COMPUTATIONAL FLUID DYNAMICS ANALYSIS ON THE FREEFALL OF A LIFEBOAT

Mateus Fujita Silveira

Thesis to obtain the Master of Science Degree in

Naval Architecture and Ocean Engineering

Supervisor: Prof. Shan Wang

Co-Supervisor: Prof. Carlos Guedes Soares

Examination Committee

Chairperson: Prof. Tiago A. Santos

Supervisor: Prof. Shan Wang

Members of the Committee: Prof. Sarat Mohapatra

December 2022

TÉCNICO
LISBOA

Computational Fluid Dynamics analysis on the freefall of a lifeboat

Mateus Fujita Silveira

DECLARATION

I declare that this document is an original work of my own authorship and that it fulfils all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

Signature

Mateus Fujita Silveira

ACKNOWLEDGEMENTS

I want to express my gratitude to Prof. Shan Wang, who has been sharing her knowledge and giving valuable advises throughout the process. Special thanks to Prof. Carlos Guedes Soares for providing the opportunity for me to study and stay in Portugal by the means of a scholarship.

This work is performed within the Strategic Research Plan of the Centre for Marine Technology and Ocean Engineering, which is financed by Portuguese Foundation for Science and Technology (Fundação para a Ciência e Tecnologia - FCT) under contract UID/Multi/00134/2013 - LISBOA-01-0145-FEDER-007629. The numerical simulations are also supported by INCD through the advanced computational resource project from FCT under contract CPCA/A1/407670/2021.

Lisbon,

Mates Fujita Silveira

RESUMO

As cargas de *slamming* são estudadas em barcos salva-vidas utilizando o programa informático de mecânica dos fluidos computacional de fonte aberta OpenFOAM. Para este efeito, a técnica de sobreposição de malha é utilizada para realizar simulações multifásicas, com base no método do volume de fluido, devido ao grande deslocamento que se espera deste tipo de formulação. Após a revisão da pesquisa de *slamming* até aos dias de hoje e de uma visão geral sobre o programa OpenFOAM e do método de malha de quimera, foi feito um estudo de validação comparando a avaliação da entrada de água de uma cunha de ângulo de *deadrise* 30° com resultados experimentais obtidos de outros estudos. Os resultados numéricos estão em bom acordo não só com as medições, mas também com os resultados numéricos do método do elemento de borda e com simulações técnicas de deformação da malha que também foram calculadas antes e publicadas. Quanto às simulações de baleeiras salva-vidas, demonstraram ser mais exigentes em termos de capacidade computacional, mas ainda uma ferramenta robusta para este fim. Vinte e uma diferentes simulações foram realizadas utilizando 7 ângulos diferentes e 3 alturas de queda diferentes. As pressões sobre vários pontos do mesmo plano transversal e longitudinal foram comparadas. Quanto às pressões, notou-se que ocorrem dois picos principais: um na proa durante a entrada de água e outro na popa, que aparece após uma segunda entrada de água devido ao giro de retorno induzido por torque de restauração. O mapa de pressão mostrou também que a quilha da embarcação é a parte crítica com as cargas mais elevadas. A cinemática foi também comparada para verificar que ângulo seria capaz de afastar mais o barco salva vidas sem mais impulsos. Neste aspecto, o lançamento de 70° apresentou o melhor desempenho para uma vasta gama de alturas, o que já pode ser um indicativo de qual a configuração que teria maiores probabilidades de salvar vidas

chaves: baleeira salva-vidas, mecânica dos fluidos, OpenFOAM, *slamming*, entrada em água, quimera

ABSTRACT

Slamming loads on lifeboats are studied in this study by using the open-source CFD toolbox OpenFOAM. For this purpose, the overset grid technic is used to perform multiphase simulations, based on the volume of fluid method, due to the large motion that is expected from this kind of formulation. After reviewing the slamming research until nowadays and making an overview of the software OpenFOAM and the chimera mesh method, a validation study was made by comparing the assessment of the water entry of a 30° deadrise angle wedge with experimental results obtained from other studies. The results were in good agreement not only with the measurements, but also with the boundary element method numerical results and deforming mesh simulations that were published before. As for the lifeboat simulations, they have shown to be more demanding of computer capacity, but still a robust tool for this purpose. Totally 21 different simulations were performed by using 7 different initial pitch angles and 3 different dropping heights. The pressure over several points in the same transversal and longitudinal plan were compared. As for the pressures, it was noticed that two major peaks occur: one on the bow during water entry and other on the stern, which appears after a second water entry due to the turnback spin induced by restoring moments. The pressure contours have shown that the keel of the lifeboat is the critical part with the highest loads. The kinematics were also compared to check which angle would be able to get further from the hazardous event without any more impulse. In this aspect, the launching with an initial pitch angle of 70° presented the best performance for a wide range of heights.

Keywords: Lifeboat, CFD, OpenFOAM, slamming, water entry, overset mesh

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

BEM – Boundary Element Method

CAD – Computer-assisted design

CG - Centre of Gravity

CFD – Computational Fluid Dynamics

CFL – Courant-Friedrichs-Lewy

DOF – Degrees of Freedom

FSI – Fluid-Structure Integration

GCI – Grid Convergence Index

LSA – Life-Saving Appliances

RANS – Reynolds Average Navier-Stokes

VOF – Volume of Fluid

1. INTRODUCTION

1.1 Slamming Problem in Lifeboat Context

The safety of life in the sea should be guaranteed for a ship to navigate. One way to secure this during extreme maritime incidents where the crew and persons on board must leave the ship is by having and using lifeboats. This kind of boat is required to be launched from a ship deck or a platform at a considerable height, gaining enough momentum and saving launch time (Huang, et al., 2021). In this scenario, the lifeboat experiences a free-fall process before approaching the water surface, entering with high speed, and causing significant pressure forces on its structure, as shown in Figure 1.1.

Figure 1.1 - Water entry of a freefall lifeboat

When one of these free-fall lifeboats needs to evacuate during these extreme events, the watercraft undergoes some stages from its launching to the navigation.First, there is the initial interaction between the craft and the launching skid where it was resting on. Commonly, the lifeboat is released by a hook and it slides along the rail pushed by the gravitational forces and resisted by the friction during this launching phase (A). Within this interaction, there is also the rotation phase (B) that occurs when the boat reaches the end of the skid and a torque appears by the combination of the rail reaction force and gravity, giving the lifeboat an initial angular momentum that depends on how fast it is launched from the skid. After leaving the skid, there is the free-falling phase (C), where the only forces moving the lifeboat are gravity and air resistance. This phase is influenced by the initial velocity, rotation, launching angle, and height, in which this last one is important to determine whether the drag force is relevant to the analysis (Andrya, et al., 2019).

The stages that are the focus of the main research fields are the water entry (D), submergence (E), and re-emergence (F). In the water entry phase, the predominant forces that take place are the hydrostatic and hydrodynamic forces, creating high accelerations over the bow of the lifeboat. Since the boat usually hits the water by bow impact, its combination with the gravitational force also creates an angular momentum, which combined with the vertical velocity also makes the hull's stern impact the water. Depending on the angle and penetration and the velocity gained during the freefall, the lifeboat can submerge underwater, preceding its later re-emergence driven by the buoyancy forces and proceeding in positive headway. The phases, from launching to navigation, are illustrated in Figure 1.2.

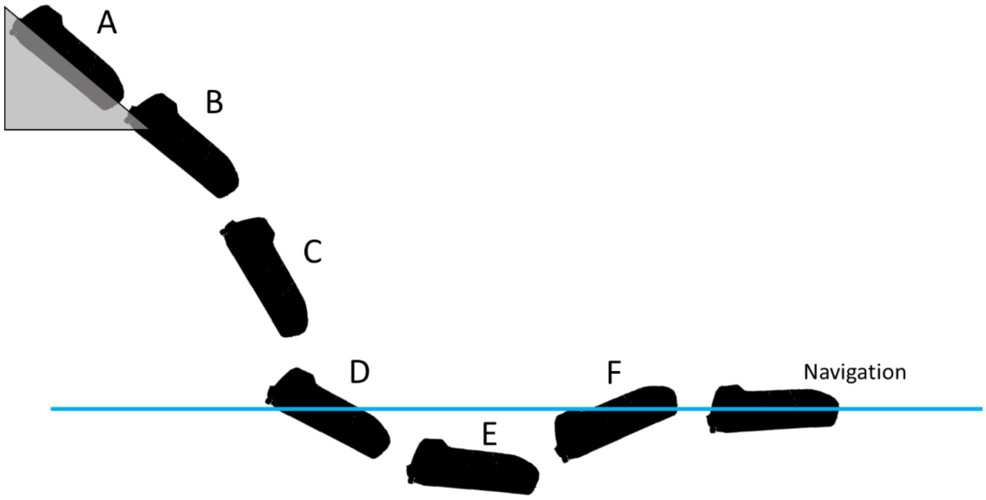


Figure 1.2 - Lifeboat phases of operation in extreme events

It is crucial to evaluate the effects of water penetration and make appropriate preparations to guarantee the lifeboat integrity in these extreme situations (Huang, et al., 2021), because if the boat is launched or designed inappropriately, the structure can be compromised causing serious damage to the hull structure and threaten the personal safety of the crew (Qiu, et al., 2020). This already happened before on the 28th of September 1994 with the MV Estonia on the Baltic Sea, which was one of the deadliest maritime disasters of the 20th century initialized by the break of the bow door due to severe slamming experience by the Ro/Ro ferry.

The water entry phase is marked by the high velocity the boat bottom enters the water, which causes impulse loads with high-pressure peaks which are also called "slamming" (Faltinsen, 1990). These impact pressures depend on the relative angle between both surfaces of the fluid and the body, and when it is small, even at moderate velocities, the contact region on the interface between fluid and body expands speedily. This area creates an air cushion enclosed by the inner depression of the water surface and the outer edges of the body, where the compressibility effects of this portion of air play a crucial part in the pressure of the fluid. At one point, the air escapes and the contact between the body and the fluid is direct, and hydroelasticity plays an important role, as might the water compressibility (Kapsenberg, 2011).

Evaluating the water entry impact is, however, a challenge even nowadays. The most reliable method to study and evaluate this is by performing experiments, but it is unrealistic to measure the pressure with this method since it would be necessary to cover all the hull sensitive areas with pressure sensors and it would limit the locations where these measurements can be taken (Huang, et al., 2021). Besides this complexity, there is also the fact that the validity of the Froude's Law of similitude can be only accepted if the relative angle between fluid and body surfaces is larger than 5°, otherwise, only full-scale experiments can be applied to assess these pressures due to the intricacy of the compressed air. Therefore, several alternatives are in development, mostly numerical such as finite element analysis (FEA) models together with CFD simulations, computational models, etc, but are based on several simplifying hypotheses and are also still improving and requiring validation from the existing experimental known results (Huang, et al., 2021).

Hence, the evaluation of slamming load is crucial for the safety and movement responses of marine structures, especially in what concerns free-fall lifeboats.

1.2 Literature review

As an important application in naval architecture, the dynamic pressure on surface ships and offshore structures during the water entry process has been a long-lasting topic. The pioneering method to estimate these impulse loads is the momentum theory, where the first to introduce was Von Kármán (1929), where this study applied momentum variation dependent on the speed rate and wetted area to reach an asymptotic theory for assessing the impact loads of seaplanes floats during landing, which was treated as an idealized two-dimensional wedge impacting a calm water surface and neglected the local up rise of the water (Wang, et al., 2021). The concept of added mass was already known by that time, and this study considered the inertia increment of a long plate with finite width to compute the restricted vertical momentum. Further developments were a motivation for other investigators, which was the case for the analysis of Pabst (1930), where it extended the study by including the forward velocity and noted the importance of the elasticity of the airplane for the impact forces, stating that for stiff float bottom and taking the air compressibility into account, the acceleration of a portion of water in an infinitely short time would produce infinitely great forces, according to the momentum theory (Pabst, 1930). This is only true when a flat plate impacts horizontally on water and when the formation of the air cushion depression is neglected, but still worth noting (Kapsenberg, 2011).

What is one of the most well-known studies which use the momentum theory approach is Wagner (1932). Here the approach of potential flow theory, which assumed irrotational flow with linearized boundary conditions without gravity, is used to assess an asymptotic solution for the water entry problem of two-dimensional bodies with small deadrise angles and also considering local jet analysis where its flow while hitting on the walls of the body is assumed to occur while water rises. In this theory, two fluid domains were considered for dividing the flow, where the inner flow domain accounts for a jet flow between the body and the free surface, and for the outer flow domain, the dynamic free surface condition and the body boundary condition were set to a horizontal line (Wang & Guedes Soares, 2013). Another groundwork has been done for developing this theory. One of these was done by Doobrovol'skaya (1969), which took advantage of the simplicity of the body geometry and transferred the potential flow problem into a self-similar flow problem in a complex plane to reach an analytical solution. This was a good solution for wedges with any deadrise angle, but it could not be extended for arbitrary geometries. Also, by using matching asymptotic expansions, the innovative but intuitive work of von Kármán (1929) and Wagner (1932) has also been strengthened theoretically in the studies of Cointe & Armand (1987) and Howison et al. (1989).

There is also a contribution to the slamming problem by using the statistical approach. The method of Ochi & Michel (1964) predicts the probability of bottom slamming by considering the event of the ship's fore foot emerging combined with the over-expenditure of a threshold re-entry velocity. This probability was assessed by linearizing ship motions and using Rayleigh distribution. Ochi & Motter (1971, 1973) included slamming coefficients depending on the shape of the lowest draught's 10% of the considered section. This has refined the results, which previously were found to not have a significant difference between U- and V-shaped sections. Even though this method is easy to use, making it popular, and can compare different ships and sustainable speeds in different sea states, it has the shortcoming of considering the flat horizontal surface of the fluid right after the section fully emerges from it, and also is applicable to short-term statistics, making it problematic to apply for longer terms (Kapsenberg, 2011).

Another well-known approach for this problem is by using the Boundary-Element Method (BEM), where the method relies on the discretization of the body surface to apply boundary conditions to obtain the Froude-Krylov forces. The advantage of this method is that, for a reasonable computational cost, it can reach satisfactory results (Kapsenberg, 2011). The application for two-dimensional impacts was firstly introduced by Greenhow & Lin (1985) and later by Zhao & Faltinsen (1993), where it applies the nonlinear boundary element method with flow approximation to numerically assess the water entry of a two-dimensional body of arbitrary cross-section. This study is still used as a benchmark even nowadays (Huang, et al., 2021). Later, this method also was further developed in other studies, such as Zhao et al. (1996), where it accounted for flow separation from the knuckles of a body and compared with a solution without this addition and with experimental results, or Sun & Faltinsen (2007), where gravity effects were considered, and Wu et al. (2010), which accounted for nonlinear velocity attributes. One numerical problem that this approach gives is at the initial contact moment due to the discontinuity of the velocity potential, where it gives an infinite pressure result (Kapsenberg, 2011). One special treatment that could be given to this is to account for the compressibility effects of the fluid, as shown by Ogilvie & Francis (1963). Also, one limitation of this method is still relying on potential flow theory, which does not account for viscosity and turbulency (Huang, et al., 2021).

Rather than solving the problem numerically using BEM, one method that has shown to be promising and it is widely studied recently is by using volumetric methods, also known as Computational Fluid Dynamics (CFD). The first time this approach has been applied to a ship section's slamming was in the studies of Arai et al. (1994), where the Euler solver, considering inviscid and incompressible fluid, and the fractional volume of fluid method (VOF) was used to compute the transient deformation of the free surface, and compared with the present analytical and experimental results. Moreover, the development of these CFD methodologies made it possible to be one of the few methods that account for viscosity with the option to couple a turbulence model to solve the Navier-Stokes equation.

The two CFD branches approaches that can be applied to these contexts are the mesh-free and mesh-based methods (Huang, et al., 2021). One of the most applied methods for the meshless segment is the Smoothed Particle Hydrodynamics (SPH). Overviews from Monaghan (1992, 1998, 2005) were given on this robust method. The studies that could be performed using this method done by using simple geometries (Sun, et al., 2018; Gong, et al., 2009), which is not only due to the complexity of capturing geometries and modeling boundary effect using particles (Huang, et al., 2021), but also the fluid-body interfaces, numerical damping, which prevents correct wave propagation in wide domains, and internal pressure oscillations, which are visible in local pressures. Even though the solver is quite quick because no big matrices need to be inverted, and each particle only interacts with its immediate neighbors, the time step is quite small since, effectively, compressibility is employed for time-stepping, which makes this method worse in the aspect of computational requirements (Kapsenberg, 2011).

As for the mesh-based volumes, also known as the Finite Volume Method (FVM), it is a method that relies on the discretization of the fluid domain. It is able to account for complex structures by a closed curve (for a two-dimensional approach) or surface (for a three-dimensional approach) in contact with numerous neighbor cells (Huang, et al., 2021). This approach allows not only be applied to predict fluid behaviors, structural loads, and deformations (Pena, et al., 2019) with good accuracy and with viscous and turbulent flows being modeled (Khojasteh, et al., 2020), but can also account for and compare different effects on slamming impacts. The studies of the effects of compressibility and air cavity were investigated for a free-falling wedge cylinder were made using the software OpenFOAM in Wang & Guedes Soares (2020), for instance, and the effects of three-dimensionality were further researched using the same software in Wang et al. (2021). These distinct aspects not only were studied but also validated with the recent experimental studies of the same wedge from Wang et al. (2015), Armand and Cointe (1986), and also, the analytical results from Korobkin (2005). Most of these studies only dealt with two-dimensional structures and the application for them is in the strip theory, widely used in ship motions research (Wang, et al., 2021).

One challenge for the FVM in the context of slamming is the large displacement that the solid body can face, especially in the case of free-falling objects or structures. There is a limit on the body's offset in the deformation mesh method because a large amplitude motion will distort the cells and crash the simulation. There are some ways to get around this. One of them is by considering the whole domain to move while the body is fixed, which was done in the studies of Huang et al. (2020) for resistance. The problem with this approach is applicable when the motion is prescribed, but for free-falling structures, it should be defined for each interaction. Another approach is the one used, for instance, in the studies of Wang et al. (2021), where the structure is placed right above the waterline, avoiding large motion amplitudes. Although this can be fairly applied for small structures and when the velocity before hitting the water is well-known, for a lifeboat the motion starts from a big dropping height and with a considerable air resistance to be accounted for before water entry.

Another approach to deal with this challenge is to incorporate an advanced mesh approach. One of them is the Immerse Boundary Method (IBM), which considers the body as a closed wall boundary moving along the domain and deactivates the internal fluid cells, while the external ones are computed. The original IBM, however, does not have specialized boundary cells outside the moving wall (Huang, et al., 2021; Mittal & Laccarino, 2005), and this oversimplifies the boundary-layer effect of the geometry and generates uncertainties in the analysis. Zheng et al. (2020) used IBM to simulate the water entry process of a wedge body and applied a ghost-cell approach to improve the original IBM's ineffective boundary layer modeling, and with results in good accordance with the benchmark experiments of Yettou et al. (2006). On the other hand, a widely applied method in CFD is the overset (chimera) mesh, where two domains are created: one for the background containing information about the fluid and another, for the geometry (chimera). Water entry studies using this approach have been validated and shown to have proven to be adaptable and effective for movements of high amplitude (Shen, et al., 2016; Ma, et al., 2018). This makes it appear as though the overset mesh approach is better appropriate for the slamming problem.

The sequence of the spatial and temporal discretization methods differs, despite the fact that the majority of CFD solvers use identical governing equations. Additionally, mesh topology, turbulence-related assumptions, and boundaries all produce discrepancies. The sources of uncertainty linked to CFD computations are increased by these variations. Stern et al (2001) and Coleman and Stern (1997) went into great detail on the simulation's numerical uncertainties in CFD as well as the verification and validation studies. Roy (2005) provided a thorough review of computational simulation verification techniques. Round-off error, iterative convergence error, and discretization error are the three primary causes of numerical error in the computational simulation. Generally speaking, discretization error has been identified as the primary contributor to numerical errors in a computer simulation. Oberhagemann (2016) stated that it is essential to be aware of the grid sensitivity of a specific flow problem in order to generate sound numerical solutions.

Uncertainty analysis is a methodical investigation done to evaluate the reliability and accuracy of a CFD solver in resolving a particular problem. Verification and validation are two techniques that are part of uncertainty analysis. The process of verification involves figuring out whether a model implementation accurately depicts the implemented method and the model's solution. Validation is the process of assessing how well a model's prediction corresponds to observed physical phenomena. As a result, validation evaluates the solver's accuracy, while verification evaluates the solver's consistency (Wang, et al., 2022). The Correction Factor (Stern et al., 2001), the Factor of Safety (Celik et al., 2008), and the Least Square Fit approach (Eça and Hoekstra, 2014) are three of the most often used techniques to analyze grid and time step uncertainty. The Richardson extrapolation method forms the foundation of the first two approaches. The ITTC (2008, 2017) guidelines for uncertainty measurement in CFD studies recommend all of these methods.

Falk, A.E., a Swedish designer, initially proposed the idea of a free-fall lifeboat in 1897. Falk obtained a patent for a design for an enclosed lifeboat with stern-sliding capabilities. Capt. White of the Bay and River Navigation Company put forth the idea for an "unsinkable submarine lifeboat" thirty years later. The first manned launch of a free-fall lifeboat from the stern was from m/s Tarcoola at Oresundsavarvet Shipyard, which was first witnessed by the nautical community until 1977 (WÄRTSILÄ). Nowadays, Free-fall lifeboats are widely recognized as a mean of evacuation for ships and offshore petroleum installations (Ré, et al., 2008). Similar to other vessels and crafts, the lifeboats are subject to IMO regulations and requirements, especially SOLAS that has human safety as priority.

Chapter 3, Reg. 4.4 of LSA Code states that “all lifeboats shall be of sufficient strength to enable them to safely launch when fully loaded with their full complement of person and equipment”, and “[…] Be capable of launched and towed when a ship is making headway at a speed of 5 knots in calm water”. Besides, even when the ship is heeled to an angle of 15 degrees on either side, the gravity davits still need to retain and slide down the lifeboat, and centrifugal brakes are used to limit the lifeboat's descent speed, which should not exceed 36 m/min. Other general requirements are also applied depending on the ship type and size. In Chapter 3, Reg. 4.7 of LSA code, part from the general requirements of lifeboat, there are additional requirements for free-fall lifeboats. One of the relevant ones for the launching are that “each free-fall lifeboat must make positive headway after entering the water and shall not come into contact with the ship after a free-fall launching against a trim of up to 10° and a list of up to 20° either way”. It should also have a factor of safety of 6 based on the ultimate strength of the materials used in design. Besides other requirements on occupation during launching, release system and painting, the certificate of approval for a free-fall lifeboat must state the free-fall certification height, the required launching ramp length, and the launching ramp angle for the free-fall certification height. This project deals specially with the model Schat Harding FF 1000, which has freefall height of 36 m and strength tested from 47 m and is mostly used for platforms and large ships (Harding, 2014). The design angle is not specified in the product catalogue, but is mentioned as 50° in Ringsberg, et al. (2017).

Regarding specific studies on free-fall lifeboats, a few can be mentioned. What is one of the largest sources is the one within the OLF-project, where an extensive model test program was performed for 14 distinct types of free-fall lifeboats with over 25,000 tests (Kauczynski, et al., 2009), which is still used nowadays for validating the different methods to assess this free-fall. For instance, the study of Ringsberg et al. (2017) presented a benchmark study intending to demonstrate the practical use of quasi-response methods for the assessment of impact loads on modern FFLBs, where eight different calculation methods were compared based on analytical plate strip models (linear and nonlinear idealizations), and FE models of different configuration and complexity (quasi-static linear, GNL, linear-elastic and transient dynamic). Qiu et al (2020) have established a mathematical model using the strip theory and Kane’s method, where the FFLB motion is calculated from the beginning in the skid until the water entry, and after compared to the star-CCM+ simulation results. Huang et al (2021) have developed a model based on Computational Fluid Dynamics to holistically simulate and analyze the water entry process, applying overset mesh technique and not only comparing it with experimental data but also studying the influence of changing the dropping height and inclined angle.

1.3 Motivation, objectives, and structure of this thesis

As is well known, the high complexity of the governing equations and the boundary conditions makes it difficult to find an analytical solution for the fluid-structure interaction problem, used for assessing slam-induced loads. Also, the intricacy and high cost of performing full-scale experiments is also an obstacle for being the ideal method to evaluate these loads. The alternative to these is to perform numerical simulations, in especial the CFD, obtaining approximated results for the real loads.

The objective of the study is to simulate the entire process of the freefall of a lifeboat during its launching and investigate the trajectory, pitch angle, velocity, acceleration, and pressures of the boat using OpenFOAM. This software has the advantage of being open source and having convenient solvers for the slamming problem of lifeboats, especially because it can deal with multiphase simulations and also the recent versions have the overset grid technique, which is suitable for simulations with high amplitude motion.

First, Chapter 1 presents the current state of the slamming problem and advancements made in previous years. In Chapter 3, the software OpenFOAM is described with emphasis on the methodology applied in the simulations of the water entry. A description of the numerical modelling of slam-induced loads on two-dimensional structures follows the introduction of the explicit finite element approach for the study of fluid-structure interaction in Chapter 3. This approach is expanded for the three-dimensional simulation of a real lifeboat model for several initial dropping heights and angles in chapter 4. Finally, Chapter 8 draws conclusions from all the studies performed.

2. REVIEW ON CFD METHOD AND SOFTWARE OPENFOAM

Open Fields Operation and Manipulation, also known as OpenFOAM, is a C++ toolbox for the development of customized numerical solvers, and pre-/post-processing utilities for the solution of continuum mechanics problems. It is employed in both academia and industry to address a broad range of computational and engineering problems. The source code is accessible and changeable, contrasting to any proprietary software.

For the purpose of this project, a multiphase simulation based on the VOF method is used. This method is relevant for engineering applications since it is applied when two immiscible fluids set the scenario studied (Manafpour & Hamzeh, 2017). Furthermore, this solver is applied over the chimera meshing technique so the solver can deal with large amplitude motion, which is the case of the majority of slamming cases.

2.1 OpenFOAM multiphase solver

The tool from OpenFOAM that supports the VOF method is the interFoam family of solvers. In especial, for overset grid technique, the overInterDyMFoam is used, which stands for overset and dynamic meshing version of interFoam. For two or more immiscible fluids, the VOF method is a surface-tracking method where the location of the fluid interface is of interest. The fluids in this model share a single set of momentum equations, and the volume fraction of each fluid in each computational cell is monitored over the entire domain.

Besides the usual physical constants of the fluids, the VOF method relies on another scalar and dimensionless parameter known as volume fraction, α, where $\alpha \in [0{,}1]$. It is defined as step function in which the cells where $\alpha = 1$ contains water and $\alpha = 0$ contains air, and the interface is smeared over the cells with α in between these two values. With this, the physical formulation of this problem of two isothermal, incompressible, and immiscible fluids is based on the continuity, momentum, and interface capturing advection equations below, respectively (Albadawi, et al., 2013).

$$\frac{\partial \rho}{\partial \mathrm{t}} + \nabla \cdot (\rho \boldsymbol{V}) = 0 \tag{2.1}$$

$$\frac{\partial (\rho V)}{\partial t} + \nabla \cdot (\rho \boldsymbol{V}\boldsymbol{V}) = -\nabla P + \nabla \cdot \tau + \rho \boldsymbol{g} + \boldsymbol{F_\sigma} \tag{2.2}$$

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \boldsymbol{V}) - \alpha(\nabla \cdot \boldsymbol{V}) = 0 \tag{2.3}$$

where ρ is the fluid density, $\boldsymbol{V}$ is the fluid velocity vector, τ the viscous stress tensor defined as $\tau = 2\mu S = 2\mu(0.5[(\nabla \boldsymbol{V}) + (\nabla \boldsymbol{V})^T])$, μ the fluid dynamic viscosity, P the scalar pressure, $\boldsymbol{F_\sigma}$ the volumetric surface tension force, $\boldsymbol{g}$ the gravitational acceleration vector, and α the interface capturing. These fluid properties vary along the domain with the parameter α in each cell, and for any intrinsic fluid property Φ can be written as a function of α as a weighted average (Larsen, et al., 2019).

$$\Phi = \alpha\Phi_{water} + (1-\alpha)\Phi_{air} \tag{2.4}$$

The evolution of α can be obtained by simplifying the interface capturing advection equation (2.3) using the continuity equation (2.1) and definition (2.4).

$$\frac{\partial \alpha}{\partial \mathrm{t}} + \nabla \cdot (\alpha \boldsymbol{V}) = 0 \tag{2.5}$$

The algorithm also faces the numerical challenge of preserving the conservation of mass while keeping the interface sharp, which is addressed by using an algebraic approach based on restraining the phase fluxes with the "Multidimensional universal limiter with explicit solution" (MULES) limiter. This adds a heuristic consideration of including a compressive term to the advection equation.

$$\frac{\partial \alpha}{\partial \mathrm{t}} + \nabla \cdot (\alpha \boldsymbol{V}) + \nabla \cdot (\boldsymbol{V_C}\alpha(1-\alpha)) = 0 \tag{2.6}$$

where V_c is the compressive velocity, considered only in the interface between air and water to avoid dispersion, and assessed by Equation (2.7), in which c_α is compressive factor used to increase compression and $\boldsymbol{\hat{n}_c} = \frac{\nabla\alpha}{|\nabla\alpha|}$.

$$\boldsymbol{V_c} = \min(c_\alpha|\boldsymbol{V}|, \max(\boldsymbol{V}))\boldsymbol{\hat{n}_c} \tag{2.7}$$

Without using density averaging, the Continuum Surface Force model (CSF) is used to derive the volumetric surface tension force in the VOF approach (Albadawi, et al., 2013).

$$\boldsymbol{F_\sigma} = \sigma\kappa(\alpha)\nabla\alpha \tag{2.8}$$

The interface curvature, $\kappa(\alpha)$, is based on the updated α value of an after advection, and it depicts the magnitude of the interface normal flux at a particular cell face, and it also shows the flux's direction.

$$\kappa(\alpha) = -\nabla(\boldsymbol{\hat{n}_c} \cdot \boldsymbol{S_f}) \tag{2.9}$$

where S_f is the surface vector of the cell face, in which f stands for the cell surface. This is also used when dealing with wall boundary conditions, where a constant static angle θ_0 should be achieved at each time step when determining the phase fraction This correction is done before calculating the curvature (Albadawi, et al., 2013).

