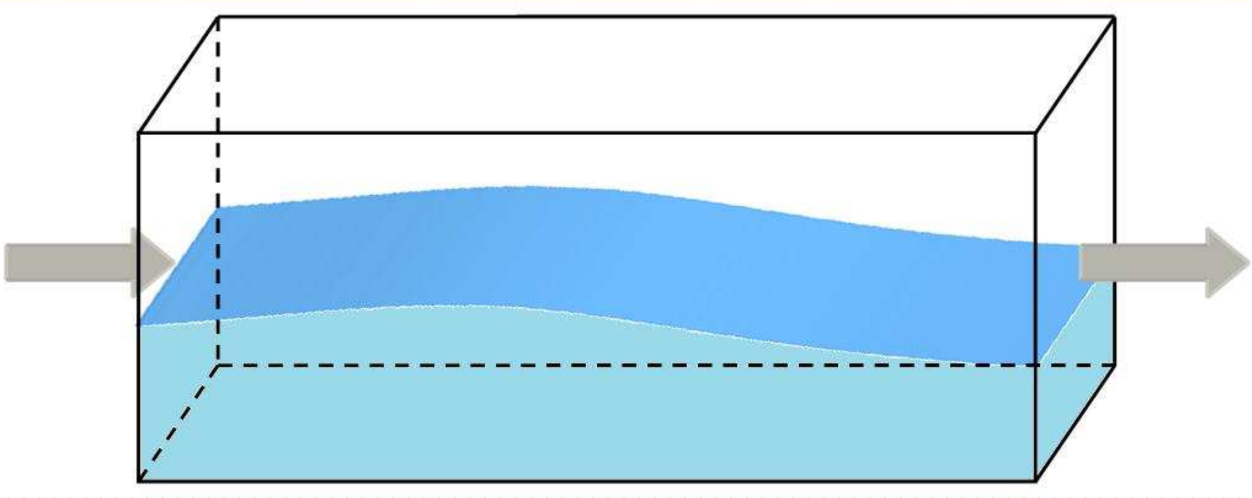


Rosebud Jasmine Lambert

DEVELOPMENT OF A NUMERICAL WAVE TANK USING OpenFOAM

Janeiro, 2012

· U C ·

UNIVERSIDADE DE COIMBRA

FCTUC FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Development of a numerical wave tank using OpenFOAM

Dissertação apresentada para a obtenção do grau de Mestrado em Energia para a Sustentabilidade

Autor

Rosebud Jasmine Lambert

Orientador

Prof. Almerindo Domingues Ferreira

Júri

Presidente	**Professor Doutor Luís Dias**
	Professor Doutor António Manuel Gameiro Lopes

Coimbra, Janeiro, 2012

Acknowledgements

Thank you to Prof. Almerindo Ferreira, my supervisor, who not only guided my work, but also was always willing to listen and offer advice, and more importantly, to give me perspective.

Thank you to all the contributors to the cfd-online.com discussion forums who gladly offer advice to less experienced users of OpenFOAM.

Many thanks must be given to Prof. Benoit, and his co-authors, who gladly provided me access to the experimental data that allowed me to perform a validation of my numerical wave tank. Thank you also to Bin-bin Zhao, who provided advice on how to obtain the experimental data.

Last, but not least, I would like to thank my family and friends, both near and far, for their understanding, support, and faith in my ability to deliver this work.

Resumo

A modelação numérica constitui uma ferramenta fundamental para o desenvolvimento da engenharia do mar e de dispositivos conversores da energia das ondas. Nesta tese demonstra-se que o software *OpenFOAM* tem potencial para ser usado na modelação de ondas num tanque, passo preliminar fundamental à modelação futura de objectos flutuantes.

Neste trabalho o software de dinâmica de fluídos computacional *OpenFOAM* é utilizado para modelar ondas computacionalmente. São geradas ondas regulares à entrada de um tanque com recurso às equações de segunda ordem de Stokes. A onda resultante é comparada com valores experimentais reais, em diversos locais ao longo do tanque, obtendo-se uma boa concordância geral. As ondas reais são geradas para o caso em que existe um obstáculo colocado no fundo do canal. Os resultados das simulações apresentam uma boa concordância com os dados experimentais, em especial na zona a montante do obstáculo. Na zona de jusante, a precisão é inferior devido à produção de harmónicos elevados. Constata-se ainda que o software *OpenFOAM* não permite simular ondas regulares com um declive H/L acima de 0.05.

A simulação dinâmica de ondas mostra que é possível modelar diferentes tipos de ondas (spilling, plunging e surging breaking waves) sobre uma superfície inclinada, e constata-se que a extensão da zona de espraiamento simulada coincide com a previsão teórica. Mostra-se ainda que o software *OpenFOAM* possui capacidades para simular objectos flutuantes que interagem com as ondas geradas através da simulação de um caso simplificado.

Palavras-chave: *OpenFOAM*, simulação numérica de ondas, ondas regulares, comprimento de espraiamento; formas de ondas; objecto flutuante

Abstract

Numerical modelling has become a valuable tool for the ocean enginneering and wave energy industries. This thesis demonstrates that OpenFOAM has the potential to be used to model the formation and propagation of waves, and a floating coastal structure or wave energy device.

In this work a numerical wave tank is developed using the computational fluid dynamic software OpenFOAM. Regular waves are generated at the inlet of the wave tank according to the Stokes second order theory. The resulting wave tank is verified against experimental data of regular waves propagating over a submerged bar. The simulation is shown to replicate the experimental values within a good degree of accuracy, although higher harmonic waves released after the submerged bar lead to minor disagreement in results after the submerged bar. In addition to these conclusions, it is found that OpenFOAM is unable to simulate regular waves with a steepness *H/L* above 0.05.

The numerical wave tank is then shown to be able to simulate spilling, plunging and surging breaking waves over a sloped surface, with simulated run-up agreeing with the theoretical run-up range. OpenFOAM is also shown to be able to simulate a floating object that moves in response to regular waves.

Keywords OpenFOAM, numerical wave tank, regular waves, breaking wave, wave run-up, floating object

Contents

LIST OF FIGURES

LIST OF TABLES

SYMBOLS AND ACRONYMS

Symbols

C – Sea state parameter	
g - Acceleration of gravity	[m s^{-2}]
h - Average water depth	[m]
H – Wave height	[m]
H_0 – Deepwater wave height	[m]
k – Wave number	[radian m^{-1}]
L – Wave length	[m]
L_0 – Deepwater wavelength	[m]
p – Pressure	[Pa]
R – Wave run-up	[m]
$R_{2\%}$ - Wave run-up exceeded 2% of the time	[m]
t – Time	[s]
T – Period	[s]
u – Velocity component of x axis	[m s^{-1}]
U – Velocity field (u, v, w)	
v – Velocity component of y axis	[m s^{-1}]
w – Velocity component of z axis	[m s^{-1}]
x – Distance along x-axis	[m]
z – Coordinate axis to describe wave motion	[m]
α – Volume fraction of water	
β – Angle of sloped wall of numerical wave tank	[degrees]
ξ_0 – Surf similarity parameter	
σ - Wave frequency	[radian s^{-1}]
ρ – Density	[kg m^{-3}]
ρ_a – Density of air	1.2 kg m^{-3}

ρ_w – Density of water	1000 kg m^{-3}
η - Wave surface elevation	[m]
μ – Dynamic viscosity	[Pa s]
φ – Velocity potential	

Acronyms

CFD – Computational Fluid Dynamics

ENONDAS – Energia das Ondas Sociedade Anonima

IEA – International Energy Agency

NWT – Numerical Wave Tanks

OpenFOAM – Open Source Field Operation and Manipulation

PCG – Preconditioned Conjugate Gradient

PISO – Pressure Implicit Split Operator

STL – Stereolithography

SWL – Still Water Level

VOF – Volume of Fluid method

1. INTRODUCTION

1.1. Goals and Objectives

Numerical models are a valuable tool for the coastal and ocean engineering community, allowing the simulation and determination of forces and wave actions before physical construction takes place. This work aims to demonstrate that numerical models could be used to aid development of marine renewable energy technologies, such as wave energy, by providing a means to simulate regular and breaking waves, as well as floating objects under wave action.

The main goal of this work is to develop a numerical model, known as a numerical wave tank, which can replicate the behaviour of waves in an experimental wave tank. This numerical wave tank will be created within a computational fluid dynamics (CFD) software known as OpenFOAM (version 1.7.1).

While numerical wave tanks have been created previously in OpenFOAM (Yong & Mian, 2010; Morgan et al., 2010; Afshar, 2010) this Master thesis will demonstrate that not only is OpenFOAM able to produce a numerical wave tank that can closely replicate experimental results but that the numerical wave tank can also correctly predict the nature of breaking waves.

Within the primary objective of creating a numerical wave tank there are several sub-objectives. The first sub-objective is to produce regular waves within OpenFOAM by utilising a periodic boundary condition at the inlet of the wave tank. Secondly, the behaviour of waves within this numerical wave tank is validated against the analytical results of the implemented wave theory and also against experimental data measured by Dingemans (1994).

Once the numerical wave tank is validated the capabilities of OpenFOAM are then demonstrated by showing that OpenFOAM is capable of simulating breaking waves on a sloped beach, including wave run-up, and correctly predicting the type of breaking wave. A final demonstration is given of a floating object subject to wave motion, with the use of a dynamic mesh.

1.2. Motivation

The world's increasing demand for energy and fossil fuels has led to a search for more sustainable technologies than conventional fuels such as coal or crude oil. Less polluting renewable energy technologies are playing an increasing role in the world's energy mix due to this reason. While technologies such as photovoltaics and wind energy have been successfully commercialized, they are unable to meet the world's energy needs on their own. This has led to an interest in other renewable energy technologies such as wave energy.

In recent years there has been an increasing commercial and academic interest in wave energy technology. Wave energy has a high theoretical potential with an estimated 8000-80,000 TWh per annum (Bhuyan, 2008). This high potential can be attributed to strong winds that occur between 30 and 60° latitude and the occurrence of powerful storms in the southern latitudes that cause high energy waves (Bhuyan, 2008).

The high theoretical potential for wave energy has led to the development of countless designs and prototypes of wave energy converters, with no single technology yet to emerge as the market leader. Yet no matter what design wave energy devices have, they must all be able to survive the tough conditions of the marine environment. Robustness of components and survivability against the power of the ocean has proved difficult to achieve.

The difficulty of installing reliable wave energy devices was recently demonstrated off the coast of Portugal, which has an estimated overall resource of 10 GW, with half of that potentially available for exploitation (Mollison & Pontes, 1992). In 2008 Portugal became the first country in the world to host an experimental wave farm located north of Porto.[1] The Aguçadoura Wave Farm consisted of three 750 kW Pelamis wave energy converters, each 120 m long and grid connected to a substation at Aguçadoura. A photograph of the Pelamis wave energy converter in operation is shown in Figure 1.

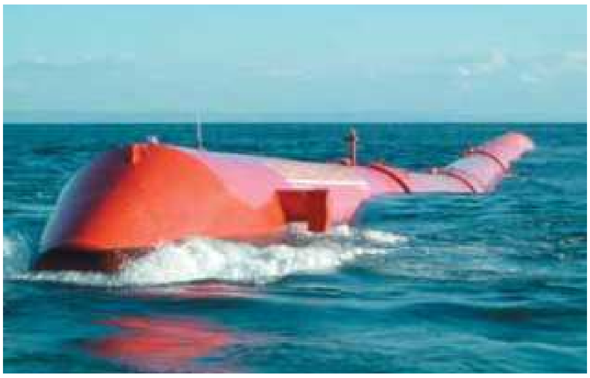


Figure 1. A Pelamis wave energy converter in the water (Pelamis Wave Power, 2011a)

Unfortunately, the Pelamis devices were removed only months after the opening of the wave farm due to damage from large waves caused by a storm (Beirão, 2010). The Pelamis wave energy converters have until the present moment not been re-installed at Aguçadoura. A second-generation model of the Pelamis, the P2, was recently installed close to the Orkney Islands in Scotland (Pelamis Wave Power, 2011b).

In recent years, strong governmental support of renewable energy in Portugal has also led to the introduction of legislation that promotes wave energy. Decree Law 225/2007 of 31 May 2007 (Ministério da Economia e da Inovação, 2007) specifies feed-in tariffs for renewable energy technologies, including wave energy. The feed-in tariff for demonstration wave projects (up to 4 MW) is approximately 0.26 €/kWh. Tariffs also exist for pre-commercial and commercial wave energy projects.

In addition to the feed-in tariff the Portuguese government also introduced Decree Law 5/2008 (Ministério da Defesa Nacional, 2008) which establishes a pilot zone for testing wave energy devices off the coast of Portugal. The pilot zone aims to attract

[1] Readers wishing to learn more about the current state of wave energy in Portugal are encouraged to read the latest Annual Report of the IEA Implementing Agreement on Ocean Energy Systems. At the time of writing the most recent report was the 2010 Annual Report (Brito-Melo & Huckerby, 2011)

demonstration and industrial wave energy projects to Portugal. The 261 km^2 pilot zone is located off the coast of São Pedro de Moel, situated between Peniche and Figueira da Foz (see Figure 2). The site was chosen for its proximity to a suitable electricity grid (allowing connection to the grid), suitable bathymetry and electricity generating potential of up to 10 TWh/y (where there is a depth of 50 m) (Brito, 2009). In 2010 a company called ENONDAS (Energia das Ondas Sociedade Anonima) was created to manage the Pilot Zone.

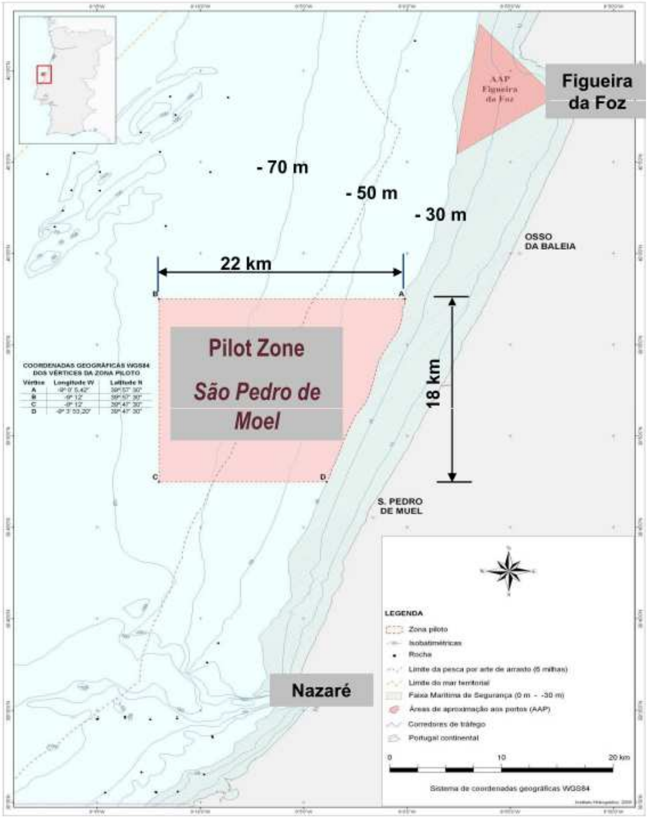

AAP Figueira da Foz
Figueira da Foz
- 70 m
- 50 m
- 30 m
OSSO DA BALEIA
22 km
A
B
C
D
18 km
Pilot Zone
São Pedro de Moel
COORDENADAS GEOGRÁFICAS WGS84
DOS VÉRTICES DA ZONA PILOTO
Vértice Longitude W Latitude N
A -9° 0' 5,42" 39° 57' 30"
B -9° 12' 39° 57' 30"
C -9° 12' 39° 47' 30"
D -9° 3' 53,20" 39° 47' 30"
S. PEDRO DE MUEL
Nazaré
LEGENDA
Zona piloto
Isobatimétricas
Rocha
Limite da pesca por arte de arrasto (6 milhas)
Limite do mar territorial
Faixa Marítima de Segurança (0 m - -30 m)
Áreas de aproximação aos portos (AAP)
Corredores de tráfego
Portugal continental
0
10
20 km
Sistema de coordenadas geográficas WGS84

Figure 2. The São Pedro de Moel Pilot Zone (Brito, 2009)

This Master thesis seeks to contribute to the field of wave energy by producing a numerical wave tank that closely replicates the behaviour of how waves interact with the seabed and demonstrates how a wave energy device behaves under the influence of waves. The development of a numerical wave tank using freely available open-source software (in this case OpenFOAM) demonstrates one possible method in which wave energy converters may be tested in the future before reaching the prototype stage, potentially preventing device failures, as demonstrated by the Pelamis. Given the positive governmental support of wave energy in Portugal, and the high theoretical potential of wave energy along the Portuguese coastline it is hoped that this Master thesis will be an illustration of how OpenFOAM may be used to assist in the development of wave energy converters.

1.3. Outline of the thesis

This dissertation is divided into seven chapters. The first and current chapter outlines the objectives and motivation underlying the work.

The second chapter introduces literature related to numerical wave tank research and outlines the experiments that are used to validate the developed numerical wave tank.

The third chapter summarizes OpenFOAM, the open source software used to develop the numerical wave tank. Governing equations of the source code are presented as well as an outline of how the program functions.

Chapter four presents the governing wave theory employed for this work, including Stokes theory, employed to generate the waves at the inlet. The cause of breaking waves and wave run-up is also discussed and the various types of breaking waves are introduced.

Chapter five presents the methodology used to create the numerical wave tank. The modelled geometry is presented, along with simulation parameters, wave parameters and detailed discussion of creating the inlet boundary condition and preventing reflection of waves from the outlet. Mesh independency is also discussed in this chapter.

Chapter six presents the results of the various simulations outlined in chapter five. Analytical and experimental validation of the numerical wave tank is presented as well as the limitations of OpenFOAM discovered during the work. The simulation and results of breaking waves is presented as well as the results of the demonstration of a floating object under the influence of regular waves.

Chapter seven concludes the written work with a summary of how the thesis objectives have been met and suggestions for future work.

2. AN INTRODUCTION TO NUMERICAL WAVE TANKS

The accurate modelling of the behaviour of water waves is an important subject for the field of coastal and ocean engineering. As computational power has increased numerical models, and numerical wave tanks (NWT), have become an increasingly viable option for the modelling of surface gravity water waves.

Numerical wave tanks can be achieved through the creation of a numerical model or with an existing program, such as OpenFOAM, as was done for this Master thesis. The use of an existing program such as OpenFOAM is arguably more accessible for working professionals and less time consuming compared to the creation of a new numerical model.

Numerical models[2] typically implement one of two types of equations to model the hydrodynamics of waves, Stokes theory (discussed in detail in 4.1) and Boussinesq-type equations. Stokes equations can be appropriate for a variety of depths while Boussinesq-type equations are used for shallow water. Boussinesq-type equations are more complex and difficult to implement than Stokes theory.

Numerical models based on Boussinesq-type equations face some limitations as they cannot model the breaking of waves without additional modification to model energy dissipation (Orszaghova et al., 2012) and the largest wave height that can be accurately modelled is limited (Chazel et al., 2010). Some attempts have been made to expand the applicability of Boussinesq based models by creating hybrid numerical models. Orszaghova et al. (2012) developed a hybrid numerical model, based on Boussinesq equations that are capable of simulating breaking and non-breaking waves by applying a different set of equations for pre- and post-breaking. Chazel et al. (2010) and Bai &

[2] Only numerical wave tanks implementing regular waves are considered within this literature review. There are numerous studies that implement solitary waves but the production of solitary waves was outside the scope of this thesis.

Cheung (2011) have both employed a two-layer approach, solving for two layers of fluid, reducing the complexity of the Boussinesq-type equations.

Stokes theory is easier than Boussinesq-type equations to implement numerically and has previously been successfully implemented in OpenFOAM to create a numerical wave tank (Afshar, 2010; Morgan et al., 2010; Yong & Mian, 2010). Afshar (2010) focussed on the calculation of the error of the wave tank's ability to produce Stokes second order waves but was unable to validate his wave tank against experimental results. Yong & Mian (2010) aimed to model a floating object using OpenFOAM and validated their work by comparing the calculated drift force on the floating object with the force measured in a set of experiments. Morgan et al. (2010) completed the work most similar to this Master thesis by modelling the experimental case of Dingemans (1994) (discussed below) and comparing the surface elevation of the water to experimental results. Stokes second order theory has also been implemented in custom-made numerical models such as Senturk (2011) and Koo & Kim (2007).

Regardless of the underlying equations of the numerical wave tank, it is important to validate the results. This can be done by comparison with the analytical (theoretical) results, comparison with other numerical model results, comparison against experimental data or a mixture of these methods. Comparison against experimental results gives the most accurate indication of how well the wave tank can simulate physical conditions.

Experiments are rarely conducted by the developer of the numerical wave tank, with developers usually relying on existing experimental results. Three experimental sets of data have been the most commonly referenced in the literature related to numerical wave tanks. Yong & Mian (2010) and Koo & Kim (2007) validated their results of the force on a floating object in a numerical wave tank against the experiments of Nojiri & Murayama (1975). A more common method of validation is to compare the surface elevation of the waves in a numerical wave tank containing a submerged bar. This tests the ability of the wave tank to model higher harmonic waves that are released after the bar. Two sets of such experimental data have been commonly referenced, Dingemans (1994) and Ohyama et al. (1995). The Dingemans (1994) experiments are considered a classical set of experiments that are routinely referenced (Chazel et al., 2010; Zhao & Duan 2010; Bai & Cheung, 2011; Morgan et al., 2010). Unlike most authors, Morgan et al. (2010) used the unscaled results of Dingemans (1994).

