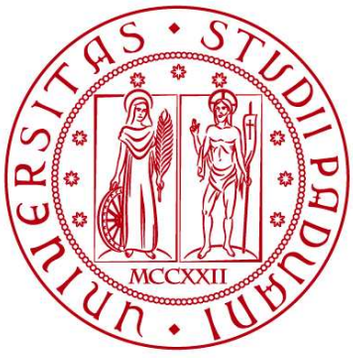
UNIVERSITÀ DEGLI STUDI DI PADOVA

DIPARTIMENTO DI INGEGNERIA CIVILE, EDILE E AMBIENTALE

Department Of Civil, Environmental, and Architectural Engineering

Corso di Laurea Magistrale in Ingegneria Civile Idraulica



TESI DI LAUREA

QUANTIFYING WAVE DISSIPATION BY POROUS WAVESCREENS WITH HIGH RESOLUTION 3D NUMERICAL MODELING

Relatore:

Ch.mo Prof. PIERO RUOL

Correlatori:

Ch.mo Prof. LUCA MARTINELLI

Ch.mo Prof. ALBERTO CANESTRELLI

Ph.D. EDWIN RAJEEV

Laureando: STEFANO BIONDI

Matricola: 1237527

ANNO ACCADEMICO 2022-2023

“Big whirls have little whirls that feed ther velocity,

and little whirls have lesser whirls and so on to viscosity”

- Lewis Fry Richardson

Abstract

In this thesis, an OpenFOAM numerical model is developed to assess the wave dissipation, induced forces and reflection on slotted porous wave screens. A model validation is performed with a physical model comparison. 108 different wave screen configurations were defined by considering: two thickness values (2.5 inches, 3.5 inches), three porosity values (10%, 20%, and 30%), two values for the distribution of the pores (represented by 3 or 6 slits with fixed porosity), three values of submergence (3, 4, and 5 times the wave height) and three slits orientations (horizontal, 45° inclined and vertical). In the analysis, A second order stokes wave was generated with fixed height and period in transition water conditions. The main found was that the slotted wave screens with increased submergence and low porosity provide more wave dissipation. In general, the wave induced force increase with increasing submergence and low porosity, while the number of slits and orientation does not affect significantly neither dissipation, reflection or forces. Finally, we found the change in wave screen thickness, between standard slit sizes values, has a negligible impact on wave dissipation. These results can be used to design preliminarly an optimal configurations of slotted wave screens. Moreover, a comparison with the analytical Borda-

Carnot theory was made. The results of the numerical model and the theory showed good agreement.

Riassunto

In questa tesi è stato sviluppato un modello numerico OpenFOAM per valutare dissipazione, riflessione dell'energia ondosa e forze agenti su frangiflutti fissi porosi. Il modello è stato validato comparando i risultati numerici con quelli derivanti da modelli fisici. Vengono definite 108 diverse combinazioni di caratteristiche geometriche delle strutture: due valori di spessore (2.5 e 3.5 pollici), tre valori di porosità (10%, 20%, and 30%), due valori per la distribuzione delle forature (3 fori e 6 fori a parità di porosità), tre valori di sommergenza (3, 4 e 5 volte l'altezza d'onda incidente) e tre valori per l'inclinazione dei fori rispetto all'asse verticale (verticali, inclinati di 45° e orizzontali). È stata generata un' onda che segue la seconda teoria di Stokes di altezza e periodo fissati in condizione di acque di transizione. Il risultato principale che viene presentato è la maggior dissipazione fornita da strutture con bassa porosità e alta sommergenza, dove la porosità è il parametro dominante. In generale, sia la forza indotta dalle onde che la riflessione sono maggiori con l'aumento della sommergenza e diminuzione della porosità. In particolare, inclinazione, distribuzione dei fori e spessore non influiscono significativamente su dissipazione, riflessione o forze. Questi risultati possono essere usati preliminarmente per la progettazione di

una configurazione ottimale di un frangiflutti poroso per la dissipazione dell'energia ondosa. Inoltre, i risultati del modello sono stati confrontati con la teoria analitica di Borda-Carnot, mostrando un buon accordo.

Keywords: *OpenFOAM, wave dissipation, numerical modeling, slotted wavescreen, wave induced forces, porous structur*

Contents

Introduction

Due to extreme weather and sea-level rise, people living in harbor areas and marine ecosystems are at greater risk of damage from coastal hazards. Various types of breakwaters are utilized in coastal and harbor regions. The porous wavescreens have recently been taken into account as an alternative tool to control wave heights to an acceptable level in order to solve various problems associated with gravity-type and floating breakwaters. To build the gravity type ones, rocks have been widely used as a construction material. Although rocks structures are reliable, they are very expensive and difficult to built and maintain. For example, for a location with mean water depth of 3.0 m and less tidal variation, slope of 1:2 and freeboard and crest width of 3 m, a typical offshore rubble mound breakwater needs about 90 m^3 of rocks/m (Nicholson et al., 1997). Additionally, emerged rubble mound breakwaters restrict free exchange of sea water between seaside and rear side. This lead to water quality problems issues on the coastal waters. Quarrying stones from mountains and underground sources has become a sensitive environmental issue recently. On the other hand, floating breakwaters are not effective with with long waves. With all these sensitive issues, it is essential to innovate cost competitive, easy to build and environmentally friendly wave damping structures. Porous wavescreens, like

series of slotted vertical walls are one such solution (Neelamani and Al-Anjari, 2018). In this work a numerical model is developed to study how the geometrical features (porosity, slits orientation, Submergence, thickness and distribution of slits) of slotted wave screens affect both dissipation and wave forces. This lead to a preliminary optimal configuration of the wavescreen.

Chapter 1: Numerical model

1.1 Numerical model description

1.1.1 *OpenFOAM*

OpenFOAM is first and foremost a C++ library, used primarily to create executables, known as applications. The applications fall into two categories:

- *Solvers*, that are each designed to solve a specific problem in continuum mechanics;
- *Utilities*, that are designed to perform tasks that involve data manipulation.

New solvers and utilities can be created by its users with some pre-requisite knowledge of the underlying method, physics and programming techniques involved.

OpenFOAM is supplied with pre and post-processing environments. The interface to the pre and post-processing are themselves OpenFOAM utilities, thereby ensuring consistent data handling across all environments. The overall structure of OpenFOAM is shown in *Figure 1.1*

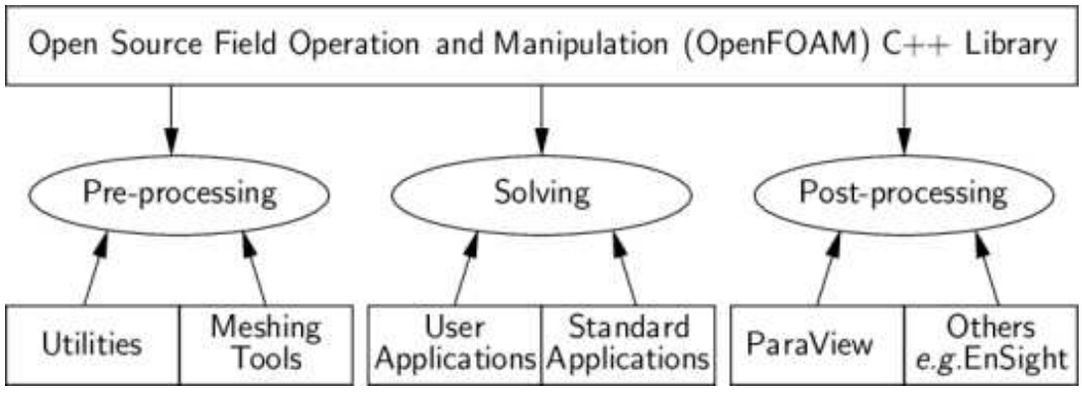


Figure 1.1: Overview of OpenFOAM structure.

1.1.2 Mesh generation

By default OpenFOAM defines a mesh of arbitrary polyhedral cells in 3-D, bounded by arbitrary polygonal faces. This type of mesh, called a polyMesh, allows for a wide range of mesh generation and manipulation options, especially when the domain geometry is complex or changes over time. However, converting meshes generated with other tools can be challenging. To address this, OpenFOAM includes cellShape tools to handle conventional mesh formats based on pre-defined cell shapes.

1.1.2.1 blockMesh

The principle behind *blockMesh* is to decompose the domain geometry into a set of 1 or more three dimensional, hexahedral blocks. Edges of the blocks can be straight lines, arcs or splines. The mesh is ostensibly specified as a

number of cells in each direction of the block, sufficient information for *blockMesh* to generate the mesh data.

Each block of the geometry is defined by 8 vertices, one at each corner of a hexahedron. The vertices are written in a list so that each vertex can be accessed using its label, remembering that OpenFOAM always uses the C++ convention that the first element of the list has label '0'. An example block is shown in *Figure 1.2* with each vertex numbered according to the list. The edge connecting vertices 1 and 5 is curved to remind the reader that curved edges can be specified in *blockMesh*.

It is possible to generate blocks with less than 8 vertices by collapsing one or more pairs of vertices on top of each other.

The local coordinate system is defined by the order in which the vertices are presented in the block definition according to:

- the axis origin is the first entry in the block definition, vertex 0 in our example;
- the x_1 direction is described moving from vertex 0 to vertex 1;
- the x_2 direction is described moving from vertex 1 to vertex 2;
- vertices 0, 1, 2, 3 define the plane $x_3 = 0$;
- vertex 4 is found by moving from vertex 0 in the x_3 direction;

- vertices 5, 6 and 7 are similarly found by moving in the x_3 direction from vertices 1,2 and 3 respectively.

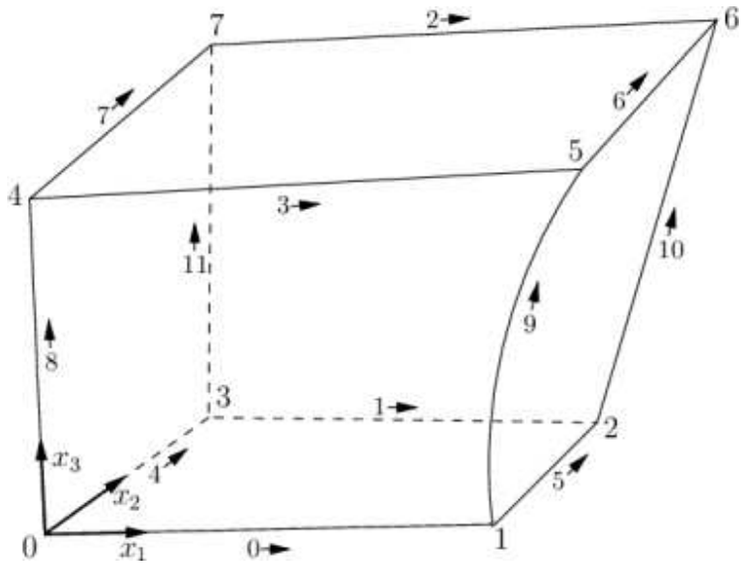


Figure 1.2: Computational order for the blocks

1.1.2.2 snappyHexMesh

The snappyHexMesh utility generates 3-dimensional meshes containing hexahedra (hex) and split-hexahedra (split-hex) automatically from triangulated surface geometries in Stereolithography (STL) format. The mesh approximately conforms to the surface by iteratively refining a starting mesh and morphing the resulting split-hex mesh to the surface. An optional phase will shrink back the resulting mesh and insert cell layers. The specification of mesh refinement level is very flexible and the surface handling is robust with a pre-specified final mesh quality. It runs in parallel with a load balancing step every iteration.

In this work, The *refinement Box* feature will be used. This feature consists of a selection of coordinates to define the domains' areas where a higher refinement is needed.

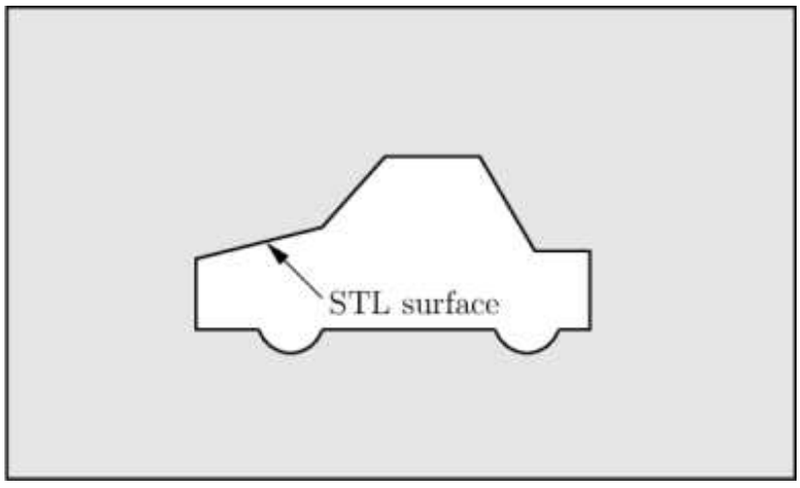


Figure 1.3: Schematic 2D meshing problem for snappyHexMesh

1.1.3 Solving

Most fluid dynamics solver applications in OpenFOAM use the pressure-implicit split-operator (PISO) or semi-implicit method for pressure-linked equations (SIMPLE) algorithms, or the two of them merged (PIMPLE). These algorithms are iterative procedures for solving equations for velocity and pressure, PISO being used for transient problems and SIMPLE for steady-state.

Both algorithms are based on evaluating some initial solutions and then correcting them. SIMPLE only makes 1 correction whereas PISO requires more than 1, but typically not more than 4.

In this work a Multiphase, incompressible fluid and Transient flow is considered. For this reason, the solver *interfoam* is used.

1.1.3.1 Finite Volume Method

The finite volume method is a numerical method used to solve partial differential equations (PDEs) that describe physical systems. It is a widely used method in computational fluid dynamics (CFD) and other fields, and is particularly well-suited for problems involving conservation laws.In the finite volume method, the domain of interest is divided into a set of discrete control volumes, or cells. The PDE is then discretized by expressing the unknown variables (such as velocity, pressure, or temperature) as averages over the control volumes. The resulting system of algebraic equations is then solved to determine the values of the unknown variables at each control volume.

One of the key advantages of the finite volume method is that it is easy to implement and can be applied to a wide range of problems. It is also well-suited for parallel computing, as the control volumes can be solved independently of each other.

Overall, the finite volume method is a powerful tool for simulating and understanding complex physical systems, and is widely used in engineering and science.

1.1.4 Post-processing

OpenFOAM data can be converted into VTK format. The converted data can be post-processed in *ParaView* or any other program supporting *VTK* format.

ParaView operates a tree-based structure in which data can be filtered from the top-level case module to create sets of sub-modules. For example, a contour plot of pressure could be a sub-module of the case module which contains all the pressure data. The strength of *ParaView* is that the user can create a number of sub-modules and display whichever ones they feel to create the desired image or animation. For example, they may add some solid geometry, mesh and velocity vectors, to a contour plot of pressure, switching any of the items on and off as necessary.

1.2 Governing equations

1.2.1 Navier-Stokes equations

The solver used in this work , *interFoam*, solves the Navier-Stokes equations for two incompressible, isothermal immiscible fluids. This means that the

material properties are constant in the region filled by one of the two fluid except at the interphase.

The constant-density continuity equation is:

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

The momentum equation are:

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

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

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

u represent the velocity, g_i the gravitational acceleration, p the pressure and μ the dynamic viscosity. $f_{\sigma i}$ is the surface tension.

The density ρ is defined as follows:

$$\rho=\alpha\rho_1+(1-\alpha)\rho_2$$

α is 1 inside fluid 1 with the density ρ_1 and 0 inside fluid 2 with the density ρ_2. At the interphase between the two fluids α varies between 0 and 1.

