Mari Taranrød Storsul

A CFD Study of Numerical Wave Tanks

Master's thesis in Marine Technology
Supervisor: Kourosh Koushan
May 2019

Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Engineering
Department of Marine Technology

NTNU
Kunnskap for en bedre verden

SINTEF

Mari Taranrød Storsul

A CFD Study of Numerical Wave Tanks

Master's thesis in Marine Technology
Supervisor: Kourosh Koushan
May 2019

Norwegian University of Science and Technology
Faculty of Engineering
Department of Marine Technology

NTNU
Norwegian University of
Science and Technology

Abstract

Computational Fluid Dynamics (CFD) is increasingly used to study wave-structure interactions. Numerical wave tanks are used to model waves and investigate wave interactions. Stability and accuracy are essential to the performance of NWTs, and accurate wave generation are among the existing technical difficulties. This thesis present guidelines for wave generation with two different methods, and benchmark the simulated waves against analytic solutions. The applied CFD software, FINE/Marine, is based on Reynolds Averaged Navier-Stokes Equations, and it applies the Volume of Fluid method to model the free surface. Wave- and surface probes are used to capture the free surface elevation in time and space. Results indicate that a steepness criteria of $1/33$ must be applied using the guidelines, and then regular sinusoidal waves of high accuracy are simulated independent of wave generation method. Sponge layer efficiency highly depends on the wave generation method. Lastly, waves loose energy as they propagate down the NWT, and further investigations should be made regarding this.

KEY WORDS:

CFD, RANS Equations, VOF, NWT, Regular Waves, Irregular Waves, 2D, 3D, FINE/Marine

Sammendrag

Markedet i dag er preget av en stadig økende bruk av numerisk fluiddynamikk for å studere bølge-struktur interaksjoner. Numeriske bølgetanker blir brukt for å modellere bølger og undersøke interaksjoner. Stabilitet og nøyaktighet er essensielt for gode resultater, hvor bølgegenerering er blant de tekniske utfordringene. Denne masteroppgaven presenterer retningslinjer for bølgegenerering med to forskjellige metoder, og sammenligner simulerte bølger med analytiske løsninger. Programmet FINE/Marine, basert på Reynolds Averaged Navier-Stokes ligninger, og benytter Volume of Fluid metoden for å modellere fri væskeoverflate. Bølge- og overflate sensorer brukes for å måle bølgehevningene i tid og rom. Resultater indikerer at maksimum steilhet ved bruk av retningslinjene ikke bør overstige $1/33$, og da vil retningslinjene resultere i regulære sinusbølger av høy kvalitet uavhengig av bølgegenererings metode. Effekten av bølgedemping viste seg sterkt avhengig av brukt metode. Avslutningsvis mistet bølgene energi da de forplantet seg nedover den numeriske bølgetanken, og ytterlige undersøkelser bør gjøres for å forbedre dette.

Nøkkelord:

Numerisk fluiddynamikk, RANS ligninger, VOF, numerisk bølgetank, regulære bølger, irregulære bølger, 2D, 3D, FINE/Marine

Acknowledgments

Kourosh Koushan, my supervisor at NTNU and SINTEF Ocean, provided valueable guidance and feedback through this project thesis. These FINE/Marine simulations would not have been possible without the support of Eloïse Croonenborghs. Fengjian Jiang gave me invaluable help as a point of contact with NUMECA's support services. Finally, my sincere thanks to Sébastien Fouques for providing helpful wave literature.

Table of Contents

List of Figures

List of Tables

List of Equations

Nomenclature

Abbreviations

AGR	Adaptive Grid Refinement
BC	Boundary Condition
CFD	Computational Fluid Dynamics
CFL	Courant-Friedruch-Lewy
DNS	Direct Numerical Simulation
EMN	Equipe Modélisation Numérique
FANS	Favre Averaged Navier-Stokes
FFT	Fast Fourier Transform
FM	FINE/Marine
FS	Free Surface
FVM	Finite Volume Method
ITTC	International Towing Tank Conference
IWG	Internal Wave Generator
JONSWAP	JOint North Sea WAve Project
LES	Large Eddy Simulation
NS	Navier-Stokes
NWT	Numerical Wave Tank
PM	Pierson-Moskowits
RANS	Reynold Averaged Navier-Stokes
VOF	Volume Of Fluid
WD	Wave Damping
WG	Wave Generator
WP	Wave Probe

Symbols

ε_g	Limit for breaking waves
ε_j	Random phase angle of wave component number j
ε_T	Dissipation of turbulent energy
γ	Non-dimensional peak shape parameter
λ	Wave length
μ	Dynamic viscosity
ν	Kinematic viscosity
ω	Angular frequency
ω_P	Peak frequency
Φ	Velocity potential
ρ	Density
σ_S	Spectral width parameter
$\mathbf{V}$	Velocity vector
ζ	Wave elevation
ζ_a	Wave amplitude
a_1	x-component of acceleration
a_3	z-component of acceleration

A_γ	A normalizing factor in the J spectrum
c	Celerity
C_μ	Turbulence constant in the standard $k-\varepsilon$ model
E	Total wave energy
g	Gravitational acceleration
H	Wave height
h	Mean water depth
H_S	Significant wave height
$H_{1/3}$	Average height of the $1/3$ highest wave
k	Wave number
k_T	Turbulent kinetic energy
p_D	Dynamic pressure
$S(\omega)$	Wave spectrum
$S_{PM}(\omega)$	The Pierson-Moskowitz spectrum
T	Wave period
t	Time variable
T_P	Peak period
u	x-component of velocity
w	z-component of velocity
J	JONSWAP

1 Introduction

Ships, cage aquaculture, offshore wind turbines and offshore platforms are examples of structures that operate in oceanic environments, constantly under influence of complex forces from wind, current and waves. In particular, waves are important during a design phase, as it causes motions and loads. Experiments in wave tanks and flumes are among the most commonly used methods for wave research today. Acquiring wave tanks, constructing prototypes and testing are both time consuming and expensive. Furthermore, these costs are often associated with redesign and re-testing. Over the past decade scientists have worked on finding an alternative way of solving these problems, while maintaining the integrity. It has led to the creation of Numerical Wave Tanks (NWTs). Lately, Computational Fluid Dynamics (CFD) has been applied to generate NWTs with a various of different numerical modeling techniques [7].

Use of CFD applications for maritime structures have progressed significantly over the past decade. Integration of more user friendly interfaces, automation and increased computer power contributes to a wider scope of CFD applications. Seakeeping, sloshing and slamming are examples of applications that involve the interface between water and air. Methods like Volume of Fluid (VOF), two-phase flow and level set simulate the sharp interface. Modeling of propagating, and especially breaking waves, are of high interest to the research community, due to challenges with numerical diffusion and white foam.

Stability and accuracy are essential to the performance of NWTs, and even though it shows promising trends, accurate wave generation are among the technical difficulties. Therefore, it is necessary to explore appropriate wave generation methods, that can reproduce actual marine conditions. The notion remains that "the pilot is more important than the plane", as simulations are highly dependent on the software user [8]. Molin reviewed basic problems with numerical wave generation, and his quote from 2001 is still applicable: "... As a matter of fact, even though it may look simple, generating a Stokes regular wave in a tank is impossible." [9, p.525]. Furthermore, CFD models have shown applicable to wave impact physics, yet the random characteristics of hostile ocean environments remaines challenging. Accordingly, ongoing numerical improvements describes irregular sea states through wave spectrums [10, 11].

An example of a successful simulation of wave-structure interaction using CFD is shown in figure 1.0.1. Here, the CFD software FINE/Marine was applied to model an exact replica of the experimental setup in a physical towing tank, and the results compared [1].

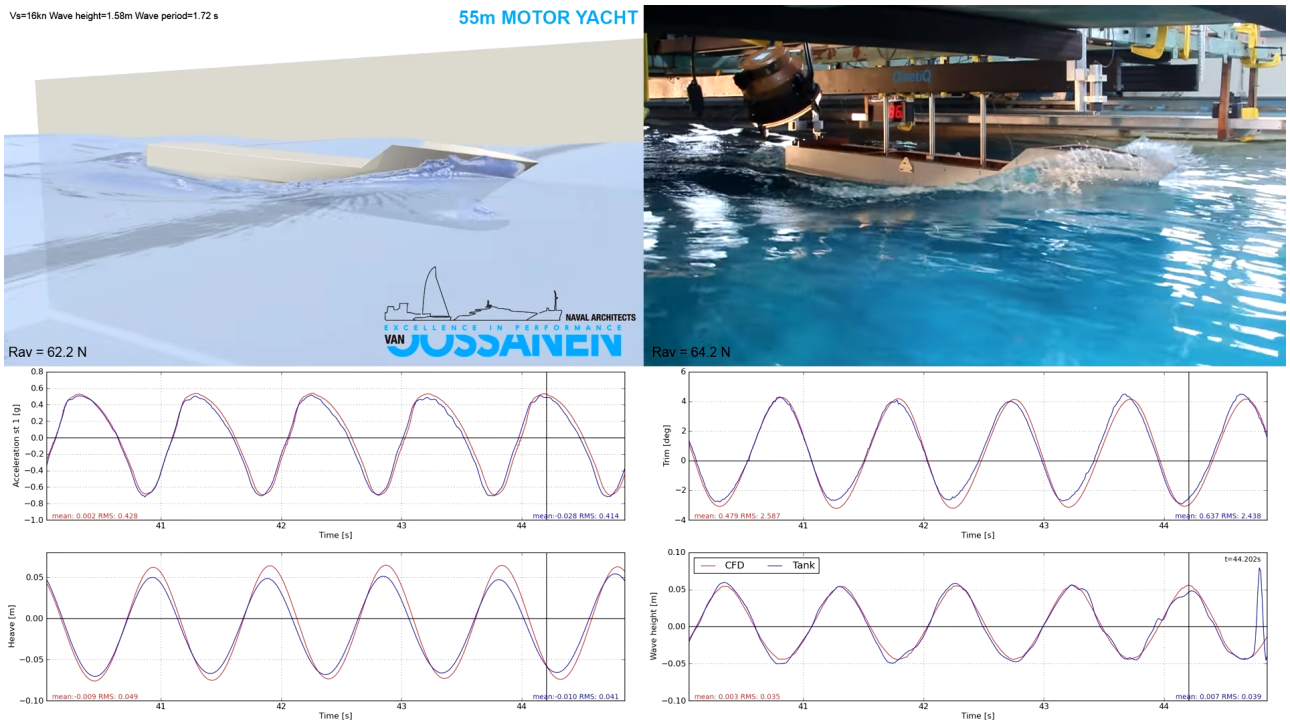


Figure 1.0.1: Snapshot of comparison between CFD and towing tank on a motor yacht, taken from the YouTube video at NUMECA's product page [1].

The aim of the present work are, therefore (1) Performing a series of dependency studies considering the effect of varying setup parameters of importance in full scale. (2) Develop guidelines that will allow wave-capturing presented with alternative approaches for wave generation. (3) Perform a validation of the simulated waves against numerical results to analytical solutions.

The thesis is organized as follows. Chapter 2 account for theory behind NWTs, including mathematical models. A literature study is presented in chapter 3. Both general model formulations and analysis approach are explained in chapter 4.1 and 4.2. Chapter 5 briefly present the results from the project thesis, as the work are continued in this master thesis. Analysis approach and results from the dependency studies are given in chapter 6 together with a discussion of the results. For the two-dimensional analysis of regular and irregular waves, the analysis approach, results and discussion are presented in respectively chapter 7 and 8. Method, results and belonging discussion for the three-dimensional analysis are presented in chapter 9. Further work is proposed in chapter 10.2. Finally, the conclusion is presented in chapter 10.1.

2 Theoretical Background

The current chapter presents relevant theory for computations and setup of the numerical wave tank. It is divided in three parts where the first is a review on wave theory needed for this thesis. Secondly, the formulation of the governing equations of fluid dynamics are presented together with additional methods used in the numerical simulation. Finally, theory and functionality related to the FINE/Marine CFD software are described briefly.

2.1 Waves

Ocean waves are mechanical waves that propagate along the free surface and are controlled by gravitation, therefore often called gravitational waves. They occur if forces act on the water, e.g. from wind, gravitation from the moon and sun, earthquakes etc. Wind is the main cause of ocean-waves, which typically have a periodicity of 15 [s]. Swells are far-traveled waves, where only long wave-periods (30 [s]) are preserved [12].

2.1.1 Basic Assumptions

Faltinsen [2] and Pettersen [13] described the basic assumptions for regular waves, irregular waves and wave spectrums presented in the following. There are four basic assumptions, which form the foundation of the velocity potential for waves. The velocity potential is convenient in analysis of fluid motion. The first assumption is that the Laplace Equation must be satisfied. Secondly, the normal velocity at the sea floor should be zero at finite water depths. The third and fourth assumptions are the non-linear kinematic boundary condition and dynamic free-surface condition respectively.

The first assumption is that the Laplace Equation (2.1.1) should be equal to zero. When the fluid is irrotational, the vorticity vector is zero everywhere in the fluid, i.e. $\omega = \nabla \times \mathbf{V}$. If the fluid is incompressible, i.e. $\nabla \cdot \mathbf{V} = 0$, it follows that the velocity potential is satisfied by the Laplace equation. Additional presumptions are that gravity is the only external force and that $z = 0$ is the mean free-surface level.

$$\frac{\partial^2 \Phi}{\partial x^2} + \frac{\partial^2 \Phi}{\partial y^2} + \frac{\partial^2 \Phi}{\partial z^2} = 0 \tag{2.1.1}$$

Secondly, the sea bottom condition is:

$$\frac{\partial \Phi}{\partial z} = 0 \quad on \quad z = -h \tag{2.1.2}$$

Kinematic boundary condition (2.1.3) require that the fluid particles remain on the free-surface.

$$\frac{\partial \zeta}{\partial t} + \frac{\partial \Phi}{\partial x}\frac{\partial \zeta}{\partial x} + \frac{\partial \Phi}{\partial y}\frac{\partial \zeta}{\partial y} - \frac{\partial \Phi}{\partial z} = 0 \quad on \quad z = \zeta(x,y,t) \tag{2.1.3}$$

The dynamic free-surface condition (2.1.4) require that the pressure in the fluid must be equal to air-pressure at the interface.

$$g\zeta + \frac{\partial \Phi}{\partial t} + \frac{1}{2}\left(\left(\frac{\partial \Phi}{\partial x}\right)^2 + \left(\frac{\partial \Phi}{\partial y}\right)^2 \left(\frac{\partial \Phi}{\partial z}\right)^2\right) = 0 \quad on \quad z = \zeta(x,y,t) \tag{2.1.4}$$

Both free-surface conditions (2.1.4 and 2.1.4) can be expressed by one equation, and if the velocity potential oscillates harmonically the equation is:

$$-\omega^2 \Phi + g\frac{\partial \Phi}{\partial z} = 0 \quad on \quad z = 0 \tag{2.1.5}$$

2.1.2 Regular Waves

A regular wave oscillate in time, and can be defined by a sine (or cosine) function. It is unambiguous by its amplitude (or height), wavelength (or period), propagation direction and phase at a given location and time [14].

Linear theory is valid when the wave amplitude is small in comparison to the wavelength, and a first order approximation can be applied to satisfy the free-surface condition. If the ratio of amplitude and length is large, non-linear effects narrows and elevates the wave crests. Wave troughs become wide and shallow. Stokes' expansion can be used to substantiate the non-linear effect in the wave equation. Additionally, linear theory tends to underpredict the wave height, e.g. for a steepness (2.1.6) of 0.1 exact theory predicts a 20% higher maximum wave elevation than linear theory.

$$\frac{H}{\lambda} \tag{2.1.6}$$

Both airy- and regular wave theory are frequently used terminologies for the same theory. It can be derived from the free-surface condition (2.1.5) together with the Laplace Equation (2.1.1) and bottom condition (2.1.2). By separation of variables the equations describing waves are derived. Results are presented in table 2.1.1 and illustrated in figure 2.1.1. The table differs between shallow- and deep

water waves because the wave behavior is different in the two scenarios. Especially note the important parameter $kh = 2\pi/L$ that express the relationship between water depth and wave height. Practical limits for deep and shallow water are respectively:

$$h \geq \frac{L}{2} \tag{2.1.7}$$

$$h \leq \frac{L}{20} \tag{2.1.8}$$

Pettersen [13] discussed the validity of Stokes' waves with the conclusion that they should not be applied for a steepness higher than $1/20$. ITTC recommends a value of around $1/50$ for tests with a ship model [15]. Waves steeper than approximately $1/7$ will break, and different wave breaking criteria are discussed in section 2.1.4.

Table 2.1.1: Velocity potential, dispersion relation, wave profile, pressure, velocity and acceleration for regular sinusoidal propagating waves on finite and infinite water depth according to linear theory, table 2.1 in Sea Loads [2, p.16].

Finite water depth		**Infinite water depth**	
$\Phi = \frac{g\zeta_a}{\omega} \frac{\cosh k(z+h)}{\cosh kh}$	(2.1.9)	$\Phi = \frac{g\zeta_a}{\omega} e^{kz} \cos(\omega t - kx)$	(2.1.10)
$\frac{\omega^2}{g} = k \tanh kh$	(2.1.11)	$k = \frac{\omega^2}{g}$	(2.1.12)
$\lambda = \frac{g}{2\pi} T^2 \tanh \frac{2\pi}{\lambda} h$	(2.1.13)	$\lambda = \frac{g}{2\pi} T^2$	(2.1.14)
$p_D = \rho g \zeta_a \frac{\cosh k(z+h)}{\cosh kh} \sin(\omega t - kx)$	(2.1.15)	$p_D = \rho g \zeta_a e^{kz} \sin(\omega t - kx)$	(2.1.16)
$u = \omega \zeta_a \frac{\cosh k(z+h)}{\sinh kh} \sin(\omega t - kx)$	(2.1.17)	$u = \omega \zeta_a e^{kz} \sin(\omega t - kx)$	(2.1.18)
$w = \omega \zeta_a \frac{\sinh k(z+h)}{\sinh kh} \cos(\omega t - kx)$	(2.1.19)	$w = \omega \zeta_a e^{kz} \cos(\omega t - kx)$	(2.1.20)
$a_1 = \omega^2 \zeta_a \frac{\cosh k(z+h)}{\sinh kh} \cos(\omega t - kx)$	(2.1.21)	$a_1 = \omega^2 \zeta_a e^{kz} \cos(\omega t - kx)$	(2.1.22)
$a_3 = -\omega^2 \zeta_a \frac{\sinh k(z+h)}{\sinh kh} \sin(\omega t - kx)$	(2.1.23)	$a_3 = -\omega^2 \zeta_a e^{kz} \sin(\omega t - kx)$	(2.1.24)

$\omega = 2\pi/T$, $k = 2\pi/\lambda$, T = wave period, λ = wavelength, ζ_a = wave amplitude, g = acceleration of gravity, t = time variable, x = direction of wave propagation, z = vertical coordinate z positive upwards , $z = 0$ mean waterlevel, h = average waterdepth. Total pressure in the fluid: $p_D - \rho g z + p_0$ (p_0 = atmospheric pressure.

$$c = \frac{\lambda}{T} = \frac{\omega}{k} \tag{2.1.25}$$

Celerity of a regular wave is given by equation 2.1.25. Waves in deep water are dispersive, which means that the celerity depend on the wavelength. Long waves have a higher celerity than short waves. Underneath deep water waves the water particles moves in circular paths with angular velocity ω and radius $\zeta_a e^{kz}$. For shallow water waves, the particles move in an ellipse. This movement create kinetic energy for the particle. The particle has potential energy due to a lift from its initial position. Total energy becomes:

$$E = \frac{1}{2}\rho g \zeta_a^2 \tag{2.1.26}$$

From the illustration in figure 2.1.1, it can be observed that the different physical variables have their respectful maximum at different time instants. Note that the dynamic pressure is negative value below the wave through and positive under the wave crest. The maximum absolute value of the horizontal velocity is either beneath a wave- through or crest as given by the acceleration.

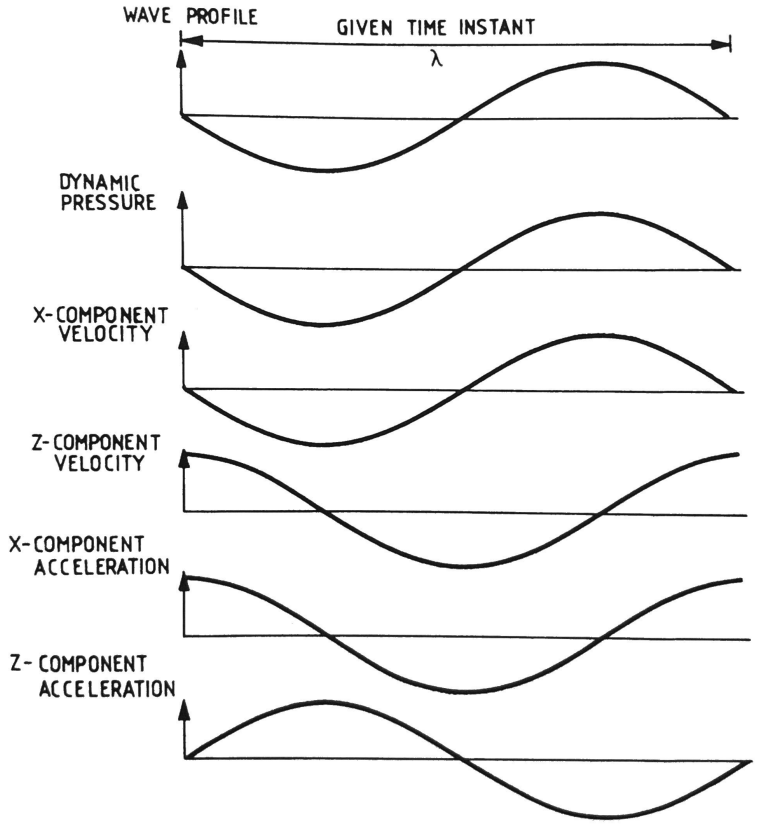


Figure 2.1.1: Wave elevation, pressure, velocity and acceleration in long-crested sinusoidal waves propagating along the positive x-axis (see table 2.1.1), figure 2.1 in Sea Loads [2, p.19].

Refraction of waves means that waves adjust its direction, length and amplitude as the water depth change. Wave diffraction is explained through interference. When two waves are combined the phase shifts are added, and thereby cause a total shift dependent on the phase difference. The phenomenon often occurs around objects that suddenly rises from the ocean, e.g. wave breaker or a light house [12].

2.1.3 Irregular Waves and Wave Spectra

Theory in this subsection is collected from literature written by Myrhaug [16], Lehn [12] and Faltinsen [2]. By observing a sea surface, the irregular or random characteristics of the waves are visible for the naked eye. Descriptions of irregular sea states are done statistically. By superimposing regular waves, random wave elevations can be composed, i.e. figure 2.1.2 and equation 2.1.27.

