.e. during launch and recovery phases.

Bibliography

[1] “Estrutura de missão para a extensão da plataforma continental website,” [Accessed: 2019-04-27]. [Online]. Available: https://www.emepc.pt/ xiii, 1, 2

[2] “Nodes,” [Accessed:2019-04-15]. [Online]. Available: https://magseisfairfield.com/technology-2/znodal/nodes/ xiii, 8

[3] “Oceanworks international,” October 2010, [Accessed April 15, 2019]. [Online]. Available: http://www.oceanworks.com/media-info/photos/ xiii, 8

[4] D. O’Gorman, “Obsip: Supporting the cascadia initiative and improving ocean bottom seismograph data quality,” 2014, [Accessed: 2019-04-25]. [Online]. Available: https://www.iris.edu/hq/science_highlights/obsip_supporting_the_cascadia_initiative_and_improving_ocean_bottom_seismog xiii, 9

[5] I. Grevemeyer, “Ocean bottom seismometer,” [Accessed: 2019-04-20]. [Online]. Available: https://www.eurekalert.org/multimedia/pub/171387.php?from=395084 xiii, 10

[6] Y. Yamashita, “Rumbling from ocean trenches could be sign that japan faces mega earthquake,” 2015, [Accessed: 2019-04-20]. [Online]. Available: http://theconversation.com/rumbling-from-ocean-trenches-could-be-sign-that-japan-faces-mega-earthquake-41464 xiii, 10

[7] D. Reuteler, “the drawings of leonardo da vinci.” [Online]. Available: http://www.drawingsofleonardo.org/ xiii, 12

[8] NASA, “Red smoke on the ground shows the vortex made by the air flowing from the wing of the airplane.” 1990, [Accessed: 2019-03-18]. [Online]. Available: https://www.nasa.gov/audience/forstudents/k-4/dictionary/Vortex.html xiii, 12

[9] K. Muralidharan, S. Muddada, and B. S. V. Patnaik, “Numerical simulation of vortex induced vibrations and its control by suction and blowing,” *Applied Mathematical Modelling*, vol. 37, no. 1-2, pp. 284–307, 2013. xiii, 13

[10] B. M. Sumer and J. Fredsøe, *Hydrodynamics Around Cylindrical Structures*, 2006. xiii, xv, 10, 11, 12, 13, 14, 15, 16, 17

[11] Z. Trávníček, A. B. Wang, and W. Y. Tu, “Laminar vortex shedding behind a cooled circular cylinder,” *Experiments in Fluids*, vol. 55, no. 2, 2014. xiii, 14

[12] W. Bell, “Turbulence vs drag—some further considerations,” *Ocean Engineering*, vol. 10, no. 1, pp. 47–63, jan 1983. xiii, 17

[13] W. M. H. Versteeg, *An Introduction to Computational Fluid Dynamics: The Finite Volume Method Approach*, 1st ed. Prentice Hall, 1996. xiii, 18, 19, 20, 21, 22, 24, 28

[14] A. Bakker, “Lecture notes in applied computational fluid dynamics,” 2002, [Acessed: 2019-03-19]. [Online]. Available: hhtp://www.bakker.org xiii, 23

[15] H. Versteeg and W. Malalasekera, *An Introduction to Computational Fluid Dynamics: The Finite Volume Method (2nd Edition)*. Pearson, 2007. xiii, 27

[16] C. J. Greenshields, "User guide - version 5.0," pp. U–1–U–235, July 2017, [Acessed: 2019-04-27]. xiii, 28, 33, 34, 38

[17] C. B. Laney, *Computational Gasdynamics*. Cambridge University Press, 1998. xiii, 37

[18] Protect planet ocean. Accessed: 14 March 2019. [Online]. Available: http://www.protectplanetocean.org 1

[19] N. O. US Department of Commerce and A. Administration, "How much of the ocean have we explored?" [Accessed: 2019-03-09]. [Online]. Available: https://oceanservice.noaa.gov/facts/exploration.html 1

[20] H. Ferreira, A. Martins, J. M. Almeida, A. Valente, A. Figueiredo, B. Cruz, and A. S. M. M. S. A, "TURTLE – Systems and technologies for Deep Ocean long term presence," no. September, 2014. 1, 5, 7

[21] *International Handbook of Earthquake & Engineering Seismology, Part B, Volume 81B (International Geophysics)*. Academic Press, 2003. 1

[22] A. Tengberg, F. D. Bovee, P. Hall, W. Berelson, D. Chadwick, G. Ciceri, P. Crassous, A. Devol, S. Emerson, J. Gage, R. Glud, F. Graziottini, J. Gundersen, D. Hammond, W. Helder, K. Hinga, O. Holby, R. Jahnke, A. Khripounoff, S. Lieberman, V. Nuppenau, O. Pfannkuche, C. Reimers, G. Rowe, A. Sahami, F. Sayles, M. Schurter, D. Smallman, B. Wehrli, and P. D. Wilde, "Benthic chamber and profiling landers in oceanography — a review of design, technical solutions and functioning," *Progress in Oceanography*, vol. 35, no. 3, pp. 253–294, jan 1995. 1