The Dingemans (1994) experiments, also referenced as Luth et al. (1994), were based on experiments first performed by Beji & Battjes (1993). Dingemans (1994) repeated the experiments at twice the scale of Beji & Battjes (1993) but the results are often presented at the scale of the experiments conducted by Beji & Battjes (1993). In this present work all geometry, wave parameters and results are modelled and presented at the scale of Beji & Battjes (1993).

Three cases were modelled by Beji & Battjes (1993) and Dingemans (1994), and are given in Table 1. Only Case A is presented and discussed in this Master thesis.

Table 1. Wave parameters used in the Dingemans (1994) experiments, given in scale of Beji & Battjes (1993)

Case	**Period *T* [s]**	**Wave Height *H* [m]**	**Wave Length *L* [m]**
A	2.02	0.02	3.738
B	2.525	0.029	4.791
C	1.01	0.041	1.488

This Master thesis builds on the work of previous authors such as Morgan et al. (2010) and Afshar (2010) but also presents detail about the limitations of using OpenFOAM to create a numerical wave tank. The Dingemans (1994) experiments are used to validate the numerical wave tank. The ability of OpenFOAM to model breaking waves is also investigated.

3. OPENFOAM

3.1. Introduction to OpenFOAM

The OpenFOAM (Open Source Field Operation and Manipulation) software is an open source computational fluid dynamic software that was first released in 2004. OpenFOAM is essentially a C++ library that is used to create *applications*. Applications can be *solvers* or *utilities*. *Solvers* are designed to solve a specific physical problem in continuum mechanics and *utilities* are used to perform tasks that involve data manipulation (OpenFOAM, 2010). OpenFOAM comes pre-equipped with many solvers and utilities.

OpenFOAM comes with a large number of preset solvers but the open source nature of OpenFOAM also means that the user can write their own solvers, although a solid understanding of the physics and underlying method of the problem is needed. While OpenFOAM lacks a graphical user interface, the customisable nature of the software has made it a popular choice for users wishing to have a degree of control over the physics and calculation of a solution to a problem. Users of OpenFOAM often make their custom solvers and utilities available to others. OpenFOAM is used by many commercial and academic organisations and has been used in many peer-reviewed papers.

OpenFOAM release version 1.7.1 for the Ubuntu operating system was used for the work of this Master thesis.

3.2. Governing equations

3.2.1. Navier-Stokes equations

The fundamental equations used by OpenFOAM are the Navier-Stokes equations for an incompressible, constant viscosity fluid. In Cartesian coordinates these equations are:

$$\rho\left(\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+w\frac{\partial u}{\partial z}\right)=-\frac{\partial p}{\partial x}+\mu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right)+\rho g_x \qquad (1)$$

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

$$\rho\left(\frac{\partial w}{\partial t}+u\frac{\partial w}{\partial x}+v\frac{\partial w}{\partial y}+w\frac{\partial w}{\partial z}\right)=-\frac{\partial p}{\partial z}+\mu\left(\frac{\partial^2 w}{\partial x^2}+\frac{\partial^2 w}{\partial y^2}+\frac{\partial^2 w}{\partial z^2}\right)+\rho g_z \qquad (3)$$

where ρ is the density of the fluid mixture [kg m^{-3}], p is the pressure [Pa], g is the acceleraration of gravity [m s^{-2}], μ is the fluid dynamic viscosity [Pa s] and u, v, and w are the velocity components of the x, y and z axes, respectively, while t represents the time.

Because the flow is assumed as incompressible, ρ is constant and the following form of the continuity equation must be satisfied:

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}=0 \qquad (4)$$

Together with the boundary conditions (described in 5.3.4), Equations (1) - (4) describe the motion of an incompressible viscous fluid flow.

3.2.2. Volume of Fluid method

OpenFOAM uses the Volume of Fluid method (VOF) to track the movement of the *free surface* (the air-water interface). This method determines the fraction of each fluid that exists in each cell of the computation mesh (known as the *volume fraction*). The equation for the volume fraction is:

$$\frac{\partial \alpha}{\partial t} + \nabla .(\alpha U) = 0 \qquad (5)$$

where U is the velocity field composed of u, v, and w and α is the volume fraction of water. α will vary between 0 and 1. If a cell is completely full of water, $\alpha = 1$, if it is full of air then $\alpha = 0$.

The volume fraction (also known as the phase fraction) α is used to determine the density of the mixture inside each cell of the mesh, (the density that is used to solve the Navier-Stokes equations). The density of the mixture is determined by:

$$\rho = \alpha\rho_w + \ (1 - \alpha)\rho_a \qquad (6)$$

where ρ_w is the density of water and ρ_a is the density of the air.

4. GOVERNING WAVE THEORY

4.1. Stokes second order theory

The accurate production of waves at the inlet of the wave tank is integral to accuracy of the wave tank. Many wave equations exist although a review of the literature of numerical wave tanks revealed that Stokes second order equations have been frequently implemented. Stokes theory is a non-linear theory for modelling regular waves.

Higher orders of Stokes theory do exist, such as the fifth order. Due to the added complexity of the fifth order equations, the second order form was implemented. This form has shown to be sufficiently accurate for the scenarios modelled.

Studies utilising Stokes second order equation include Yong & Mian (2010), Koo & Kim (2007) and Senturk (2011). Among these papers, Stokes second order equation is presented in different forms. The form that will be used in this Master thesis is taken from Dean & Dalrymple (1984).

4.1.1. Particle velocity under the wave

The particle velocity according the Stokes second order theory can be broken into horizontal and vertical components, where u represents the particle velocity in the longitudinal direction (Equation (7)) and w represents the particle velocity in the vertical direction (Equation (8)).

$$u = -\frac{\partial \varphi}{\partial x} = \frac{H}{2}\frac{gk}{\sigma}\frac{\cosh k(h+z)}{\cosh kh}\cos(kx - \sigma t) + \frac{3}{16}\frac{H^2 \sigma k \cosh 2k(h+z)}{\sinh^4 kh}\cos 2\,(kx - \sigma t) \tag{7}$$

$$w = -\frac{\partial \varphi}{\partial z} \quad = \frac{H}{2}\frac{gk}{\sigma}\frac{\sinh k(h+z)}{\cosh kh}\sin(kx-\sigma t) + \frac{3}{16}\frac{H^2 \sigma k \sinh 2k(h+z)}{\sinh^4 kh}\sin 2\,(kx-\sigma t) \tag{8}$$

Both u and w are partial derivatives of the velocity potential φ. H is the height of the wave from crest to trough [m], g is the gravitational acceleration [m s^{-2}], h is the average water depth and z is the coordinate axis to describe wave motion [m] (where $z = 0$ is the still water level (SWL)). t is the time [s] and x [m] is the distance along the longitudinal direction. σ is the wave frequency [radian s^{-1}] (determined by Equation (9)) and k is the wave number [radian m^{-1}] (determined with Equation (10) where L is the wavelength [m]):

$$\sigma = \sqrt{gk \tanh kh} \tag{9}$$

$$k = \frac{2\pi}{L} \tag{10}$$

4.1.2. Confirmation of validity of Stokes second order waves

As noted in Dean & Dalrymple (1984), Stokes equation in the second order is not a very good approximation for high waves in shallow water (shallow water is defined as h/L < 1/20, deep water waves are defined as $h/L \geq$ 1/2). To address this issue a simple method, known as the Ursell parameter was developed by Ursell (1953). The Ursell parameter indicates the nonlinearity of long surface gravity waves in a fluid and can be used to determine if Stokes second order theory is valid. The Ursell parameter can be shown to be reduced to Equation (11) as shown by Dean & Dalrymple (1984).

$$\frac{L^2 H}{h^3} < \frac{8\pi^3}{3} \tag{11}$$

If the Ursell parameter is satisfied, it is appropriate to use Stokes equations in the second order. All scenarios modelled in this thesis satisfied the Ursell parameter.

4.1.3. Surface elevation of the waves

For Stokes second order waves the distance the water surface is displaced from the still water level is described by Equation (12) for the surface elevation η (also known as the water surface displacement).

$$\eta = \frac{H}{2}\cos(kx - \sigma t) + \frac{H^2 k}{16}\frac{\cosh kh}{\sinh^3 kh}(2 + \cosh 2kh)\cos 2(kx - \sigma t) \qquad (12)$$

This equation was used to verify if the surface elevation of waves in the numerical wave tank agreed with the theoretical surface elevation for Stokes second order waves (shown in sections 6.1.1, 6.2.1 and 6.2.2).

A graphical summary of the physical wave parameters used in the governing equations is given in Figure 3.

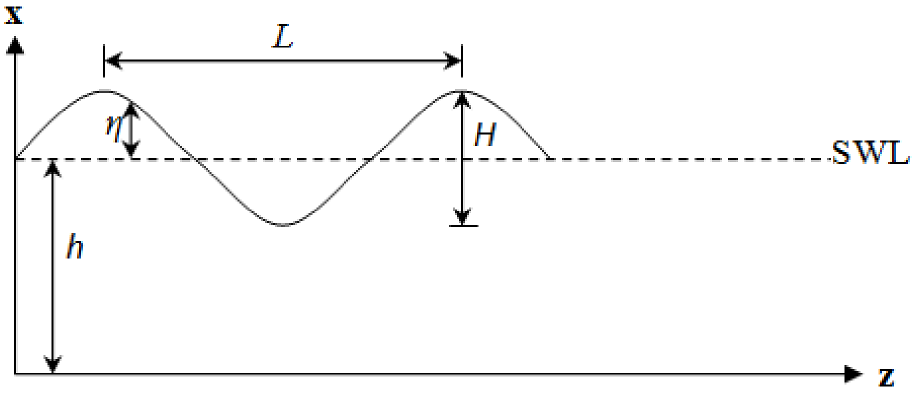


Figure 3. Variables of a wave used in the governing wave theory

4.2. Breaking waves

The breaking of waves can occur in both shallow and deep water although each is due to different mechanisms. In shallow water the change in the water depth as waves approach a shallow region or beach causes shoaling, refraction and diffraction (Vincent et al., 2002). Breaking waves in shallow water are classified into different breaker categories, as discussed in 4.3.1.

In deep water, waves break due to hydrodynamic instability. Fenton (1990) has developed an expression that can be used to predict when a regular wave in deep water will become unstable and break. This expression was based on experiments by Williams (1981) who determined the upper limit of the height/depth ratio (*H/h*). The Fenton expression is given by Eqation (13) but has been transformed to give the upper limit of the wave *steepness*, determined by *H/L*, rather than *H/h*.

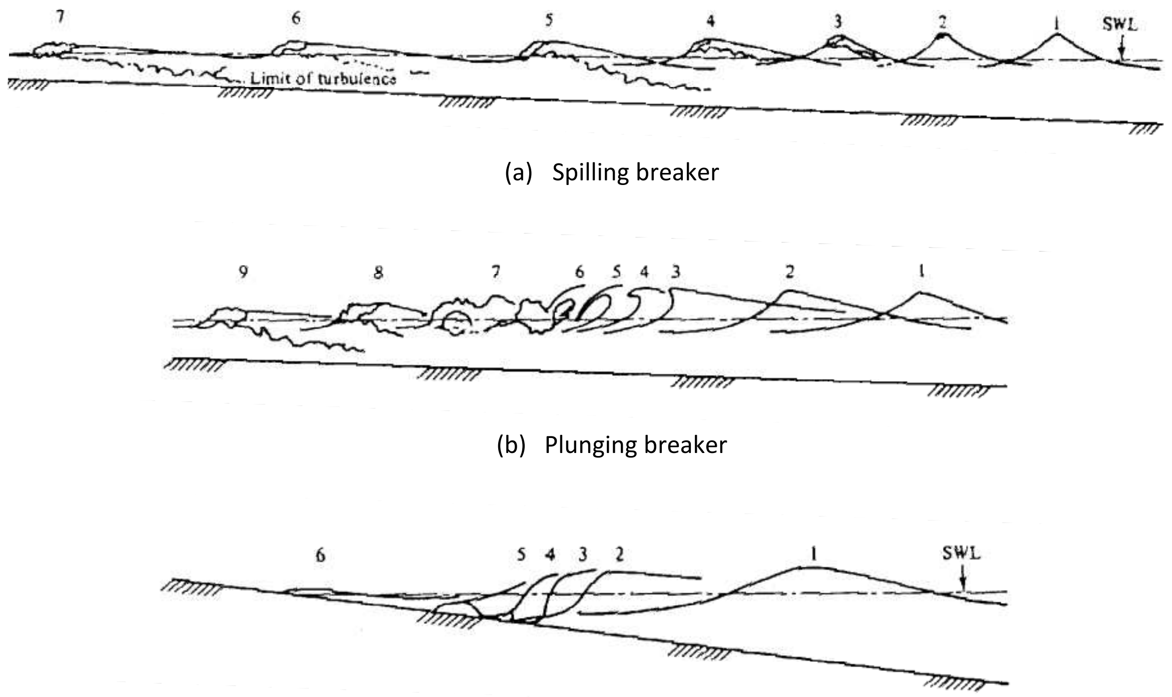
$$\left(\frac{H}{L}\right)_{max} = \frac{0.141063 + 0.0095721\left(\frac{L}{h}\right) + 0.0077829\left(\frac{L}{h}\right)^2}{1 + 0.0788340\left(\frac{L}{h}\right) + \ 0.0317567\left(\frac{L}{h}\right)^2 + 0.0093407\left(\frac{L}{h}\right)^3} \qquad (13)$$

4.3. Waves in the surf zone

In addition to the modelling of Stokes second order waves it will be shown that OpenFOAM is also capable of simulating waves in the surf zone, the region where waves break close to the shore. Using a parameter known as the *surf similarity*, the breaker type can be predicted. This parameter can also be used to predict the wave *run-up,* i.e., how far waves will move up the slope above the still water level.

4.3.1. Breaker types

The breaker type is the form of the wave at the time of breaking. Breaker types can be classified into four different types (Galvin, 1968). A diagram showing three of the types is given in Figure 4.



(a) Spilling breaker

(b) Plunging breaker

(c) Surging breaker

Figure 4. Three breaker types – spilling (a), plunging (b) and surging (c). The numbers indicate the stages of the breaking process. (Richardson, 1996)

Spilling breakers occur on mildly sloping beaches. Breaking begins with aerated water near the top of the wave, which then moves down the front surface. *Plunging breakers* occur on steeper beaches. The crest of the wave curls forward and falls on the base of the wave. *Surging breakers* occur on even steeper beaches. The crest remains unbroken and very little breaking occurs.

A fourth breaker type of wave, the *collapsing breaker*, was also identified by Galvin (1968) and occurs at the water's edge. This breaker is a combination of *plunging* and *surging* breakers (Dean & Dalrymple, 1991) and is identified by a crest that never fully breaks. The lower face of the wave steepens and falls. A photograph of a collapsing wave is shown in Figure 5.

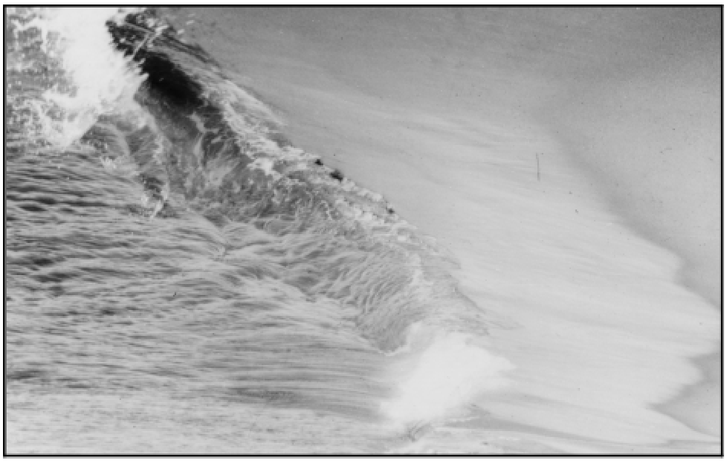


Figure 5. A photograph of a collapsing wave (Smith, 2003)

4.3.2. Surf similarity

The surf similarity parameter ξ_0 (Battjes, 1974a), also known as the breaker parameter or Iribarren number (Iribarren & Nogales, 1949), indicates the breaker type that can be expected. The surf similarity parameter, given by Equation (14), uses the angle β of the beach, the deepwater wave height H_0, deepwater wavelength L_0 or period T:

$$\xi_0 = \frac{\tan\beta}{\sqrt{\frac{H_0}{L_0}}} = \frac{\tan\beta}{\sqrt{\frac{2\pi H_0}{gT^2}}} \tag{14}$$

As discussed by Hughes (2004) it is common to specify the local wave height H at or near the toe of the slope, instead of the deepwater wave height, H_0. For the work of this Master thesis, the local wave height H was used.

The critical values of ξ_0 noted by Battjes (1974a) are given in Table 2.

Table 2. Critical values of the surf similarity parameter, ξ_0, used to predict breaker type (Battjes, 1974a)

Breaker type	Critical value of ξ_0
Surging or collapsing	$\xi_0 > 3.3$
Plunging	$0.5 < \xi_0 < 3.3$
Spilling	$\xi_0 < 0.5$

4.3.3. Wave run-up

Using the surf similarity parameter the wave run-up, can be determined. An early empirical formula for wave run-up was developed by Hunt (1959) and was later modified by Battjes (1974b). The modified formula (Equation (15)) gives the run-up that will only be exceeded 2% of the time.

$$R_{2\%} = HC\xi_0 \tag{15}$$

The parameter C depends on sea conditions, ranging between $C = 1.49$ for fully developed seas, and $C = 1.87$ for young seas (Van der Meer & Stam, 1992). Equation (15) can only be applied to plunging waves (see 4.3.1) according to Van der Meer & Stam (1992). Figure 6 indicates the run-up of a wave, $R,$ impacting a slope with angle β. SWL is the still water level.

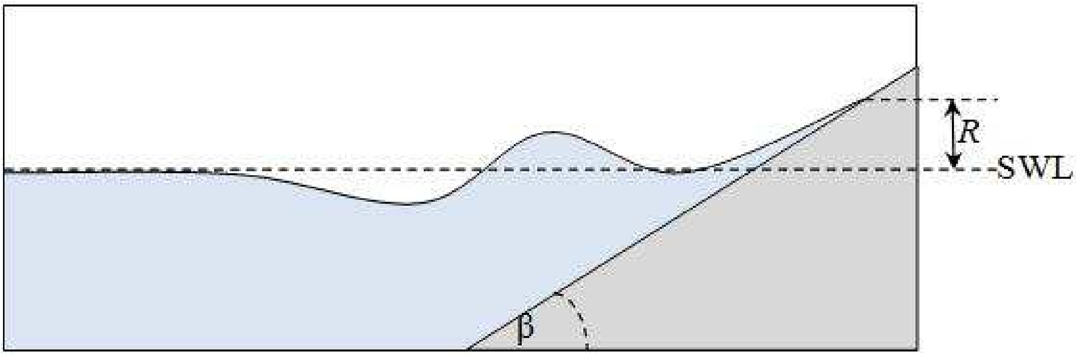


Figure 6. Run-up, *R*, of a wave breaking on a slope with angle β

5. MODELLING METHODOLOGY

5.1. Definition of scenarios and geometries to be modelled

All scenarios modelled in this work were two-dimensional cases. OpenFOAM always operates in three-dimensional coordinates but was instructed to solve for two dimensions (see 5.3). All geometry had a thickness of 0.1 m in the *y* (transverse) direction. The following section describes the geometry of all scenarios modelled.

Four different scenarios were modelled using OpenFOAM:

Scenario 1: A basic numerical wave tank with flat bottom

Scenario 2: The verification tank based on the experiments conducted by Beji & Battjes (1993) and Dingemans (1994)

Scenario 3: A demonstration of regular waves hitting a sloped surface with angle β

Scenario 4: A demonstration of implementing a floating object under the influence of waves

5.1.1. Scenario 1: Basic numerical wave tank

The geometry of the basic numerical wave tank is shown in Figure 7.

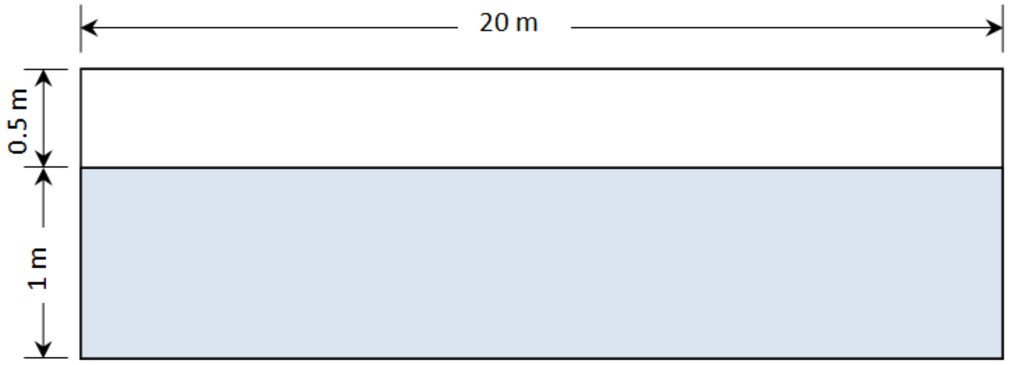


Figure 7. Geometry of the basic numerical wave tank, Scenario 1 (not to scale)

5.1.2. Scenario 2: Verification tank

The geometry of the verification tank based on the scaled experimental results of Dingemans (1994) is shown in Figure 8. The position of the wave gauges (that recorded the surface elevation) are also indicated Figure 8, with the exact position of each gauge given in Table 3. Note that the modelled geometry was to the same scale as the Beji & Battjes (1993) experiment. The elevation of the bed is described in Table 4.

