Delft University of Technology

CIE5050-09 Additional Gradutation Work, Research Project

CFD modelling: The interaction between extreme waves and a lighthouse upon a shoal

Ruben Ansorena Ruiz 4718208

2018-2019

Preface

This document represents the additional graduation work (CIE5050-09, TU Delft) of Ruben Ansorena Ruiz. First of all I would like to thank my colleagues Akshay Patil and Marco Moretto for guiding me on both the Ubuntu operative system and CFD programming. Besides, thanks to my daily supervisor Alessandro Antonini and professor Marion Tissier for their guidance and support during the realization of this thesis.

After delivering this project, a spine is left inside of me. I am confident that by learning how to mesh using software and with some more time I could deliver a proper 3D simulation. For that reason, I will come back to this in the future. I just wanted to leave this written in paper.

Ruben Ansorena Ruiz
July 19th, 2019. Delft, The Netherlands.

Table of contents

Chapter 1

Introduction

Lighthouses are structures subjected to extreme weather conditions which have to resist strong surge. Therefore to ensure their survival over time, it is necessary to determine the loads that these structures will face during its lifetime.

However, load prediction on lighthouses is troublesome due to the multiple factors that affect the loading. First, the geometries of lighthouses vary considerably from one to another, not usually representing the classic cylindrical shape by which they are modelled in laboratories. Second, climate change transforms the statistical properties of the sea states and will bring more extreme events in the near future. As so as sea level rise. Lastly, the bathymetry around a lighthouse is considerably different to the ones used to model lighthouses in labs, which can affect wave propagation and breaking. Therefore loading is affected.

As the traditional physical methods are both expensive and time consuming, researchers have developed computational tools to accurately represent reality. In this report, waves2Foam (a toolbox within OpenFOAM) will be used to reproduce some tests performed on a physical wave flume. The goal is to represent the conditions of the lab in the computational model so that this model can then be used to represent different geometries and wave loads over lighthouses in an easy and cheap way.

Additionally, the most of the current tests were performed with waves directly hitting the lighthouse structure. However in this study, the waves break over a shoal and then the mass of water reaches the structure. This causes a smaller maximum load but the load is applied during a longer time that when a wave breaks directly on the lighthouse.

In this study the physical model performed by Piermodesto Caputo (year 2017/2018) and supervised by Professor Renata Archetti and Dr. Alessandro Antonini (co-supervisor) will be modeled using waves2Foam (OpenFOAM). The physical model was performed in Plymouth University laboratory ”COAST”. The goal of the model was to determine the loads that the Dubh Artach lighthouse will face in the future taking into account sea level rise and extreme weather conditions due to climate change.

Chapter 2

Problem Statement and Goals

Information about waves breaking on a lighthouse is not easily available. Currently, physical tests are performed to predict loads over lighthouses, however these tests are expensive and time consuming. By modelling a wave flume in OpenFOAM, large amounts of test can be performed at the same time, decreasing the costs of performing tests and eliminating the need of setting up multiple physical tests.

The goal of the computational simulation is therefore representing a physical wave flume in OpenFOAM so that different situations can be modelled in an easy, cheap and efficient way. For doing that, the computational outputs of water surface elevation and force will be compared to the ones from the available physical flume. Only analysis for regular waves will be considered in this thesis.

Chapter 3

Physical model

The experiments were conducted in the wave flume tank of the COAST laboratory of Plymouth University. This facility is 35 m long, 0.6 m wide and has a maximum working depth of 0.8 m. The side walls and the central position of the bed, are glass-made for maximum optical access. Regular waves, focused waves and plunging breakers are all possible, with a maximum height of 0.35m. Although the paddle has an active reflected wave energy absorption, a poly-ether foam is placed at the downstream end of the flume to ensure a passive absorption rate. The study of the bathymetry proved that the rock laying underneath the lighthouse could be reliably scaled by a 1:5 inclined slope. The slope, 3.18 m long 0.5 m high, was holed at both sides to allow free circulation of water. Besides, it also had some holes for the wave gauges installation. During the testing phase the shoal was completely covered by water; this means that the water level was set at 0.5 m. The lighthouse structure is represented by a steel cylinder. The cylinder's dimensions are: 12 cm diameter and height of 50 cm. At the top, the cylinder was fixed under a traverse structure crossing the flume in the height of its upper edge. Thus, the test cylinder can be considered as a structural component with a two-points bearing, which is free of transverse static forces.

In the next figure, the wave flume dimensions are shown.

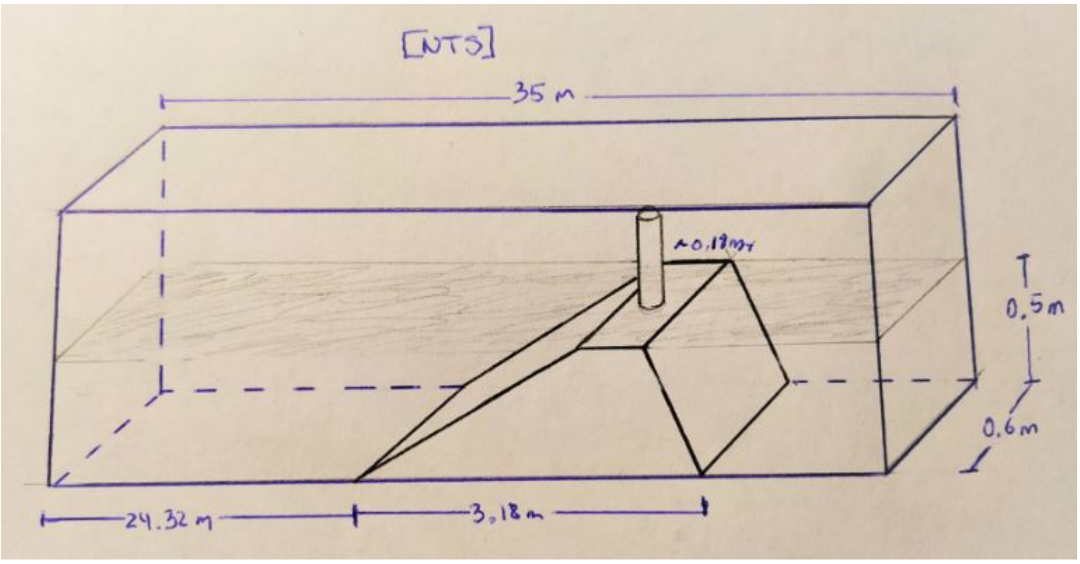

Figure 3.1: Flume representation. Not To Scale [NTS]

In the picture above, the wave generation zone is at the leftmost part whereas the wave absorption zone is at the rightmost part. Waves propagate from left to right.

The experiment was performed for multiple wave climates. However, for simplicity and time limitations for this additional thesis work, only the simulations that generated regular waves are used.