Regarding the floating body, it is modelled as a free rigid body where the forces considered are the gravity and the surface forces of pressure and shear stress (Benites-Munoz, et al., 2020). At each time step, the six DoF solver from OpenFOAM performs the integration of pressure and viscous stress component over the wetted surface S_B to assess the resultant force and moment around the CG.

$$\boldsymbol{F} = m\boldsymbol{g} + \int_{S_B}(-p\boldsymbol{I} + \tau) \cdot \boldsymbol{n}\, dS \tag{2.10}$$

$$\boldsymbol{M} = \int_{S_B} \boldsymbol{r} \times (-p\boldsymbol{I} + \tau) \cdot \boldsymbol{n} \, dS \tag{2.11}$$

where $m\boldsymbol{g}$ is the gravitational force over the body, p is the normal pressure, $\boldsymbol{I}$ is the unit tensor of size 3, τ is the viscous stress tensor, $\boldsymbol{n}$ is the normal vector to the surface and $\boldsymbol{r}$ is the position vector with respect to the CG. The accelerations are obtained by dividing both resultants to their respective inertia term in accordance with the motion constraint applied if any. With these accelerations, velocity and displacement can be integrated by the Newmark integration.

$$\dot{x}_{t+\Delta t} = \dot{x}_t + (1-\gamma)\Delta t \ddot{x}_t + \gamma \Delta t \ddot{x}_{t+\Delta t} \tag{2.12}$$

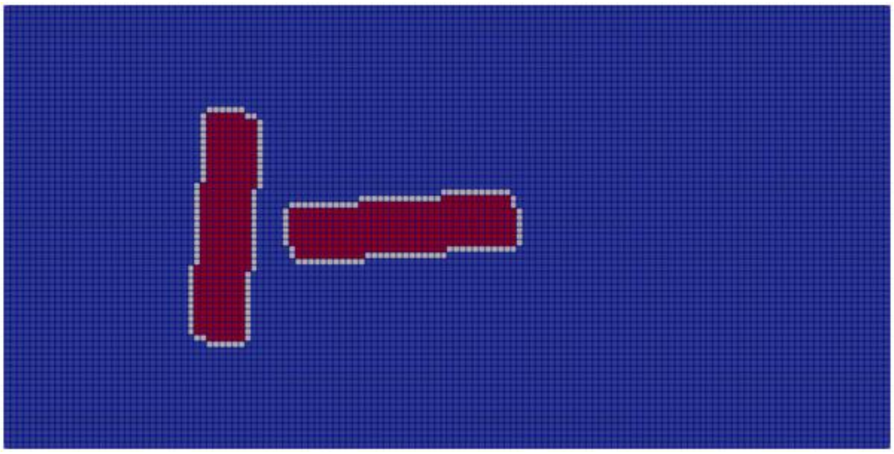
$$x_{t+\Delta t} = x_t + \Delta t \, \dot{x}_t + \frac{\Delta t^2}{2}[(1-2\beta)\ddot{x}_t + 2\beta \ddot{x}_{t+\Delta t}] \tag{2.13}$$

where x is the position or angle of a given degree of freedom, $\dot{x}$ its velocity (first derivative) and $\ddot{x}$ its acceleration (second derivative), Δt is the time step, t is a given instant while $t + \Delta t$ is the next step. The recommended values of $(\gamma, \beta) = (0.5, 0.25)$ are used as OpenFOAM default (Benites-Munoz, et al., 2020).

2.2 Overset meshes

A general implementation for the use of unconnected (also known as Chimera) meshes is the overset framework in OpenFOAM, where meshes that are static and dynamic are both supported. In this approach, two independent and disconnected meshes are created: the background and overset (chimera). This method is very helpful in situations involving mesh motion and interactions. It avoids the problems and instabilities associated with deforming meshes (Tisovska, 2019).

The principle of this meshing approach is to give, at each time step, a label to each cell of both domains, where it describes if the cell is calculated, where the equations are solved for this cell; hole, where there is no computation for this cell; or interpolated, which is when the value is computed from the nearest elements of the other domain (background or overset). This categorization is illustrated in Figure 2.1.



.

Figure 2.1 – Cell types where the red cells are holes, the white are interpolated and the blue, calculated

Between the cells, there are the donors, which are the ones that provide values, and the acceptors, whose value gets set from interpolation. In these cells, a field ϕ (i.e., pressure) is determined by weighted average of this same field from the neighbour donor cells ϕ_i.

$$\phi = \sum_{i=1}^{n} w_i \phi_i \tag{2.14}$$

The normalized weights w_i and which cells are donors depends on the interpolation method. The simplest one and also the fastest that is considered in this project is the "cellVolumeWeight", which uses weights proportional to the volume of the acceptor cell inside a given donor cell and normalized to the total volume of the acceptor.

2.3 Overview of simulations in OpenFOAM

2.3.1 Operation and inputs

OpenFOAM is usually used in Linux operational systems. To use it on a Windows computer, an Emulator is required. For this project, a Linux computer from CENTEC is used and it is accessed remotely by the SSH terminal emulator PuTTY. A typical workspace of OpenFOAM consists of a directory named after the project's title.

In one of these directories, there should be one directory per domain used in a simulation. In the case of overset grit technique, it would be two: one for the background and another for the floating body. Each of these contains the following directories.

- **0:** containing the information regarding the initial and boundary conditions of the simulation. This directory only exists in the main domain, which is the background, in this case.
- **constant:** miscellaneous data regarding physical constants (such as gravity, fluid density, and viscosity, etc) and, when simulating objects other than the fluids (such as the lifeboat) stl files with the object surface and the main information and conditions of this object in the simulation (such as mass, inertia, degrees of freedom, etc).
- **system:** the core of the simulation where the mesh, the ambient, the solver, and its main features are described.

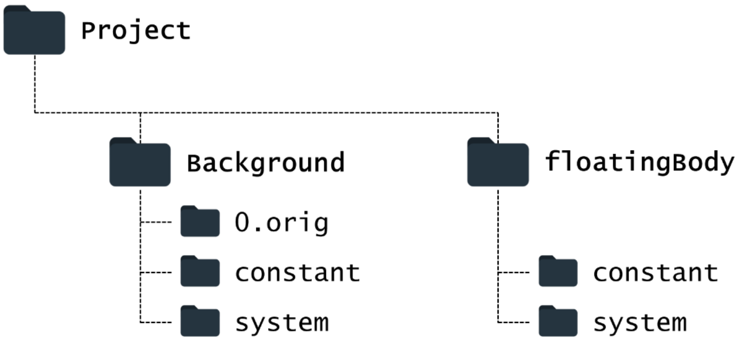


Figure 2.2 - Usual arrangement of an OpenFOAM overset grid workspace

Each of these directories contains several plain text files with input information in accordance with what is required by each function or by the simulation itself. Most of these files used have the default entries by one of the multiphase tutorials as input. The ones that are modified or defined with the simulation are further described in this chapter.

2.3.2 Mesh generation

One of OpenFOAM's possibilities is to create the mesh domain to be used in the simulations. This is created by the command "blockMesh", where the mesh description should be in a text file named "blockMeshDict" inside the directory "system".

The domain is created based on the scale of the simulation. The first step is to define the 8 vertices of the block-shaped domain in the cartesian space. The definition of this process in the simulation is important especially because the boundary faces of the domain are considered to be walls by OpenFOAM. In freefalling simulations, the object entry generates waves that may be reflected when reaching the domain boundary. Since this effect is undesired in the simulation, the domain must be large enough, so that the waves generated do not influence the motion.

After defining the domain boundaries, it is necessary to mesh them. In this study, the cells are hexametrical, which is defined by providing the number of divisions in each cartesian direction: x, y, and z. There is the possibility of creating a homogeneous mesh or with a constant expansion ratio. In a direction, the number of cell divisions can be also distributed not equally, which is suitable for slamming problems where the water entry region should be refined.

The final configuration that is provided to the "blockMeshDict" is the boundary, in which each face of the domain block (patches) is identified and assigned a certain type. There are many types convenient to each situation, but for this project, the three that are used are:

- **Patch:** generic type lacking any geometry or topology details regarding the mesh. It is usually used for an inlet or an outlet.
- **Wall:** for a patch that coincides with a solid wall, required for some physical modelling. Used for the bottom and the floating body itself.
- **Empty:** for solutions in less than 3 dimensions (2D/1D), the type used on each patch whose plane is normal to the dimension(s) for which no solution is required.

The last patch boundary face type is usually used to perform two-dimensional simulations. In these simulations, the domain itself is not two-dimensional, but an extrusion of the 2D situation with a certain thickness. The empty patch allows the solver to neglect the dimension that is not considered.

In addition to the "blockMesh" command, the "snappyHexMesh" is also used for the mesh generation. This utility creates 3-dimensional meshes containing hexahedra (hex) and split-hexahedra (split-hex) automatically from triangulated surface geometries, or tri-surfaces, in Stereolithography (STL) or Wavefront Object (OBJ) format. By iteratively improving a beginning mesh and morphing the resulting split-hex mesh to the surface, the mesh approximately conforms to the surface (Greenshields, 2015).

In order to use this command, the file snappyHexMeshDict should be located in the system directory. It contains several information regarding the refinement of the object surface, which depends on how fine or coarse the background mesh was defined previously in blockMeshDict. The important aspect of this command is the stl file that should be generated before running it and located in constant/triSurface directory. This can be created using CAD software.

2.3.3 Setting fields and boundary conditions

The final step before running the solver is to set the background by using “setFields” command. The “setFieldsDict” is the text file containing the main information regarding it and is located in a system directory. The input is simpler, and it just requires which space of the domain should contain water, while the rest is considered to have air, and fills each cell with its volume fraction α.

Also, besides the setFieldsDict, the setFields command needs the text files inside the 0 directory, regarding the initial condition. In this directory, there are several files containing the boundary condition of a certain physical quantity (e.g., hydrostatic pressure, velocity field, etc) of the patches (boundary). There are basic and derived conditions that can specify a fixed value or a normal gradient of the patch field and can consider the normal gradient to be null or to be calculated from other patch fields and other derived or similar conditions. For the empty 2D front and back, the empty condition is applied.

One of the basic physical quantities that require the specification of the conditions is the alpha.water, which turns into a binary file that specifies whether the cell of the mesh is air (0) or water (1) after running setFields. For this, the condition of zeroGradient is applied to every patch, except the atmosphere, in which the inletOutlet is applied. This boundary condition is the same as the zeroGradient, but it switches to fixedValue when there is fluid flowing inwards the domain. For this case, the fixed value is 0, which means that the phase fraction is fixed with a value of 0, corresponding to 100\% air.

There is also the p_rgh related to the hydrostatic pressure. For this, all the patches usually have the zeroGradient condition, except the atmosphere, which has the totalPressure. In this case, the total pressure p_0, which also accounts for the dynamic pressure $p_0 = p + \rho U^2/2$, remains constant by the specified value of p_0. So, for a change in the velocity U, the pressure p is adjusted accordingly.

As for U, it is related to the velocity field of the fluid. For walls such as the bottom, the fixed value of (0,0,0) is applied, or, for recent versions of OpenFOAM, the noSlip condition, in order to represent the no-slip principle. For the floatingObject, the movingWallVelocity is applied, which sets the relative velocity to the moving wall to a specified value (similarly, **0**). For the atmosphere, the pressureInletOutletVelocity is considered, which specifies zeroGradient at all times, except on the tangential component, which is set to fixedValue for inflow, defaulting to 0.

For patches where some inflow occurs but the inlet flow velocity is unknown, it is typical to combine the totalPressure condition for pressure and pressureInletOutletVelocity for velocity, usually applied for atmosphere patch. This combination allows the normal velocity to find its own value when there is inflow and applies a standard condition combination for outflow. Under these circumstances, a rapid rise in velocity presents a risk of instability, but the rise is moderated by the reduction of inlet pressure, hence the driving pressure gradient, as the inflow velocity increases (Greenshields, 2015).

In addition to these specifications, there is also the pointDisplacement file that is more important for the floatingObject patch. Every patch, except this one has a fixed value of (0,0,0), while this patch has the calculated condition. This is important because this defines that the solver will calculate the motion of the body based on its dynamics. For overset mesh simulation, there is zoneID which identifies which domain a certain cell belongs. All of the patches usually are zeroGradient except the empties. A summary of the boundary conditions can be seen in .

Table 2.1 - Summary of the boundary conditions

	alpha. water	**p_rgh**	**U**	**pointDisplacement**	**zoneID**
inlet and outlet	zeroGradient	zeroGradient	noSlip	fixedValue (0,0,0)	zeroGradient
front and back	empty	empty	empty	empty	empty
atmosphere	inletOutlet	totalPressure	pressureInletOutletVelocity	fixedValue (0,0,0)	zeroGradient
bottom	zeroGradient	zeroGradient	noSlip	fixedValue (0,0,0)	zeroGradient
floating object	zeroGradient	zeroGradient	movingWallVelocity (0,0,0)	calculated	zeroGradient
overset sides	overset	overset	overset	overset	overset

If the simulation is 3D, the front and back conditions will be similar to the inlet and outlet. Similarly, to the empty condition, the overset condition is also special and is used to the walls of the overset domain boundary. For some simulations, it is also possible to use the symmetryPlane when the simulation is set for one side of the symmetryPlane. This should replace all fields in the inlet or outlet boundary.

2.3.4 Running the simulation

The group of commands described previously is the pre-processing of all data stored in the three directories into files that the software will use. After running them and checking the mesh and background, the solver can be run.

For the purpose of running the solver, there are files that need attention for good and correct performance. One of them is related to the object to be studied, which is the “dynamicMeshDict” in the system directory. This file contains information regarding the specific solver used for the body motion (which, for this case, would be the “sixDoFRigidBodyMotion”), data about the dynamic constants of the body, such as mass, inertia matrix, the centre of mass, etc, and constraints on the motion (if it is restricted in one direction, point, plane, etc). Usually, for 2D simulations, the plane restriction is applied, while for free fall of symmetric objects, the axis is restricted to move only vertically to avoid horizontal motion due to numeric error and for faster performance.

Another important file located in the system directory is the “controlDict” file, which is used for case-control. This file contains the core of the simulation in which information regarding the solver is to be used, the time step of the time discretization, the start and end time, if the data files are in binary or ASCII format, etc. In especial for multiphase solvers, the time step is set to be adjusting to a maximum

Courete number, which is a measure of the rate at which information is transported under the influence of a flux field and given by $Co = \frac{u\Delta t}{\delta x}$, where u is velocity, Δt is time step and Δx is length interval.

One important aspect of the "controlDict" file is the functions. Some specific data can be taken as a separate output with these functions and can be used for analysis and plot creation. Some of these functions are: "Probes" or "patchProbes"**,** used for gathering the time series data of certain physical quantities in a specified point. In the case of studying the free fall of objects, it would be relevant to obtain the pressure on the surface of it; and **Force,** used to obtain the pressure, viscous and total force over a patch, which, in the case of this study, would be for the free-falling object.

3. SLAMMING LOAD ON THE TWO-DIMENSIONAL WEDGE DURING WATER ENTRY

For a first approach of using the open-source software OpenFOAM, simple cases were run to validate the output results. Since this study is focused on the slamming phenomena, the simulations needed to test a freefall in a multiphase background, with water and air. Also, the overset mesh method is tested, since this method is applied to investigate large-amplitude motions. The description of the case study, its results, discussion, and conclusions are further described in the following sections.

3.1 Initial setting and motivation

3.1.1 Description of the experimental test

One simple case that is widely studied in the slamming context is the water entry of a prismatic wedge on water. The study performed in Wang et al (2015) develops numerical methods and experiments to investigate the progressions of the pressure distribution on the impact and top sides, air-water interfaces, and wedge kinematics. These results of pressures and wedge motion are used to compare them with the numerical results from the simulation. The wedge is symmetric with 48 mm height and a deadrise angle of 30°. The dimensions and arrangement of the pressure sensors from this study can be seen in Figure 3.1.

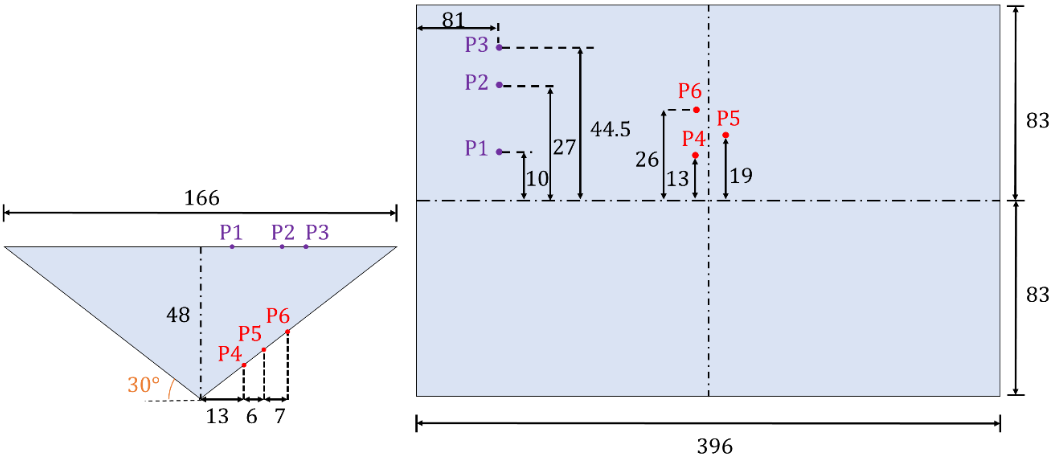


Figure 3.1 - Arrangement of the prismatic wedge and sensors from Wang et al. (2015). Units in mm.

For this study, only the pressure sensors on the impact surface are considered, which are P4, P5, and P6, which are the ones entering the water and taking the high pressures at first. Since they are close to the middle, the effect due to the border can be neglected, which is important for comparison with a two-dimensional simulation.

The study case for this validation has an initial velocity of 2.50 m/s before hitting the water, which is the case 4 from Wang et al. (2015). Another particularity of the experiment is the total mass of 12.8 m of the wedge, leading to a mass distribution per unit length of 32.3 kg/m. This data is important to make the right mass conversion to the two-dimensional simulation.

3.1.2 Description of the simulation setting

The open-source CFD software OpenFOAM is used to perform simulations of the dropping wedge in two-dimensions. It was already used before to study this same case by Wang et al. (2021). The difference is the overset (chimera) mesh approach used here, which divides the discretization of the background domain, responsible for containing information regarding the fluids, and the overset mesh, for the solid body.

First, the model considered for the study is two-dimensional and half of the scenario (wedge and background), is simplified by using the boundary condition of the symmetry patch. This consideration in important due to the considerable reduction of cells to be meshed, which also reduces the computational effort and time. With this, the general mesh with background, waterline, and wedge is modelled as shown in Figure 3.2.

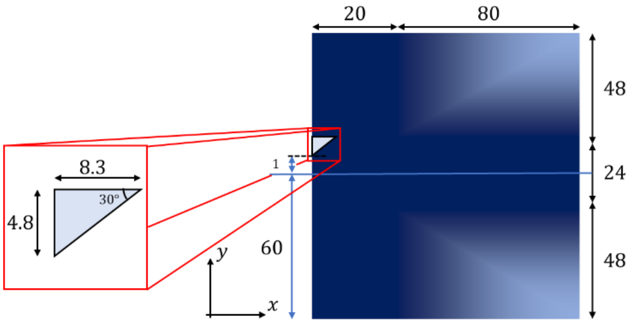


Figure 3.2 - Background meshing with dimensions in cm, where darker regions has denser mesh

Some aspects are important to be emphasized in this approach. The wedge starts with an initial velocity above the water, and the 0.01 m of positive vertical offset is given for the simulation to not start with a disturbed state. Also, the region where the water entry occurs has a denser mesh, since it is where all the events occur, and it is more important to be refined for better results. This refinement is done previously with blockMesh by setting a percentage of cells per axis and expansion ratio. This distribution in the xy plane is shown in Figure 3.3.

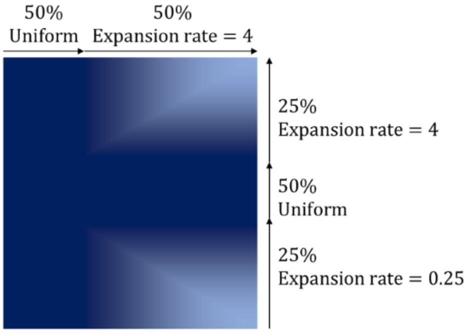


Figure 3.3 - Cell distribution along the xy plane

An important consideration is the way OpenFOAM performs two-dimensional simulations. Even so, the simulation is still 3D with a certain thickness in the z-direction. What makes the simulation 2D is the combination of being a small thickness (of 0.01 m considered in the simulations), and the empty boundary condition given to both sides of the domain parallel to the xy plane, which represents the directions that are not supposed to be solved (Greenshields, 2015). As for the other sides, the boundary conditions are summarized in Figure 3.4, where the atmosphere and wall are the standard ones in OpenFOAM.

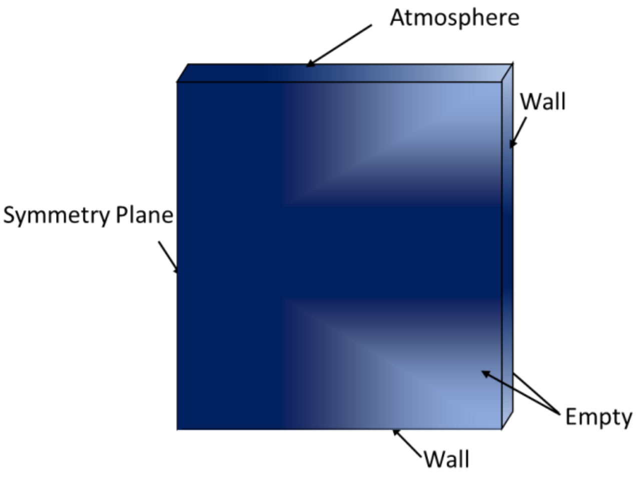


Figure 3.4 - Summary of boundary conditions for each side (patch)

What concerns the freefalling wedge, the tool of overset mesh was used. This was made by using the command blockMesh to create a mesh with cells divided in hexahedrons and snappyHexMesh to remodel the existing mesh into the desired geometry. In the case of this study of external flow, the mesh is modelled externally from the solid body. This new domain was designed to be about 2 times bigger than the wedge, has uniform meshing, the same z-direction thickness, and with same cell density and size as the water entry region mesh. The dimensions and boundary conditions for the overset domain are shown in Figure 3.5.

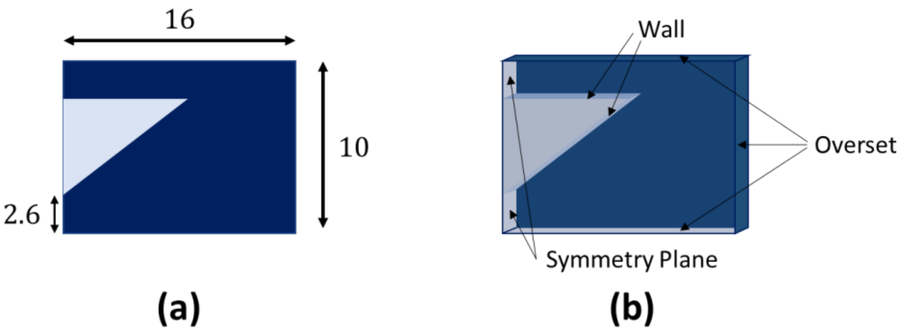


Figure 3.5 - (a) Overset domain's dimensions in cm and (b) boundary conditions

The patches that represent the wedge (the ones with wall boundary conditions in Figure 3.5 b) have more specifications due to the solid body nature. For the simulations, these patches were treated as a rigid body, where the 6 degrees of freedom OpenFOAM solver is applied to obtain the solution of the wedge's motion. Also, the motion is restrained to occur in the vertical direction only, since it is a two-dimensional simulation of a symmetric freefall. The mass is also defined based on the 32.3 kg/m result given in section 3.1.1, which for half a section with a thickness of 0.01 m gives 0.1615 kg. One final adjustment to avoid spinning is inputting the vertical center of mass to be close to the keel vertex (about 1 cm height) and high moment of inertia (about 10^{16} kg-m^2).

Table 3.1 - Summary of the wedge parameters

Parameter	Value
Width (x length)	83 mm
Height (y length)	48 mm
Thickness (z length)	10 mm
Deadrise angle	30°
Mass	161.5 g
Initial velocity	-2.5 $\vec{j}$ m/s

Another aspect to be described regarding the simulations input is the multiphase distribution. As seen in Figure 3.2, the waterline is input in $y = 0$, which means that every cell below this will be given an alpha value of 1, meaning water, while everything above, 0, meaning air. The simulations are also set to be incompressible laminar flow since the speed is negligibly negligible compared to the speed of sound and the low velocity of the small-sized body results in a small Reynolds value, which makes this solver suitable for this study case (Wang et al., 2021). The constants and properties used for these simulations are further described in Table 3.2.

Table 3.2 - Constants and properties considered for the simulations

Parameter	Value
Gravity	-9.8065 $\vec{j}$ m/s^2
Water density	998.2 kg/m^3
Water kinematic viscosity	1 mm^2/s
Air density	1 kg/m^3
Air kinematic viscosity	1.58 mm^2/s

Finally, it is important to mention that three simulations were performed with different mesh sizes and following the Figure 3.3 division. They are described by the smallest cell size in the water entry region, which are 4 mm (coarse), 2 mm (medium), and 1 mm (fine), defined also by a constant refinement ratio of 2.

3.2 Results and discussion

3.2.1 Overview of the simulations before post-processing

Before checking and discussing the results obtained after the simulations, it is important to know how the simulations behaved when facing this study case with different meshing densities. The simulations were performed on a regular desktop equipped with an Intel Core i7-4570 @ 3.2 GHz with 94.1 GB of RAM memory and using 4 parallel simulations. Further details of the meshing and CPU time can be seen in Table 3.3. It is notable that with the increase in the number of cells, the CPU is also more demanded by taking longer to process the whole simulation.

Table 3.3 - Cells details and CPU time of the simulations

Label	**Minimum Cell [mm]**	**CPU [h]**	**Cell number background**	**Cell number overset**
Coarse	4	0.11	14,751	2,751
Medium	2	1.07	59,008	11,008
Fine	1	37.34	236,032	44,032

Also, regarding the chimera mesh meshing, it is important to check if the wedge patches were properly meshed since it is an iterative meshing tool for cell removal and adaptation. A refinement box was used to further improve the snappy meshing process. The results of this meshing for the medium simulation can be seen in Figure 3.6.

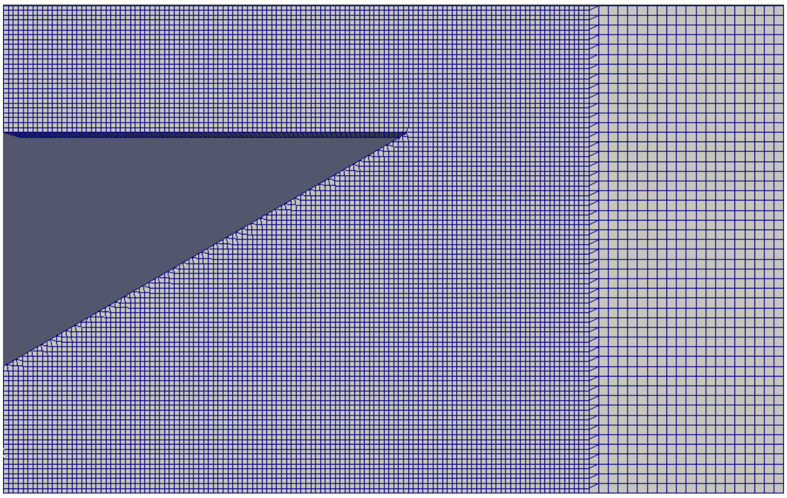


Figure 3.6 - Details on the wedge section with refinement (medium meshing)

Also, to register the slamming peaks for pressure, velocity, and accelerations for the initial velocity of 2.5 m/s, the simulations were run from t_i = 0 s to t_f = 0.06 s, which is sufficient even according to Wang et al. (2015) results. The time step was set as $\Delta t = 10^{-5}$ s, adjustable for a maximum Courant number of 0.25, as done in Wang et al. (2021).

3.2.2 Pressure sensors

The first results that can be used as a parameter for validation are the pressure on the probes sensors input as described in Figure 3.1. For this comparison, two validation data are used, the experimental measurement, called "experimental" in the next plots, and the two-fluid Boundary Element Method (BEM), called as "numerical", both results obtained by Wang et al. (2015) studies. The CFD results by using three overset meshes setting are shown and compared in the Figures below.