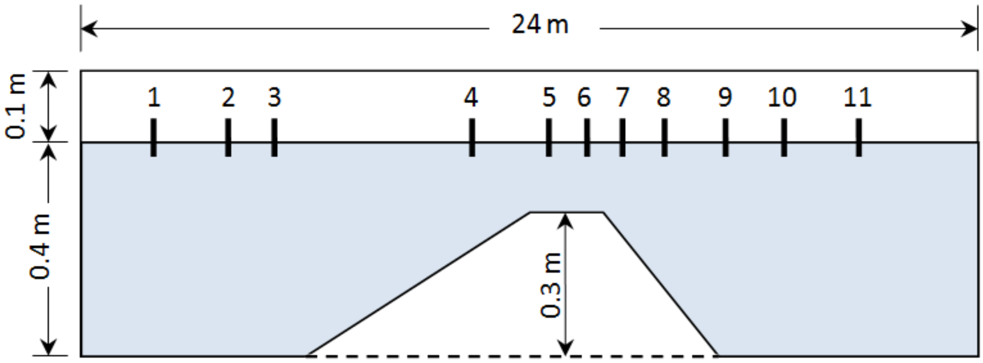


Figure 8. Geometry of the Scenario 2 verification tank including position of the wave gauges (vertical axis not to scale)

Table 3. Position of wave gauges for verification wave tank, as used in the Dingemans (1994) experiments

Wave gauge number	*x* position [m]
1	2
2	4
3	5.2
4	10.5
5	12.5
6	13.5
7	14.5
8	15.7
9	17.3
10	19
11	21

Table 4. Bed elevation of verification tank (Scenario 2)

x distance [m]	*z* distance [m]
0	-0.4
6	-0.4
12	-0.1
14	-0.1
17	-0.4

Basic verification of this tank was conducted by removing the submerged bar at the bottom of the tank and comparing the simulated results to the theoretical results of the governing wave equations (see 6.2.1).

5.1.3. Scenario 3: Sloped tank

Within Scenario 3 three different cases were modelled to simulate the three types of breaking waves (see 4.3.1). These cases are named Scenario 3A (spilling breaker), Scenario 3B (plunging breaker) and Scenario 3C (surging breaker). Each of the cases required different geometry. Scenarios 3B and 3C used modified versions of the same tank. The gradient of the slope for Scenario 3B and 3C is 1:6 and 1:2.1, respectively. The geometry used for Scenario 3A is given in Figure 9, while scenarios 3B and 3C are depicted in Figure 10. The angles of each slope are, respectively, β_1=1.5°, β_2=9.5° and β_3=25.4°.

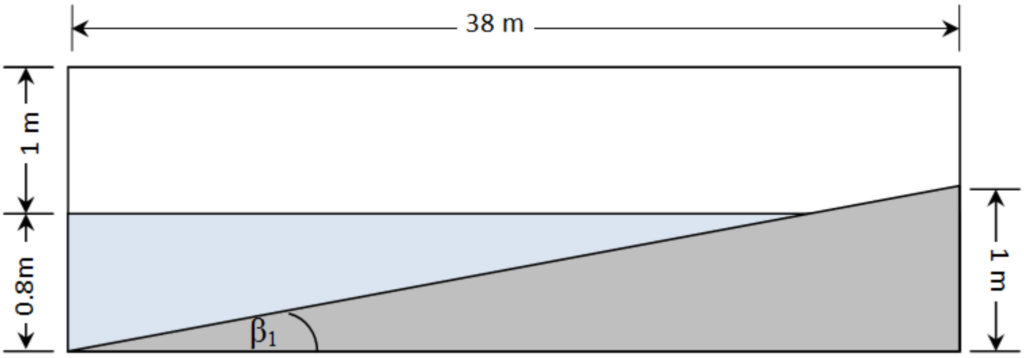


Figure 9. Geometry of sloped tank (Scenario 3A) (not to scale)

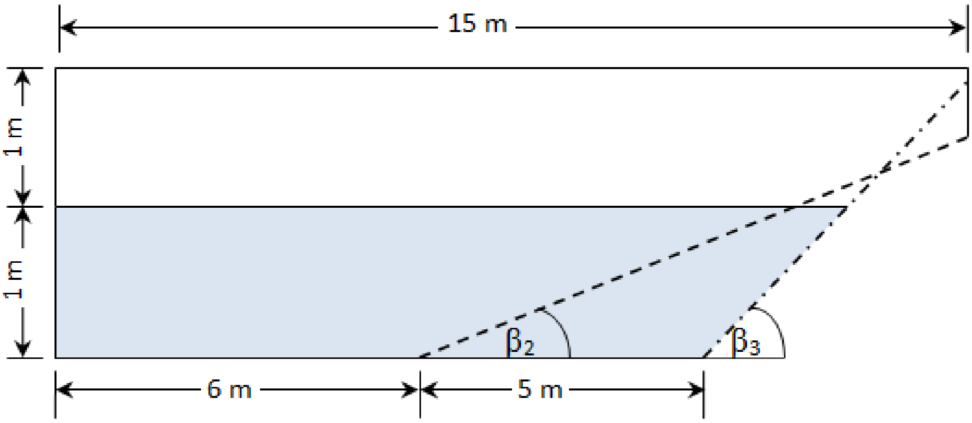


Figure 10. Geometry of scenario 3B (β_2) and 3C (β_3) (figure not to scale)

5.1.4. Scenario 4: Tank with floating object

The geometry used for the demonstration case of a floating object under the influence of regular waves is given in Figure 11, where the grey box represents the floating object.

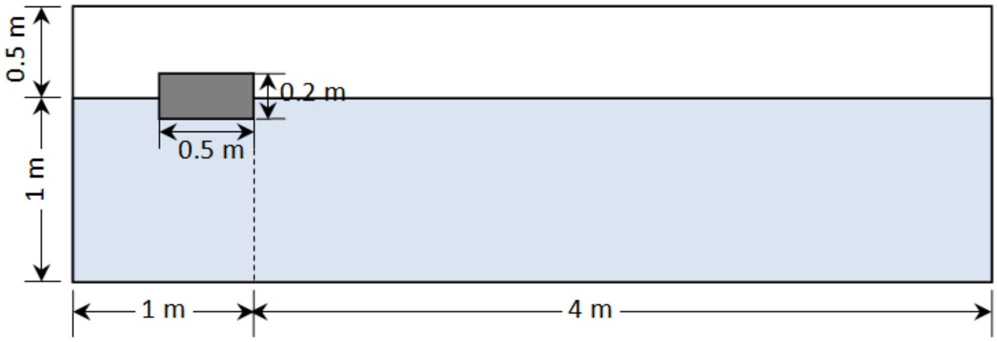


Figure 11. Geometry of floating object scenario (Scenario 4)

5.2. Input wave parameters

The input wave parameters used for each scenario are summarised in Table 5.

Table 5. Input wave parameters for each scenario

Scenario	Wave length L [m]	Wave Height H [m]	Period T [s]	Water depth h [m]	Steepness H/L
1	5	0.1	1.94	1	0.02
2	3.738	0.02	2.02	0.4	0.00535
3A	5	0.2	2.05	0.8	0.04
3B	5	0.2	1.94	1	0.04
3C	5	0.1	1.94	1	0.02
4	5	0.2	1.94	1	0.04

5.3. Production of appropriate boundary conditions in OpenFOAM

In order to replicate the behaviour of a physical wave tank the boundary conditions of the numerical wave tank need to be chosen to recreate physical behaviour. The numerical wave tank consists of 5 boundaries: inlet, outlet, atmosphere, bottom and frontAndBack. frontAndBack describes the boundary of both the front and back of the wave tank but was given an “empty” condition for all parameters to allow OpenFOAM to solve for two dimensions only. The location of each boundary is shown in Figure 12. For Scenario 2 and Scenario 3, the boundary *bottom* included the sloped geometry.

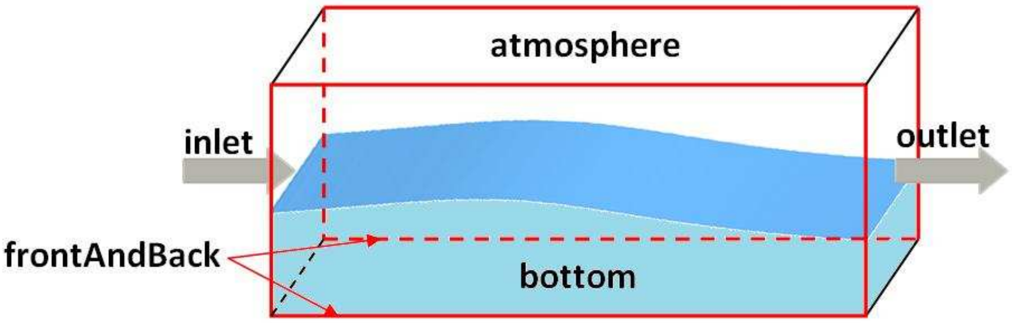


Figure 12. Location and name of each boundary of the numerical wave tank

Three files are required by OpenFOAM to fully describe the behaviour of each boundary. These files are called:

1. *alpha1*, used to determine the volume fraction
2. *U*, used to determine the velocity
3. *p_rgh*, used to determine the dynamic pressure

5.3.1. Producing waves at the inlet

There are two possible methods for producing waves at the inlet using OpenFOAM, the creation of a piston-type wave maker, as is used in physical wave tanks, or the use of a moving boundary condition. To create a piston-type wave maker in OpenFOAM requires the creation of a dynamic mesh that is computationally expensive. For the purposes of this thesis a moving boundary condition was implemented.

A moving boundary condition that allows the Stokes second order particle velocity to be specified as an equation (as given in 4.1.1) was needed. While OpenFOAM comes with many pre-existing boundary conditions, such as “fixed pressure” and “moving wall”, there is no pre-existing boundary condition that allows the input of *x*-axis and *z*-axis velocity.

In response to the need to create numerical waves in OpenFOAM, users have created a boundary condition known as *groovyBC*. *groovyBC* has altered the libraries and source code of OpenFOAM to allow a boundary that can be programmed with equations. The *groovyBC* boundary condition was used to implement the Stokes second order particle velocity equations (see 4.1.1) to describe the inlet velocity. *groovyBC* was also used to describe the phase (water or air) at the inlet. When the simulated surface was equal to or below the theoretical surface elevation (given by Equation (12), see 4.1.3) the phase fraction was forced to equal 1.

Waves created at the inlet using the *groovyBC* boundary condition are shown in Figure 13. The colours in Figure 13 indicate the phase fraction, i.e., red is water, blue is gaseous air, green represents the water-air interface.

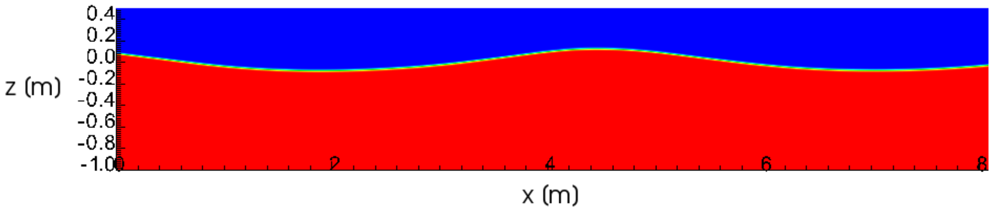


Figure 13. An example of Stokes second order waves created in OpenFOAM using the *groovyBC* inlet condition (inlet located on the left hand side). NB. Whole tank not shown

5.3.2. Preventing reflection of waves at the outlet

To correctly model waves in a numerical wave tank it is necessary to consider the reflection of waves from the boundary of the numerical wave tank. This has caused considerable discussion in a number of articles including Senturk (2011), Koo & Kim (2007), Yong & Mian (2010) and Morgan et al. (2010). A number of solutions have been attempted to absorb incident wave energy:

- *Numerical damping*– a damping coefficient is added to the momentum equation of the OpenFOAM solver
- *A beach* - a secondary structure is added to the end of the wave tank to absorb the energy of the waves
- *A sponge layer* – A porous material is placed at the end of the tank to absorb the energy
- *Increasing mesh size* at the end of the tank to dissipate waves

The last three of these options were attempted by Morgan et al. (2010) who found that all options increased runtime in OpenFOAM and further complicated the model. In their study, reflection was avoided simply by increasing the length of the numerical flume from 45 m to 90 m.

Based on the experience of Morgan et al. (2010) reflection of waves from the outlet was prevented by extending the wave tank to double the length of the modelled geometry. I.e., for the cases of the basic numerical wave tank (Scenario 1), the verification wave tank (Scenario 2) and the floating object demonstration (Scenario 4) the length of the tank was extended to 40 m, 48 m and 10 m, respectively, for implementation in

OpenFOAM. Care was taken to ensure the simulation was not run long enough that waves reflected from the outlet affected the region being studied.

5.3.3. Other boundary conditions

The *zeroGradient* condition[3] was used for alpha1 for the outlet and bottom, to allow surface tension effects between the wall and the water-air interface to be ignored.

A no-slip condition was implemented for the velocity U of the outlet and bottom by forcing the velocity at the wall to zero (as used in OpenFOAM tutorials involving water/air interaction).

The pressure p_rgh at the inlet, outlet, and bottom was set to *bouyantPressure*, which sets the pressure based on the atmospheric pressure gradient. This condition was used in examples provided with the OpenFOAM release where water interacts with a wall.

For the atmosphere a combination of boundary conditions was implemented that maintains stability while permitting both outflow and inflow according to the internal flow (as recommended by OpenFOAM). The *inletOutlet* condition was used for alpha1 of the atmosphere. This condition implements zeroGradient when the velocity vector points out of the domain, with alpha1 specified to equal a value of zero when the velocity vector points into the domain.

The *pressureInletOutletVelocity* condition was used for U of the atmosphere boundary, which applies zeroGradient on all components, except where there is inflow, in which case a value of zero is applied to the tangential component. Pressure at the atmosphere boundary was set to *totalPressure*, where pressure is calculated based on velocity and total pressure (specified to zero).

[3] *zeroGradient* is a generic condition that can be applied to different parameters. *zeroGradient* means that the gradient of the quantity is zero, i.e., the value is constant. In effect *zeroGradient* sets the same value at the boundary as in the neighbouring cell.

5.3.4. Summary of boundary conditions

A summary of all the boundary conditions implemented is given in Table 6.

Table 6. Summary of the boundary conditions implemented in OpenFOAM

Boundary	**alpha1**	**U**	**p_rgh**
Inlet	groovyBC	groovyBC	bouyantPressure
Outlet	zeroGradient	No-slip	bouyantPressure
Atmosphere	inletOutlet	pressureInletOutletVelocity	totalPressure
Bottom	zeroGradient	No-slip	bouyantPressure
frontAndBack	empty	empty	empty

5.4. Choosing the OpenFOAM solver

It is important to choose or design a solver to match the physical problem of water waves moving in a tank. This Master thesis utilised pre-existing solvers that were supplied with the OpenFOAM release.

5.4.1. interFoam

The *interFoam* solver takes into account the movement of air and water and is specifically designed for solving two incompressible, isothermal immiscible fluids based on a volume of fluid phase-fraction approach (OpenFOAM, 2010). This solver captures the phase (and therefore shape) of the water at different times.

The *interFoam* solver has previously been used for problems similar to that which is attempted in this thesis. *interFoam* (formerly known as *rasinterFoam* in early versions of OpenFOAM) was presented as a recommended solver for a surface piercing body under wave action at the 4th OpenFOAM Workshop (Paterson et al., 2009). *interFoam* has also been used in a conference working paper (Morgan et al., 2010) to produce waves breaking over a submerged bar.

The *interFoam* solver was used for majority of the work of this Master thesis involving production of waves in a basic wave tank and waves moving over a submerged bar. The *interFoam* solver was also used for the case of waves interacting with a slope.

5.4.2. interDyMFoam

The *interDyMFoam* is a dynamic solver that performs the same function as *interFoam* with the added ability of the mesh being able to change during the simulation. *interDyMFoam* is capable of producing automatic mesh motion as well as topological changes to the mesh such as addition or removal of a cell layer, boundaries that can be attached and detached, and a sliding interface, where a pair of detached surfaces move relative to each other.

The automatic mesh motion function of *interDyMFoam* was used for the demonstration of a floating object under the influence of waves. The *interDyMFoam* solver calculates the force on the surface of the floating body due to the wave motion and then solves the six degrees-of-freedom equation of motion (Jasak et al., 2008).

5.5. Other simulation parameters

5.5.1. Physical properties of all simulations

In addition to the boundary conditions given in 5.3 the simulation was implemented with other physical properties summarised in Table 7. All simulations ignored turbulence effects (laminar simulation was used).

Table 7. Summary of physical properties of all scenarios modelled

Parameter		**Value**
Gravity		9.81 m s^{-2}
Water	Density	1000 kg m^{-3}
	Kinematic Viscosity	1.0x10^{-6} m^{2}s^{-1}
Air	Density	1.2 kg m^{-3}
	Kinematic Viscosity	1.48x10^{-5} m^{2}s^{-1}
Surface tension		0.07 Nm^{-1}

The value of surface tension implemented was used in OpenFOAM examples with a water-air interface.

5.5.2. Simulation control properties

All simulations were initialised to begin each simulation with water below the still water level and air above the still water level. The air-water interface was calculated at each time step using the Volume of Fluid method. Time steps of 0.001s were used for each simulation with results written every 0.1s.

OpenFOAM allows the user to control which numerical scheme is used to solve for terms, such as derivatives, that appear in the implemented applications. In this work, the default option was used for all numerical schemes except for the solution of divergence terms. As suggested in the OpenFOAM user manual (OpenFOAM, 2010) for use with the *interFoam* solver, the following options were chosen for the divergence terms:

Table 8. Numerical schemes used for solution of divergence terms

Divergence Term	**Numerical Scheme**	**Notes**
div(rho*phi,U)	limitedLinearV 1	Produces good accuracy when used with *interFoam*,
div(phi, alpha)	vanLeer	Van Leer flux limiter
div(phirb,alpha)	interfaceCompression	Specialised scheme for producing a smoother interface

OpenFOAM also allows flexibility in determining how each solver is run. The PISO (pressure-implicit split-operator) algorithm for use with transient problems was implemented for all modelled scenarios. The preconditioned conjugate gradient (PCG) linear solver was used to solve for velocity and pressure.

5.5.3. Properties of floating object scenario

In order for a floating object to be modelled in OpenFOAM the mass, centre of mass, density and moments of inertia need to be determined. Based on the geometry of the box given in 5.1.4 and assuming a density of 888 kg m^{-3} and a centre of mass in the centre of the box, the characteristics given in Table 9 were determined.

Table 9. Properties of floating object

Mass	8.889 kg
Moment of Inertia I_x	0.037 kg m^{-2}
Moment of Inertia I_y	0.215 kg m^{-2}
Moment of Inertia I_z	0.193 kg m^{-2}

5.6. Production of an appropriate mesh

5.6.1. Generation of mesh

The OpenFOAM utility, *blockMesh*, was used to generate the base mesh for each scenario. The base mesh did not contain any obstructions or sloped bottom. The number of cells in the mesh was specified as well as any grading of the cells in a given direction. All scenarios were generated with a single cell in the y direction with a thickness of 0.1 m.

After the base mesh was created the *snappyHexMesh* utility was implemented. This utility adjusts the base mesh to match the desired geometry. The shape of the desired geometry (such as the bottom for the verification tank) was defined using a Stereolithography (STL) file created in Solidworks. All cells outside the desired geometry were removed by *snappyHexMesh* and the cells around the edge were deformed to follow the shape of the desired geometry.

5.6.2. Testing mesh independency for Scenarios 1 and 2

To test the validity of the generated mesh, independency was tested using the numerical wave tank of Scenario 2. This scenario was used for mesh independency because it requires the highest quality mesh due to the behaviour of waves after the submerged bar. Several meshes were tested including grading around the water surface (cells at water surface were five times smaller than at the atmosphere and bottom boundaries) and extra cells inserted to cover gauge 6 to 11 (between 13 and 22 m, known as Section B). Section B is highlighted in Figure 14.

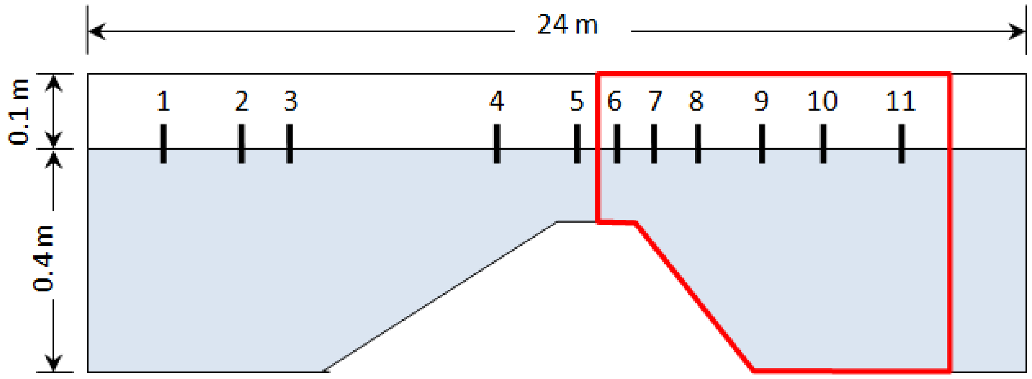


Figure 14. Mesh shape for Scenario 2. Section B, given extra refinement for Mesh D, is highlighted in red

The details of each mesh tested are given in Table 10. Note that the average cell size is determined before cells are removed by *snappyHexMesh*. Representative sections of each mesh are shown in Figure 15 and Figure 16. In these figures, the blue line represents the SWL.