[23] W. Parker, K. Doyle, E. Parker, P. Kershaw, S. Malcolm, and P. Lomas, "Benthic interface studies with landers. consideration of lander/interface interactions and their design implications," *Journal of Experimental Marine Biology and Ecology*, vol. 285-286, pp. 179–190, feb 2003. 6

[24] A. Tengberg, F. D. Bovee, P. Hall, W. Berelson, D. Chadwick, G. Ciceri, P. Crassous, A. Devol, S. Emerson, J. Gage, R. Glud, F. Graziottini, J. Gundersen, D. Hammond, W. Helder, K. Hinga, O. Holby, R. Jahnke, A. Khripounoff, S. Lieberman, V. Nuppenau, O. Pfannkuche, C. Reimers, G. Rowe, A. Sahami, F. Sayles, M. Schurter, D. Smallman, B. Wehrli, and P. D. Wilde, "Benthic chamber and profiling landers in oceanography — a review of design, technical solutions and functioning," *Progress in Oceanography*, vol. 35, no. 3, pp. 253–294, jan 1995. 6

[25] A. Eleftheriou, Ed., *Methods for the Study of Marine Benthos*. John Wiley & Sons, Ltd, may 2013. 6

[26] A. Freiwald and J. M. Roberts, Eds., *Cold-Water Corals and Ecosystems*. Springer Berlin Heidelberg, 2005. 6

[27] A. Heger, N. J. King, B. D. Wigham, A. J. Jamieson, P. M. Bagley, L. Allan, O. Pfannkuche, and I. G. Priede, "Benthic bioluminescence in the bathyal north east atlantic: luminescent responses of vargula norvegica (ostracoda: Myodocopida) to predation by the deep-water eel (synaphobranchus kaupii)," *Marine Biology*, vol. 151, no. 4, pp. 1471–1478, jan 2007. 6

[28] H. E. Studies, “Hadal-lander a,” [Acessed: 2019-03-26]. [Online]. Available: https://web.whoi.edu/hades/hadal-lander-a/ 6, 7

[29] P. Bagley, I. Priede, A. Jamieson, D. Bailey, E. Battle, C. Henriques, and K. Kemp, “Lander techniques for deep-ocean biological research,” *Underwater Technology*, vol. 26, no. 1, pp. 3–12, sep 2004. 7

[30] T. de L. Wenneck, T. Falkenhaug, and O. Bergstad, “Strategies, methods, and technologies adopted on the r.v. g.o. sars MAR-ECO expedition to the mid-atlantic ridge in 2004,” *Deep Sea Research Part II: Topical Studies in Oceanography*, vol. 55, no. 1-2, pp. 6–28, jan 2008. 7

[31] “Miningimpact blog,” [Acessed: 2019-04-14]. [Online]. Available: https://www.oceanblogs.org/eadsm/2019/05/08/bobo-and-the-7-7-dwarfs-how-to-sense-the-deep-sea/ 7

[32] T. van Weering, I. Hall, H. de Stigter, I. McCave, and L. Thomsen, “Recent sediments, sediment accumulation and carbon burial at goban spur, n.w. european continental margin (47–50°n),” *Progress in Oceanography*, vol. 42, no. 1-4, pp. 5–35, dec 1998. 7

[33] R. Person, Y. Aoustin, J. Blandin, J. Marvaldi, and J.-f. Rolin, “From bottom landers to observatory networks,” vol. 49, no. June, pp. 581–593, 2006. 7

[34] “Seafloor network & observatory components,” [Acessed: 21-08-2019]. [Online]. Available: http://www.oceanworks.com/our-business/science-environmental/observatorycomponents/ 8

[35] G. H. Sutton, “Ocean bottom seismology: History and current status,” in *Ocean Seismo-Acoustics*. Springer US, 1986, pp. 821–840. 8, 9

[36] M. C. Schmidt-Aursch and W. C. Crawford, “Ocean-bottom seismometer,” in *Encyclopedia of Earthquake Engineering*. Springer Berlin Heidelberg, 2015, pp. 1735–1748. 9, 10

[37] Y. Jun and C. Ping, “Suppression of vortex shedding from a rectangular cylinder at low Reynolds numbers,” *Journal of Fluids and Structures*, vol. 43, pp. 15–27, 2013. [Online]. Available: http://dx.doi.org/10.1016/j.jfluidstructs.2013.08.001 12

[38] B. K. Gandhi, S. N. Singh, V. Seshadri, and J. Singh, “Effect of bluff body shape on vortex flow meter performance,” vol. 11, pp. 378–384, 2004. 12

[39] W. P. Graebel, *Advanced Fluid Mechanics*. Academic Press, 2007. 13

[40] R. Bourguet, G. E. Karniadakis, and M. S. Triantafyllou, “Lock-in of the vortex-induced vibrations of a long tensioned beam in shear flow,” *Journal of Fluids and Structures*, vol. 27, no. 5-6, pp. 838–847, jul 2011. 14

[41] C. J. Baker, “The turbulent horseshoe vortex,” *Journal of Wind Engineering and Industrial Aerodynamics*, 1980. 15

[42] L.-L. Eckerle, W.A., “Horseshoe Vortex Formation Around a Cylinder,” *The American Society of Mechanical Engineers*, pp. 1–9, 1986. 15