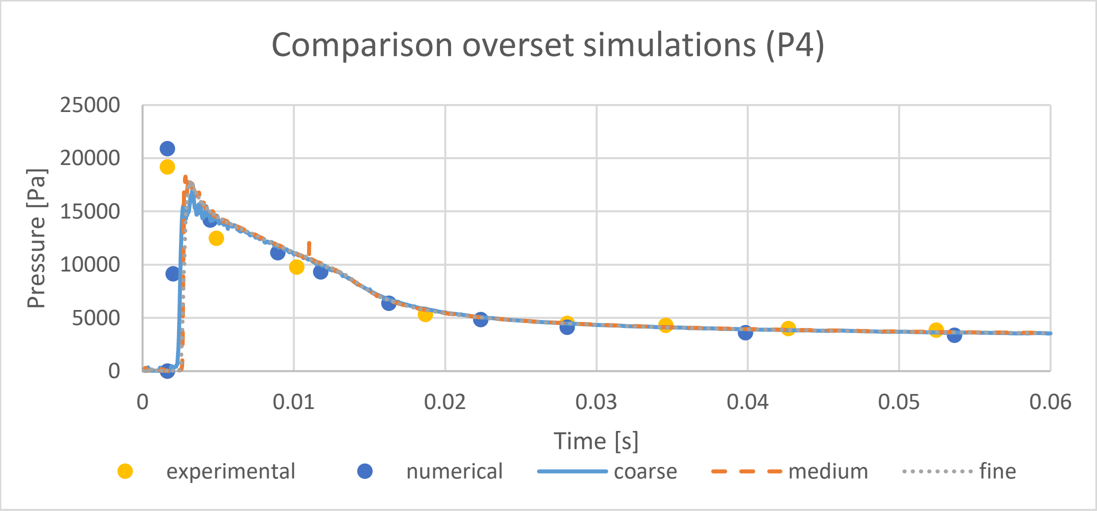


Figure 3.7 - Comparison of pressure results for sensor P4

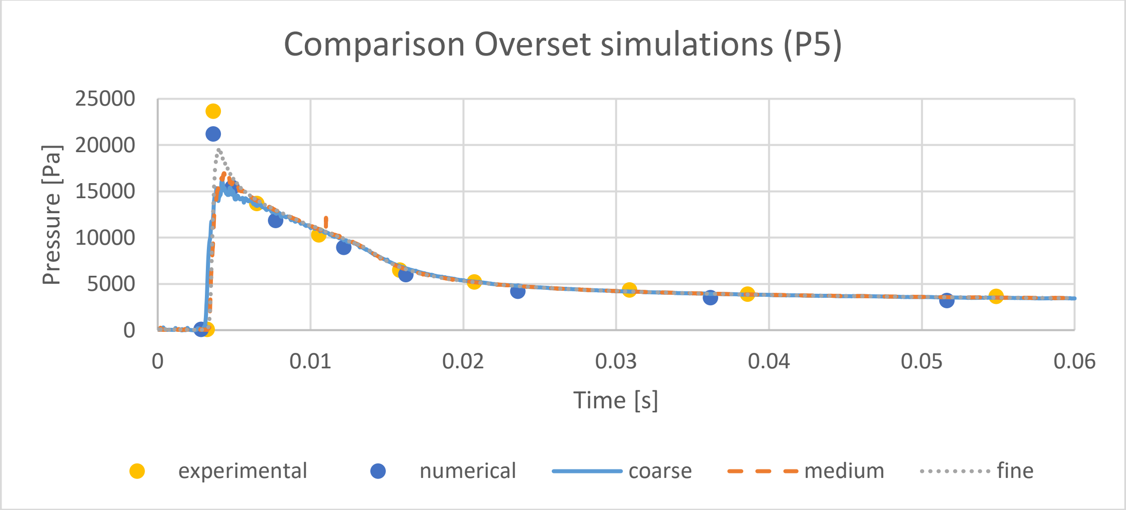


Figure 3.8 - Comparison of pressure results for sensor P5

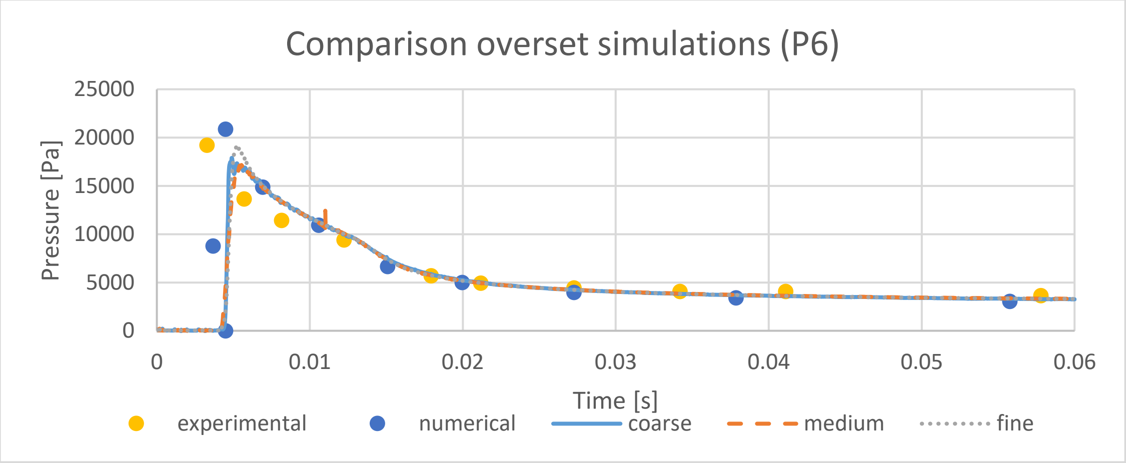


Figure 3.9 - Comparison of pressure results for sensor P6

What is notable, at first, is that the more refined is the mesh, the clearer the result gets, especially at the peak. This occurs, of course, due to the better-quality mesh, capable of better describing the continuous ambient as more discretized it gets. In general, the results are also in good agreement with the three approaches. The punctual differences that appear are the appear delay to the beginning of the ascension and the peak, which can be explained by the initial shift given to not disturb the background mesh in the simulation. Also, the peaks differ from each other due to the high uncertainties present in the experimental and numerical analysis, as concluded by Wang et al. (2021), which can also be noted in the large uncertainty bar presented in the results of Wang et al. (2015).

3.2.3 Wedge's kinematics

What was also presented as a result of the studies of Wang et al. (2015) is the kinematics of the wedge, including acceleration, velocity, and wedge draft. The acceleration comparison can be seen in Figure 3.10, as well as vertical velocity in Figure 3.11 and draft in Figure 3.12.

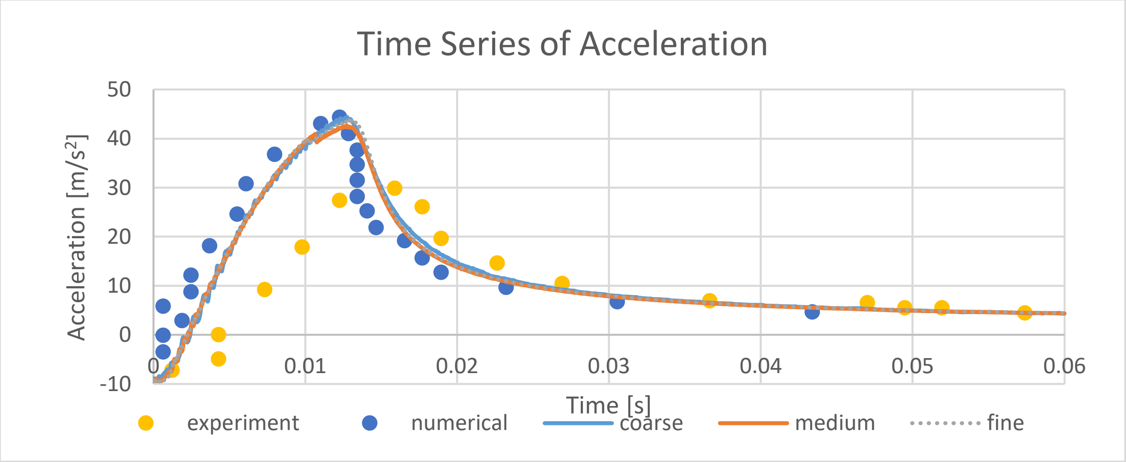


Figure 3.10 - Wedge acceleration comparison

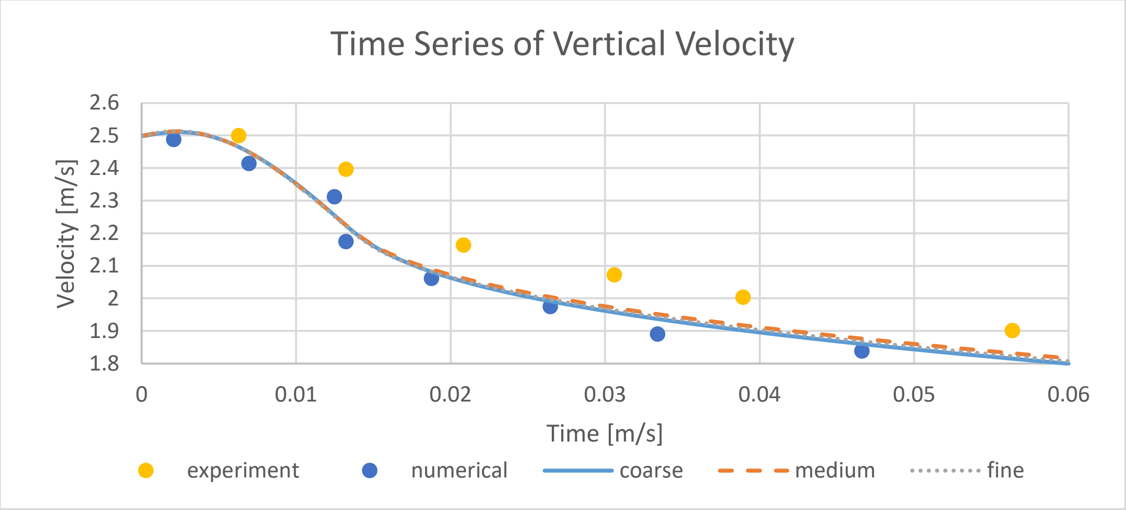


Figure 3.11 - Wedge velocity comparison

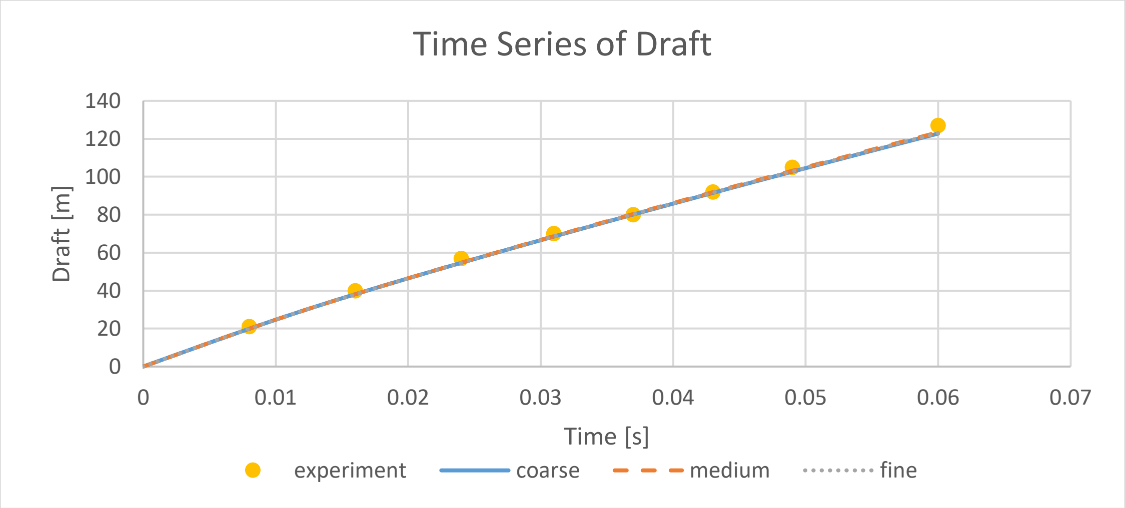


Figure 3.12 - Wedge draft comparison

In the draft time series, the experiment data was taken by using a software that takes data from image plots. Due to the high similarity between the experimental results and the numerical ones, just the experimental were considered.

What is notable in all three cases is that the 2D simulations are in better agreement with the numerical solutions, even considering the three different meshes. As for the difference concerning the experimental result and the numerical ones, the acceleration peak is notably higher, resulting also in differences in behaviour in the velocity time series. This has to do with three-dimensional effects neglected in the numerical methods, which were tested in Wang et al. (2021) by performing the same simulation but considering it three-dimensional and applying the actual length in a 3D domain. What is shown is that the acceleration peak reduces significantly near the experimental results, which means that this divergence is expected, and this model should agree better with the two-dimensional BEM numerical approach, which has few deviations due to potential flow consideration.

3.2.4 Uncertainty analysis

In order to perform the convergence research and quantify the discretization errors, the constant Courant-Friedrichs-Lewy (CFL) number is used, along with the approach of a correction factor based on Richardson extrapolations based on ITTC (2008 and 2017) standards. First, a set of at least three simulations of a given study based in a certain situation. with systematic parameter refinement, preferentially with the same refinement ratio, defined by the dividing discretization parameter $\Delta x_{i,m}$, which in the case of this project is cell size.

$$r_i = \frac{\Delta x_{i,2}}{\Delta x_{i,1}} = \frac{\Delta x_{i,3}}{\Delta x_{i,2}} \tag{3.1}$$

Another important parameter to determine convergence is by calculating the convergence ratio. Considering measurements $\hat{S}_{i,1}$ (fine), $\hat{S}_{i,2}$ (medium) and $\hat{S}_{i,3}$ (coarse) and their variation between consecutive $\varepsilon_{i,12} = \hat{S}_{i,2} - \hat{S}_{i,1}$ and $\varepsilon_{i,23} = \hat{S}_{i,3} - \hat{S}_{i,2}$, the convergence ratio is defined as:

$$R_i = \frac{\varepsilon_{i,12}}{\varepsilon_{i,23}} \tag{3.2}$$

This parameter defines the convergence, where $0 < R_i < 1$ states for monotonic convergence, $R_i < 0$ states for oscillatory convergence and $R_i > 1$, divergence. In the generalized Richardson extrapolation, by using integer powers of Δx_i as a finite sum, the error is expanded in a power series expansion. The number of terms kept in the expansion, the magnitude (importance) of the higher-order terms, and the estimates' accuracy all have an impact on the accuracy. Only the leading term, which yields one-term estimates for the error $\delta^*_{RE_{i,1}}$, and order of accuracy p_i, can be predicted when there are three solutions.

$$p_i = -\frac{\ln(R_i)}{\ln(r_i)} \tag{3.3}$$

$$\delta^*_{i,1} = \frac{\varepsilon_{12}}{r_i^{p_i} - 1} \tag{3.4}$$

For good practice, approximate and extrapolated relative errors are calculated for each mesh refinement, as well as the Grid Convergence Index (GCI), which is a measurement of the distance, in percent, between the computed value and the asymptotic numerical value.

$$e_a^{21} = \frac{\hat{S}_2 - \hat{S}_1}{\hat{S}_1} \tag{3.5}$$

$$e_{ext}^{21} = \frac{(\hat{S}_2 - \hat{S}_1)\left(r_i^{p_i} - 1\right)}{r_i^{p_i}\, \hat{S}_2 - \hat{S}_1} \tag{3.6}$$

$$GCI^{21} = \frac{1.25\varepsilon_{21}}{r_i^{p_i} - 1} \tag{3.7}$$

The uncertainties associated can be assessed by a factor of safety approach (Roache, 2003), which can be used to define the uncertainty U_i where Equation (3.4) is multiplied by a factor of safety

FS = 1.25 for careful grid studies to bound simulation error. The factor of safety strategy, albeit not suggested by Roache (2003), can be employed in circumstances when the answer is corrected with an error estimate from RE and also calculate a corrected uncertainty, U_{ic}.

$$U_1 = F_S \left|\delta^*_{RE_{i,1}}\right| \tag{3.8}$$

$$U_{1c} = (F_S - 1)\left|\delta^*_{RE_{i,1}}\right| \tag{3.9}$$

With this, the maximum acceleration and maximum slamming pressure coefficient, $C_p = p(t)/[0.5\rho V^2(t)]$, where $p(t)$ and $V(t)$ are the instantaneous pressure and velocity, and ρ is the water density, are analysed by this method, and the summary is shown in Table 3.4.

Table 3.4 - Uncertainty calculation for maximum acceleration and pressure coefficient Cp

Uncertainty calculation		Maximum acceleration [m/s^2]	Maximum C_p
Output values	Ø$_1$ (fine)	43.77	6.801
	Ø$_2$ (mid)	42.66	6.925
	Ø$_3$ (coarse)	44.41	5.948
Refinement ratio	r	2.00	2.00
Convergence ratio	$\epsilon_{21}/\epsilon_{32}$	-0.6345	-0.1275
Order of accuracy	p	0.6564	2.9709
Approximate relative error	e^{21}_a	0.0254	-0.0183
	e^{32}_a	-0.0410	0.1411
Extrapolated relative error	e^{21}_{ext}	-0.0422	0.0027
	e^{32}_{ext}	0.0767	-0.0202
Grid convergence index (GCI)	GCI^{21}_{fine}	2.80E-05	0.011
	GCI^{32}_{fine}	9.27E-05	0.018
Uncertainty	U_1	0.014%	0.335%
	U_2	0.046%	2.578%
Corrected Uncertainty	U_{1c}	0.003%	0.067%
	U_{2c}	0.009%	0.516%

The convergence ratio for both is negative for both parameters, indicating that the solution is oscillatory convergent. Also, the numerical uncertainties in the fine-grid solution U_{21} are 0.014% for the Acc_{max} and 0.335% for the C_{Pmax}. As for the values of U_{32} (using medium and coarse models), the uncertainties are all still below 3%. With the correction factor, the uncertainties U_{32C} are all below 1%. For the maximum acceleration, it is notable that $U_2 > U_1$, indicating that the fine resolutions do not always improve the simulation results.

3.2.5 Comparison with deforming mesh method

The same scenarios were investigated in Wang et al. (2021), but with another approach. Instead of chimera meshes, the researchers applied the morphing mesh using the same software for reducing computational expenses. The simulations from this investigation were recreated using parallelization to reduce the computational time. Small differences between the minimum size and number of cells are seen between both due to the separated treatment given to both chimera and background domains, but a comparison can be already made by the computational time of each case, shown in Table 3.5.

Table 3.5 - Comparison between computational times, in hours, of morphing and overset mesh methods

	Deforming Mesh	Overset Mesh	% Difference
Coarse	0.035	0.11	214%
Medium	0.462	1.07	132%
Fine	4.896	37.34	663%

What is already seen is that the overset mesh method requires more of the CPU than the morphing method, even by using 4 parallel processors to run the simulations. This happens mostly due to the drawback of additional computational load introduced by the interpolation process between domains (Berton, et al., 2017).

Other comparisons are made between the dynamic and kinematic results of the wedge water entry problem. The first is the pressure from the sensors P4, P5, and P6, as shown in the Figures below, by considering the finest mesh of both methods. In general, the results are in accordance with each other, and differences are observed mostly at the peaks. While for P4 and P5 this difference remained at 3.4% and 4.4%, respectively, for P6 this was more discrepant, with about a 10% difference.

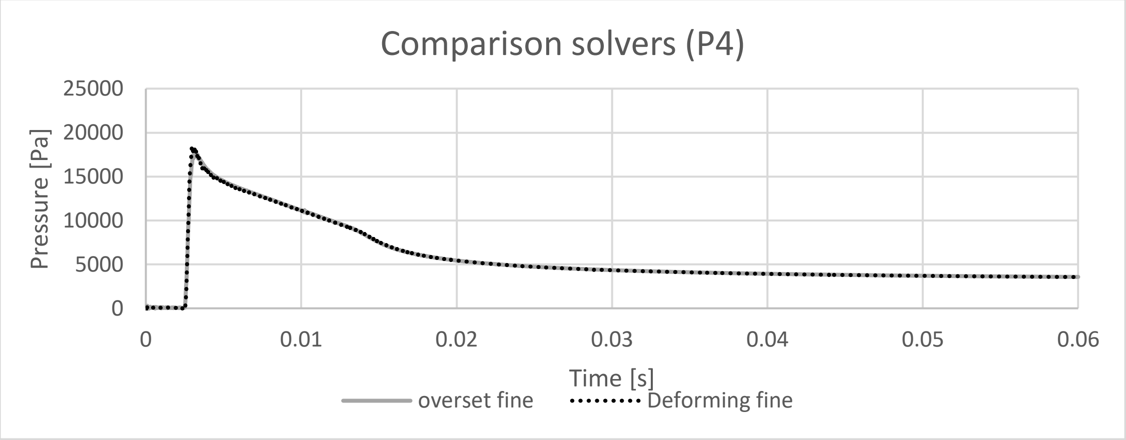


Figure 3.13 - Pressure on P4 comparison for the two fine meshes of overset and deforming method

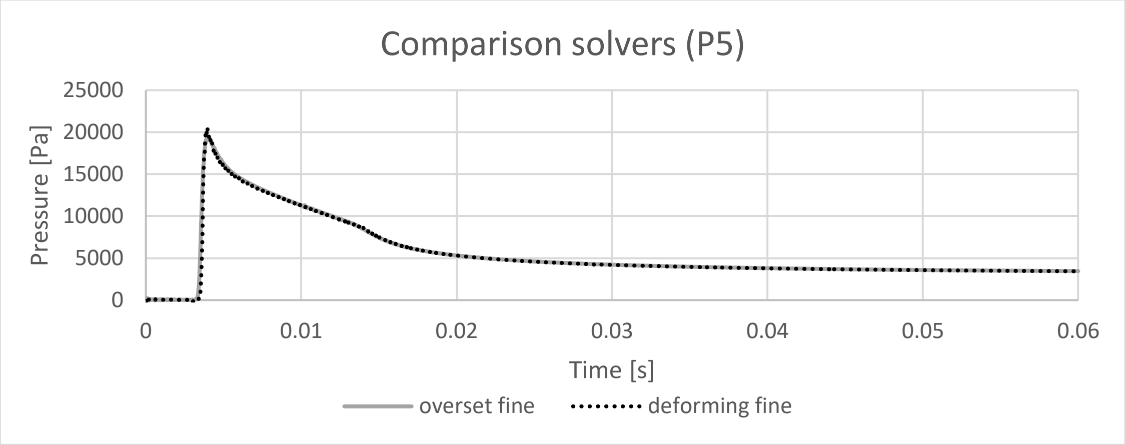


Figure 3.14 - Pressure on P5 comparison for the two fine meshes of overset and deforming method

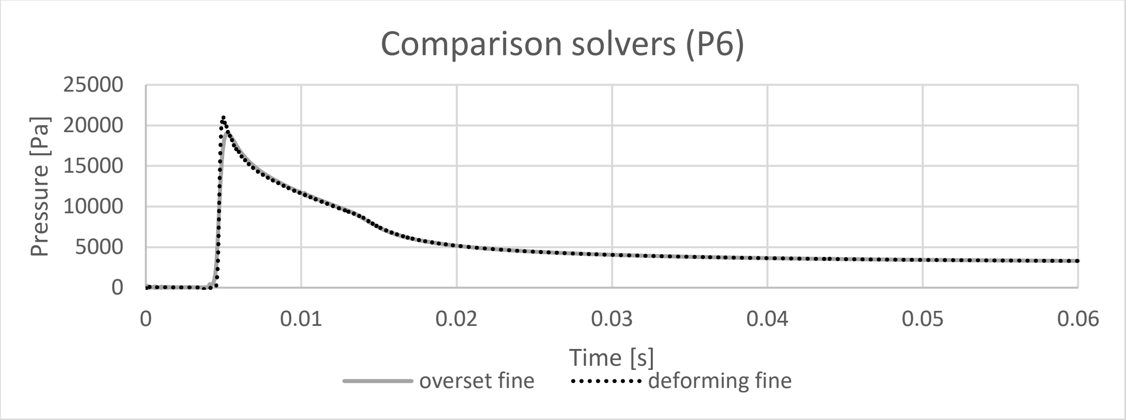


Figure 3.15 - Pressure on P6 comparison for the two fine meshes of overset and deforming method

Another parameter that can be used in this analysis is the kinematics, which the comparisons are shown in the plots below. There are differences of 4.2% between the peaks of acceleration but mostly show the same results. This variation is even less perceptible in the velocity and penetration results shown in Figure 3.17 and Figure 3.18, respectively.

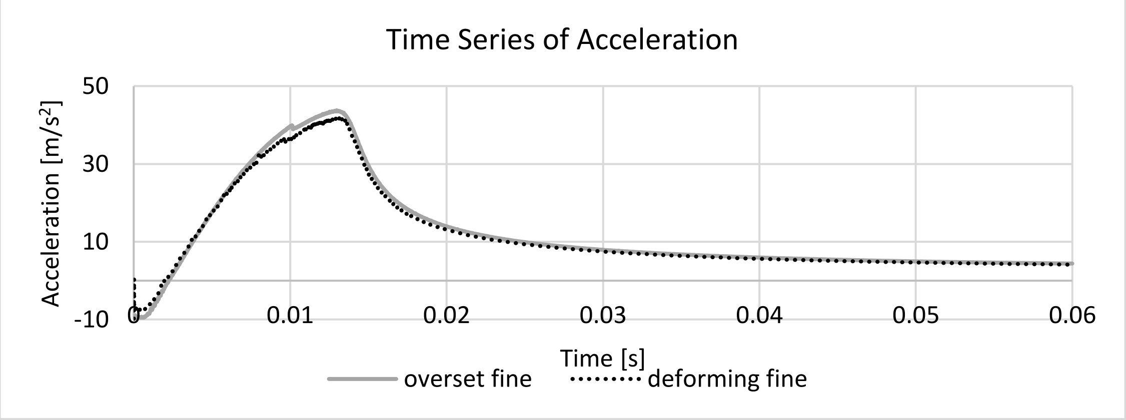


Figure 3.16 – Acceleration comparison for the two fine meshes of overset and deforming method

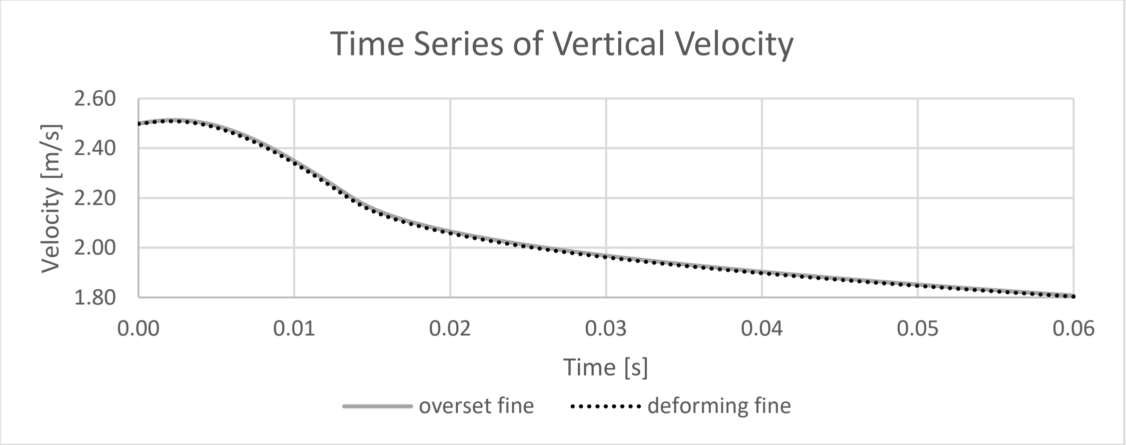


Figure 3.17 – Velocity comparison for the two fine meshes of overset and deforming method

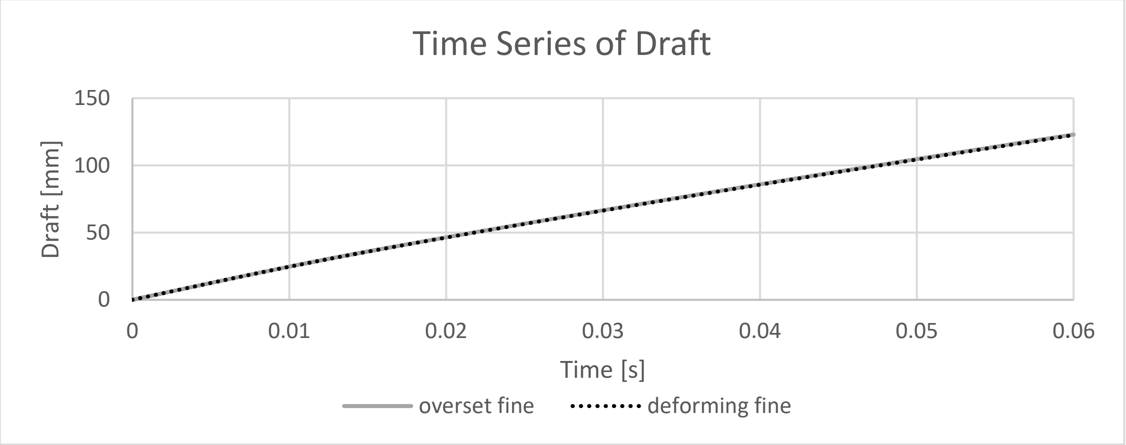


Figure 3.18 – Draft comparison for the two fine meshes of overset and deforming method

One final comparison is the contour plots of the water and air. This is done by using Paraview to post-process the results from OpenFOAM. In this case, the specific output shown is the fluid fraction of the VOF method (α_i) and can be seen in Figure 3.19. Similarly, the results are also mostly the same with the resolution overset mesh being slightly better due to the structured mesh in the background and near-body (Lopez Mejia, et al., 2021) without being deformed.