Table 10. Mesh parameters for mesh independency test

Mesh	**Average cell size *x* direction [m]**		**Average cell size *z* direction [m]**		**Notes**
	Base Mesh	Section B	Base Mesh	Section B	
A	0.04	-	0.02	-	Uniform grading throughout tank
B	0.02	-	0.01	-	Uniform grading throughout tank
C	0.02	-	0.008	-	Vertical grading around the still water level
D	0.02	0.01	0.008	0.004	Vertical grading around the still water level with additional refinement in Section B
E	0.01	-	0.004	-	Vertical grading with very high refinement throughout
F	0.02	-	0.01	-	Uniform grading throughout; snappyHexMesh altered to create a smoother slope

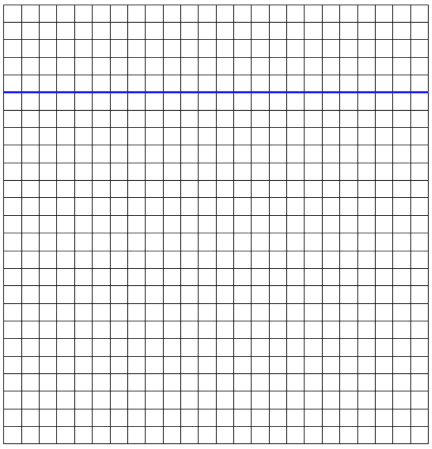
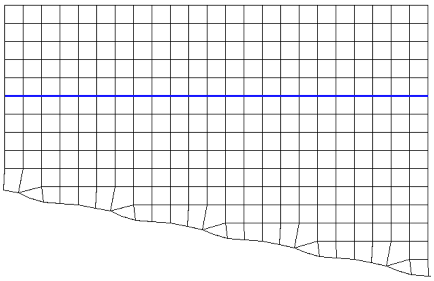
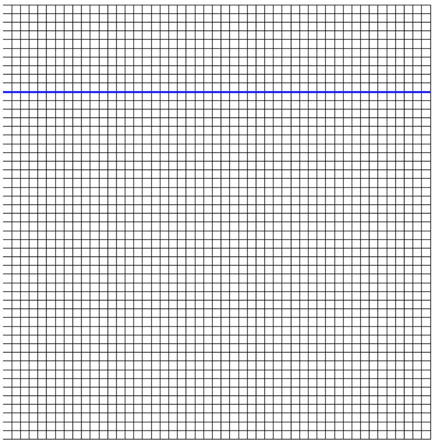
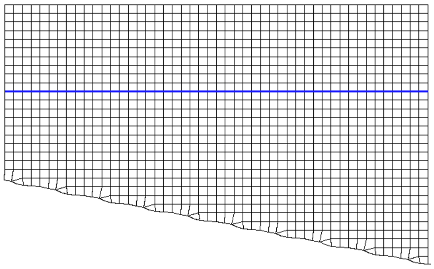
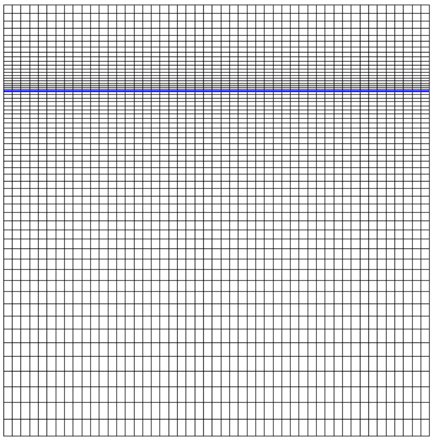
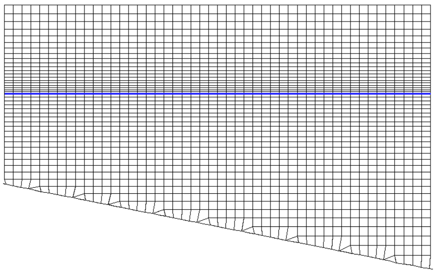
Mesh A	
0-1 m	**14-15 m**
	
Mesh B	
0-1 m	**14-15 m**
	
Mesh C	
0-1 m	**14-15 m**
	

Figure 15. Close-up of representative sections of Mesh A, B and C

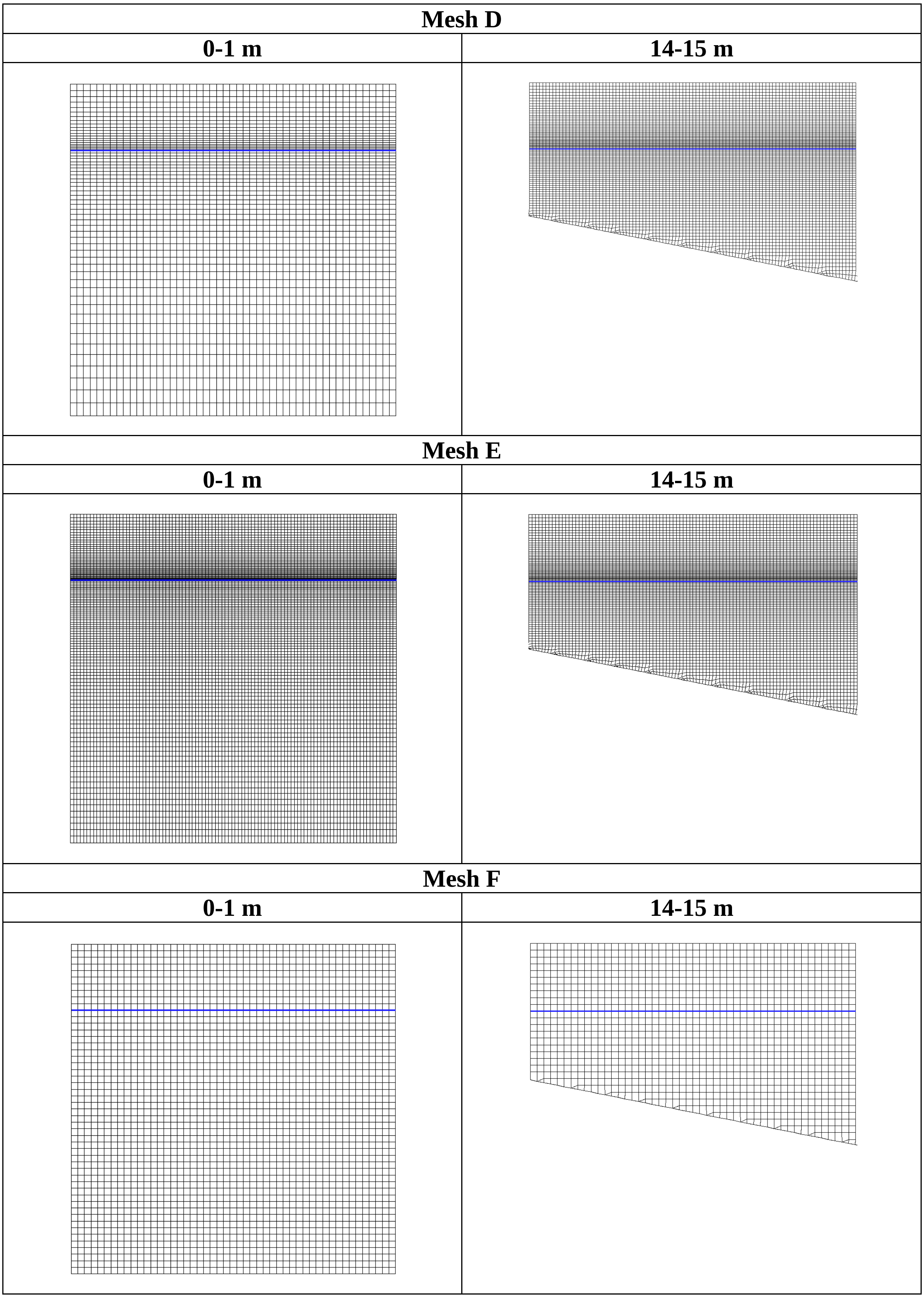


Figure 16. Close-up of representative sections of Mesh D, E and F

Figure 17 to Figure 27 presents the results of the mesh independency tests. It can be seen that little difference is seen between the meshes for the earlier gauges. After gauge 5, Mesh A shows a reduction in the simulated surface elevation compared to the other meshes (due to Mesh A’s poorer resolution). Meshes B, C, D, E, and F show very little variation between them after gauge 5. Due to the quicker processing time of Mesh B, this mesh quality was chosen for Scenario 1 and 2. Note that there are two sets of experimental results for gauge 3 (named 3(A) and 3(B)). Only the results for gauge 3(A) are shown for the mesh independency results.