3.1 Available data

As previously mentioned, several tests were performed in this flume but for the present project, only the regular wave data will be compared. The current available data comprises water surface elevation and forces at the cylinder. These parameters were obtained for the following tests:

Test ID	Wave height, H (m)	Wave frequency, Freq. (Hz)	Wave period, T (s)
1	0.10	0.83	1.2048
2		0.67	1.4925
3		0.54	1.8519
4		0.51	1.9608
5		0.41	2.4390
6	0.126	3	4
7		0.67	1.4925
8		0.54	1.8519
9		0.51	1.9608
10		0.41	2.4390
11	0.15	3	4
12		0.67	1.4925
13		0.54	1.8519
14		0.51	1.9608
15		0.41	2.4390

Table 3.1: Tests performed in the physical flume

Chapter 4

Governing equations and wave theories

4.1 Governing Equations

In the OpenFOAM software, Navier-Stokes equations are used to represent the fluid (water in this case) motion and hence the governing equations. These equations are as follows for each coordinate axis:

$$\rho g_x - \frac{\partial p}{\partial x} + \mu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right) = \rho\left(\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z}\right) \tag{4.1}$$

$$\rho g_y - \frac{\partial p}{\partial y} + \mu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2}\right) = \rho\left(\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z}\right) \tag{4.2}$$

$$\rho g_z - \frac{\partial p}{\partial z} + \mu\left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2}\right) = \rho\left(\frac{\partial w}{\partial t} + u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z}\right) \tag{4.3}$$

Where:

- ρ: water density. Fresh water since the water used in the physical flume was fresh water. $\rho = 1000\frac{kg}{m^3}$
- t: time [s]
- p: pressure [Pa]
- g: gravity acceleration. g = 9.81 [m/s]
- μ: viscosity. $\mu = 10^6[m^2/s]$
- u, v, w: velocity component in x, y and z direction respectively [m/s]

Furthermore as we are considering incompressible flow, ρ is constant and the continuity equation is as follows:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \tag{4.4}$$

4.2 Wave theories

Different wave theories have been developed over the past. Each of these theories have an application range over which they represent reality properly. Under the assumption of frictionless fluid, the Stokes waves theory governs. Therefore, this theory properly represents waves in deep (h/L $>$ 0.5) and intermediate (0.05 $>$ h/L $<$ 0.5) waters, since the influence of bottom friction over the waves is there negligible. When entering shallow waters (h/L $<$ 0.05), cnoidal and solitary wave theories represent better the wave propagation.

In order to use the proper wave theory for each situation, the figure 4.1 can be used. Figure 4.1 shows which wave theory to use taking into account:

- H: wave height [m]
- h: water depth [m]
- T: wave period [s]

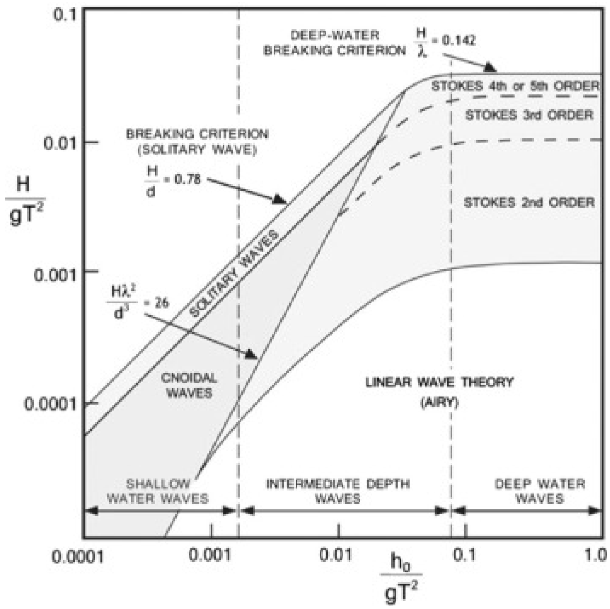

Figure 4.1: Applicability of various wave theories (Le Méhauté, 1976)

For the present research project, the wave heights and period used and their wave theory (according to Le Méhauté, (1976)) are shown:

- H = 0.15 m; T = 1.205: Stokes third order
- H = 0.126 m; T = 2.439: cnoidal wave theory
- H = 0.10 m; T = 1.205: Stokes second order

The above wave theories will be used as input for the simulations.

Chapter 5

Methodology

In order to model the physical flume, a first simplification was done. Therefore, the computational flume's length was reduced to 17.17 m. This flume starts at the point where the first gauge of the physical flume is (see figure A.7). By doing this, the amount of mesh cells will be drastically reduced and it won't cause any problem since in the first flume meters, the waves are just propagating.

At first, in order to ease the calculations and getting to adequate parameters more easily, 2D simulations will be performed (number of cells in the z-direction is 1). This simulations will not include the lighthouse structure, but just the shoal. This will ease the computational calculations and therefore will take less time to run a simulation. This is of importance since in the beginning the parameters need to be optimized in an iterative way. After the 2D model output resembles the physical flume data, the 2D simulation is converted to 3D and the new simulations can be performed until the computational model output should coincide with the physical flume data.

In order to properly model the waves propagating, some rules of thumb are used to define the mesh grid size:

$$\Delta Y = \frac{H}{20 - 40} \tag{5.1}$$

$$\Delta X = \frac{L}{80 - 100} \tag{5.2}$$

$$\frac{\Delta X}{\Delta Y} = 1 - 2 \tag{5.3}$$

Therefore, by playing with the different factors, a first estimate for the grid size can be given:

Test ID	H (m)	Freq. (Hz)	T (s)	Δ X (m)	Δ Y (m)	Δ X / Δ Y	No cells X	No cells Y	No cells Z
1		0.83	1.2048	0.014	0.0067	2.1	1250	180	44
2		0.67	1.4925	0.019	0.0067	2.8	917	180	32
3	0.10	0.54	1.8519	0.025	0.0067	3.7	696	180	24
4		0.51	1.9608	0.026	0.0067	4	650	180	23
5		0.41	2.4390	0.034	0.010	3.4	505	1200	18
6		0.83	1.2048	0.017	0.0084	2.0	1000	143	34
7		0.67	1.4925	0.023	0.0084	2.8	734	143	26
8	0.126	0.54	1.8519	0.025	0.0084	2.9	696	143	24
9		0.51	1.9608	0.026	0.0084	3.1	650	143	23
10		0.41	2.4390	0.030	0.0084	3.6	573	143	20
11		0.83	1.2048	0.015	0.0075	2	1166	160	41
12		0.67	1.4925	0.020	0.0075	2.7	856	160	30
13	0.15	0.54	1.8519	0.025	0.0088	2.8	696	136	24
14		0.51	1.9608	0.026	0.0088	3	650	136	23
15		0.41	2.4390	0.034	0.0100	3.4	505	120	18

Table 5.1: Parameters for defining grid size

Notice that the table above is just an initial estimate of the different grid sizes. Later the grid size will be modified until the one that gives the most accurate correlation between the physical and empirical data is reached. Besides, not all this tests will be performed due to the time available for developing this additional thesis.

So at this point, the different tests to perform are already defined. In the following lines, the different solvers and tools used in waves2Foam will be briefly explained (see appendix A for further information about the solvers).

- blockMesh: Tool for creating paralepipped meshes.
- snappyHexMesh: Additional tool for accurately define a mesh created with blockMesh. Uses a .stl surface to trim the mesh generated by blockMesh.
- waveGaugesNProbes: Tool for introducing wave gauges and probes to the simulation.
- setWaveParameters: pre-processing tool used to read the parameters inside waveProperties.input. It converts the water depth and wave period information into a wave number.
- setWaveField: Utility used to set the initial conditions from a user defined wave theory.
- waveFoam: solver to the study of wave propagation and interaction. Impermeable structures may be present, but not for this case.

These solvers take the information from dict files. For instance, in order to run blockMesh, the blockMeshDict should be available. In the appendix A, the different dicts needed for running adequately each one of the above solvers will be explained.