The surface tension $f_{\sigma i}$ is modelled as continuum surface force. it is calculated as follows:

$$f_{\sigma i} + \sigma\kappa\frac{\partial\sigma}{\partial x_i}$$

σ is the surface tension constant and κ the curvature. The curvature can be approximated as follows:

$$\kappa = -\frac{\partial\eta_i}{\partial x_i} = -\frac{\partial}{\partial x_i}\left(\frac{\partial\alpha/\partial x_i}{|\partial\alpha/\partial x_i|}\right)$$

1.2.1.1 Volume of Fluid Method

Volume of fluid (VOF) is a numerical method for tracking and locating free surface which is the interface of air and water in the present study. This method is used by OpenFOAM to specify the fraction of each fluid (air and water) in each cell. The volume (phase) fraction equation is presented below in which α represents the volume (phase) fraction, t represents time and U refers to velocity. α is always between 0 and 1. α= 0 means the cell is fully filled by air and α= 1 means the cell is only filled by water.

$$\frac{\partial\alpha}{\partial t} + \nabla \bullet (\alpha U) = 0$$

The density of each cell can be calculated by the following equation, where ρ_w is the water density and ρ_a is the air density:

$$\rho = \alpha\rho_w + (1 - \alpha)\rho_a$$

Note that this density is the density of the mixture of air and water inside each cell.

1.2.2 Stokes First Order Wave Theory

There are various types of wave equations and theories used in development of numerical wave tank. Among all the theories, Stokes wave theory has been often applied to the studies which investigating behaviour of waves.

Stokes first order wave or *Airy wave theory* refers to a linear wave theory which is used for modelling of gravity waves on the surface of a fluid. Stokes first order wave theory is used in coastal and ocean engineering for simulating behaviour of the waves. This theory is also used for simulating tsunami waves before reaching the coastal area. Airy wave theory has usually been used for estimation of wave characteristics. The results of this linear theory would be accurate not only for shallow water region with a small fraction of wave height and water depth but also for deep water area with a small fraction of wave height and wavelength.

1.2.3 Stokes Second Order Wave Theory

Stokes second order wave theory refers to a non-linear theory which is used for modelling of periodic regular free surface waves. This wave theory is generally used for simulation of the interaction between waves and structures

(both shore-based and offshore). They are applied on the studies in order to specify wave behaviours such as free surface elevation and flow particle velocity. As Stokes theory does not work well for shallow water, it is mostly used for deep water and transition depth areas while for shallow water cnoidal theory provides more accurate estimation.

Shallow water waves are defined as waves with h L < $\frac{1}{20}$ and deep water waves are defined as waves with h L ≥ $\frac{1}{2}$. The Ursell number which is derived from Stokes wave expansion indicates the nonlinearity of long surface gravity waves on a fluid layer. This parameter which has been developed by Ursell (1953) can be used for checking the applicability of using Stokes second order wave theory. As it is important to check the applicability of second order wave theory, present master thesis checked the eligibility of using Stokes second order against the following equation and it was satisfied by all cases of third scenario and sub-scenarios in which Stokes second order theory is used.

$$\frac{L^2 H}{h^3} < \frac{8\pi^3}{3}$$

There are also different orders of Stokes theory such as fifth order. As pointed out in previous section, Stokes second order theory is commonly used in the literature due to complexities of fifth order as well as its well

approximation and accuracy in the results. Stokes second order theory is used in different forms by many studies such as Ohyama et al. (1995); Koo and Kim (2007); Zhan et al. (2010); Li and Lin (2010); Senturk (2011) and Lambert (2012). This master thesis follows the form used by Lambert (2012).

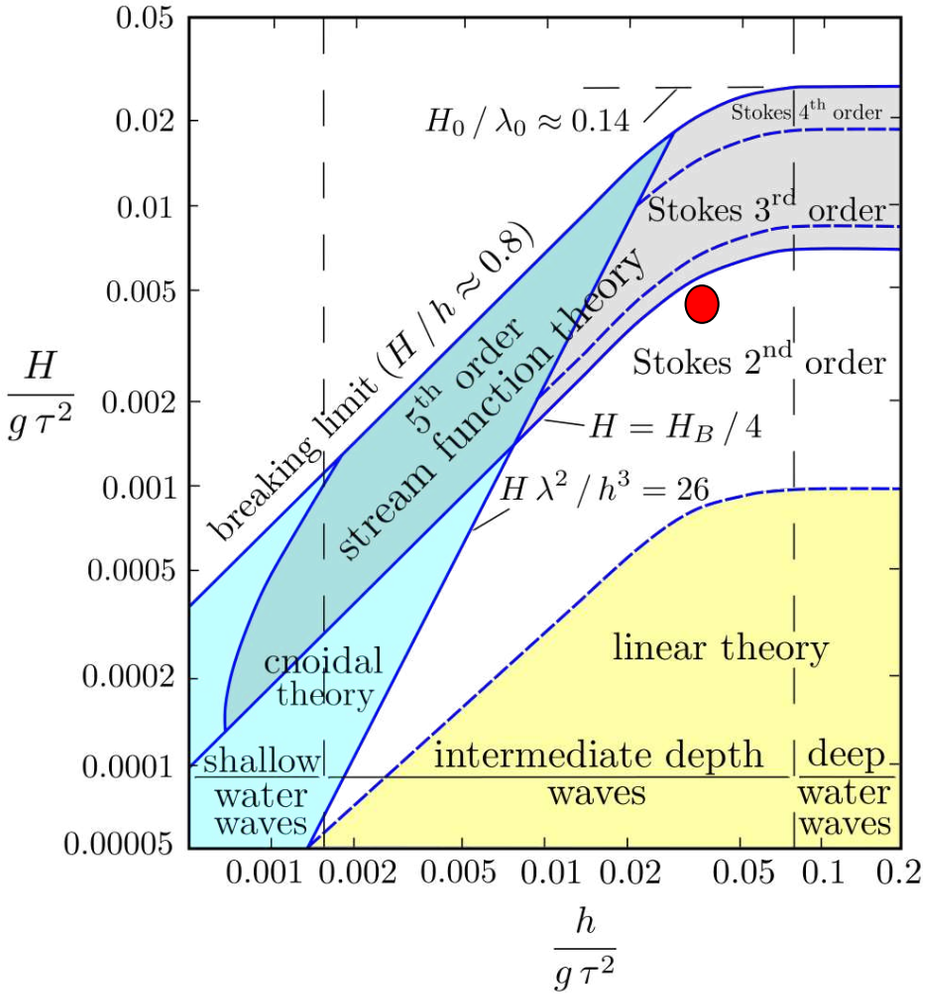


Figure 1.4 Le Mehauté chart for wave theories. The red point represents the combination of depth and wave steepness that define the most suitable wave theory

1.2.3 Surface Elevation Equation

In waves based on Stokes second order theory, the following equation, known as surface elevation equation, shows the displacement of water surface from still water level (SWL). The result of the numerical model developed in present study is validated against the analytical result of surface elevation equation.

$$\eta = \frac{H}{2}\cos(kx - \omega t) + \frac{H^2 k}{16}\frac{\cosh(kh)}{sinh^3(kh)}(2 + \cosh(2kh))\cos 2(kx - \omega t))$$

1.2.4 particle velocity equation

Particle velocity is the velocity of a particle which is transferred by a wave. The particle velocityaccording the Stokes second order theory is given as below:

$$u = \frac{H}{2}\frac{gk}{\omega}\frac{\cosh\ k(h+y)}{\cosh(kh)}\cos(kx - \omega) + \frac{3}{16}\frac{H^2\omega\cosh 2k}{sinh^4 kh}\cos 2(kx - \omega t)$$

$$v = \frac{H}{2}\frac{gk}{\omega}\frac{\sinh\ k(h+y)}{\cosh(kh)}\sin(kx - \omega) + \frac{3}{16}\frac{H^2\omega\sinh 2k}{sinh^4 kh}\sin 2(kx - \omega t)$$

u is the horizontal component of particle velocity and v is the vertical component. Both u and v are partial derivatives of velocity potential. H represents the wave height from peak to trough in [m], g represents acceleration due to gravity in $[\frac{m}{s^2}]$, h represents water depth in [m] while y is the vertical coordinate to describe wave motion (the points in which = 0

makes a line known as still water level). *t* represents time in [s], *x* represents the distance along longitudinal direction in [m], ω represents the frequency of the wave in [Rad.s-1] and *k* represents the wave number in [Rad.m-1].

1.3 waves2Foam

waves2Foam is a toolbox recently developed by OpenFOAM users to simulate free surface wave generation and absorption (Jacobsen et al., 2012). A relaxation zone technique known as active sponge layer has been applied to the library as well as a large range of different wave theories. The base of this toolbox is *interFoam* while an active sponge layer zone defined as relaxation zone method has been added to the solver (OpenFOAMWiki, 2013a). Different types of waves such as current-type waves, regular waves, solitary waves, and irregular waves as well as a combined wave, a combination of the any other types, are defined in this library. Regular waves consist of Stokes first order wave theory (Airy wave), Stokes first order standing wave theory, Stokes second order wave theory, modulated Stokes second order wave theory, Stokes fifth order wave theory, First order cnoidal theory and Stream function wave theory. In this master thesis, Stokes second order wave theory is used.

Many scientists developed their studies using this toolbox. Jacobsen et al. (2012), who are the developers of the toolbox, have implemented a C++ toolbox that has the ability to generate waves (not only wave propagation but also wave breaking) as well as the ability to absorb waves by applying a relaxation zone. In Jacobsen et al. (2012), rectangular and circular ring shape relaxation zones are defined and in is recommended that other shapes of relaxation zone can be easily modelled by modifying the code and programming. The result of the study is validated against the laboratory experiment carried out by Chapalain et al. (1992). Moreover, Ransley et al. (n.d.) published a poster in which they studied the development of open-source CFD software in modelling the interaction of waves and Wave Energy Converters (WECs). They showed that waves2Foam is capable to simulate behaviour of waves interacting with WECs in addition to generate and absorb waves.

Another study carried out by Jensen et al. (2014), in which they conducted a numerical study using waves2Foam to investigate the porous media equations mentioned in different literature references. Finally it was concluded that, OpenFOAM is capable to predict the interaction between waves and structures. Due to the result of their study, they have recently added a new tutorial into their toolbox.

In current study, waves2Foam is used to simulate the interaction between waves and porous structures. The version OpenFOAM v2006 and waves2Foam r-2025 are used in the context of this present master thesis.

1.3.1 Relaxation zone technique

The relaxation zone technique is meant to remove spurious reflection from numerical simulations. The relaxation zone technique is based on a weighting between the computed solution of the velocity field and the indicator field with a target solution. The relaxation zone technique is divided into two flavours, namely explicit and implicit relaxation, where explicit/implicit refers to the time integration. Furthermore, the specification of the weighting (Exponential weight, Free polynomial, Third order polynomial) and location of the relaxation zones in the computational domain are described.

The relaxation could be implemented with a specific shape. The relaxation shape specifies where in the computational domain a certain relaxation zone is applied. There are currently four different relaxation shapes available. These are:

- Rectangular;
- Semi-cylindrical;
- Cylindrical;

In this work a Rectangular relaxation scheme is Implemented. The rectangular shape is the most useful shape, because it works equally well for 2D and 3D simulations. It is defined based on the coordinates of the two diagonal corner points of a rectangle and the direction of one of the lateral sides (and direction of relaxation).

1.3.2 PIMPLE algorithm

The PIMPLE Algorithm is a combination of PISO (Pressure Implicit with Splitting of Operator) and SIMPLE (Semi-Implicit Method for Pressure-Linked Equations). All these algorithms are iterative solvers but, as mentioned in *section 1.1.3*, PISO and PIMPLE are both used for transient cases whereas SIMPLE is used for steady-state cases.

The best way to think about the PIMPLE algorithm is to imagine it as a SIMPLE algorithm for every time step, where outer correctors are the iterations, and once converged will move on to the next time step until the solution is complete. Better stability is obtained from PIMPLE over PISO for this reason, especially when dealing with large time steps where the maximum Courant number may consistently be above 1 or when the nature of the solution is inherently unstable.

Chapter 2: Model generation and validation

In the following chapter is described how the numerical wave flume was set and validated and which were the metods to compute the results.

The main goal of this work was to compute wave energy dissipation and dynamic force one the screen. In order to achieve this goal the following objectives have been pursued:

- Model generation;
- Model validation;
- Grid refinement;
- Wavescreens geometries generation through python scripts;
- Implement a turbulence model;
- Integration of the outputs to compute dissipation and forces;
- Find a theoretical explanation for the phenomena that occur.

2.1 Numerical waveflume

2.1.1 Numerical domain

The wave flume for the numerical simulations was generated using the blockMesh tool in OpenFOAM. Initially, the size of the cells was set to 0.1 m throughout the entire domain. However, the refinementBox feature in the system/snappyHexMeshDict file allows us to define areas of interest where the cell size can be gradually reduced by splitting each cell's side in half multiple times, depending on the desired level of refinement. In this case, the cells were split three times, resulting in a final size of 0.005 m. This process increased the total number of cells to approximately 10^6units.

Two different refinement boxes were used in the simulations. These boxes were placed in the center of the domain (relative to the wave propagation direction from left to right). The centered refinement box covered the entire elevation of the domain (from bottom to top along the Z-axis), allowing to compute precise pressure values. The other refinement box was placed along the free surface to track the wave height.

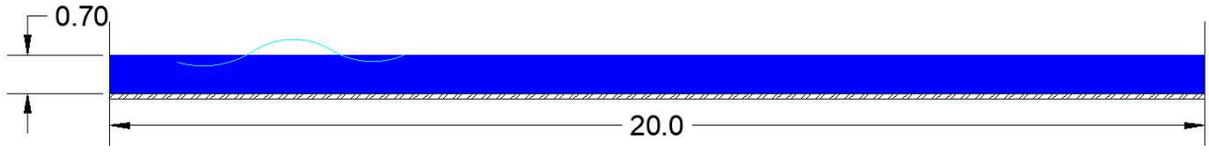


Figure 2.1 numerical domain dimensions

The dimensions of the three-dimensional numerical wave flume were as follows:

- Length = 20 m;
- Depth = 0.7 m;
- Width = 1 m;
- Height = 1.3 m.

2.1.2 Boundary conditions

Boundary conditions are specified as follows:

- *Inlet*: a relaxation scheme was implemented; wave height is set to 0.1 m according to stokes I theory, period of 1.5 s. The relaxation zone follows a rectangular shape, with a length of 1.5 times the wave length, which is 3.5 m. This leads to a 5.5 m long inlet zone.
- *Outlet*: a sponge absorption layer to avoid reflection is set by implementation of a rectangular relaxation zone of twice the wave length.
- *Walls*: a free slip condition was set, to avoid additional dissipation due to the interaction between walls and water. the pressure is set to zeroGradient.

- *Bottom*: no slip conditions were set to simulate the interaction with the wave, so the conditions were the same as the wavescreen.
- *Atmosphere*: An inletOutlet condition was set for the velocity. The pressure was set to totalPressure.

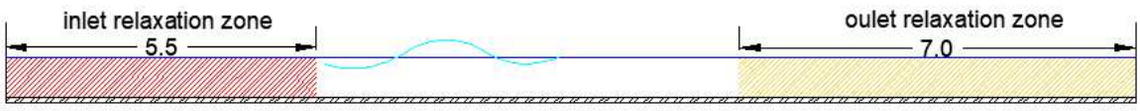


figure 2.2 relaxation schemes zones

2.2 Validation Case

A comparison with a physical experiment was carried out for the numerical model validation. Martinelli et al. (2018) investigated the intensity of the load applied by non-breaking waves on the recurved parapet wall of vertical breakwaters. In this study, the water depth was 0.70 m and the bottom was horizontal. The piston type mode was used, with active wave absorption.

2.2.1 Setup

A vertical caisson breakwater was built directly on the horizontal bed. The recirculation system is open, although not forced, to maintain the same water

level in front and behind the structure, even in presence of wave overtopping. The crest freeboard, equal to the parapet height *h*, is 0.14 m.

The distance of the model structure from the wave generator is 15.8 m.