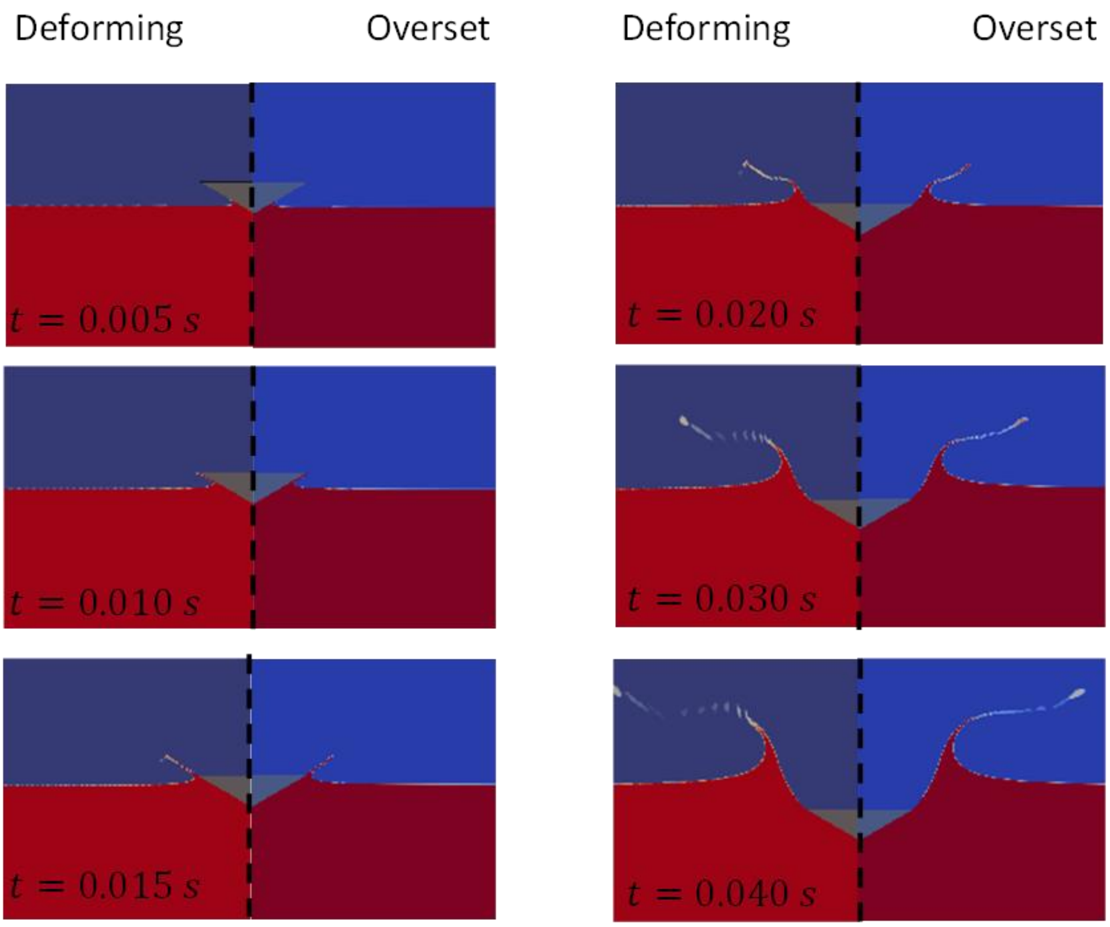


Figure 3.19 - Contour plots comparison for both finest meshing types

In general, both methods agree with each other without jeopardizing the final result that a CFD simulation should show, which is observed in most comparison studies between these two solvers (Alletto, 2022; Lopez Mejia, et al., 2021). Some differences are observed mostly at the peaks. These regions are known to variate to a great extent for how refined the mesh and time step are, and the difference between resolutions observed in Figure 3.19 and also, in the results related to dynamics, such as acceleration (that comes from force integration) and pressures, can be an explanation for that together with the convergence velocity, which tends to be faster for overset mesh approach (Lopez Mejia, et al., 2021).

3.3 Conclusions

Numerical two-dimensional simulations of water entry of a 30° deadrise wedge are carried out by the opensource software OpenFOAM. The main results are compared with experimental and numerical solutions from the previous study made on this same wedge with the objective of validation of the overset mesh approach. Pressures and kinematics parameters are compared in this study.

The overset mesh is an approach where the ambient with fluids and the dropping body is meshed in two different domains, called background, and overset (chimera), respectively. The background is generated by the meshing command and setting each cell to be water or air. The solid body is created by removing and adapting cells iteratively to gain the shape of the solid body provided by a stl.

Overall, the pressures were in accordance with the experimental and numerical results obtained from Wang et al. (2015). The main divergences occurred due to an initial delay of the 1 cm offset given to not disturb the background domain. Also, the peaks had divergences, but the uncertainty from both, numerical and experimental results, is too high, and these kinds of differences are expected.

As for the kinematics parameters: acceleration, velocity, and displacement, the simulations agreed more with the BEM numerical solution, which is expected due to the same assumption of two-dimensional freefall from an infinite wedge with a certain mass linear density, while the experiment is a wedge with the same linear density, but with a 3D and finite dimension, where the border effect is not negligible.

Uncertainty analysis was also carried out to analyse the convergence of the parameter's maximum acceleration and pressure coefficient. What is found is that both have oscillatory convergence and have uncertainty below 5% and even below 1% by applying correction factors, and not necessarily a better resolution will give better results, as shown by the maximum acceleration uncertainties.

Comparisons between morphing and overset mesh for this type of scenario were also made by recreating the results from Wang et al. (2021). Mostly the results were the same and differences can be spotted at peaks of pressure, which is usually more conservative for deforming mesh, and in the resolution of the results, which has shown to be better for overset mesh, which happens due to the structured and undeformed background and near-body. Even though the computational time for morphing mesh was fairly lower than a chimera, so in this scenario where the motion amplitude does not crash the simulation, both presented to be robust methods to assess this problem and can be used to examine other hydrodynamic impact issues and model wedges of various forms.

4. LIFEBOATS CFD SIMULATIONS

The previous chapter discussed the validation of the overset mesh applied to the slamming. This first study performed simulations over a simpler case well-known in the literature. In this chapter, the simulations are performed in a 3D scenario and progress for application on free-fall lifeboats.

To evaluate the possibility of performing the loads' assessment and the vessel's motion prediction, a case study is done by using a model of a lifeboat and a simpler scenario of how its free-fall would occur by just considering the journey from the free-fall event (as in Figure 1.2) and neglecting the initial velocity and spin given by the skid, which means that the lifeboat will start from rest, at a dropping height and with initial falling angle.

4.1 Initial setting and pre-processing

4.1.1 Lifeboat model

The first step before setting the simulations is to define the lifeboat that will serve as the object of study. In this field of research, one typically used model is the Schat Harding 1000 (Ringsberg et al., 2017; Huang, et al., 2021). This has to do with the experimental studies made on the full-scale model (Kauczynski, et al., 2009) and is mostly used to compare them with the simulation results. The bodyline of the lifeboat is shown in Figure 4.1 (b).

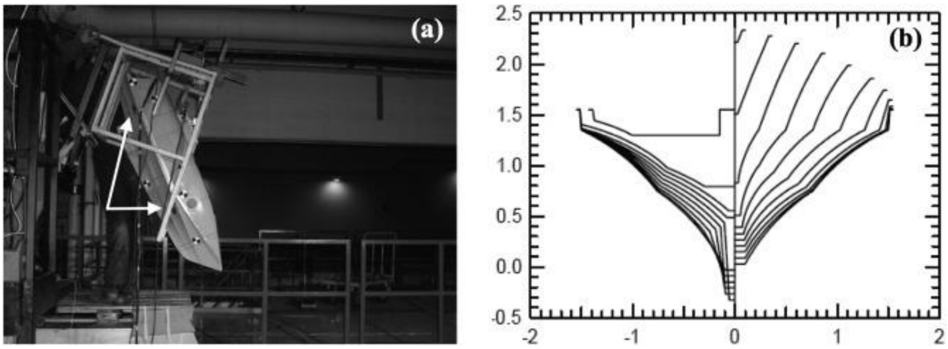


Figure 4.1 – (a) Lifeboat skid setup; (b) Lifeboat's body plan (Ringsberg, et al., 2017)

Table 4.1 – Main particulars of the full-scale lifeboat (Ringsberg, et al., 2017)

Parameter	Value
Overall length [m]	12.57
Overall width [m]	3.34
Displacement [ton]	16.8
LCG forward of stern [m]	5.29
Radius of gyration in pitch [% of L_{OA}]	25%

For this project, however, it was not possible to use the exact same model due to copyright matters. Even so, the geometry could be reproduced by using the body plan from Figure 4.1 (b) and the side view of the ship in Rhinoceros from Figure 4.2. For this, the contours of the chines were taken from the sideview, and their position was taken from the horizontal line interpolation given by the body plan. The tridimensional contour was the intersection between extrusions.

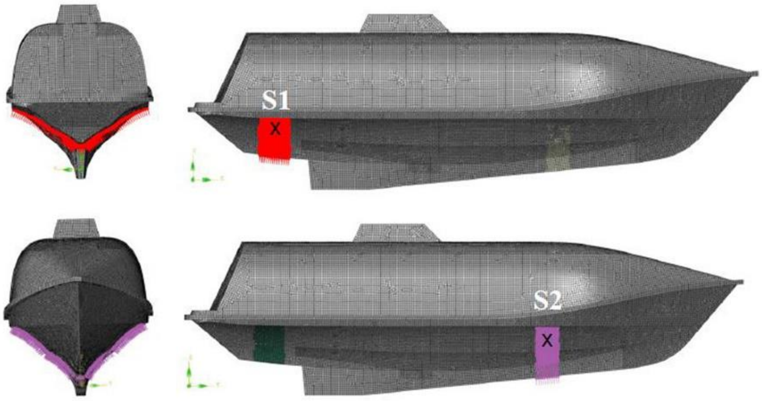


Figure 4.2 - Front and sideview of the lifeboat model (Ringsberg, et al., 2017)

In this process, still, although the general shape of the hull can be taken, some original information is lost along it. The network surface command from Rhino can make a good job on creating a surface between the chines contour, but it might not reproduce the full curvature of them as seen in Figure 4.1. Also, even in Figure 4.2, the sizes of both sideviews do not match, meaning that distortions from the hull's original dimension can appear. Finally, the upper part of the hull is not detailed in the body plan, so the reproduction was based on Figure 4.2 by connecting network surfaces, parts of cylinders, and planes. The final reproduction can be seen in Figure 4.3.

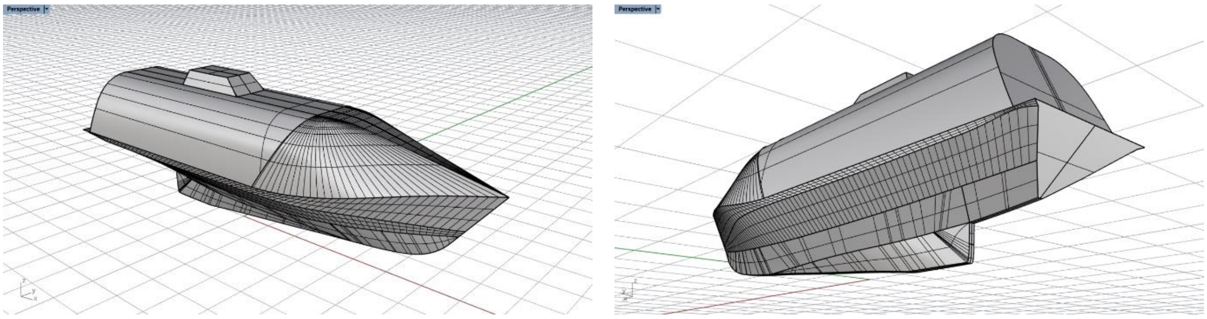


Figure 4.3 - Reproduction of the lifeboat hull surface in Rhinoceros

Even with these problems, it was possible to create an enclosed surface in Rhino that can serve as a model for this study. Both measures of length overall and beam diverted by about 5.4% from the original measures (by increment for length and decrement, for beam), which is considered acceptable for this approach. With this, a stl file containing this geometry could be exported for the meshing process.

4.1.2 Description of the simulation setting

Overall, the domain and mesh are set with similar philosophy from section 3.1.1 with the major difference of being a 3D simulation, and instead of empty patches, there are going to be two more walls on the back and front. The lifeboat also starts from rest with an initial dropping height, H , which 10 m, 20 m and 30 m are used, and a falling angle of α, where simulations from 10° to 70° with 10° of step are used, totalizing 21 simulations. 80° Simulation was initially intended to be performed, but the simulation diverged, meaning that the critical angle for converging solutions is in between 70° and 80°. The scenario, axis convention, and cell concentration division are illustrated in the Figures below. The origin is set to be on the waterline with coordinates (x, y) equal to the centre of gravity, and the cells are chosen to have about 40 cm in the water entry region and chimera.

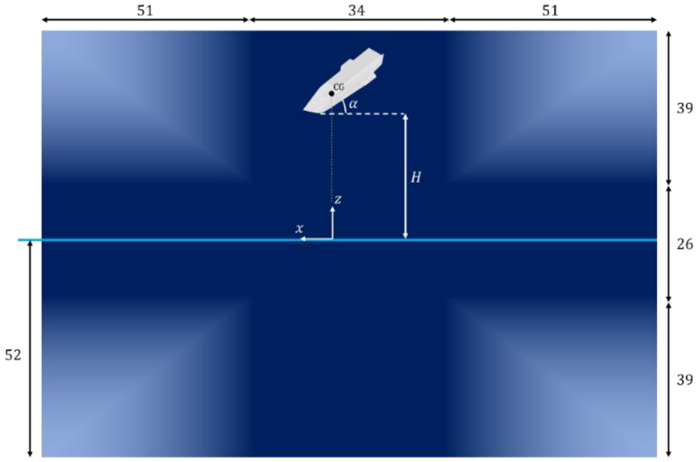


Figure 4.4 - Sideview of the background domain for the lifeboat. Dimensions in meters.

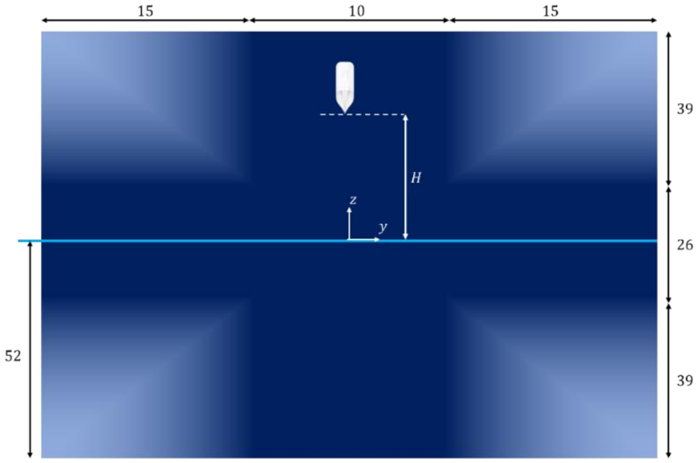


Figure 4.5 - Sideview of the background domain for the lifeboat. Dimensions in meters.

The division and distribution of the cells also follow what is proposed in Figure 3.3, but also considering the other side of the symmetric plan (which means the proportion in each axis is 25%, 50% 25%). The physical constants are also the same in Table 3.2. Although this situation would end in a turbulent flow, the simulation is set as laminar to use a more stable solver with fewer crashing chances.

As for the overset mesh, the domain is illustrated in Figure 4.6, and is also set to be about two times larger than the dimensions of the lifeboat. The cells are also set to be uniform with the same dimensions as the water entry region.

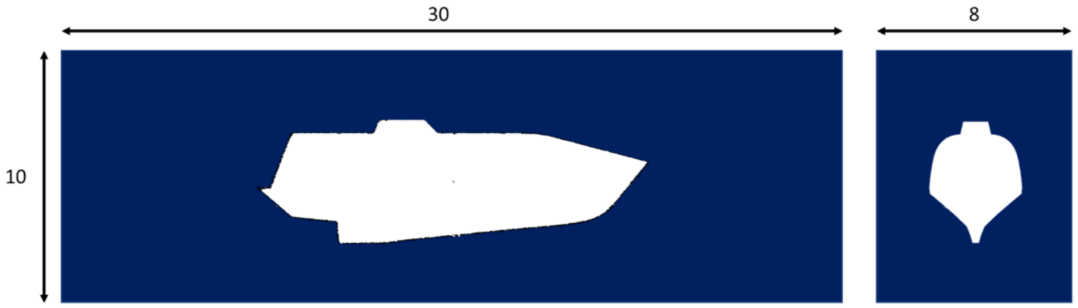


Figure 4.6 - Dimensions of the overset domain in meters

One important aspect is the time step used, which was 10^{-3} s, and to capture sufficient details for a water entry problem, the Courant number is set to be always smaller than 0.3 based on the analyses of Muzaferija (1999). 5 seconds of journey, starting from the beginning of the freefall

4.2 Results and discussion

4.2.1 Preliminary analysis

Similar to what is done in section 3.2.1, an overview of the general simulation can be done with comments and considerations on it. First, regarding the mesh and computational effort, an Intel Core i9-4570 @ 3.2 GHz with 126 GB of RAM memory was used, and since these simulations are expected to demand a lot more than the wedge did, they were parallelized into 27 processors. The general overview of the simulations' parameters is shown in Table 3.1

Table 4.2 - Summary of the simulations' components of the lifeboat

Parameter	Value
Background cells	1,141,292
Overset cells	36,292
Minimum cells [cm]	40
Time [h]	About 4 h

What is already noticeable is how much more demanding these simulations get. Even though the simulation time was reasonable, this was just possible by using almost 7 times more parallel processors and also using a mesh less refined with larger minimum cell size. This is expected due to the increment of one dimension in the simulations and also in the motion of the body, increasing the complexity of the calculations for an overset approach. What also influences this is the meshing. In the case of the overset, which has the snapping process, the result can be seen in Figure 4.7.

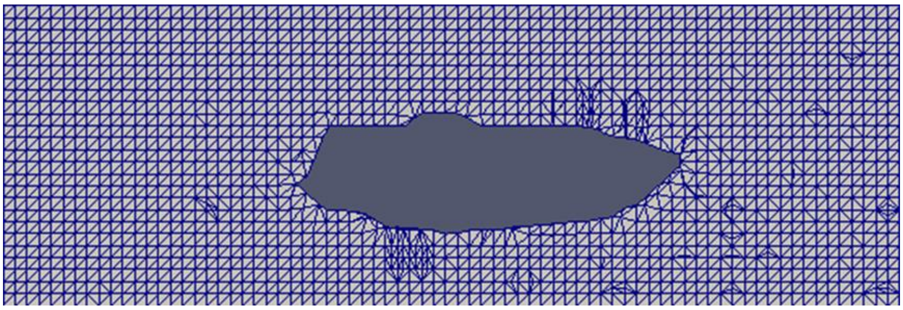


Figure 4.7 - Sideview of the overset mesh with the lifeboat's external surface

The end of the process resulted in a shape similar to the designed lifeboat but with some deviations due to the snapping process. This alteration is expected for coarse meshes where the refinement of the snapping process is not enough to acquire all the information, as seen in Figure 4.8.

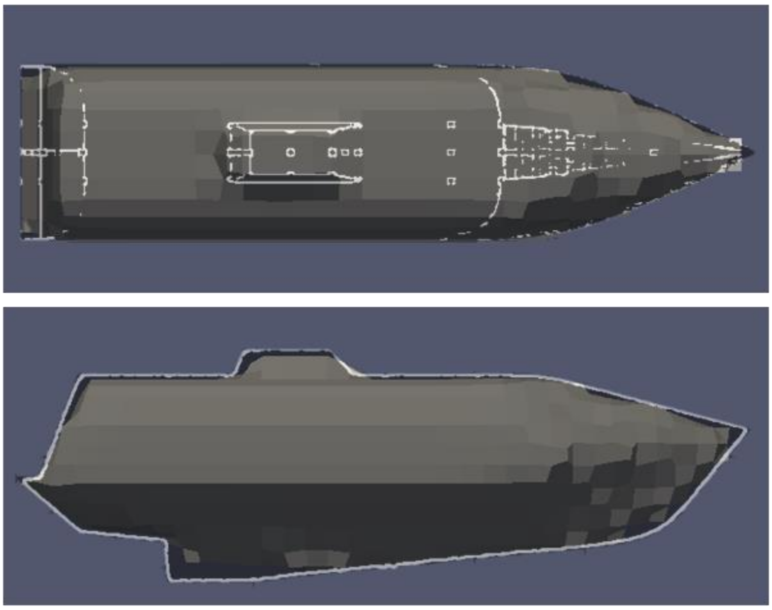


Figure 4.8 - Lifeboat patch in overset compared to sideview and up view from rhino 3D

This was the result of a choice to have faster and more stable simulations where the whole lifeboat journey can be described. The computational time of the whole simulation is extremely sensitive to how refined the mesh and time step are defined, as it was possible to notice in Table 3.3 - Cells details and CPU time of the simulations. Since the geometry is based but not the same as the Schat Harding 1000 used in the experiments, the time of simulation was prioritized over the mesh refinement even to see how the results and the lifeboat in general behave.

By being modelled over the Schat Harding 1000, even if slightly different from each other, the model can still be checked on the pressure sensors used in the experiments of Kauczynski, et al., 2009 by at least seeing how it behaves. The pressure sensors are positioned as shown in Figure 4.2. The comparison with the experimental is shown in Figure 4.9 and Figure 4.10.

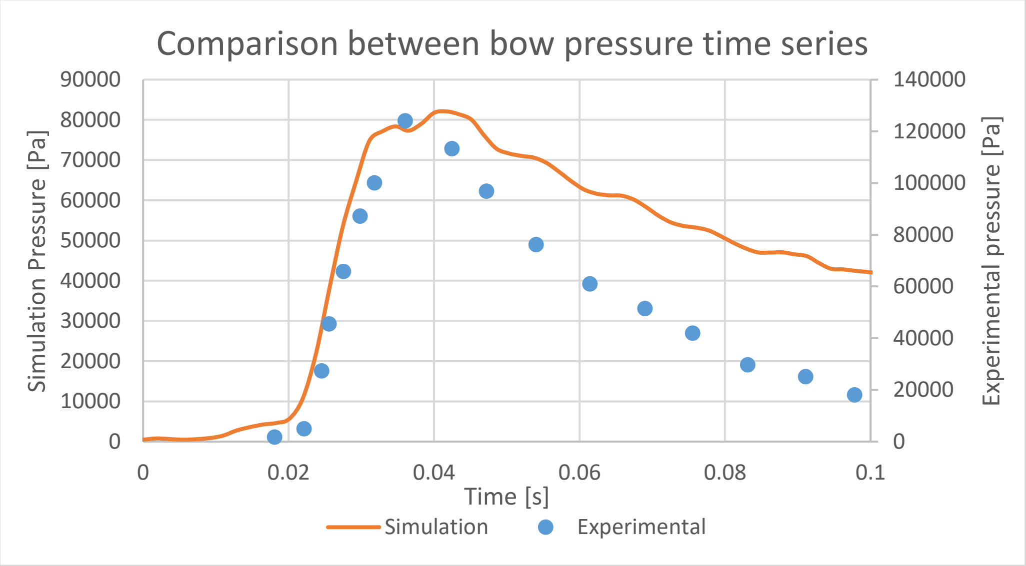


Figure 4.9 - Bow pressure compared with experimental results

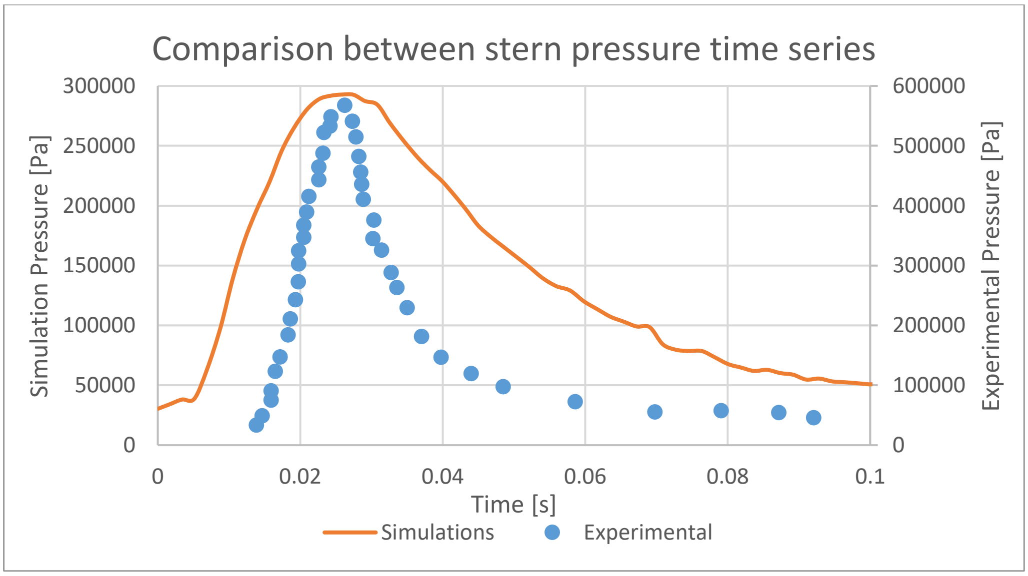


Figure 4.10 - Stern pressure compared with experimental results

What is first noted when comparing the results is that the numerical values do not match, which was expected due to the changes in geometry, mesh, and turbulence model. Qualitatively, the behaviour matches each other, with differences in how fast the peak grows and decays. This has to do with how both pressures are measured, while one uses a plate sensor that measures the average force over a surface with a larger area which makes the peak decays faster, while the computational cell is infinitely small (Huang, et al., 2020). This means that, physically, OpenFOAM can assess and give great contributions, and qualitatively, it would require more refinement and time to simulate each case.

4.2.2 Pressures

After the preliminary analysis, the study can be extended to do an overview of the pressure along the hull. In order to do this, the pressures on the intersection between the hull and three planes are considered as shown in Figure 4.11.

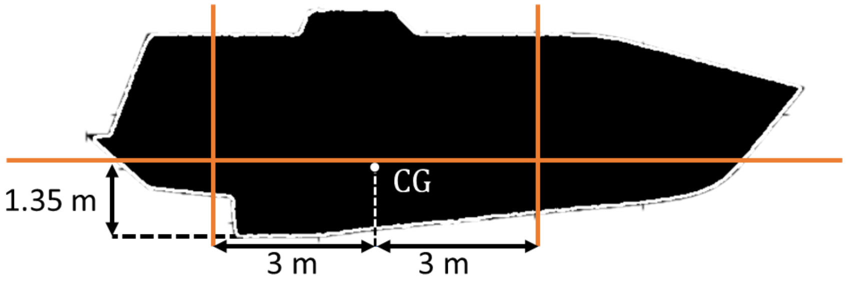


Figure 4.11 - Planes of pressure study in scale

The transversal planes in bow and stern are divided into 11 points over the impact surface (hull below) where the pressures are taken, while for the longitudinal plane, 21 points are considered. First, regarding the stern plane, the control points for measuring pressure during the simulated time are shown in Figure 4.12.

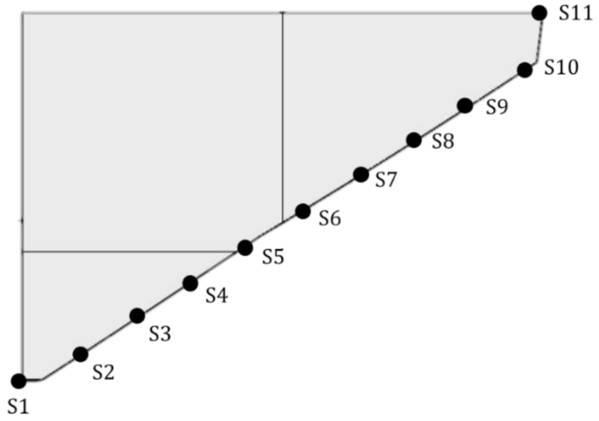


Figure 4.12 - Stern plane with pressure sensors

For comparing results and reducing the amount of data, just the cases 10°, 30°, 50°, and 70° are analysed. First, it is possible to compare the results of three heights from 10° in Figure 4.13 to Figure 4.15. Some simulation results are the same due to the space between control points being not enough to be in different cells with different properties and parameters, which is the case of S1, S2, and S3, for instance.