After the first 2D model simulation finishes, the results will be analyzed. For that, multiple methods will be used to ensure that the simulation is going properly. The aspects that will be looked at are the following ones:

- Wave height analysis: The wave height will be given for the different locations of the flume. With this, we will see if the wave height follows the classic pattern when approaching a shoal. The wave height decreases in the beginning and then it increases until it breaks. In the following picture, it is shown how the wave should behave when approaching a shoal:

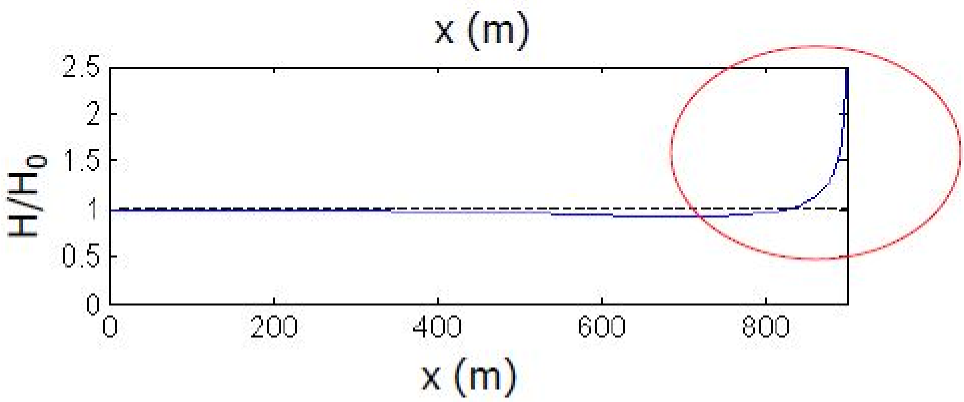

Figure 5.1: Theoretical variation of wave height when approaching a shoal. Source: CIE4325 - Ocean Waves (TU Delft), lecture notes (2019)

- Analysis of breaking point: The point at which waves should break is calculated using the simple rule of $H_{breaking} = 0.8 \times h$. Where h is depth.
- Comparison of the data readings: The gauges' readings from the physical flume and the computational flume will be plotted one against other to see if differences arise. This is a visual comparison.

Chapter 6

Computational simulations

The first simulations were performed to adjust the model. Mainly the goal was to adjust parameters such as:

- Grid size: The grid geometry must be the as close as possible to a square (2D simulations) or a cube (3D simulations). Apart from that, the resolution has to be small enough to avoid the wave damping over the domain and large enough to avoid an excessive running time. See equations 5.1, 5.2 and 5.3 for initial estimates for grid size.
- Time step: Increment of time over which the equations of the model are solved. (Fixed at 0.0025 s for both 2D and 3D simulations)
- Courant number: Defined as the ratio between the time step and the distance between adjacent cell elements. In short, the distance that any information (wave speed, pressures, etc.) travels over the time step, has to be lower than the distance between mesh elements. (Courant number fixed at 0.4 for 2D simulations and 0.5 for 3D simulations).

Once the above is known, the solvers described in appendix A are runned.

After the simulations were completed. A recurrent error appeared: Initial velocities at the interface between snappyHexMesh and blockMesh domains. In the next figure, the domain of snappyHexMesh is shown (area around shoal and wave propagation zone).

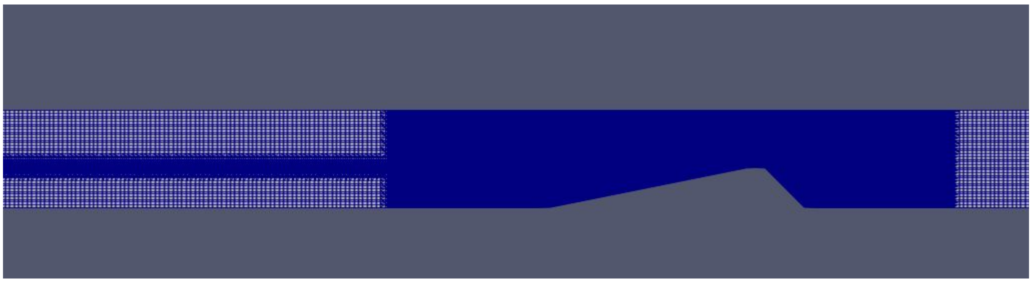

Figure 6.1: snappyHexMesh high definition area

As mentioned above, at the interface, the following initial velocities appear

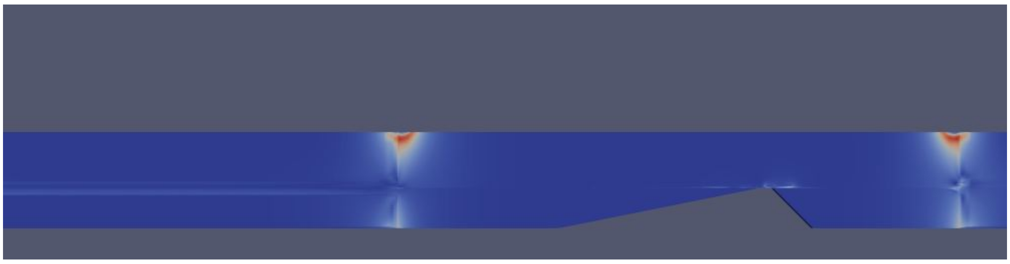

Figure 6.2: Velocity at t=0 for the above snappyHexMesh high definition area (see figure 6.1)

What causes those disturbances is the abrupt change of cell size between the blockMesh domain and the snappyHexMesh domain. However, multiple geometries, grid sizes and amount of transition cells from one domain to another have been used unsuccessfully. The initial velocities were always present. Those initial velocities would then interfere in the wave propagation, giving quite unrealistic results.

In order to palliate this issue, the snappyHexMesh definition zone was extended to the whole domain of the mesh. This would dramatically increase the running time, but it ensured that the initial velocities were minimum and that there are not disturbances at the interface blockMesh-snappyHexMesh, since there is not interface. In the following image, the initial velocities for the final 2D simulation are shown. For the 3D simulation these disturbances still present but they'll be considered minimum.

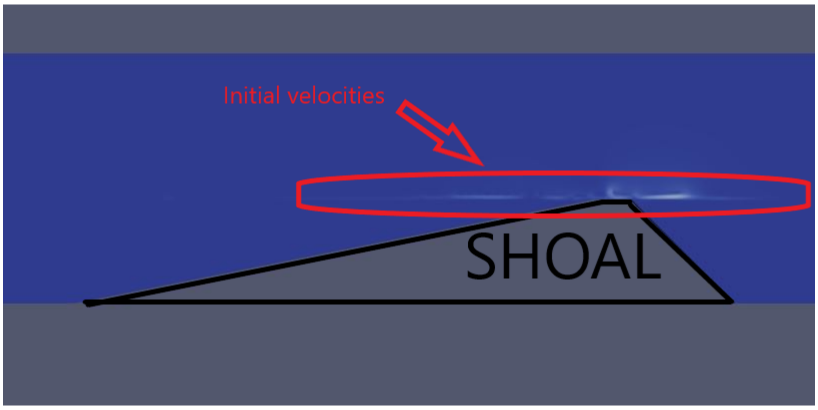

Figure 6.3: Initial velocities for final 2D simulation

Since the initial velocities are present in the air domain (and not within the water domain) and they are relatively small, it is considered that those initial velocities won't really affect the simulation.

6.1 Simulation A: wave height of 0.10 m and wave period of 1.205 s

From all the simulations, this is the one with smallest wave height and wave length. This is a negative aspect when looking at grid sizes, since more cells are needed to adequately represent reality. However, from a precision point of view, the smallest simulation in the physical wave flume is the one which can have the largest accuracy in the gauge readings.

In the appendix B a representation frame by frame of the wave reaching the shoal and breaking is shown.