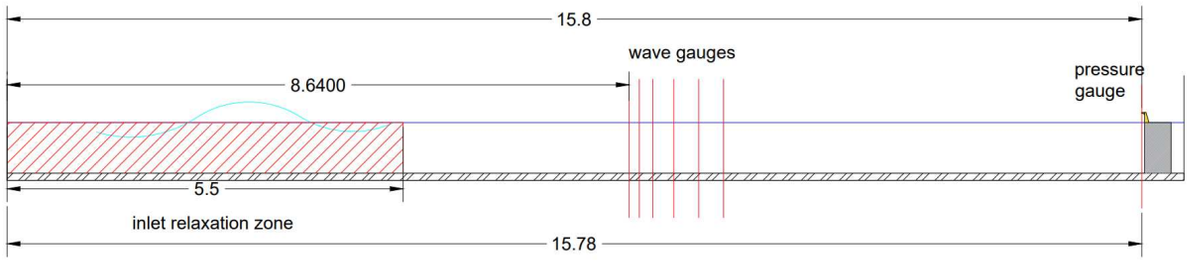


Figure 2.3 physical experiment set-up

Four types of parapets were tested, each one characterized by a different exit angle, i.e. the angle between the vertical plane and the plane tangent to the wall recurve: 0° (plane vertical wall); 45°; 60°; 90°.

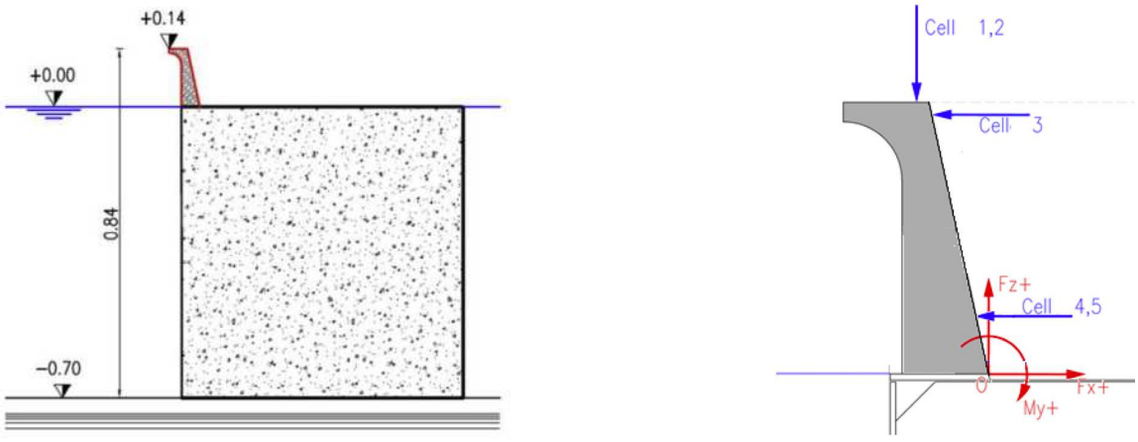


figure 2.4 parapet geometrical features

The 90° case was modeled In this work. The numerical model kept the same geometries and probe locations.

The parapet's geometry was created using a CAD software, then converted to STL format and meshed using the snappyHexMesh tool in OpenFOAM. The cell size reference was set to 0.005 m. Two different wave fields were modeled, both regular and irregular. The wave height and period values were selected from scenarios proposed in the numerical paper. Two-dimensional cases lasting 80 seconds were simulated. The horizontal component of the force (N/m) was calculated by integrating the dynamic pressure along the vertical axis at each time step:

$$F = \int_0^h p_d dz$$

Where h is the parapet height and p_d was plotted for every time step along the vertical in OpenFOAM solving the Navier Stokes equations. Peak value of the force in time was considered.

For the first case (WS8 in the paper) , a Stokes I wave with a height of 0.16 m and a period of 1.6 seconds was set at the inlet as a boundary condition.

For the second case (WS11 in the paper) , a stokes I order wave with a height of 0.14 m and period of 1.8 s was set at the inlet of the boundary condition.

Reflection occurred at the structure, which was modeled with a no-slip condition to simulate the physical wave flume with accuracy. Six wave

gauges were used to compute the wave height in the flume, and the peak and significant wave heights were calculated.

2.3 Wave energy dissipation

Wave energy is a quantity of primary interest. In water waves, the most used energy measure is the mean wave energy density per unit horizontal area. It is the sum of the kinetic and potential energy density, integrated over the depth of the fluid layer and averaged over the wave period. Simplest to derive is the mean potential energy density per unit horizontal area E_{pot} of the surface gravity waves, which is the deviation of the potential energy due to the presence of the waves:

$$E_{pot} = \int_{-h}^{\eta} \rho g dz - \int_{-h}^{0} \rho g dz$$

The overbar denotes the mean value (which in the present case of periodic waves can be taken either as a time average or an average over one wavelength in space).

The mean kinetic energy density per unit horizontal area $E_{kin}$of the wave motion is similarly found to be (Phillips, 1966):

$$E_{kin} = \overline{\int_{-h}^{0} \frac{1}{2}\rho[|U + u_x|^2 + u_z^2]dz} - \int_{-h}^{0} \frac{1}{2}\rho\,|U|^2 dz$$

Where U is the current velocity, set to zero in this study case.

This defines the final espression for the kinetic energy:

$$E_{kin} = \int_{-h}^{0} \frac{1}{2} \rho [{u_x}^2 + u_z^2] dz$$

Wave energy dissipation is the loss of energy that occurs as waves propagate through the water. It is an important factor in the behavior of waves, as it determines the amount of energy that is available to do work (such as breaking on a beach or driving a wave energy converter).

To compute energy, both velocity and surface elevation (η) values were needed. The first term determines the kinetic energy, while the latter refers to the potential energy. To have a precise value of the wave energy offshore and inshore, it is necessary to integrate over a wavelength both velocity square and η. The reflection that occurs offshore is not perfect. Inshore, there's one dominating amplitude instead, which gives the actual length over which we defined the integration process.

The energy dissipation was computed as follows:

$$E_{reflected} = E_{tot\ offshore} - E_{incident}$$

$$\Delta E_{screen} = E_{incident} - E_{transmitted} - E_{reflected} - \Delta E_{bottom\ friction}$$

The Incident wave Energy ($E_{incident}$) was defined with a *Freestream* (without structure) simulation by integration of both kinetic and potential energy in the offshore control volume. In the same simulation, the energy was computed in the inshore control volume, in order to define the energy losses due to the bottom friction ($\Delta E_{bottom\ friction}$)

$E_{reflected}$ was defined by subtraction of the incident wave energy to the total offshore energy in a simulation with the screen.

$E_{transmitted}$ was defined integrating the energy inshore.

With all the informations given above, it is possible to compute ΔE_{screen}, which defines the actual dissipation due to the presence of the porous wavescreen.

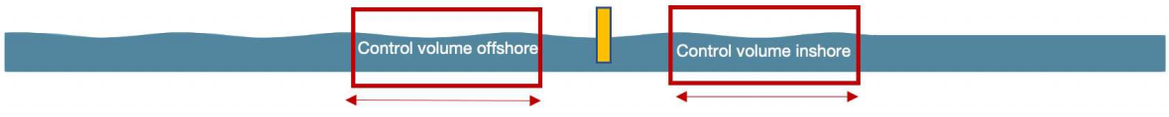


Figure 2.5: integration control volumes

This value (ΔE_{screen}) was compared to the Borda- Carnot equation . This theory explains, with good approximation, the values obtained.

2.3.1 Borda-Carnot equation

In fluid dynamics the Borda–Carnot equation is an empirical description of the mechanical energy losses of the fluid due to a (sudden) flow expansion. It describes how the total head reduces due to the losses. This is in contrast with Bernoulli's principle for dissipationless flow (without irreversible losses), where the total head is a constant along a streamline. The equation is named after Jean-Charles de Borda (1733–1799) and Lazare Carnot (1753–1823).

This equation is used both for open channel flow as well as in pipe flows. In parts of the flow where the irreversible energy losses are negligible, Bernoulli's principle can be used.

In this work, a comparison with both the sudden expansion and contraction of a pipe is made.

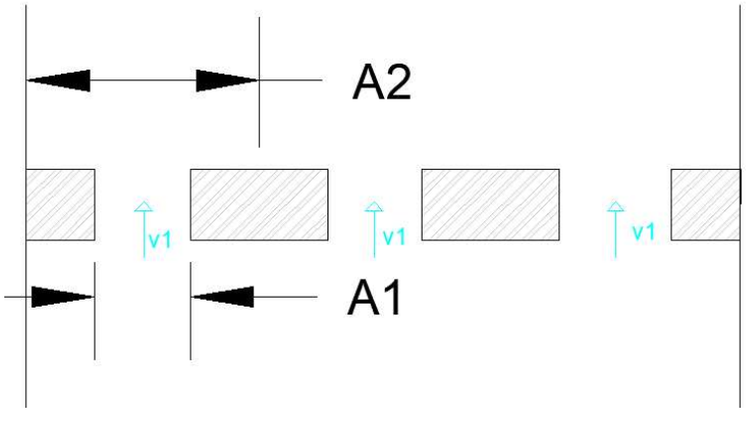


figure 2.6 Borda theory scheme

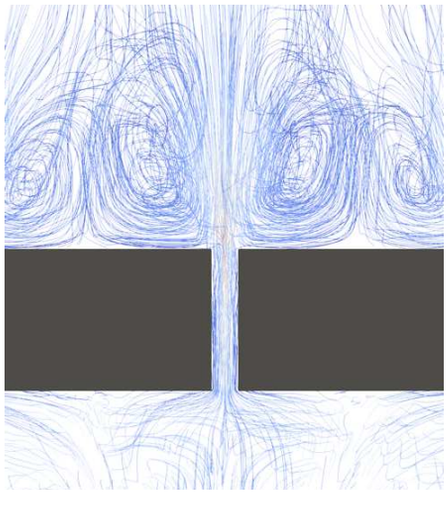


figure 2.7 numerical model streamlines

The Borda–Carnot equation is applied to the flow through a sudden expansion of a horizontal pipe. At cross section 1, the mean flow velocity is equal to v_1 and the cross-sectional area is A_1. The corresponding flow quantities at cross section 2 – well behind the expansion (and regions of separated flow) – are $v_2$and A_2, respectively. At the expansion, the flow separates and there are turbulent recirculating flow zones with mechanical energy losses. The loss coefficient ξ for this sudden expansion is approximately equal to one: ξ ≈ 1.0. Due to mass conservation, assuming a constant fluid density ρ, the volumetric flow rate through both cross sections 1 and 2 has to be equal (*fig 2.6*):

$$A_1 v_1 = A_2 v_2 \quad \text{so} \quad v_2 = \frac{A_1}{A_2} v_1$$

Consequently , integrating in the slits and according to the Borda–Carnot equation the mechanical energy loss in this sudden expansion is:

$$\Delta E_{expansion} = \int_{-h_2}^{\eta} \frac{1}{2} \rho \left(1 - \frac{A_1}{A_2}\right)^2 v_1^2 \, dz$$

$$\Delta E_{contraction} = \int_{-h_2}^{\eta} \frac{1}{2} \rho (\frac{1}{\mu} - 1)^2 v_1^2 \, dz$$

Values of the orbital velocity v_1 will be computed in the middle of the slits. Since the thickness of the screen is small, friction is neglectable. $\frac{A_1}{A_2}$ is the porosity and μ was fixed to 0.61 according to the shape value.

This theoretical values for dissipation are valid for:

- Steady flow
- No wall friction losses
- Equal pressure in Upstream/downstream wall of the wave screen
- Turbulent flow

2.3.2 Wave induced pressure

According to Stokes' first linear wave theory, the dynamic pressure p_d of a surface wave can be expressed as:

$$p_d = \rho g h \, (1 - cos(k\, x - \omega t))$$

Where:

ρ is the fluid density, g is the acceleration due to gravity and h, k , ω , x and t are defined as *section 1.2.3*.

This equation assumes that the wave is small enough that the velocity and acceleration of the fluid particles are small compared to the local acceleration due to gravity. It also assumes that the wave profile is sinusoidal.

In this work, dynamic pressure will be compared with the Modified Goda equations for Non breaking waves (Wiebe et al. 2014).

Goda (1974) proposed equations for estimating the pressure distribution acting on a vertical caisson breakwater due to nonbreaking waves, as shown in *Fig. 2.8* In Goda's original definition, the breakwater was elevated on a rubble mound with crest elevation at depth d, with the base of the upright section at depth h' and depth offshore of the breakwater h. Goda (2010) took the design wave height, H_{max}, as 1.8 times the significant wave height ($H_{1/3}$), measured at depth h_b, at a distance of 5 $H_{1/3}$ seaward of the breakwater. The horizontal pressure distribution was then defined by four parameters, three of which are pressures: the maximum horizontal pressure, p_1, which is assumed to occur at the SWL; the pressure at the bed, p_2; and the pressure at the base of the upright section, p_3. The magnitudes of these pressures may be calculated as

$$p_1 = \frac{1}{2}(1 + \cos\beta)(\alpha_1\lambda_1 + \alpha_2\lambda_2 cos^2\beta)\rho g H_{max}$$

$$p_2 = \frac{p_1}{\cosh\left(\frac{2\pi h}{L}\right)}$$

$$p_3 = \alpha_3 p_1$$

The fourth parameter is η^*, the point above the SWL at which pressure goes to zero, and it may be calculated as

$$\eta^* = 0.75(1 + \lambda_1 cos\beta)\, H_{max}$$

Pressure is assumed to vary linearly from p_1 to p_2 and from p_1 and η^*. β indicates the angle of wave approach, and L is the wavelength of the design wave. The coefficients λ_1 and λ_2 account for varying breakwater geometry or structural type and are set equal to unity for a standard upright breakwater. The wave pressure coefficient, α_1, was empirically determined from laboratory data, while α_2 represents the increase in wave pressure with decreasing depth. Similarly, α_3 is determined by linearly interpolating between p_1 and p_2.

$$\alpha_1 = 0.6 + \frac{1}{2}\left[\frac{4\pi h/L}{\sinh\left(\frac{4\pi h}{L}\right)}\right]^2$$

$$\alpha_2 = \min\left\{\frac{h_b - d}{3h_b}\left(\frac{H_{max}}{d}\right)^2 , \left(\frac{2d}{H_{max}}\right)\right\}$$

$$\alpha_3 = 1 - \frac{h'}{h}\left[1 - \frac{1}{\cosh\left(\frac{2\pi h}{L}\right)}\right]$$

The relative elevation of the base of the structure with respect to the SWL is termed theair gap, a, which is negative for a partially submerged structure. Wiebeet al. (2014) used linear interpolation betweenp1at the SWL and either

p=0 at η^* (positive air gap) or p_3 at the bed (negative airgap) to calculate p_5, the pressure at the base of the structure (case a<0)

$$p_5 = (1 + \frac{a}{h'})(p_1 - p_3) + p_3$$

In this case, a =h'.

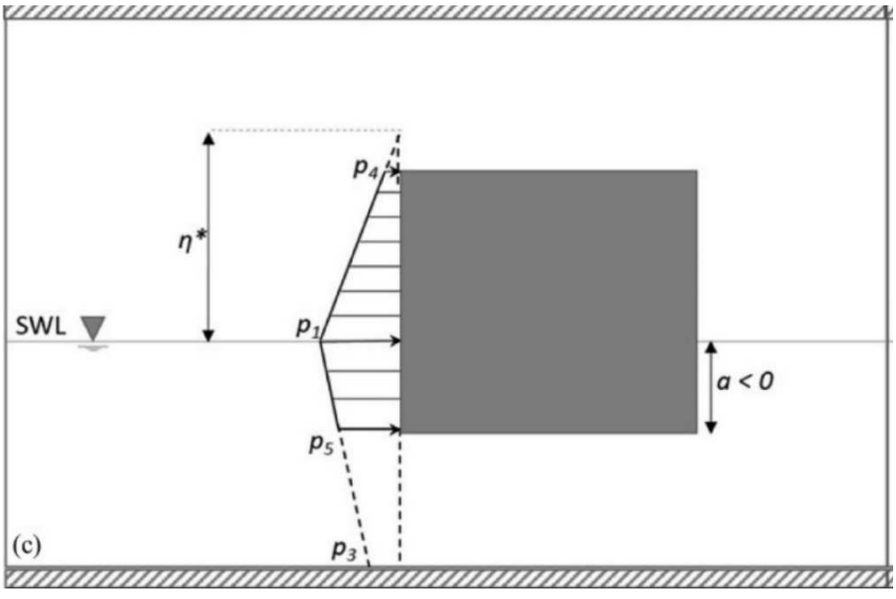


Fig 2.8 pressure distribution on a partially submerged structure according to the modified Goda formula (partially submerged case)

2.3.3 Turbulence modeling

Turbulence modeling refers to the use of mathematical models or empirical relationships to predict the behavior of turbulent flows. Turbulent flows are characterized by complex, chaotic patterns of motion that are difficult to

predict using traditional analytical methods. As a result, CFD simulations of turbulent flows typically rely on turbulence models to predict flow features.