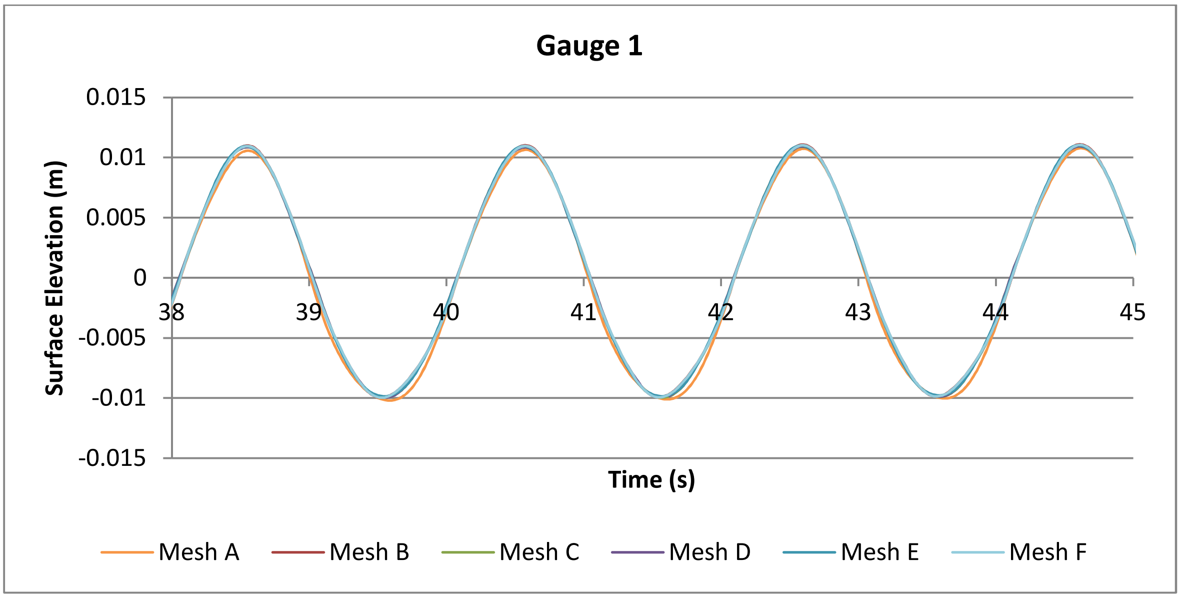


Figure 17. Mesh independency results for gauge 1

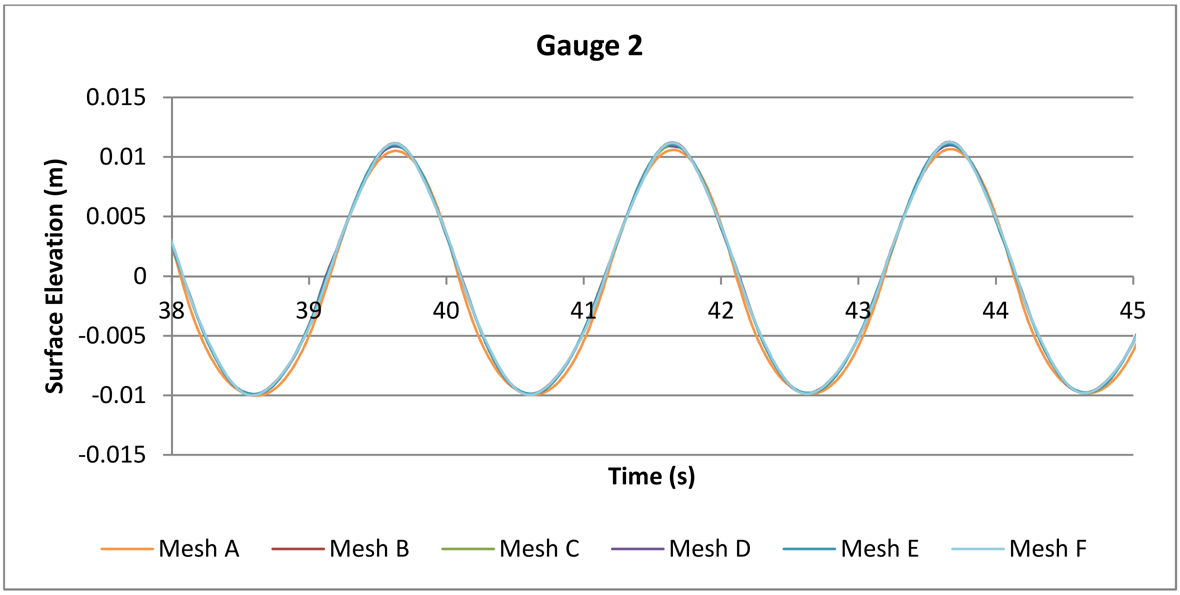


Figure 18. Mesh independency results for gauge 2

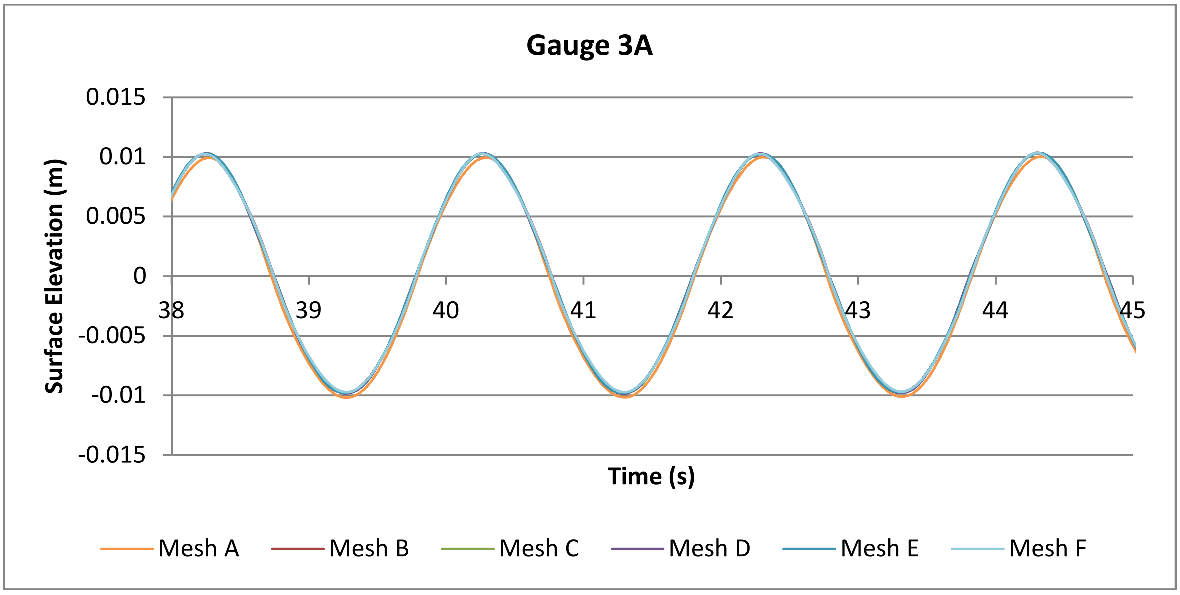


Figure 19. Mesh independency results for gauge 3A

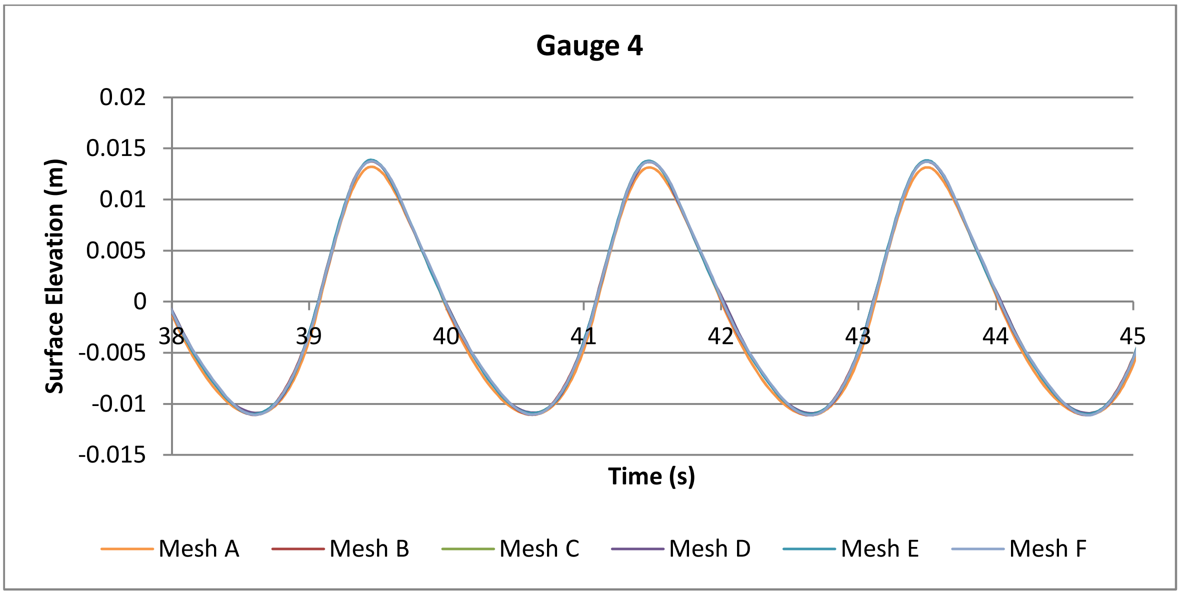


Figure 20. Mesh independency results for gauge 4

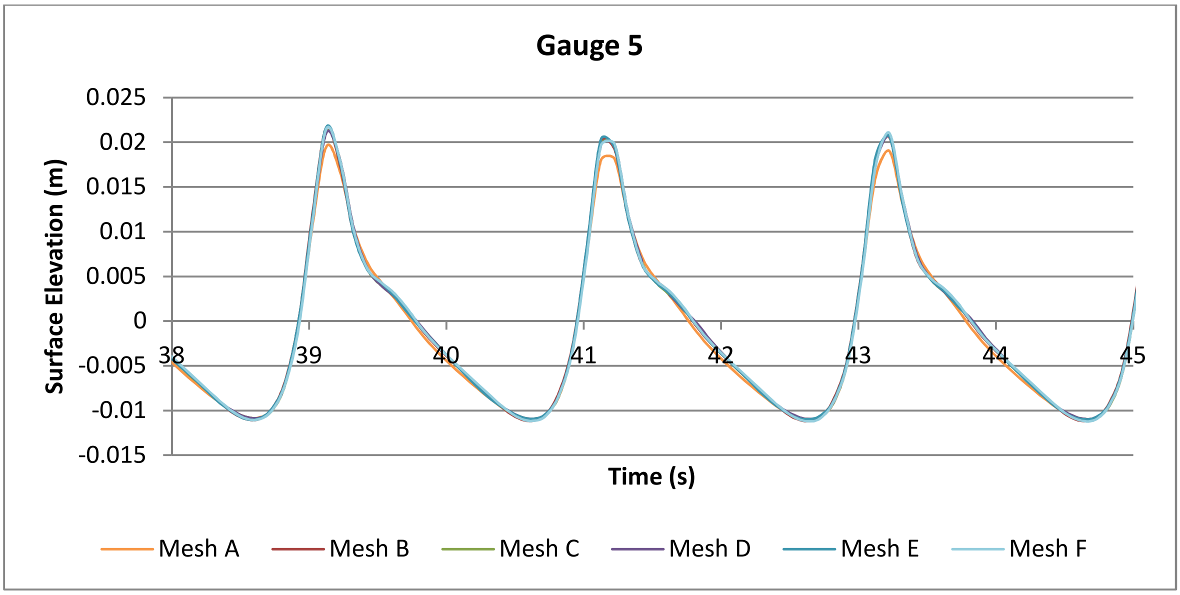


Figure 21. Mesh independency results for gauge 5

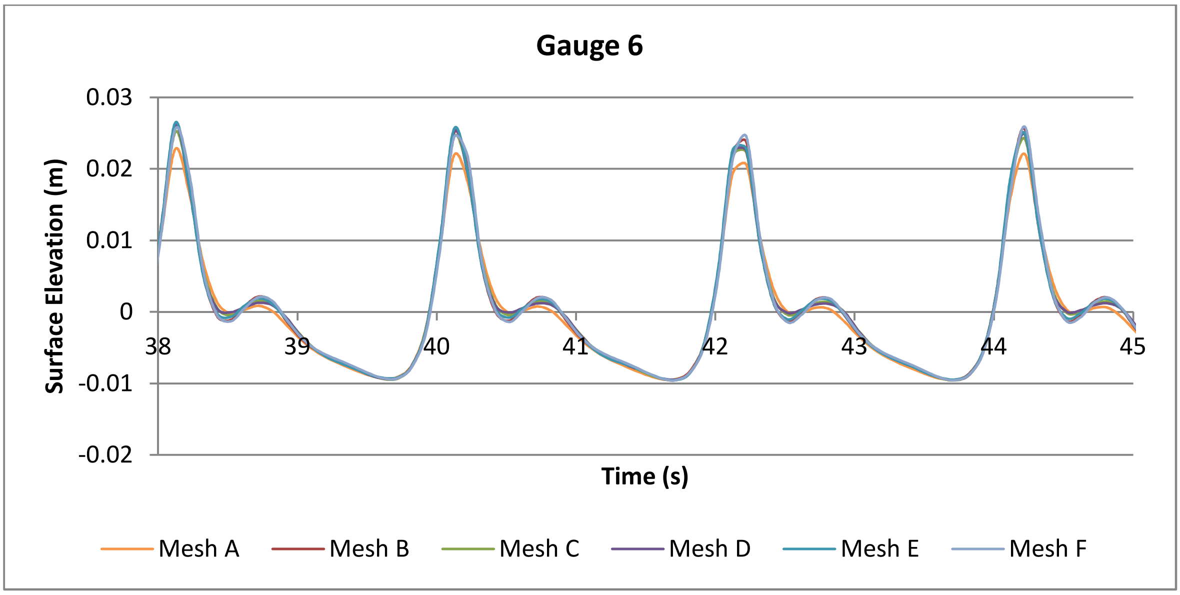


Figure 22. Mesh independency results for gauge 6

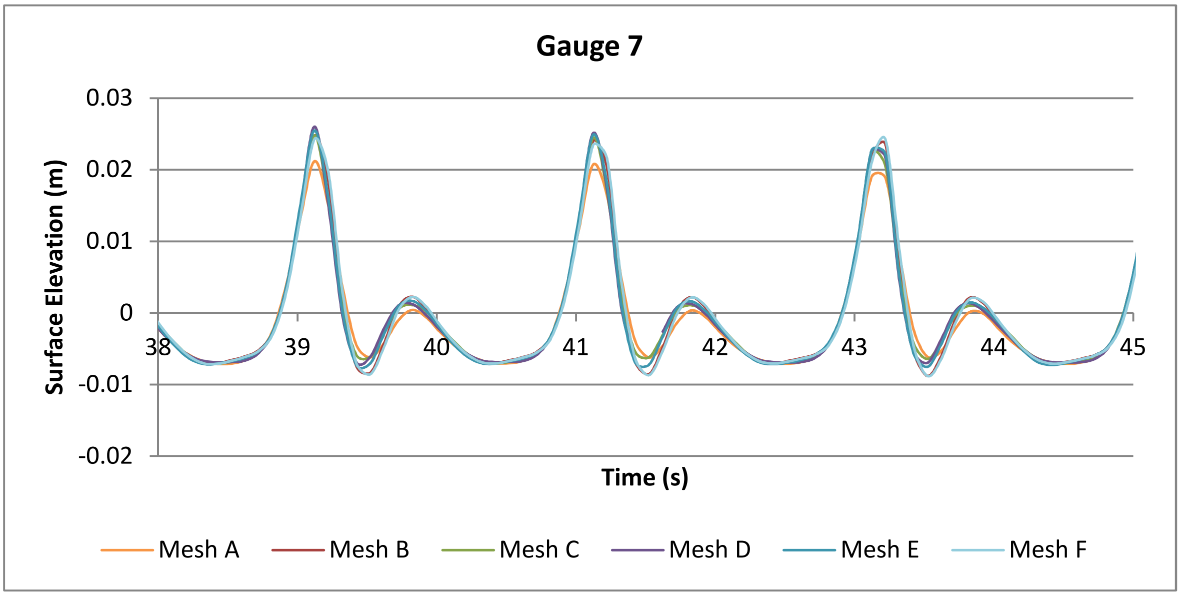


Figure 23. Mesh independency results for gauge 7

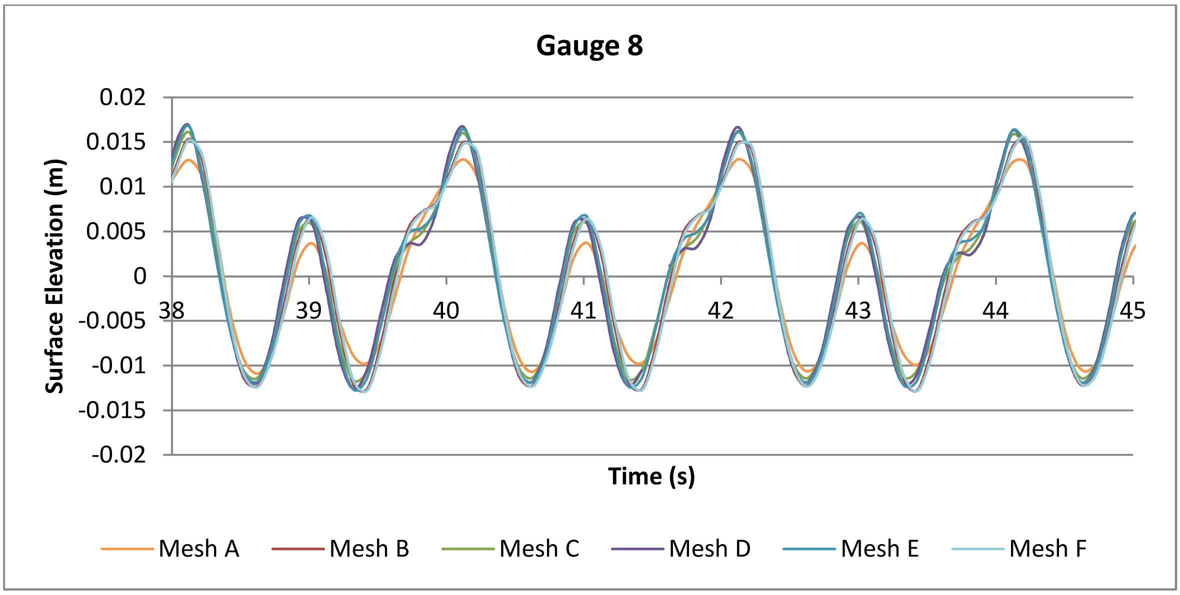


Figure 24. Mesh independency results for gauge 8

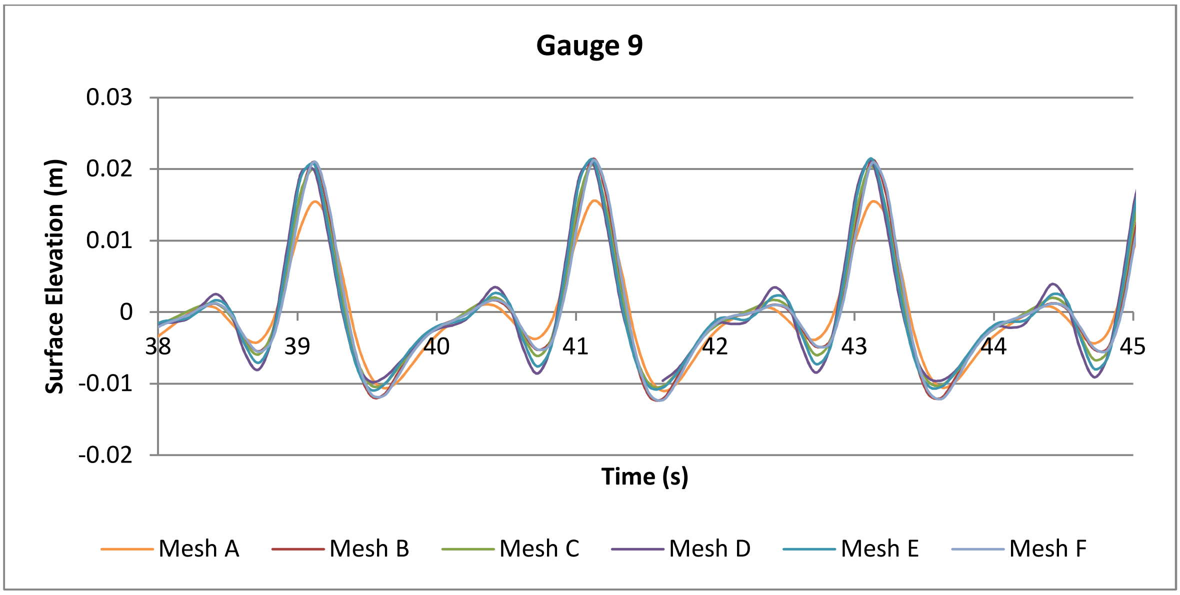


Figure 25. Mesh independency results for gauge 9

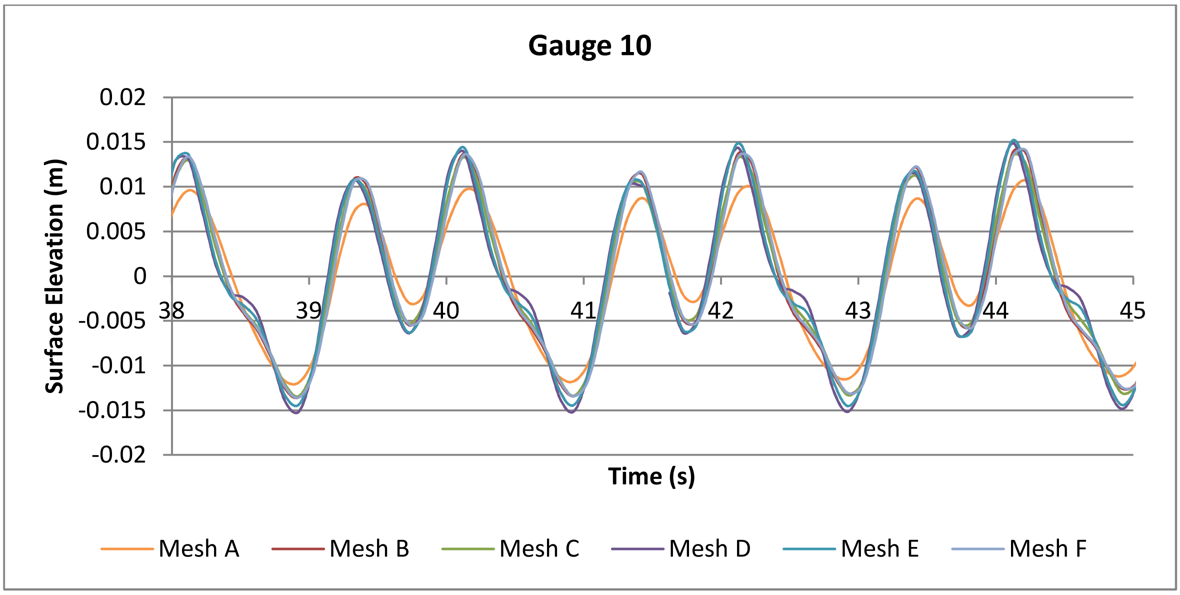


Figure 26. Mesh independency results for gauge 10

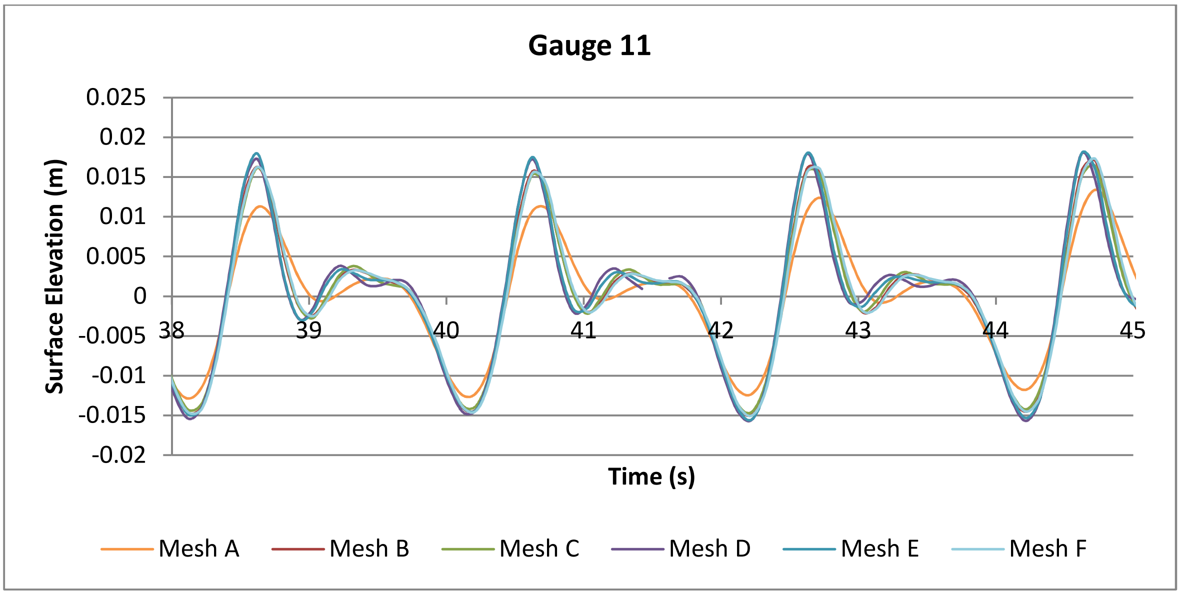


Figure 27. Mesh independency results for gauge 11

5.6.3. Mesh size for Scenarios 3 and 4

The mesh quality for Scenario 3 and Scenario 4 is given in Table 11. For both scenarios, grading was uniform throughout the tank.

Table 11. Mesh size for scenarios 3 and 4

Scenario	Cell size *x* direction [m]	Cell size *z* direction [m]
3A	0.02	0.01
3B	0.015	0.01
3C	0.015	0.01
4	0.02	0.06

Given the accuracy of the results of Mesh B in the mesh independency test this was considered the minimum quality needed for Scenario 3 and 4, for which there was no experimental data to verify the results. In Scenario 3B and Scenario 3C cell size in the *x* direction was increased to 0.015 m.

Scenario 4 implemented *interDyMFoam* and required more computational time than *interFoam*. It was found that the size of the cells affected the ability of *interDyMFoam* to simulate. The longest simulation was achieved by reducing the resolution in the *z* direction to 0.06 m.

5.7. Post-processing and analysis of simulation results

After each simulation is completed, the results are written to a folder containing the physical properties of the model for each time step (of 0.1s). This data can be analysed using ParaView, a post-processing program included with the OpenFOAM 1.7.1 release.

ParaView is an open source program for data analysis and visualisation that can be used to visualise the results of simulation in OpenFOAM. ParaView is specifically designed to handle large datasets and while not specifically designed for OpenFOAM it is the recommended program for post-processing of OpenFOAM simulations. Results can be manipulated in similar ways as commercial programs such as Fluent. ParaView was used for all post-processing of simulation results of this Master thesis, including production of graphics and animations.

6. MODELLING RESULTS AND DISCUSSION

6.1. Scenario 1: Basic numerical wave tank

6.1.1. Basic verification

The behaviour of the basic flat bottomed numerical wave tank was verified by comparing the simulation results with the expected ideal surface elevation determined by Equation (12). Using the wave parameters listed in 5.2, Scenario 1 was simulated with the results shown in Figure 28. It can be seen that the simulated results using OpenFOAM closely match the expected results. The use of a finer mesh did not improve the results. Given that the wave tank was validated (within a certain criteria, see 6.1.2) for a basic wave tank with no sloped bottom, modelling of Scenario 2 was attempted (see 6.2).

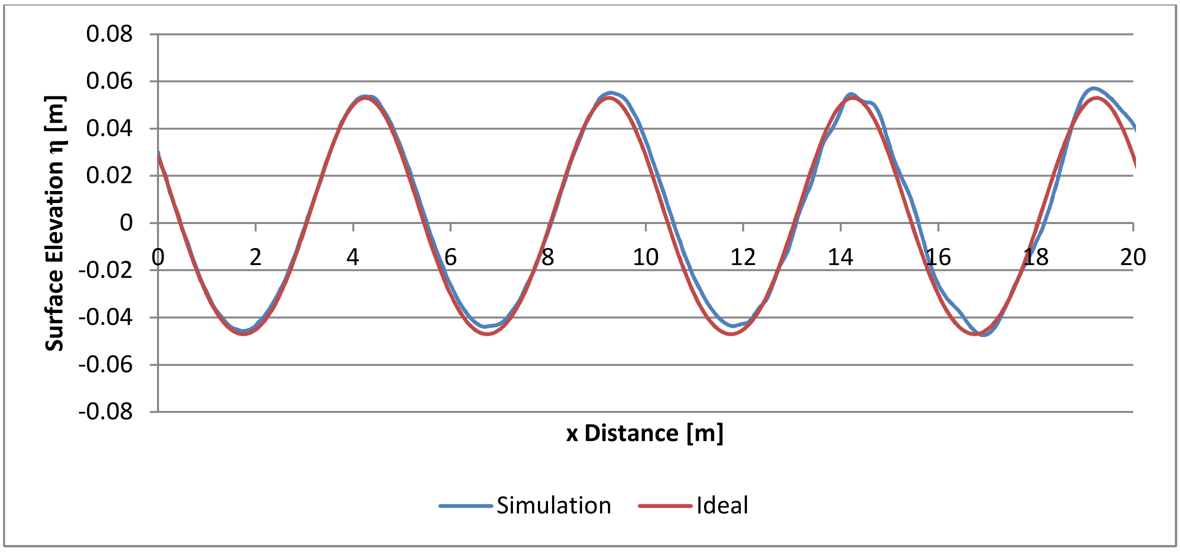


Figure 28. Comparison of simulated surface elevation and theoretical surface elevation for Scenario 1 numerical wave tank

6.1.2. Limitations of modelling regular waves in OpenFOAM

During the computational simulation, it was found that OpenFOAM does not correctly simulate waves with steepness of 0.05 and above, despite being correctly generated by the inlet boundary condition. Incorrect simulation was found to manifest itself in three different ways:

1. Waves are continuous but do not match ideal surface elevation,
2. Damping of waves,
3. Regular waves with a steepness below the critical breaking value (see 4.2) are shown to break.

Two of the simulation limitations are discussed in further detail in this section. Given the limitations of generating waves with steepness above 0.05, it is recommended that only waves with steepness lower or equal to 0.05 are simulated in OpenFOAM.

6.1.2.1. Damping of waves

Damping of waves was visible with a steepness of 0.1, the example given uses wave parameters of $L = 1$ m, $h = 1$ m and $H = 0.1$ m. The damping of the waves can be seen in Figure 29, which illustrates the difference between the simulated waves and the expected ideal surface elevation based on equation (12). The damping of the wave can be attributed to a high horizontal velocity of air close to the interface, as seen in Figure 30.

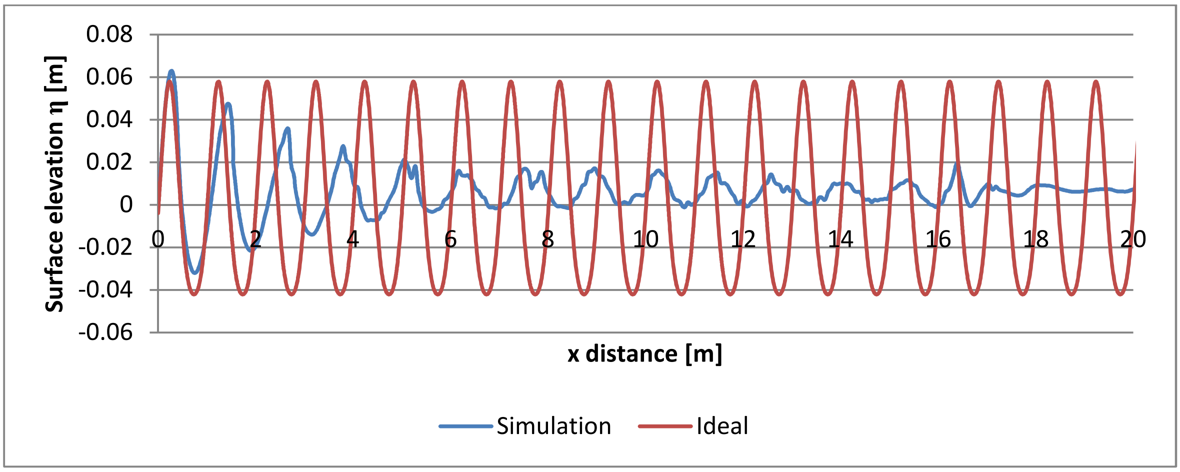


Figure 29. Ideal and simulated surface elevation along the basic numerical wave tank at *t*=25s. Damping of the simulated waves is visible

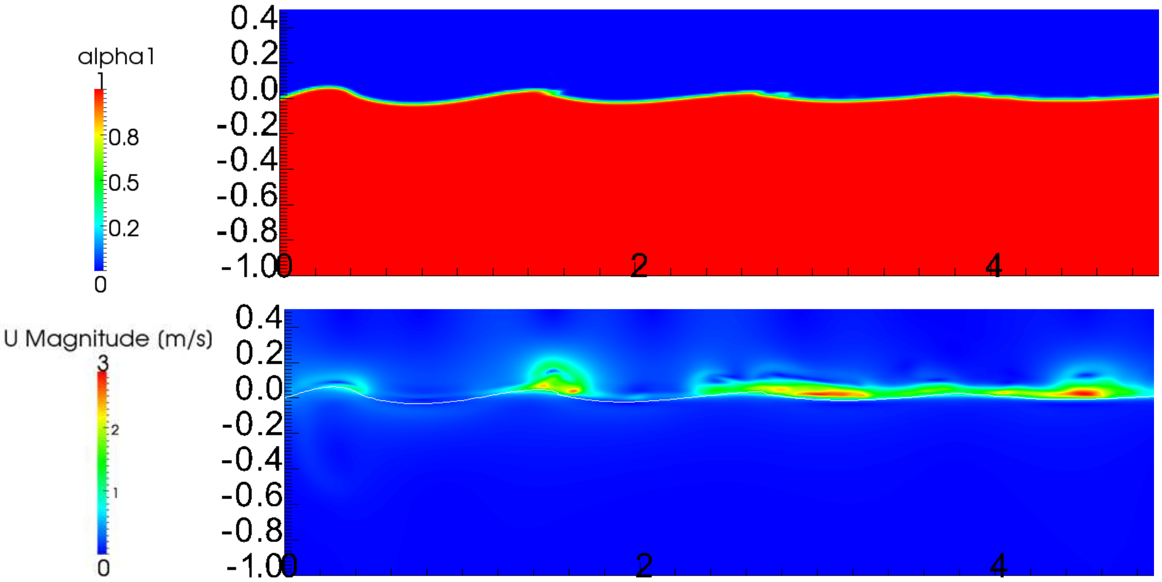


Figure 30. Damping of waves with steepness = 0.1. The phase fraction (alpha1) and velocity (U) at *t*=25s are shown. Axes are x[m].