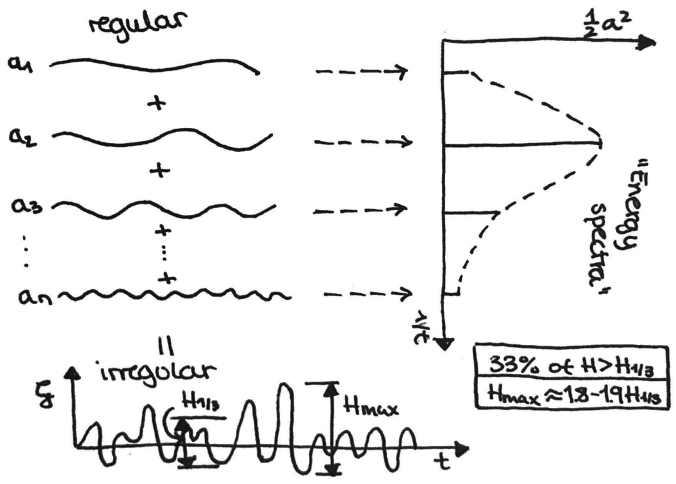


Figure 2.1.2: Example of superimposing regular waves into an irregular wave. In the figure the wave amplitude, otherwise expressed as ζ_a is expressed as a.

$$\zeta = \sum_{j=1}^{N} \zeta_{aj} \sin(\omega_j t - k_j x + \varepsilon_j) \tag{2.1.27}$$

ε_j is the random phase angle of wave component number j. The wave process is assumed stationary and ergodic within a short time frame, and the wave elevation normally distributed with zero mean and variance σ^2. Total energy in a sea state, found by adding the contributions from each linear wave component in a small frequency interval $\Delta\omega$, is given by:

$$\frac{E}{\rho g} = \sum_{j=1}^{N} \frac{1}{2} \zeta_{aj}^2 = \sum_{j=1}^{N} S(\omega_j) \Delta\omega \tag{2.1.28}$$

Figure 2.1.2 show a schematic illustration of each wave component's energy content. This is called a wave spectrum and is presented as a continuous curve. The energy spectrum of the wave elevation is denoted $S(\omega)$. The area under the graph is equal to the total energy in the wave, and it is denoted m_0. It is directly connected to the standard deviation of the wave elevation (2.1.29).

$$m_0 = \sum \frac{1}{2} \zeta_{aj}^2 = \sum \sigma_j^2 = \sigma^2 \tag{2.1.29}$$

Significant wave height, H_S, and the peak period, T_P, are parameters of interest when describing a sea state, and they are expressed by the moments of the wave spectrum (2.1.29). H_S, defined as

the average value of the $1/3$ highest waves, is expressed as $H_S = H_{m0} = 4\sqrt{m_0}$. Peak period are defined through the peak frequency of the spectrum ω_P. If ω_P is given in $[rad/s]$ the peak period are expressed as $T_P = 2\pi/\omega_P$.

In a design analysis the true spectrum of a location is unknown. Therefore the application of standardized wave spectrums are used. They depend on the weather conditions at specific locations, and cannot be expected to be valid. To select a spectrum, the frequency range should be taken under consideration. Additionally, the integrability and number of parameters should be evaluated. Pierson-Moskowitz, ITTC and JONSWAP are fully developed and frequently used spectrums. They are briefly described in thesis, and the connection between them explained.

PIERSON-MOSKOWITZ

The following theory is collected from DNV GL's recommended practice DNV-RP-H103 [3] and the book *Hydrodynamics of Offshore Structures* [17] from 1987. Pierson-Moskowitz (PM) spectrum was developed by the offshore industry for sea states in the North Atlantic generated by local winds. It assumes deep water, North Atlantic data, unlimited fetch, uni-directional seas and no swell. The spectrum is given by:

$$S_{PM}(\omega) = \frac{5}{16} H_S^2 \, \omega_P^4 \, \omega^{-5} \, e^{-\frac{5}{4}\left(\frac{\omega}{\omega_P}\right)^{-4}} \tag{2.1.30}$$

ITTC spectrum

A modification of the PM spectrum in terms of significant wave height and zero crossing frequency. The spectrum can be written as:

$$S_{ITTC}(\omega) = \alpha g^2 \omega^{-5} e^{-\frac{4\alpha g^2 \omega^{-4}}{H_S^2}} \tag{2.1.31}$$

where $\alpha = 0.0081/k^4$ and $k = \sqrt{g/\sigma}/3.54\omega_z$, with the average zero crossing frequency $\omega_z = \sqrt{m_2/m_0}$ and standard deviation $\sigma = \sqrt{m_0} = H_S/4$. For $k = 1$ this reduces to the one parameter PM spectrum. For a given significant wave height and mean period, the two spectrums will give the same results. The two spectrums can be compared by use of equation 2.1.32, which is the same form as the PM spectrum implying that the characteristic frequency for the ITTC sectrum is the same as the modified PM spectrum.

$$\omega_0 = 0.710\omega_z \tag{2.1.32}$$

JONSWAP

JONSWAP spectrum, $S_F(\omega)$ was developed as a modification for limited fetch of the PM spectrum. The validity range of the spectrum is defined in equation 2.1.33. $\gamma =$ non-dimensional peak shape parameter, $\sigma_S =$ spectral width parameter and $A_\gamma = 1 - 0.287 \ln \gamma$ is a normalizing factor. $\sigma_S = \sigma_{Sa}$ for $\omega \leq \omega_P$ and $\sigma_S = \sigma_{Sb}$ for $\omega \geq \omega_P$. Average values for the experimental data are $\gamma = 3.3$, $\sigma_{Sa} = 0.07$ and $\sigma_{Sb} = 0.09$. For $\gamma = 1$ the JONSWAP spectrum is equal to the PM spectrum.

$$S_F(\omega) = A_\gamma S_{PM}(\omega)\, \gamma^{e^{-\frac{1}{2}\left(\frac{\omega-\omega_P}{\sigma_S \omega - P}\right)^2}} \tag{2.1.33}$$

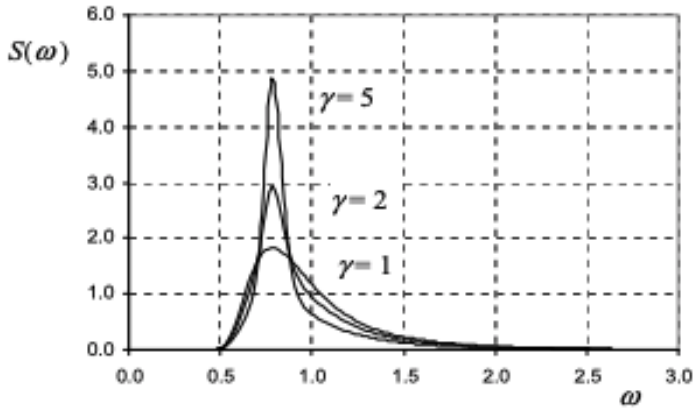


Figure 2.1.3: JONSWAP spectrum for $H_S = 4.0$ [m] and $T_P = 8.0$ [s] for $\gamma = 1$, $\gamma = 2$ and $\gamma = 5$. Figure 2-2 in DNV-RP-H103 [3, p.15].

2.1.4 Wave Breaking Criteria

Defining a simple and precise breaking criteria is not an easy task, and several papers have tried to view this topic. Steep waves are often associated with wave breaking. Stokes established a set of criteria to initiate wave-breaking [4]:

- Particle velocity at the crest equals phase velocity.
- 120° sharp point angle at the crest.
- Wave height to wave length approximately $1/7$.
- Particle acceleration equals 0.5g at the crest.

Massel [4] collected experimental data from different papers to study the limiting wave heights in comparison to Stokes limit. All the experiments had lower heights than from the Stokes limit, as illustrated in figure 2.1.4. The limiting value found by best fit to data was $\varepsilon_g = 0.019$, where ε_g are defined as

$$\varepsilon_g = \frac{H}{gT^2}. \tag{2.1.34}$$

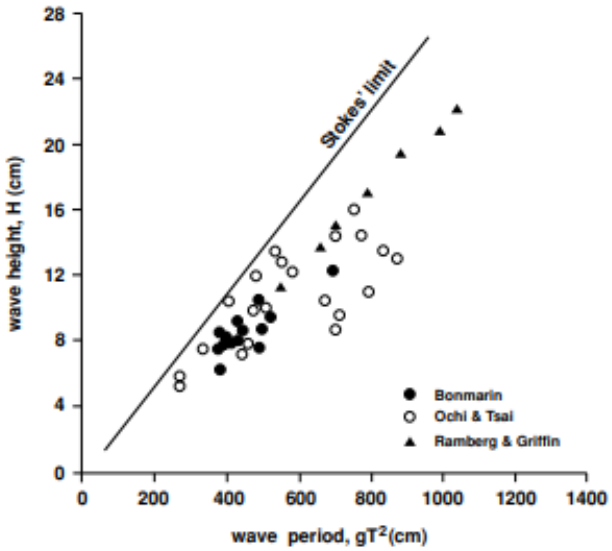


Figure 2.1.4: Limiting wave height H as a function of the period parameter gT^2, figure 4.18 [4, p.117].

2.2 CFD

Computational Fluid Dynamics, with abbreviation CFD, is the numerical study of fluids in motion or at rest. Physics of the fluid can be described through the fundamental equations called the governing equations in CFD. These are solved iteratively in each element of the discretized domain. Theory in this section is collected from the course *MECH4620 Computational Fluid Dynamics* at the University of New South Whales [18] and CFD Online [19].

A CFD analysis consist mainly of three parts; pre-processing, the solver and post-processing. Creation of geometry, mesh generation, material properties and boundary conditions are included in the pre-processing stage. The solver part contain transport equations, physical models and solver settings. Turbulence, combustion, radiation and other physical processes are included within the physical models. In solver settings the following are decided: initialization, solution control, monitoring solutions and convergence criteria. Post-processing is the visualization and processing of the results included contour-, streamline- and vector plots.

2.2.1 Governing Equations

As previously mentioned, the governing equations of fluid dynamics are the main foundation of CFD. Physical boundary conditions must be stated numerically to get an accurate result, and are a decisive part of the simulation. The Navier-Stokes Equation (NS) given in equation 2.2.1 describe the fluid motion in x-direction.

$$\begin{aligned}\underbrace{\frac{Du}{Dt}}_{\text{acceleration}} &= \underbrace{\frac{\partial u}{\partial t}}_{\text{local acceleration}} + \underbrace{u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+w\frac{\partial u}{\partial z}}_{\text{advection}} \\ &= -\underbrace{\frac{1}{\rho}\frac{\partial p}{\partial x}}_{\text{pressure gradient}} + \underbrace{v\frac{\partial^2 u}{\partial x^2}+v\frac{\partial^2 u}{\partial y^2}+v\frac{\partial^2 u}{\partial z^2}}_{\text{diffusion}}\end{aligned} \tag{2.2.1}$$

Different methods of solving a CFD problem, and especially turbulence, are through Reynolds Averaged Navier-Stokes Equations (RANS), Favre Averaged Navier-Stokes Equations (FANS), Large Eddy Simulation (LES) and Direct Numerical Simulation (DNS). They all have different model complexity and hereby different requirements for computational effort. DNS is free of modeling which provide accurate solutions with a high level of details about the current process. It needs a high level of data resources, and therefore the use is limited to small physical domains. LES has a medium demand for computer resources as it accurately solve the large eddies, and model small fluctuations. RANS and FANS have low computational effort, but it require additional turbulence models. Subsection 2.2.2 briefly describe what turbulence is and some approaches to turbulence modeling.

Turbulent flows can be described by a mean value and its statistical fluctuating component. NS Equations are averaged either by time-averaging (RANS) or by removing the fluctuating density component (FANS). This means that the output of the RANS Equations (2.2.2) are the Reynold averaged velocity and pressure. The fluctuating component in the RANS Equations must be modeled by a turbulence model. One of these terms are the Reynolds Stresses, which is the main root to problems regarding modeling of turbulence. This is called the *Turbulence Closure Problem* and describe the scenario where the averaged equations have too many unknowns and not enough equations.

$$\begin{aligned}\rho\left[\frac{\partial}{\partial x}(\overline{u}^2)+\frac{\partial}{\partial y}(\overline{uv})+\frac{\partial}{\partial z}(\overline{uw})\right] = &-\frac{\partial \overline{p}}{\partial x}+\mu\left[\frac{\partial^2 \overline{u}}{\partial x^2}+\frac{\partial^2 \overline{u}}{\partial y^2}+\frac{\partial^2 \overline{u}}{\partial x^2}\right] \\ &-\underbrace{\left[\frac{\partial}{\partial x}\left(\rho\overline{u'^2}\right)+\frac{\partial}{\partial y}\left(\rho\overline{u'v'}\right)+\frac{\partial}{\partial z}\left(\rho\overline{u'w'}\right)\right]}_{\text{Reynolds stresses}}\end{aligned} \tag{2.2.2}$$

2.2.2 Turbulence

Turbulent flows are associated with the existence of random fluctuations in the fluid. In CFD they are modeled by a mean value and its corresponding statistical fluctuating component. Some of the most frequently used models, briefly described in this thesis, are the Standard $k-\varepsilon$ model, Standard $k-\omega$ model and Shear Stress Transport (SST) model. Turbulent flows are characterized by:

1. They are random, or irregular, and the only way of quantifying the characteristics are by statistical methods.
2. Turbulence are diffusive. Rates of momentum, heat and mass transfer are increased by the presence of rigorous mixing.

3. Turbulent flows have a large Reynolds number.
4. The phenomenon of turbulence is rotational and three-dimensional and vorticity are important in the description of turbulent flows.
5. Internal energy are increased at the expense of kinetic energy due to work done by viscous shear stresses. This is called dissipation, and turbulence needs a continuous supply of energy to compensate for the viscous losses.
6. The equations of fluid dynamics govern the flow since turbulence is a continuum phenomenon.

Standard k-ε model

In the k-ε model turbulent viscosity can be evaluated as $\mu_T = \overline{\rho} C_\mu \frac{k_T^2}{\varepsilon_T}$, where C_μ is a turbulence constant, ε_T is the dissipation of turbulent energy and k_T turbulent kinetic energy. The expression for the turbulent viscosity can be included in the transport equations by substituting the Reynolds Stress expression into the conservation equation. Additionally, two transport equations are required for the $k-\varepsilon$ model. Flow behavior is assumed isotropic and the model is applied to fully turbulent flows with high Reynolds number. Positive aspects of the model are that it is robust, efficient and easy to apply. Its free-stream solutions are accurate, but for complex flows involving severe pressure gradients, separation or strong streamline curvature the results are poor. It also over-predicts the turbulence kinetic energy at wall regions.

Standard k-ω model

The Standard $k-\omega$ model is frequently used on boundary layer flows, and it is has superior performance for near-wall treatment. It requires a very fine mesh resolution. Free-stream flows are not a suited for the application of the model, since it under-predicts the turbulent kinetic energy.

SST k-ω model

Shear Stress Transport model, commonly called the SST model, combines the best features of the $k-\varepsilon$ and $k-\omega$ models. The outer region is solved by the former model and the inner by the latter. This is done by blending functions which include the distance from the wall, and it determine the success of the model.

2.2.3 Spatial Discretization

To solve the partial differential equations that describe the physics of fluid flows, the fluid domains are split into smaller cells. The governing equations are discretized and solved in each cell. There is an art in meshing, and it is important to define a good mesh from the start to ensure an accurate solution. The mesh should be refined enough to not influence the results. Recommendations from lectures in *MECH4620 CFD* stated to always use a structured mesh to resolve boundary layers and in locations where the flow gradients will be strongest. A structured mesh can be characterized by regular connectivity while an unstructured mesh has irregular connectivity. Elements common in a

two-dimensional mesh are triangles or rectangles, and tetrahedral or bricks in a three-dimensional grid.

Before running a simulation the mesh should be checked for its characteristic features, e.g. skew, negative volumes, concave cells, twisted cells, orthogonality and aspect ratio. A mesh convergence study is crucial to ensure mesh convergence and thereby an accurate solution. To study the convergence, and ensuring a steady simulation, the residuals, monitor points of interest and domain imbalance should be evaluated.

2.2.4 Temporal Discretization

Time discretization is just as important as spatial discretization to ensure numerical stability. A time integration can be either implicit or explicit. Either way, a certain stability criteria needs to be fulfilled. The Courant-Friedrichs-Lewy (CFL) number are often used as a criteria to maintain stability. During a simulation the maximum CFL number should be kept significantly lower than unity. By adjusting the time step this can be maintained.

2.2.5 Numerical Diffusion

Numerical diffusion occurs due to advective and diffusive fluxes across cell faces in a mesh. A flux describe the amount of property that crosses an area per unit of time. In the NS Equations (2.2.1) the advection and diffusion terms are marked. Instabilities and numerical diffusion are consequences of violating physical principles. Discretization schemes play an important role in numerical diffusion. If a spatial discretization scheme assume uniform concentration inside the control volume, numerical diffusion occur if the assumption is false. Coefficients multiplying variables cannot be negative due to physics, but this can occur in numerics. Central differences, implicit method and quick method could be applied to avoid diffusion. Boundary conditions are very important since the advection term include a first derivative and the diffusion term a second order derivative. In case of a free-surface boundary, the fluxes are of high importance.

2.2.6 Volume of Fluid

Theory described in this subsection is collected from the article written by Hirt and Nichols in 1979 [20]. The Volume of Fluid (VOF) method is a numerical method used to model free-surface boundaries. It has an automatic treatment of intersecting boundaries by a function that varies between unity and zero. A cell filled with water corresponds to unity, an air-filled cell is equal to zero, and all values in between correspond to partially filled cells. VOF is without numerical smoothing and provide a coarse interface. Stored information is kept to a minimum.

2.3 FINE/Marine

All theory stated in this section is collected from the FINE/Marine 7.2 (FM) theory guide [6] and the NUMECA user manual [5] found in the FM documentation. FINE/Marine is a CFD tool created by NUMECA that stands for "Flow INtegrated Environment for computations on unstructured hexahedral meshes dedicated to Marine applications" [5, p.12].

Three software systems have been integrated in the FM interface: HEXPRESS, ISIS-CFD and CFView. Each software is developed to solve a part of the CFD analysis. Four relevant features are: (1) Wave Generator (WG) (2) Internal Wave Generator (IWG) (3) Wave Damping (WD) (4) Adaptive Grid Refinement (AGR).

2.3.1 HEXPRESS

Theory in this subsection is collected from the HEXPRESS 7.2 User Guide [21]. HEXPRESS is a mesh generator software that follows a top-down approach meaning that it starts from an initial mesh, adapt to geometry, snap to geometry, optimize and finally put on viscous layers. CAD models can be imported directly into HEXPRESS, or it is possible to use the simple built in CAD manipulation functions. HEXPRESS is designed with algorithms for mesh optimization and automation to reduce the users interaction to a minimum. One decision the user needs to make is either structured or unstructured mesh as explained in section 2.2.3. In the following, the different steps of the mesh generation are briefly explained. HEXPRESS generates a hexahedral mesh of the computational domain in *initial mesh*. Under *adapt to geometry* the user can change the refinement through boxes and surface refinement. *Snap to geometry* projects the created mesh to the geometry and *optimize* ensures that the elements in the mesh are of high quality. The final step of *viscous layers* insert layers with high aspect ratios by subdividing the cells in the defined layer.

2.3.2 ISIS-CFD

FINE/Marine use the ISIS-CFD flow solver developed by Equipe Modélisation Numérique (EMN). It uses the incompressible unsteady RANS equations. Spatial discretization of the transport equations are based on the finite volume method (FVM). Additional models included transport equations are needed in case of turbulent flows, as briefly discussed in section 2.2.2.

Waves can be modeled with the ISIS-CFD solver by an external boundary. Volume fraction is used to model the wave height that varies in time. A velocity field is imposed at the boundary. The simulation method is not equal to a wavemaker and the flow does not exactly correspond to the wave train in the flow domain.

2.3.3 CFView

CFView is the flow visualization tool integrated in FINE/Marine. It allows a multi-window environment for two- and three-dimensional plotting. The solver interpolates the flow properties over the cells, and CFView is used for the post-processing. It provides the option of numerical probes where flow features can be investigated. Among the strongest features provided in CFView we find extraction of local values and quantity distribution along a curve for detailed analysis in specific regions.

2.3.4 Boundary Conditions

During the grid generation in HEXPRESS there are options to choose between six different boundary conditions: inlet, outlet, solid, external, mirror or full non-matching. To match the desired flow, the solid wall boundary condition gives the following options: slip, no slip, wall-function and synthetic jet. The slip wall boundary conditions correspond to zero shear stress at the wall. Here turbulent effects are neglected and the tangential velocity-component can be different from zero.

The external boundary condition treats both pressure and velocity conditions. It can also be used to impose waves as a wave generator. In this thesis far field and prescribed pressure has been applied. By using the far field type it is possible to prescribe velocity, mass fraction and turbulence by either entering constant or using default values. Dirichlet or Neuman condition is applied depending on the local flow, and the code will automatically adapt to the right condition. Another possibility with this boundary condition type is to define the far field manually by specifying the profile data. The type called prescribed pressure is a Dirichlet boundary condition. A Dirichlet boundary condition specifies the value that a solution need to take on along the boundary of a domain. Often, it is referred to as a fixed boundary condition. In FINE/Marine two types of conditions can be applied: updated hydrostatic pressure and frozen pressure.

Updated hydrostatic pressure is recommended for top and bottom patches of the domain with multi-fluid flow for boat simulations [5]. The pressure value is set to be $-\rho_{fluid} g(y(t) - y_0)$, and it will evolve during the computation according to the mesh and free surface position. Fluid is free to both enter and exit when using this boundary condition.

2.3.5 Wave Modelling

The ISIS-CFD enables wave field generation by imposing a velocity field u and water height which vary in time. Either Stokes waves of first to third order, or wave spectra based on first-order Airy wave components can be applied. Note that it is not equal to a physical wavemaker, and some transient waves are expected close to the wavemaker.

Wave Generator

Through the external boundary condition, the Wave Generator can be enabled. It can generate either regular or irregular waves. The regular waves are based on Stokes wave theory. Figure 2.3.1 show the schematic definition of waves used in FM. D is a positive value and represents the water depth, H the height of the wave and L is the wavelength. Wave statistics can be defined either through the length

or the period, then the dispersion relation is known. The value of D decides whether the simulation represents deep water ($D > L/2$), intermediate ($L/20 < D < L/2$) or shallow water ($D < L/20$). Direction of propagation is the direction the waves propagate. After deciding D, H and L, the wave order can be estimated or chosen. If the estimate button is used, the wave order is computed according to Le Méhaute.