There are several common methods used in CFD:

- *k-ε*: The k-ε model is a two-equation turbulence model that is based on the Reynolds-averaged Navier-Stokes (RANS) equations. It involves solving for two turbulence quantities: the turbulent kinetic energy (k) and the turbulent dissipation rate (ε). The k-ε model is relatively simple and computationally efficient, but it can be less accurate in predicting the behavior of complex turbulent flows.
- *K-ω*: The k-ω model is a two-equation turbulence model that is similar to the k-ε model, but it involves solving for different turbulence quantities: the turbulent kinetic energy (k) and the specific dissipation rate (ω). The k-ω model is generally considered to be more accurate than the k-ε model, but it is also more computationally demanding.
- *K-ω SST*: The k-ω SST model is an improved version of the k-omega model that is based on the detached eddy simulation (DES) approach. It involves solving for the same turbulence quantities as the k-ω model, but it includes additional terms to account for the effects of separation and strong adverse pressure gradients.

- *K-ω SST buoyancy corrected*: the buoyancy correction is used in the k-omega SST turbulence model to account for the effects of buoyancy on the flow. Buoyancy is a force that acts on a fluid due to differences in the density of the fluid and the surrounding air or water. It can have a significant impact on the flow of fluids, particularly in cases where there are large temperature or humidity gradients.

The wave height of the turbulence model without the buoyancy modified model was significantly smaller. This was due to the fact that turbulent kinetic energy was transformed from the total kinetic energy and dissipated due to viscous forces, so an unsuitable turbulence model reduced the total kinetic energy excessively. The increase of buoyancy modified model effectively reduces the energy dissipation and increases the stability of the model to contribute to the accuracy of the calculation. (Wang et al, 2018) Therefore, the turbulence model adopted in the present study is the recently presented buoyancy-modified k-ω SST model, which could result in stable wave propagation without significant wave damping over the length of the wave flume (Devolder et al., 2018). The purpose of including a buoyancy term in this study is twofold. The objective is to suppress the turbulence level at the free water surface, i.e. in the zone where the governing direction of the density gradient is vertical (predominantly horizontal free water surface). More specific, this is the zone near the interface where non-breaking waves

are propagating the model switches to a laminar regime near the free water surface, preventing excessive wave damping.

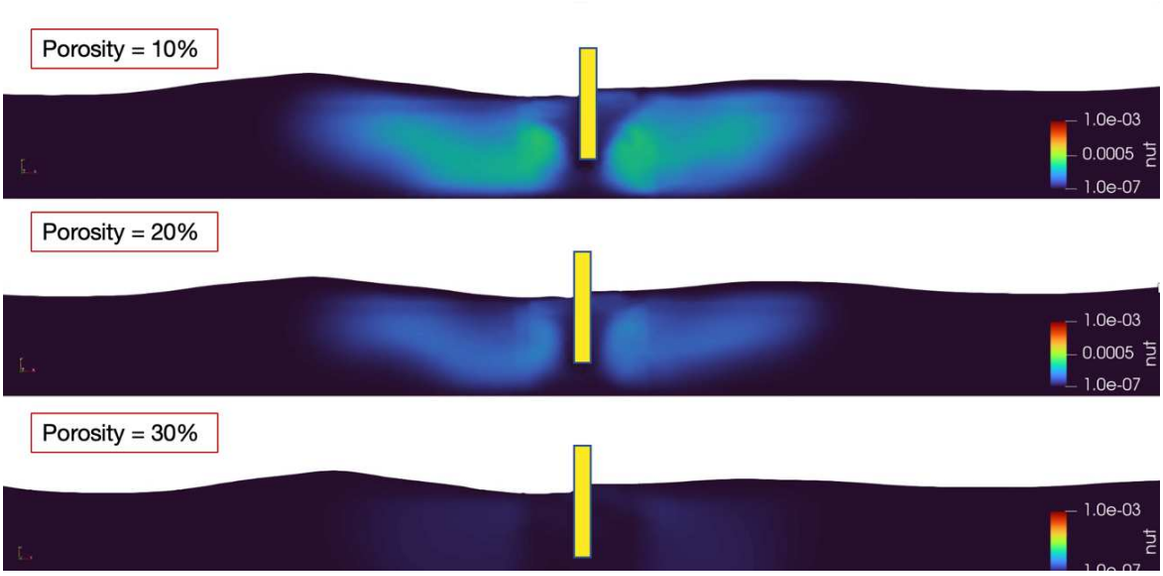


figure 2.9 turbulence model with varying porosity comparison

2.4 Geometries

A python script for the geometries generation wass implemented. Since the ideal and physical case (physical = wave flume ICEA) are both considered per linear meter, the screen width is fixed to W = 1m.

The screen was divided into two parts, void and solid, which had different dimensions depending on the number of slits and porosity. The submergence is calculated as the distance from the bottom of the screen to the water surface. The structure goes up to the top of the domain. Screeen's height is

fixed to 1m as well. Therefore, the screen define a 1 square meter area, extendable in both direction for real case (i.e a 200m breakwater).

The dimensions of void and solid are defined V and S. porosity is called d, which means that:

$$V = \frac{W}{N} d$$

where N is the number of slits. On the contrary; the solid part is defined by:

$$V = \frac{W}{N} (1 - d)$$

This dimension is distributed symmetrically on the screen , generating N+1 blocks, where the first and last block have a S/2 width.

The last parameter, orientation, is obtained in the geometry by rotating the slits around a point obtained matching the diagonal and the baricenter of the slit itself.

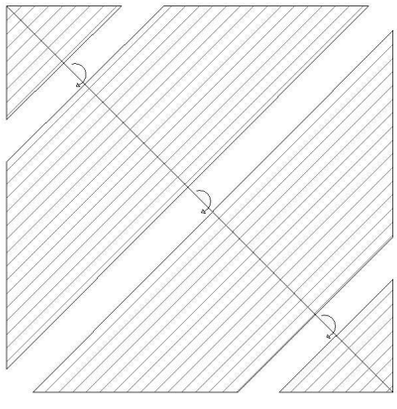


figure 2.10 rotational axis of the slits

The final results were obtained by using three different combination of porosity, three of orientation, three of submergence and two number of slits. Two different values of thickness are provided, referring to usual industrial sizes for aluminium panels (2.5 and 3.5 inches). The final different combination were 108.

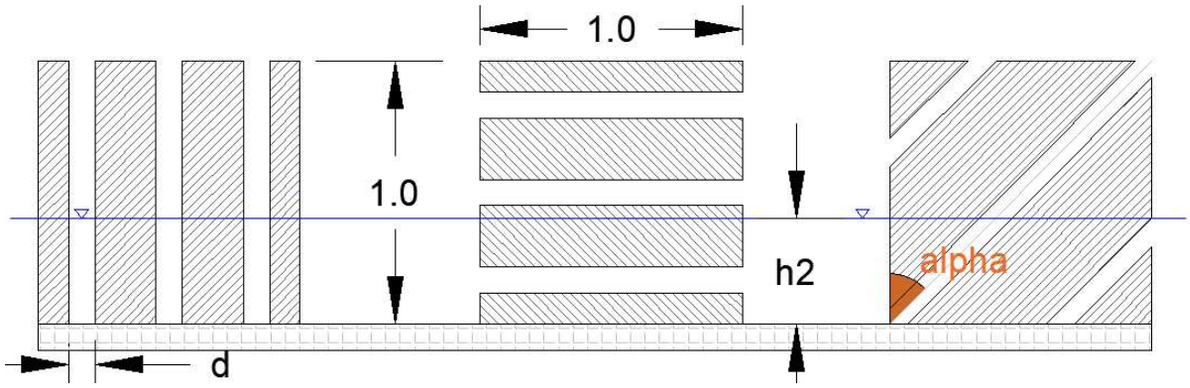


figure 2.11 wavescreen features

2.5 Grid refinement

In numerical simulations, damping can occur due to the discretization of the system, such as using a coarse grid mesh. When a continuous system is modeled using a discrete computational grid, the solution can become less accurate and the amplitude of the oscillations can be damped. The numerical dissipation, which is the damping effect caused by the discretization, can be more pronounced when a coarser grid is used. This is because a coarse grid does not accurately resolve the small scale features of the solution and thus,

the energy of the system is not conserved. This can lead to a damping of the oscillations and a reduction in the overall accuracy of the solution.

Grid refinement refers to the process of increasing the resolution or detail of a mesh, typically in a computational fluid dynamics (CFD) simulation. Refining the mesh can improve the accuracy of the simulation, but it also increases the computational time and resources required to run the simulation. As a result, it is important to balance the need for accuracy with the need for computational efficiency when deciding on the level of mesh refinement to use.

Refinement can be defined depending on the approach achieved:

- *Local refinement*: This involves increasing the resolution of the mesh in specific areas or regions where it is needed, while leaving the rest of the mesh at a coarser resolution. Local refinement can be useful when the flow exhibits significant variations in scale or when certain features or phenomena are of particular interest.
- *Global refinement*: This involves increasing the resolution of the entire mesh uniformly, in order to improve the accuracy of the simulation overall. Global refinement can be useful when the flow is relatively uniform and there are no specific areas that require higher resolution.

- *Adaptive refinement*: This involves dynamically adjusting the resolution of the mesh based on the flow conditions and other factors. Adaptive refinement can be useful when the flow exhibits complex or rapidly changing phenomena that are difficult to predict in advance.

In this work, the grid refinement was done in a local way as follows:

15 different scenarios of refinement had been chosen (with increasing computational time required) in order to reach a good precision. What is important for the model is to avoid wave damping, but requiring as less computational power as possible.

The parameter analyzed in this work is wave height. Mean wave height was computed as an output at 2 wave length distance from the boundary, where the input wave was 0.12 m, following the Stokes II order theory. A convergencence was expected to be found to the value of 0.12.

The value chosen as a reference for the grid analysis sets the cell dimensions to 0.00625 m in the surface area, corresponding to the scenario number 12 of the simulations for grid refining.

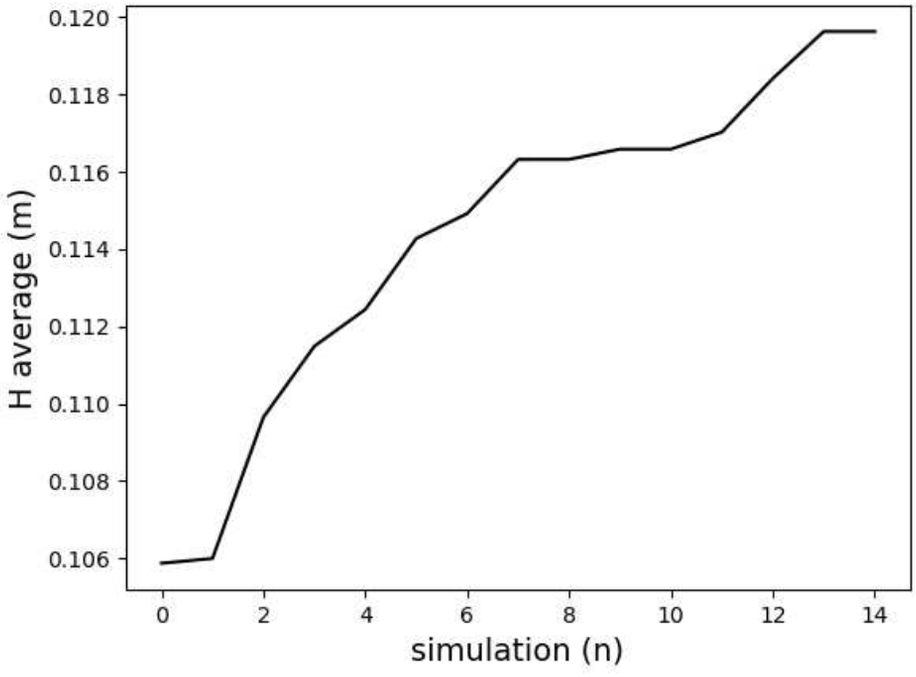


figure 2.11 grid refinement analysis plot

2.5.1 Y+ and boundary layers

Y+ is a dimensionless quantity used in computational fluid dynamics (CFD) numerical modeling to determine the appropriate mesh size and near-wall treatment for accurately predicting turbulent flows. It represents the ratio of the distance from a solid surface to the thickness of the boundary layer, normalized by the molecular viscosity of the fluid.

In CFD simulations, the behavior of fluids near solid surfaces is important to accurately model. The boundary layer, which is a thin layer of fluid in close proximity to the surface, experiences friction and slows down relative to the bulk fluid. This results in a gradient of velocities, temperatures, and other properties within the boundary layer, which can cause turbulence. The

thickness of this boundary layer is important for accurate predictions of heat transfer and other properties.

The Y+ value determines the type of turbulence model to use, and the appropriate mesh size and near-wall treatment for accurate predictions. If Y+ is less than 1, then the flow is considered to be in the viscous sublayer, and a wall function can be used to model the near-wall flow. If Y+ is greater than 30, then the flow is considered to be in the log-law region, and a turbulence model can be used to predict the flow. In between these values, the appropriate treatment depends on the specific flow and the desired level of accuracy.

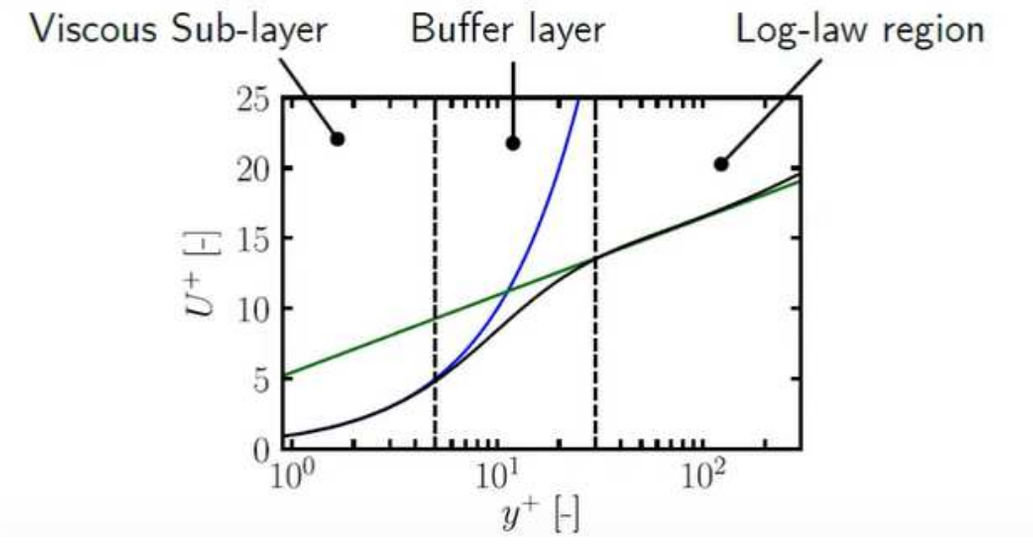


figure 2.13 definition of layers with respect to the Y+

In this work, 4 different scenarios of Y+ had been considered. First, a very fine mesh with a viscous sublayer and Y+ < 1 had run, defining the most

precise mesh of all. The values were compared to more coarse meshes with Spalding method wall functions in order to save computational time.