Let's now take a look to the wave height at each location. For this, the wave height has been calculated using a zero-up crossing method. Wave height has been calculated for the gauges situated at the zone before the shoal. See figure below:

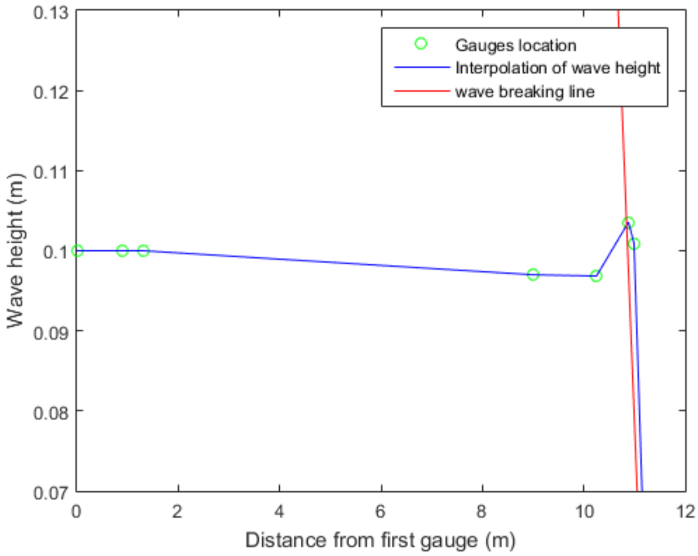

Figure 6.4: Wave height development

In the above figure we can observe how the wave height is behaving as expected: wave height decreases and then increases until it breaks. The breaking point line has been calculated using the simple rule of: $H_{breaking} = 0.8 \times h$ where h is the water depth. As we can see, the waves break just after that point. This is considered to be acceptable since the breaking point calculated with the simple rule is just an approximation.

Now, the physical gauge data is compared to the computational flume gauge data. In the following figures, the result of comparisons is shown:

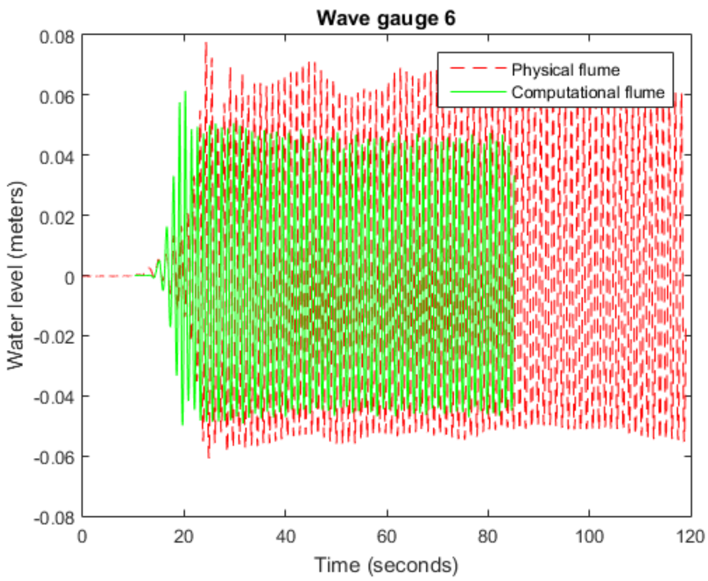

Figure 6.5: Gauge 6 data comparison

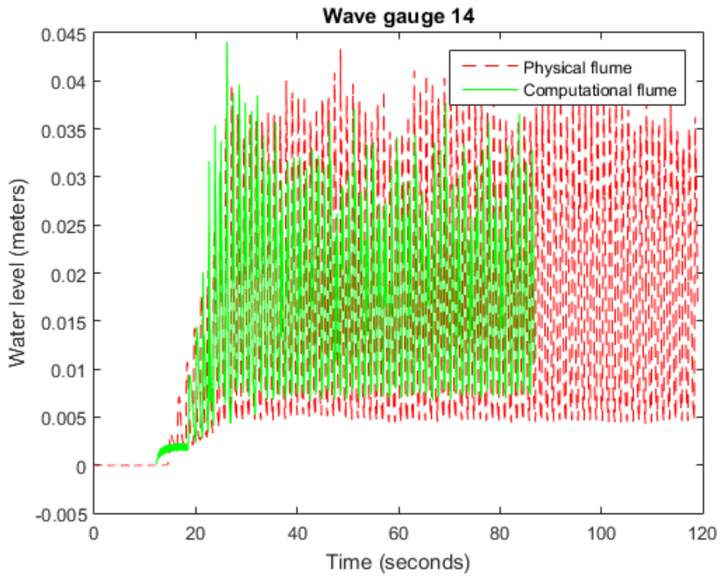

Figure 6.6: Gauge 14 data comparison

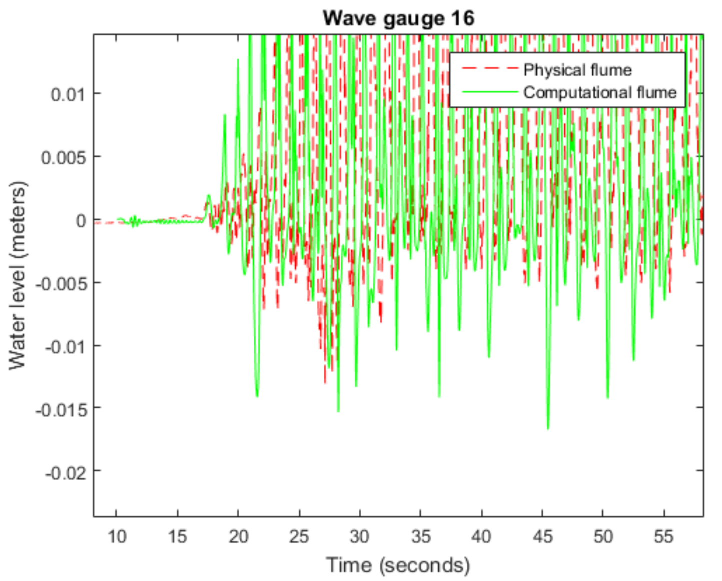

Figure 6.7: Gauge 16 data comparison

When looking at the above figures, we have to keep in mind that the data from the physical data has been obtained for a 3D set-up. This means that the gauges 14 and 16 data will be affected by the lighthouse. In the computational data (for the 2D simulations), the waves reaching those gauges are only influenced by the shoal.

In general, from observing the above figures, it can be said that the wave length (and period) matches for both the physical and computational data, but the water surface doesn't. For the gauges 14 and 16, it could mean that the lighthouse is affecting the wave height by taking energy from the waves. However, for the gauge 6, this cannot be the case. Personally, I find the wave height from the physical data a bit large for what it was supposed to be: wave amplitude of 0.05 meters. The wave amplitude goes up to 0.06 meters for the physical data, which is not very accurate. In this zone, the waves propagate without any disturbance from the bottom and therefore, the wave height has to be equal to the generated wave height of 0.10 meters. To check if this would happen when running larger wave heights and periods, another analyses have been done for larger wave heights (see section 6.2).

Anyways, as it was mentioned in the methodology, this first simulations are just to adjust the parameters to then pass to the 3D simulation. So the results have been considered adequate for that purpose.

Additionally, another 2D simulation is done to see how different wave heights and lengths can affect the model output.

6.2 Simulation B: wave height of 0.15 m and wave period of 2.439 s

In contrast to the simulation A, this simulation models the larger wave heights and periods (lengths) of all the physical tests done. This larger dimensions allow the grid size to be larger and therefore the running time decreases.
In the appendix C a representation frame by frame of the wave reaching the shoal and breaking is shown.

As previously done, lets take a look to the wave height at each location:

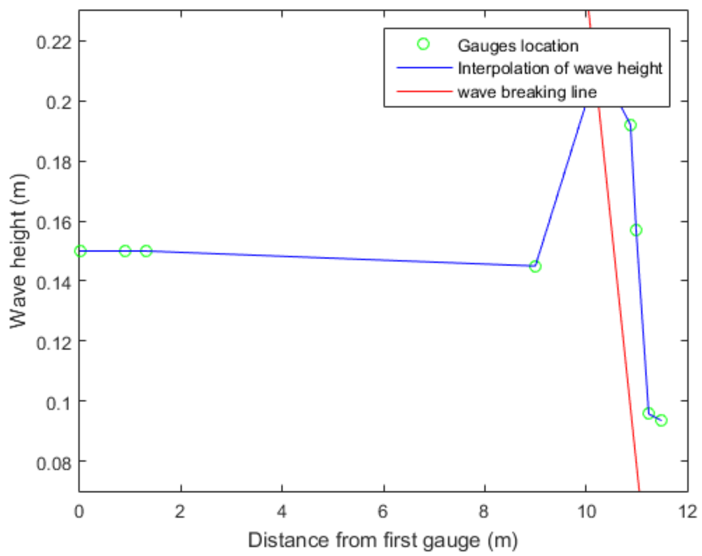

Figure 6.8: Wave height development for simulation B

In the above figure, as so as for simulation A, the wave height behaves as mentioned in the methodology (see figure 5.1).