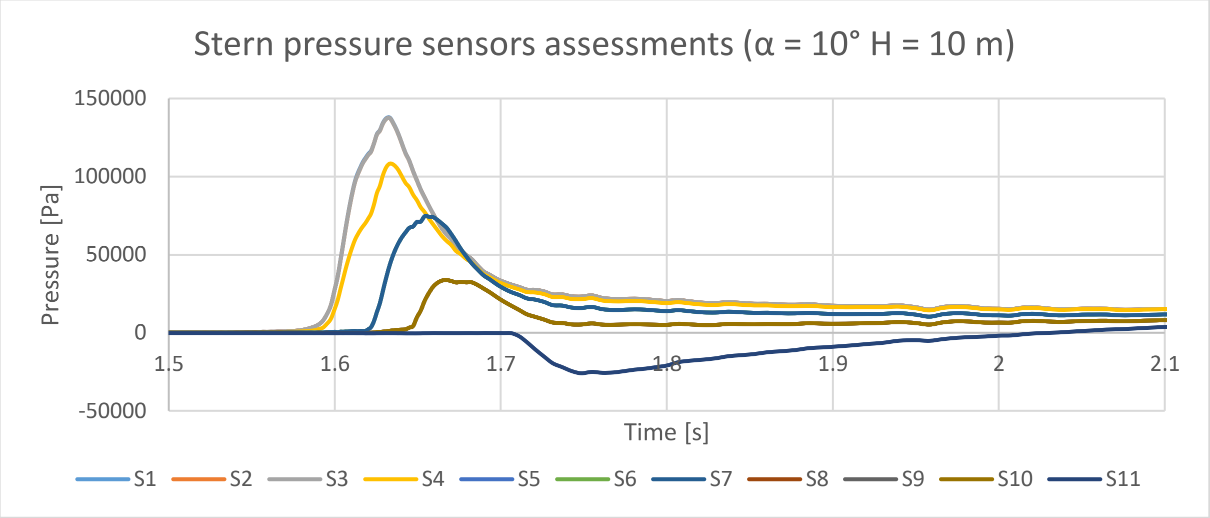


Figure 4.13 - Stern pressure time series for H = 10 m and α = 10°

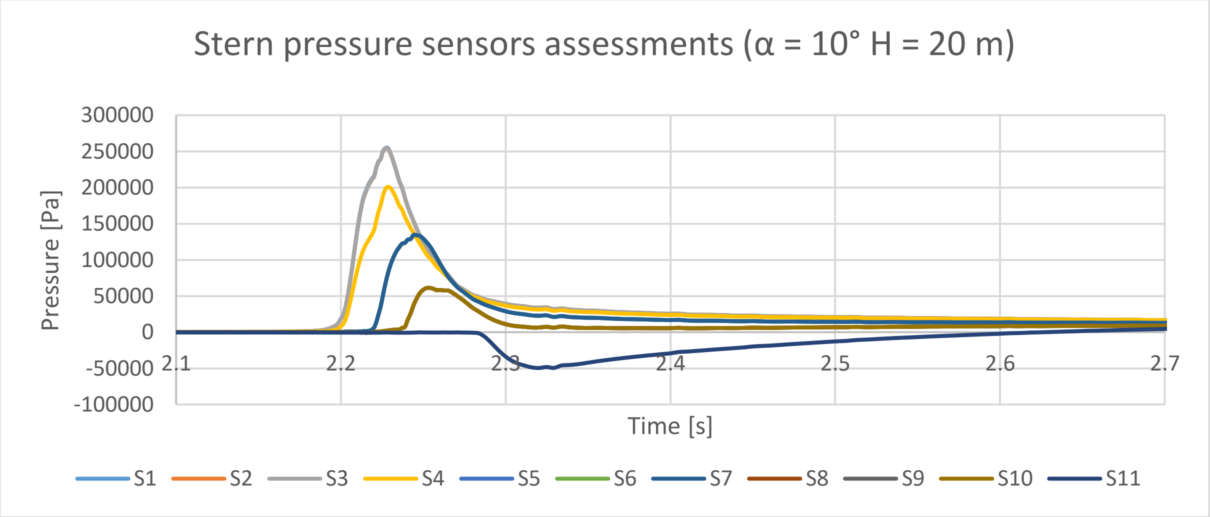


Figure 4.14 - Stern pressure time series for H = 20 m and α = 10°

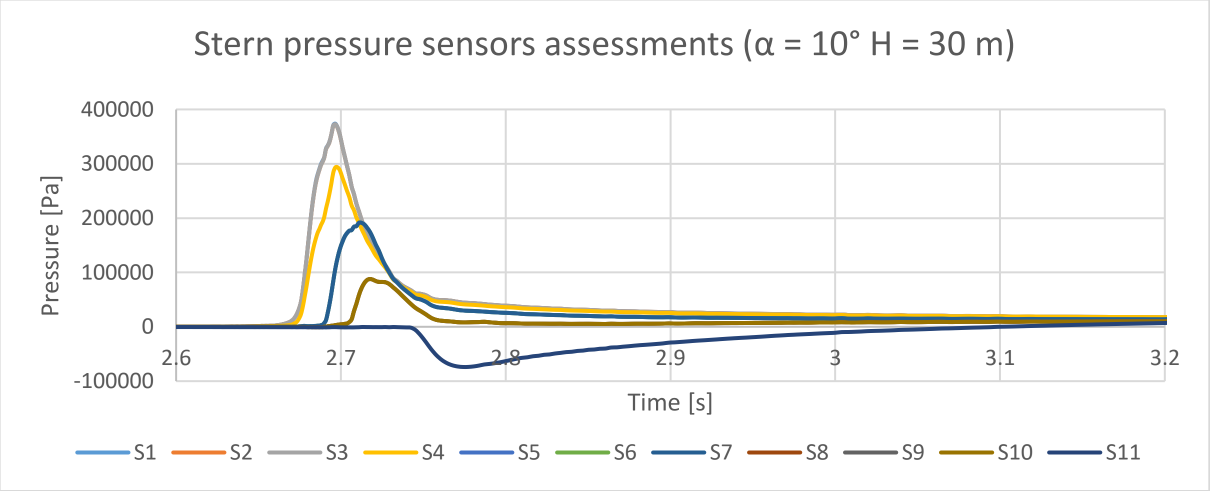


Figure 4.15 - Stern pressure time series for H = 20 m and α = 10°

The behaviour of the pressure curves is mostly the same for different heights, with the difference in the pressure peak, which is strongly influenced by the gained velocity from freefall. They differ not only by the peak value on top but also by the decay velocity, which becomes faster the higher it starts from. Also, the highest peak of them all ate the first three, which are the ones that came more from below on the keel. The behaviour is mostly the same for all simulations with different initial dropping height, so the analysis from here and beyond focus just on angle with a fixed height of 10 m.

When extending the analysis to the other angles, plots are made and shown in Figure 4.16 to Figure 4.18. In general, they all present the same behaviour with different peak values, mostly. The highest peak is the 3 lowest points on the keel, while the highest one at S11 has negative pressure due to the water jet detachment at this point, and in general, the higher the point, the lower the pressure will be, except in 50° where S5, S6, and S7 were lower than S4. The angle with the highest pressure was 50°, which already can indicate which scenario would have larger impact loads.

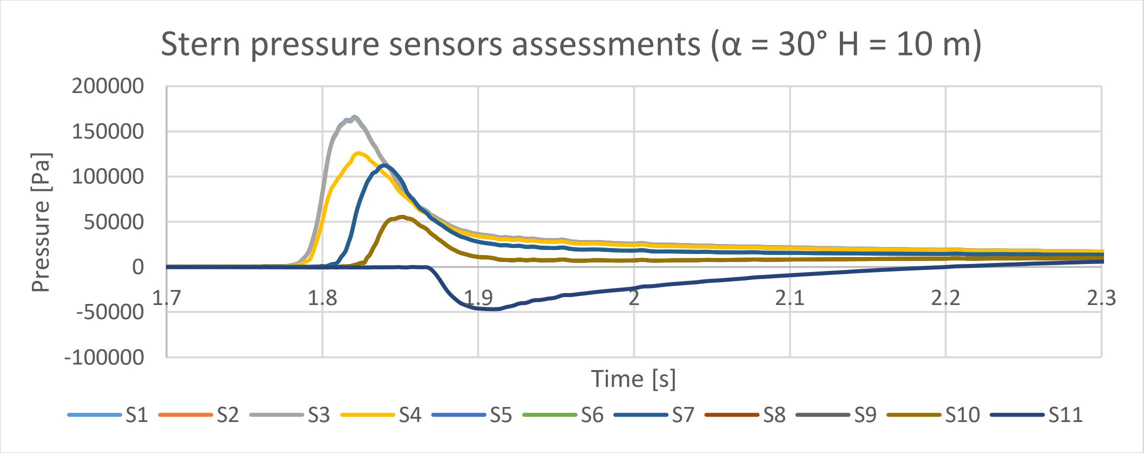


Figure 4.16 - Stern pressure time series for H = 10 m and α = 30°

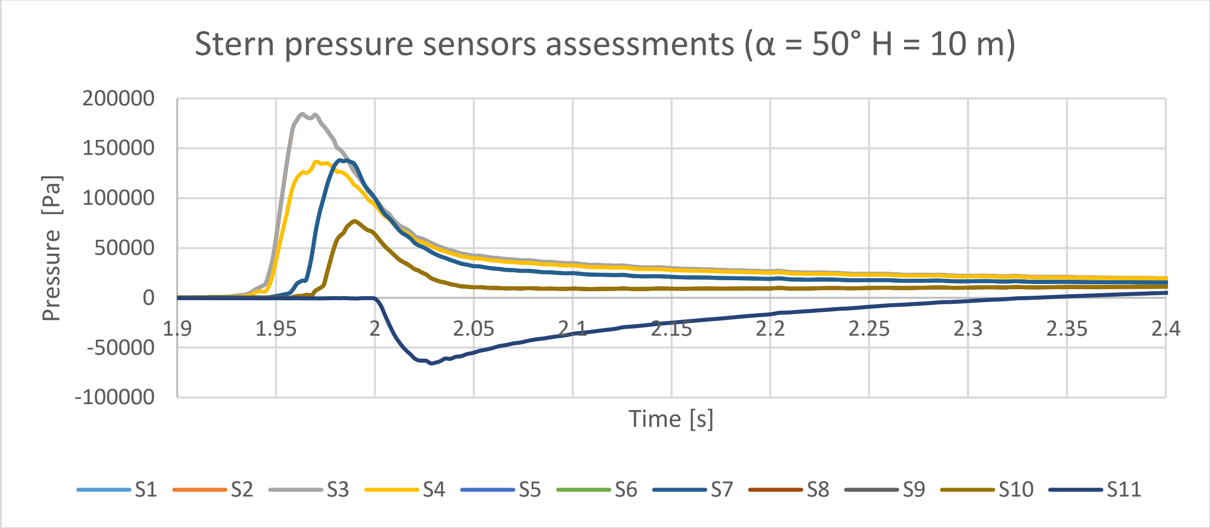


Figure 4.17- Stern pressure time series for H = 10 m and α = 50°

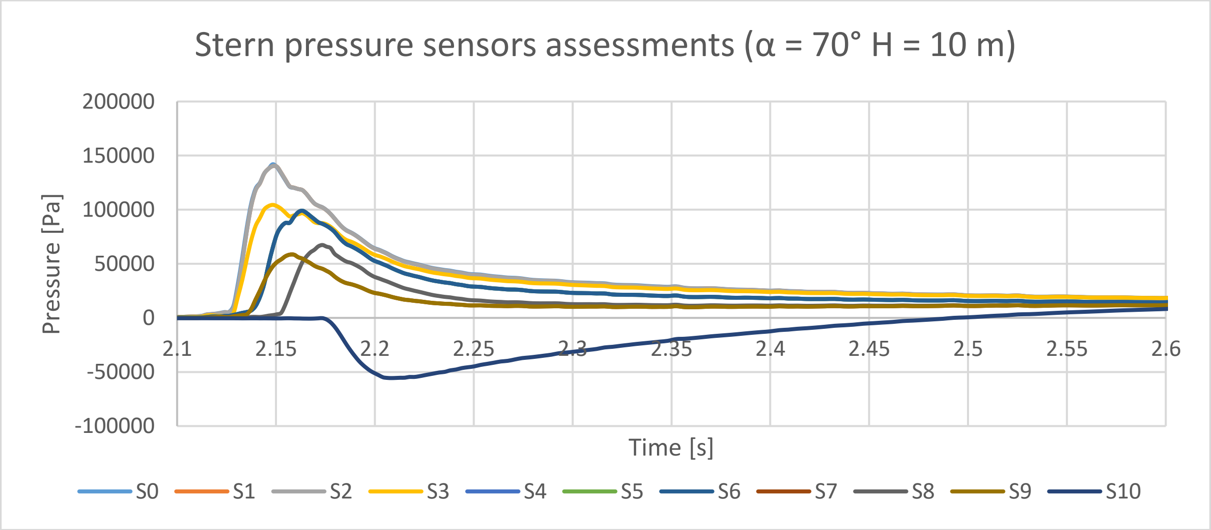


Figure 4.18- Stern pressure time series for H = 10 m and α = 70°

The same analysis can be performed for the selected bow plane, where the control points divided are shown in Figure 4.19. As for the results of these control points, they are shown in Figure 4.21 to Figure 4.23.

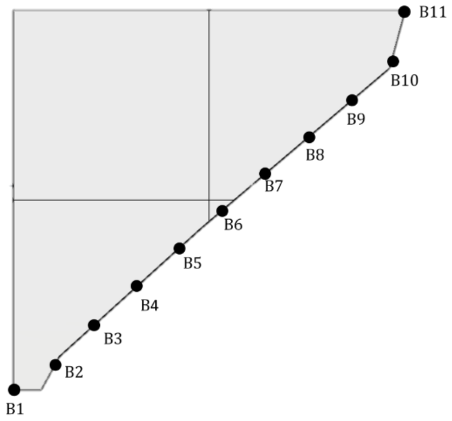


Figure 4.19 - Bow plane with pressure sensors

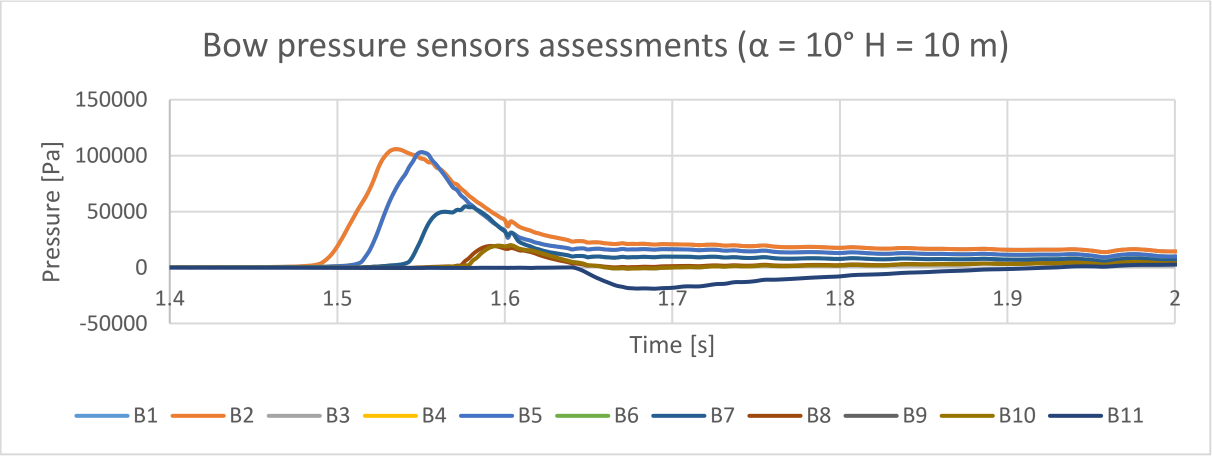


Figure 4.20 - Bow pressure time series for H = 10 m and α = 10°

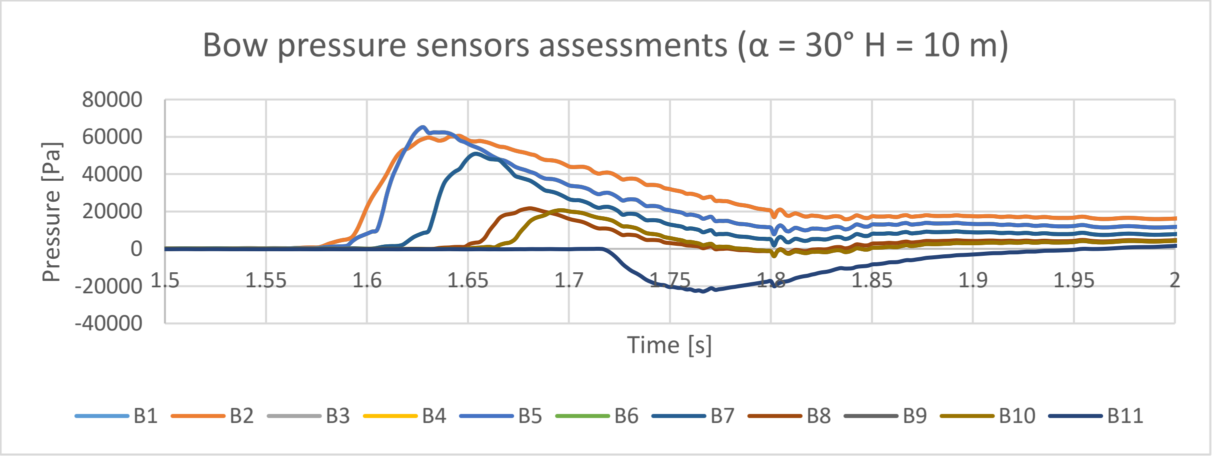


Figure 4.21- Bow pressure time series for H = 10 m and α = 30°

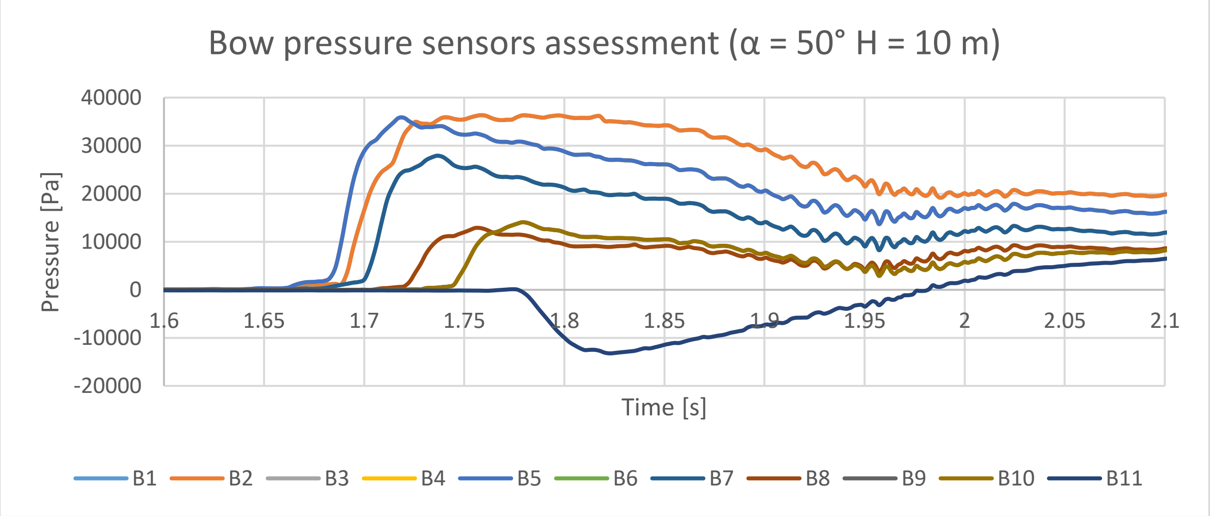


Figure 4.22- Bow pressure time series for H = 10 m and α = 50°

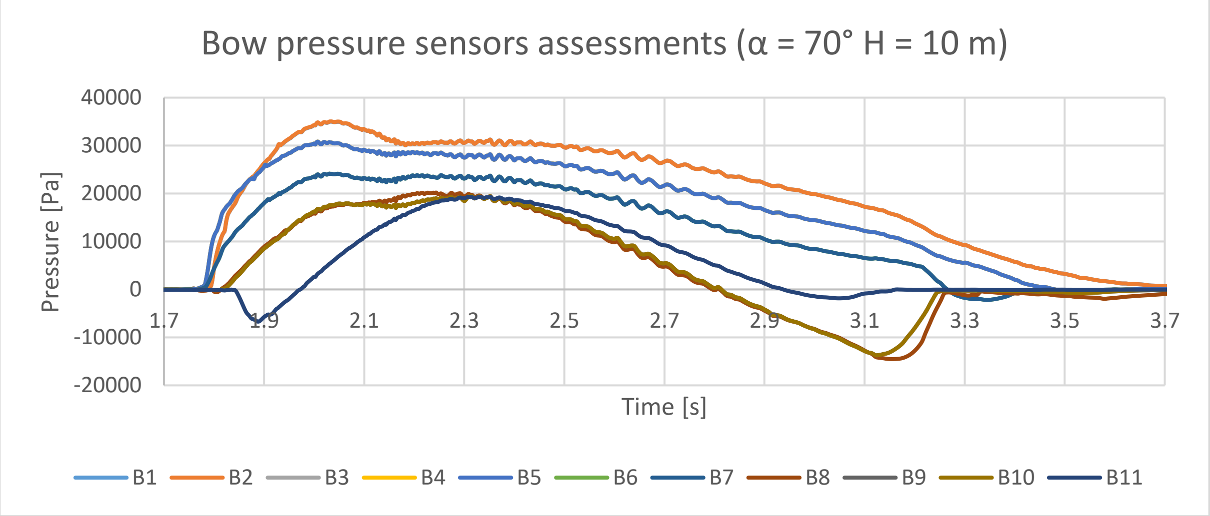


Figure 4.23 - Bow pressure time series for H = 10 m and α = 70°

What is noticeable in the cases is that they are considerably lower than the stern entry, indicating that the rotation during water entry can provoke another slamming itself on the back of the hull. They have different shapes even if they have peaks, especially in the case of 70° compared to the rest. This can be explained by how the boat enters the water as shown in Figure 4.45. As for the behaviours, similarly, the bottom control points are usually the highest peak, where the first two competing for the highest, while the less critical was the highest one on the boat due to the water detachment. It results in larger peak bells and being a demanded region, especially in what concerns forward velocity.

Another analysis is assessing the longitudinal planes as defined in Figure 4.24 and with results as shown in Figure 4.25 to Figure 4.28. The control points for this plane were too much information for the plot, which is why the plots only show the results from sensors with odd numbers in them. Usually, there are two highlighted peaks, one for L1, which means the forward border, and another one for the other in the middle before ending the simulation, except for the 10° launch, which also has to do with the way the boat dives. What is also notable is that the one with a higher peak is also 50°, but for a small margin when compared with the 30° case.

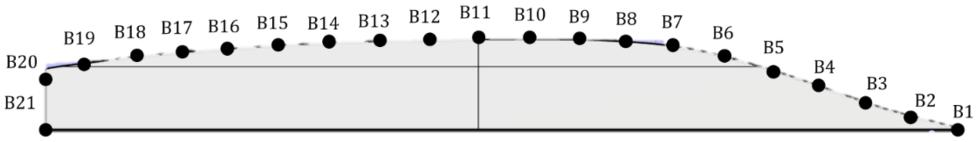


Figure 4.24 - Longitude plane with pressure sensors

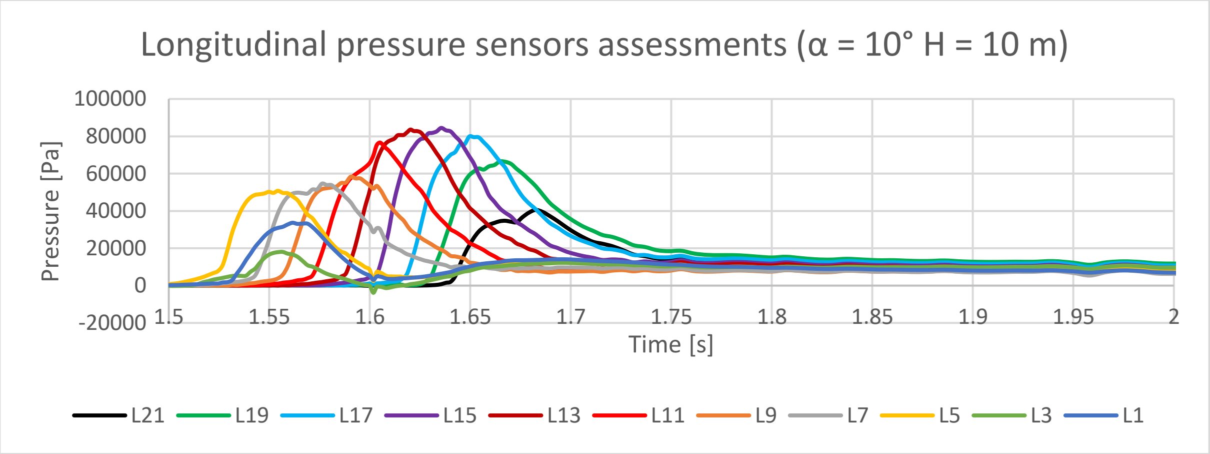


Figure 4.25 - Longitudinal pressure time series for H = 10 m and α = 10°

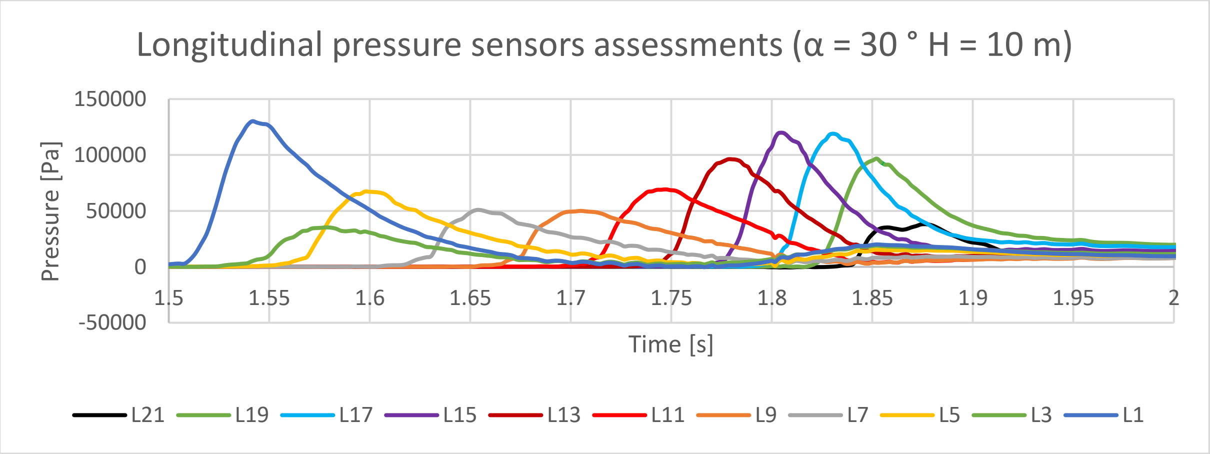


Figure 4.26- Longitudinal pressure time series for H = 10 m and α = 30°

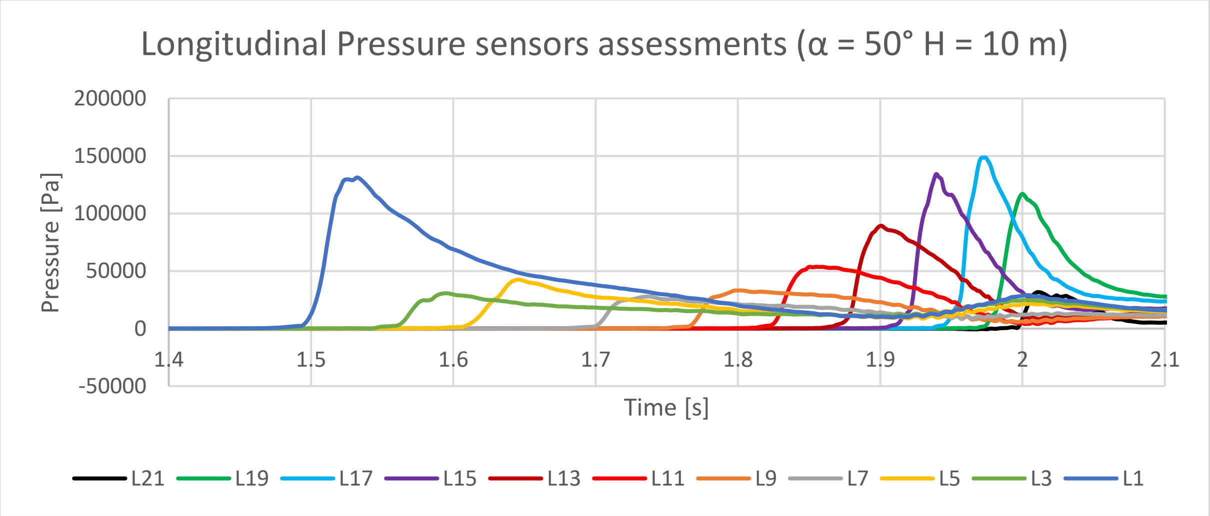


Figure 4.27- Longitudinal pressure time series for H = 10 m and α = 50°

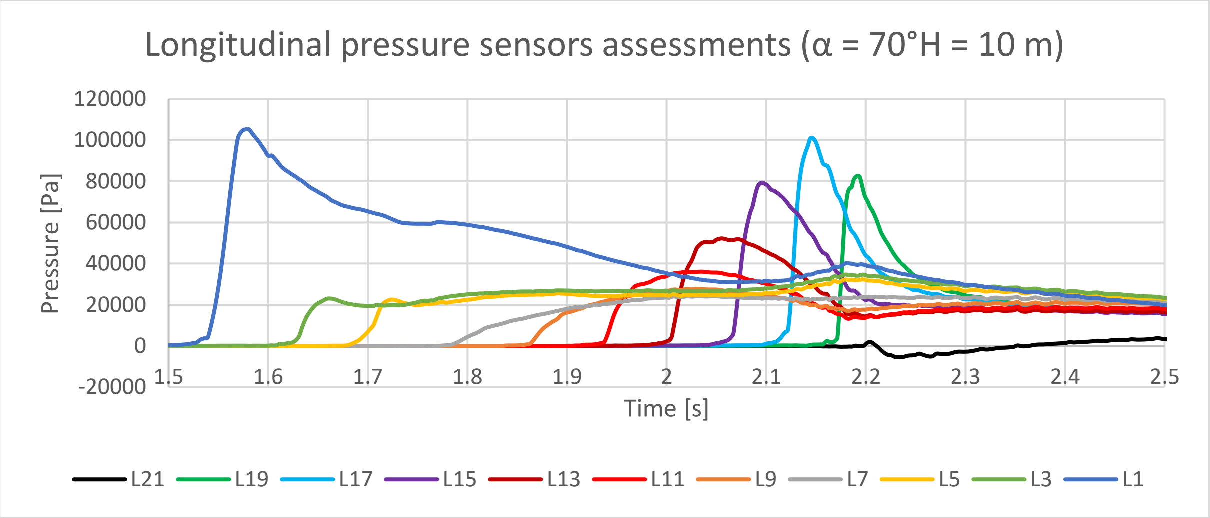


Figure 4.28- Longitudinal pressure time series for H = 10 m and α = 70°

When using probes, it is possible to see how the pressure behaves over time at a specific point, but it is not possible to analyse what happens to the whole surface. This can be assessed by using paraView at the time step specified. Some of these results are shown in Figure 4.29 to Figure 4.32.