[43] H. Drescher, “Messung der auf querangeströmter zylinder ausgeübten zeitlich veränderten drücke,” *Z. Flugwiss*, vol. 4, pp. 17–21, 1956. 16

[44] M. P. Joel H. Ferziger, *Computational Methods For Fluid Dynamics*, 3rd ed. Springer, 2001. 18, 19, 20, 22

[45] R. L. R. Gary A. Flandro, Howard M. McMahon, *Basic Aerodynamics: Incompressible Flow*, ser. Cambridge Aerospace Series. Cambridge University Press, 2011. 22

[46] E. L. Houghton, *Aerodynamics for Engineering Students*, 5th ed. Butterworth-Heinemann, 2003. 22

[47] I. ANSYS, "Notes in modeling turbulent flows," pp. 6–2–6–49, 2006, [Acessed: 2019-03-19]. [Online]. Available: hhtp://www.fluentusers.com 23, 24, 25

[48] I. Fluent, "Fluent 6.1 user's guide," pp. 10–1–10–82, February 2003, [Acessed: 2019-03-22]. 23, 24, 25

[49] J. Bardina, P. Huang, and T. Coakley, "Turbulence modeling validation, testing, and development," April 1997. 23, 24, 25

[50] H. F. Wang, Y. Zhou, and J. Mi, "Effects of aspect ratio on the drag of a wall-mounted finite-length cylinder in subcritical and critical regimes," *Experiments in Fluids*, vol. 53, no. 2, pp. 423–436, apr 2012. 25, 40

[51] T. OKAMOTO and M. YAGITA, "The experimental investigation on the flow past a circular cylinder of finite length placed normal to the plane surface in a uniform stream," *Bulletin of JSME*, vol. 16, no. 95, pp. 805–814, 1973. 25, 40

[52] T. KAWAMURA, M. HIWADA, T. HIBINO, I. MABUCHI, and M. KUMADA, "Flow around a finite circular cylinder on a flat plate : Cylinder height greater than turbulent boundary layer thickness," *Bulletin of JSME*, vol. 27, 1984. 25, 40

[53] H. Sakamoto and M. Arie, "Vortex shedding from a rectangular prism and a circular cylinder placed vertically in a turbulent boundary layer," *Journal of Fluid Mechanics*, vol. 126, pp. 147–165, jan 1983. 26, 40

[54] H. Sakamoto and S. Oiwake, "Fluctuating forces on a rectangular prism and a circular cylinder placed vertically in a turbulent boundary layer," *Journal of Fluids Engineering*, vol. 106, no. 2, pp. 160–166, jun 1984. 26, 40

[55] S. Okamoto and Y. Sunabashiri, "Vortex shedding from a circular cylinder of finite length placed on a ground plane," *Journal of Fluids Engineering*, vol. 114, no. 4, pp. 512–521, dec 1992. 26, 40

[56] C.-W. Park and S.-J. Lee, "Free end effects on the near wake flow structure behind a finite circular cylinder," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 88, no. 2-3, pp. 231–246, dec 2000. 26, 40

[57] D. Zhang, "Comparison of various turbulence models for unsteady flow around a finite circular cylinder at re=20000," *Journal of Physics: Conference Series*, vol. 910, p. 012027, oct 2017. 26, 35

[58] CornellX, "A hands-on introduction to engineering simulations," [Accessed: 2019-04-25]. [Online]. Available: https://www.edx.org/course/a-hands-on-introduction-to-engineering-simulations 28

[59] H. J. Medina, A. Beechook, J. Saul, S. Porter, S. Aleksandrova, and S. Benjamin, “Open source computational fluid dynamics using openfoam,” Nov. 2015. 28

[60] “Catia™ 3dexperience® - dassault systèmes® 3d software.” [Online]. Available: https://www.3ds.com/products-services/catia/ 31

[61] I. T. T. Conference, “Practical guidelines for ship cfd applications,” ITTC - Recommended Procedures and Guidelines, Tech. Rep., 2014. 32, 37

[62] T. Khamlaj, “Computation of y+,” Oct 2018, [Acessed: 2019-05-14]. [Online]. Available: https://tariqkhamlaj.com/2018/10/15/computation-of-y/ 36

[63] G. Kalitzin, G. Medic, G. Iaccarino, and P. Durbin, “Near-wall behavior of RANS turbulence models and implications for wall functions,” *Journal of Computational Physics*, vol. 204, no. 1, pp. 265–291, mar 2005. 36

[64] B. Launder and D. Spalding, “The Numerical Computation of Turbulent Flows, booktitle = Numerical Prediction of Flow, Heat Transfer, Turbulence and Combustion, publisher = Elsevier, year = 1983, pages = 96–116, doi = 10.1016/b978-0-08-030937-8.50016-7.” 36

[65] J. H. Lienhard, *Synopsis of lift, drag, and vortex frequency data for rigid circular cylinders*. Technical Extension Service, Washington State University, 1966. 39

[66] M. M. Zdravkovich, *Flow Around Circular Cylinders: Volume 2: Applications*. OXFORD UNIV PR, 2003. 40, 50

[67] “Water - density, specific weight and thermal expansion coefficient.” [Online]. Available: https://www.engineeringtoolbox.com/water-density-specific-weight-d_595.html 41