The next step for verification is to compare the data of the gauges from the physical model to the one from the computational model:

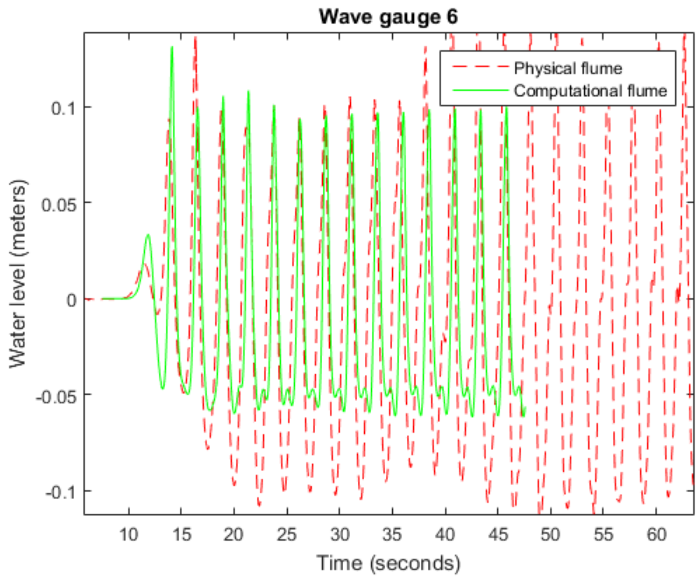

Figure 6.9: Gauge 6 data comparison

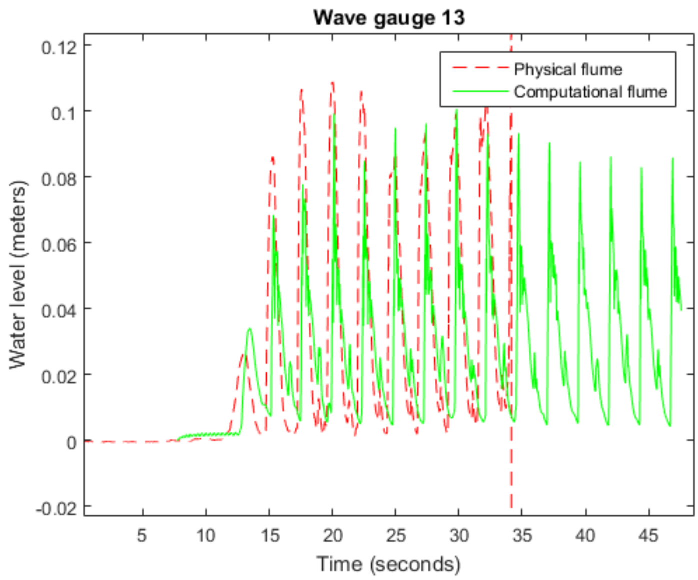

Figure 6.10: Gauge 13 data comparison

In this simulation B, specially for gauge 6 looks like the crests of the waves are well represented. However, there is a clear difference at the wave through. At this position the waves should not be affected by the bottom. It looks like the through this time is affected by the initial velocities. At appendix C, the frames (specially the first) show a strange wave shape.

6.3 Simulation C: 3D simulations

Two 3D simulations have been run successfully. The first one with a wave period of 2.439 seconds and wave height of 0.15 meters. The second one with a wave period of 1.205 seconds and a wave height of 0.126 meters.

The first simulation was able to run for enough time to get 8 seconds of simulation. The second, having less resolution (larger cell size) around the cylinder was able to run for 13 seconds (due to both time limits in the cluster and the excessive amount of cells the 3D simulation has). This small running time don't allow to obtain an adequate analysis about how wave height develops over the flume. However, we can compare the wave gauges from the computational and physical flume. In the following figures, the results are shown:

6.3.1 Wave height of 0.15 and wave period of 2.439 seconds

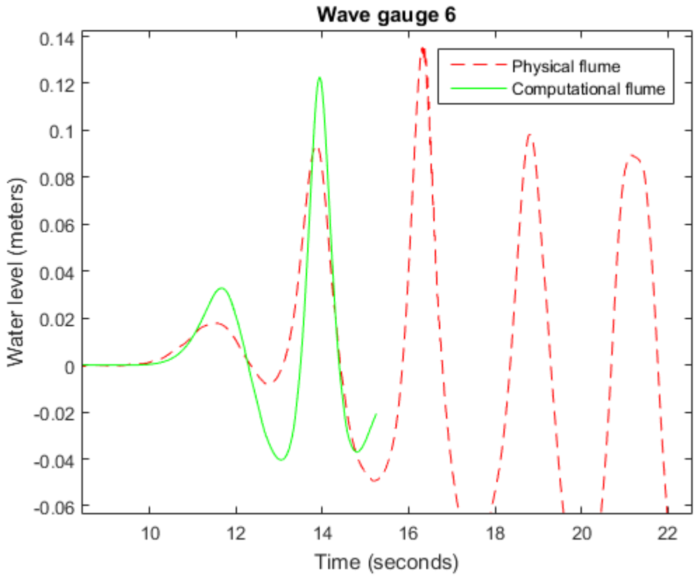

Figure 6.11: Gauge 6 data comparison

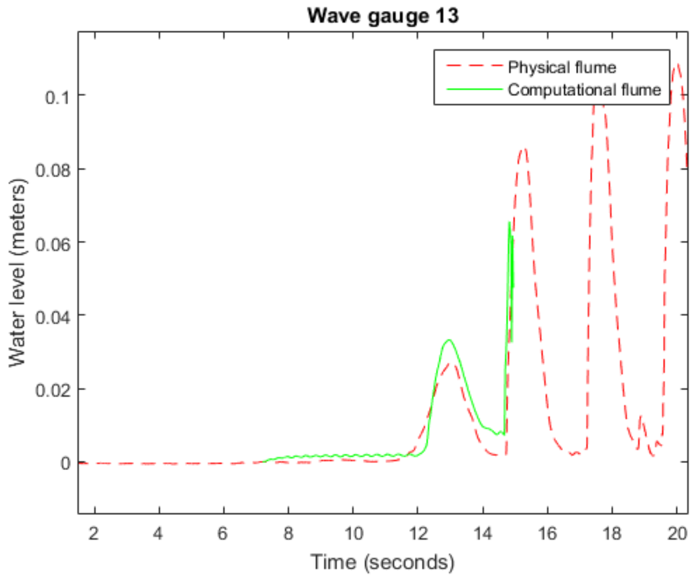

Figure 6.12: Gauge 13 data comparison

6.3.2 Wave height of 0.126 and wave period of 1.205 seconds

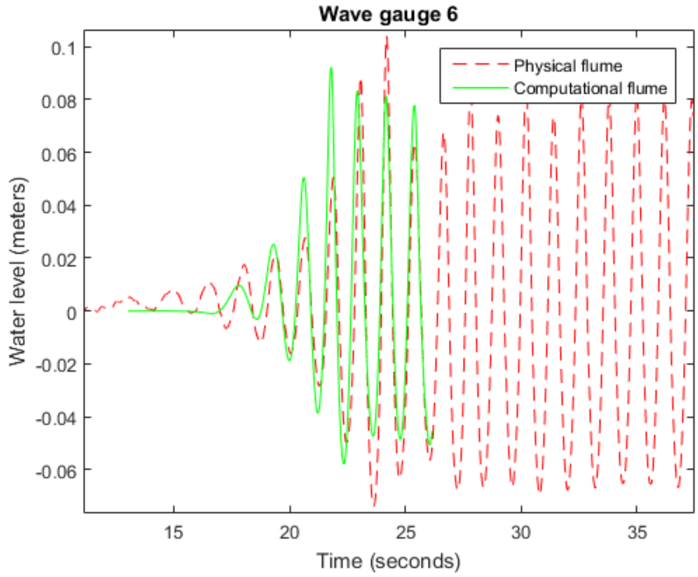

Figure 6.13: Gauge 6 data comparison

6.3.3 Comments on the 3D simulations

It is hard to compare data when the computational models has only run for 8 and 13 seconds. However, in the graphs, it can be seen that the 3D simulations could resemble reality. Anyways, more time is needed to adequately compare. Specially because in the computational model looks like the waves develop faster than for the physical flume (for both 2D and 3D simulation), so even if the first generated waves don't match the readings from the physical data, it doesn't mean that the rest will be wrong.