Table 1 grid comparison

Ei=17.1 J/m^2	Spalding Wall Function			Solving Boundary layer		
	DE (%)	F(N/m)	comp time (16 cores)	DE (%)	F(N/m)	comp time (16 cores)
Y+<5	64.80%	252.4	88 h 15 min	63.30%	249	67 h 15 min
5<Y+<30	65.80%	253	18 h 24 min	ND	ND	ND
30<Y+<100	67.20%	253.8	17 h 9 min	ND	ND	ND

As shown in *Table 1*, the Spalding wall function does not work well for Y+ < 5, which leads to avoid this mesh size. Different behaviour for 30 < Y+ < 100, where the results are acceptable and the computational time is 20% of the one required for the fine mesh.

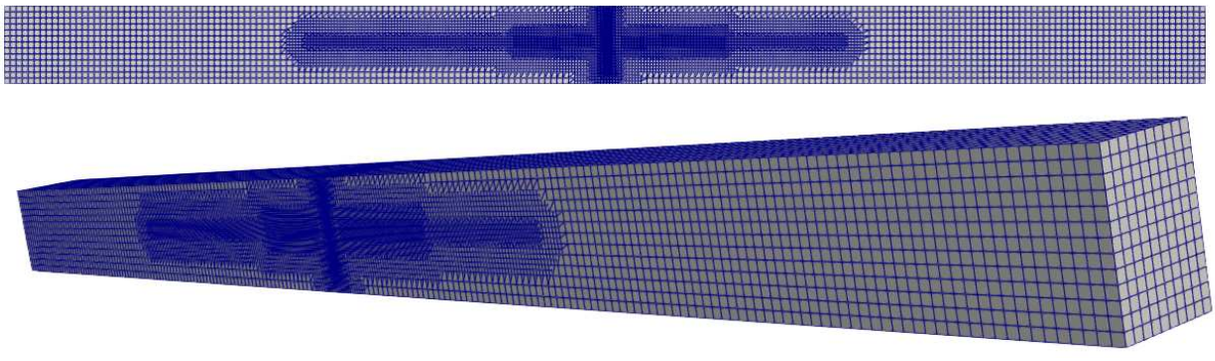


figure 2.14 final grid mesh

Chapter 3: Results

In this chapter, all the results obtained are compared and discussed.

3.1 Model validation results

In this section, the validation results are shown. For the physical model, we referred on the comparison among Forces and both peak and significant wave height, while for the pressure distribution and Energy the comparison was made with the second order Stokes theory.

A 2-dimensional, 80 s simulation was run on 1 core for 3 hours. As outputs, pressure values and surface elevation tracking were considered. The significant wave height H_s was calculated as:

$$H_s = \frac{1}{n} \sum_{H_{\frac{2}{3}}}^{H_{max}} H_i$$

Where n is a third of the numer of waveheight registered and $H_{\frac{2}{3}}$ is the value of the waveheight that goes over the 66% of maximum value. Pressure is calculated by integration of the dynamic pressure on the structure. Kr is computed by Fourier transform, and H peak is computed as the maximum wave height registered.

Table 2 validation results

parameter	WS8			WS11		
	Num	Fis	Err %	Num	Fis	Err %
H,s (m)	0.163	0.166	1.8	0.136	0.137	0.7
H,peak (m)	0.169	0.17	0.5	0.141	0.14	0.7
F,peak (N)	27.5	28	1.8	21.1	20	5.5
Kr	0.92	0.93	1.1	0.95	0.95	0

As shown in *Table 2* the physical and numerical model show excellent agreement.

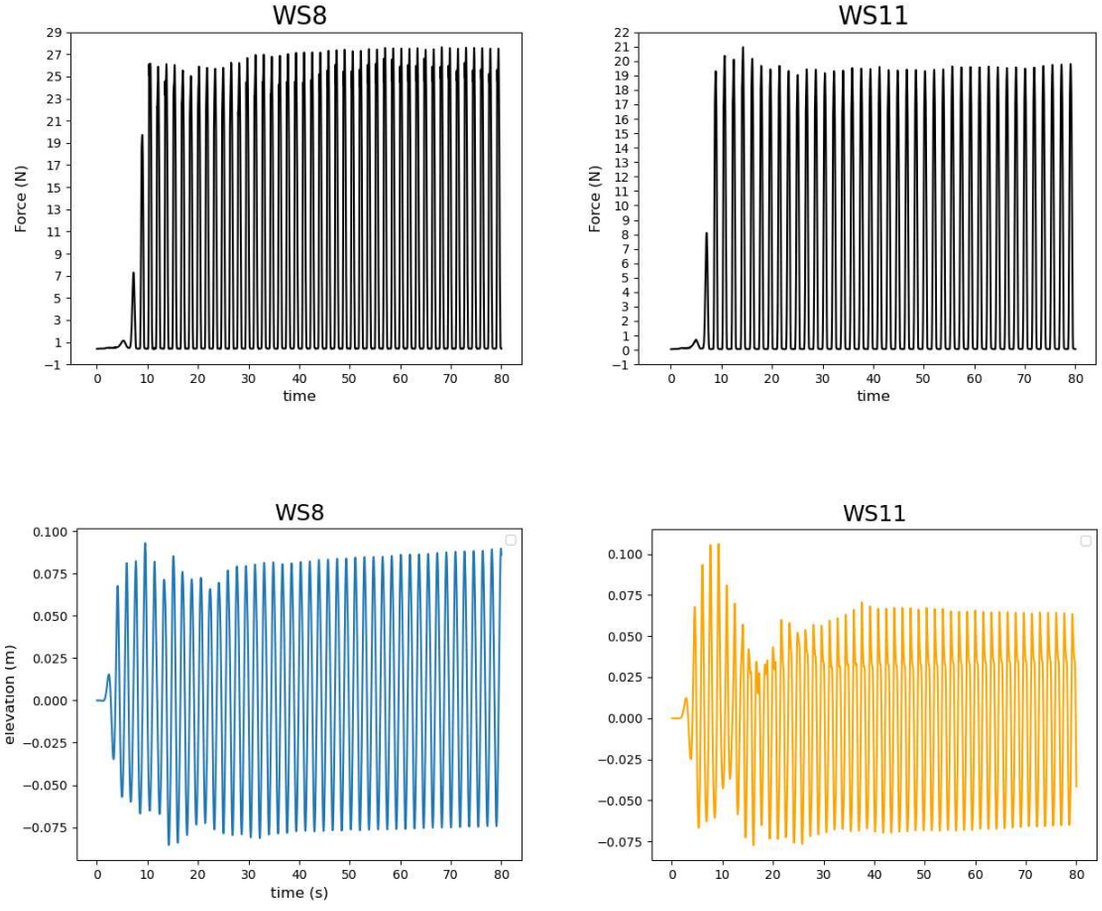


Figure 3..1 Forces and Surface Elevation for cases WS8 and WS11

Based on the comparison of numerical and physical model results, it can be concluded that the model is performing well the wave generation and propagation.

3.1.1 Pressure comparison with goda formula

Three simulation with a non-porous structure were run to test the pressure distribution and compare it with the modified Goda formula for partially

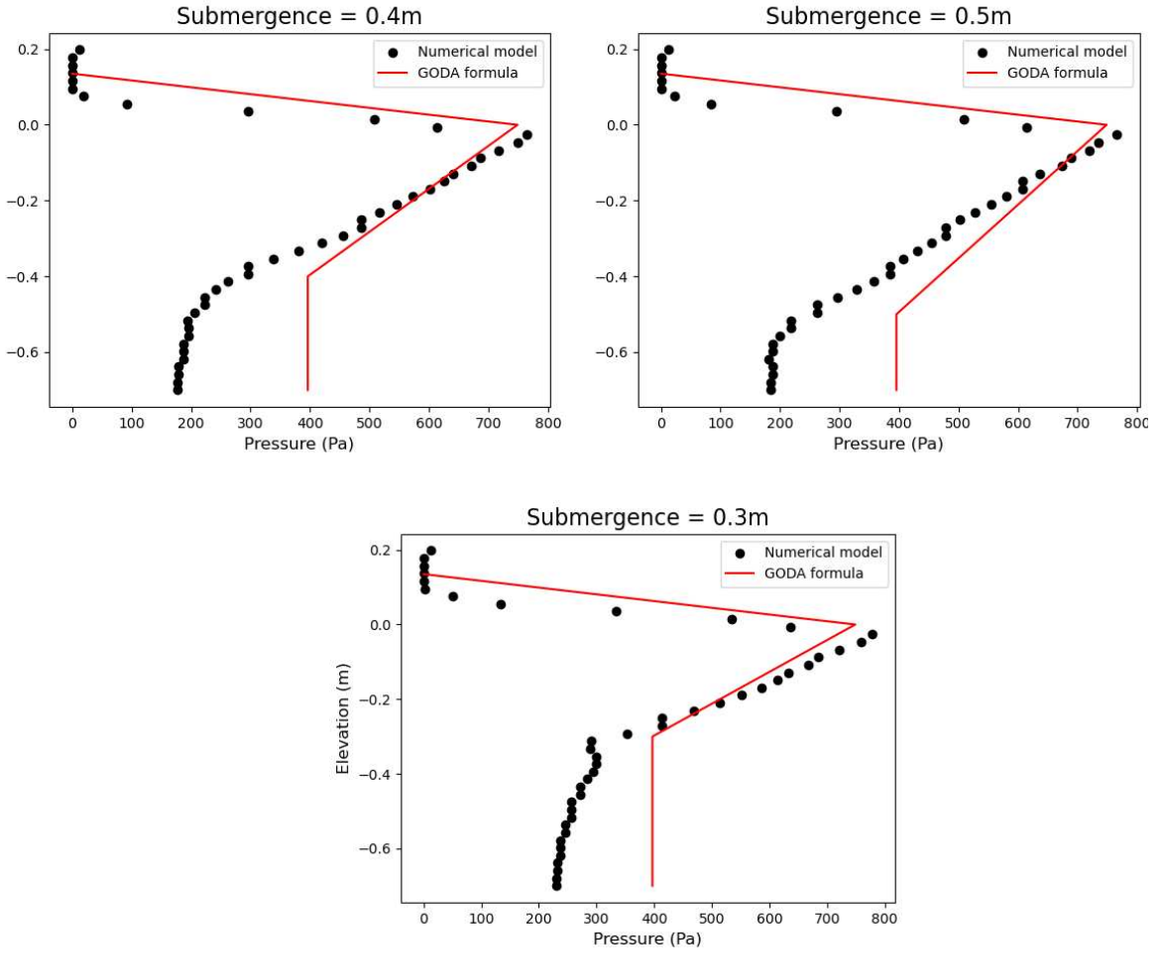


Figure 3.1 GODA modified formula – model comparison

submerged structures (Wiebe et. Al, 2014). The difference between the simulation is according to the different values of submergence.

As seen in the plots, the sperimental theory and the numerical model show good agreement.

3.1.2 Comparison with second order Stokes Theory

A comparison with the second order theory was made. First, the surface elevation was checked. The Second Order theoretical wave is computed as shown in *section 1.2.3*, while the numerical values where averaged over a section.

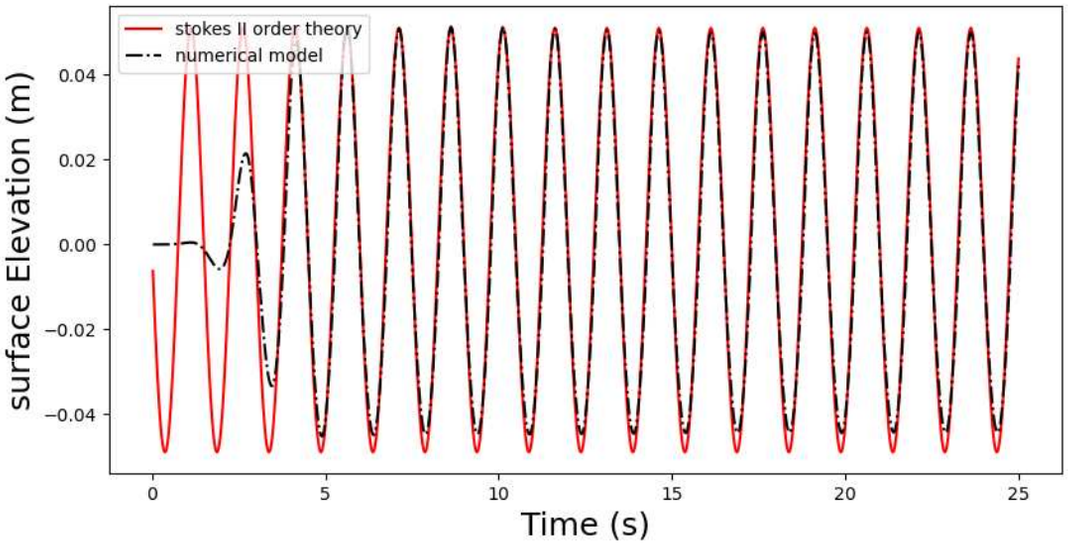


Figure 3.2 Stokes second order theory – model comparison on surface elevation

Which showed good agreement.

Then, to validate the energy integration, a simulation with perfect reflection was run. The energy was expected to be four times the original energy:

$$\mathrm{E} \approx \frac{1}{2}\, \rho\, g\, \eta(t)^2_{Stokes\ II}\ ;$$

Therefore:

$$E_{p,reflected} = 2\, \rho\, g\, \eta(t)^2_{Stokes\ II\ order}$$

Then the comparison between theory and numerical model is shown in the chart below:

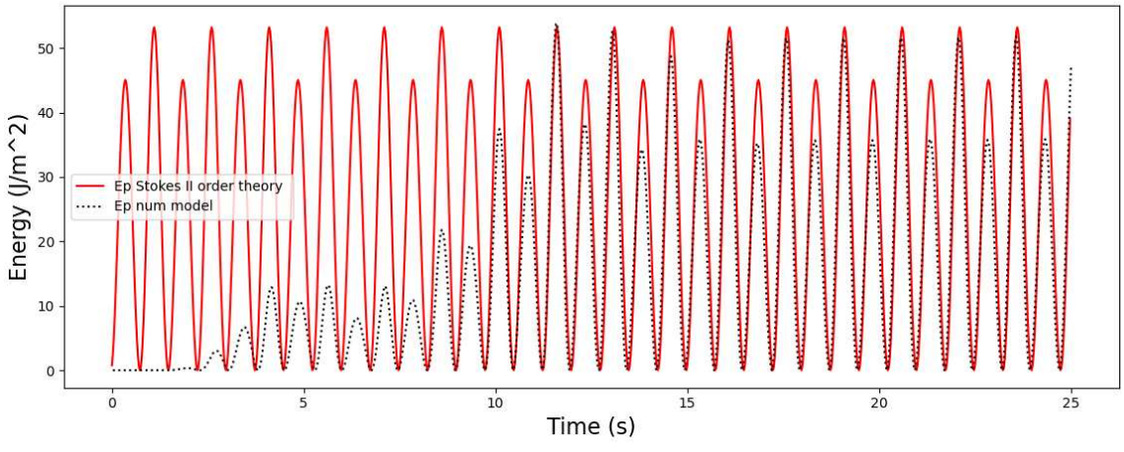


Figure 3.3 Stokes second order theory – model comparison on effects of reflection

The predicted energy and the theoretical one showed good agreement. A slight difference is made by a not perfect modeling of the through, although the peak are computed with very good agreement.

3.2 Model results

In the next section, all the results for the different parameters and features are analyzed and compared.