Afshar (2010) has also noted the effect of unwanted air velocity in OpenFOAM. Afshar (2010) noted that an anomalous horizontal velocity is generated close to the interface, increasing the error of waves that are propagating in the wave tank. This behaviour can be attributed to the VOF (Volume of Fluid) scheme used by the *interFoam* solver for free surface modelling. This method uses the difference in density between air and water to determine where the air-water interface is located. Given that the density of water is approximately 1000 times the density of air, this leaves a region of air near the interface with a velocity far greater than the velocity of the water (Afshar, 2010).

To combat the effects of the unwanted air velocity Afshar (2010) suggests “relaxation” of the air velocity by modifying the source code to replace the unwanted high velocity with a velocity of zero at each time step. Afshar (2010) demonstrated that this method could reduce the error of the air velocity. Another method suggested by Paterson (2008) is to modify air convection in the governing equations (see 3.2) by multiplying the convective term by the phase fraction value.

Neither of these methods was implemented for this Master thesis as the steepness of the waves for the experimental validation tank had a steepness well below the limit where the effects of the unwanted air velocity are seen. All of the scenarios modelled were simulated using a steepness less than 0.05 where OpenFOAM was shown to correctly model the waves.

6.1.2.2. Breaking waves

In addition to damping waves, OpenFOAM was also shown to cause breaking waves at less than the theoretical limit discussed in 4.2. This behaviour was only witnessed above a steepness of 0.05. An example of a regular wave breaking at less than the theoretical limit is shown in Figure 31 and can again be attributed to the production of unwanted air velocity (see 6.1.2.1). This wave, with parameters $L = 2$ m, $h = 1$ m and $H = 0.2$ m, $H/L = 0.1$, should only break at a steepness of $H/L = 0.14$, according to Equation (13).

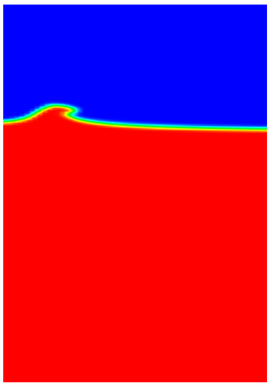


Figure 31. Example of a regular wave breaking below the theoretical limit. The parameters used are *L*=2 m, *h*=1 m, *H*=0.2 m, and *H/L*=0.1 m. The theoretical breaking limit of this wave is *H/L*=0.14 m

6.2. Scenario 2: Verification of numerical wave tank against experimental data

The experimental results of the Dingemans (1994) experiment (see Chapter 2) were used to verify that the numerical wave tank could simulate results that closely replicate physical behaviour. Only Case A of the experiments was modelled: $T = 2.02$ s, $H = 0.02$ m and $L = 3.738$ m.

For the purposes of comparison between the experiments and simulation results only data from 38-45s is shown. At this time, all wave gauges experienced a fully developed wave and no reflection from the outlet had reached gauge 11.

6.2.1. Basic verification of tank for experimental comparison

Before analysing the surface elevation at each wave gauge the behaviour of the wave tank was tested to verify that the numerical wave tank behaves correctly with the given dimensions. The submerged bar (see geometry in 5.1.2) was replaced with a flat bottom and the parameters of Case A (see Chapter 2, Table 1) inserted into the inlet boundary condition.

The wave pattern across the tank at $t = 35$ s is shown in Figure 32 and compared to the ideal surface elevation at that time (calculated using Equation (12)). It can be seen that tank behaves correctly for the given geometry and waves used for Scenario 2.

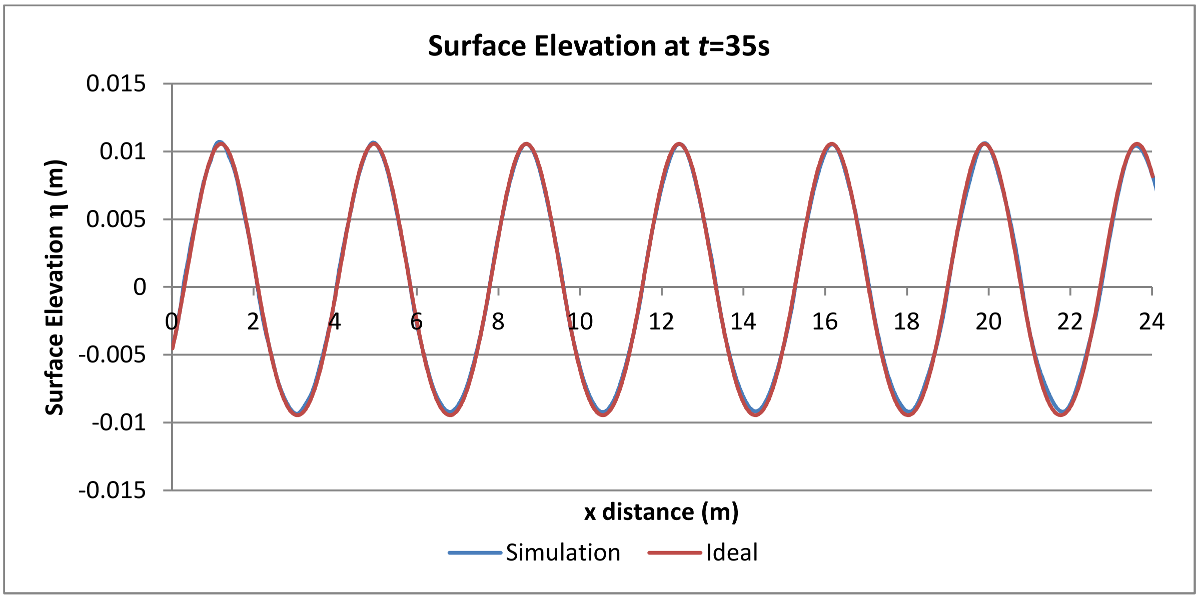


Figure 32. Surface elevation at *t*=35s along Scenario 2 experimental verification tank without submerged bar

6.2.2. Surface elevation results at each wave gauge

Comparisons of the simulation with the experimental results of Dingemans (1994) are given from Figure 33 to Figure 44. Note that two experimental measurements have been taken at gauge 3, labelled 3(A) and 3(B). The single simulation results for gauge 3 has been compared to both measurements.

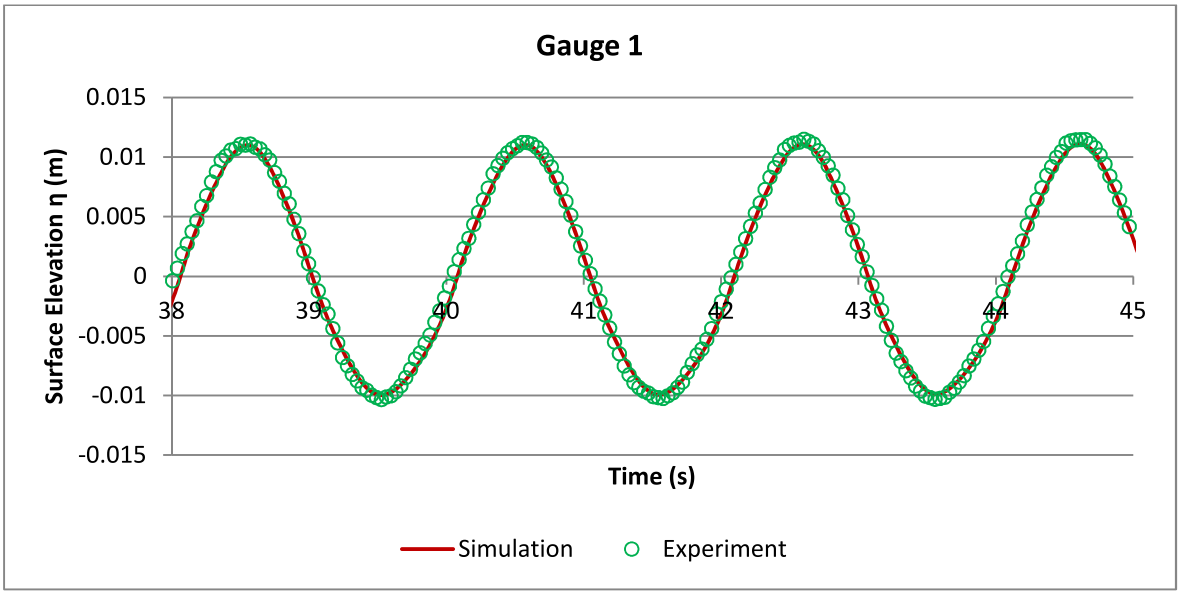


Figure 33. Comparison of experimental and ideal surface elevation for gauge 1, Scenario 2

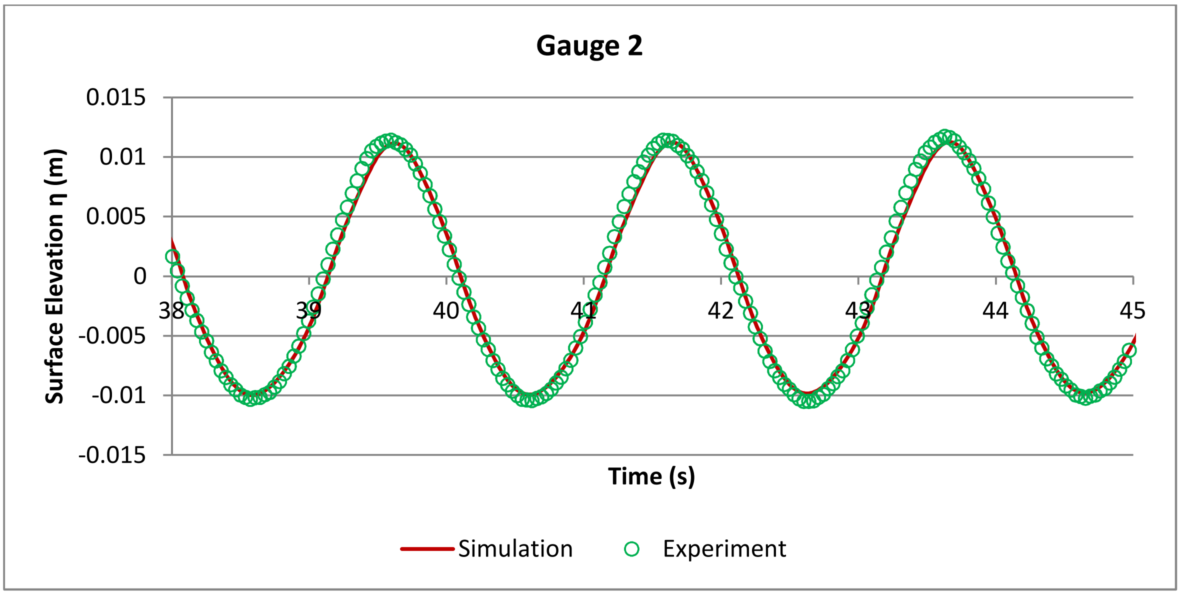


Figure 34. Comparison of experimental and ideal surface elevation for gauge 2, Scenario 2

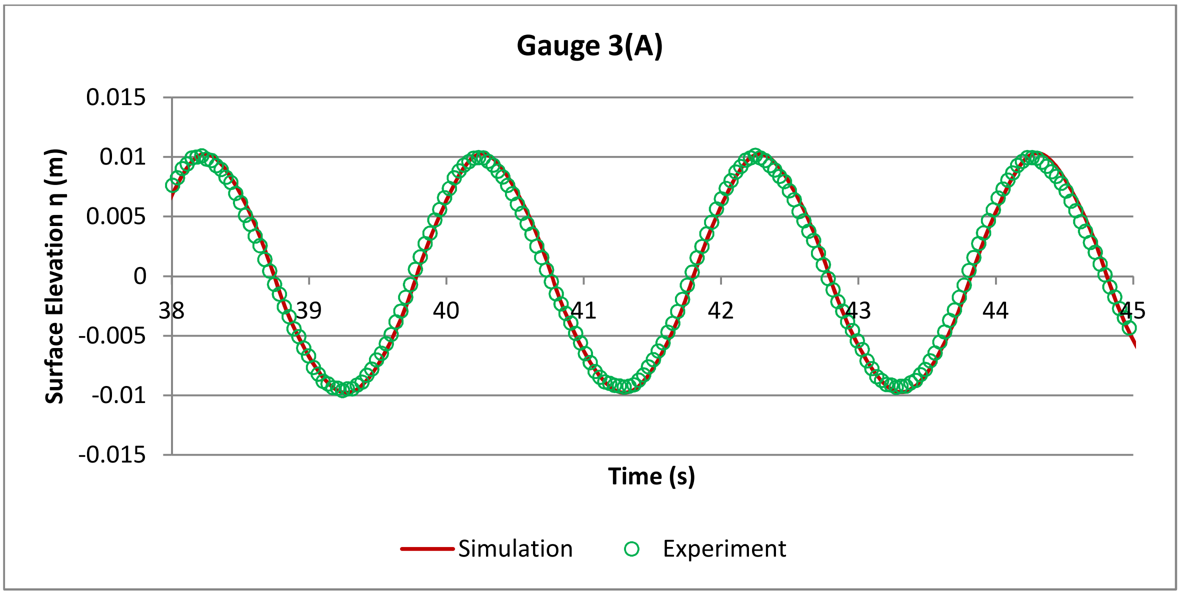


Figure 35. Comparison of experimental and ideal surface elevation for gauge 3(A), Scenario 2

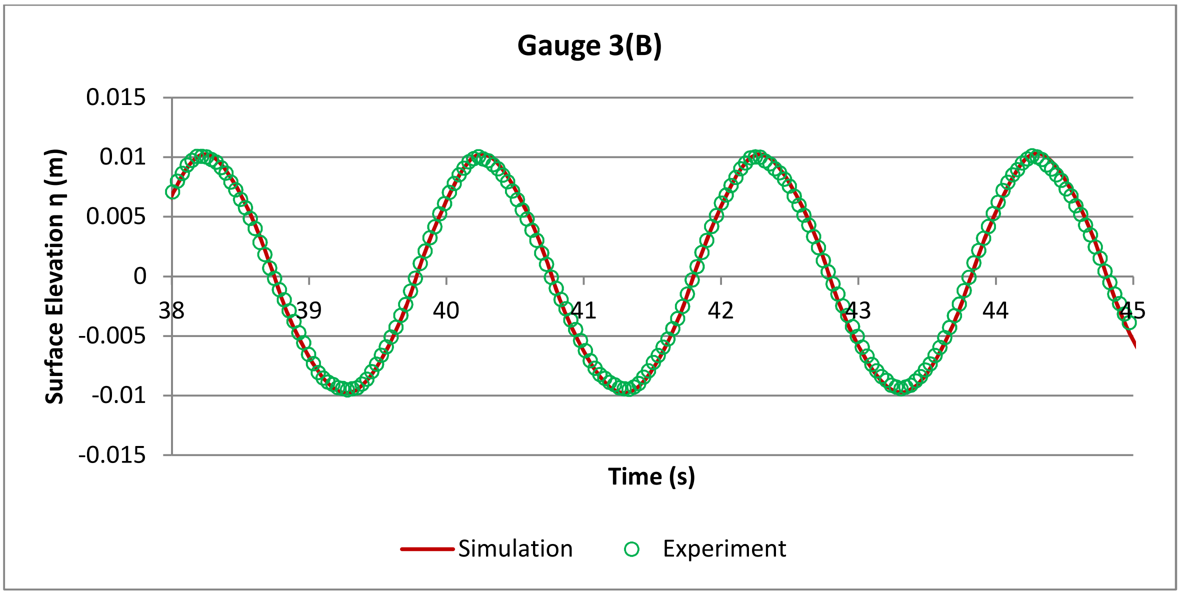


Figure 36. Comparison of experimental and ideal surface elevation for gauge 3(B), Scenario 2

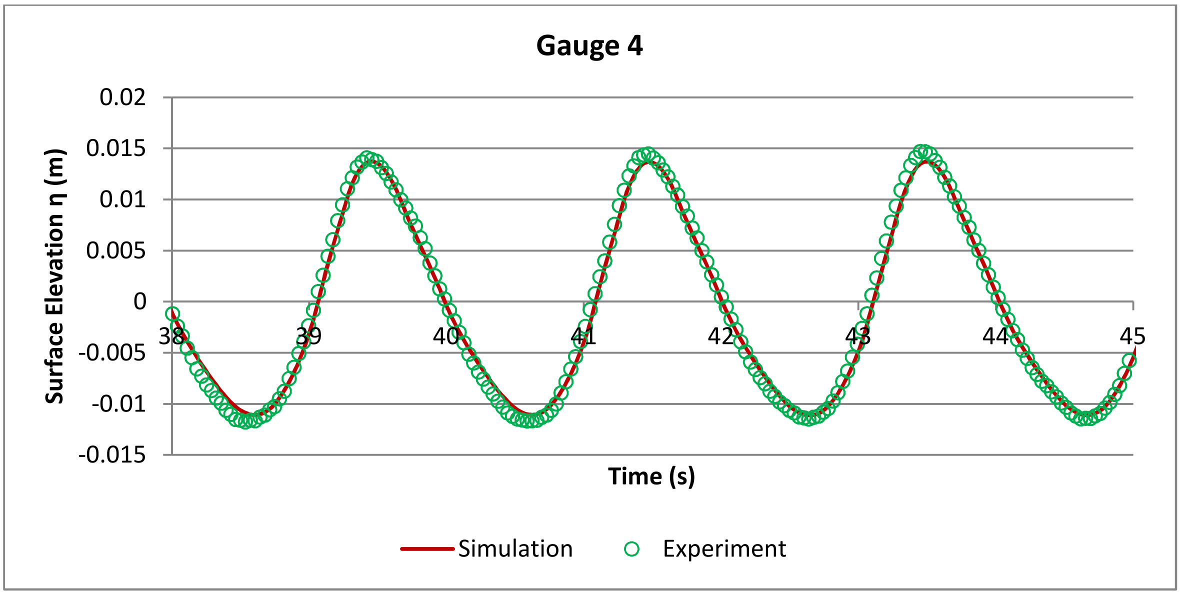


Figure 37. Comparison of experimental and ideal surface elevation for gauge 4, Scenario 2

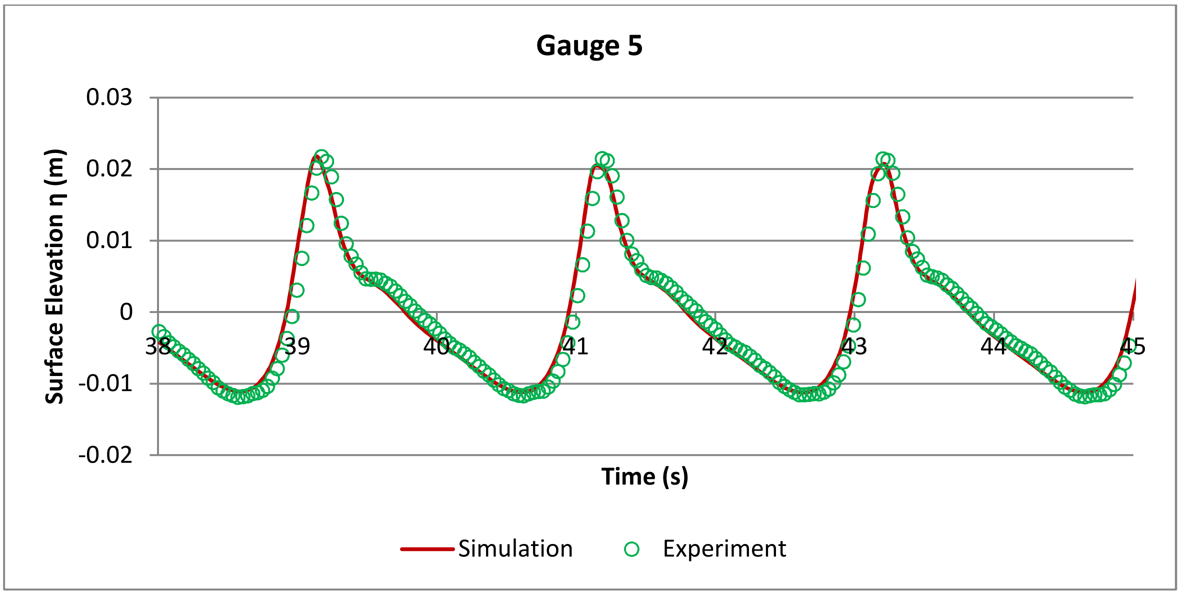


Figure 38. Comparison of experimental and ideal surface elevation for gauge 5, Scenario 2