In short, looking at the gauges output the 3D simulation set-up looks similar to the physical flume. Anyways, longer simulation times are required to confirm this thought.

Chapter 7

Conclusion and future recommendations

This research additional thesis sets the basis for the development of a computational flume able to represent a physical flume in which waves reach a lighthouse upon a shoal. Still some work needs to be done. Specially to pass to the 3D simulation. As it will be mentioned below, it is advised to redo the mesh, having small cells at the wave propagation area and around the shoal and the cylinder and larger cells outside of this domain. Like at figure 6.1.

In the following lines, some recommendations will be given since the initial goal of the thesis hasn't been reached.

The first recommendation, following the already performed work, would be to run the simulation for a longer time. Currently, taking into account cluster (time use) limitations at the TU Delft's cluster and the amount of cells for the current 3D simulation, the simulation can not finish running completely. Therefore, this first recommendation would be to let run the 3D simulation, since both the 2D analysis and the little of the 3D simulations that were performed indicate that the run can be successful and it could adequately represent the physical flume. This being said, the simulation would be very time-consuming. It is estimated that for running a simulation of 120 seconds, which is the duration of the physical flume data, it would take approximately 70 days. This is not efficient at all. Actually it is pretty bad running time for just some waves reaching a cylinder over a shoal. For that reason, a simulation of less time (simulation time, reduce the 120 seconds to 40) can be run to see if the 3D model actually resembles the physical data or not.

In order to make the simulation more time-efficient, the amount of cells have to be reduced. For that reason it is recommended to mesh the problem using third-parties software such as Solomon or ICEM. After meshing using this other method, the simulation can be then started again to check if the initial velocities are still present. it is quite likely that they disappear since something seems to be wrong with the mesh cut done by snappyHexMesh. However, the exact problem has not been found. The output of checkMesh didn't give any strange cell.

References

- Antonini, A.; Rabi, A. and Dassanayake D.T. (2019, June 12th). Efficacy of Analysis Techniques in Assessing Broken Wave Loading on a Cylinder Upon a Shoal. [PowerPoint presentation]
- Jacobsen, N. (2017). waves2Foam Manual. Deltares, The Netherlands.
- Holthuijsen, L. H. (2010). Waves in oceanic and coastal waters. Cambridge university press.
- Pappas, A., D'Ayala, D., Antonini, A., & Raby, A. (2019). Finite element modelling and limit analysis of Fastnet lighthouse under impulsive ocean waves. In Structural Analysis of Historical Constructions (pp. 881-890). Springer, Cham.
- Chenari, B. (2014). Numerical modelling of regular wave propagation using OpenFOAM (Master's thesis).
- Caputo, P. (2018). Hydrodynamic Loading and Structural Dynamic Assessment of an Offshore Lighthouse (Master's thesis).
- Raby, A., Bullock, G. N., Banfi, D., Rafiq, Y., & Cali, F. (2015, November). Wave loading on rock lighthouses. In Proceedings of the Institution of Civil Engineers-Maritime Engineering (Vol. 169, No. 1, pp. 15-28). Thomas Telford Ltd.

Appendices

Appendix A

Solvers used

A.1 blockMesh

The first step when running a computational simulation of this kind is to create an adequate mesh. For this project, as the geometry of the flume is quite simple, just blockMesh and snappyHexMesh (see section A.2) will be used to define it.

In order to run blockMesh, blockMeshDict have to be present in the directory [caseName]/constant/polymesh/. In this file, the main geometry of the flume is defined. As snappyHexMesh (see section A.2) will be used later on, in blockMeshDict, only the paralepipid shape of the flume has to be defined. Besides, in order to reduce the computational time, the flume will be shortened until the point where the first gauge of the physical flume is. Therefore, the computational flume will have the following dimensions: 17.17x0.6x1.2 m.

A.2 snappyHexMesh

After a mesh is already created, snappyHexMesh can be used to create more complicated meshes than the ones we could achieve by only using blockMesh. this solver needs the dict snappyHexMesh located at [case]/system/. snappyHexMesh uses triangulated surface geometries which overlap an already existing mesh. Then the user can decide if the tool creates a mesh in the exterior or interior of the surface. This can be seen in the following figure:

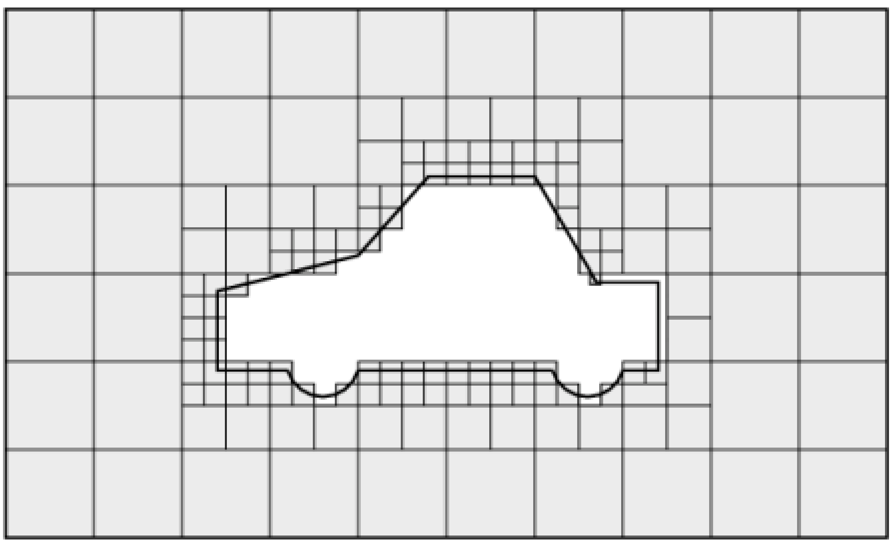

Figure A.1: Cell removal in snappyHexMesh meshing process. Source: CFD Direct, OpenFOAM v6 User Guide: 5.4 Mesh generation with snappyHexMesh (2019)

For the present analysis, blockMesh was used to define a paralepipid mesh which then will be trimmed using snappyHexMesh. For trimming the mesh, a surface representing the shoal (see figure A.2) has been done using FreeCAD. This surface was converted to .stl format so that snappyHexMesh could work with it. For the 3D case, another surface with the shoal and the lighthouse will be used (see figure A.3).