[68] E. Ranjith, A. Sunil, and L. Pauly, “Analysis of flow over a circular cylinder fitted with helical strakes,” *Procedia Technology*, vol. 24, pp. 452–460, 2016. 44

Appendix A

A.1 OpenFoam Files

The following documents were considered for the meshing, initialization and simulation processes:
blockMeshDict:

```
/*--------------------------------*- C++ -*----------------------------------*\
| =======                 |                                                 |
| \\      /  F ield       | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration   | Version: 4.x                                    |
|   \\  /    A nd         | Web: www.OpenFOAM.org                           |
|    \\/     M anipulation |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    object      blockMeshDict;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
convertToMeters 1;
vertices
(
    (-5 -5 -0.25)
    (15 -5 -0.25)
    (15 5 -0.25)
    (-5 5 -0.25)
    (-5 -5 4.75)
    (15 -5 4.75)
    (15 5 4.75)
    (-5 5 4.75)
);
blocks
(
    hex (0 1 2 3 4 5 6 7) (40 20 10) simpleGrading (1 1 1)
);
edges
(
);
boundary
```

```
(
    inlet
    {
    type patch;
    faces
    (
       (0 4 7 3)
    );
    }
    outlet
    {
    type patch;
    faces
    (
       (2 6 5 1)
    );
    }
    top
    {
    type patch;
    faces
    (
       (4 5 6 7)
    );
    }
    right
    {
    type patch;
    faces
    (
       (1 5 4 0)
    );
    }
    left
    {
    type patch;
    faces
    (
       (3 7 6 2)
    );
    }
    bot
    {
    type wall;
    faces
    (
       (0 3 2 1)
```

```
        );
        }
);
mergePatchPairs
(
);
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

controlDict:

```
/*------------------------*- C++ -*-------------------------*\
| =======                 |                                                    |
| \\      /  F ield       | OpenFOAM: The Open Source CFD Toolbox              |
|  \\    /   O peration   | Version: 4.x                                       |
|   \\  /    A nd         | Web: www.OpenFOAM.org                              |
|    \\/     M anipulation |                                                   |
\*-------------------------------------------------------------------*/
FoamFile
{
        version       2.0;
        format        ascii;
        class         dictionary;
        object        controlDict;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
application      pimpleFoam;
startFrom      latestTime;
startTime      0;
stopAt      endTime;
endTime      3000;
deltaT      0.001;
writeControl      timeStep;
writeInterval      500;
purgeWrite      10;
writeFormat      binary;
writePrecision      8;
writeCompression      off;
timeFormat      general;
timePrecision      6;
runTimeModifiable      yes;
adjustTimeStep      yes;
maxCo      20;
maxDeltaT      0.125;

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

fvSchemes:

```
/*--------------------------------*- C++ -*----------------------------------*\
| =======                 |                                                 |
| \\      /  F ield       | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration   | Version: 4.x                                    |
|   \\  /    A nd         | Web: www.OpenFOAM.org                           |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    object      fvSchemes;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
ddtSchemes
{
    default     CrankNicolson 0.9;
}
gradSchemes
{
    grad(p)     Gauss linear;
    grad(U)     Gauss linear;
    default     cellLimited Gauss linear 1;
}
divSchemes
{
    default     none;
    div(phi,U)      Gauss linearUpwind grad(U);
    div(phi,k)      Gauss upwind;
    div(phi,omega)      Gauss upwind;
    div((nuEff*dev2(T(grad(U)))))      Gauss linear;
}
laplacianSchemes
{
    default     Gauss linear corrected;
}
interpolationSchemes
{
    default     linear;
}
snGradSchemes
{
    default     corrected;
}
wallDist
```

```
{
    method      meshWave;
} // * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

fvSolutions:

```
/*--------------------------------*- C++ -*----------------------------------*\
| =======                 |                                                 |
| \\      /  F ield       | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration   | Version: 4.x                                    |
|   \\  /    A nd         | Web: www.OpenFOAM.org                           |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    object      fvSolutions;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
solvers
{
    p
    {
        solver GAMG;
        tolerance 1e-6;
        relTol 0.01;
        smoother GaussSeidel;
        nPreSweeps 0;
        nPostSweeps 2;
        cacheAgglomeration on;
        agglomerator faceAreaPair;
        nCellsInCoarsestLevel 100;
        mergeLevels 1;
       minIter 3;
    }
    pFinal
    {
        solver GAMG;
        tolerance 1e-6;
        relTol 0.0;
        smoother GaussSeidel;
        nPreSweeps 0;
        nPostSweeps 2;
        cacheAgglomeration on;
        agglomerator faceAreaPair;
        nCellsInCoarsestLevel 100;
```