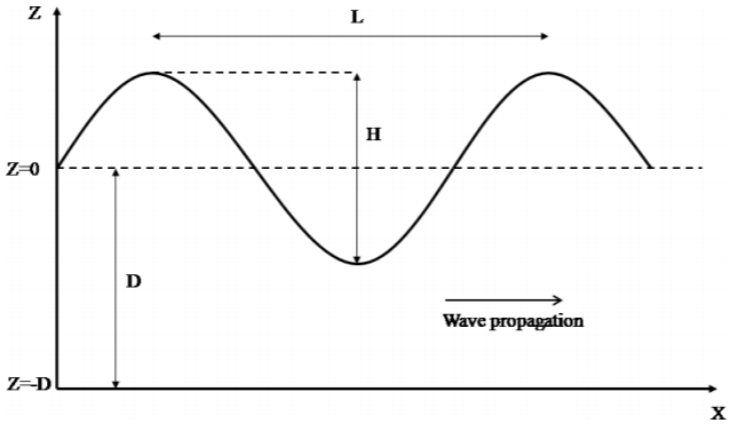


Figure 2.3.1: Schematic wave definition used in FINE/Marine, figure 8.18 in the user manual [5, p.223].

As mentioned in section 2.1.2, linear waves should be modeled by a sinusoidal surface profile with small amplitude and steepness. NUMECAs' user manual gives some pointers on best practice for use of the wave generator. The wave generator information tool computes useful information of the wave generation setup. Time configuration should be unsteady, maximum number of non-linear iterations 20 and convergence criteria 2.

First order Stokes waves is based on Airy theory, and the description of wave motion on the free surface is based on potential theory. The free-surface elevation is generated for the 1D following equation:

$$\eta(x,t) = -a\sin(kx - \omega t) \tag{2.3.1}$$

where a is the wave amplitude, k is the wave number and ω is the angular frquency. The velocity potential is found by applying the Laplace equation, and the velocity components are obtained by derivation, as described in section 2.1.2.

Irregular waves are implemented in FM through wave spectrums. The theory behind wave spectrums in general is described in section 2.1.3. ITTC, JONSWAP, JONSWAP 3 Parameters and Pierson-Moskowitz are commonly used spectrum that can be selected. As contrary to regular waves it is recommended to use adaptive grid refinement, and be very careful with the refinement of the free-surface.

Modeling irregular waves are done by superimposing regular waves with different amplitudes and phases. They are described through the relation between amplitude and frequency, and the defining variables are the significant wave height and characteristic period. A discrete number of wave components are needed for numerical approximations.

Internal Wave Generator

Where the WG creates waves at the defined boundary condition, the IWG create waves within the domain. It add a momentum source term in the NS equations. This is a more natural way of creating waves and it ensures a more clear wave signal. A sponge layer should be used as a numerical beach at end boundaries if this method is applied. IWG will generate waves that propagate along the x-axis, either in positive-, negative- or both directions. It has been verified for fixed domains without any moving bodies.

The length of the generator is 1 wavelength and it is centered at the source point defined by the user. Probes should be placed at least 1.5 wavelength from the source point. Sponge layers at each end of the domain are recommended to be around 3 wavelengths.

Using the IWG has some limitations pointed out by NUMECA. Restarting a computation is not recommended. Irregular waves can not be generated in both directions, i.e. negative and positive direction. Lastly, the IWG has only been validated without a moving body and for fixed domains [5].

2.3.6 Wave Damping

To avoid reflection in the fluid domain, sponge layers can be applied. They work as a numerical beach and damp the free surface elevation. It uses Darcy's law to damp the momentum in z-direction. It can be enabled through under Additional models in the FINE/Marine GUI. Another method of damping waves is to use *Multifluid Smoothing*, but it is not recommended in the user manual [5]. A length of minimum 3 wavelengths is recommended in the theory guide [6]. Parameters, S_{min} and S_{max}, are manually entered to specify the lengths of the damping areas.

2.3.7 Adaptive Grid Refinement

In FINE/Marine completely parallelised grid adaption is possible. The procedure is called during the flow computation, where firstly a refinement criterion is calculated and secondly the grid refined based on the criterion. Each refinement means dividing the cell in four in 2D and eight in 3D. The water surface only need a fine grid in one direction, and the total number of cells can thus be reduced by applying this method. The directional refinement needs a criterion to specify the cell size in different directions. In FM there are many options, and in this thesis the free surface criterion and free surface tensor criterion are briefly described.

The free surface criterion refines the mesh normal to the free surface. This is done by directional refinement when the surface is aligned with the mesh and isotropic refinement when the surface is at an angle to the mesh. Figure 2.3.2 illustrate this. The refinement treshold is the value that correspond to the desired grid cell size. At least two buffer layers are advised since the mesh must be refined in a zone around the free surface, and this is not done automatically. Free surface tensor criterion is similar to the above-mentioned, but it is computed differently. It uses matrices and eigenvalues associated with vector directions. For unsteady flow with wave breaking it is advised to use the tensor criterion.

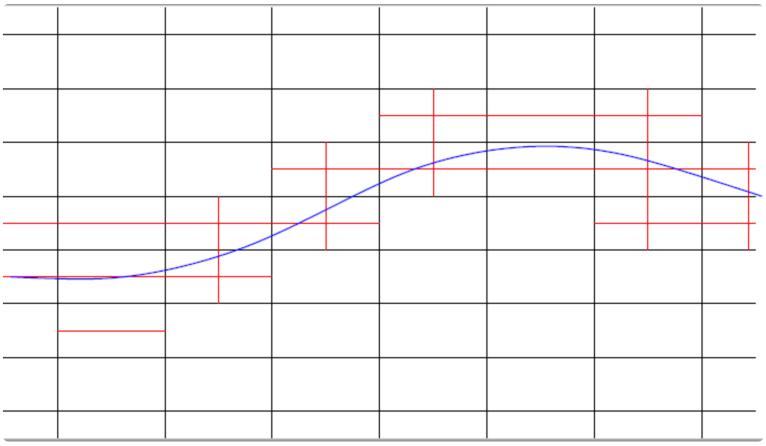


Figure 2.3.2: Free surface refinement criterion with isotropic and directional refinement, figure 5.1 in the Theory Manual [6].

2.3.8 Discretization Schemes

It is possible to change the numerical scheme applied in the FINE/Marine calculation regarding the turbulence-, momentum-, multi-fluid-, cavitation- and passive scalar equations. The multi-fluid equations determine how the mass fraction discretization is handled, and only available for multi-fluid computations. Eight different discretization schemes are applicable in FINE/Marine: BRICS, BICS, GDS, IGDS, MGDS, AVLSMART, HRIC and CICSAM.

The GDS scheme use upwind or centered discretization at the momentum equation. Upwind discretization schemes could result in numerical diffusion, which results in smearing of the interface. Central differencing can cause non-physical oscillations around the interface. Pros of the GDS scheme is that it could be used with any time step, as it is not limited by a CFL number. The Inter-Gamma scheme, IGDS, is based on the GDS scheme with an additional downwind difference scheme. However, by applying this, a CFL number limitation of 0.3 is introduced. To increase this limit a correction were made, and hence the MGDS scheme background. Blended Interface Capturing Scheme, BICS, combines advantages of the IGDS and GDS scheme for CFL number below 10. It blends the two schemes in the way that for high Courant number it behaves like GDS and for low numbers as IGDS. An additional reconstruction results in the BRICS scheme, which is recommended by the User Guide [5] for interface capturing the free surface elevation. More detailed information about the schemes could be found in the FINE/Marine theory guide [6]. The AVLSMART scheme is based on a third order QUICK scheme, and it is implemented using the chi scheme methodology. It showed improved convergence for many situations, and it is illustrated in figure 2.3.3.

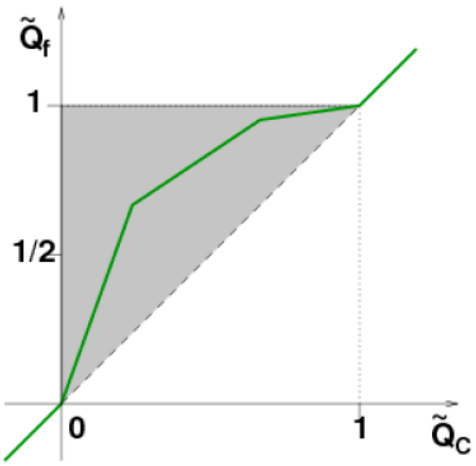


Figure 2.3.3: AVLSMART discretization scheme in an normalized variable diagram, figure 1.6 in the Theory Manual [6].

2.3.9 Python Language

FINE/Marine use the object-orientated programming language Python. This section briefly describe the most essential information to understand a python script. The character # at the beginning of a line marks a comment, and it will be skipped during the execution of a script. Strings are enclosed by single or double quotes respectively, i.e. ’ and ". Python has both conditional and loop statements and in FINE/Marine they can be used to run through a set of calculations or generating macros at grid points. Additionally, function definition, file handling, using library functions and execute scripts inside another script are all some of the possibilities enabled using Python.

All actions executed in the FINE/Marine interface are recorded in a script. This gives flexibility for automation of project management. Through the terminal in both Linux and Windows, python scripts can be executed directly. This enables the possibility to run FINE/Marine through the terminal without opening the software.

3 Waves Applied in CFD

This chapter present a review of existing literature, and it is separated into three sections: (1) Studies performed on numerical wave tanks (NWTs). (2) Waves applied in CFD analysis in offshore applications. (3) Use of NWTs in the maritime sector. The first section focus on the setup of the NWTs and the two latter on the results itself. Offshore and maritime applications are separated to distinguish between respectively fixed and moving bodies in the industry.

3.1 Numerical Wave Tank

Currently, physical model experiments in wave tanks are the most common method for researching waves and wave-structure interactions. A wave tank is usually long and narrow with a wave maker at one end and a damping beach at the other. Expenses and time consumption associated with model tests are constructing a prototype, renting the wave tank, re-design and testing. Researchers focus on numerical wave tanks has increased, and CFD codes are used to numerically represent physical wave tanks [7].

Different numerical methods are used in research to simulate ocean waves. In this section literature and previous work on numerical wave tanks are presented. Already in 2002 the 23rd International Towing Tank Conference (ITTC) gave a brief review of issues related to the use of numerical models of wave basins. Parasitic effects in wave tanks, non-linear effects, instabilities of wave trains, spatial variations and reflection effects was discussed. Cotter (1992) present a particular interesting study, suggesting an alternative way of calculating the wave length by use of wave period, distance between two probes and elapsed time between two maximums surface elevations [9].

Mazaheri et al. [22] used numerical models called FLUENT and FLOW-3D, both based on the NS equations and VOF method. They vaildated the models by comparing the horizontal component of particle velocity and free-surface elevation. Additionally, they tested dissipation zones with four different slopes. Applied wave characteristics were $H = 0.1$ [m], $h = 0.6$ [m] and $T = 2$ [s].

Leung and Du [23] used FLUENT and the VOF method to model a two-dimensional numerical wave tank. The waves were simulated by a dynamic meshing method and damped by porous media, which efficiently absorbed the wave energy. Tian et al. [7] also used FLUENT and the VOF method to model a three-dimensional wave tank and further look at wave propagation and hydrodynamic forces. They validated the waves through analytical theory for both a structure-free wave basin and other benchmark scenarios.

The moving boundary at the free surface remain the major challenge in numerical wave making. Proper boundary conditions are of importance due to its influence of the result. Dong and Huang

[24] developed a numerical scheme for solving two-dimensional wavemaking, inducing small- and finite-amplitude waves verified by analytic solutions. The study used $H = 1.0$ [cm], $h = 0.4$ [m] and $T = 1.25$ [s] which gives an Ursell number of 0.67.

A three-dimensional numerical wave tank, presented by Kim et al. [25], simulated characteristics of nonlinear multidirectional waves using a finite-difference scheme and marker-and-cell method.

Numerical wave tanks can be used to realize an open sea condition numerically. During these types of long time simulations, efficient damping schemes are highly important to ensure accuracy and reliable results. Physical experiments can often have challenges with reliability due to reflection of long time simulations. Numerical instabilities may occur when reflected waves are not eliminated properly, and Koo et al. [26] looked at various types of artificial damping schemes by use of different ramping functions.

Bihs et al. [27] developed a three-dimensional wave tank by use of a new level set method with improved density interpolation, allowing modeling of complex waves. The relaxation method was applied for simulation of a numerical beach and waves generated at the inlet by a Dirichlet type of condition. Benchmark cases was tested using REEF3D, as mentioned more under the offshore application section below. Using the VOF algorithm for interface-capturing has showed to be successful. Additionally, they applied a method of generating waves using a relaxation method where the wave generation was in a relaxation zone of one wavelength.

Miquel et al. [28] used REEF3D to analyze different methods of wave generation and absorption. The RANS equations and level set method for surface capturing was applied. Different wave types was computed to validate the model, and additional validation was performed using OpenFOAM. The paper divide between active and passive wave absorption, and both methods investigated. Dirichlet boundary condition and the relaxation method were applied for wave generation.

Ducrozet et al. [29] present a study of the latest HOST model, focusing on detailed third order simulation of wave generation. Stability and accuracy was discussed and validation by use of experimental data for both 2D and 3D cases.

Wave absorbing method affect the quality of the generated waves, and Zhe et al. [30] used the VB language for wave simulation and validation. The study was done using realistic full scales of the numerical wave tank, hereby avoiding scaling problems. Porous media for damping was used, and validation done using a two-dimensional wave tank simulating waves with different wave properties.

Zhi-Fu et al. [31] used the time domain boundary element method for simulating irregular waves propagation. Wave energy spectrum theory was used to obtain theoretical solutions that were used to validate the irregular waves. Introductory simulations with regular waves were performed to verify the numerical schemes. A total simulation time of 200 [s] with time increment of $T/40$ were used for the irregular simulations.

Saghi et al. [32] performed a study of linear and nonlinear wave generation in a two-dimensional numerical wave tank, using a clustering technique VOF method. The paper had an increased focus on mesh generation techniques. Some benchmark cases were performed to validate the model in addition to be validated against theory.

Another possibility is to use parallel implementation and validation based on fully nonlinear potential flow theory, as Nimmala et al. [33]. Their study focused on three-dimensional analysis of large scale, to duplicate algorithmic and exact physical features. By use of experimental data and analytic solutions, the results were validated.

Baquet and Kim [34] modeled a fully-nonlinear, steep, irregular wave field of three-hour duration without structures in it. The paper investigated possibilities of coupling potential theory and CFD for computational efficiency, and energy loss were compensated by applying factors of wave spectral frequency components on the input wave spectrum. Extensive validations against analytical waves and wave calibration tests were performed.

3.2 Offshore Application

Waves are one of the main parameters of interest in the design stage of structures in harsh marine environments. Exploring wave making methods is therefore necessary to ensure accurate estimations of physical ocean environments [7].

STAR-CCM+, a commercial CFD platform for simulations operating under postulated real-world conditions, is frequently applied in previous studies [35]. Pakozdi *et al.* [10] simulated long crested breaking waves and their impact on a rectangular cylinder- and deck structure. The study used potential flow theory to initialize the NS-VOF simulation, and showed that long distance nonlinear wave propagation could be simulated accurately and efficiently. A moving mesh technology at the boundary condition was used to simulate a wavemaker. A higher order integration scheme was applied to reduce numerical diffusion and make a larger time step possible. Data from physical experiments were used to validate the model at selected wave probes. The inlet and outlet were placed far away to avoid any reflecting waves. In addition, the VOF method was used to capture the free surface.

Magnitude and distribution of hydrodynamic loads on a fixed multicolumn offshore platform were investigated for extreme wave events by Nagi *et al.* [36] in 2017. Experimental measurement were used to validate the waves created by the STAR-CMM+ software, assuming laminar flow and the VOF method applied. Downstream wave reflections was limited by a damping zone.

REEF3D is a frequently used open source CFD model focusing on marine and coastal flows, solves the NS Equations in three dimensions. The level set method is used to capture the interface at the free surface. Several studies use this model to investigate irregular wave forces on offshore structures. For example, Aggarwal *et al.* [37] studied irregular wave forces on a large vertical cylinder. Irregular waves were defined through the JONSWAP wave spectrum and validated using a numerical tank in addition to analytic equations for loads on a cylinder in regular waves, as validated by wave gauges. Fast Fourier Transform (FFT) is applied to simplify the random sea surface into a sum of linear waves. The $k-\omega$ model is used to model turbulence and the level set method for the free surface. Generation and absorption of waves is done by the relaxation method.

A similar study was performed by Aggarwal *et al.* [11], invesigating statistical parameters for free surface elevation around a monopile and corresponding wave forces. The study used the Bretchneider spectrum for irregular wave generation, and validated the resulting waves by comparing wave spectrums at different locations in the tank. An adaptive time stepping scheme was used in the numerical model, and turbulence was modeled by $k-\varepsilon$.

3.3 Maritime Application

Numerical methods are increasingly important in predicting ship resistance in waves. Wu *et al.* [38] researched CFD simulations of ship motions and added resistance in waves for a high speed trimaran, using governing RANS equations and VOF method. A flexible flap wavemaker and artificial damping zones were modeled in the NWT. The numerical model was benchmarked against an experiment in a towing tank. Ye *et al.* [39] used a propriatory in-house solver to study added resistance and vertical ship motions in regular head waves. Non-linear factors disturb the results at individual wavelengths. Bal et al. [40] investigated flow characteristics around three-dimensional bodies and waves reflected by the sidewalls in a numerical towing tank. Effects of reflected waves from the sidewalls are also discussed. Miyata et al. [41] chose a different approach to study ship motion and resistance. The study especially highlighted the two codes TUMMAC and WISDAM as major contributors in further development of ship design technology.

Over the past two decades, use of numerical methods in ship hydrodynamics have been popular. The effectiveness of the tool increasingly comply with technological development. Because of the importance of viscosity in ship hydrodynamics, simulations are frequently researched. Performance analysis and forecast in addition to hull optimization are areas where CFD is effective [42].

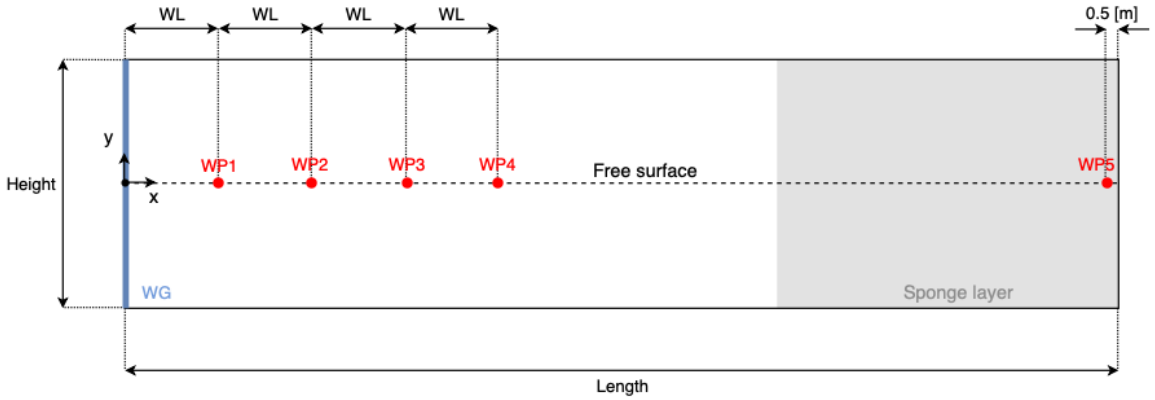
All mentioned models in this literature study have in common that some level of detail must be given up for reduced computational cost, which is reasonable for most engineering problems. Slender structures are one example of a case where giving up detailed information is not accepted [27].

It is beyond the scope for this thesis to provide a detailed analysis of existing literature, thus only a limited number of references are presented above. See [43] for a detailed review of literature on numerical wave tanks.

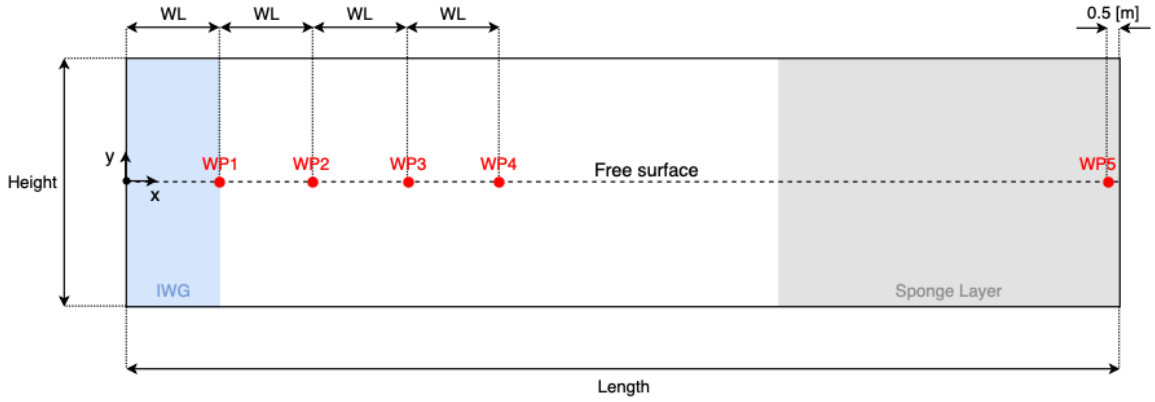
4 Model Formulation and Analysis Approach

4.1 Model Formulation

Analysis of numerical wave tanks (NWT) with regular and irregular waves was performed using the FINE/Marine software with two different approaches of wave-generation: the wave generator (WG) and the internal wave generator (IWG). This chapter include the model formulation usin in the different scenarios illustrated in figure 4.1.1. The analysis approach of the CFD computation is briefly presented.



(a) Method 1: Wave Generator.



(b) Method 2: Internal Wave Generator.

Figure 4.1.1: Schematics both domains including placement of wave probes (WP) and sponge layer.

4.1.1 CAD Model

A three dimensional CAD model of the NWT was created by the build-in tool in HEXPRESS in section 2.3. First, a rectangular box with corners in $(0, -10, 0)$ and $(62.5, 10, 0.001)$ define the domain. Further, boundary condition types were defined as mirror and external. Mirror was applied at the front and back surface. By using this boundary condition, the flow on both sides of the mirror are projected to the other side of the mirror. External boundary condition was used on the remaining surfaces: upstream-, downstream-, upper- and lower boundary. In the FM graphical user interface (GUI) further definitions of the external boundary conditions were made, as described in the section 4.1.3. 2D grid generation was turned on and an additional internal surface was defined at the free surface.

4.1.2 Mesh Generation

Defining the grid generation as two-dimensional implies that the thickness of the cells in z-direction is set to the thickness of the domain. The fluid motion equations are following solved in only two dimensions. Note that the mesh generation approach described in this section applies to all meshes in the mesh refinement study for both wave generation methods. The initial mesh consisted of uniform cells of size 0.5 [m], resulting in 5 000 cells. Global number of refinements were limited to 10 and surface refinement was activated at the defined internal surface at $y = 0$ [m]. Settings used for the surface refinement were: maximum 10 refinements, maximum aspect ratio of 12.5, refinement diffusion equal to Global and target cell size of respectively 0.0625 and 0.0039 on the x- and y-axis. Inspiration to these settings was taken from recommendations given in *Demo Case 4: Free-Surface Refinements* in the FINE/Marine documentation. Neither curve refinement, box refinement and trimming were activated during this analysis. Further, the features *snap to geometry*, *optimize* and *viscous layers* were applied with default settings. This resulted in a grid with 141 250 cells, pictured in figure 4.1.2.