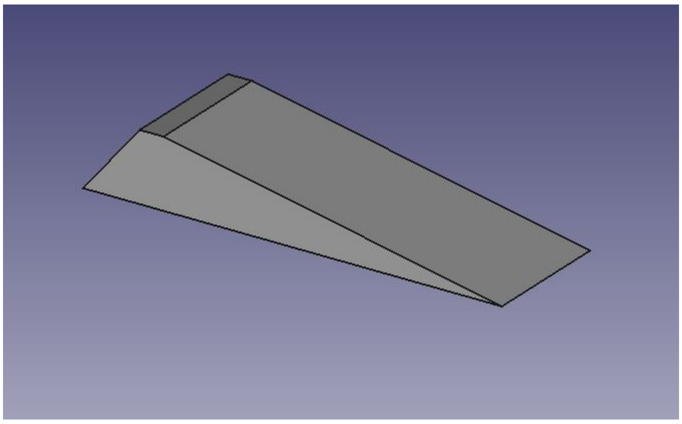

Figure A.2: Shoal for 2D simulation

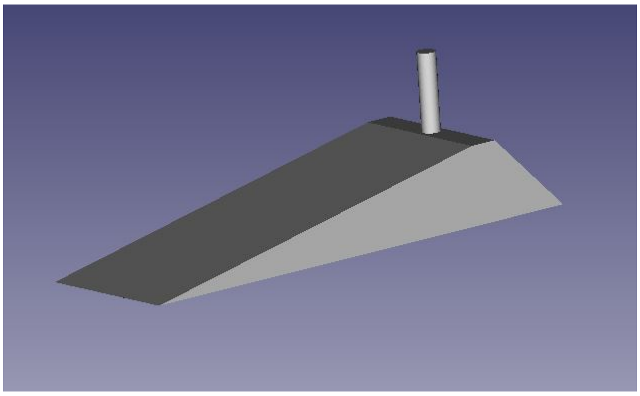

Figure A.3: Shoal and lighthouse for 3D simulation

After running snappyHexMesh, the final result of the mesh is the following one:

Figure A.4: Final mesh for 2D analysis

As in the above picture, the cell-size can be hardly appreciated, a detailed image is shown below:

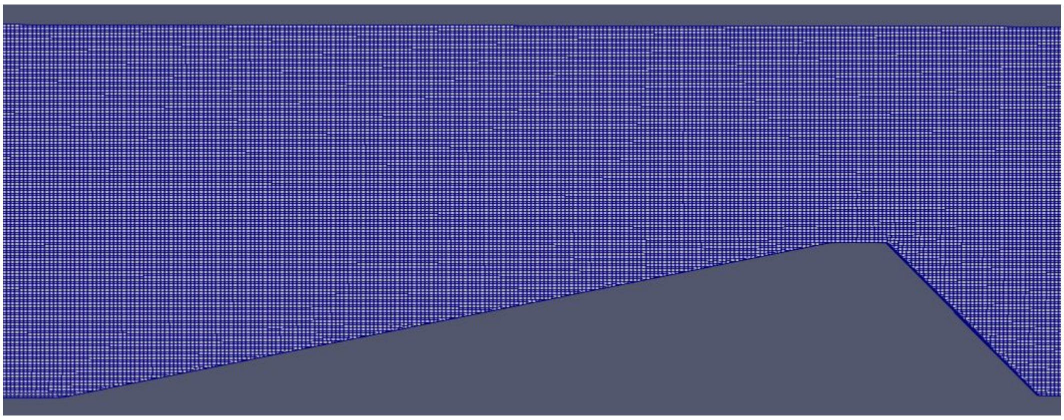

Figure A.5: Caption

The side-view of the 3D mesh is the same as the one above, but if we take a look to the shoal part we can see the cylinder's geometry:

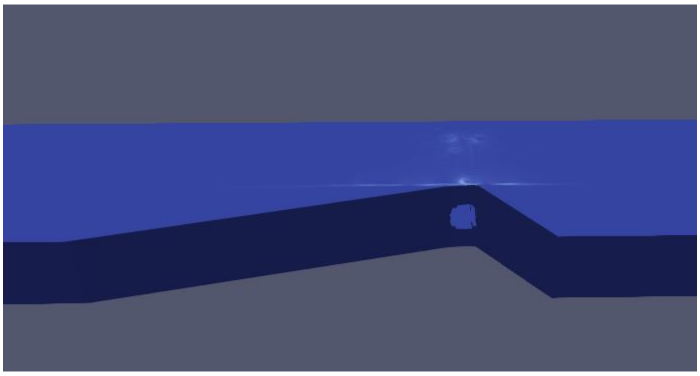

Figure A.6: Detail of the lighthouse meshing for 3D analysis

A.3 waveGaugesNProbes

Gauges are needed to get information about pressures, velocities and water surface elevation. In order to define the probes position, probesDefinition dict is needed at [case]/constant. This dict contains the information about the probe's geometry and the physical property (or properties) that we want to analyze. In the following figure, the probes' position is shown:

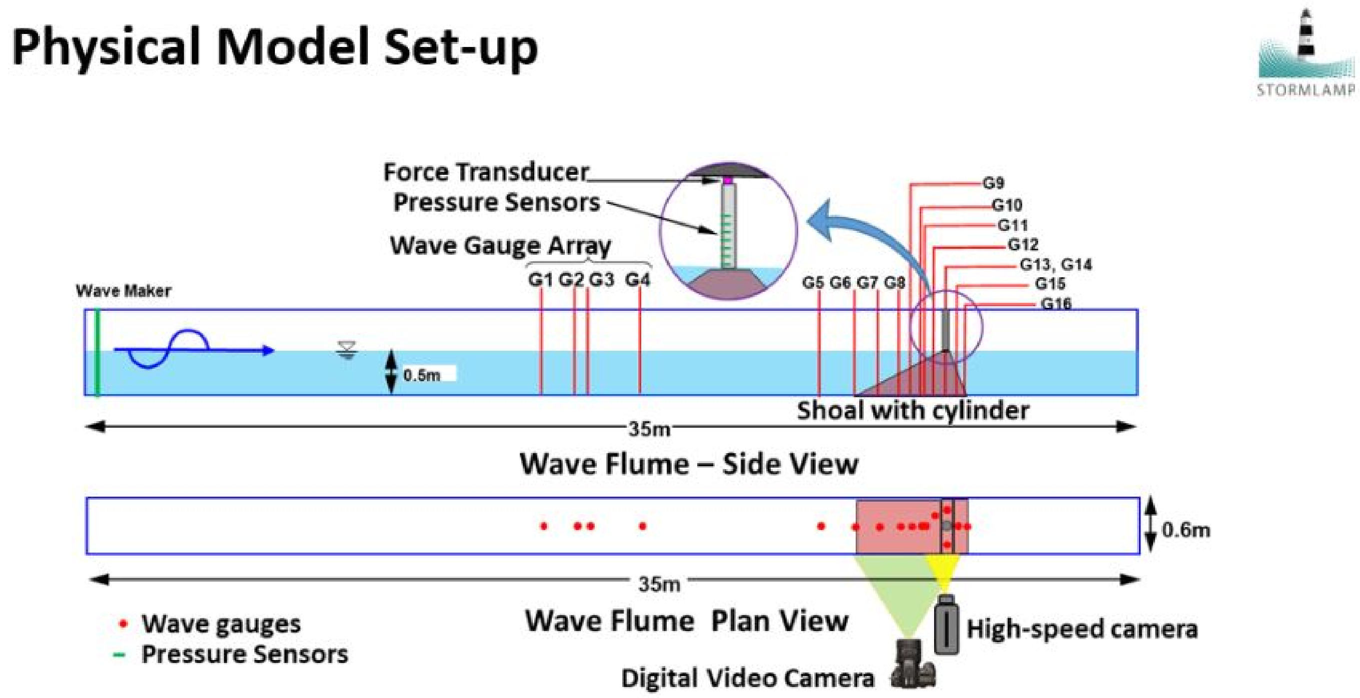

Figure A.7: Wave gauges set-up. Source: Antonini, A.; Rabi, A. and Dassanayake D.T. (2019, June 12th) [PowerPoint presentation]

A.4 setWaveParameters

Once the mesh and probes are ready, the solvers for setting up the wave environment can be used. First of all, the dict waveProperties.input need to be included at [case]/constant. Here the regular wave parameters are set.