3.2.1 Thickness

Two values of thickness were considered, according to the standard aluminium panel sizes in the United States:

- 2.5 inches
- 3.5 inches

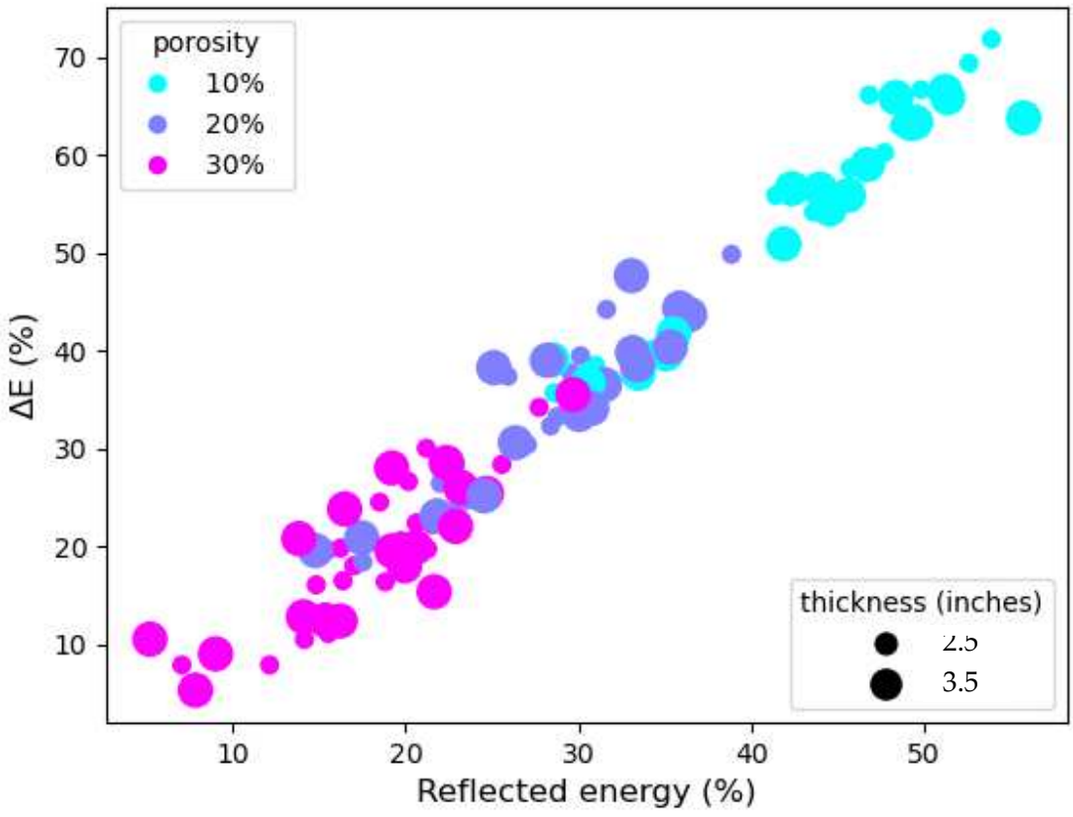


Figure 3.4 Dissipation/Reflection analysis with different thickness and porosity values

The first comparison is between Reflected Energy and Dissipation. The calculation for the dissipation has been made considering the percentage of incident energy that is dissipated by the structure, while the reflected energy was the percentage with respect to the total energy offshore.

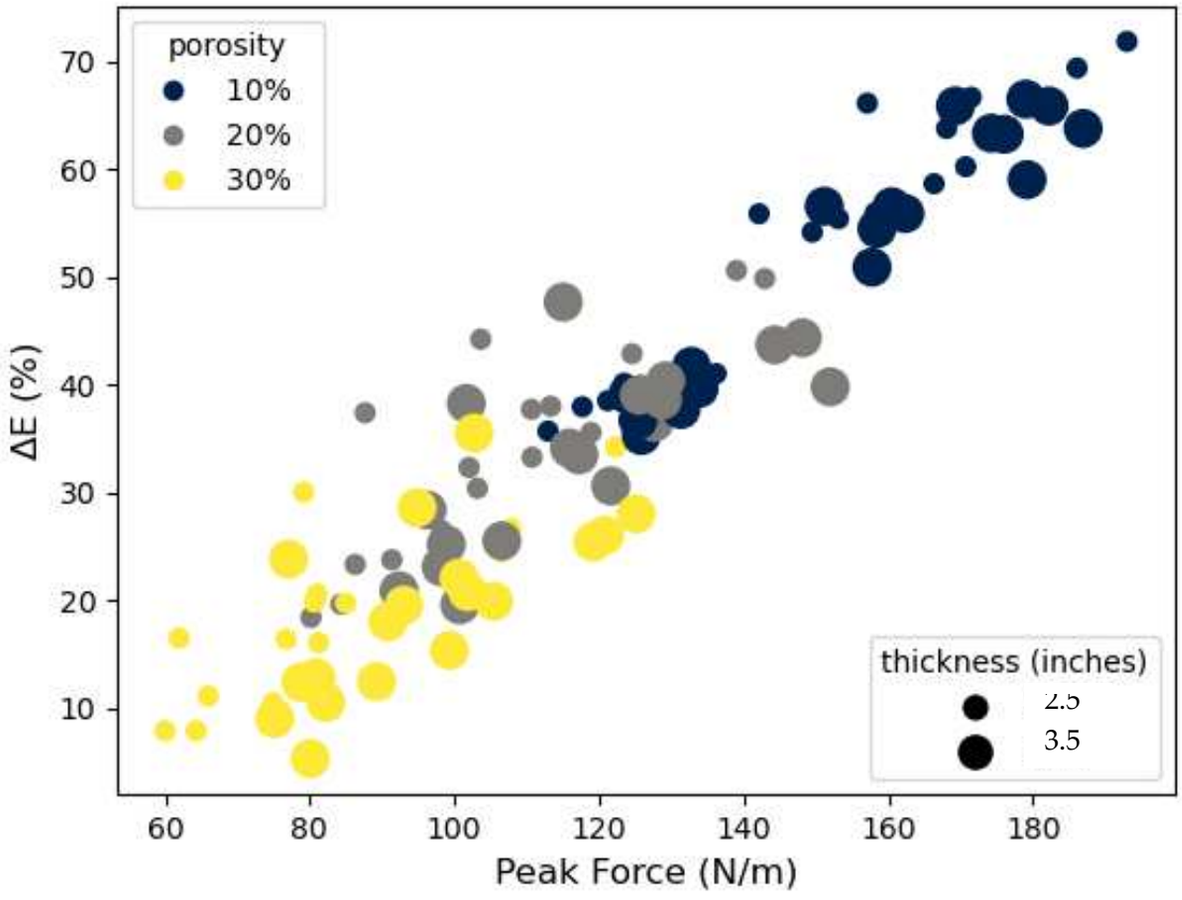


Figure 3.5 Dissipation/Forces analysis with different thickness and porosity values

The second comparison was made between the dissipation and Peak Force on the structure.

The main conclusion is that the thickness of the screen does not affect significantly neither Dissipation, Forces or Reflection. Dissipation is increased by a 5% average value, which would lead to an increase of 40% of material.

The first value (2.5 inches) was chosen to optimize the screen.

3.2.2 Number of slits

Number of slits (N) is a parameter that identifies the distribution of the pores all over the structure. The values selected for the number of slits (keeping the same porosity) were:

- 3 slits;
- 6 slits.

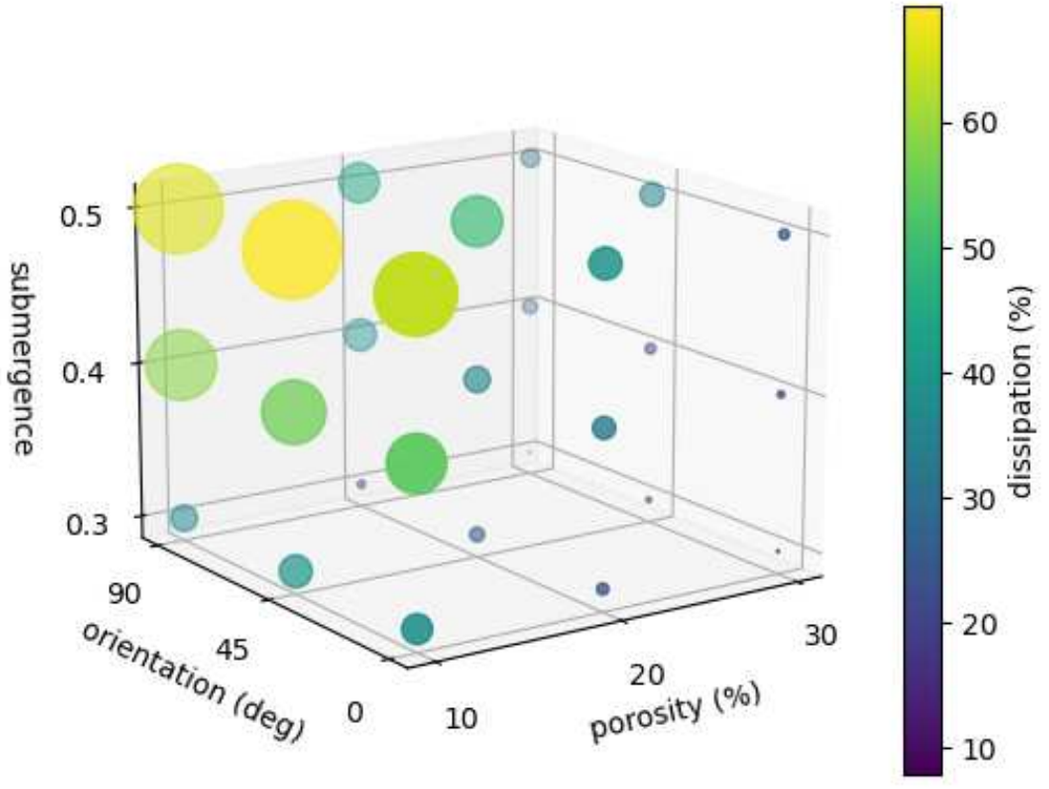


Figure 3.6 three-dimensional plot of the different parameters keeping N fixed to 3

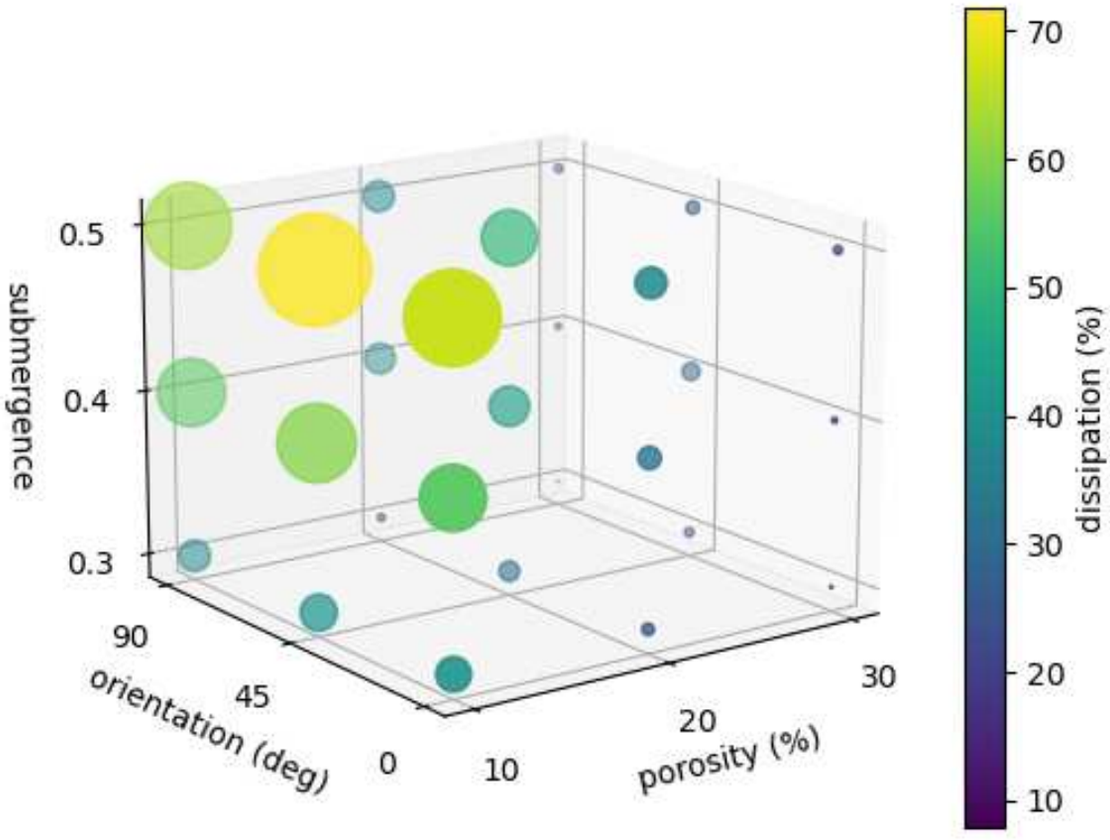


Figure 3.7 Dissipation: three-dimensional plot of the different parameters keeping N fixed to 6

The two plots do not show significant difference. Therefore, we concluded that distribution of slits does not affect significantly the dissipation. From this point on, the number of slits was fixed to 3 for the optimization of the screen.

Moreover, Reflection and Forces were considered.

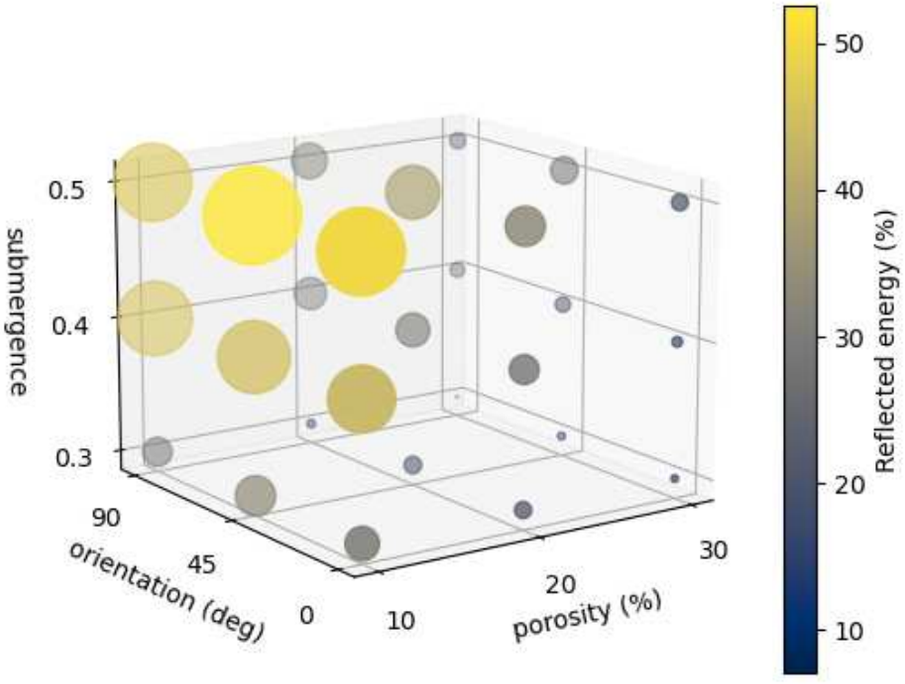


Figure 3.8 Reflection: three-dimensional plot of the different parameters keeping N fixed to 3

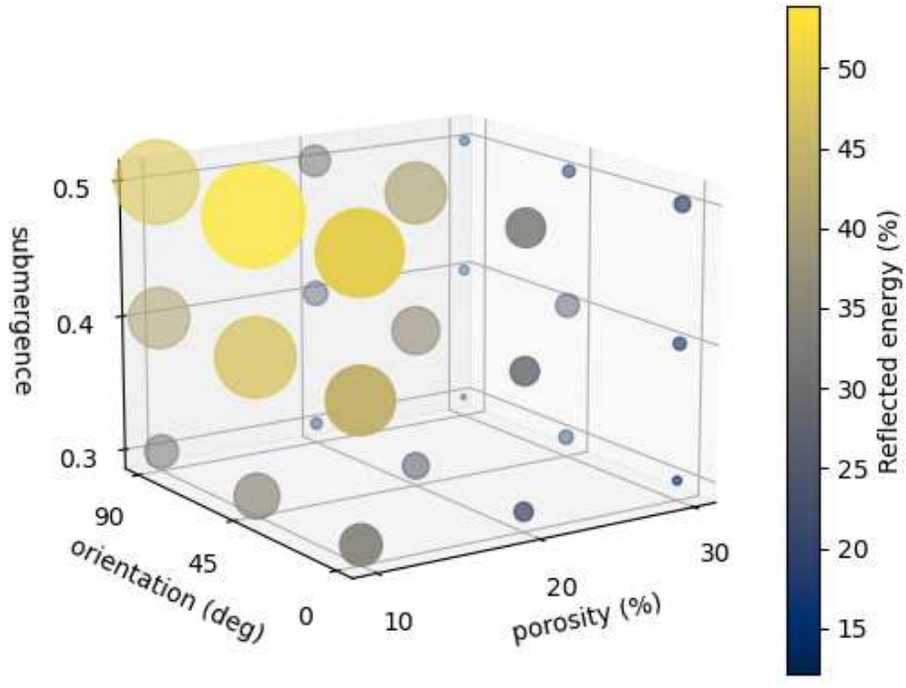


Figure 3.9 Reflection: three-dimensional plot of the different parameters keeping N fixed to 6

Reflection shows the same behaviour as Dissipation. Therefore, the same conclusion was made.