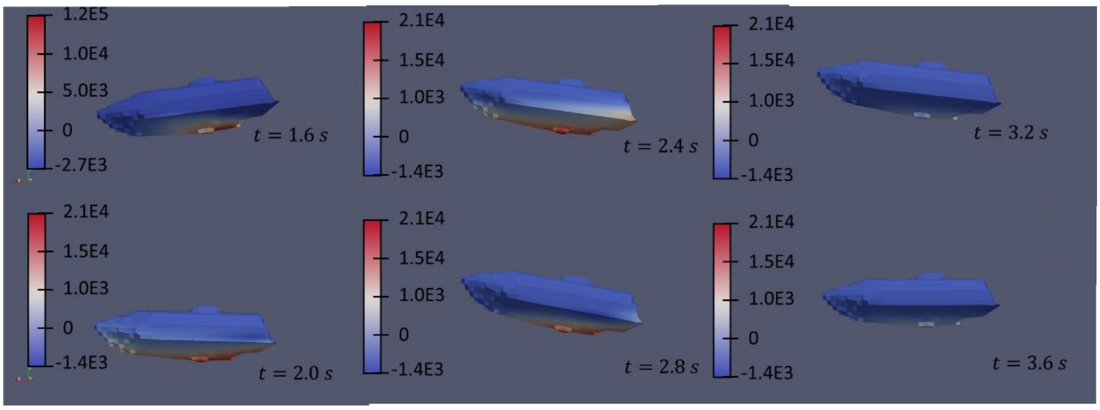


Figure 4.29 - Pressure map, in Pa, for 10°

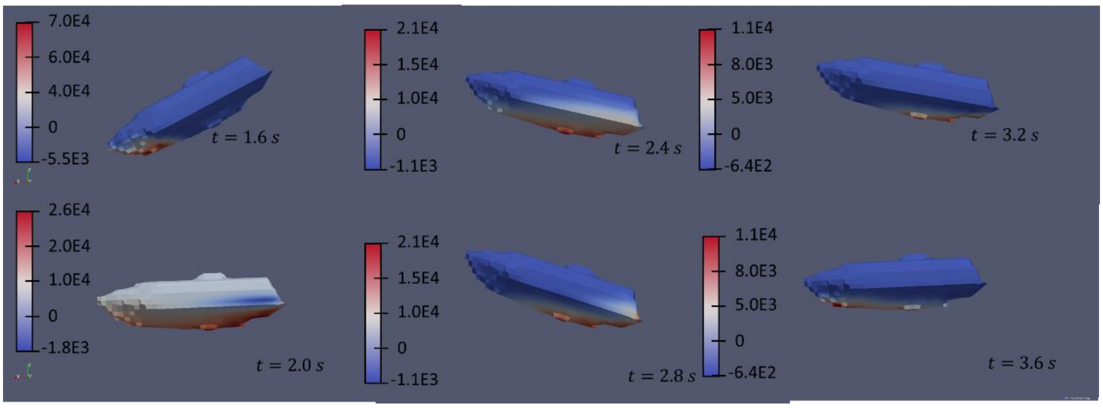


Figure 4.30 - Pressure map, in Pa, for 30°

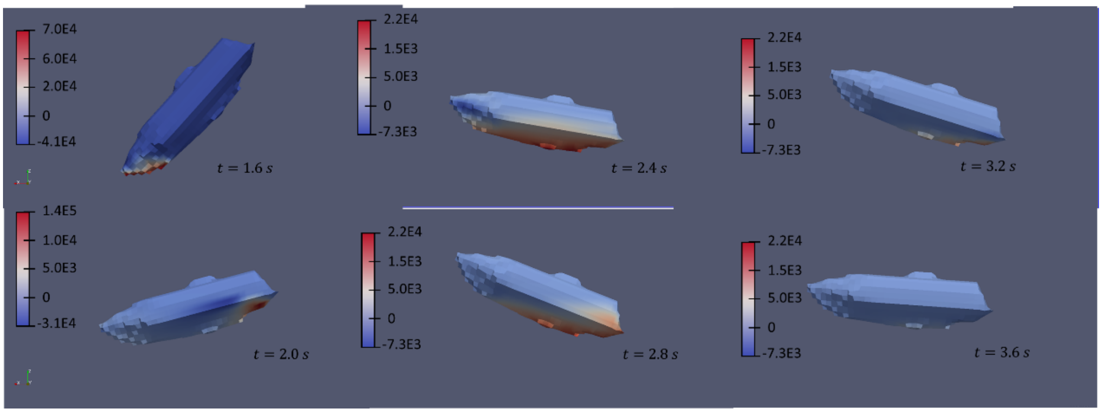


Figure 4.31 - Pressure map, in Pa, for 50°

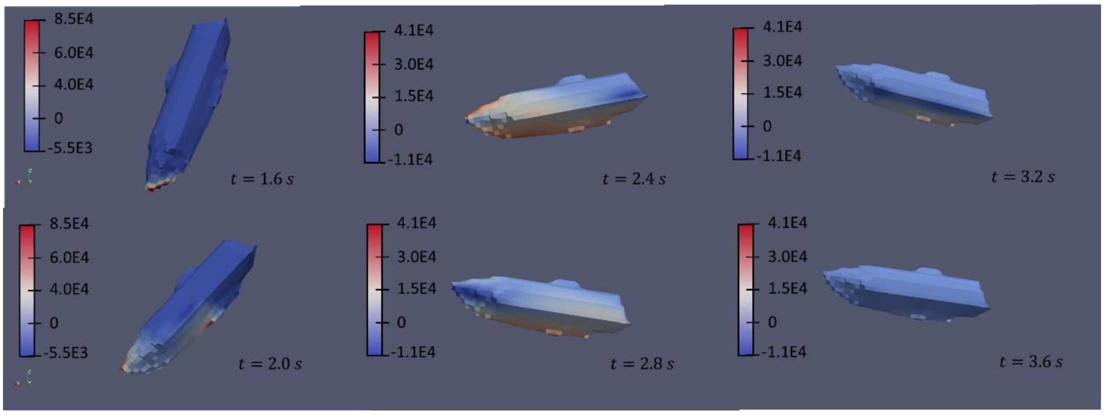


Figure 4.32 - Pressure map, in Pa, for 70°

For the output time step of 0.2 s, the instant where the lifeboat first touches the water is 1.6 s, which is why is always the first instant shown in the Figures above. It is limited to the output interval previously defined and also on the memory of the computer used, but it has the advantage of mapping the pressure over the whole surface and even indicating where the pressure sensors should be input in the next simulation.

What is shown in all scenarios is that the first two-time instants considered are the most critical for the lifeboat, where the highest peaks appear due to the water entry. After this, the pressure decays until further stabilization at a lower value. What is noticeable in all simulations is that the keel is the boat element where the pressure peaks concentrate, which is expected since there is an impact region with the water at high speed.

Two regions are of interest where the peak loads concentrate: the bow deck and the stern. The bow is the first to enter and receive the highest loads due to the impact on calm waters. After this, the turnback that the ship suffers due to the restorative moment given by the water buoyancy makes the stern also hit the water, creating another slamming impact, and one with ever higher peaks than the bow. This just not happens with the 70° scenario because in this case, the lifeboat dives before getting this turnback rotation, and the peak for diving is lower than for hitting.

4.2.3 Kinematics

In this part, the general kinematics of each simulation is discussed. What is important to emphasize here is that more parameters are studied and analysed here, since the simulation has planar motion instead of unidirectional as it was with the wedges in chapter 3. This kind of assessment also has importance besides the impact loads due to the importance of how the lifeboat will respond when it is required in an emergency. In general, it should get as far away as possible from the hazardous event when launched, and also its journey should not harm or injure the people on board. Since much data is expected from 21 simulations, one plot per height and parameter is done for different angles.

Firstly, what can be analysed is the acceleration in each direction. The horizontal acceleration in this kind of study is important because it is what gives the initial velocity that makes the vessel move forward from the dangerous zone. The results are shown in Figure 4.33 to Figure 4.35.

At first glance at the three plots, what is notable is that the higher the dropping height is, the higher the acceleration gets. The peaks can give an idea of which cases the boats would not go far, being the ones with the smallest peak due to the fact that they would not acquire much initial horizontal velocity. However, the best ones cannot be told yet because right after the big positive peak, there is a negative acceleration due to resistance given by the acquired velocity, which is why is necessary to analyse the integration into velocity.

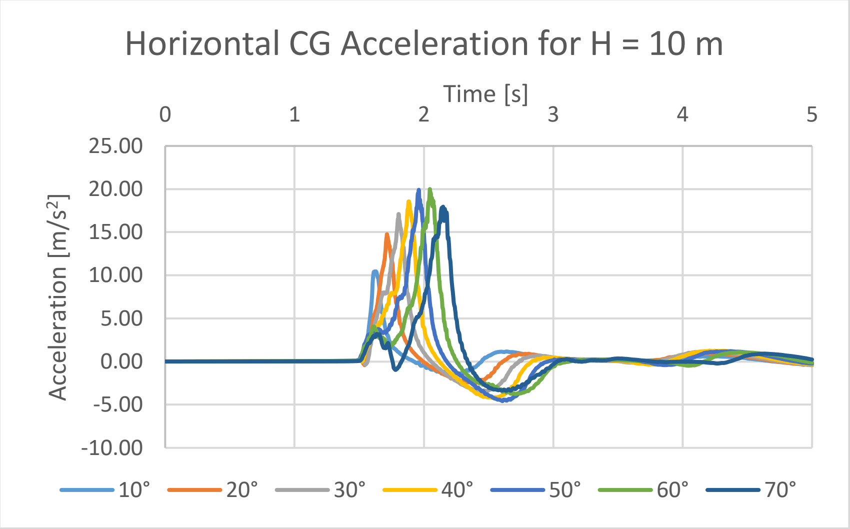


Figure 4.33 - Lifeboat's horizontal acceleration for H = 10 m

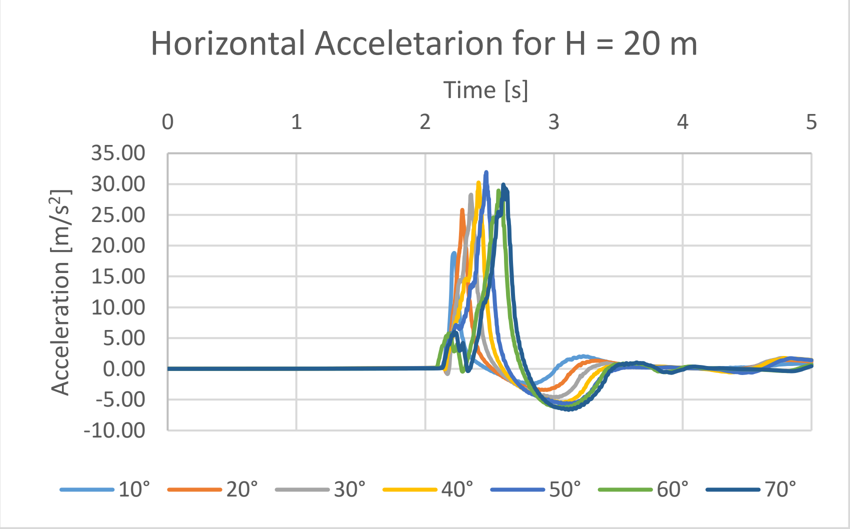


Figure 4.34 - Lifeboat's horizontal acceleration for H = 20 m

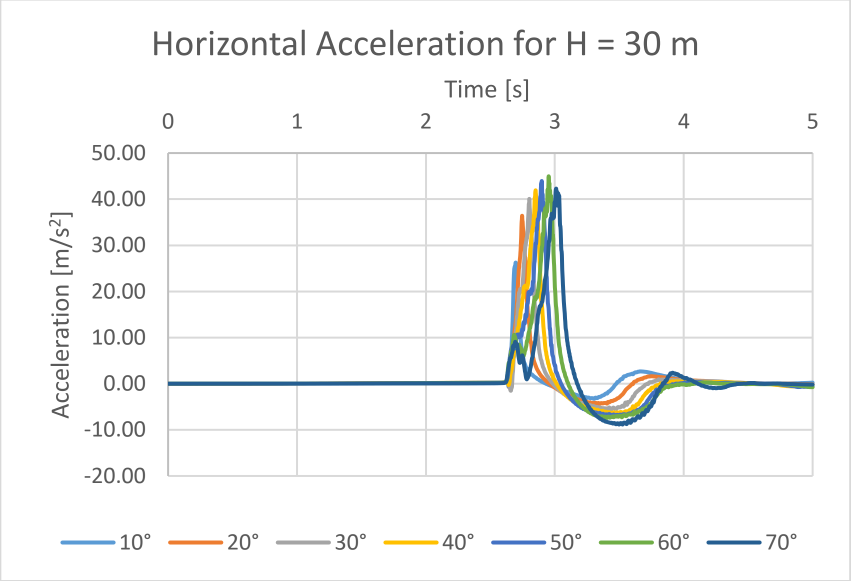


Figure 4.35 - Lifeboat's horizontal acceleration for H = 30 m

Similarly, this analysis can be extended to the vertical acceleration, shown in Figure 4.36 to Figure 4.38. The behaviour was similar to what is seen for the horizontal acceleration, there is a positive peak before a negative one, which is expected of floating bodies in heave oscillation after hitting the water. Also, the peaks seem to be larger, which has most to do with how long it takes to override and invert the vertical velocity and how was the trajectory on the water. The free-fall stage is marked by the constant acceleration of g for vertical acceleration and close to 0 for horizontal. Also, regarding the dropping heights, what is remarkable is that the greater the height, the greater the acceleration peak is, which is expected because the highest the boat fall, the highest the velocity before hitting the water which should be nullified.

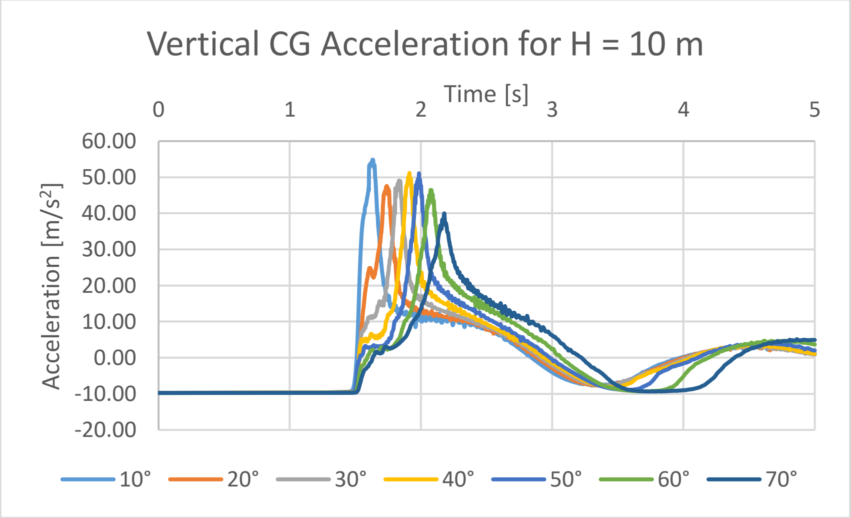


Figure 4.36 – Lifeboat’s vertical acceleration for H = 10 m

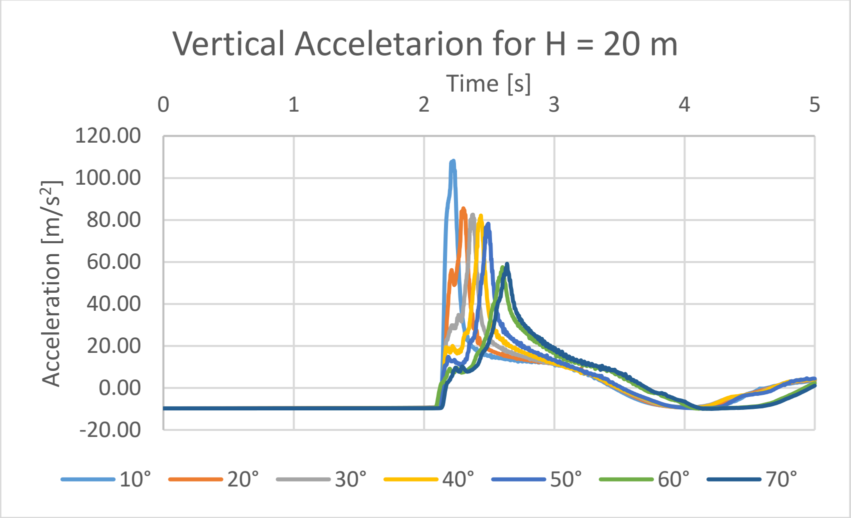


Figure 4.37 - Lifeboat's vertical acceleration for H = 20 m

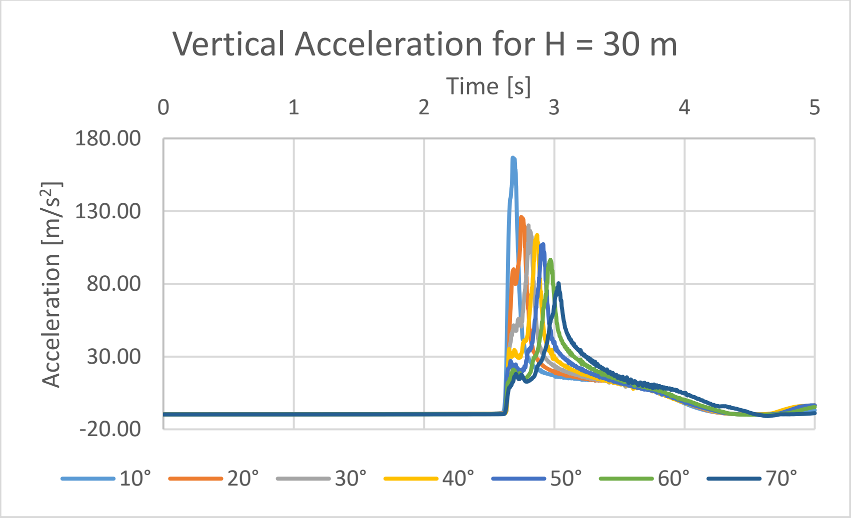


Figure 4.38 - Lifeboat's vertical acceleration for H = 30 m

Acceleration is an important factor to consider when dealing with watercrafts, and lifeboats especially, because exaggerated acceleration can increase the probability of injuries on board since these deals directly with the inertia of the body that tends to maintain its momentum while the external forces brutally change it. Depending on the acceleration intensity, it can be even fatal to humans (Pearce, 2020). Table 4.3 shows what are the peaks of total resultant acceleration taken from this simulation, and from this, the cases with the highest chances of injuries onboard. Of course, the higher the dropping height, the worse will be the acceleration, and also the smaller the angle is, the worse it gets.

Table 4.3 – Maximum resultant CG acceleration in g

	10°	20°	30°	40°	50°	60°	70°
10 m	5.7	5.0	5.2	5.5	5.5	5.1	4.4
20 m	11.2	9.0	8.8	8.8	8.5	6.5	6.7
30 m	17.2	13.4	12.9	12.1	12.1	10.7	9.2

One way to assess if the case's accelerations are adequate is to use the IMO Combined Acceleration Response (CAR), shown in (Netherlands Regulatory Framework - Maritime, 1993), where the Square Root Sum of the Squares (SRSS) acceleration should not surpass the ellipsoid with the axis of 15 g's in the +/- x-axis and 7 g's in another axis.

$$CAR_i = \sqrt{\left(\frac{a_{x_i}}{15g}\right)^2 + \left(\frac{a_{y_i}}{7g}\right)^2 + \left(\frac{a_{z_i}}{7g}\right)^2} \leq 1 \tag{4.1}$$

To assess which cases simulated could be dangerous for people, the seats are assumed 90° and front-facing. Additional centripetal and tangential accelerations due to rotation are neglected, which is reasonable for small crafts, and in this case, the chairs' angles and positions are unknown. In this case, the conversion of acceleration from the global cartesian coordinates to the seat relative's coordinate can be done by rotation matrix for a given pitch angle θ using Equation 4.2.

$$\begin{bmatrix} a_{x_i} \\ a_{z_i} \end{bmatrix} = \begin{bmatrix} cos\,\theta & -sin\,\theta \\ sin\,\theta & cos\,\theta \end{bmatrix} \begin{bmatrix} A_{x_i} \\ A_{z_i} \end{bmatrix} \tag{4.2}$$

where a_{xi} and a_{zi} are the horizontal and vertical accelerations relative to the chair coordinates and A_{xi} and A_{zi} are the accelerations with respect to the global cartesian coordinates. The pitch angle could be retrieved from OpenFOAM's log file, and it is presented in Figure 4.56 to Figure 4.58. The same behaviour from the vertical offsets can be seen in the time series of pitch, except for the constant angle when the boat is free-falling, due to the absence of relevant torque. What can be seen is that the larger the angle, the larger the spin for the lifeboat to reach equilibrium. Since the minimum peaks are almost near each other, this also means that the angular velocity and acceleration are higher for greater angles.

With this, Table 4.4 contains the results for the maximum CAR obtained with these premises. What is seen is that for greater heights and lower angles, the acceleration can be problematic and risky for the fleeing crew and passengers. This does not mean that people will get injured in these other cases, but the probability of human injuries is greater (IMO, 1991).

Table 4.4 - Maximum CAR for each simulation

	10°	20°	30°	40°	50°	60°	70°
10 m	0.81	0.72	0.74	0.78	0.79	0.73	0.63
20 m	1.60	1.29	1.25	1.25	1.21	0.92	0.96
30 m	2.45	1.90	1.84	1.73	1.73	1.53	1.31

Another kinematic parameter that can be analysed is velocity. This measure has less to do with how smooth or hard will the lifeboat enter the water, but more related to how the kinetic energy gained with the free-fall is converted. Ideally, the falling velocity should be turned integrally into forward velocity, but this is not what happens, since there is a loss of energy to waves, viscosity and even for heave oscillating. To see how this conversion occurred for each angle and height, the horizontal velocity plots are shown from Figure 4.39 to Figure 4.41. In these plots, a peak followed by a positive forward velocity with some variations is observed. As it was analysed before, the lifeboats with smaller horizontal acceleration peaks ended up with smaller peak velocities and stabilized at an inferior velocity, not being appropriate to flee from a hazardous event, while the ones with greater peaks dispute the highest velocity depending on the angle and height. In this case, 60° has shown to be more efficient for lower heights, while 70°, is for higher.

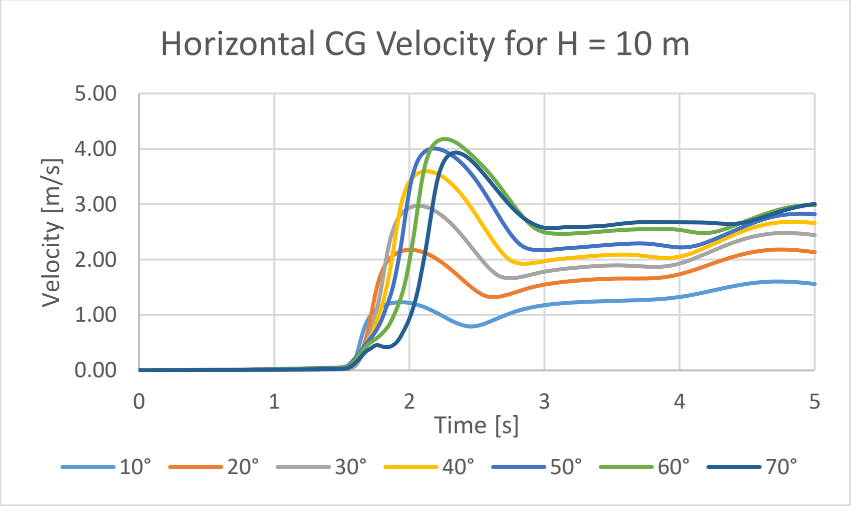


Figure 4.39 - Lifeboat's horizontal velocity for H = 10 m

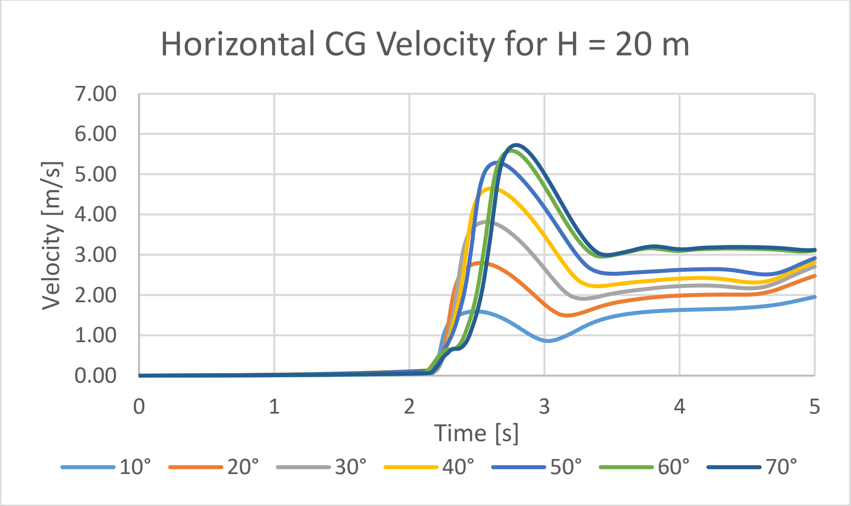


Figure 4.40 - Lifeboat's horizontal velocity for H = 20 m

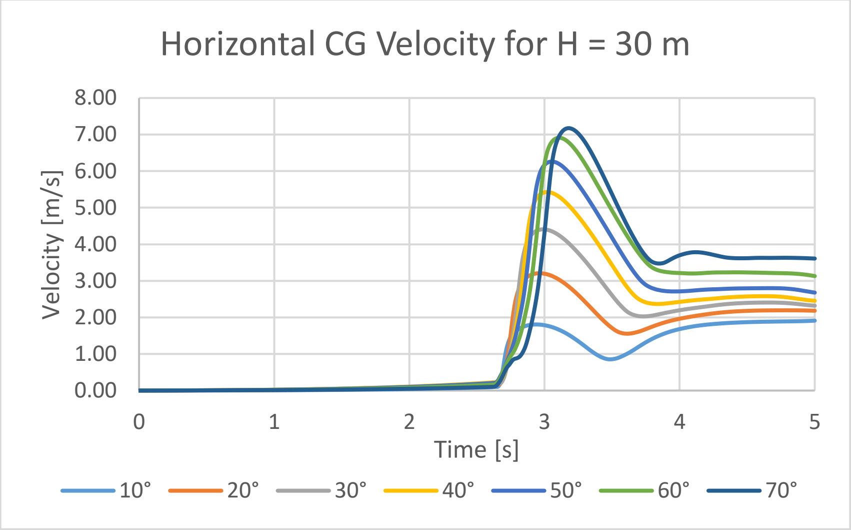


Figure 4.41 - Lifeboat's horizontal velocity for H = 30 m

As for the vertical velocity, it is shown in Figure 4.42 to Figure 4.44. The initial tendency is linear with a constant rate of g, as seen before with the acceleration, and after the free-fall stage, it oscillates around zero, which the tendency would be to continue until this dissipates into waves, mostly. What is also remarkable is that even with different initial velocities, the first positive peak right after entering the water was close to 5 m/s or below it.

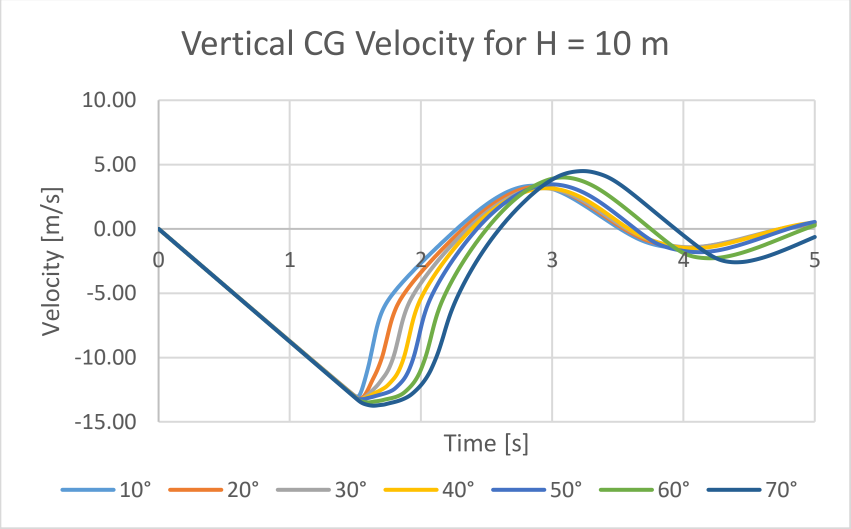


Figure 4.42 - Lifeboat's vertical velocity for H = 10 m

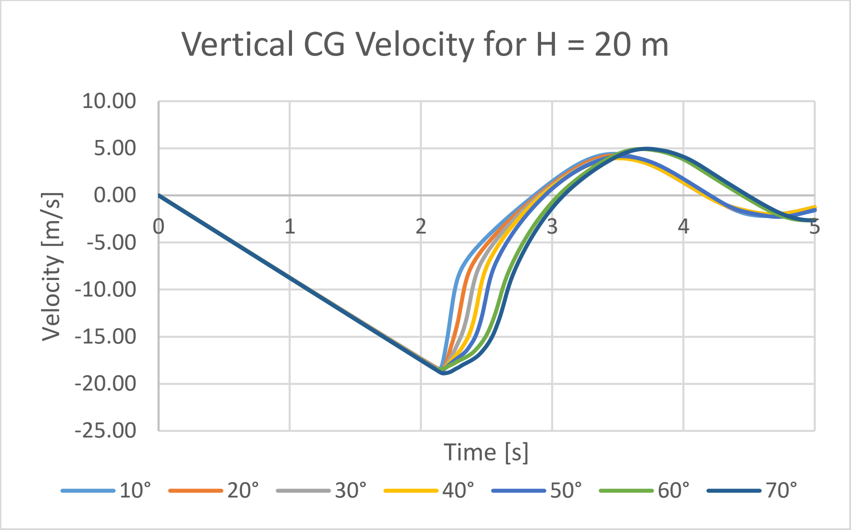


Figure 4.43 - Lifeboat's vertical velocity for H = 20 m

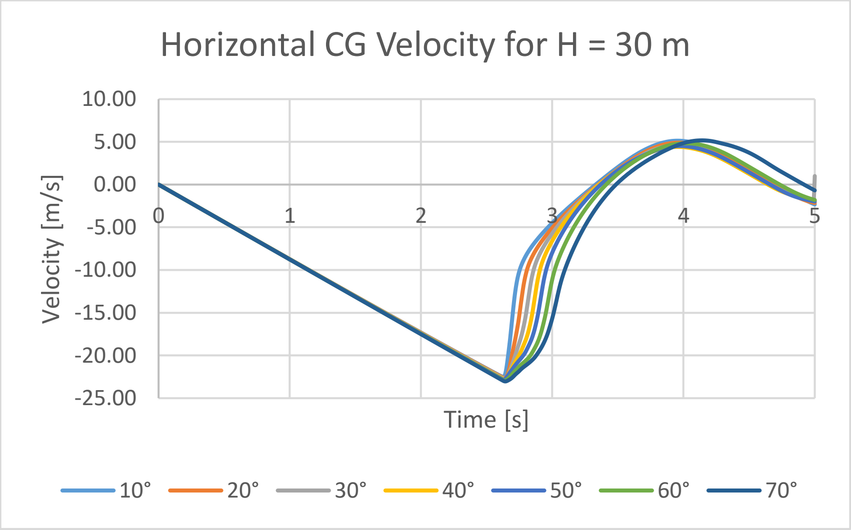


Figure 4.44 - Lifeboat's vertical velocity for H = 30 m

One aspect to notice not only in the velocities' plots, but also slightly in accelerations is that 60° and 70° had a distinct behaviour compared to the rest, not as an outlier from the rest, but because their water entry occurs differently from the others. The reason is due to the lowest point that enters first the fluid, where for angles below 50°, approximately, the bow part of the keel is the first to touch, while for the ones above this angle, it is the bow deck, which is illustrated in Figure 4.45. The slope's change of the keel curvature changes how the boat is pushed and even suffers from a brake right when the slope changes. This explains why the horizontal acceleration has two peaks and why their velocities peaks are a little delayed.