Then the setWaveParamenters application reads the waveProperties.input dict and the derivated wave parameters are written into the new created file waveProperties.

A.5 setWaveField

The setWaveField application uses an input wave theory specified by the user in waveProperties.input dict and takes as initial state the U, pd and alpha files within the [case]/0 folder. With this input parameters the application loops over every cell in the computational domain and sets the VOF-ratio an velocity field. The VOF-ratio is a Volume Of Fluid technique, commonly used in computational CFD to track and locate the free surface , or in other words, fluid-fluid interface.

A.6 waveFoam

Takes all the above information and process it to generate and absorb the modelled waves.

Appendix B

Simulation A

Frames for wave breaking:

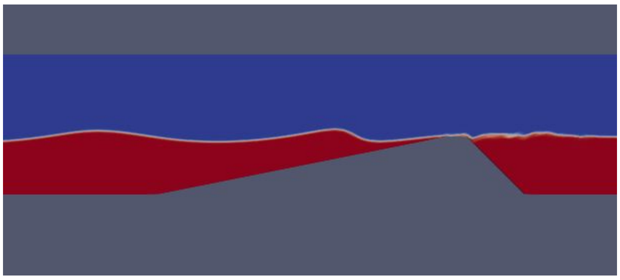

Figure B.1: First frame, simulation A

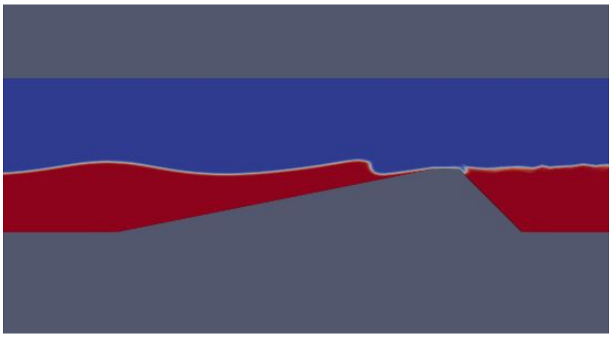

Figure B.2: Second frame, simulation A

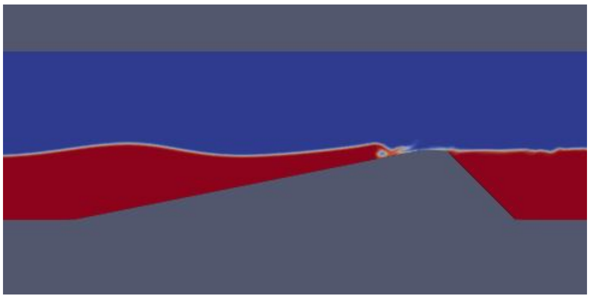

Figure B.3: Third frame, simulation A

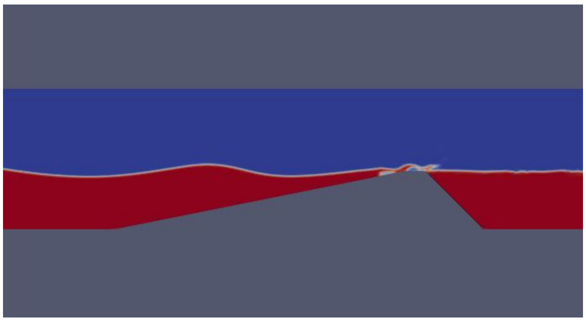

Figure B.4: Fourth frame, simulation A

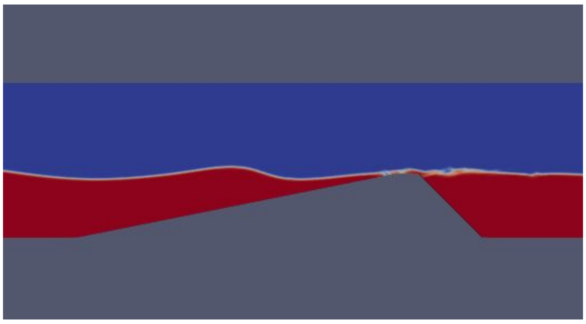

Figure B.5: Fifth frame, simulation A

Appendix C

Simulation B

Frames for wave breaking:

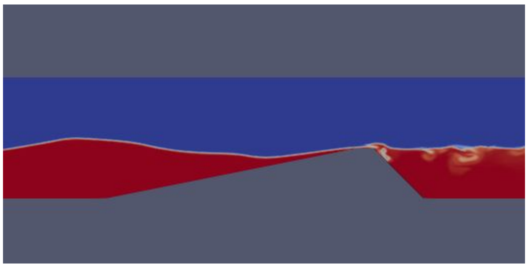

Figure C.1: First frame, simulation B

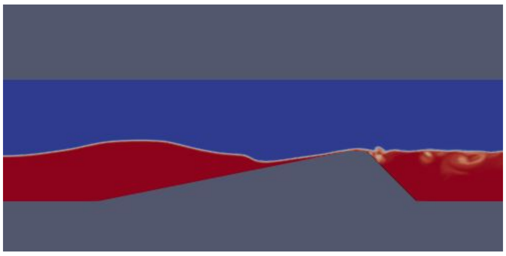

Figure C.2: Second frame, simulation B

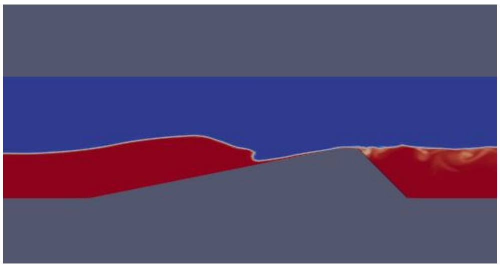

Figure C.3: Third frame, simulation B

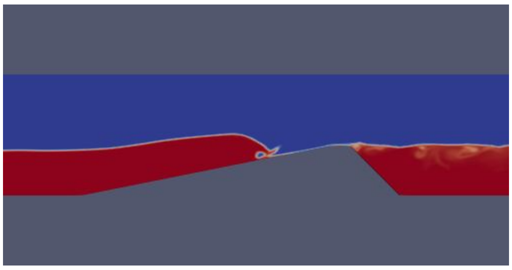

Figure C.4: Fourth frame, simulation B

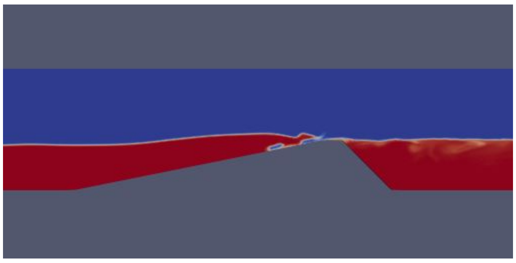

Figure C.5: Fifth frame, simulation B

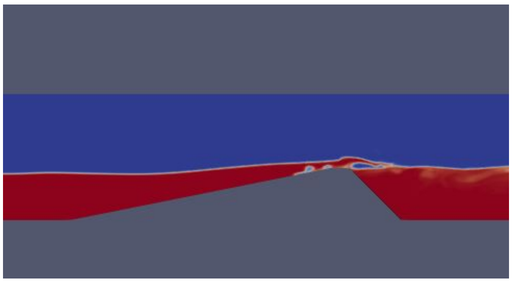

Figure C.6: sixth frame, simulation B

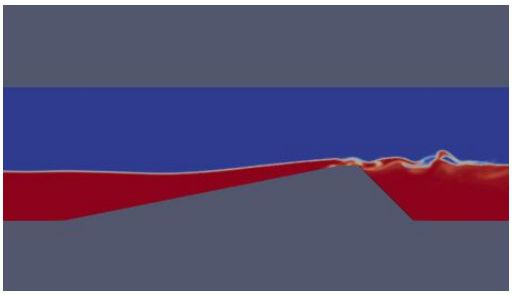

Figure C.7: seventh frame, simulation B