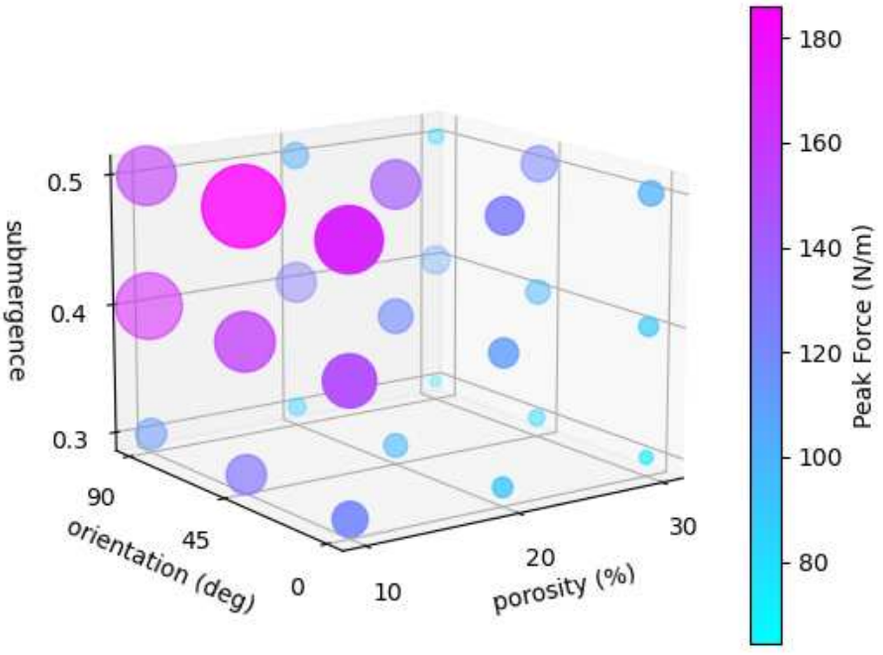


Figure 3.10 Forces: three-dimensional plot of the different parameters keeping N fixed to 3

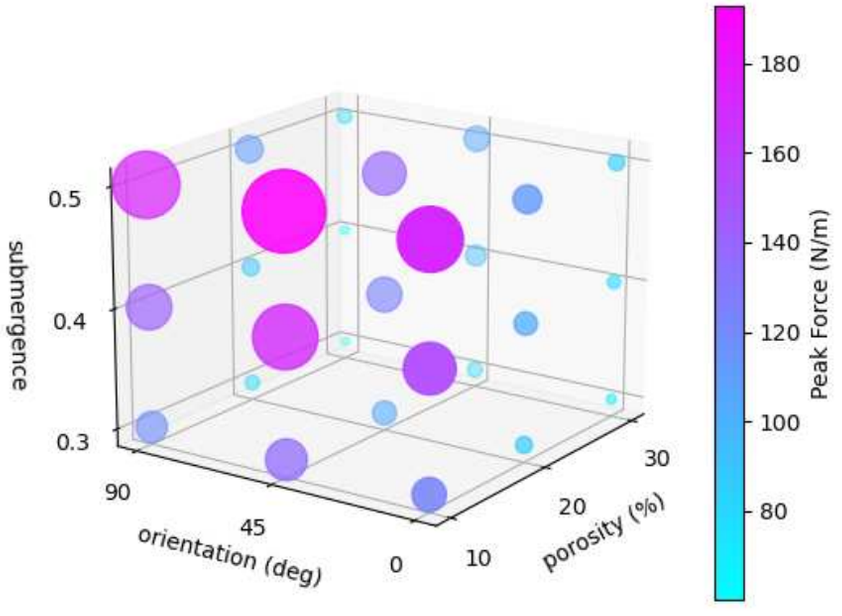


Figure 3.11 Reflection: three-dimensional plot of the different parameters keeping N fixed to 6

Peak forces do not change with porosity distribution.

3.2.3 Porosity and Submergence

Three values of porosity are considered:

- 10%;
- 20%;
- 30%.

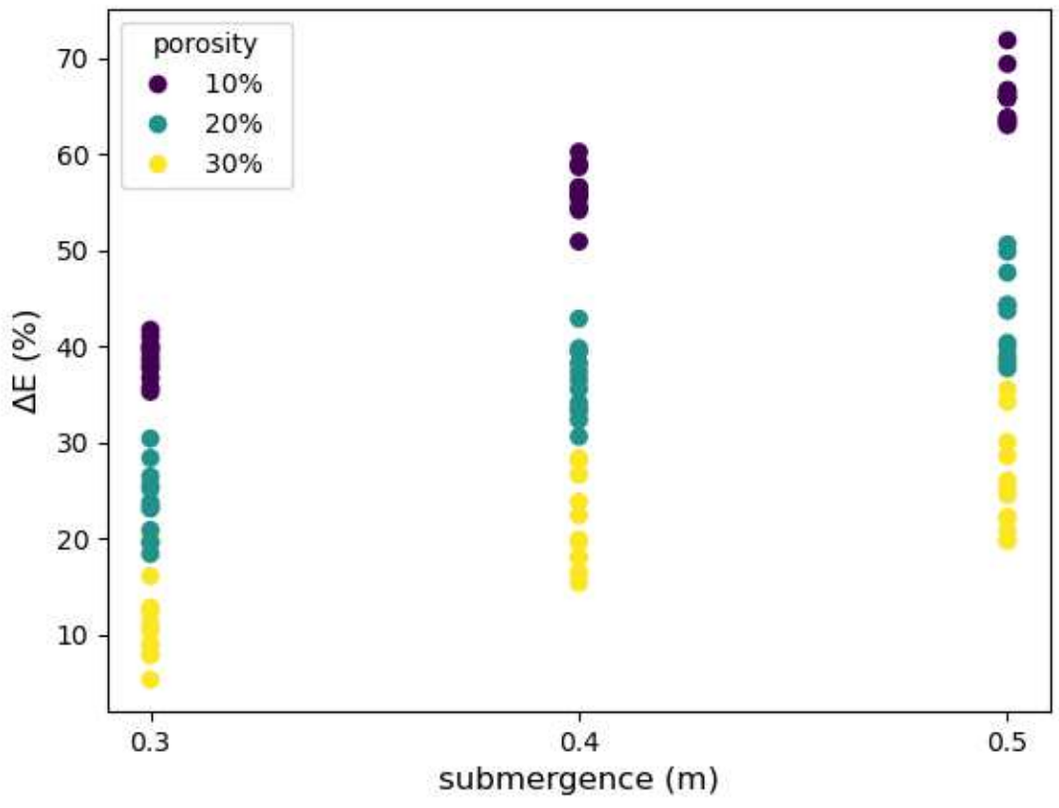


Figure 3.12 dissipation varying with porosity and submergence

The case of porosity 10% show a better increment then the others.

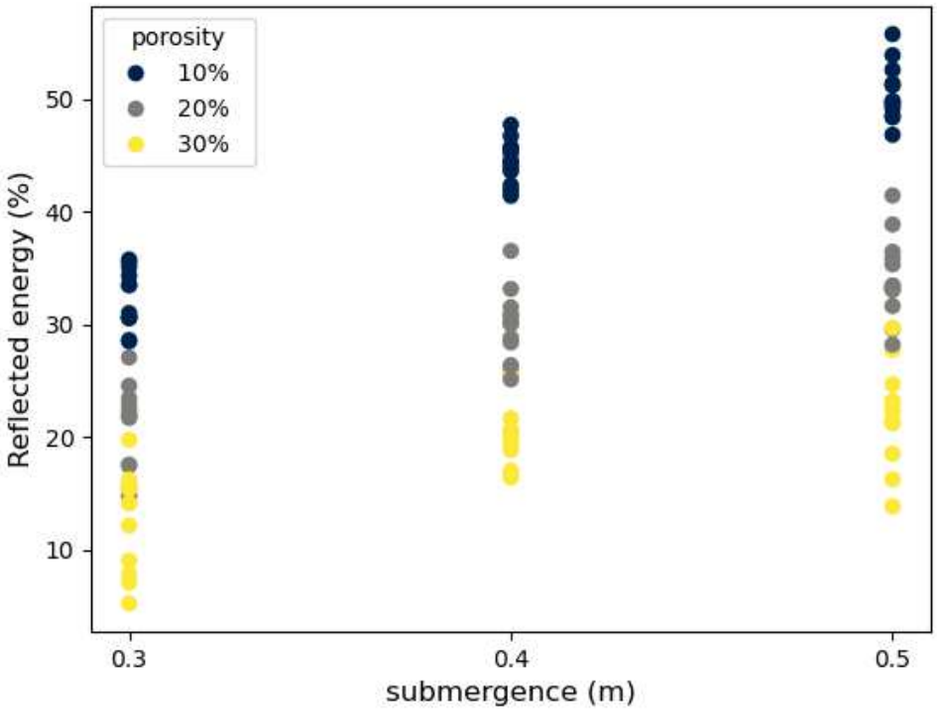


Figure 3.13 Reflection varying with porosity and submergence

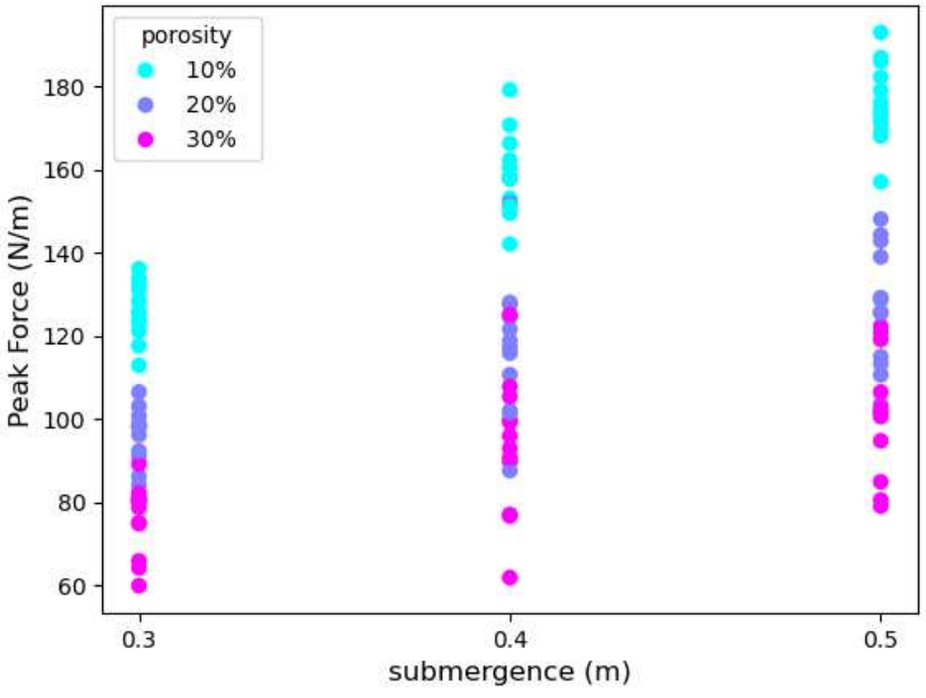


Figure 3.14 Peak forces varying with porosity and submergence

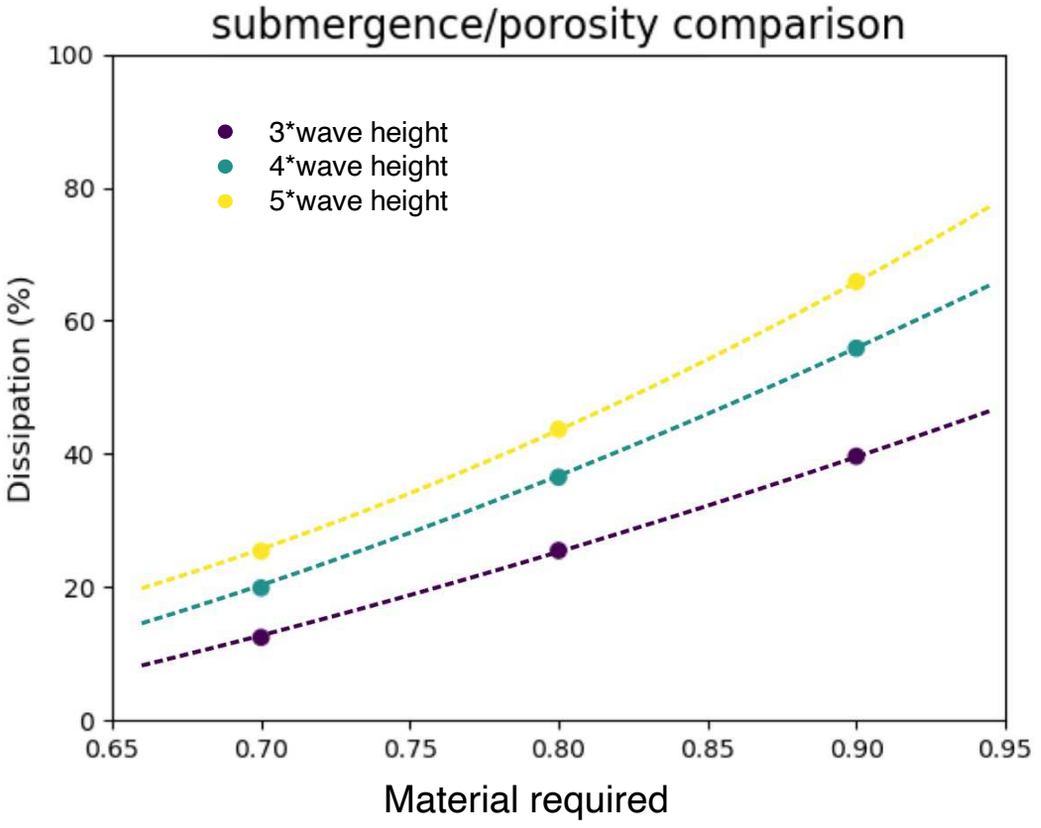


Figure 3.19 porosity-submergence comparison

As shown in the charts, with increasing porosity Forces, Reflection and dissipation reduce. With increasing submergence, Force, Dissipation and Reflection increase.

3.2.4 Slits orientation

Three values were chosen for the slits orientation:

- 0 degrees (vertical);
- 45 degrees;
- 90 degrees (horizontal).