```
        mergeLevels 1;
        minIter 2;
    }
    U
    {
        type coupled;
        solver PBiCCCG;
        preconditioner DILU;
        tolerance (1e-08 1e-08 1e-08);
        relTol (0.01 0.01 0.01);
        minIter 3;
    }
    UFinal
    {
        type coupled;
        solver PBiCCCG;
        preconditioner DILU;
        tolerance (1e-08 1e-08 1e-08);
        relTol (0 0 0);
        minIter 3;
    }
    k
    {
        solver GAMG;
        tolerance 1e-5;
        relTol 0.01;
        smoother GaussSeidel;
        nPreSweeps 0;
        nPostSweeps 2;
        cacheAgglomeration on;
        agglomerator faceAreaPair;
        nCellsInCoarsestLevel 100;
        mergeLevels 1;
minIter 3;
    }
    kFinal
    {
        solver GAMG;
        tolerance 1e-6;
        relTol 0.001;
        smoother GaussSeidel;
        nPreSweeps 0;
        nPostSweeps 2;
        cacheAgglomeration on;
        agglomerator faceAreaPair;
        nCellsInCoarsestLevel 100;
        mergeLevels 1;
```

```
        minIter 3;
    }
    omega
    {
        solver GAMG;
        tolerance 1e-5;
        relTol 0.01;
        smoother GaussSeidel;
        nPreSweeps 0;
        nPostSweeps 2;
        cacheAgglomeration on;
        agglomerator faceAreaPair;
        nCellsInCoarsestLevel 100;
        mergeLevels 1;
        minIter 3;
    }
    omegaFinal
    {
        solver GAMG;
        tolerance 1e-6;
        relTol 0.001;
        smoother GaussSeidel;
        nPreSweeps 0;
        nPostSweeps 2;
        cacheAgglomeration on;
        agglomerator faceAreaPair;
        nCellsInCoarsestLevel 100;
        mergeLevels 1;
        minIter 3;
    }
}
PIMPLE
{
    nOuterCorrectors 3;
    nCorrectors 2;
    nNonOrthogonalCorrectors 1;
    pRefCell 0;
    pRefValue 0;
}
relaxationFactors
{
    fields
    {
        p 0.3;
    }
    equations
    {
```

```
        U 0.7;
        k 0.7;
        omega 0.7;
        nut 0.7;
    }
}

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

snappyHexMeshDict:

```
/*--------------------------------*- C++ -*----------------------------------*\
| =======                 |                                                 |
| \\      /  F ield       | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration   | Version: 4.x                                    |
|   \\  /    A nd         | Web: www.OpenFOAM.org                           |
|    \\/     M anipulation |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    object      snappyHexMeshDict;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
castellatedMesh true;
snap true;
addLayers true;
geometry
{
    cylinder.stl
    {
        type triSurfaceMesh;
        name cylinder;
        regions
        {
        }
    }
    refinementBox
    {
        type searchableBox;
        min (-2 -3 4.5);
        max (15 3 4.5);
    }
    refinementBox1
    {
        type searchableBox;
```

```
            min (-6 -6 -0.25);
            max (15 6 6);
        }
        refinementBox3
        {
            type searchableBox;
            min (-2.5 -2.5 -0.25);
            max (12 2.5 3.25);
        }
};
castellatedMeshControls
{
        maxLocalCells 100000;
        maxGlobalCells 2000000;
        minRefinementCells 0;
        maxLoadUnbalance 0.10;
        nCellsBetweenLevels 6;
        features
        (
        {
            file ”cylinder.eMesh”;
            level 0;
        }
        );
        refinementSurfaces
        {
            cylinder
            {
               level (4 5);
               patchInfo
               {
                  type wall;
               }
            }
        }
        resolveFeatureAngle 30;
        refinementRegions
        {
            refinementBox
            {
               mode inside;
               levels ((2 2));
            }
            refinementBox1
            {
               mode inside;
               levels ((1 1));
```

```
            }
            refinementBox3
            {
                mode inside;
                levels ((3 3));
            }
            cylinder
            {
                mode distance;
                levels ((0.5 4));
            }
        }
        locationInMesh (2 0 0);
        allowFreeStandingZoneFaces true;
}
snapControls
{
        nSmoothPatch 3;
        tolerance 2.0;
        nSolveIter 30;
        nRelaxIter 5;
            nFeatureSnapIter 10;
            implicitFeatureSnap false;
            explicitFeatureSnap true;
            multiRegionFeatureSnap false;
}
addLayersControls
{
        relativeSizes false;
        expansionRatio 1.2;
        firstLayerThickness 0.0001;
        minThickness 0.0001;
        layers
        {
            cylinder
            {
                nSurfaceLayers 20;
            }
        }
        nGrow 0;
        featureAngle 180;
        maxFaceThicknessRatio 0.5;
        nSmoothSurfaceNormals 1;
        nSmoothThickness 10;
        minMedialAxisAngle 90;
        minMedianAxisAngle 90;
        maxThicknessToMedialRatio 0.3;
```

```
        nSmoothNormals 3;
        nMedialAxisIter 10;
        slipFeatureAngle 30;
        nRelaxIter 5;
        nBufferCellsNoExtrude 0;
        nLayerIter 50;
        nRelaxedIter 20;
}
meshQualityControls
{
        #include ”meshQualityDict”
        relaxed
        {
            maxNonOrtho 75;
        }
            nSmoothScale 4;
            errorReduction 0.75;
}
writeFlags
(
        scalarLevels
        layerSets
        layerFields
);
mergeTolerance 1e-6;

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

U:

```
/*--------------------------------*- C++ -*----------------------------------*\
| =======                 |                                                 |
| \\      /  F ield       | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration   | Version: 4.x                                    |
|   \\  /    A nd         | Web: www.OpenFOAM.org                           |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
        version       2.0;
        format        ascii;
        class         volVectorField;
        object        U;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
dimensions [0 1 -1 0 0 0 0];
internalField uniform (0.08 0 0);
boundaryField
```

```
{
    outlet
    {
        type zeroGradient;
    }
    top
    {
        type zeroGradient;
    }
    bot
    {
        type fixedValue;
        value uniform (0 0 0);
    }
    inlet
    {
        type fixedValue;
        value uniform (0.08 0 0);
    }
    cylinder
    {
        type fixedValue;
        value uniform (0 0 0);
    }
    left
    {
        type zeroGradient;
    }
    right
    {
        type zeroGradient;
    }
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

p:

```
/*--------------------------------*- C++ -*----------------------------------*\
| =======                 |                                                 |
| \\      /  F ield       | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration   | Version: 4.x                                    |
|   \\  /    A nd         | Web: www.OpenFOAM.org                           |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
```

```
    class       volVectorField;
    object      p;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
dimensions [0 2 -2 0 0 0 0];
internalField uniform 0;
boundaryField
{
    outlet
    {
        type fixedValue;
        value uniform 0;
    }
    top
    {
        type zeroGradient;
    }
    bot
    {
        type zeroGradient;
    }
    inlet
    {
        type zeroGradient;
    }
    cylinder
    {
        type zeroGradient;
    }
    left
    {
        type zeroGradient;
    }
    right
    {
        type zeroGradient;
    }
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

omega:

```
/*--------------------------------*- C++ -*----------------------------------*\
| =======                 |                                                 |
| \\      /  F ield       | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration   | Version: 4.x                                    |
|   \\  /    A nd         | Web: www.OpenFOAM.org                           |
|    \\/     M anipulation |                                                 |
```

```
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      omega;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
dimensions [0 0 -1 0 0 0 0];
internalField uniform 0.8;
boundaryField
{
    outlet
    {
        type zeroGradient;
    }
    top
    {
        type zeroGradient;
    }
    bot
    {
        type omegaWallFunction;
        value uniform 80000;
    }
    inlet
    {
        type fixedValue;
        value uniform 0.8;
    }
    cylinder
    {
        type omegaWallFunction;
        value uniform 80000;
    }
    left
    {
        type zeroGradient;
    }
    right
    {
        type zeroGradient;
    }
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

nut:

```
/*--------------------------------*- C++ -*----------------------------------*\
| =======                 |                                                 |
| \\      /  F ield       | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration   | Version: 4.x                                    |
|   \\  /    A nd         | Web: www.OpenFOAM.org                           |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      nut;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
dimensions [0 2 -1 0 0 0 0];
internalField uniform 0;
boundaryField
{
    outlet
    {
        type calculated;
        value uniform 0;
    }
    top
    {
        type zeroGradient;
    }
    bot
    {
        type calculated;
        value uniform 0;
    }
    inlet
    {
        type calculated;
        value uniform 0;
    }
    cylinder
    {
        type calculated;
        value uniform 0;
    }
    left
    {
        type zeroGradient;
```

```
    }
    right
    {
        type zeroGradient;
    }
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

k:

```
/*--------------------------------*- C++ -*----------------------------------*\
| =======                 |                                                 |
| \\      /  F ield       | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration   | Version: 4.x                                    |
|   \\  /    A nd         | Web: www.OpenFOAM.org                           |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      k;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
dimensions [0 2 -2 0 0 0 0];
internalField uniform 0.000008;
boundaryField
{
    outlet
    {
        type zeroGradient;
    }
    top
    {
        type zeroGradient;
    }
    bot
    {
        type kqRWallFunction;
        value uniform 0.000000000000000001;
    }
    inlet
    {
        type fixedValue;
        value uniform 0.000008;
    }
    cylinder
```

```
    {
        type kqRWallFunction;
        value uniform 0.000000000000000001;
    }
    left
    {
        type zeroGradient;
    }
    right
    {
        type zeroGradient;
    }
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

turbulenceProperties:

```
/*--------------------------------*- C++ -*----------------------------------*\
| =======                 |                                                 |
| \\      /  F ield       | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration   | Version: 4.x                                    |
|   \\  /    A nd         | Web: www.OpenFOAM.org                           |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    object      turbulenceProperties;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
simulationType RAS;
RAS
{
    RASModel kOmegaSST;
    turbulence on;
    printCoeffs on;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

transportProperties:

```
/*--------------------------------*- C++ -*----------------------------------*\
| =======                 |                                                 |
| \\      /  F ield       | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration   | Version: 4.x                                    |
|   \\  /    A nd         | Web: www.OpenFOAM.org                           |
|    \\/     M anipulation  |                                                 |
```

```
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    object      transportProperties;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
transportModel Newtonian;
nu [ 0 2 -1 0 0 0 0 ] 1.e-06;
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

Appendix B

B.1 On the Fly Documents

These documents are data files and were written in Notepad2.

forcesgnuplot: This command was used to plot the drag and lift forces or coefficients on the fly (Figure B.1).

```
set yrange [-0.3:1]
set title "coefficients"
set ylabel 'coefficients'
set xlabel 'time'
plot  "< sed s/[\\(\\)]//g ./postProcessing/forceCoeffs_object/0/forceCoeffs.dat" using 1:3 with lines title "cd" ,\
      "< sed s/[\\(\\)]//g ./postProcessing/forceCoeffs_object/0/forceCoeffs.dat" using 1:4 with lines title "cl"
pause 5
reread
```

Figure B.1: Forcesgnuplot file structure.