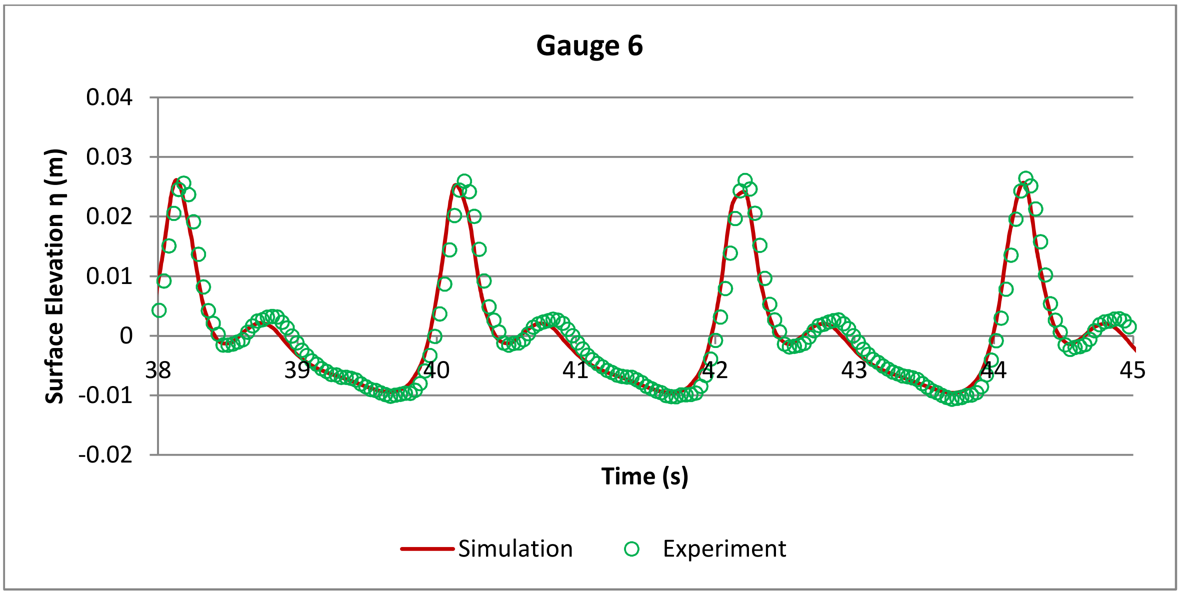


Figure 39. Comparison of experimental and ideal surface elevation for gauge 6, Scenario 2

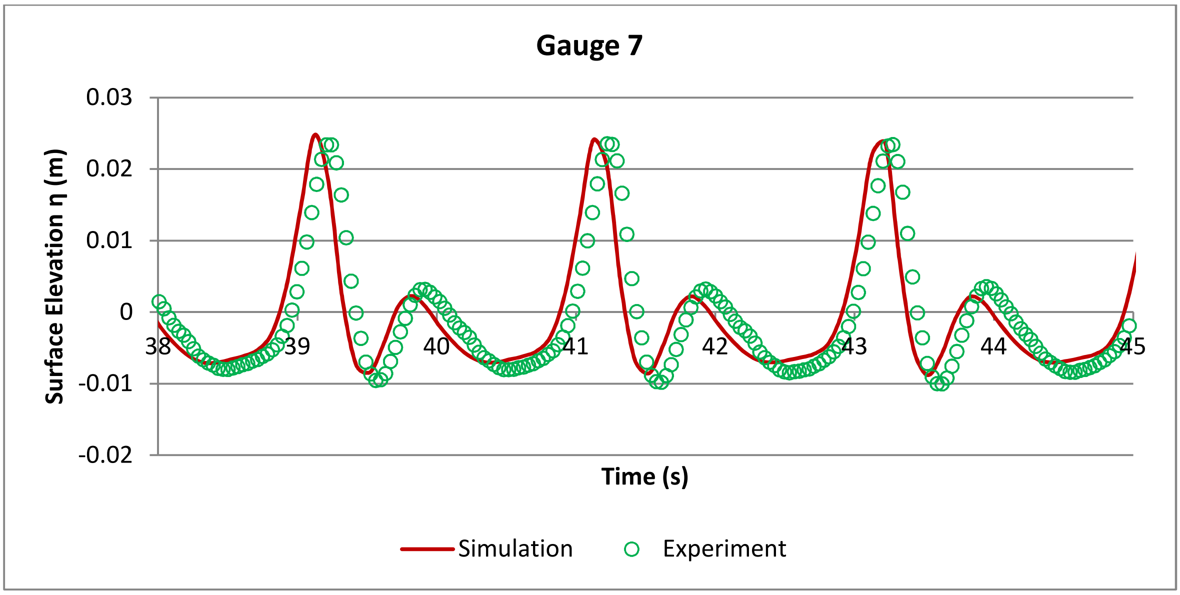


Figure 40. Comparison of experimental and ideal surface elevation for gauge 7, Scenario 2

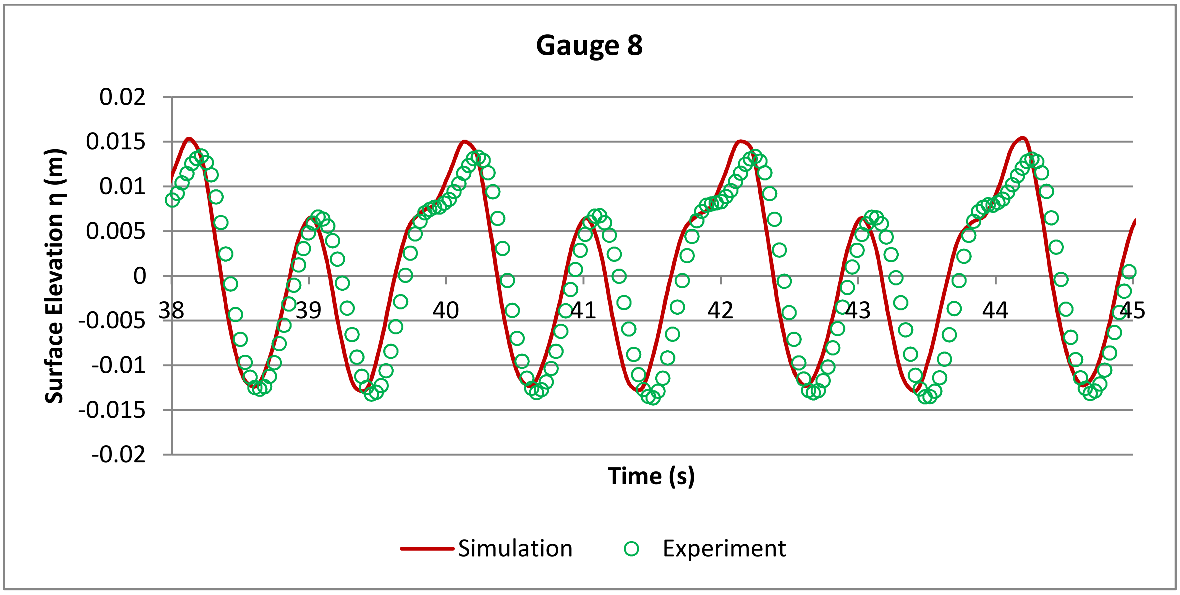


Figure 41. Comparison of experimental and ideal surface elevation for gauge 8, Scenario 2

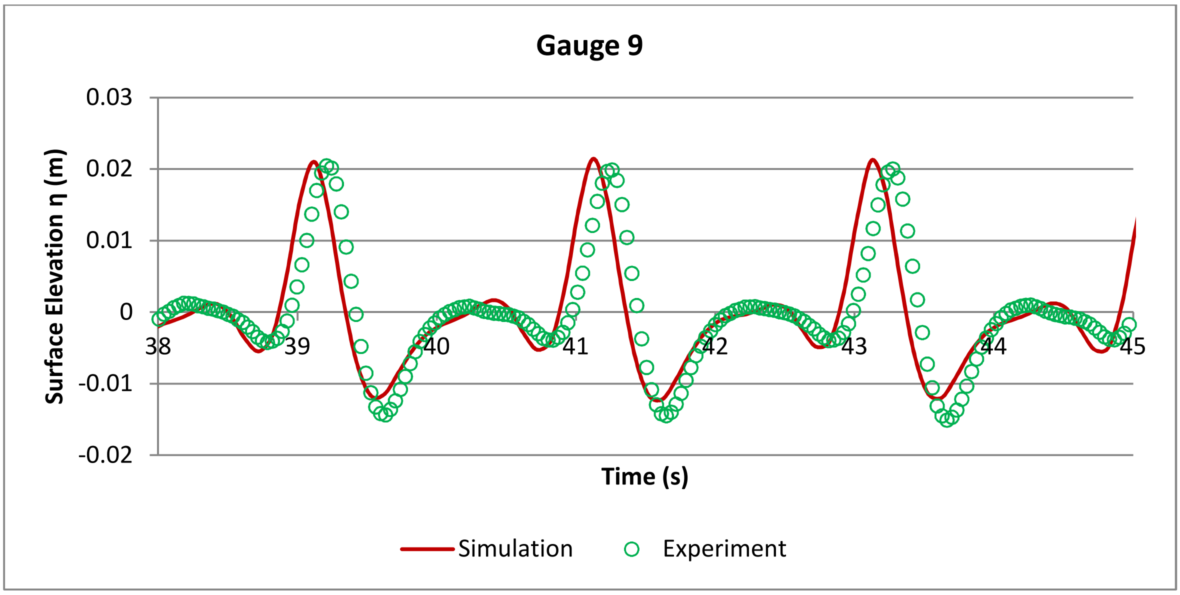


Figure 42. Comparison of experimental and ideal surface elevation for gauge 9, Scenario 2

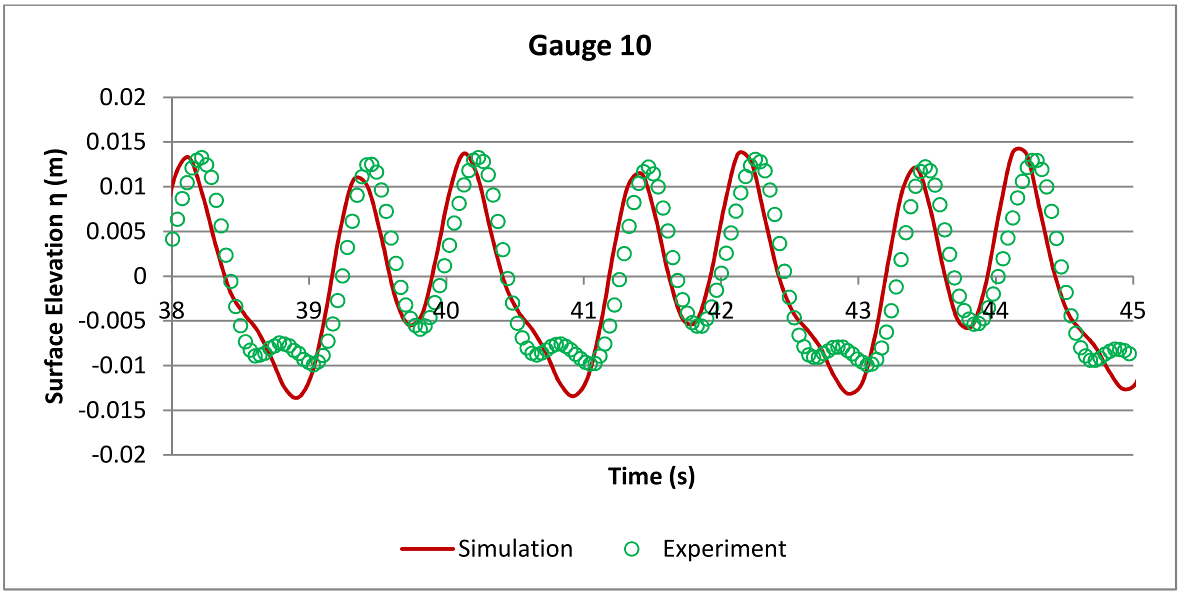


Figure 43. Comparison of experimental and ideal surface elevation for gauge 10, Scenario 2

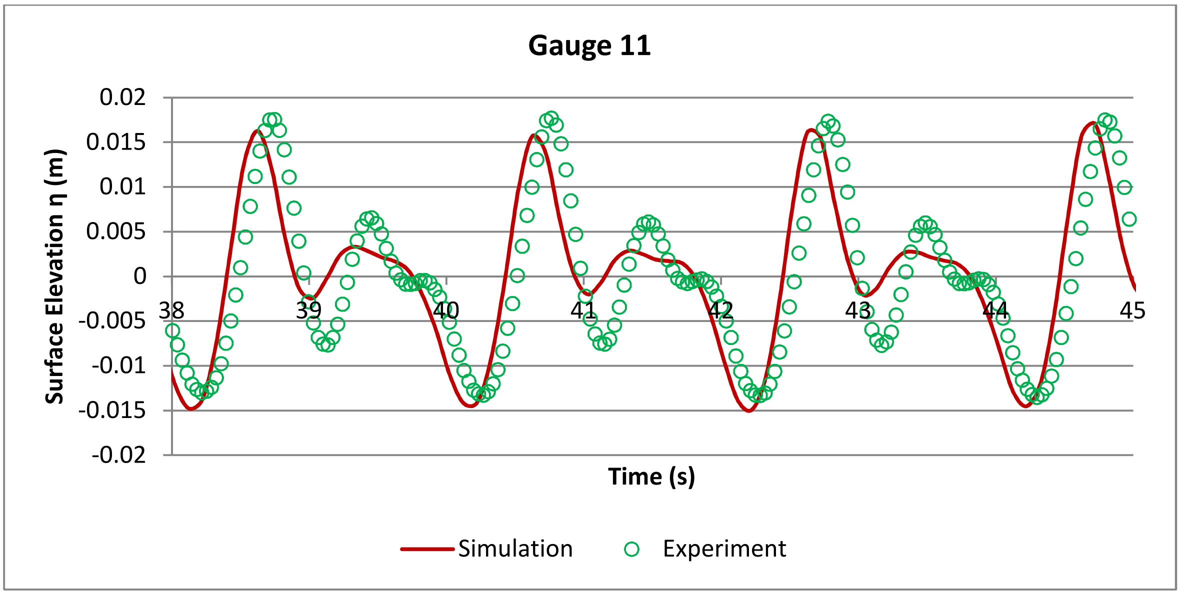


Figure 44. Comparison of experimental and ideal surface elevation for gauge 11, Scenario 2

It can be seen that results for the wave gauges located in front of the bar (gauges 1 to 4) closely match the surface elevation measured during the Dingemans (1994) experiments. Gauges 5 and 6 match the magnitude of the surface elevation but have some difficulty matching the secondary crest that develops from gauge 5.

Results begin to show variation that is more significant from gauge 7, located after the submerged bar. It has been noted by other authors, (including Morgan et al. (2010), Bai & Cheung (2011), Chazel et al. (2010)) that these gauges are the most difficult to reproduce by numerical models due to dispersive effects that become more pronounced by super-harmonics released after the submerged bar (Chazel et al., 2010).

The discrepancy between experimental results and numerical simulations of wave evolution behind the bar was the focus of a study by Shen & Chan (2011) who noted that numerical models keep a regular wave profile in front of the bar as higher harmonics stay in phase with the primary wave. Once the water begins to deepen (i.e. after the bar), the higher harmonics propagate as free waves. It is these free harmonic waves after the bar that the numerical model developed for this Master thesis has difficulty in modelling.

This limitation of the developed numerical model is evident in the results from gauge 7, where a delay of approximately 0.1 s can be seen. This may be due to incorrect simulation of the celerity (wave speed) of the harmonics that have been released after the slope. The results of gauges 7 to 9 display the 0.1 s time delay but the amplitude of the simulated surface elevation closely matches the experimental measurements. A delay in the simulated surface elevation of wave gauges behind the bar was also seen in Bai & Cheung (2011).

Gauge 10 and 11 exhibit the most significant variation from the experimental results with difficulty matching the surface elevation of the secondary crests as well as displaying the 0.1 s delay.

The mesh sensitivity test (see 5.6.2) indicates that improving the quality of the mesh by reducing cell volume and increasing the number of cells in the region after the submerged bar did not improve the simulated results. This is particularly evident through Meshes D, E and F. Additional refinement in the region behind the bar (Mesh D) made little difference to the results, as did improving refinement throughout the tank (Mesh E). In addition to this, improving the smoothness of the downward slope (Mesh F), also did not improve the results. Therefore, the delay of the simulated results cannot be attributed to the mesh.

The surfaces in the numerical model were assumed to be smooth, an assumption resulting from the use of the laminar simulation option. Roughness of the surface can be specified using the turbulence model, which was not included within the scope of this work. It is possible that the lack of specification of roughness and/or turbulence may have contributed to the discrepancy between the simulation and experimental results of wave gauges after the submerged bar.

6.3. Scenario 3: Simulation of regular waves against a slope

6.3.1. Production of various breaker types

Using the moving inlet boundary with OpenFOAM it was possible to produce three types of breakers that exist in the surf zone, spilling, plunging and surging (see 4.3.1). These three breaker types represent the three ranges of the surf similarity that can predict the breaker type (see 4.3.2). The surf similarity parameter of each breaker simulated is given Table 12 (based on geometry and input wave parameters). Snapshots of the breakers created are shown in Figure 45, Figure 46 and Figure 47. Note that in the figures only part of the wave tank is shown. In the figures the progression of the wave is marked by numbers, as was also shown in Figure 4.

Table 12. Surf similarity parameters of simulated breakers

Scenario	Breaker type	Surf similarity parameter ξ_0
3A	Spilling (Figure 45)	0.15
3B	Plunging (Figure 46)	0.91
3C	Surging (Figure 47)	3.64

Figure 45, Figure 46 and Figure 47 demonstrate that OpenFOAM is capable of simulating the correct type of breaking wave within the surf zone. In Figure 45, the spilling breaker, from step 4 aerated water can be seen from the top of the wave, followed by movement of this aerated water down the front face (seen in step 6). The continued movement of aerated water from steps 6 to 8 shows a similar behaviour to that displayed in the example spilling breaker in Figure 4 (4.3.1). The plunging breaker in Figure 46 displays the distinctive curling characteristic of a plunging breaker. Figure 47 demonstrates a surging breaker with no breaking of the crest of the wave.