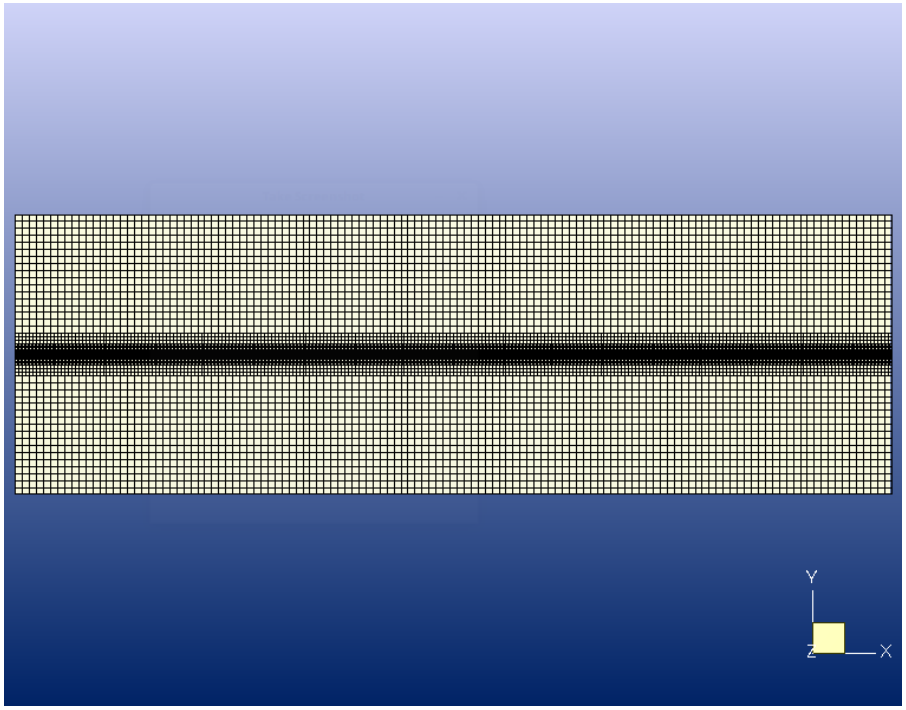


Figure 4.1.2: Snapshot of mesh from FINE/Marine.

Adaptive Grid Refinement

In Chapter 23 of the FINE/Marine User Guide [5] best practices of Adaptive Grid Refinement (AGR) suggest the following for usage of the free surface criterion. The initial mesh should be reasonably fine and very regular, i.e. all the cells should be as close to rectangular as possible. Sharp surface capturing is one of the elements it is made for, and it is thus suitable for this thesis. All the example cases are with structures, therefore the scenario closest to a Numerical Wave Tank is a ship in waves. Recommendations to original cell size with reference to the overall length of the ship is made here. The mesh generation in this thesis is influenced by the grid in demo case 4 described below.

Demo Case 4: Free-Surface Refinements

Additionally, the FINE Marine software come together with several public demo cases. They include all necessary elements to run a computation in few clicks. Demo case 4 is used to show multi-fluid computation, free-surface refinements and Adaptive Grid Refinement (AGR). Figure 4.1.3 illustrates the setup of a towed shop in still water, fixed position, with free surface. The mesh strategy applied in this case is to capture the wave pattern with a low number of cells and a reasonable accuracy. Adaptive Grid Refinement with Free surface tensor criterion were applied. Mesh data of the demo case is summarized in table 4.1.1. In this thesis the mesh method applied in the demo case is of high interest, and is used as a reference point for meshing the free surface.

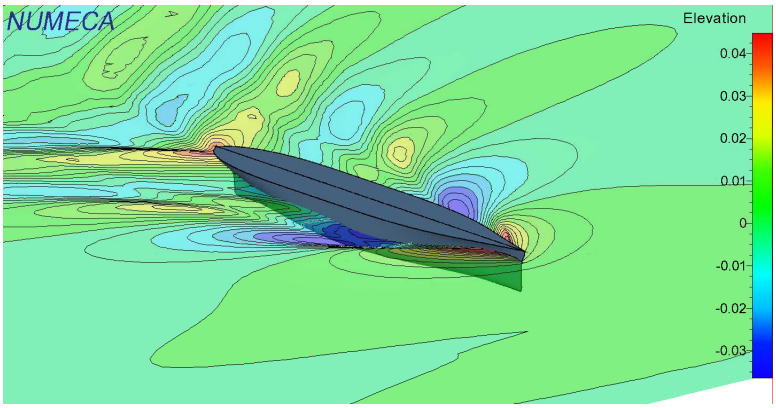


Figure 4.1.3: Free-Surface Refinements of Numeca's Demo Case 4.

Table 4.1.1: Mesh Properties of Demo Case 4.

Surface Refinement	Target Cell Size	x-axis	0.1 [m]
		y-axis	0.1 [m]
		z-axis	0.0017 [m]
	Maximum Nb. of Refinements		7
	Maximum Aspect Ratio		100
	Refinement Diffusion		Global
Adaptive Grid Refinement	Criterion	Refinement Criterion Type	Free Surface (Tensor)
		Target Grid Spacing Normal to Free Surface	0.004 [m]
		Minimum Size Limit for Refined Cells	0 [m]
	Grid Quality	Nb. of Layers Copying Full Criterion Value	2
		Nb. of Layers Copying Fraction Value	0
		Fraction	0.6
	Box	Restricted Refinements in X, Y and Z-directions	Applied
	Control	Nb. of Steps Before First Call to Refinement Procedure	125
		Nb. of Steps Between Calls to Refinement Procedure	25
		Maximum Nb. of Cells per Partition	500.000

In this thesis, AGR was enabled through the FINE/Marine GUI with the free surface tensor criterion. Target grid spacing normal to free surface was set as 0.004 [m] and 0 as minimum size limit for refined cells. Diffusion criterion were 2 layers copying the full criterion value and 1 layer copying fraction value of 0.6. Longitudinal direction only was allowed in the boundary layers. Under computation control, 250 steps before first call to the refinement procedure with 50 as number of steps between each call to the refinement procedure. Limitations regarding available memory must be considered when selecting the settings for refinement control. Recommendations in the User Guide gives the estimate of $100.000 - 150.000$ cells for each GB of memory per partition as a maximum. The default of 500.000 cells should be appliable on a standard computer and is therefore used in this thesis.

4.1.3 Boundary Conditions

Back in the FINE/Marine GUI, configurations for the boundary conditions were applied to simulate the physical boundaries of a deep water tank. The bottom boundary should simulate a sea floor with zero velocity. A boundary layer will form close to the sea floor due to shear and following no-slip at the boundary. As long as the deep water criteria $h/L > 0.5$ is satisfied, the NWT should in theory simulate a deep water scenario. This implies that the bottom boundary should not affect the wave pattern at the free surface. The downstream boundary should not have any waves passing through, i.e zero velocity. As done in *Demo Case 4*, the top and bottom boundaries were defined as prescribed pressure. At the inlet boundary, of the scenario using the Wave Generator, the wave-maker was selected with wave properties described in section 4.1.5. In the scenario using the Internal Wave

Generator the inlet boundary are equal to the outlet boundary, i.e. zero velocity. A summary of the schematics of the external boundary conditions are presented in figure 4.1.4. All far field boundaries are defined by zero velocity in both x- and y-direction. The prescribed pressure boundaries are defined as updated hydrostatic pressure.

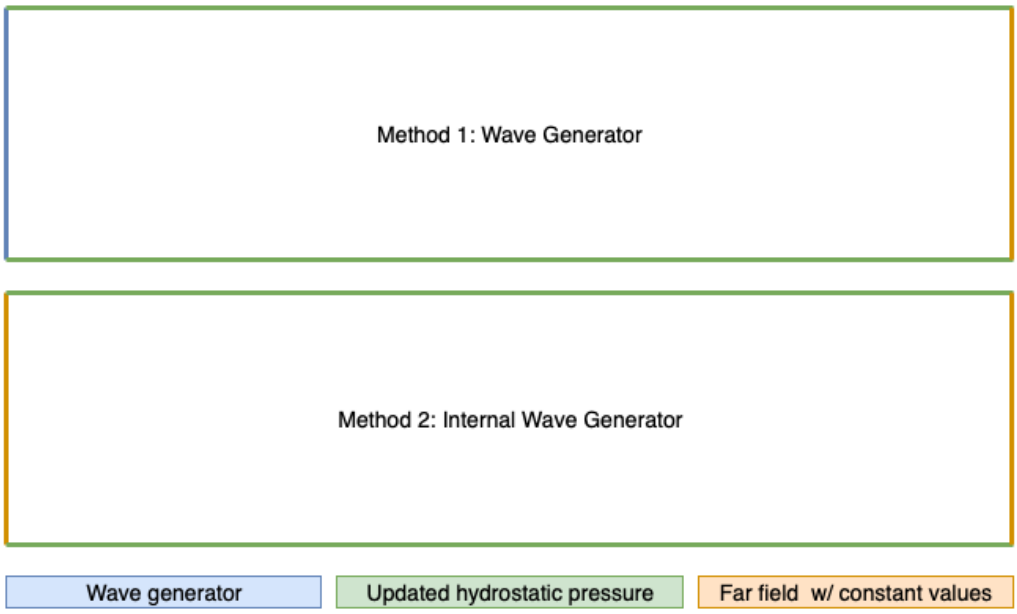


Figure 4.1.4: Schematics of external boundary conditions.

4.1.4 Fluid Model, Additional Models and Solver Settings

To model the interface between air and water, the multi-fluid model was enabled in the FINE/Marine GUI. The VOF method then satisfies the kinematic- and free-surface conditions at the surface. Fluid properties used are listed in table 4.1.2, and they are decided from the average temperatures in Stavanger, Norway [44, 45]. Gravitational acceleration was defined in negative y-direction with a value of 9.81 $[m/s^2]$.

Table 4.1.2: Fluid properties of the multi-fluid scenario.

Fluid	**Temperature** $[°C]$	**Dynamic viscosity** $[Pa \cdot s]$	**Density** $[kg/m^3]$
Salt water	10	1.397e-003	1027.00
Air	10	1.77e-005	1.25

Wave damping was applied as an additional model, mainly to avoid reflection and other non-linear effects. It is called sponge layers or damping zones, and is the method of modeling a numerical beach. A limitation of the model is that it is only possible to generate sponge layers at the end of a domain. Length of the sponge layer is defined as 3.5 wavelengths, i.e. 21.858 [m], for the wave described in section 4.1.5. Even though this thesis mainly deals with a two-dimensional flow, and turbulence is a three-dimensional phenomena, the SST $k-\omega$ turbulence model was applied. The internal wave generator (IWG) falls under the category of additional models, and it was applied when the Wave Generator at the boundary not were used.

Solver settings used were maximum 20 non-linear iterations and convergence criteria of 2 for both regular and irregular analysis, as recommended in the user manual [5] for irregular analysis. With regular waves, the solution was saved every 4000th time step of size 0.01 [s] and total number of time

steps equal to 8000. The default discretization scheme for two-phase flows, BRICS, was used. It stands for Blended Reconstruction Interface Capturing, and it is recommended for accurate modeling of the free surface. For irregular waves, the number of time steps and the frequency of saving the solution were larger, as described further in chapter 8.

4.1.5 Wave Generation

Regular Waves

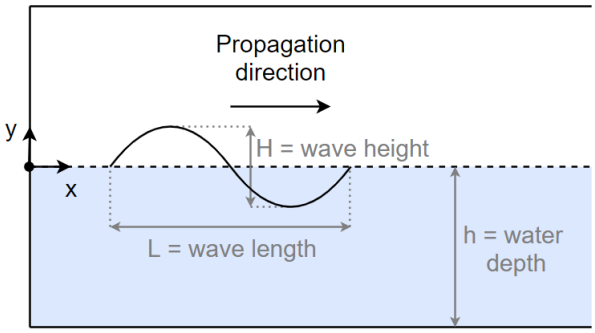


Figure 4.1.5: Variable definition as used for the wave generation.

Table 4.1.3: Wave statistics.

Parameter	Value
H $[m]$	0.1
h $[m]$	10.0
T $[s]$	2.0
L $[m]$	6.25
H/L [-]	0.016
k [-]	1.006
c $[m/s]$	0.016

Regular 1st order sinusoidal Stoke waves are generated by both the wave generator and the internal wave generator (with properties given in table 4.1.3). Figure 4.1.5 illustrates the variable definition used for the wave generation. Note that in section 2.1.2 both L and λ are used for the wave length. The propagation direction in the figure is in positive x-direction.

The user manual [5] recommends using the build-in wave generation info tool as indicator for the setup of the wave generation project. Figure 4.1.6 is a snapshot of the tool before pushing the *Compute* button, and figure 4.1.7 shows the recommendations.

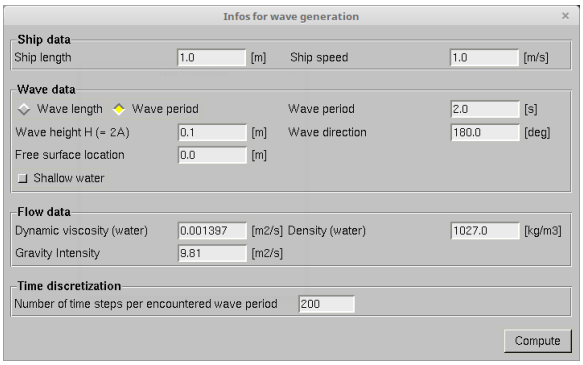


Figure 4.1.6: Wave generation info tool with wave properties from table 4.1.3.

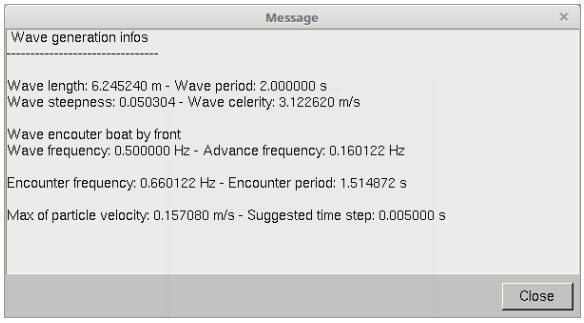


Figure 4.1.7: Suggested computation setup calculated from the wave generation info tool in figure 4.1.6

.

Irregular Waves

As for regular waves, setting up irregular waves are the same for both methods of wave generation. Spectrum type, depth, peak period, significant height, direction of propagation, reference point (WG) and source location (IWG) are parameters defined by the user.

Settings used in the analysis is listed in table 4.1.4. Note that this computation used over 10 days to reach 200 000 out of original 1 000 000 time steps, which means that running the computation in full 3 [hour] simulation would take over 50 days. During the thesis the goal simulation length was decreased to 500 [s]. As described in the theory chapter, the parameter called significant height represents the mean wave height similar to the third highest of the waves, measured from crest to trough. A more modern way of describing the parameters is four times the standard deviation of the surface elevation. Peak period correspond to the frequency with highest amount of wave energy in the spectrum.

Table 4.1.4: Irregular wave generation.

Parameter	**Value**
Spectrum type [-]	ITTC
Depth [m]	10
Peak period [s]	2
Significant height [m]	0.1
Direction of propagation [-]	positive x-direction
Source location [(x,y)]	(3.12262,0)

4.1.6 Python Scripting

Throughout this thesis python scripting has been used to make the project management more efficient, especially the pre-procecssing described in this chapter. The first time the model formulation was done through the interface. After finishing the solver settings, the python script with the FINE/Marine commands was saved. In Appendix C, the generalized code is attached. It contain all the necessary data to recreate the 2D simulation with regular waves. While creating the code, debugging was done. Some error messages appeared in the interface, some of them indicating the mistake in the code, which made it more managable to correct them.

During the sensitivity analysis several external parameters and settings were changed, and to make this as efficient as possible the pre-processing was done by scripting. In the FINE/Marine interface the script was executed, and the result visualized.

A short psaudo explanation of the code comes in the following. The code begins with a definition of variables that changed during the sensitivity analysis. Further the project was defined and HEXPRESS opened. All supported macros regarding HEXPRESS start with the extension *HXP*, following the command. Domain dimensions was defined and initial boundary conditions selected. 2D mesh generation mode was turned on and an internal surface created. Further, grid properties was defined and the mode switched back to FINE/Marine. Following, external boundary conditions set, wave generation with right properties enabled and wave damping turned on. Then computation control variables were defined, and output control parameters given. In FINE/Marine the extension before the command is *FM*.

4.2 Analysis Approach

A flow chart of the analysis approach of the CFD investigation is presented in figure 4.2.1. Definition of the problem was step one: a numerical wave tank with creation of regular sinusoidal waves and irregular waves. The three next steps is the CFD analysis thoroughly described in detailed outline to the right in the figure. Pre-processing and solver settings was described in the previous chapter, and the post-processing are briefly explained further in the following paragraphs. The fifth step implicit involves a dependency study, described in detail in chapter 6. Once the user was satisfied with the accuracy of the result, the analysis could be stopped.

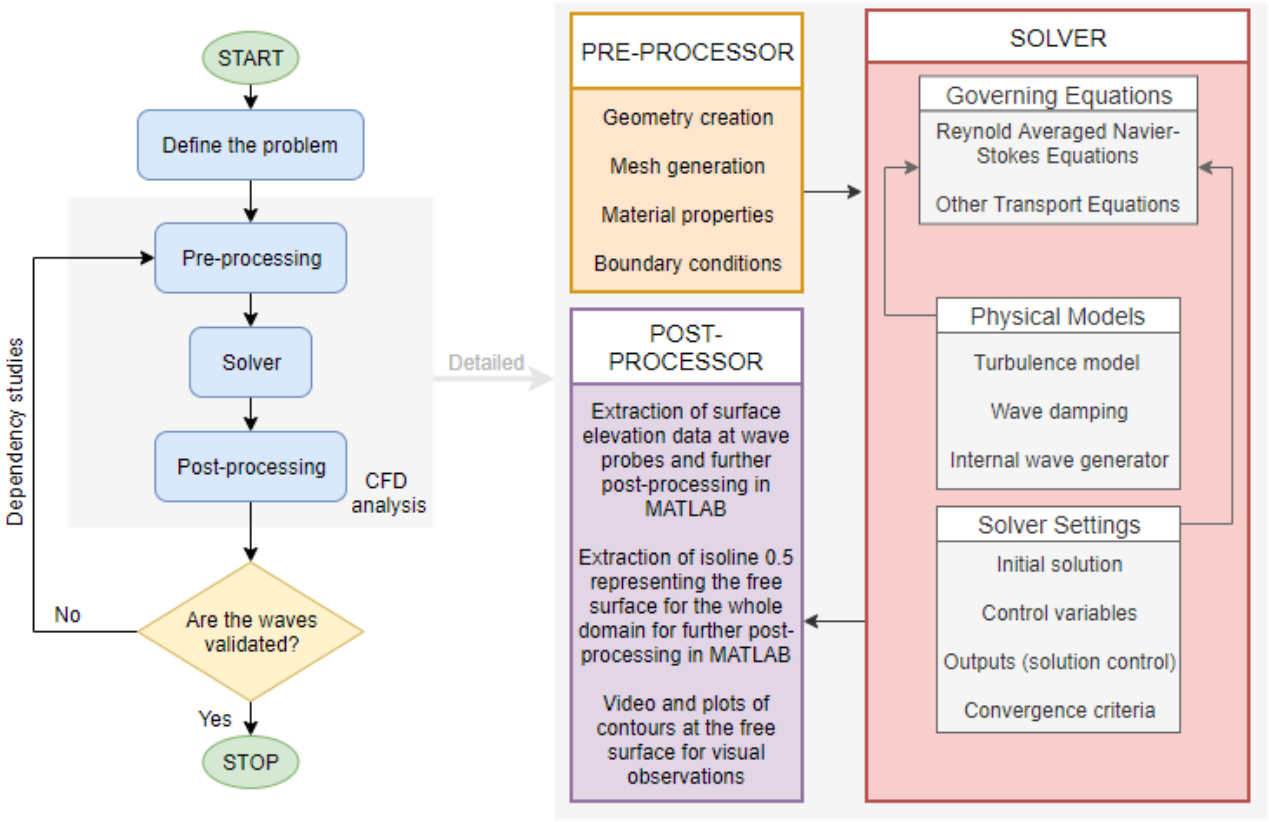


Figure 4.2.1: Flow chart describing the analysis approach including a detailed outline of the CFD analysis.

Post-processing were done by extracting wave probe data from the generated file *points_probe.dat*. Surface probe data at a selected time steps, defined throughout the thesis, was extracted in CFView. The data of mass fraction 0.5, representing the free surface, was extracted as a cartesian data set. This could be done either by use of Python scripting or manually extracting the data. During the sensitivity analysis, waves were compared to theory, i.e. sinusoidal waves with the same characteristics. For projects with irregular waves, wave spectrums was used to validate the sea state. MATLAB was used for processing all the extracted data, and the code is attached in Appedix D. *WAFO*, a freely redistributable MATLAB toolbox, was used for the statistical wave analysis, e.g. Fast Fourier Transform to get the wave energy spectrum. Data from probes placed at the end of the damping zone were also analyzed to check for possible wave reflection effects.

All simulations were run on a computer offered at NTNU with 32 cores using between 10 to 32 parallel computations.

5 | Project Thesis Results

This master thesis is a continuation of the project thesis *An Introductory CFD Study of Numerical Wave Tanks* by the same author, delivered in December 2018. In this chapter a brief summary of the results from the project are presented, since they were used for further research in the current thesis.

The numerical wave tank during the mentioned study had length and height of 40 and 10 [m] respectively. Wave damping zone length varied between 6 and 10 [m] dependent of the wave generation method used. Introductory refinement studies of mesh and time step size were performed. Note that the current thesis have used the same wave properties for the sensitivity analysis. During the mesh refinement study different number of refinements and target cell size were tested. The final mesh contained 13 600 cells, and gave the most accurate result. Surface refinement settings applied were: 10 number of refinements globally and locally, and a target cell size of $dx = 0.5$ [m] and $dy = 0.00617$ [m]. During the time step refinement study, the following conclusion was drawn. Very small time steps did not give considerably better results than by use of 0.01 [s]. However, during these analysis a time step of 0.005 [s] could be applied due to simulation of relatively small size, and thereby not too high computational costs. In larger simulations, a compensation of a coarser time step size could be necessary.

The analysis showed that both wave height and length decreased downstream the numerical wave tank regardless of wave generation method. A phase difference of π at the inlet of the numerical wave tank were found by use of the Wave Generator (WG) and Internal Wave Generator (IWG). By use of the IWG the resulting mean water level was below the initial free surface. Waves generated with the WG kept its maximum wave height relatively constant while propagating down the tank. Additionally, reflection challenges caused the solution to explode by use of the WG.