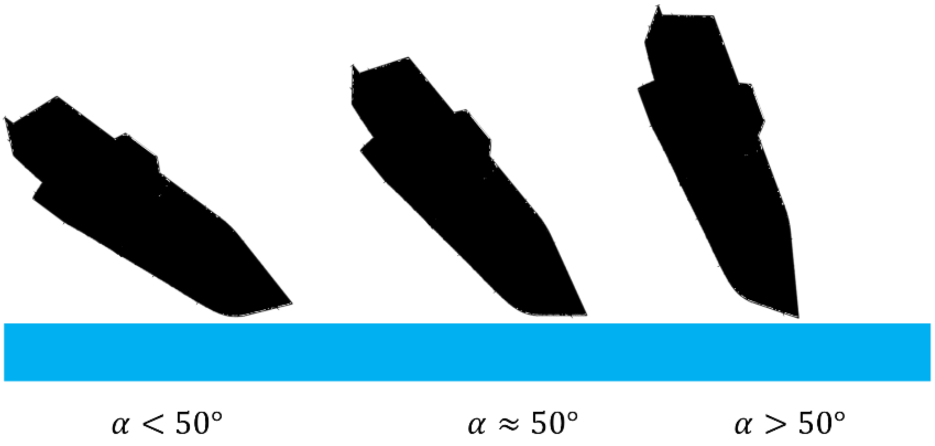


Figure 4.45 - How the lifeboat enters the water depending on the angle

After integrating the acceleration, it is possible to integrate the velocity on time and obtain the offset from the initial position. The horizontal displacement is shown in Figure 4.52 to Figure 4.54. This shows effectively which scenario, in a 5-second interval, would go further when dropped from a certain height. In this interval, the best ones that could go further were the 60°, for smaller heights, and 70°, for bigger. If the interval was smaller, the other boats would be more suitable due to the delay in the velocity peak caused by the angle of water entry, but even so, the higher peak and stabilization on a higher velocity make them displace further.

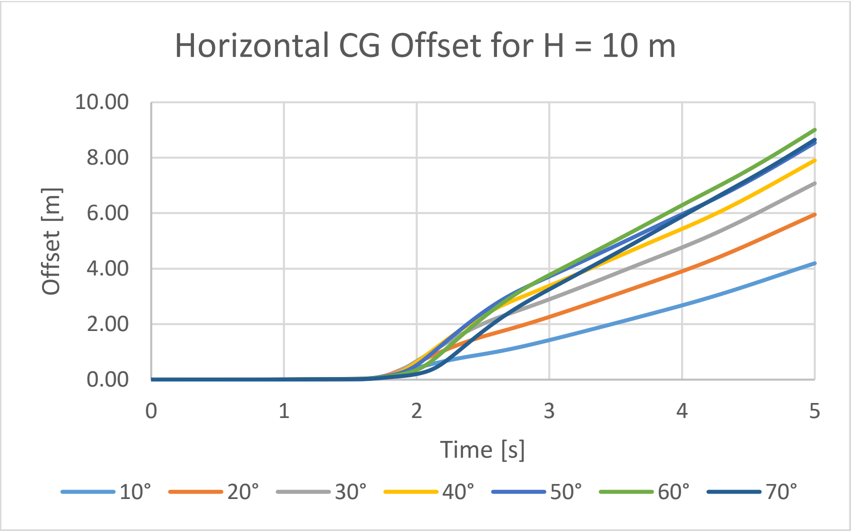


Figure 4.46 - Lifeboat's horizontal CG position for H = 10 m

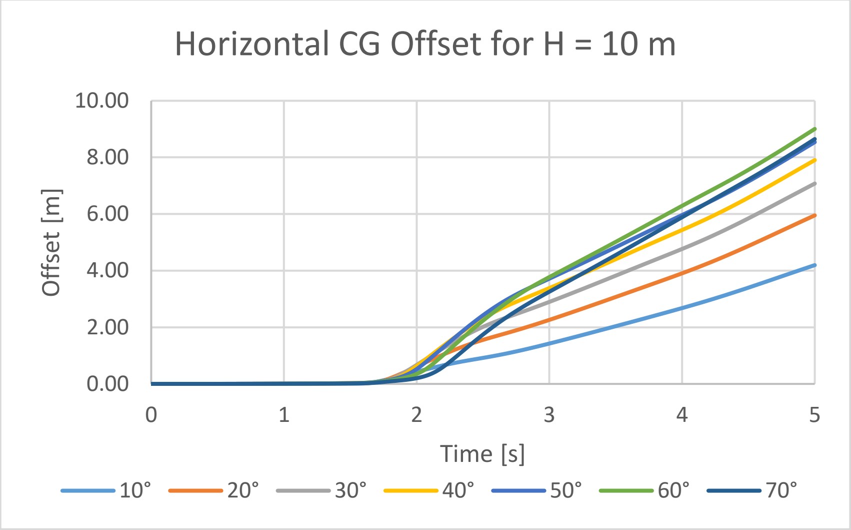


Figure 4.47 - Lifeboat's horizontal CG position for H = 20 m

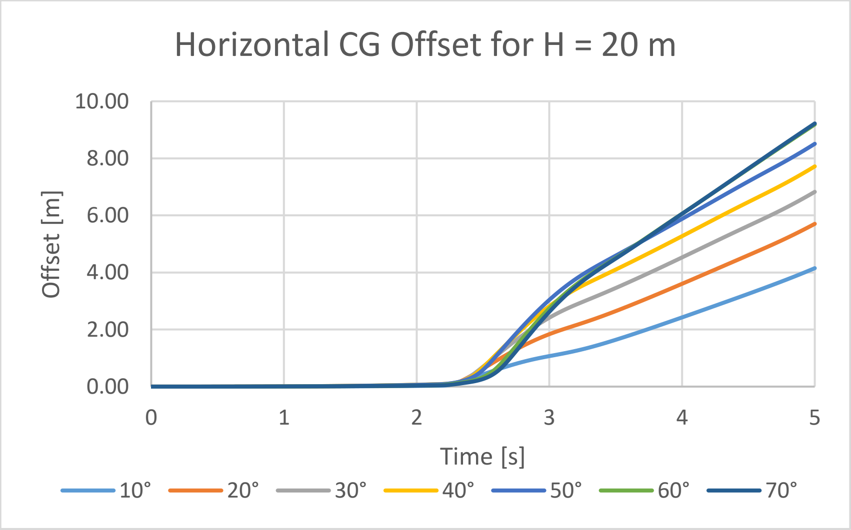


Figure 4.48 - Lifeboat's horizontal CG position for H = 10 m

As for the vertical offset, Figure 4.49 to Figure 4.51 presented what was expected from the vertical velocity part. Initially, during the free-fall stage, the time series is parabolic due to the constant gravity and when entering the water, the boat starts to freely oscillate with reducing amplitude due to loss of energy.

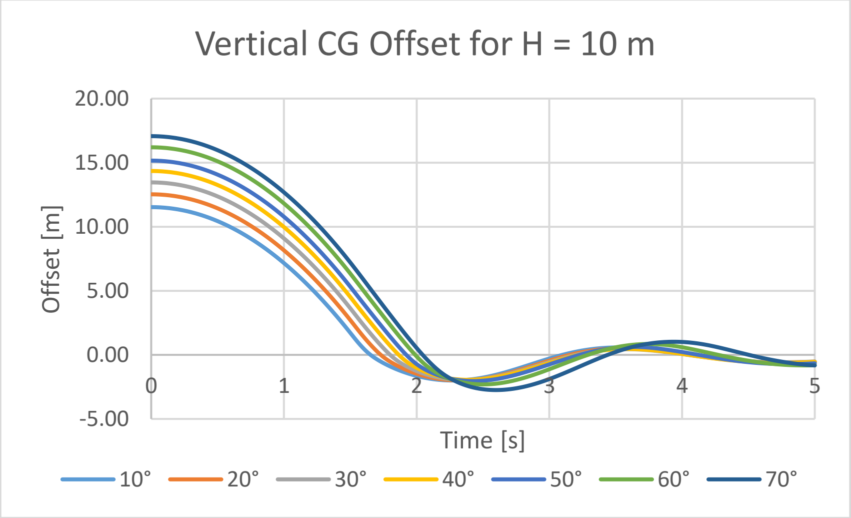


Figure 4.49 - Lifeboat's vertical CG position for H = 10 m

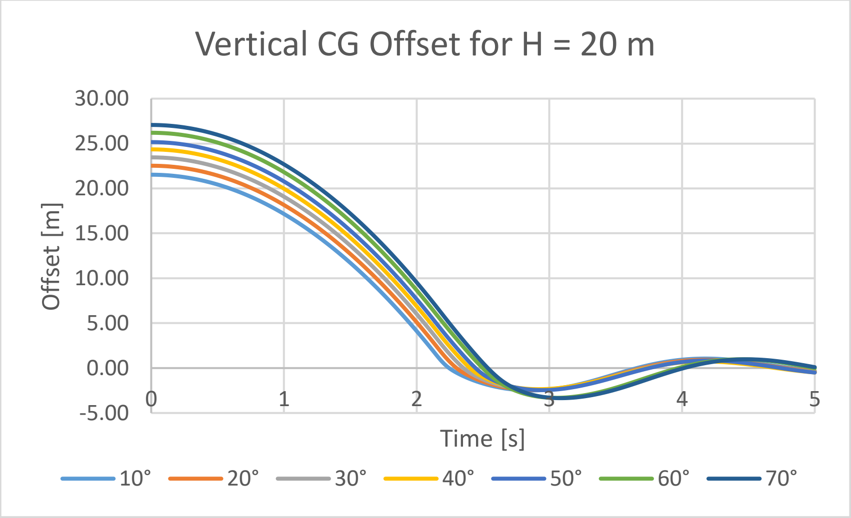


Figure 4.50 - Lifeboat's vertical CG position for H = 20 m

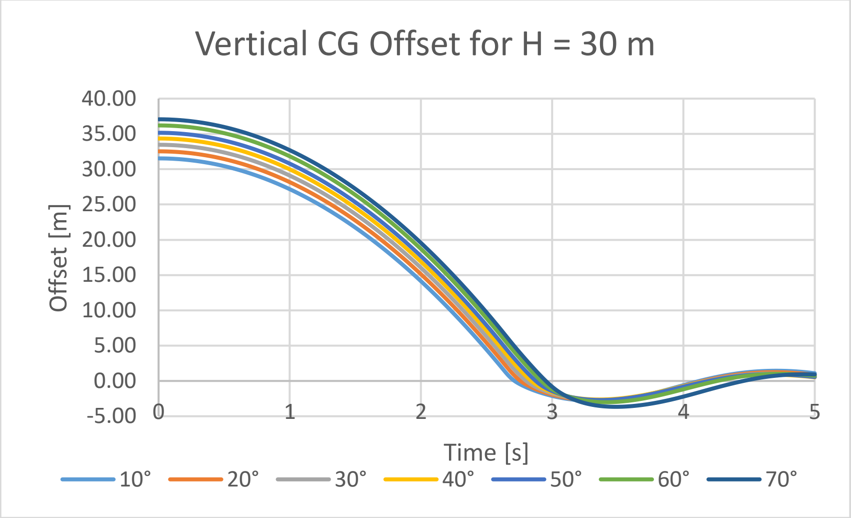


Figure 4.51 - Lifeboat's vertical CG position for H = 30 m

A parametric study between vertical and horizontal positions using time as a parameter can be done to reproduce the trajectory of the CG along time. Since all simulations were performed to have 5 seconds, it is possible to notice at what point the lifeboat is when $t = 5\ s$ which is when its trajectory is "interrupted". Besides the stabilization that the ship starts by damped oscillations after water entry, the greater angles have a longer first valley when entering the water, which is due to the pitch rotation that the vessel suffers to reach the equilibrium angle after a large amplitude is given. This also has to do with the submergence that the ship goes through at these particular high angles.

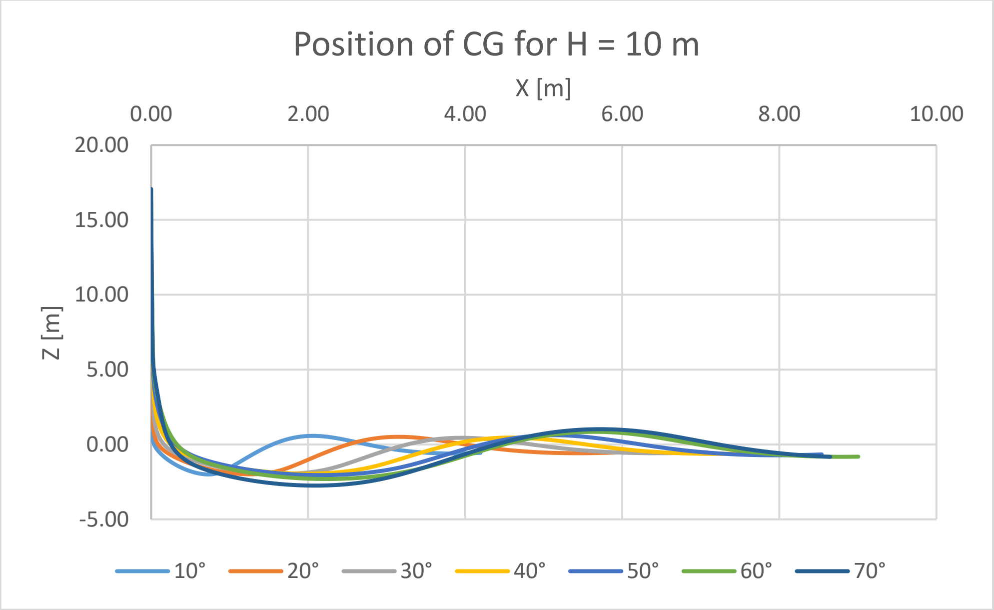


Figure 4.52 - Position of CG using time as parameter (H = 10 m)

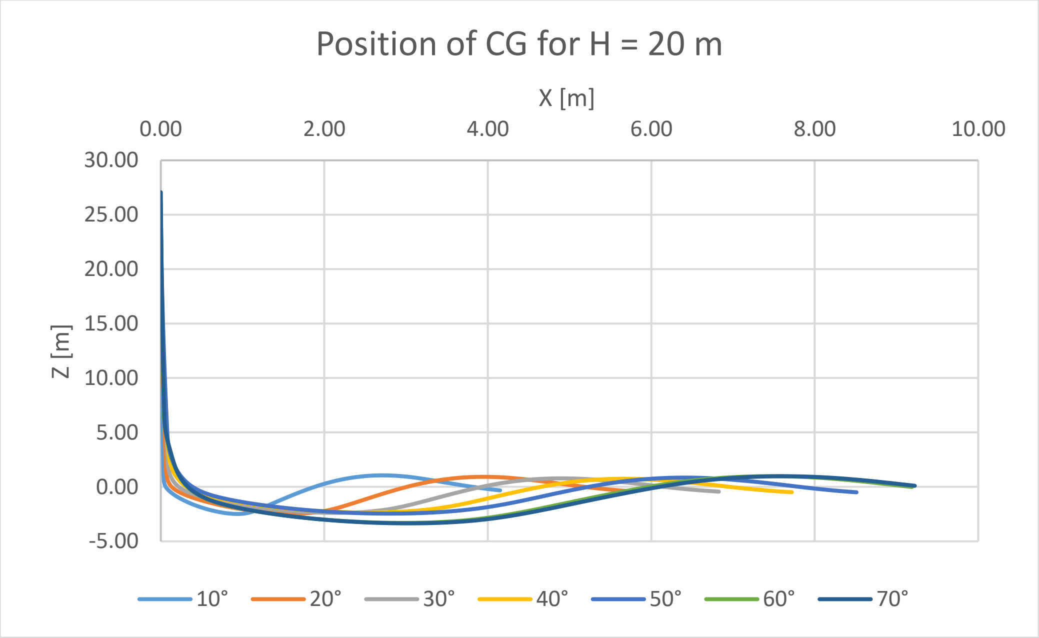


Figure 4.53 - Position of CG using time as parameter (H = 20 m)

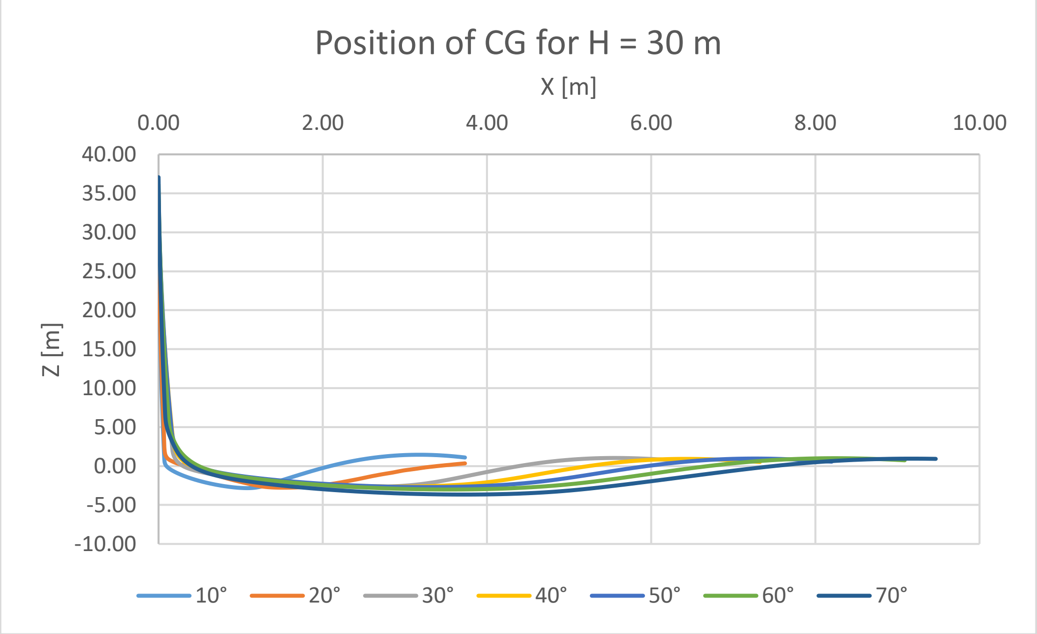


Figure 4.54 - Position of CG using time as parameter (H = 30 m)

It is also notable that the changes on the trajectory with different heights did not bring considerable differences as did the other parameters, except for the diving part or the number of oscillations. From one point of view, this can mean that changing the height did not bring a much faster response to leave the hazardous event. On the other hand, the analysis would be different if the free-falling time were not considered. If removed and just considering the water entry part, these 5 s simulations turn into 2.38 s limited by the 30 m drop height case. This already shows differences in Figure 4.55.

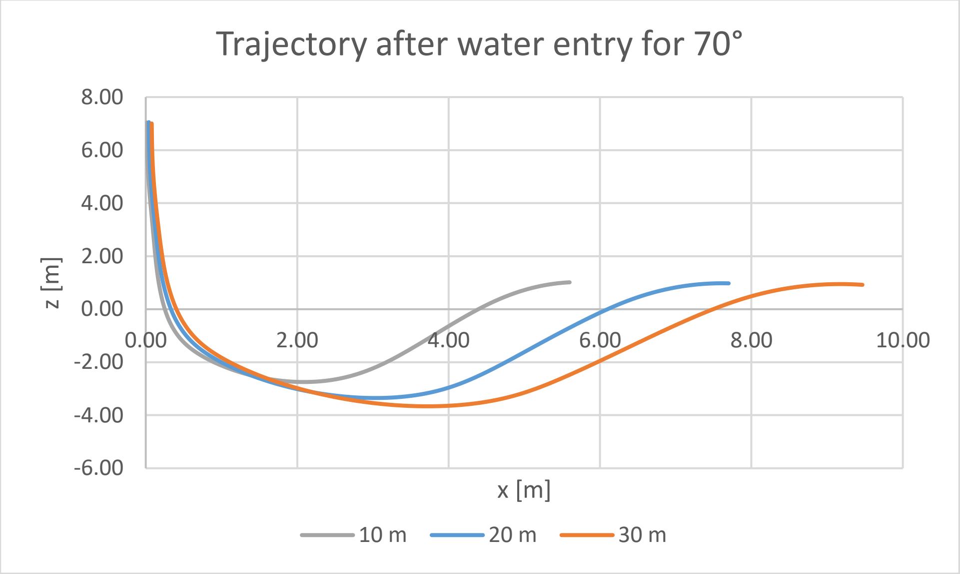


Figure 4.55 - Position of CG for water entry with α = 70°

The same behaviour from the vertical offsets can be seen in the time series of pitch, except for the constant angle when the boat is free-falling, due to the absence of relevant torque. What can be seen is that the larger the angle, the larger the spin for the lifeboat to reach equilibrium. Since the minimum peaks are almost near each other, this also means that the angular velocity and acceleration are higher for greater angles.

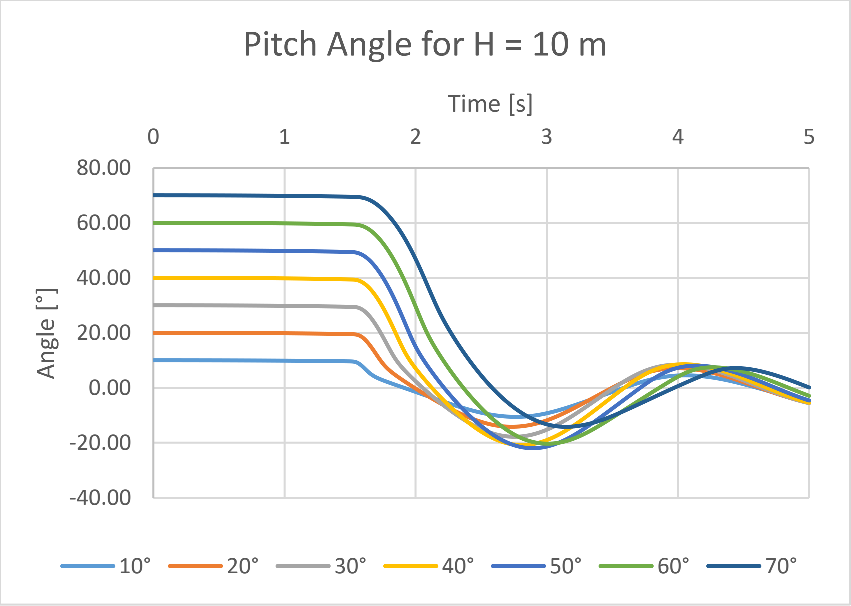


Figure 4.56 - Lifeboat's pitch angle for H = 10 m

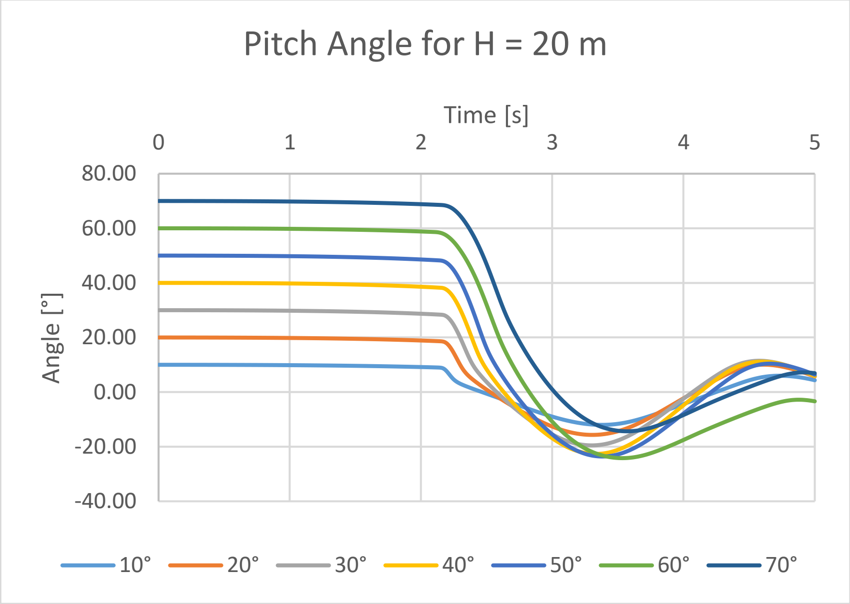


Figure 4.57 - Lifeboat's pitch angle for H = 20 m

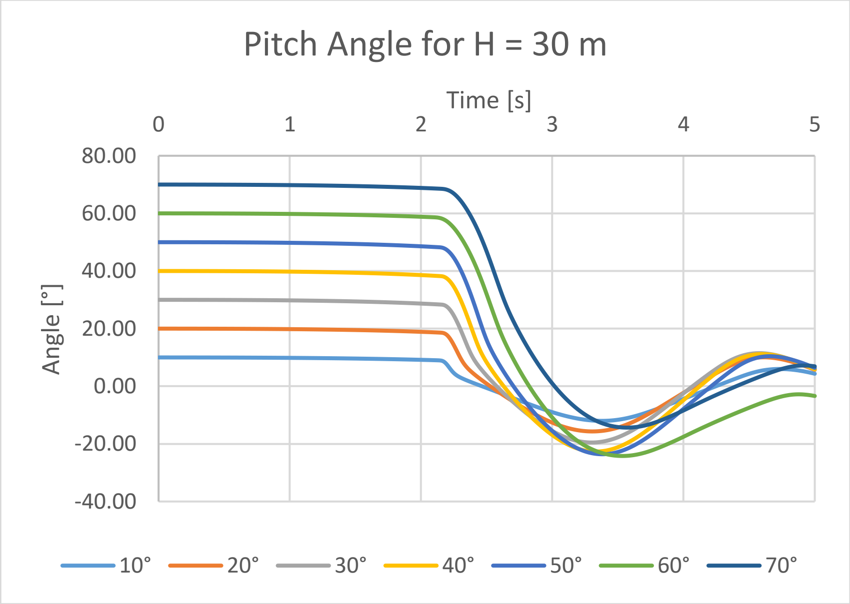


Figure 4.58 - Lifeboat's pitch angle for H = 10 m

The information on the CG position over time and pitch motion can tell how the motion of the lifeboat along its whole journey. More than that, OpenFOAM stores the parameter α of the whole domain for each time step set for output. This mapping can also be plotted using paraView, in special by removing the cells considered holes and slicing them in the middle of the longitudinal plan. The results of this simulation can be seen in Figure 4.55 to Figure 4.58, where just the angles 10°, 30°, 50°, and 70° were plotted and displayed.