Residuals: This command was used to plot the residual on the fly (Figure B.2).

```
set logscale y
set title "Residuals"
set ylabel 'Residual'
set xlabel 'Iteration'
plot "< cat log | grep 'Solving for Ux' | cut -d' ' -f9 | tr -d ',' title 'Ux' with lines,\
"< cat log | grep 'Solving for Uy' | cut -d' ' -f9 | tr -d ',' title 'Uy' with lines,\
"< cat log | grep 'Solving for Uz' | cut -d' ' -f9 | tr -d ',' title 'Uz' with lines,\
"< cat log | grep 'Solving for omega' | cut -d' ' -f9 | tr -d ',' title 'omega' with lines,\
"< cat log | grep 'Solving for k' | cut -d' ' -f9 | tr -d ',' title 'k' with lines
pause 5
reread
```

Figure B.2: Residuals file structure.

Appendix C

C.1 FFT Process

After the simulation is performed, it creates a data file with the time-domain force coefficients. In order to obtain the shedding vortex frequency, it is necessary to transfer from the time domain to the frequency domain the lift coefficient data (Figure C.1). For this, and using Microsoft Excel, it is used the tool presented in ”Tools/Data Analysis” called ”Fourier Analysis”.

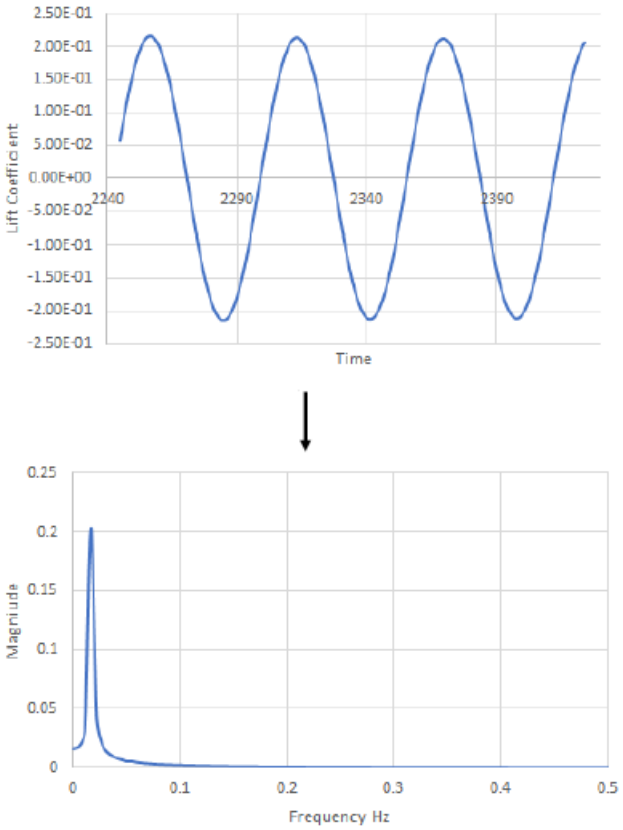


Figure C.1: FFT.

1. If ”Analysis ToolPak” is not installed in Microsoft Excel, go to ”Tools” menu, click in ”Add-Ins” and select ”Analysis ToolPak” check box;
2. Label the cells from A1 to E1 as the Figure C.2;
3. In the Figure C.2, t is the time interval, D is the number of data and f_s is the sampling frequency ($f_s = \frac{D}{t}$);
4. Fill the column E, called ”FFT complex”, using ”Tools/Data” ”Analysis/Fourier Analysis”. Use as input the column B and select the output range as column D.
5. Calculate FFT magnitude using: $FFTmag(A2) = \frac{2}{D} * IMABS(E2)$;

6. Calculate FFT frequency using: $FFTfreq(A2) = x * \frac{fs}{D}$, where x is zero in the first cell (C2), and after is the row number less two;