6 Dependency Studies

Nonphysical damping is one of the essential challenges using numerical wave tanks. Strict criteria for mesh resolution, time step size and general accuracy are important to avoid this, and thus thorough dependency studies must be done to ensure an accurate solution [27]. In this thesis several dependency studies were performed. This chapter present both the method, results and a discussion from the dependency studies on type of boundary conditions, mesh refinement, damping zone length, overall dimensions and discretization scheme.

6.1 Overall Dimensions

6.1.1 Analysis Approach

Sensitivity of overall dimensions was investigated in this thesis. With too small domain boundaries, the model could represent other physical scenarios than intended in this thesis. Examples are shallow water effects due to a small height, and reflections or not fully developed waves due to a short tank length. In the following analysis all parameters, except the height and length, were kept constant. The following list contain the runs simulated in this part of the sensitivity study.

Having a wide domain in itself is not using large amounts of capacity, it is the grid refinement and number of elements that are of importance. Meaning, having large enough overall dimensions was prioritized in this thesis. Additionally, the grid size at the top and bottom boundaries was coarse while the grid close to the free surface needed to be very fine to capture the waves accurately.

1. Height 10 [m], length 37.5 [m].
2. Height 10 [m], length 50 [m].
3. Height 10 [m], length 62.5 [m].
4. Height 6 [m], length 62.5 [m].
5. Height 14 [m], length 62.5 [m].
6. Height 18 [m], length 62.5 [m].

During this dependency study, the settings described in this paragraph were used. Boundary conditions as mirror for both front and back face, and external condition with far field on all other surfaces were used. Surface refinement activated with a target cell size of 0.125 and 0.00195 [m]

respectively in x- and y-direction. A maximum aspect ratio of 100 and refinement diffusion of 4. Unsteady time configuration, multi-fluid model, initial solution of 0 [m/s] in both x- and y-direction. Sponge layers of 2 wavelengths and numerical discretization scheme of multi-fluid as BRICS. Control variables as: maximum 20 non-linear iterations, convergence criteria of 2 orders, save solution every 4 000th time steps. The simulation length was selected to be 8 000 time steps of size 0.01 [s]. Outputs activated were surface/volume-probe for saving the mass fraction every time step, and wave probes as described in chapter 4, section 4.1.

These computations and the following, until section 6.4, are done using the Internal Wave Generator, unless other is stated. The previously mentioned challenges with sinkage were present during these computations, but it is neglected to investigate the sensitivity of other parameters separately.

6.1.2 Results

Note that the plots presented in the results are from wave probe 2, 2 wavelengths downstream the numerical wave tank. Figure 6.1.1 shows data from wave probe 2, illustrating how the detected surface elevation was compared to theory and the different lengths. A slightly higher wave crest and through was observed for the longest length, thus giving the best result of the three lengths tested. L stands for the overall domain length, given in meters.

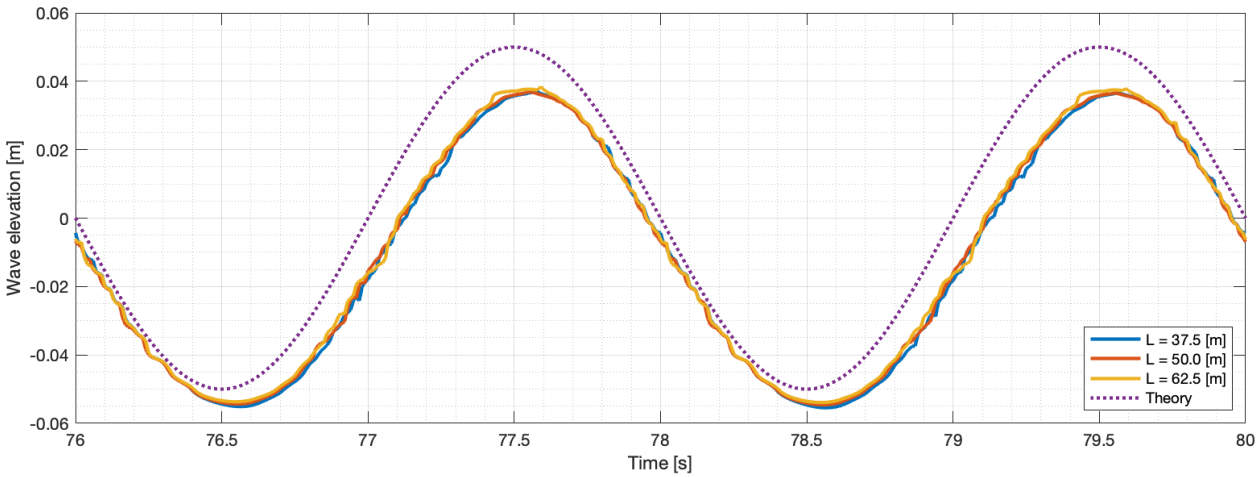


Figure 6.1.1: Plot of the free surface elevation over time at wave probe 2 for the three different lengths compared to theory.

In figure 6.1.2 the free surface elevation at wave probe 2 is presented for the four different heights tested, and compared to theory. Here h is the water depth given in meters, as defined in figure 4.1.5. For water depths from 5 [m] and above, there were not any noteworthy difference. Using $h = 3$ [m] the though is shallower, which may indicate the presence of shallow water effects. The deep water limit is $h > \lambda/2$ with $\lambda/2 = 3.12262$ [m], as defined in the theory section, meaning that $h = 3$ [m] falls under the category intermediate water depth.

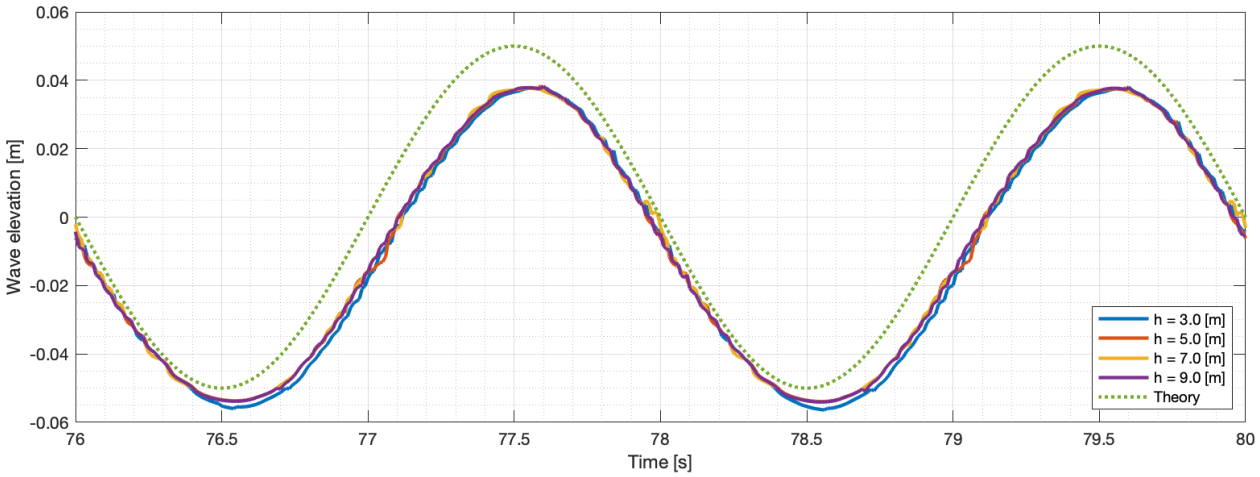


Figure 6.1.2: Plot of the free surface elevation over time at wave probe 2 for the four different heights compared to theory.

6.2 Damping Zone

6.2.1 Analysis Approach

During the project thesis some troubles with reflection from the back boundary occurred. These happened while using the Wave Generator (WG). To prevent this phenomena, a dependency study of damping zone length was performed using the Internal Wave Generator (IWG). Therefore, an additional validation of the improvement using longer damping zones was done. The same settings, described in the following paragraph, were applied for both methods. As recommended in the user manual [5], sponge layers should be 3 times the wavelength. Five different lengths were tested, as listed below.

1. 2.0 x wavelength = 12.49048 [m].
2. 2.5 x wavelength = 15.6131 [m].
3. 3.0 x wavelength = 18.73572 [m].
4. 3.5 x wavelength = 21.85834 [m].
5. 4.0 x wavelength = 24.98096 [m].

The settings described in this paragraph were used for this respective sensitivity study. Total domain length of 62.5 [m] and height of 20 [m]. Boundary conditions as mirror for both front and back face, and external condition with far field on all other surfaces were used. Surface refinement activated with a target cell size of 0.125 and 0.00195 [m] respectively in x- and y-direction. A maximum aspect ratio of 100 and refinement diffusion of 4. Unsteady time configuration, multi-fluid model, initial solution of 0 [m/s] in both x- and y-direction. Numerical discretization scheme of multi-fluid as BRICS. Control variables as: maximum 20 non-linear iterations, convergence criteria of 2 orders, save solution every 4 000th time steps. The simulation length was selected to be 8 000 time steps of size 0.01 [s]. Outputs activated were surface/volume-probe for saving the mass fraction every time step, and wave probes as described in chapter 4, section 4.1.

6.2.2 Results

Figure 6.2.1 presents the free surface elevations captured through the isoline representing mass fraction 0.5. At $x = 50$ [m], the generated waves are close to sufficiently damped using the longest damping zones, i.e. $DZ = 3.5$ and 4 times the wavelength. The free surface at wave probe 5, 0.5 [m] from the tank end, is shown in 6.2.2. By observing the trends a sinking mean free surface level is detected. This correspond to challenges experienced during the project thesis. This problem is addressed further later in this master thesis. Additionally, reflections are observed as fluctuations and especially a peak at $t = 28$ [s], using smaller damping zones, i.e. $DZ = 2.0$, 2.5 and 3.0. To verify that a sponge layer of 3.5 times the wavelength was sufficient, an additional computation was performed by use of the Wave Generator at the boundary. All other settings during this computations were equal to the ones described in the previous paragraph. Figure 6.2.3 illustrates the discovered wave elevation at wave probe 5, note that they are in the $e-4$ [m] range. This plot shows that the reflection issues experienced in the project thesis were solved.

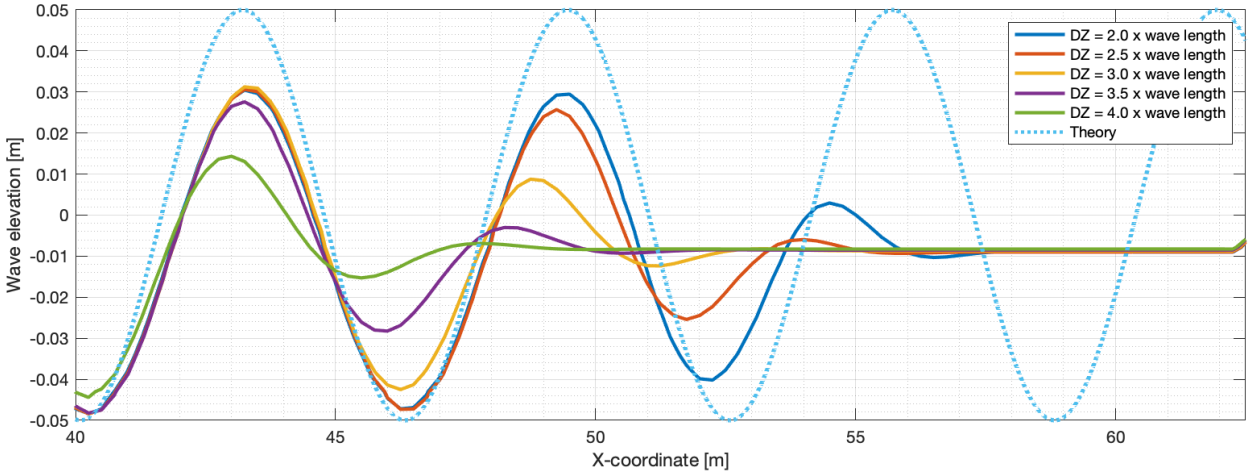


Figure 6.2.1: Free surface elevation downstream numerical wave tank for five different damping zone lengths compared to theory.

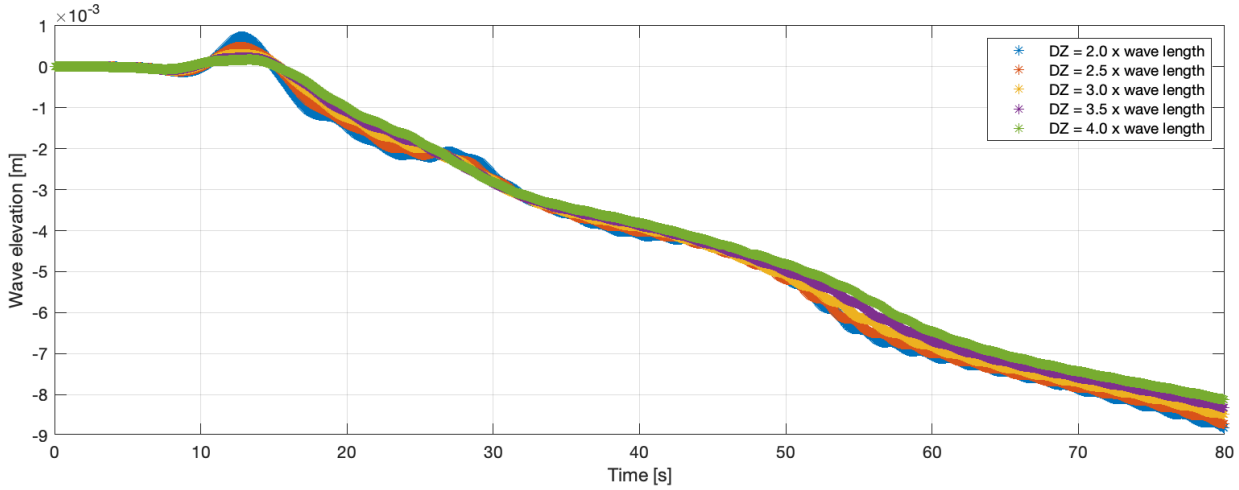


Figure 6.2.2: Plot of the free surface elevation over time at wave probe 5, 0.5 [m] from the tank end boundary, for five different damping zone lengths.

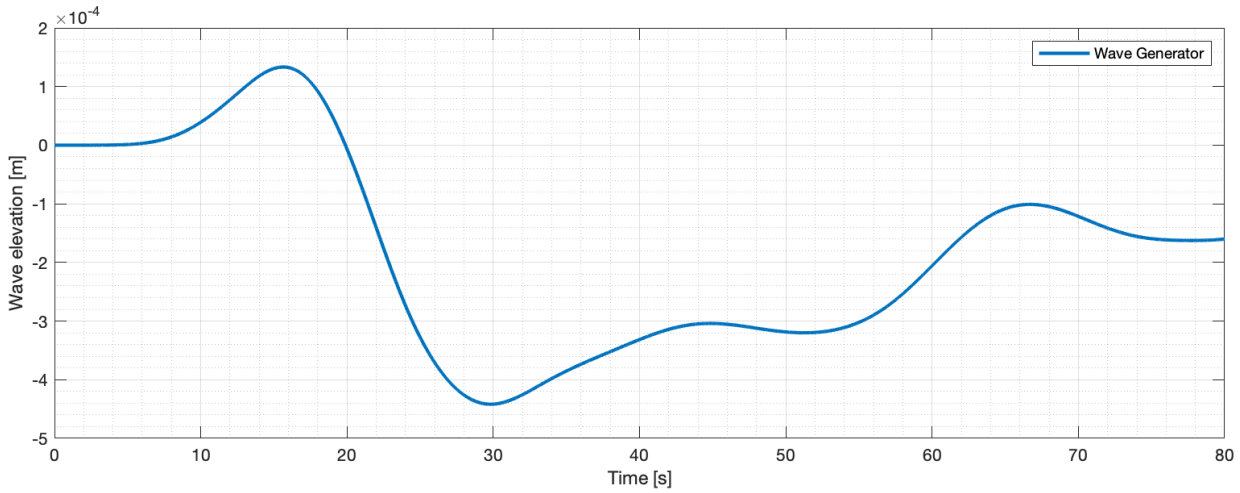


Figure 6.2.3: Plot of the free surface elevation over time at wave probe 5, 0.5 [m] from the tank end boundary, using the Wave Generator.

6.3 Mesh Refinement

6.3.1 Analysis Approach

A further mesh refinement study, than the introductory sensitivity analysis performed in the project thesis, was necessary. As presented in chapter 5, the target cell size during the project thesis was quite coarse. Before doing any further mesh refinement study, the target cell size of the surface refinement was changed to $dx = 0.0625$ [m] and $dy = 0.0039$ [m]. This resulted in a grid with 141 250 cells. First, the effect of a decreased aspect ratio (AR) was investigated. Starting at $AR = 200$ and dividing by 4 at each decreased aspect ratio, ending at $AR = 12.5$, three different aspect ratios were tested. The result is presented in the following section.

Settings presented in this paragraph were used for this respective mesh refinement study. Total domain length of 62.5 [m] and height of 20 [m]. Boundary conditions as mirror for both front and back face, and external condition with far field on all other surfaces were used. Unsteady time configuration, multi-fluid model, initial solution of 0 [m/s] in both x- and y-direction. Sponge layers of 3.5 wavelengths and numerical discretization scheme of multi-fluid as BRICS. Control variables as: maximum 20 non-linear iterations, convergence criteria of 2 orders, save solution every 4 000th time steps. The simulation length was selected to be 8 000 time steps of size 0.01 [s]. Outputs activated were surface/volume-probe for saving the mass fraction every time step, and wave probes as described in chapter 4, section 4.1.

After the initial refinement, the method of Adaptive Grid Refinement (AGR) was tested. In the user manual [5], this is recommended for irregular wave generation, so it was tested out using regular waves first. Use of AGR could result in creation of big files since several mesh files are saved during the computation. Time consumption should also be considered, as this introduce another element of the computation which take time. In section 4.1.2 the details about the AGR-setting are presented. Other settings used were the same as described above. In the following section results from applying AGR is compared to a computation without it.

6.3.2 Results

Changing the aspect ratio caused a difference of $0.1e-6$ [m] in wave height, which is so small that it is neglected. Figure 6.3.1 presents a plot of the free surface elevation over time at wave probe 2. It compares the result with and without use of AGR to theory. Using AGR the maximum crest- and through heights were closer to theory than without AGR. The wave profile of the wave elevation without AGR appears smoother than by use of AGR. As mention in the theory section, peaks are of most interest, therefore using the AGR gives the better result in this case. However, the accuracy is sufficiently accurate without AGR when simulating regular sea states.

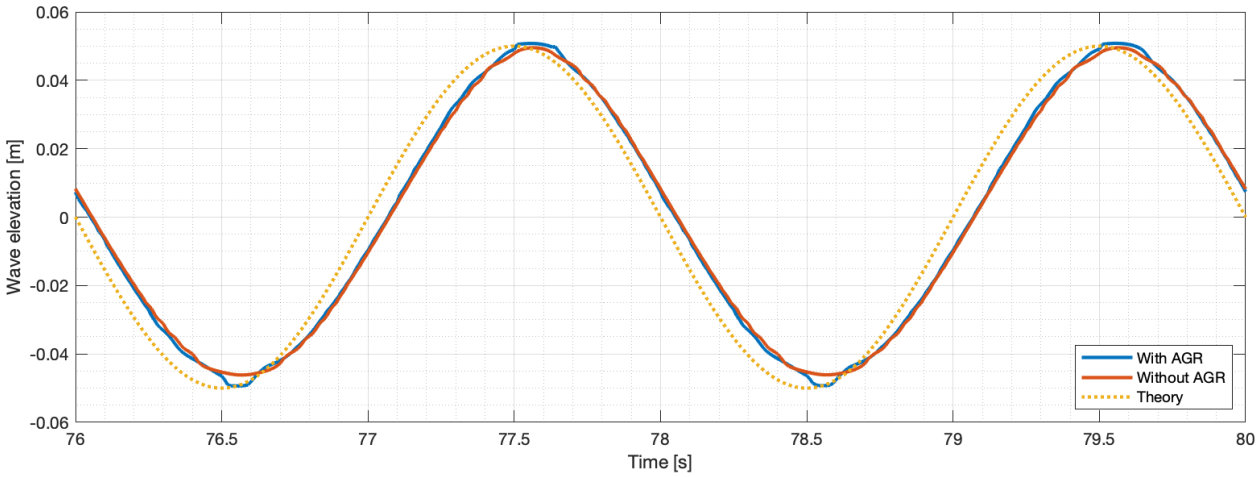


Figure 6.3.1: Plot of the free surface elevation over time at wave probe 2 for computation with and without AGR compared to theory.

Figure 6.3.2 is a plot of the surface elevation over time at wave probe 3, placed 18.7 [m] downstream the tank, for the new computation and a wave probe placed 20 [m] downstream the tank from the project thesis. The phase difference was taken into account and the two waves presented for the same phase. As seen in the plot, the result from the project thesis is more piece-wise and non-smooth.

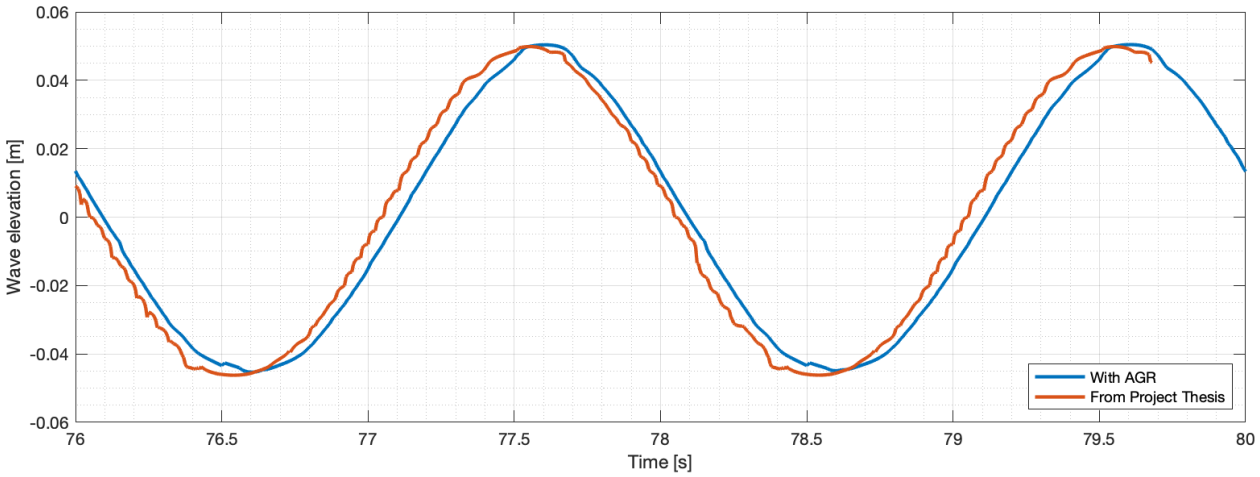


Figure 6.3.2: Plot of the free surface elevation over time at wave probe 3 (placed 18.73572 [m] downstream) for computation with AGR compared to final result in the project thesis at a wave probe placed 20 [m] downstream.