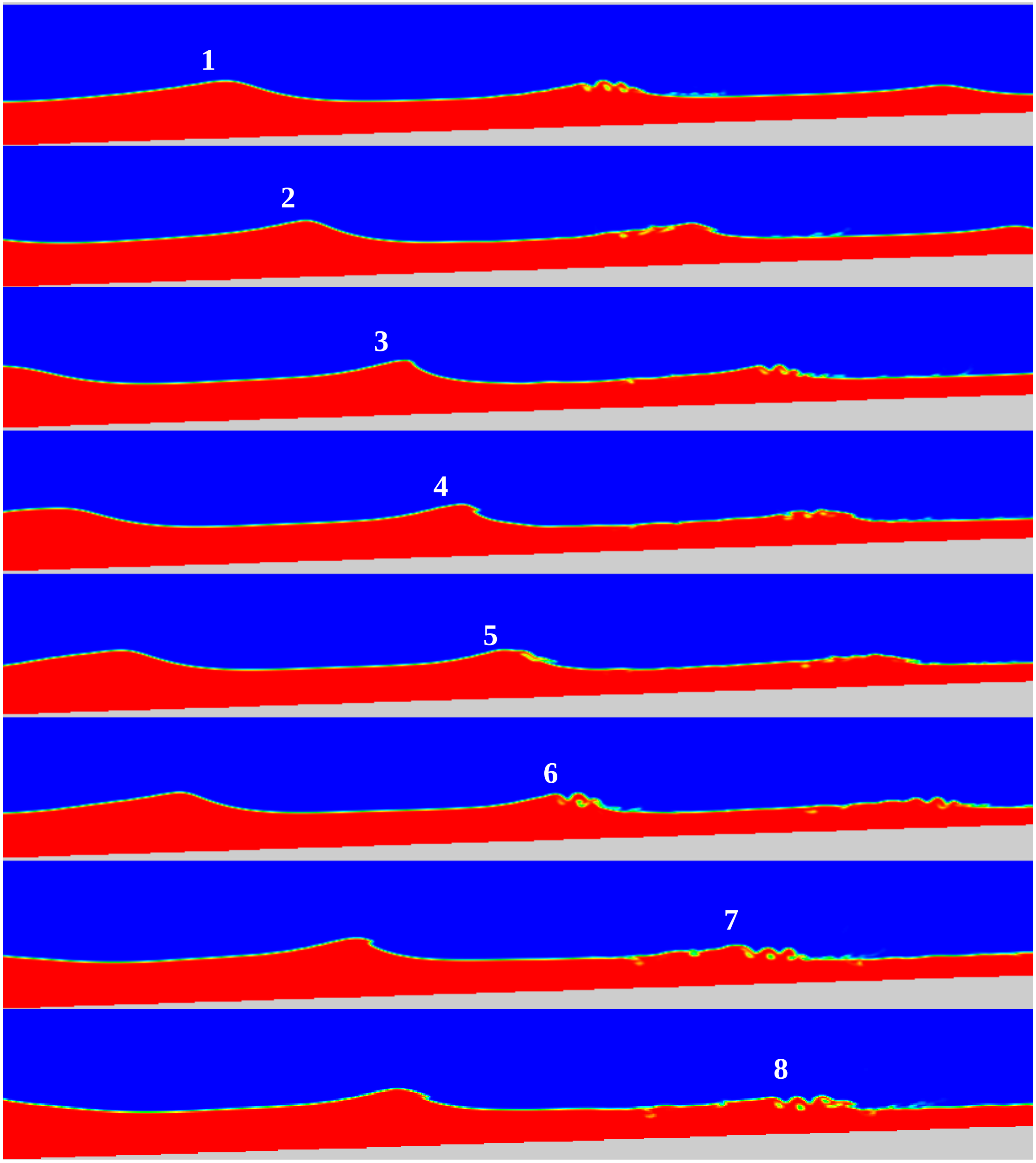


Figure 45. Scenario 3A: Formation of a spilling breaker using OpenFOAM

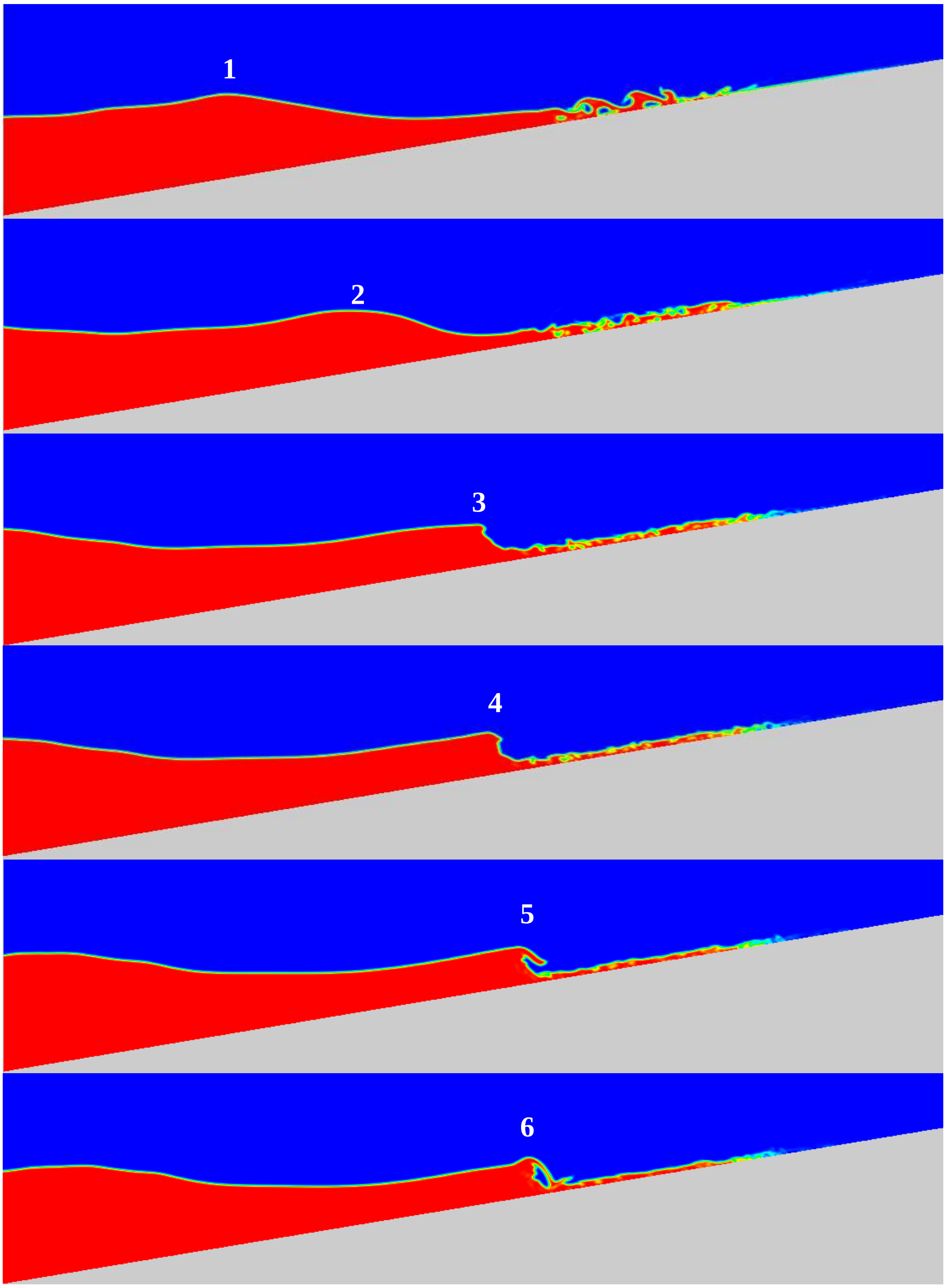


Figure 46. Scenario 3B: Formation of a plunging breaker using OpenFOAM

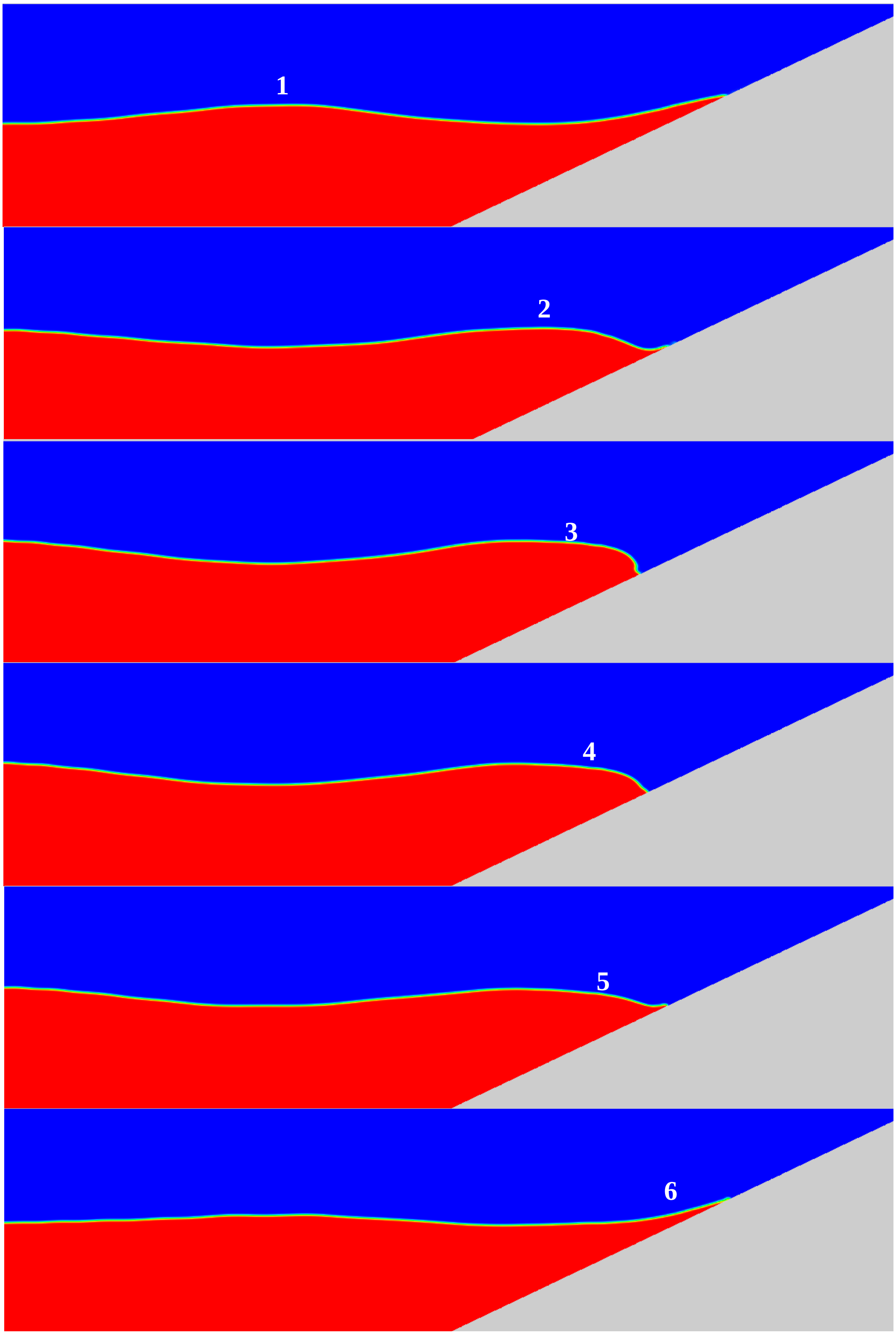


Figure 47. Scenario 3C: Formation of a surging breaker using OpenFOAM

6.3.2. Run-up of plunging breaker

Based on a simulation of 25 s using the wave parameters given in section 5.2 the average run-up simulated by OpenFOAM for Scenario 3B was determined. The first three run-ups were ignored to allow the simulation to ramp up. The measured run-up values are given in Table 13.

Table 13. Simulated run-up levels of plunging breaker, Scenario 3B

Run-up number	Run-up above SWL [m]
4	0.32
5	0.34
6	0.28
7	0.31
8	0.35
9	0.25
AVERAGE	0.31

These simulated run-up levels can be compared to the calculated theoretical range of 0.27 m to 0.34 m calculated using Equation (14), Equation (15) and the minimum and maximum sea state parameters. The results of the OpenFOAM simulation of Scenario 3B demonstrate that OpenFOAM can simulate wave run-up within the theoretical expected range.

6.4. Scenario 4: Demonstration of a floating object impacted by regular waves

Using the dynamic mesh option of OpenFOAM, as explained in 5.4.2, a floating object, with properties as defined in 5.5.3, was simulated to demonstrate that OpenFOAM can model a floating object under the action of regular waves. The results of Scenario 4 are shown as series of snapshots in Figure 48. Figure 48 clearly illustrates that the mesh moves with the floating object. The object is subject to action from the wave and is carried along in the *x* direction.

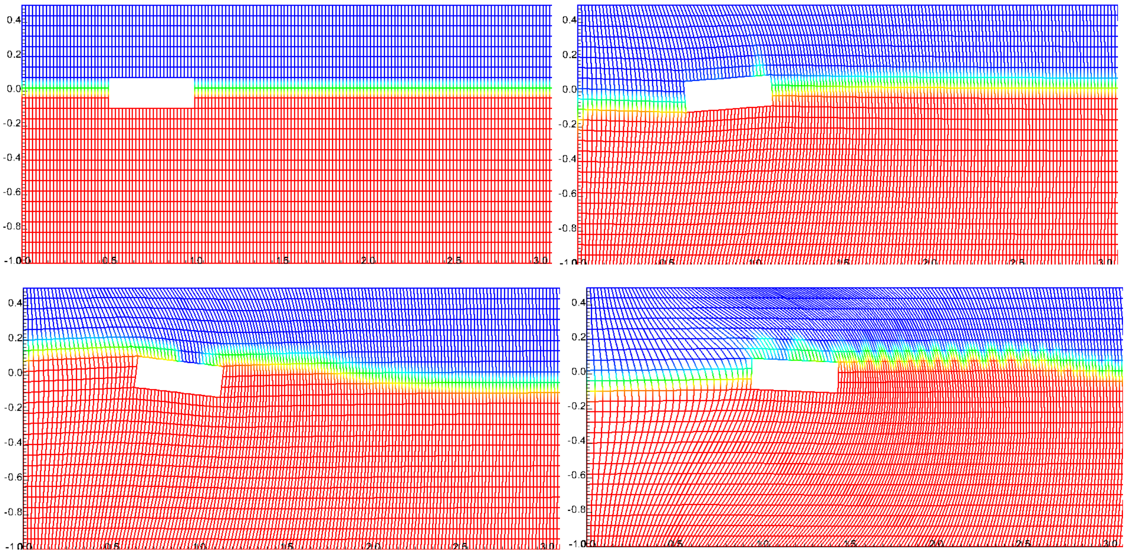


Figure 48. Floating object with dynamic mesh under the influence of regular waves. Red represents water, blue represents air and the black line identifies the air-water interface. The axes are in metres.

The cell size in the mesh used is quite large because of the difficulty of simulating a dynamic mesh. Any improvement in the mesh affected the ability of *interDyMFoam* to simulate. Furthermore, this simulation failed once the floating object attempted to overturn.

Nonetheless, Scenario 4 demonstrates that OpenFOAM is capable of simulating a floating object and the action resulting from regular waves. With further refinement of the mesh (for example using a sliding interface allowing the box to rotate 360°) and improvements of simulation parameters, OpenFOAM may be used to simulate and measure impacts on floating objects, as was achieved by Yong & Mian (2010).

7. CONCLUSIONS AND RECOMMENDATIONS

This work has demonstrated that OpenFOAM can be a valuable tool for the ocean engineering and wave energy industries. It has been shown that OpenFOAM can be used to successfully model a numerical wave tank if conditions and parameters of OpenFOAM are appropriately set.

This work determined that OpenFOAM should only be used to model regular waves with a steepness (H/L) less than or equal to 0.05. Above this steepness, waves may not match the ideal surface elevation, may be damped out along the tank and have been shown to break below the critical breaking value. This behaviour is due to an unwanted high air velocity that is caused by the VOF free surface tracking method. Possible solutions outlined by Afshar (2010) and Paterson (2008) are suggested for future work that aims to model waves with a steepness above 0.05.

The numerical wave tank developed was only used to model waves below the critical steepness of 0.05 and was shown to simulate within a good degree of accuracy when compared to the experimental results. Results in front of the submerged bar showed the strongest agreement with the experimental results. The results of wave gauges behind the submerged bar showed some minor disagreement compared to experimental results due to the higher harmonic waves that are released after the bar, a common issue for numerical models. The quality of the mesh was shown not be responsible for the delay exhibited in the simulation.

In addition to generating the behaviour of regular waves with a submerged bar it was also shown that OpenFOAM can simulate spilling, plunging and surging breaking waves over a smooth sloped beach. OpenFOAM was also able to simulate the wave correctly as predicted by the surf similarity parameter, which indicates what type of breaker will form. The run-up of a plunging wave was also simulated within the theoretical range of sea conditions. The ability of OpenFOAM to perform these tasks has not been demonstrated before.

Finally, OpenFOAM was shown to be able to model a floating object that moves in response to regular waves. While the simulation had some limitations, this is an important step towards modelling more complex scenarios, such as a wave energy device that can react in response to waves.

Based on the achievements and work of this Master thesis the author recommends some areas of improvement that may be targeted for future work. The numerical wave tank could be improved by developing a time independent method of preventing reflection, allowing longer simulation times. Further attention may also be given to improving the simulation of harmonics after the submerged bar. The ability of OpenFOAM to model turbulence should be investigated and the ability to model floating objects should also be improved, possibly by using a sliding interface to allow the floating object to overturn. Further experimental validation of the behaviour of the numerical wave tank may be possible if solitary waves are simulated and compared to existing experimental data sets (e.g. Hsiao et al., 2008).

In conclusion, this work has shown that OpenFOAM can be used to create a numerical wave tank that generates regular waves, that those waves are generated correctly and can interact with submerged bars and sloped beaches to a good degree of accuracy. It was also shown that a floating object can be modelled using a dynamic mesh. With further development of this work, this author believes that OpenFOAM has the potential to be used to model a floating wave energy device or floating coastal structure.

8. REFERENCES

Afshar, M.A., (2010), “Numerical Wave Generation in OpenFOAM”, Master Science Thesis, Department of Shipping and Marine Technology, Chalmers University of Technology, Sweden.

Bai, Y., Cheung, K.F., (2011), “Depth-integrated free-surface flow with a two-layer non-hydrostatic formulation”, International Journal for Numerical Methods in Fluids, doi: 10.1002/fld.2566.

Battjes, J.A., (1974a), “Surf Similarity”, Proceedings of the 14th International Conference on Coastal Engineering, Copenhagen, Denmark, pp. 466-480.

Battjes, J.A., (1974b), “Computation of set-up, longshore currents, run-up and overtopping due to wind-generated waves”, Report 74-2. Committee on Hydraulics, Dept. Of Civil Engineering, Delft Univ. Of Technology, Delft, The Netherlands.

Beirão, P., (26th November 2010), Presentation: “Wave Energy”, Renewable Energy Systems Course, University of Coimbra.

Beji, S., Battjes, J.A., (1993), “Experimental investigation of wave propagation over a bar”, Coastal Engineering, 19, pp. 151-162.

Bhuyan, G. S., (2008), “Harnessing the Power of the Oceans”, IEA Open Energy Technology Bulletin, 52.

Brito, A., (2009), “Portuguese Pilot Zone”, Presentation: Energiuka, Oslo, 4-5 February 2009.

Brito-Melo, A., Huckerby, J. (eds) (2011), “Implementing Agreement on Ocean Energy Systems Annual Report 2010”, IEA.

Chazel F., Benoit, M., Ern, A., (2010), “Validation of a double-layer Boussinesq-type model for highly nonlinear and dispersive waves”, Proceedings of the International Conference on Coastal Engineering 32, Shanghai, China.

Dean, R.G., Dalrymple, R.A., (1984), “Water Wave Mechanics for Engineers and Scientists”, Prentice-Hall, New Jersey.

Dean, R.G., Dalrymple, R.A., (1991), “Water Wave Mechanics for Engineers and Scientists”, World Scientific Press, Teaneck, New Jersey.

Dingemans, M., (1994), “Comparison of computations with Boussinesq-like models and laboratory measurements”, Mast-G8M technical report H1684, Delft Hydraulics, Delft, The Netherlands.

Fenton, J.D., (1990), “Non-linear Wave Theories”, in The Sea, Vol.9: Ocean Engineering Science, Le Méhauté., B., Hanes, D.M. (eds), Wiley, New York.

Galvin, C.J. Jr., (1968), “Breaker type classification on three laboratory beaches”, Journal of Geophysical Research, 73(12), pp. 3651-3659.

Hsiao, S., Hsu, T., Lin, T., Chang, Y., (2008), “On the evolution and run-up of breaking solitary waves on a mild sloping beach”, Coastal Engineering, 55, pp. 975-988.

Hughes, S.A., (2004), “Estimation of wave run-up on smooth, impermeable slopes using the wave momentum flux parameter”, Coastal Engineering, 51, pp. 1085-1104.

Hunt, I.A., (1959), “Design of seawalls and breakwaters”, Journal of the Waterways and Harbors Division, no. WW3. American Society of Civil Engineers, Vol. 85, pp. 123–152.

Iribarren, C.R., Nogales, C., (1949), “Protection des Ports”, XVIIth International Navigation Congress, Lisbon, Portugal, Section II, Communication, pp. 31–80.

Jasak, H., (2008), “Dynamic Mesh Handling in OpenFOAM”, 47th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, 5-8 January, Orlando, Florida, 2008 (AIAA 2009-341).

Koo, W.C., Kim, M.H., (2007), “Fully nonlinear wave-body interactions with surface-piercing bodies”, Ocean Engineering, 34(7), pp. 1000-1012.

Luth, H.R., Klopman, G., Kitou, N., (1994), “Kinematics of waves breaking partially on an offshore bar: LVD measurements for waves without a net offshore current”, Technical Report H1573, Delft Hydraulics, Delft, The Netherlands.

Ministério da Economia e da Inovação, (2007), “Decreto-Lei n.o 225/2007 de 31 de Maio”, Diário da República, 1.a série—No. 105.

Ministério da Defesa Nacional, (2008), “Decreto-Lei n.º 5/2008 de 8 de Janeiro”, Diário da República, 1.ª série — No. 5.

Mollison, D., Pontes, M.T., (1992), “Assessing the Portuguese wave-power resource”, Energy, 17, pp. 255-268.

Morgan G., Zang, J., Greaves, D., Heath, A., Whitlow, C., Young, J., (2010), “Using the rasInterFoam CFD model for wave transformation and coastal modelling”, ICCE No 32 (2010): Proceedings of 32nd Conference on Coastal Engineering, Shanghai, China, 2010.

Nojiri, N., Murayama, K., (1975), “A study on the drift force on two dimensional floating body in regular waves”, Transactions of the West-Japan Society Naval Architect, 51, pp. 131-152.

Ohyama, T., Kioka, W., Tada, A., (1995), “Applicability of numerical models to nonlinear dispersive waves”, Coastal Engineering, 24, pp. 294-313.

OpenFOAM, (2010), “User Guide Version 1.7.1”, Available from http://www.openfoam.com/archive/1.7.1/docs. Last accessed 16th December 2011.

Orszaghova, J., Borthwick, A.G.L., Taylor, P.H., (2012), “From the paddle to the beach – A Boussinesq shallow water numerical wave tank based on Madsen and Sørensen’s equations”, Journal of Computational Physics, 231(2), pp. 328-344.

Paterson, E., (2008), “Multiphase and free surface flow simulations”, Third OpenFOAM workshop, Politecnico di Milano, Milan, Italy, 10-11 July 2008 .

Paterson, E., Smith, K., Ford, S., (2009), “Simulation of Wakes, Wave Impact Loads and Seakeeping Using OpenFOAM”, Presentation at 4th OpenFOAM Workshop, Montreal, Canada, 1-4 June 2009.

Pelamis Wave Power (2011a), “Prototype during seatrials”, Available from http://www.pelamiswave.com/gallery?category=images&page=3 . Last accessed: 16th December 2011.

Pelamis Wave Power, (2011b), “E.ON P2 Pelamis Test Update – 14th May 2011”, Available from http://www.pelamiswave.com/news?archive=1&mm=5&yy=2011. Last accessed: 16th December 2011.

Richardson, J.E., (1996), “Surf Similarity”, Flow Science Technical Note FSI-96-00-TN44.

Senturk, U., (2011), “Modeling nonlinear waves in a numerical wave tank with localized meshless RBF method”, Computers & Fluids, 44(1), pp. 221-228.

Shen. L., Chan, E., (2011), “Numerical simulation of nonlinear dispersive waves propagating over a submerged bar by IB-VOF model”, Ocean Engineering, 38, pp. 319-328.

Smith, J.M., (2003), “Chapter II-4: Surf Zone Hydrodynamics”, *in* Coastal Engineering Manual EM 1110-2-1100, U. S. Army Corps of Engineers.

Ursell, F., (1953), “The Long-Wave Paradox in the Theory of Gravity Waves”, Mathematical Proceedings of the Cambridge Philosophical Society, 49, pp. 685-694.

Van der Meer, J.W., Stam, C.J.M., (1992), “Wave runup on smooth and rock slopes of coastal structures”, Journal of Waterway, Port, Coastal and Ocean Engineering, 118 (5), pp. 534-550.

Vincent, C.L., Demirbilek, Z., Weggel, J.R., (2002), "Chapter II-3: Estimation of nearshore waves", Coastal Engineering Manual (CEM), U.S. Army Corps of Engineers.

Williams, J.M., (1981), "Limiting gravity waves in water of finite depth", Philosophical Transactions of the Royal Society of London A, 302, pp.139-188.

Yong, L., Mian, L., (2010), "Wave-body interactions for a surface-piercing body in water of finite depth", Journal of Hydrodynamics, 22(6), pp. 745-752.

Zhao, B., Duan, W., (2010), "Fully nonlinear shallow water waves simulation using Green-Naghdi theory", Journal of Marine Science and Application, 9(1), pp. 1-7.