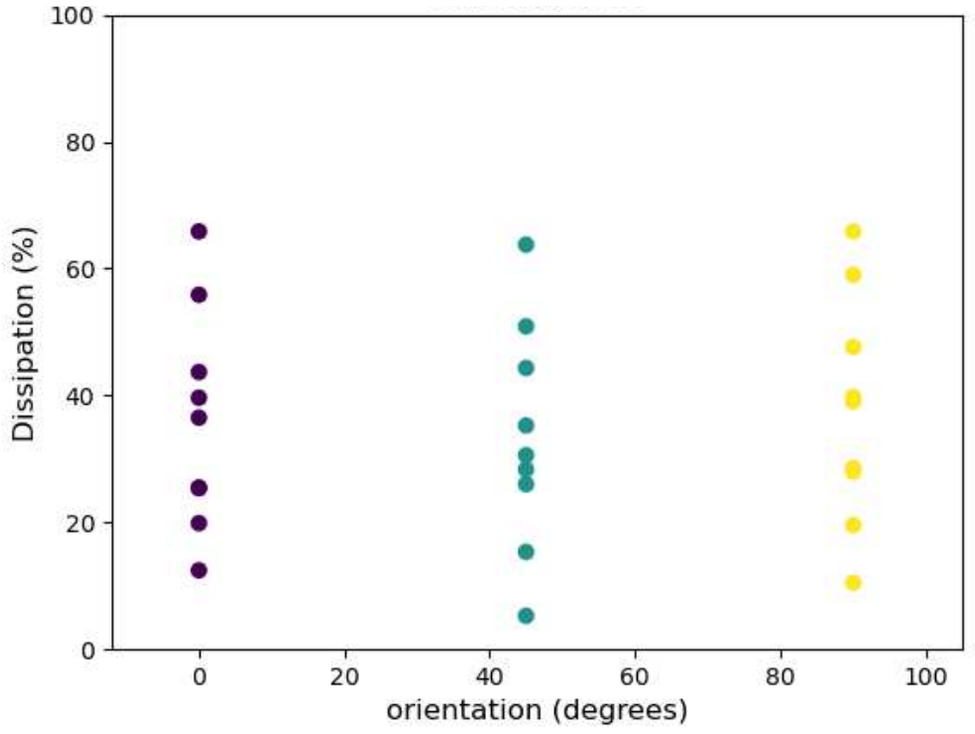


Figure 3.15 Dissipation and Peak forces comparison with varying Slits Orientation

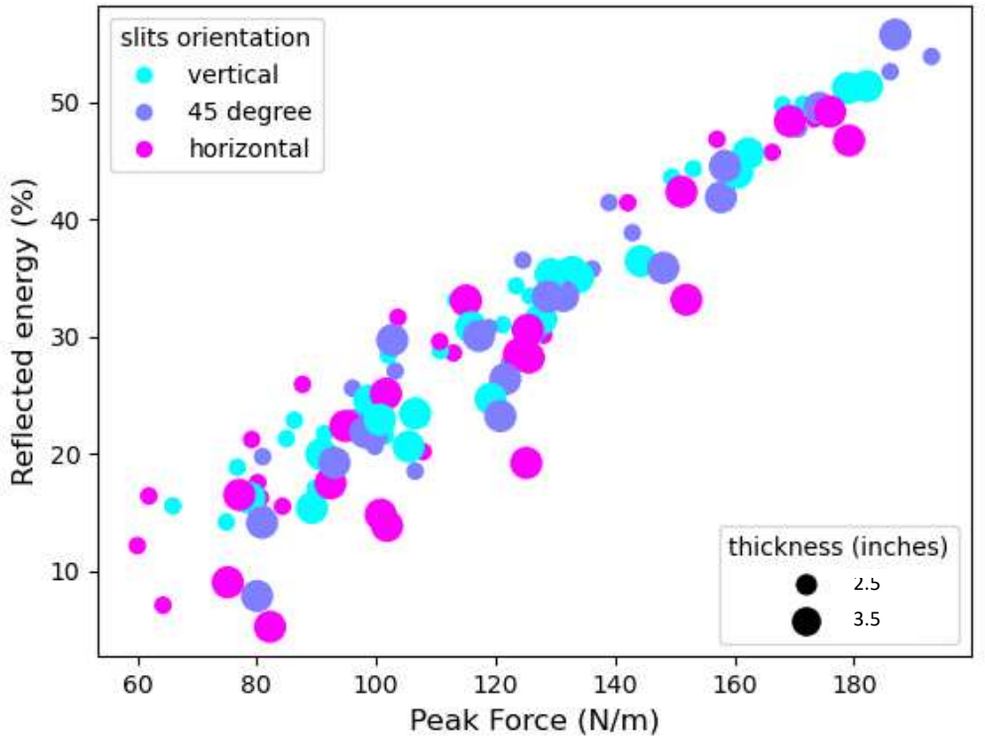


Figure 3.16 Reflection and Peak forces comparison with varying Slits Orientation

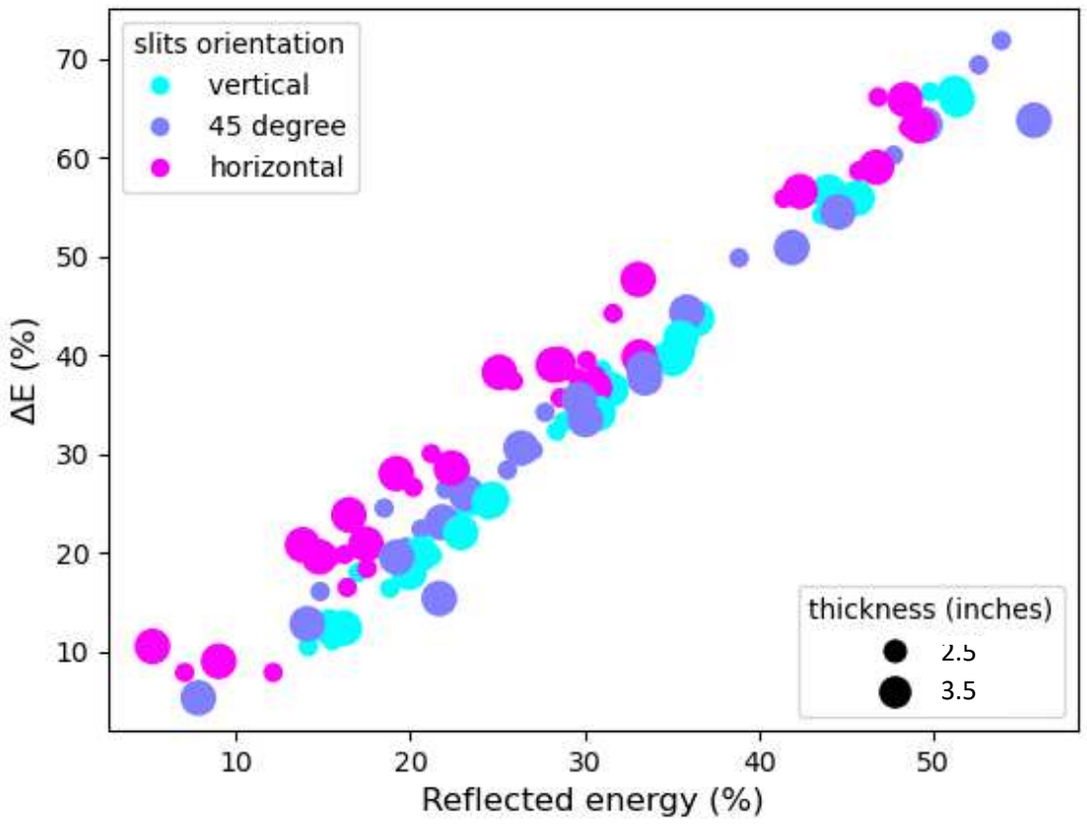


Figure 3.17 Dissipation and Reflection comparison with varying Slits Orientation

Vertical slits show a slightly better overall behaviour, but the dissipation peaks are reached by the 45 degrees case. These orientation shows an increment in Force, though.

3.2.5 Comparison with Borda-Carnot theory

A comparison with borda theory was made to explain the physical process that occurs. All the configuration with 2.5 inches fixed thickness are analyzed and compared in function of porosity

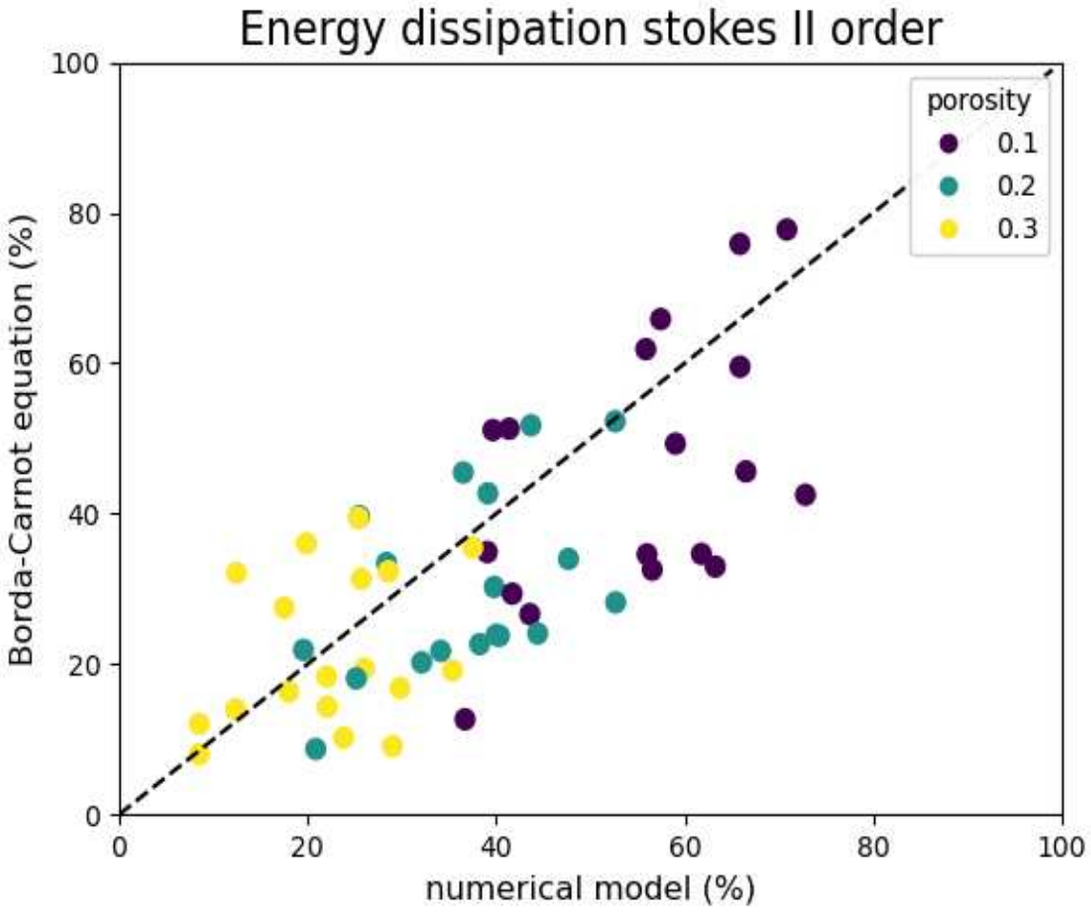


Figure 3.18 Borda Carnot theory – model comparison

As shown in the plot, the Borda-Carnot theory explains well the phenomena that occurs in the numerical model. There are some differences, due to the main hypothesis of the theory (particularly the steady flow one), which are not respected by the case analyzed in this thesis.

Chapter 4: Conclusions and Further studies

4.1 Conclusions

With respect to the results shown in this thesis, we can conclude that:

- Geometrical features (Inclination, Number of slits) do not affect significantly the overall dissipation provided by the screen;
- Submergence and porosity are the most effective parameters to consider;
- Dissipation, Reflection and Forces increase with lower porosity values;
- Dissipation, Reflection and Forces increase with submergence;
- Vertical slits have a general better efficiency overall. Inclining the slits provides different flow features due to the different vortex shapes;
- Porous wavescreens provide up to 70% wave energy dissipation;
- Borda theory is a solid explanation of the phenomena that take part in the wave dissipation process.

4.2 Further studies

This study leads to:

- Implement an analytical solution of the wavescreen dissipation, useful for practical purposes;
- Study the interaction between waves and multiple wavescreens, with multiple configurations and find;
- Find the optimal combination overall combining market prices, optimized volume and shape;
- Implement the tide and tidal current in the model;
- Numerical study on the different wave fields and scenarios (deep/shallow water);
- Study how do waves non-linearities influence wave dissipation and flow features.

List of figures

References

- *The dynamic of the Upper Ocean,* O.M. Phillips, 1966, The Cambridge University press.
- *Volume of Fluid (VOF) Method for the Dynamics of Free Boundaries* (C.W. Hirt and B.D. Nicholson, Journal of Computational Physics 39, 201-225, 1981)
- *Water Wave mechanics for engineers and scientists,* R. G. Dean, R. A. Dalrymple, 1984, World Scientific.
- *Vertical wave barrier: Wave transmission and wave forces* (David L. Kriebe, Coastal Engineering chapt. 100, 1992)
- *Innovative Wave Damping Structures,* H. Oumeraci, German-Chinese Joint Symposium on Coastal and Ocean Engineering (Joint 2002), Rostock, Germany.
- *Second Order analysis of dynamic pressure profiles, using horizontal measured wave velocity component,* F.T. Pinto & A.C.V Neves, Transaction on the Built environment, 2003.
- *Hydraulic Performance of a submerged wave absorber for Coastal Protection,* H. Oumeraci , G. Koether, Advanced in Coastal Engineering, World Scientific, 2008.

- *Wave Scattering and Beach Morphodynamics behind a Permeable Curtain Wall-Pile Breakwater of Finite Length* (J.Y. Le et al. 2009, Journal of Coastal Research,2009,Special Issue No. 56. Proceedings of the 10th International Coastal Symposium ICS 2009, Vol. II (2009), pp. 995-999, Coastal Education & Research Foundation)
- *Navier-Stokes-Brinkman system for interaction of viscous waves with a submerged porous structure* (L.Guta, S. Sundar, Coastal Engineering, 2010)
- *A pimpleFoam tutorial for channel flow, with respect to different LES models* (O. Penttinen, CFD with OpenSource software, a course at Chalmers University of Technology Taught by Hakan Nilsson 2011)
- *Effect of the Slope on gradient of Submerged Breakwaters on Wave Energy Dissipation*, Dong-Soo Hur, Kwang-Ho Lee & Dong-Seok Choi (2011), Engineering Applications of Computational Fluid Mechanics, 5:1, 83-98.
- *Three-dimensional interaction of waves and porous coastal structures Part II: Experimental validation*, Javier L. Lara et al., Coastal Engineering 64, 2012.
- *Numerical Modelling of Regular Wave Propagation Using OpenFOAM* (B. Chenari, 2015, Ocean Engineering 108)

- *Application of a buoyancy-modified k-ω SST turbulence model to simulate wave run-up around a monopile subjected to regular waves using OpenFOAM*, B. Devolder et al., Coastal Engineering 170, 2017.
- *Experimental investigation on non-breaking wave forces and overtopping at the recurved parapets of vertical breakwaters*, L. Martinelli et al., Ocean Engineering 141 52-67, 2018.
- *Performance of a buoyancy-modified k-ω and k-ω SST turbulence model for simulating wave breaking under regular waves using OpenFOAM* (B. Devolder et al., 2018, Coastal Engineering)
- *Liquid sloshing in a rectangular tank with vertical slotted porous screen: Based on analytical, numerical, and experimental approach*, S. K. Poguluri et al., Ocean Engineering 189 (2019).
- *Wave transmission through multilayered porous breakwater under regular and irregular incident waves* (S. Koley, Coastal Engineering, 2019)
- *Modified Goda Equations to Predict Pressure Distribution and Horizontal Forces for Design of Elevated Coastal Structure*, T. Tomiczek et al., 2019, American Society of Civil Engineers, Ocean Engineering.

- *Numerical modeling of the interactions between waves and a Jarlan-type caisson breakwater using OpenFOAM*, Dong-xuWang et al., Ocean engineering 188, 2019).
- *comparison of standard k-ε and SST k-ω turbulence model for breast shot water wheel simulation,* D. Adantal et al., Journal of Mechanical Science and Engineering, 2020.
- *Modeling of solitary wave interaction with emerged porous breakwater using PLIC-VOF method* (S. Booshi et al., Ocean Engineering 241, 2021)