The final plot of the result from the performed mesh refinement study is shown in figure 6.3.3. Free surface refinement downstream tank is plotted against theory for computations with and without use of AGR. Crests are higher using AGR, and throughs are lower or equal to the computation without AGR. Otherwise they are similar and correspond well to theory.

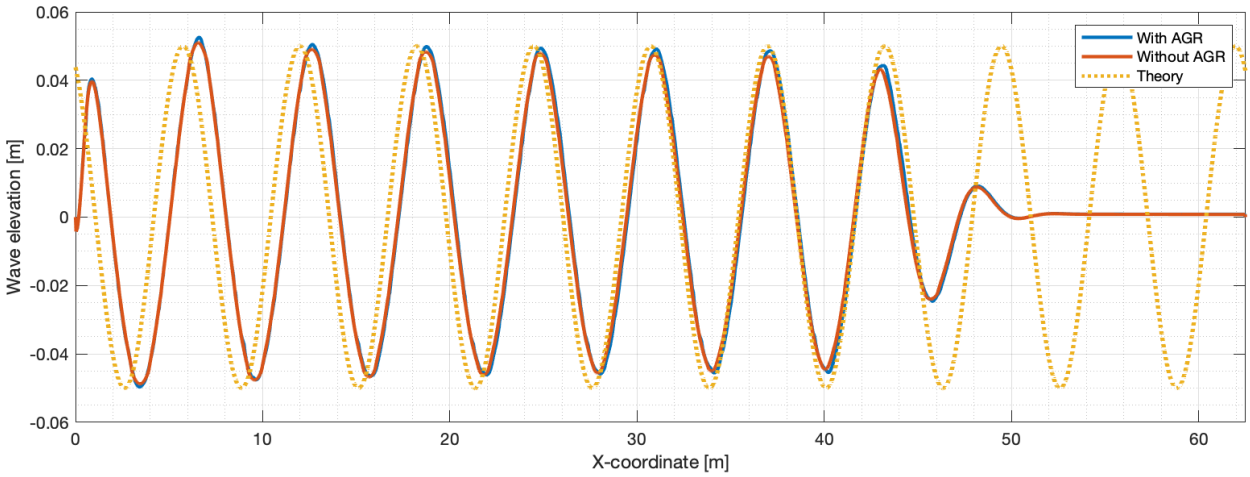


Figure 6.3.3: Free surface elevation downstream numerical wave tank for computation with and without AGR compared to theory.

6.4 Boundary Conditions

6.4.1 Analysis Approach

After going through the already mentioned sensitivity analysis, there were still problems with wave sinkage. This implied that some other parameter was the main cause of the challenges. A boundary condition dependency study with the three different ways of modeling the seafloor, listed below, was tested.

1. External condition, far field with constant values.
2. External condition, prescribed pressure, updated hydrostatic pressure.
3. Solid wall boundary condition, slip wall (zero shear stress).

Settings applied was a total height of 20 [m], length of domain equal to 62.5 [m]. Boundary conditions as mirror for both front and back face, and external condition with far field on all other surfaces, except the y-min surface that was investigated in this sensitivity analysis. The mesh generation described in section 4.1.2 was used in this analysis, resulting in 141 250 elements. All other variables than the lower boundary were kept constant: Unsteady time configuration, multi-fluid model, initial solution of 0 [m/s] in both x- and y-direction. Sponge layers of 3.5 wavelengths and numerical discretization scheme of multi-fluid as BRICS. Control variables as: maximum 20 non-linear iterations, convergence criteria of 2 orders, save solution every 4 000th time steps. The simulation length was selected to be 8 000 time steps of size 0.01 [s]. Outputs activated were surface/volume-probe for saving the mass fraction every time step, and wave probes as described in chapter 4, section 4.1.

6.4.2 Results

Figure 6.4.1 shows a plot of the wave elevation at wave probe 2 for three different lower boundary conditions. Far field result in a total wave height lower than the expected 0.1 [m], where slip wall and prescribed hydrostatic pressure gives approximately an overall height of 0.1 [m]. Using hydrostatic pressure, the wave crest and through are slightly higher than by use of slip wall. In figure 6.4.2 the isoline of mass fraction 0.5 is presented for each boundary compared against theory. Both hydrostatic pressure and slip wall corresponds well with theoretical values.

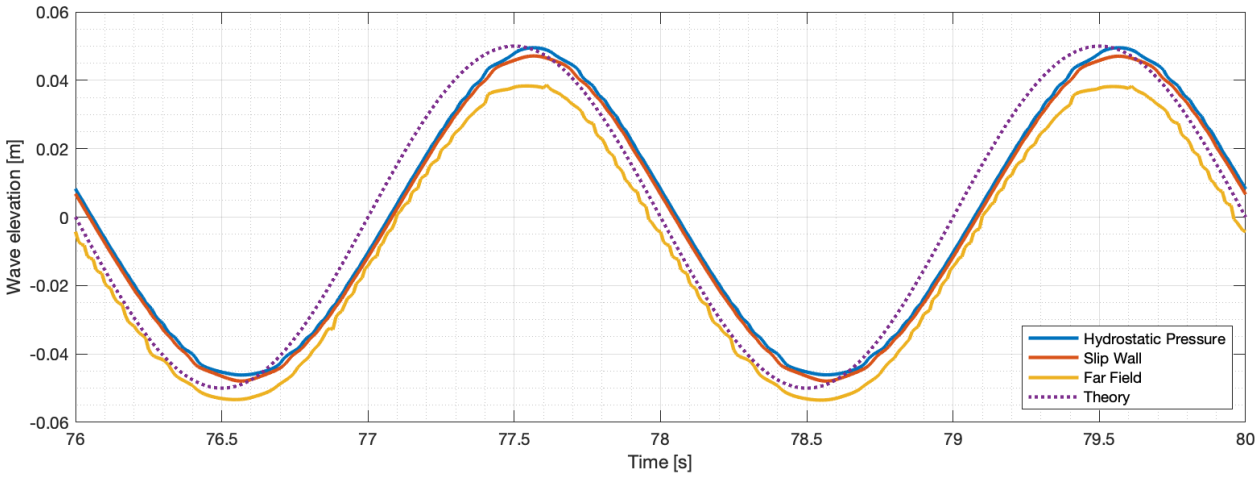


Figure 6.4.1: Plot of the free surface elevation over time at wave probe 2 for the three different boundary conditions compared to theory.

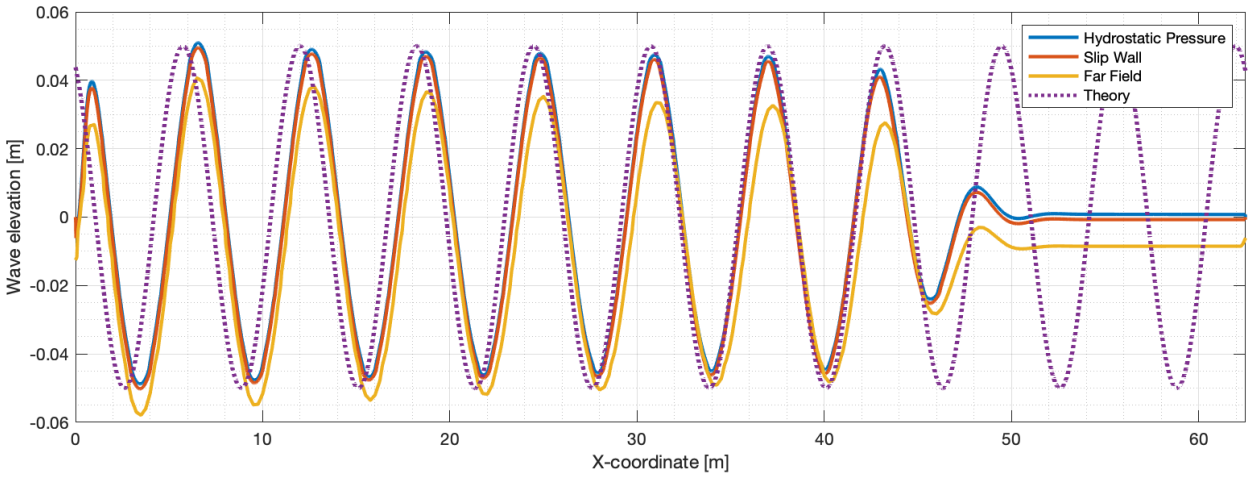


Figure 6.4.2: Free surface elevation downstream numerical wave tank for three different boundary conditions compared to theory.

In figure 6.4.3 the wave probe detected an increase of $6.69e-4$ [m] of the free surface height using the updated hydrostatic pressure condition. With slip wall a decreased free surface height of $6.23e-4$ [m] was detected. Both slopes in the $7.5e-6$ [m/s] order, with opposite signs. For the far field condition a decrease of $8.327e-3$ [m] with a negative slope of $1.04e-4$ [m/s]. A more detailed cut of the plot is presented in figure 6.4.4. The updated hydrostatic pressure boundary condition is close to zero at all times, which indicate that this method provide the most realistic simulation of the physical problem.

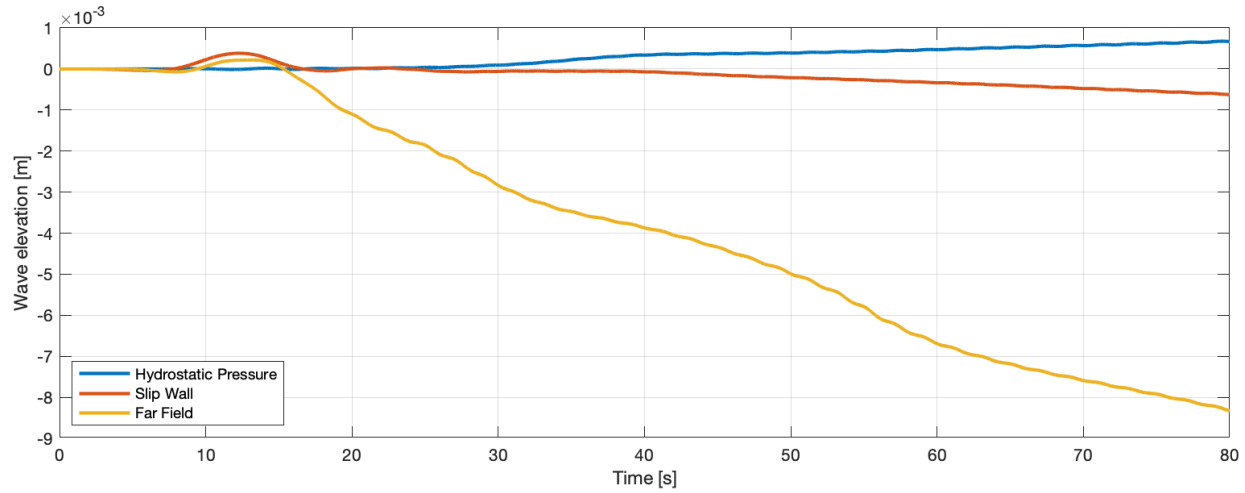


Figure 6.4.3: Plot of the free surface elevation over time at wave probe 5, at the end of the damping zone, 0.5 [m] from the end of the wave tank, for the three different boundary conditions.

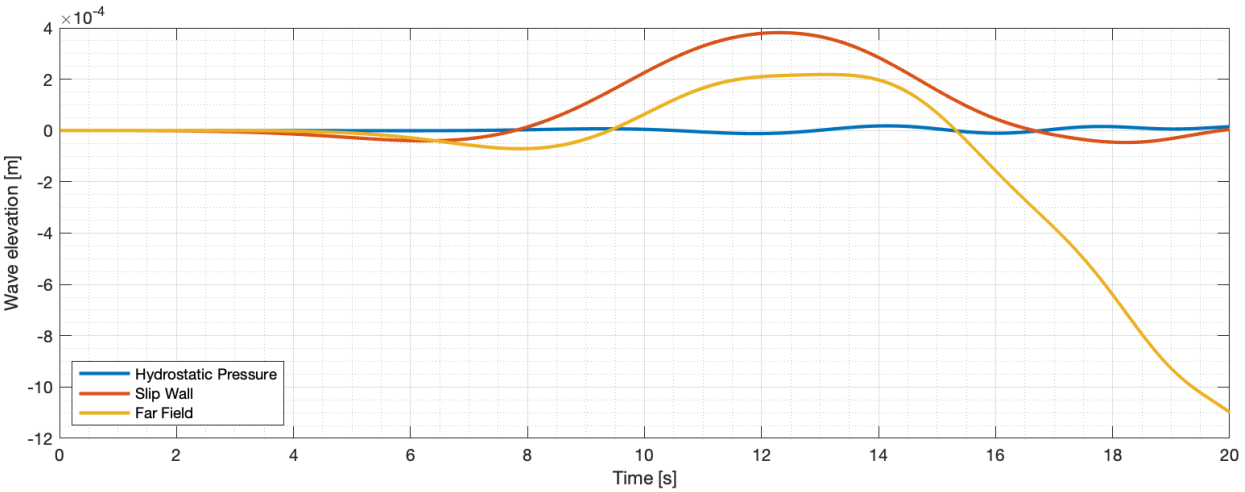


Figure 6.4.4: Detailed plot of graph 6.4.3 from t=0 to t=20 [s].

6.5 Discretization Scheme

6.5.1 Analysis Approach

Choice of discretization scheme can affect the results, especially while using two-phase flows simulating the free surface. Three different recommended discretization schemes were tested and the results presented in the next section. The different discretization schemes are listed on the following page, and their theoretical background are presented in chapter 2, section 2.3.8.

Settings presented in this paragraph were used for this respective discretization scheme sensitivity study. A total height of 20 [m], length of domain equal to 62.5 [m] and damping zone length equal to 3.5 times the wavelength. Boundary conditions as mirror for both front and back face, and external condition with far field on vertical boundaries and updated hydrostatic pressure on horizontal boundaries. Unsteady time configuration, multi-fluid model, initial solution of 0 in both x- and y-direction. Control variables as: maximum 20 non-linear iterations, convergence criteria of 2 orders, save solution every 4 000th time steps. The simulation length was selected to be 8 000 time

steps of size 0.01 [s]. Outputs activated were surface/volume-probe for saving the mass fraction every time step, and wave probes as described in chapter 4, section 4.1.

1. AVLSMART
2. BICS
3. BRICS

6.5.2 Results

In figure 6.5.1 free surface elevation represented by isoline 0.5 of the mass fraction at time step 8 000 is presented. It shows that by use og BICS and BRICS, the result is approximately the same, as seen in figure 6.5.2. By use of AVLSMART, both wave crest and through is kept closer to constant downstream the tank, while by use of the two other, a slight decrease in wave energy is detected. However, it also looks more smoothed than the other two, raising the question of how smoothing affects the overall accuracy. Additionally, free surface height increases at the tank end using the AVLSMART scheme.

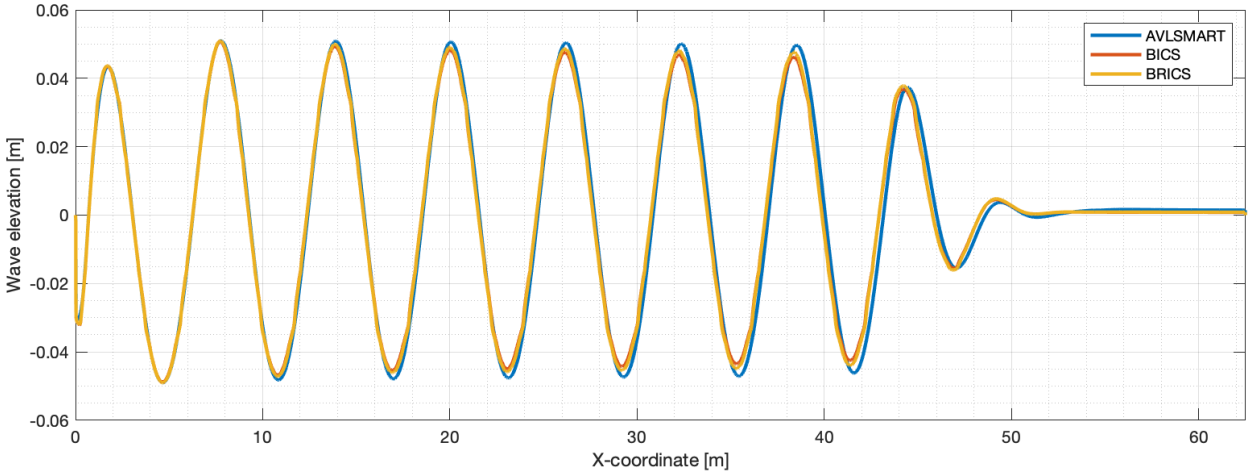


Figure 6.5.1: Free surface elevation downstream numerical wave tank for three different discretization schemes.

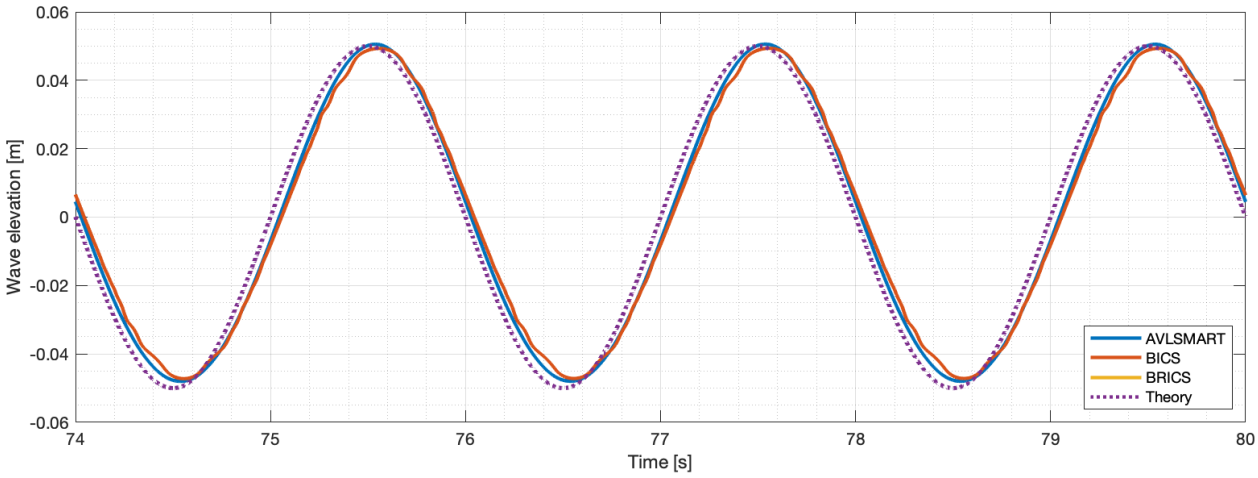


Figure 6.5.2: Plot of the free surface elevation over time at wave probe 2 for the three different discretization schemes.

A quantification of wave statistics through mean free surface, standard deviation and variance, of the generated wave elevation time series are presented in table 6.5.1. MWL stands for mean water level, Std for standard deviation and Var for variance. Results are for all calculations during the dependency studies. The results found in the table are discussed further in the following section.

Table 6.5.1: Wave statistics for waves simulated during the dependency studies.

Dependendency study	**Parameter**	**MWL** [mm]	**Std** [-]	**Var** [-]
Length	37.5 [m]	- 5.3	0.0320	0.0010
	50.0 [m]	- 4.9	0.0321	0.0010
	62.5 [m]	- 4.4	0.0322	0.0010
Height	6 [m]	- 4.8	0.0327	0.0011
	14 [m]	- 4.3	0.0322	0.0010
	18 [m]	- 4.4	0.0323	0.0010
Damping zone	2.5 x WL	- 4.2	0.0323	0.0010
	3.0 x WL	- 4.1	0.0323	0.0010
	3.5 x WL	- 4.1	0.0322	0.0010
	4.0 x WL	- 4.0	0.0322	0.0010
Boundary condition	Hydopress	0.2	0.0328	0.0011
	Slipwall	- 0.4	0.0327	0.0011
	Farfield	- 4.1	0.0322	0.0010
Grid/mesh	AGR	0.2	0.0335	0.0011
Discretization	AVLSMART	0.2	0.0338	0.0011
	BICS	- 0.3	0.0330	0.0011
	BRICS	- 0.0	0.0000	0.0000

6.6 Discussion

In the **overall dimension** sensitivity study found in section 6.1 the longest tank length gave the most accurate result, thus being the length used for further analysis in the present thesis. Compared to a physical towing tank, e.g. the tank at Tyholt with dimensions 175 x 10.5 x 5.6 [m] (*length x width x depth*), the model length of 62.5 [m] is short. One of the perks of using CFD is that it is possible to model without scaling, which can give more realistic results. Using physical experiments, scaling often cause a big source of uncertainties. This has already been pointed out by Zhe et al. [30] that wrote: "Compared with laboratory tests, numerical wave tanks can also simulate waves under realistic scales, which is difficult to accomplish in laboratory experiments." Even though the height sensitivity analysis showed similar results for water depths higher than 3 [m], a depth of 10 [m] were applied in the next chapters, with a coarse mesh close to the upper and lower boundaries. Again, if this is compared to the wavetank at Tyholt, this is more than enough to avoid shallow water effects.

Sinkage was a problem at the early phase of performing the dependency studies, as in the project thesis. Another observation worth noticing was that the problems with sinkage only occurred using external far field boundary together with the IWG, i.e. there were no sinkage problem using the WG.

Wave probe 5 is taken 0.5 [m] from the wave tank end, at the end of the **wave damping zone**, where the boundary was defined as zero velocity. The ideal scenario would bee zero velocity on the graph (constantly zero with no slope). Due to sinkage, this did not occur for this analysis. However, the

sinkage stays below -10% of the total wave height in the time series of 80 [s]. When the same settings were applied and waves generated by WG, this resulted in a elevation below -0.5% of the total wave height. Koo et al. [26] discussed how controlling the energy reduction of a long wave is difficult even with an efficient artificial damping zone installed, and therefore this damping zone length was accepted in this thesis. Additionally, previous papers have used shorter damping zone lengths and gotten results with sufficient accuracy. Bihs et al. [27] used a numerical beach of 2 wavelengths where the pressure was set to its hydrostatic values based on the still water level, and Zhe et al. [30] used 1 wavelength while commenting that this was lower than the recommendations of more than twice the wavelength.

Damping zone investigations were performed using the IWG, and different damping effects using the two methods were discovered at a later point in the thesis. Normally, sponge layers are porous media which works in a passive way of dissipating incident wave energy [28]. As stated in the documentation [6], the sponge layers in FINE/Marine has an adaptive effect which works better by use of the IWG, as discovered in this thesis. This questions whether the damping zone length of 3.5 wavelengths, as used further in this thesis is sufficient. More about this will be discussed at a later point.