7. Finally plot FFT magnitude in function of FFT frequency.

	A	B	C	D	E	F	G	H	I
1	time	data	FFT freq	FFT mag	FFT complex		t	D	fs
2	2244.883	5.75E-02	0	0.016332	16.724231018614		179.3466537	2048	11.41923
3	2244.971	5.96E-02	0.005576	0.018638	18.3185247818797-5.35529855527061i	1			
4	2245.059	6.16E-02	0.011152	0.031023	27.403253450841-16.0690020099976i	2			
5	2245.147	6.36E-02	0.016727	0.203136	153.809156972791-140.040187347568i	3			
6	2245.235	6.56E-02	0.022303	0.041453	-27.8167474660503+32.0625543628212i	4			
7	2245.323	6.77E-02	0.027879	0.019038	-11.1177558225642+16.0142826945753i	5			
8	2245.411	6.97E-02	0.033455	0.012497	-6.40819838040088+11.0768484611122i	6			
9	2245.499	7.16E-02	0.039031	0.009364	-4.25913994692807+8.59118138526691i	7			
10	2245.587	7.36E-02	0.044606	0.007495	-3.03025014199976+7.05183173926561i	8			
11	2245.675	7.56E-02	0.050182	0.006074	-2.08042278673728+5.86189568932318i	9			
12	2245.763	7.77E-02	0.055758	0.005892	-2.25776291014497+5.59509831762607i	10			
13	2245.852	7.95E-02	0.061334	0.005015	-1.67567531374304+4.85386833213735i	11			
14	2245.94	8.15E-02	0.06691	0.004466	-1.37313866279277+4.36191397314968i	12			
15	2246.028	8.35E-02	0.072485	0.004041	-1.15929890471276+3.9726675462617i	13			
16	2246.116	8.54E-02	0.078061	0.003697	-0.995961034706824+3.65228335640 68i	14			
17	2246.204	8.74E-02	0.083637	0.003409	-0.866425802610763+3.38202978064062i	15			
18	2246.292	8.93E-02	0.089213	0.003181	-0.805740373390725+3.15663086509621i	16			
19	2246.381	9.12E-02	0.094788	0.002961	-0.690405540811445+2.952728788651 07i	17			
20	2246.469	9.32E-02	0.100364	0.002778	-0.619864653449247+2.77666006403878i	18			
21	2246.557	9.51E-02	0.10594	0.002618	-0.561908692912987+2.62092709871185i	19			
22	2246.646	9.69E-02	0.111516	0.002476	-0.51268580028467+2.48253939588999i	20			

Figure C.2: FFT process.

Appendix D

D.1 Three dimensions views

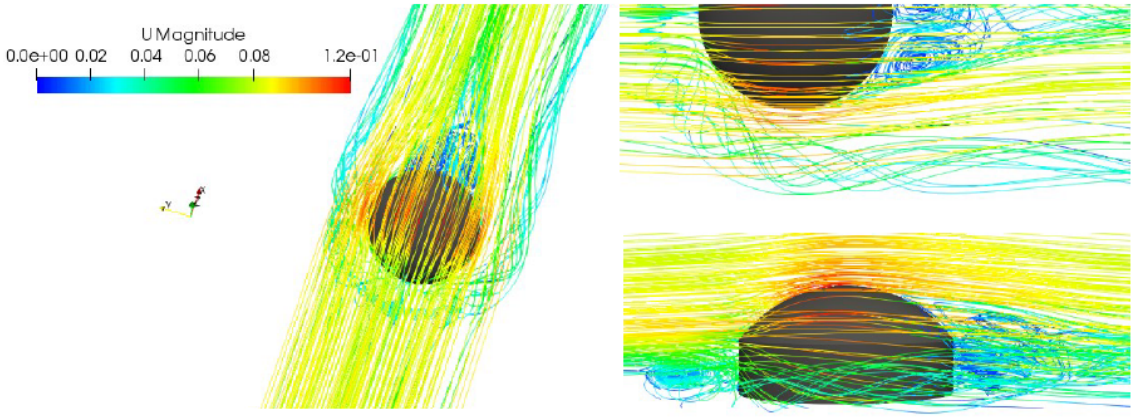


Figure D.1: Three dimensional flow around spherical cap. Close view of horseshoe vortex.

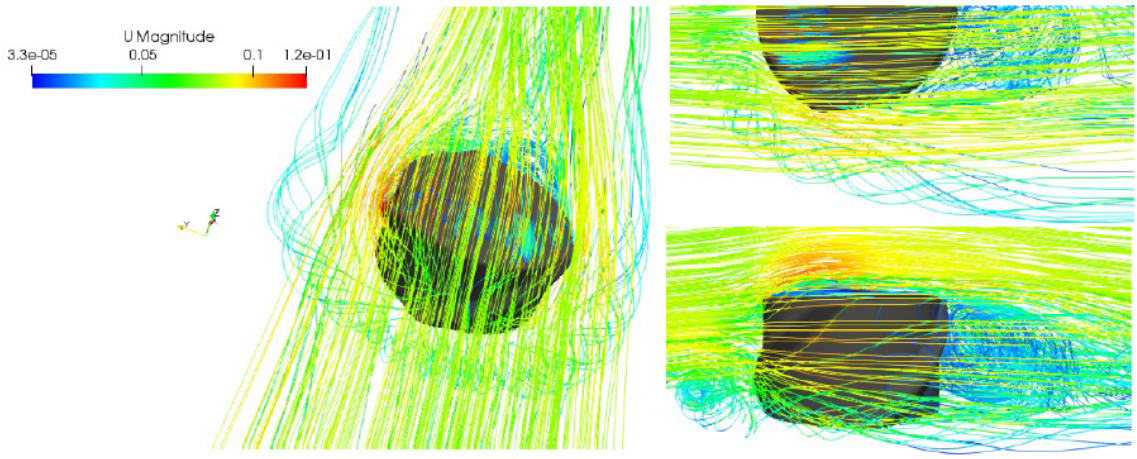


Figure D.2: Three dimensional flow around cylinder with three helical strakes. Close view of horseshoe vortex.