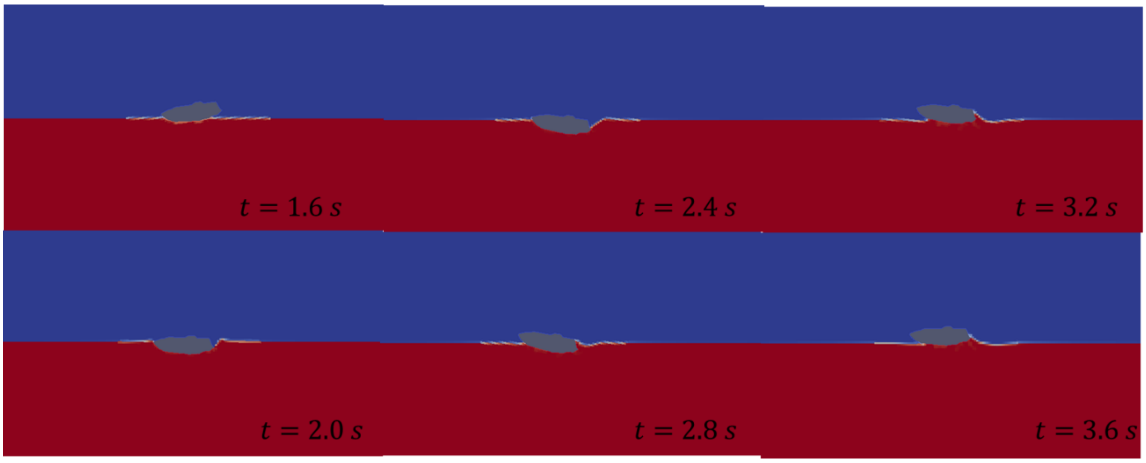


Figure 4.59 - Water contour for 10°

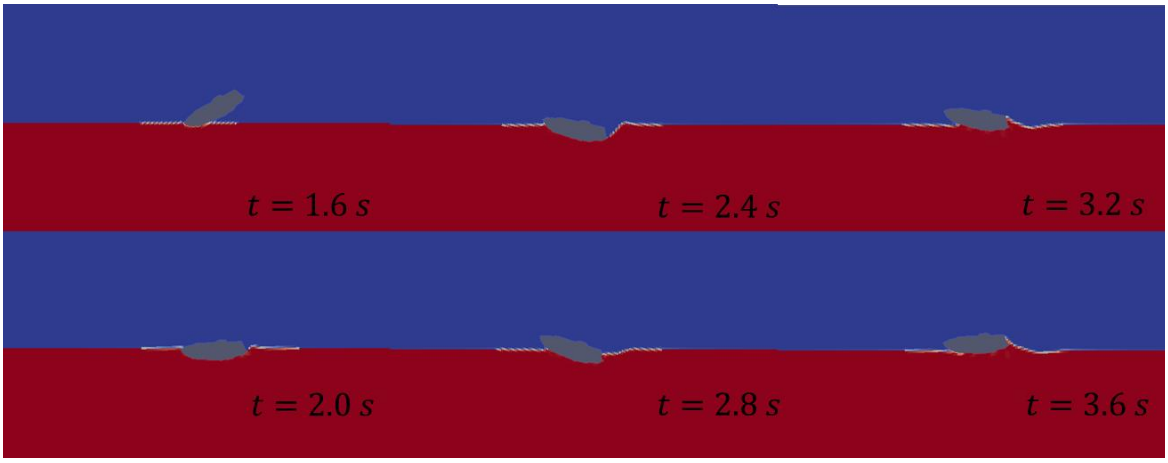


Figure 4.60 - Water contour for 30°

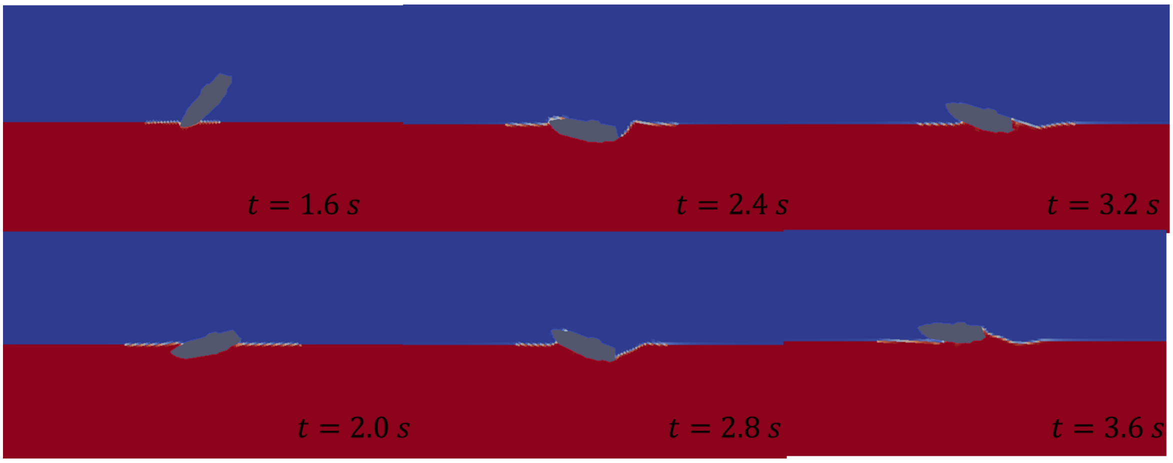


Figure 4.61 - Water contour for 50°

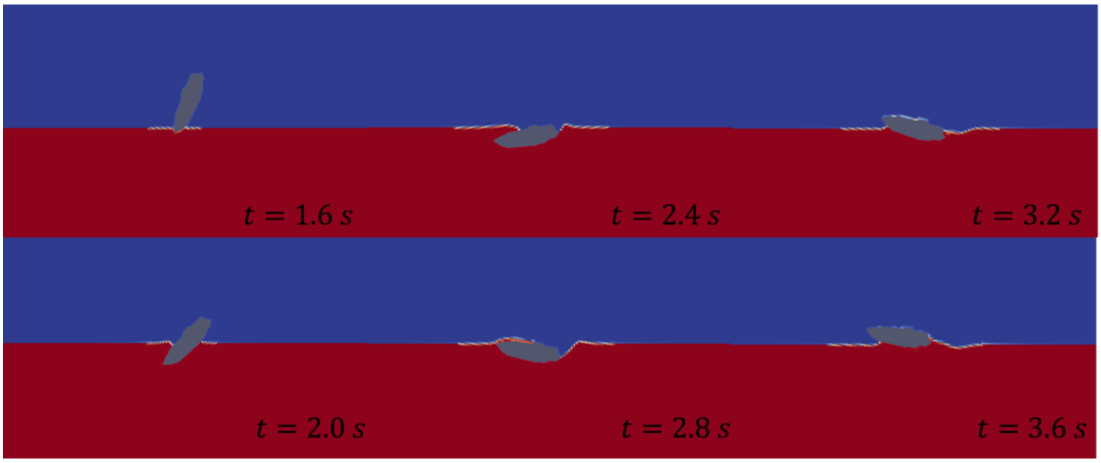


Figure 4.62 - Water contour for 70°

Comparing the water entry of the presented situations, it is notable how the increment of the angle makes the lifeboat more likely to submerge, as seen in Figure 4.59, where the lifeboat dives closer to the free surface, while for 70°, in Figure 4.62, the boat already enters almost completely under the water, which did not occur because it did not have enough velocity to it. For the first two angles, mostly just the under hull had hit the water, making the keel absorb most of the impact and converting less energy into actual motion for the lifeboat.

4.3 Conclusion

Simulations of a freefall lifeboat were carried out in order to assess impact loads, predict how the lifeboat will behave when launched with a certain angle and dropping height, and also evaluate if OpenFOAM can be a good tool for this type of analysis. Similarly, the pressures and kinematics of the lifeboat were compared.

The first step was to define the lifeboat model and set the inputs of the simulation. A geometry based on the Schat Harding 1000 was modelled in Rhino with the data available in the literature. The creation of the model was possible with a penalty of having distortions of about 5.4%regarding the original dimensions and some chines simplified. Even so, the model is still based on a real lifeboat with reasonable parameters for this study.

The preliminary analysis of the simulation has shown that the simulation of this phenomenon is demands more on computer capacity than the 2D wedge simulations since the scale and complexity of the problem were raised. In order to have converging simulations in a reasonable amount of time to reduce the resolution of the snappyHexMesh problem and use a less refined mesh for both: overset and background mesh. The comparison with experimental pressures from MARINTEK studies resulted in a similar curve with slower decay and not matching numerical peak pressures. This has mostly to do with mesh refinement and differences in geometry. Applying turbulence model could also help to improve this result.

After this, several simulations for various dropping angles and heights were performed and compared. Regarding the differences in height, usually, the parameters pressure, acceleration, velocity, and displacement tend to increase with the increment of height, which is expected. As for angles, the behaviour changed for each case.

Regarding the pressures, in general, the peaks are concentrated in two regions: the bow deck where it first hits the water, and the stern, where it occurs the second water entry due to the boat spinning back after the restoration moment had acted. This stern peak is usually higher than the bow, except for the 70°case where the submergence of the boat smoothed the peak. The pressure time series has also shown that 70° usually has lower loads, while the others alternate which has a higher peak.

As for the kinematics, it is important to see how far the lifeboat can get away from the hazardous event. In general, 70° has shown a good performance in this, where it got the highest speeds in a small amount of time, while 10° was the worse. The acceleration can also be an assessment for how probable someone may get injuries inside the lifeboat, and by applying IMO's criteria, just 70° and 60° can be used for a 20 m dropping height, while every case would be less probable for 10 m and would also have a higher probability for 30 m.

5. CONCLUSIONS AND RECOMMENDATIONS

5.1 Main Conclusions

The slamming phenomenon has still been a long-lasting topic for the naval industry and engineering. It started more than 80 years ago with the work of Von Kármán (1929), which used the approach of momentum theory to describe these loads and evolved into several methods that are still in development nowadays. To approach this problem, there are analytical solutions, in which the most popular was the studies of Wagner (1932), which used potential flow with local jet analysis to reach an asymptotic solution for water entry of wedges. There are also numerical methods, such as the BEM, pioneered by Zhao & Faltinsen (1993) and further developed to account for gravity and non-linear terms, and also the computational fluid dynamics, where the FVM can well account for some hydrodynamic aspects, such as the flow separation and viscosity, previously neglected, and also complex structures, such as lifeboats, the main topic of this research.

For conducting the studies of slamming, OpenFOAM was selected as the CFD software to perform these numerical calculations due to its open-source philosophy and multiphase solvers implemented. The numerical method used for this purpose was the interFoam family of solvers, which uses the VOF approach with incompressible fluid for implementing a two-phase fluid scenario with air and water. With this, the overset mesh method was used as a meshing technique, where two independent domains are created: the background, which contains information regarding the fluids, and the chimera, surrounding the floating body. This procedure is suitable for slamming problem that deals with high amplitude motions, and it does not have the risk of crashing the simulation due to the large deformation as the morphing mesh technique does.

For the first approach with this methodology, simple two-dimensional simulations of the water entry of a 30° wedge were carried out. The scenario was fully based on the experimental investigation executed by Wang et al. (2015), where pressure and kinematic parameters, such as velocity, draught, and acceleration, were measured and can be used as validation data for the model, together with the BEM method also calculated by the researchers. Three simulations were performed with a constant refinement ratio of 2 for the later performance of uncertainty analysis.

The first results for pressure could be compared. In general, they presented good adherence with the BEM and experimental results, especially after the peak, which was the part that presented higher divergence. This part of interest is expected to have high uncertainties in both: CFD and experimental analysis, which could also be seen by the error bar in the plots of Wang et al. (2015). As for the kinematics parameters, the adherence was better with the BEM method results, while a higher discrepancy was observed between these both results and the experimental one, especially in velocity and acceleration terms. This difference occurred mainly due to the three-dimensional nature of the experiment, while the models are both two-dimensional with differences in approach only.

The convergence of the parameter's maximum acceleration and pressure coefficient was also examined using the constant CFL method to perform uncertainty analysis. By adding correction factors, it is discovered that both oscillatory convergence and uncertainty fall below 5% and even below 1%. This proves that a higher resolution does not always equate to better outcomes, as evidenced by the maximum acceleration uncertainties. Finally, a comparison between overset and morphing mesh techniques was performed based on the studies of Wang et al (2021). The majority of the results were consistent, with differences being visible at pressure peaks, typically more conservative for deforming mesh, and in the resolution of the results, demonstrated to be better for overset mesh. Even though the computational time for morphing mesh was somewhat lower than that for chimera, both presented to be reliable methods to assess this problem and can be used to look at other hydrodynamic impact problems.

The main focus of this project was to apply this combining VOF method and overset mesh technique with freefall lifeboats. These structures are used for evacuating a ship when facing a dangerous situation, and the high velocity at which they enter implies slamming loads that need to be assessed to ensure safety and structure integrity. The lifeboat model needed to be defined, and the simulation's inputs needed to be set. Using the data from the literature, a geometry based on the Schat Harding 1000 was modelled with simplified chines and distortions to the original dimensions of roughly 5.4%. The model is nevertheless based on a genuine lifeboat and has appropriate characteristics for our purpose.

The simulations demonstrated to have a higher complexity compared to the wedge simulation, which is something to expect when entering the three-dimensional simulations universe with more degrees of freedom, which is why just two were considered, pitch and heave and a less refined mesh. Discrepancies between the pressure values were found when compared to the Marintek experimental measurements due to these reasons and also the differences of nature to obtain this result.

Following this, several simulations were run for various descending angles and heights and then compared. The parameters pressure, acceleration, and velocity typically tend to rise with the height increment, which is to be expected. The displacement depends on considering just the water entry process or the whole journey with the freefall part, in which the first tends to be fairly different while the other does not present considerable differences.

In terms of pressures, the peaks are typically focused in two areas: the bow deck, where the boat first enters the water, and the stern, where it enters the water again. Except for the 70° case, where the boat's submersion softened the peak, this stern peak is typically higher than the bow. The kinematics was used to also assess how harmful can be the freefall and the water entry process for people by assessing accelerations. It has shown that for 60° and 70°, the probability is the lowest when compared to other angles, while the drop height of 30 m was always dangerous when considering IMO’s criteria. Another aspect of kinematics was how well a lifeboat can escape the hazardous event without extra propulsion, which would require time until the given acceleration turns into actual velocity. From this point of view, 70° was the one that performed the best, whereas 10°, was the least, showing how effective can be the submergence phase.

Finally, the software OpenFOAM has shown to be a powerful tool for assessing slamming loads, but it has limitations, as do all measurement tools and methods. It is only possible to use OpenFOAM on a Linux operating system with high computational capacity and possibility to parallelize the simulations, otherwise they can take a long time to run. Furthermore, for new users is a challenge because not everything is well documented, and the environment lacks a guided user interface (GUI). Currently, the fluid-structure interaction (FSI) is only possible with rigid body, meaning it is not yet feasible to work with flexible structures in the present version.

5.2 Recommendations for Future Work

In general, the project has shown that OpenFOAM is a robust tool to deal with slamming problems and can be used not only to measure the impact loads but also to view the performance of the lifeboat when entering the water to flee from the hazardous event and save lives. Much can be done to further improve and develop the methodology applied.

One of the steps that could be further improved is the lifeboat validation. Several simplifying hypotheses were applied, starting with a similar but not the same geometry, which can change considerably the results to be compared. Even with enough pictures and views to recreate a replica of the lifeboat, these images can bring distortions to some important parts of the craft which only possible to assess its relevance by testing with a high-fidelity representation of the lifeboat.

The model itself can also go through further development. Here, the priority was to find converging and coherent solutions for the full journey of the vessel, starting from the freefall. A model where the mesh is fine enough to assess the results in a reasonable amount of time while converging results is still to be matured. In addition, the laminar assumption can be removed by adding turbulence models. For instance, the SST $k - \omega$ model's capacity to simulate the flow along boat hulls has been used to demonstrate that it is a reliable RANS turbulence modelling technique. This inclusion could be compared to the already known laminar simulations with possible benefits and trade-offs.

OpenFOAM is still a tool to be further explored, as well as the freefall lifeboat scenario. Here, just the freefall over calm water is studied, but the real freefall lifeboat and also part of the experimental measurements from Kauczynski et al. (2009) were done over waves. OpenFOAM is capable of setting waves by defining boundary conditions and specifying wave models, which can vary from solitary and regular models (OpenFOAM, 2019). Depending on the modeling, the further 3D parameter would be needed to be estimated or obtained, such as the whole inertia matrix.

A great asset to be achieved in this whole project is the integration of the CFD simulations with the finite element structural model for a fluid-structure investigation (FSI). This would require a combination of solvers exchanging information on pressures and forces (CFD to FEM) and displacement (FEM to CFD). A strong candidate that could combine both is the preCICE coupling library, which already integrates OpenFOAM to several FEM software, where the main one is CalculiX (preCICE, 2022). The adapter is still in continuous progress to support many features, which include motions of larger scale and corrections regarding gravity in the FEM model, but can already run simulations with the overset environment in OpenFOAM, for instance.

6. REFERENCES

Albadawi, A., Donoghue, D. B., Robinson , A. J., Murray, D. B., & Delauré, Y. M. (2013). Influence of surface tension implementation in Volume of Fluid and coupled Volume of Fluid with Level Set methods for bubble growth and detachment. *International Journal of Multiphase Flow, 53*, 11-28. doi:10.1016/j.ijmultiphaseflow.2013.01.005

Alletto, M. (2022). Comparison of Overset Mesh with Morphing Mesh: Flow Over a Forced Oscillating and Freely Oscillating 2D Cylinder. *OpenFOAM® Journal, 2*, 13-30. doi:10.51560/ofj.v2.47

Andrya, S., Janardhanan, S., & Nair, E. (2019, May). Structural Response of Free -Fall Lifeboats During Emergencies. *International Journal of Recent Technology and Engineering (IJRTE)*, 211-216.

Arai, M., Cheng, L.-Y., & Inoue, Y. (1994). A Computing Method for the Analysis of Water Impact of Arbitrary Shaped Bodies. *Journal of the Society of Naval Architects of Japan, 1994*(176), 233-240. doi:10.2534/jjasnaoe1968.1994.176_233

Benites-Munoz, D., Huang, L., Anderlini, E., Marín-Lopez, J. R., & Thomas, G. (2020). Hydrodynamic Modelling of An Oscillating Wave Surge Converter Including Power Take-Of. *Journal of Marine Science and Engineering, 8*(10), 771. doi:10.3390/jmse8100771

Berton, A., D'Orrico, F., & Sideri, M. (2017). Overset grids for fluid dynamics analysis of internal combustion engines. *Energy Procedia, 126*, 979-986. doi:10.1016/j.egypro.2017.08.240

Cointe, R., & Armand, J.-L. (1987). Hydrodynamic Impact Analysis of a Cylinder. *J. Offshore Mech. Arct. Eng., 109*(3), 237-243. doi:10.1115/1.3257015

Dobrovol'skaya, Z. (1969). On some problems of similarity flow of fluid with a free surface. *Journal of Fluid Mechanics, 36*(4), 805-829. doi:10.1017/s0022112069001996

Duraisamy, K., Iaccarino, G., & Xiao, H. (2019). Turbulence Modeling in the Age of Data. *Annual Review of Fluid Mechanics, 51*(1), 357-377. doi:10.1146/annurev-fluid-010518-040547

Faltinsen, O. (1990). Water Impact and Entry. In O. M. Faltinsen, *Sea Loads on Ships and Offshore Structures* (pp. 282-308). Cambridge: Cambridge Univertisy Press.

Gong, K., Liu, H., & Wang, B. (2009). Water Entry of a Wedge Based on Sph Model with an Improved Boundary Treatment. *Journal of Hydrodynamics, 21*(6), 750-757. doi:10.1016/S1001-6058(08)60209-7

Greenhow, M., & Lin, W.-H. (1985). Numerical simulation of nonlinear free surface flows generated by wedge entry and wavemaker motions. *In Proc. 4th Int. Conf. on Numerical Ship*, (pp. 24-27). Washington, DC.

Greenshields, C. J. (2015, May 21). OpenFOAM, The open source CFD Toolbox. *User Guide*.

Howison, S., Ockendon, J., & Wilson, S. (1991). Incompressible water-entry problems at small deadrise angles. *Journal of Fluid Mechanics, 222*, 215-230. doi:10.1017/s0022112091001076

Huang, L., Tavakoli, S., Li, M., Dolatshah, A., Pena, B., Ding, B., & Dashtimanesh, A. (2021). CFD analyses on the water entry process of a freefall lifeboat. *Ocean Engineering, 232*, 109115. doi:10.1016/j.oceaneng.2021.109115

Huang, L., Tuhkuri, J., Igrec, B., Li, M., Stagonas, D., Toffoli, A., . . . Thomas, G. (2020). Ship resistance when operating in floating ice floes: A combined CFD&DEM approach. *Marine Structures, 74*, 102-817. doi:10.1016/j.marstruc.2020.102817

IMO. (1991, November 6). *Resolution A.689*. IMO.

Iranmanesh, A., & Passandideh-Fard, M. (2017). A three-dimensional numerical approach on water entry of a horizontal circular cylinder using the volume of fluid technique. *Ocean Engineering, 130*, 557-566. doi:10.1016/j.oceaneng.2016.12.018

Kapsenberg, G. (2011). Slamming of ships: where are we now? *Phil. Trans. R. Soc. A., 369*, 2892–2919. doi:10.1098/rsta.2011.0118

Kárman, T. v. (1929). *The impact on seaplane floats during landing.* Washington, DC;: National Advisory Committee for Aeronautics.

Kauczynski, W. E., Werenskiold, P., & Narten, F. (2009). Documentation of operational limits of free-fall lifeboats by combining model tests, full-scale tests, and computer simulation. *Proceedings of the ASME 2009 28th International Conference on Ocean, Offshore and Artic Engineering. Volume 2: Structures, Safety and Reliability, 2*. doi:10.1115/OMAE2009-79959

Khojasteh, D., Tavakoli, S., Dashtimanesh, A., Dolastshah, A., Huang, L., Glamore, W., . . . Iglesias, G. (2020). Numerical analysis of shipping water impacting a step structure. *Ocean Engineering, 209*, 107-517. doi:10.1016/j.oceaneng.2020.107517

Korobkin, A. (2005). Analytical models of water impact. *European Journal of Applied Mathematics, 15*(6), 821-838. doi:10.1017/S0956792504005765

Larsen, B. E., Fuhrman, D. R., & Roenby, J. (2019). Performance of interFoam on the simulation of progressive waves. *Coastal Engineering Journal, 61*(3), 380-400. doi:10.1080/21664250.2019.1609713

Lopez Mejia, O. D., Mejia, O. E., Escorcia, K. M., Suarez, F., & Laín, S. (2021). Comparison of Sliding and Overset Mesh Techniques in the Simulation of a Vertical Axis Turbine for Hydrokinetic Applications. *Processes*, 1933. doi:10.3390/pr9111933

Ma, Z., Qian, L., Martínez-Ferrer, P., Causon, D., Mingham, C., & Bai, W. (2018). An overset mesh based multiphase flow solver for water entry. *Computers and Fluids, 172*, 689-705. doi:10.1016/j.compfluid.2018.01.025

Manafpour, M., & Hamzeh, E. (2017). The Multiphase Capability of OpenFoam CFD Toolbox in Solving Flow Field in Hydraulic Structure. doi:10.3217/978-3-85125-564-5-045

Mittal, R., & Laccarino, G. (2005). Immersed boundary method. *Annual Review of Fluid Mechanics, 37*(1), 239-261. doi:10.1146/annurev.fluid.37.061903.175743

Monaghan, J. (1988). An introduction to SPH. *Computer Physics Communications, 48*(1), 89-96. doi:10.1016/0010-4655(88)90026-4

Monaghan, J. (1992). Smoothed Particle Hydrodynamics. *Annual Review of Astronomy and Astrophysics, 30*, 543-574. doi:10.1146/annurev.aa.30.090192.002551

Monaghan, J. (2005). Smoothed particle hydrodynamics. *Rep. Prog. Phys., 68*, 1703. doi:10.1088/0034-4885/68/8/R01

Muzaferija, S. (1998). A two-fluid Navier-Stokes solver to simulate water entry. *Proceeding of the 22nd symposium of naval hydrodynamics*.

Netherlands Regulatory Framework - Maritime. (1993, 06 22). *616 Evaluation of Free-fall Lifeboats launch performance*. Retrieved from Netherlands Regulatory Framework (NeRF) – Maritime: https://puc.overheid.nl/nsi/doc/PUC_1746_14/1/

Oberhagemann, J. (2016). *On Prediction of Wave-Induced Loads and Vibration of Ship Structures with Finite Volume Fluid Dynamic Methods.* Duisburg: Doctoral Dissertation.

Occhi, M. K. (1964). Extreme behaviour of a ship in rough seas - Slamming and shipping of green water. *Proceedings of the 5th Symposium on Naval Hydrodynamics* (p. 2). Washington D.C.: Naval Ship Research and Development Center.

Occhi, M. K. (1964). Prediction of occurrence and severity of ship slamming at sea. *Proceedings of the 5th Symposium on Naval Hydrodynamics.* Washington D.C.: Naval Ship Research and Development Center.

Occhi, M. K., & Motter, L. (1971). A Method to Estimate Slamming Characteristics for Ship Design. *Mar Technol SNAME, 8*(2), 219–232. doi:10.5957/mt1.1971.8.2.219

Occhi, M. K., & Motter, L. (1973). Prediction of slamming characteristics and hull responses for ship design. *Trans. SNAME*, 144-176.

Ogilvie, & Francis, T. (1963). Compressibility effects in ship slamming. *Schiffstechnik*, 147-154. Retrieved from http://resolver.tudelft.nl/uuid:d7ce4504-6d78-4b04-8c09-796cbc2857de

OpenFOAM. (2019). *Wave modelling*. Retrieved from OpenFOAM: User Guide v2112: https://www.openfoam.com/documentation/guides/latest/doc/guide-wave-modelling.html

Pabst, W. (1930). *Theory of the landing impact of seaplanes.* Washington DC.: National Advisory Committee for Aeronautics.

Paterson, E. G., Wilson, R. V., & Stern, F. (2003). *General-Purpose Parallel Unsteady Rans Ship Hydrodynamics Code: CFDSHIP-IOWA.* College of Engineering , Hydroscience & Engineering. Iowa City: The University of Iowa.

Pearce, C. (2020, April 22). *BBC Science Focus Magazine*. Retrieved from BBC Science Focus Magazine: https://www.sciencefocus.com/science/whats-the-maximum-speed-a-human-can-withstand/

Pena, B., Muk-Pavic, E., Thomas, G., & Fitzsimmons, P. (2019). Numerical analysis of a leading edge tubercle hydrofoil in turbulent regime. *Journal of Fluid Mechanics, 878*, 292-305. doi:10.1017/jfm.2019.611

preCICE. (2022). *preCICE*. Retrieved from Welcome to preCICE: https://precice.org/index.html

Qiu, S., Ren, H., & Li, H. (2020). Computational Model for Simulation of Lifeboat. *Journal of Marine Science and Engineering, 8*(9), 631. doi:10.3390/jmse8090631

Qiu, S., Ren, H., & Li, H. (2020, August). Computational Model for Simulation of Lifeboat Free-Fall during Its Launching from Ship in Rough Seas. *Journal of Marine Science and Engineering, 8*, 631. doi:10.3390/jmse8090631

Review of code and solution verification procedures for computational simulation. (2005). *Journal of Computational Physics, 205*(1), 131-156. doi:10.1016/j.jcp.2004.10.036

Ringsberg, J. W., Heggelund, S., Lara, P., Jang, B.-S., & Hirdaris, S. E. (2017). Structural response analysis of slamming impact on free fall lifeboats. *Marine Structures, 54*, 112-126. doi:10.1016/j.marstruc.2017.03.004

Roache, P. J. (1998). Criticisms of the "Correction Factor" Verification Method. *Journal of Fluids Engineering, 125*(4), 732-733. doi:10.1115/1.1588693

Shen, Z., Hsieh, Y.-F., Ge, Z., Korpus, R., & Huan, J. (2016). Slamming Load Prediction Using Overset CFD Methods. *Offshore Technology Conference*. doi:10.4043/27254-MS

Sun, H., & Faltinsen, O. M. (2007). The influence of gravity on the performance of planing vessels in calm water. *Journal of Engineering Mathematics, 58*, 91-107. doi:10.1007/s10665-006-9107-5

Sun, P., Marrone, S., Ming, F., & Zhang, A.-M. (2018). An accurate and efficient SPH modeling of the water entry of circular cylinders. *Applied Ocean Research, 72*, 60-75. doi:10.1016/j.apor.2018.01.004

Tisovska, P. (2019). Description of the overset mesh approach in ESI version of OpenFOAM. *In Proceedings of CFD with OpenSource Software*. doi:10.17196/OS_CFD#YEAR_2019

Wang, J., Lugni, C., & Faltinsen, O. M. (2015). Experimental and numerical investigation of a freefall wedge vertically entering the water surface. *Applied Ocean Research, 51*, 181-203. doi:https://doi.org/10.1016/j.apor.2015.04.003

Wang, S., & Guedes Soares, C. (2013). Comparison of simplified approaches and numerical tools to predict the loads on bottom slamming of ship structures. *Developments in Maritime Transportation and Exploitation of Sea Resources*, 157-168. doi:10.1201/b15813-21

Wang, S., & Guedes Soares, C. (2020). Effects of compressibility, three-dimensionality and air cavity on a free-falling wedge cylinder. *Ocean Engineering, 217*, 107-589. doi:10.1016/j.oceaneng.2020.107589

Wang, S., González-Cao, J., Islam, H., Gómez-Gesteira, M., & Guedes Soares, C. (2022). Uncertainty estimation of mesh-free and mesh-based simulations of the dynamics of floaters. *Ocean Engineering, 256*, 111386. doi:10.1016/j.oceaneng.2022.111386

Wang, S., Xiang, G., & Guedes Soares, C. (2021). Assessment of three-dimensional effects on slamming load predictions using OpenFoam. *Applied Ocean Research, 112*, 102646. doi:https://doi.org/10.1016/j.apor.2021.102646

Wu, G.-X., Xu, G.-D., & Duan, W.-Y. (2010). A summary of water entry problem of a wedge based on the fully nonlinear velocity potential theory. *Journal of Hydrodynamics, 22*(5), 859-864. doi:10.1016/S1001-6058(10)60042-X

Xiao, H., & Paola, C. (2019). Quantification of model uncertainty in RANS simulations: A review. *Progress in Aerospace Sciences, 108*, 1-31. doi:10.1016/j.paerosci.2018.10.001

Yettou, E.-M., Desrochers, A., & Champoux, Y. (2007). A new analytical model for pressure estimation of symmetrical water impact of a rigid wedge at variable velocities. *Journal of Fluids and Structures, 23*(3), 501-522. doi:10.1016/j.jfluidstructs.2006.10.001

Zhao, R., & Faltinsen, O. (1993). Water entry of two-dimensional bodies. *Journal of Fluid Mechanics, 246*, 593-612. doi:10.1017/S002211209300028X

Zhao, R., Faltinsen, O. M., & Aarsnes, J. V. (1996). *Water Entry of Arbitrary Two-Dimensional Sections withand Without Flow Separation.* MARINTEK, Norwegian University of Science and Technology.

Zheng, K., Yang, Z., Zhao, X., & Chaofan, L. (2019). Numerical simulation of water entry of a wedge using a modified ghost-cell immersed boundary method. *Journal of Marine Science and Technology, 25*(2). doi:10.1007/s00773-019-00666-9