Firstly, performing the **mesh refinement** study, the target cell size in the surface refinement from the project thesis was decreased. Further, a study of how changing the aspect ratio affected the results were performed, resulting in a neglectable difference. AGR gave better results regarding the wave crest and through maximas, and is therefore recommended when the computational power is available. However, it is not always that using AGR give better results, since analysis consist of a play between computational cost/time and accuracy. In the plot comparing use of AGR and the mesh from the project thesis, AGR was clearly the better option. Using irregular waves, when the wave profiles are not as predictable as for regular waves, AGR was applied throughout this thesis.

The number of cells created during the simulation using AGR can be controlled by looking at the file *nb_cell.dat* which contain the total number of cells created during each computation. After a simulation is finished, the evolution can be checked by using *Result Analysis* tool found in the GUI. Memory consumption can be controlled in several ways. Both mesh and solution is stored, and since the mesh size vary while using Adaptive Grid Refinement, the solver will allocate a lot of space and partly fill it up. If the .dat-file shows that the maximum approaches 100%, the settings should be changed. Regarding settings for saving the surface probe data, one should consider not saving every time step as this will fill up the memory quickly. Wave probes save the elevation every time step for accurate statistics. Surface probe data of the mass fraction can then be used to check grid quality, and visually see how the flow evolves.

As mentioned in section 6.4, about **boundary conditions**, far field gives poor results compared to slip wall and updated hydrostatic pressure. This might be because far field is often used for more "far away from the problem" types of situations. Miquel et al. [28] used symmetry planes for both side walls an the top of the tank, and the bottom boundary as no-slip wall. This gave the idea to try out some similar boundaries. Free surface elevation in the damping zone stayed below 0.7% of the total wave height using the updated hydrostatic pressure. Additionally, as seen in table 6.5.1, updated hydrostatic pressure gave the mean water level closest to the theoretical at 0.0 [mm].

A **time step size** refinement study was performed in the project thesis, with the conclusion that a time step of 0.01 [s] is the coarsest size that should be applied. In the present thesis a smaller time step could have been used, but with regards to computational time a concious choice of using 0.01 [s] was made. With a finer time step size the calculations would have been even heavier, and only a few computations could have been performed in the time range when this thesis was done.

Muniyandy Elangovan [46] performed a study highlighting that a time step of 0.01 was too coarse, but still sufficient to get accurate results. One suggestion could have been to use a finer time step after performing all sensitivity analysis, to ensure stability and accuracy in the final result.

7 | 2D Regular Waves

Guidelines for wave generation were made based on the dependency studies presented in chapter 6, and given in the following section. To further investigate the validaty range for the guidelines presented, 9 waves with different wave properties were tested for each wave generation method, in total 18 different computations. The case-numbering and wave properties used are presented in tables 7.0.1 and 7.0.2 below. Note that in table 7.0.1, the orange marking represent wave generation using IWG.

Table 7.0.1: Numbering of the performed computations, with the internal wave generator (IWG) marked in orange and the wave generator (WG) in white.

T [s] / **H** [m]	**1.5**		**2.0**		**2.5**	
0.1	1	2	3	4	5	6
0.2	7	8	9	10	11	12
0.3	13	14	15	16	17	18

Table 7.0.2: Wave properties of generated waves.

Wave	**T** [s]	**L** [m]	**H** [m]	**H/L** [-]	**k** [-]	**c** [m/s]
A	1.5	3.512947	0.1	0.028466	1.7886	2.3420
B	1.5	3.512947	0.2	0.056932	1.7886	2.3420
C	1.5	3.512947	0.3	0.085398	1.7886	2.3420
D	2.0	6.245240	0.1	0.016012	1.0061	3.1226
E	2.0	6.245240	0.2	0.032024	1.0061	3.1226
F	2.0	6.245240	0.3	0.048036	1.0061	3.1226
G	2.5	9.758187	0.1	0.010247	0.6439	3.9033
H	2.5	9.758187	0.2	0.020495	0.6439	3.9033
I	2.5	9.758187	0.3	0.030743	0.6439	3.9033

Both the wave source point using IWG, and general wave probe (WP) placement, were dependent of the wavelength. Placement of wave probes along the x-axis are listed in table 7.0.3. Some of the WPs were placed within the damping zone, and they are marked with a parenthesis around the number representing the distance along the x-axis.

Table 7.0.3: Placement of wave probes along the x-axis given in meters.

T [s]	**WP 1**	**WP 2**	**WP 3**	**WP 4**	**WP 5**
1.5	3.51	7.03	10.54	14.05	(60.00)
2.0	6.25	12.49	18.74	24.98	(60.00)
2.5	9.76	19.52	(29.27)	(39.03)	(60.00)

Placements along the x-axis given in parenthesis
illustrates that they are placed within the damping zone.

7.1 Guidelines for Regular Wave Generation in FINE/Marine

Generation of numerical wave tank

When designing the CAD model of the numerical wave tank, the overall dimensions must be thoughtfully selected. The tank length must be long enough to include several wave lengths and a long damping zone. However, the length is of less importance than the height, which should be below the deep water limit of $D = L/2$.

Boundary conditions are always important to model the physical problem correctly, and especially of interest while using IWG. Therefore, recommended boundary condition on both upper and lower domain walls (in 2D y-min and y-max) is external condition with updated hydrostatic pressure. Mirrors should be applied to the front and back surface to represent a symmetry boundary. Finally, external condition with far field is recommended for the tank end using WG, and both tank inlet and outlet using IWG.

The FINE/Marine user manual [5] recommend a damping zone length of 3 wavelengths. In the present thesis, a damping zone length of *at least* 3.5 wavelengths is recommended.

Discretization

A refined mesh around the air-water interaction zone is especially important to capture the free surface accurately. Restrict the maximum number of refinements according to the initial mesh and the target cell size. Apply surface refinement along an internal surface representing the interface with target cell size of $dx = 0.0625$ [m] and $dy = 0.0039$ [m]. Ensure that trimming is not used at any surfaces. If the available computer resources are high, adaptive grid refinement (AGR) should be applied. For regular waves the effect of enabling AGR is not significant regarding accuracy, but it generally refines the surface in an appropriate matter. Settings using AGR are described in chapter 8, section 8.1.

Apply the default scheme BRICS for multi-fluid discretization as it showed better behavior than all other schemes tested in this thesis.

A time step size of 0.01 [s] should be the coarsest applied time step if enough computer capacity is available. It is preferred to use a finer time step .

Wave Generation

Two methods can be applied for generating waves: the internal wave generator (IWG) within the domain or the wave generator (WG) placed at an external boundary condition. As validated in this thesis, both methods generate waves of high quality. Due to limited damping capacity using the

WG, this thesis recommend using the IWG. Remember that if you apply the internal wave generator, the source location should be half the generated wavelength. Additionally, using IWG, wave probes should be placed one wavelength from the wave generation zone.

Validation of the waves should be performed to ensure right statistical properties of the generated waves. By use of the elevation time series captured at the wave probe(s), statistical properties should be evaluated, e.g. mean wave height, wave spectrum, minimum and maximum wave crest and through heigts, standard deviation, variance and mean water level. All giving indications of how accurate the waves are according to either theory or experimental data.

Calculation Setup

When it comes to simulation length, this thesis recommend at least 60 seconds, preferably longer, to ensure stability. The residuals need to be checked to be certain that the simulation is stable, note that they will have small fluctuations due to waves present. With the coarse time step size of 0.01 [s], this gives at least 6 000 time steps.

Solver settings applied should be maximum 20 non-linear iterations, convergence criteria of 2 orders and save solution as often you want to. Saving frequently will influence the computation time. If the computation crashes, you will have backup of the results, and it it also possible to begin post-processing in HEXPRESS after the first saving.

7.2 Results

In this section a selection of the results from the 18 computations are presented to highlight the applicability and accuracy of the suggested guidelines for regular wave generation. All results are discussed further in the following section.

The first plot, in figure 7.2.1, presents the wave elevation at wave probe 2 by use of both the IWG (computation 3) and WG (computation 4), trying to simulate wave *D*. Note that the phase difference between the two methods is accounted for by subtracting half of the wave period from the time-axis of wave elevation using IWG. Both waves generated by IWG and WG corresponds well in comparison to theory.

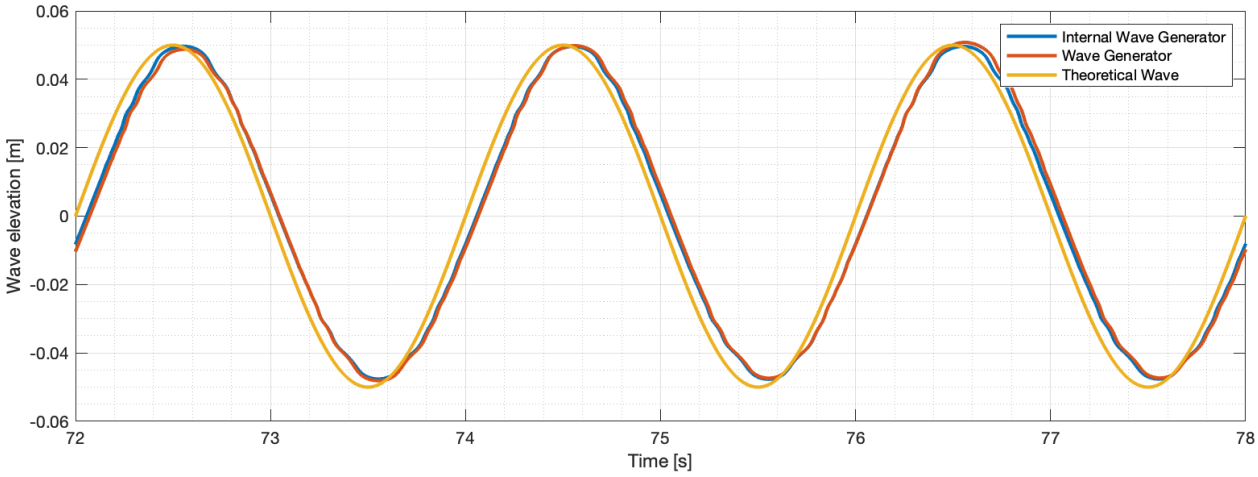


Figure 7.2.1: Plot of free surface elevation at wave probe 2 for computations 3 and 4 compared to a theoretical wave D.

Quantification of the wave statistics of the time series are presented in table 7.2.1, where MWL = Mean Water Level, Std = Standard Deviation, Var = Variance, Min = Minimum water level, Max = Maximum water level and Range the difference between Min and Max, representing the wave height. The percentage deviation of the range from the theoretical of 0.1 [m], varies in the range from 3.2% to 8.9%. Also, note that by use of IWG the MWL is slightly above the theoretical of 0.0 [m], whilst by use of WG the deviation is even less than for IWG, and below zero. Standard deviation and variance are low for both methods, enhancing the model strength.

Table 7.2.1: Wave statistics for waves simulated during computations (comp) 3 and 4, whilst propagating down the wave tank, trying to simulate wave D.

		MWL [m]	**Std** [-]	**Var** [-]	**Range** [m]	**Min** [m]	**Max** [m]
Comp. 3	WP 1	1.5269e-4	0.0351	0.0012	0.1034	-0.0505	0.0529
	WP 2	3.0023e-4	0.0333	0.0011	0.1076	-0.0517	0.0559
	WP 3	3.0690e-4	0.0319	0.0010	0.1080	-0.0517	0.0564
	WP 4	3.2117e-4	0.0307	0.0009	0.1059	-0.0523	0.0537
Comp. 4	WP 1	-1.4084e-4	0.0339	0.0012	0.1098	-0.0497	0.0601
	WP 2	-0.8773e-4	0.0326	0.0011	0.1061	-0.0508	0.0553
	WP 3	-0.9553e-4	0.0312	0.0010	0.1088	-0.0517	0.0571
	WP 4	-0.6815e-4	0.0300	0.0009	0.1076	-0.0530	0.0547

Figure 7.2.2 and 7.2.3 are plots of free surface elevation at wave probe $1-4$ for computation 3 and 4 respectively, both simulating wave *D*. Firstly, note the phase difference of half a period between the two plots. Secondly, small phase shifts are seen downstream the wave tank. From WP 1 to WP 2 it shifts right, from WP 2 to WP 3 it shift even more, and so on. By use of the IWG the wave shifts more than with the WG. Absolute value of wave crest height and wave through height decrease slightly downstream the wave tank, more by use of the IWG than WG. Both still presenting waves with high accuracy. Table 7.2.1 include statistical information of these waves, to quantify the accuracy of the simulated waves.

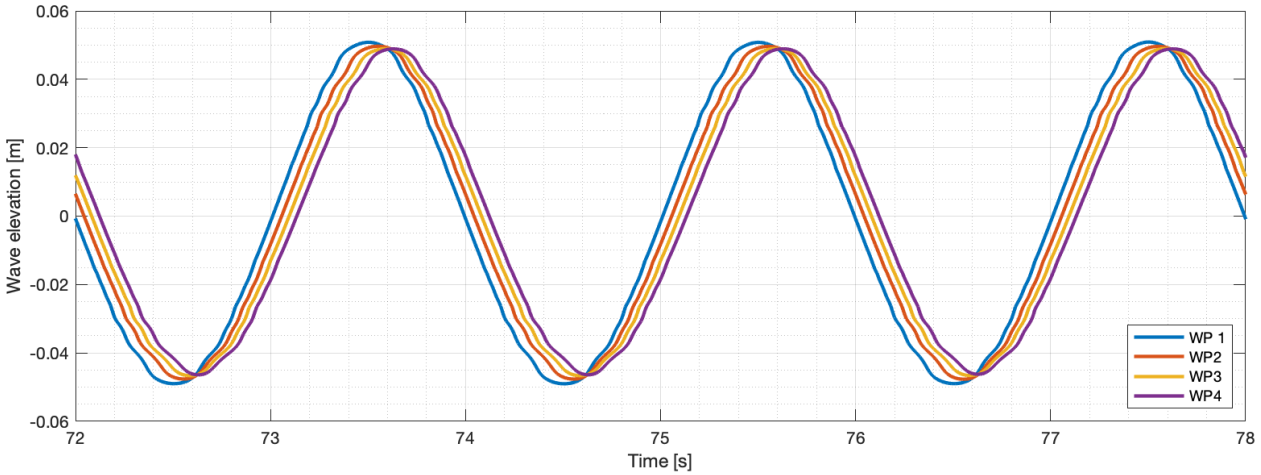


Figure 7.2.2: Plot of free surface elevation at wave probe 1-4 for computation 3 (IWG), simulating wave D.

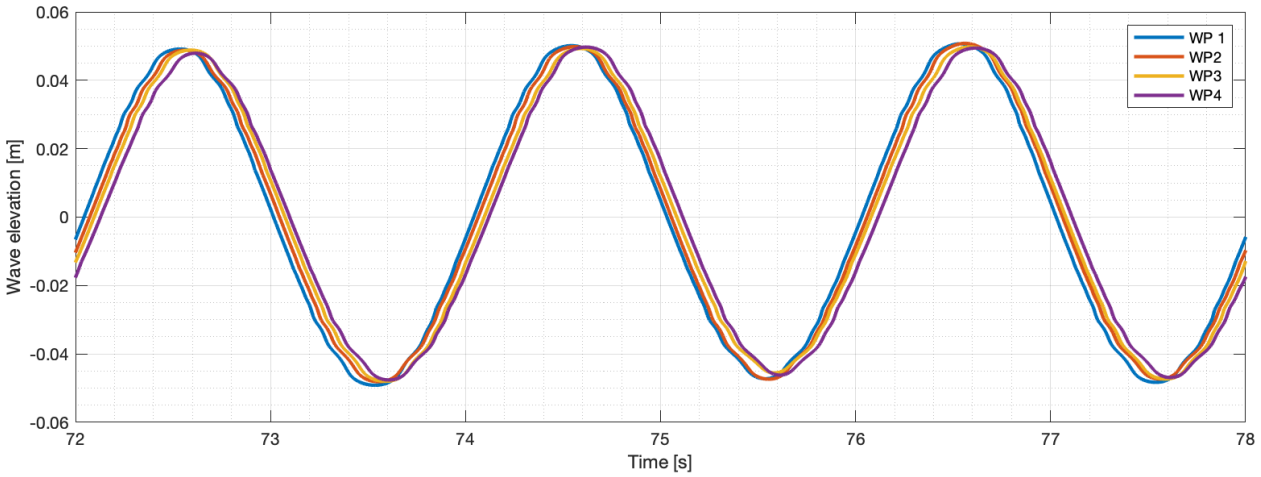


Figure 7.2.3: Plot of free surface elevation at wave probe 1-4 for computation 4 (IWG), simulating wave D.

Conservation of wave energy while waves propagate down the numerical wave tank is another important part of simulating accurate waves. Numerical diffusion might be a reason for the loss in energy. In a perfect world, the area under the graph of the wave spectrum, i.e. the total wave energy, should be constant throughout the domain. Figure 7.2.4 present the wave spectrums for computation 3 and 4 for both methods, resulting in 4 lines for each wave generation method. Dotted lines present spectrums for the WG and full lines the IWG. Several trends can be seen; (1) waves have higher spectral density and total energy using IWG compared to WG, (2) as the waves propagate down the tank maximum spectral density and total energy decrease, and (3) at wave probe 2 the spectral density is approximately the same for IWG and WG.

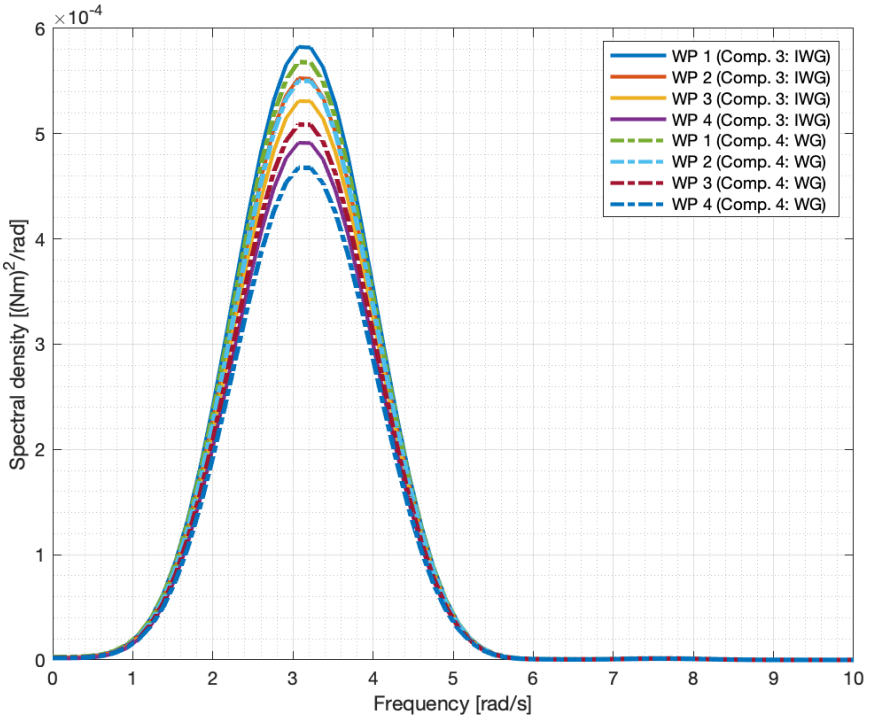


Figure 7.2.4: Spectrums for computations 3 and 4 at wave probes 1-4 (comp. 3: IWG, comp. 4: WG).

Table 7.2.2 and 7.2.3 present the maximum spectral density and the percentage decrease in maximum spectral density with regards to wave probe 1. The peak frequency for all spectrums are $\omega_p = 3.0680$ [rad/s] $= 0.49$ [Hz], and deviates with -2.0% from the theoretical spectrum with peak frequency

$1/T = 0.5$ [Hz]. From table 7.2.3 the decrease in maximum spectral density and hereby total energy is decreasing faster for IWG at WP 2, and at WP 3 and 4 the WG decreases the most.

Table 7.2.2: Spectral density for computation 3 and 4.

	MSD WP 1	**MSD WP 2**	**MSD WP 3**	**MSD WP 4**
Comp. 3	5.8252e-4	5.5276e-4	5.3095e-4	4.9137e-4
Comp. 4	5.6814e-4	5.5017e-4	5.0874e-4	4.6770e-4

MSD = Maximum Spectral Density given in [(Nm)^2/Hz]

Table 7.2.3: Persentage decrease in Maximum Spectral Density (MSD) with regards to MSD at WP 1.

	DMSD WP 2	**DMSD WP 3**	**DMSD WP 4**
Comp. 3	5.38 %	9.71 %	18.55 %
Comp. 4	3.26 %	11.76 %	21.47 %

DMSD = Percentage Decrease in Maximum
Spectral Density with regards to MSD at WP 1

To validate the waves, and further understand the physics of the numerical simulation, a surface probe capturing the mass fraction was applied. Figure 7.2.5 and 7.2.6 shows plots of mass fraction 0.5, representing the free surface, for computation 3 and 4 respectively. Both plots show the theoretical wave D and the wave zone illustrated in gray color. An interesting feature is how the sponge layer damps the waves differently by use of the two wave generation methods. The sponge layer in FINE/Marine is clearly much more efficient using the IWG than WG. This might be a reason for the troubles with computation 16, as described below. By use of both IWG and WG the wave period decreases slightly downstream the tank. This coincide good with the decreasing total wave energy.

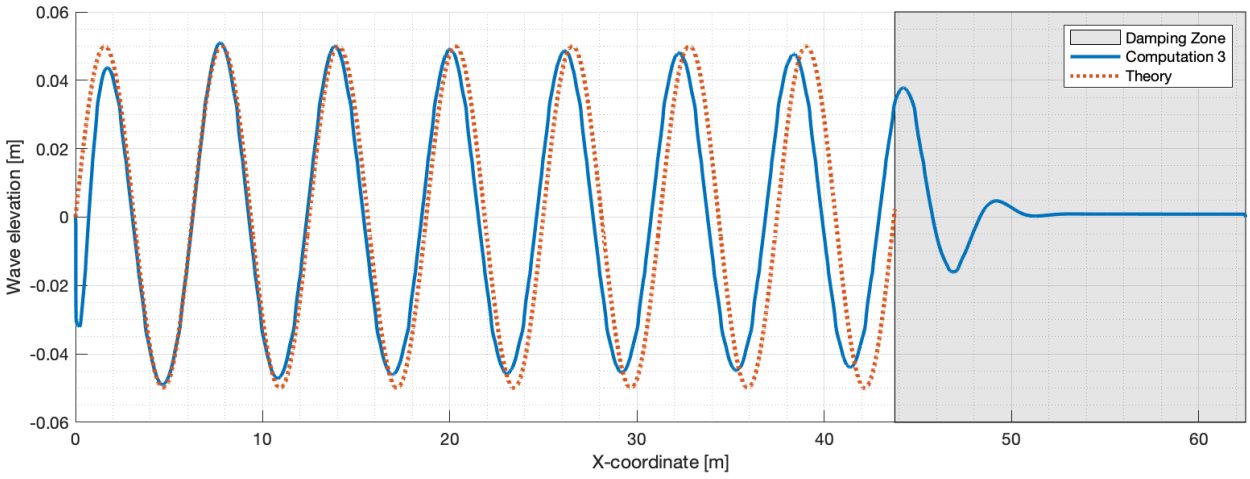


Figure 7.2.5: Mass fraction 0.5 representing the free surface in the whole domain for computation 3 using IWG.

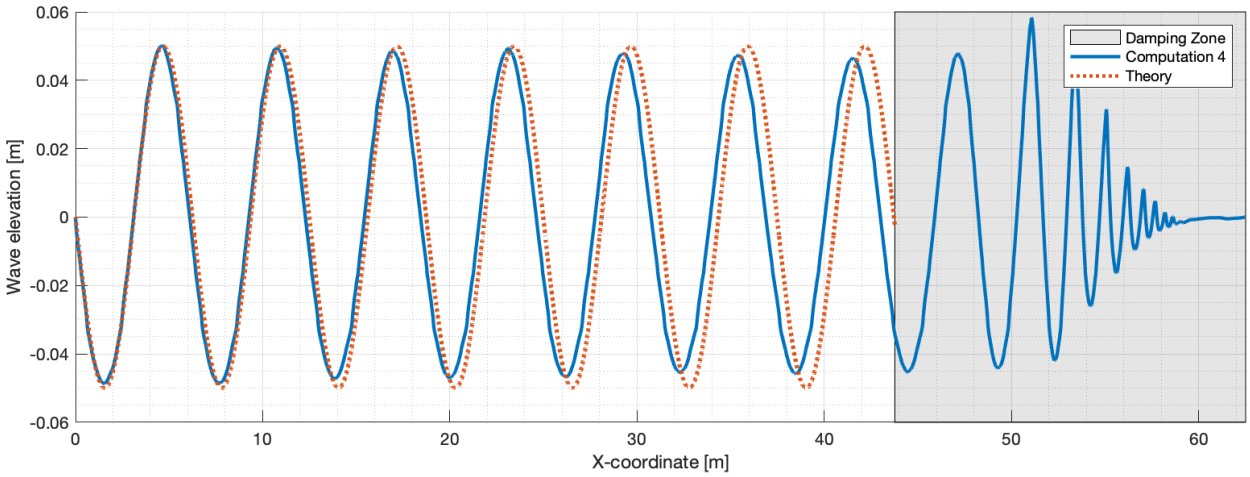


Figure 7.2.6: Mass fraction 0.5 representing the free surface in the whole domain for computation 4 *using WG.*

Until now, results have been presented only for computation 3 and 4. 16 additional computations were performed to investigate how applicable the proposed guidelines are. A summary of the most important wave statistics from the computations are presented in table 7.2.4. Calculation 16 exploded all times it was tried, showing that it lays outside the validity range of the presented guidelines. In appendix A.2 seven figures illustrating the mass fraction of computation 16 are presented. They show how the computation is unstable and eventually explode, from time step $t = 35.6$ to 38.0 [s] with a time step value of 0.4 [s].

The wave statistics from wave 1 and 2, 3 and 4, 5 and 6, etc. have similar standard deviation and variance, showing a correlation betweenthe wave generation methods. Mean water level for all computations, except number 16, are of order $e-4$ [m].

Table 7.2.4: Wave statistics calculated from time series of all 18 calculations.

	Mean Water Level [m]	**Standard Deviation** [-]	**Variance** [-]
Calc. 1	1.72e-4	0.0649	0.0011
Calc. 2	4.25e-4	0.0313	0.0010
Calc. 3	3.00e-4	0.0333	0.0011
Calc. 4	-0.87e-4	0.0326	0.0011
Calc. 5	2.19e-4	0.0337	0.0011
Calc. 6	-2.77e-4	0.0326	0.0011
Calc. 7	-1.28e-4	0.0629	0.0040
Calc. 8	-0.62e-4	0.0605	0.0037
Calc. 9	3.75e-4	0.0659	0.0043
Calc. 10	-3.48e-4	0.0649	0.0042
Calc. 11	6.01e-4	0.0668	0.0045
Calc. 12	-4.94e-4	0.0645	0.0042
Calc. 13	-5.77e-4	0.0881	0.0078
Calc. 14	0.12e-4	0.0850	0.0072
Calc. 15	5.09e-4	0.0984	0.0097
Calc. 16	1.6745	4.7344	22.4148
Calc. 17	8.80e-4	0.0999	0.0100
Calc. 18	-7.55e-4	0.0981	0.0096

7.3 Discussion

Figure 7.3.1 is collected from the theory guide [6] and presents the estimation of wave order used in FINE/Marine. The validity of the generated waves is shown, where the green line border separate first and second order with a maximum steepness of 0.14. It can only be applied for intermediate to deep water. All waves tested in this thesis are governed by linear theory, and should therefore correspond to linear theory.

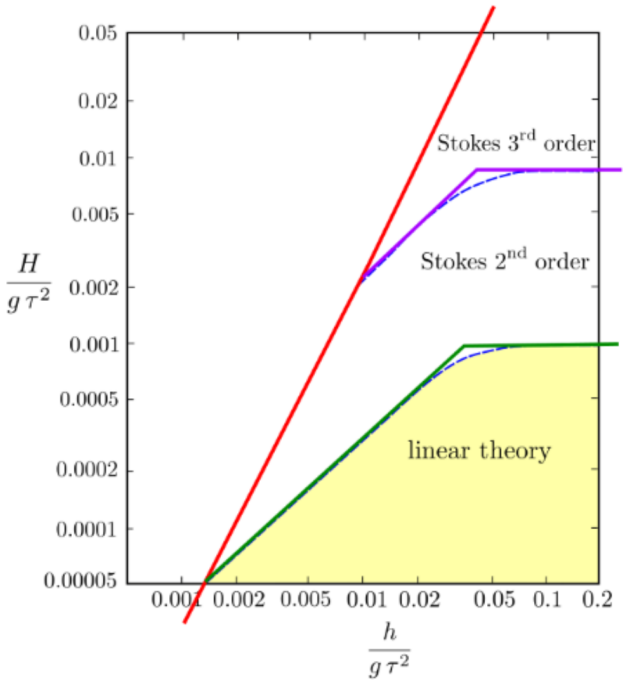


Figure 7.3.1: Estimation of the wave order, figure 19 in the theory guide [6].

To include a brief part of how the waves could excite structures, the excitation time scales can be reviewed. This adds to the physical understanding of the accurate wave generation. In *TMR4225 Marine Operations* the syllabus explain how first order wave excitation forces result in motions in wave-frequency range of $5 - 30$ [s]. Second order wave drift forces results in mean and low-frequency, resonant motions in the range $30 - 500$ [s].

Regarding placement of wave probes, inspiration has been draught from Miquel et al. [28] which used four wave gauges with placement dependent of wavelength. Additionally, they used waves with different heights and lengths, i.e. different steepness, to validate the model. Each of their computations were 90 [s] long. Their setup was similar to the present thesis regarding wave probe placement, computational length and validation method.

Figure 7.2.1 and table 7.2.1 highlight that by use of the proposed guidelines, waves close to theory are generated, and both wave generation methods provide waves with minimal statistical differences from each other. A small phase difference between the theoretical and simulated waves are notes, i.e. the generated waves have a small shift to the right. This can be because of energy losses that acumulate downwords the wave tank and affect the wave period. By observing the two plots in figure 7.2.2 and 7.2.3, this is verified. A small and clear shift to the right is detected as the wave propagate down the tank. Small numerical inaccuracies, e.g. simulated wave length or period inconsistent with theory, accumulate as they propagate and become larger.

During this analysis, energy losses were detected all the way down the tank, as seen in the energy spectrums presented in figure 7.2.4. Numerical diffusion at the interface between the two fluids can be a reason for this energy loss. A possible way to investigate this is to make the mesh around the surface even finer, and lower the time step, to see if that gives a lower energy loss. However, some energy losses are expected using two-phase flows due to its complexity. Baquet and Kim [34] investigated how energy loss could be compensated by applying factors of wave spectral frequency components on the input wave spectrum. Further work could include an investigation of this. A trend of more energy during wave generation with IWG compared to WG was observed. This indicate that a small energy loss at the boundary is connected to use of WG.

Residuals in FINE/Marine were controlled for all computations, ensuring that the simulations were stable. The residuals converged with a fluctuating component due to the presence of waves.

In chapter 6 different behaviours of the sponge layer was mentioned, and below a detailed plot of the two methods applied to wave *D* is presented in figure 7.3.2. The sponge layer should dissipate the wave energy in the damping zone, as it does clearly for the IWG method. However, by use of WG, the damping zone is significantly less effective in energy dissipation. As mentioned in the theory section, the WG creates waves at the defined boundary condition by imposing a velocity field *u* and a time dependent wave height. IWG create waves within the domain by adding a momentum source in the NS equations. Since the sponge layer apply Darcy's law to damp the momentum in z-direction, this can be a reason why better damping are experienced in the IWG computations. Further work could include an investigation of another possible damping method in FINE/Marine, the *Multifluid Smoothing*.

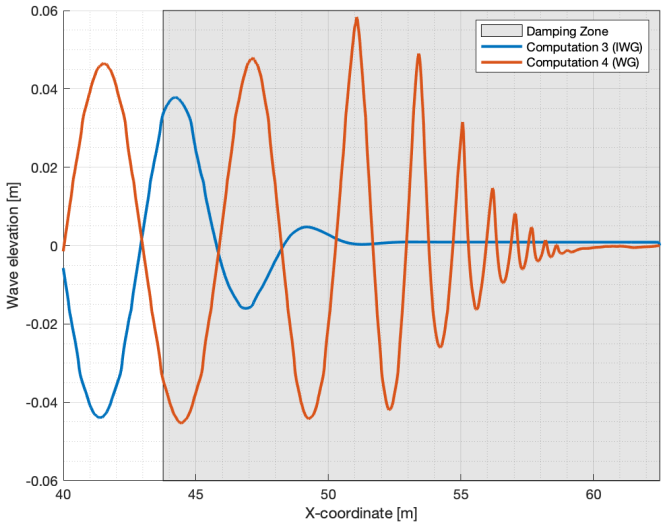


Figure 7.3.2: Detailed part of the isoline representing mass fraction 0.5 in the damping zone for computation 3 and 4.

The ideal scenario would be that these proposed guidelines would be sufficient to capture all types of regular sinusoidal waves governed by linear theory, as illustrated in figure 7.3.1. This is not the case, as several of the tested waves falls outside the validity range of the guidelines. In appendix A.1 the isoline representing mass fraction 0.5 is plotted for all wave simulations, except compuation 16 that exploded unabling the possibility to extract the surface probe data from HEXPRESS. A list shortly summing the conclusions of the computations are found on the next page.

1. Computations 1 and 2 experiences a small decrease of wave height downstream the tank, and the same problems with damping the waves generated with the WG as previously mentioned.

2. Computations 5 and 6 experiences the same damping challenges, but wave height remains good acording to theory throughout the domain. By visually observing the elevation at the end of the damping zone, it looks like some reflection problems are starting to occur by use of the WG.

3. In computation 7 there is a clear tendency that the wave height decrease downstream the tank with almost 45% of the original wave height at $x = 40$ [m]. The overall wave height constantly decrease in computation 8, and reach a 90% reduciton of overall wave height. For both calculations the maximum wave throughs are above the theoretical from the beginning of the domain. Additionally, a clear disturbance can be seen between $x = 50$ [m] and the end of the domain. This implies that some reflections have ocurred.

4. Computations 9 and 10 tend to have the same behaviour as computation 3 and 4 respectively. Also here some irregularities can be seen at the end of the damping zone for the WG simulation. The same applies for computations 11 and 12.

5. In computations 13 and 14 the same tendencies as in computation 7 and 8 are observed. An even clearer reflection is starting to propagate in negative x-direction as observed around $x = 56$ [m].

6. Computation 15 show good comparison to theory, descptite a small decrease in wave height at the end of the tank. This computation was also performed using IWG, and in this computation an increased mean surface is detected in the damping zone, i.e. aproximately $y = 0.01$ [m].

7. Computations 17 and 18 also behaves as computatin 3 and 4. The only difference is a slightly more visable problem in the damping zone, which could be due to difficulties of dissipating energy of long waves using sponge layers [28].

Table 7.3.1: Overview of validated computations (marked green) and waves outside the validity range (marked red) of the proposed method.

H [m] \ **T** [s]	**1.5**		**2.0**		**2.5**	
	IWG	WG	IWG	WG	IWG	WG
0.1	✓	✓	✓	✓	✓	✓
0.2	X	X	✓	✓	✓	✓
0.3	X	X	✓	X	✓	✓

Table 7.3.1 present an overview of the validated compuations, where the green cheked boxes means that the wave is validated with sufficient accuracy against analytic solutions. Waves *B* and *C* did not work with any of the two wave generation methods. They were the steepest waves tested, with steepness equal to 0.057 and 0.085 [-] respectively. The third steepest wave was *F* with steepness 0.048. This wave caused difficulties using WG, and was the computation that crashed each time. Wave *E* and *I* followed with steepness 0.032 and 0.030 respectively. This implies that the validity of the guidelines is for waves with steepness lower than 0.030. In the theory chapter, section 2.1.2, it was mentioned that Pettersen [13] used a limit of $1/20$ for Stokes' waves to be valid, and ITTC [15] recommended a steepness around $1/50$ for applications using ship models. With this in mind, a steepness criteria of $1/33$ could be acceptable for commercial cases. Another aspect of these

simulations is that they are only compared to theory. In a real life ocean there is no such thing as a regular sinusoidal wave. However, it is important to be able to recreate regular waves since irregular seas are discribed through superpositioning of regular waves. Numerical calculations will as of today not simulate the physical reality 100% as numerical solutions cannot fully recreate physical phenomena, e.g. turbulence, etc.

To reuse Molin's review of basic problems with numerical wave generation from 2001, which was mentioned in the introduction; "... As a matter of fact, even though it may look simple, generating a Stokes regular wave in a tank is impossible." [9, p.525]. In the present thesis regular waves with a limited steepness were generated with high quality and low uncertainty, which are sufficient to use in wave-structure interaction simulations.

8 | 2D Irregular Waves

While simulating irregular waves, a long simulation time is essential to ensure that all components of the applied theoretical spectrum are generated. In the two courses *TMR4182 - Marine Dynamics* and *TMR4215 - Sea Loads* the rule of thumb is 4 [hours] simulation time. Co-supervisor Eloïse Croonenborghs at SINTEF Ocean state that 3 [hours] are sufficient for validating the simulation properly. In the present thesis, due to limited computer capacity, a length of 500 [s] is used. Miquel et al. [28] showed that 90% of the theoretical spectrum components were generated within the first 500 [s], and Zhi-Fu et al. [31] got results with a sufficient accuracy using only 200 [s] .

8.1 Guidelines for Irregular Wave Generation in FINE/Marine

The guidelines described in this section differs from the ones given in section 7.1 by mesh generation method and solver settings. As previously mentioned, the FINE/Marine user manual [5] only recommend using adaptive grid refinement (AGR) while simulating irregular waves, since there are less control of how fine the grid need to be close to the free surface since it is constantly changing properties. Irregular waves can be generated in FINE/Marine either by use of the internal wave generator or the wave generator at a boundary, both simulating waves of high quality.

An initial mesh with cell size of 0.5 x 0.5 [m] should be applied. Further use of surface refinement resulting in a fine and very regular mesh, giving cells as close to rectangular as possible when applying AGR. Use a target grid spacing normal to free surface as 0.004 [m] and 0 [m] as a minimum size limit for refined cells. Diffusion criterion of 2 layers copying the full criterion value, and 1 layer copying a fraction value of 0.6. Apply longitudinal direction only allowed in the boundary layers. Under computation control, use 250 steps before first call to the refinement procedure with 50 as number of steps between each call. The default value of 500 000 cells, as a maximum for computation control can be applicable on all standard computers. If you have more capacity available, a recommendation of 100 000 $-$ 1500 000 cells for each GB of memory should be used as a maximum.

With regards to solver settings, the only difference between the regular and irregular analysis is the simulation length. This thesis suggest using at least 500 [s] if the computational resources are limited, however one must then consider using safety factors, as this relatively short duration provides uncertainty of the result. A simulation length of 3 [hours] are preferred if possible.

8.2 Results

This section present the result of two-dimensional irregular wave generation using FINE/Marine. Two results are highlighted: (1) irregular waves generated by use of IWG over 500 [s], and (2) irregular waves generated by use of WG over 100 [s]. Time series of the wave elevation is plotted, their energy spectrums presented and compared to theory, and its statistical properties listed.

As described in section 4.1.5, the irregular waves generated in this chapter are by use of the ITTC spectrum with following characteristics: $H_S = 0.1$ [m], $h = 10$ [m], $T_P = 2$ [s], source location $(x, y) = (3.12262, 0)$ and propagation in positive x-direction.

The time series of the first highlighted computation is presented in figure 8.2.1. It plots the time series of the free surface elevation captured at wave probe 2, 2 wavelengths downstream the numerical wave tank. Figure 8.2.2 plots the wave spectrums of the time series at wave probes 1 to 4 against the theoretical ITTC spectrum. Miquel et al. [28] showed that 90% of the theoretical spectrum components were generated within the first 500 [s], and further investigations should be performed before this conclusion can be drawn for this case.

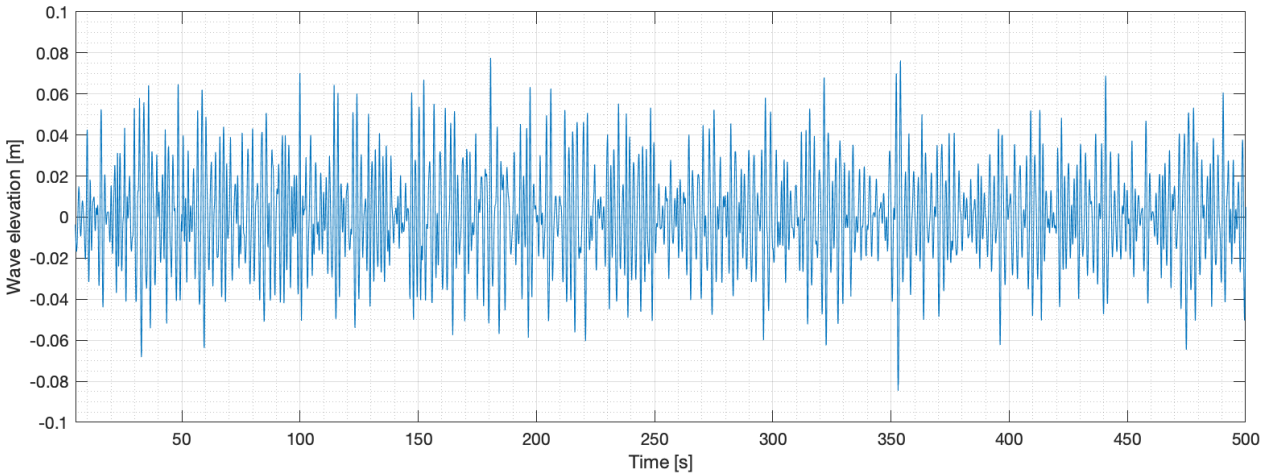


Figure 8.2.1: Time series of free surface elevation at wave probe 2 for irregular computation using IWG for wave generation.

In MATLAB the energy spectrums of the wave elevation time series were calculated. From the spectrums, the zero moment, m_0, was found and used to find the actual significant wave height, H_S, peak period T_P, standard deviation Std, which is the second root of the variance, and total wave energy calculated as the area under the spectrum curve. Peak period was found by $1/(\omega_P/2\pi)$ where ω_P was found by the build-in function *findpeaks(data,x)*, this is not included in the attached script, as it was done separately. m_0 was found by use of the WAFO package in MATLAB. Calculated properties are listed in table 8.2.1, where WGM stands for wave generation method.

Table 8.2.1: Wave statistics for irregular waves calculated from time series captured at wave probes during the simulation.

WGM	**H_S** [m]	**T_P** [s]	**Std** [-]	**E** [(Nm)^2]
IWG	0.0901	1.86	0.0225	0.0033
WG	0.0909	1.86	0.0227	0.0034

Deviations from theory, and applied wave statistics in FINE/Marine, of 10.99% of the significant wave height and 7.53% in peak period were found.

Theoretical maximum spectral density was $1.87e-4$ [(Nm)^2/rad] as read from the plot in figure 8.2.2. Calculated maximum from the simulation was $1.69e-4$ at wave probe 2, which differ from theory with -10.65%.

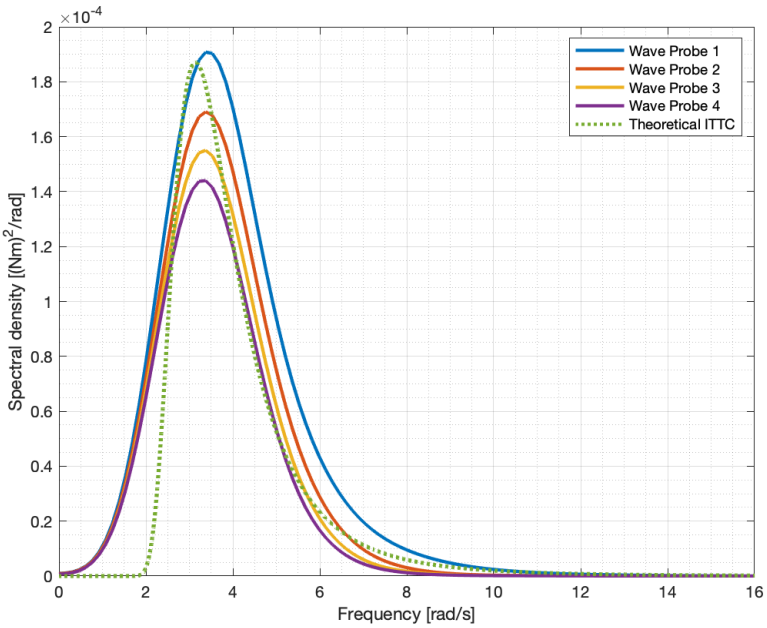


Figure 8.2.2: Energy spectrums for wave probes 1-4 for the irregular analysis using IWG.

Time series of free surface elevation by use of the WG are presented in figure 8.2.3. Here it is possible to see irregular characteristics with multiple peaks, zero crossing periods, etc. Surface elevations from $t = 10$ to 100 [s] are used to exclude the data before the waves reach wave probe 2. In table 8.2.1 calculated wave statistics for the analysis using WG are presented. They are coinciding well to the ones using IWG, implying that the methods give waves of equal quality. The graph has deviations from the theoretical ITTC spectrum in the range $0-2$ [rad/s], this also applies to the spectrum generated using IWG.

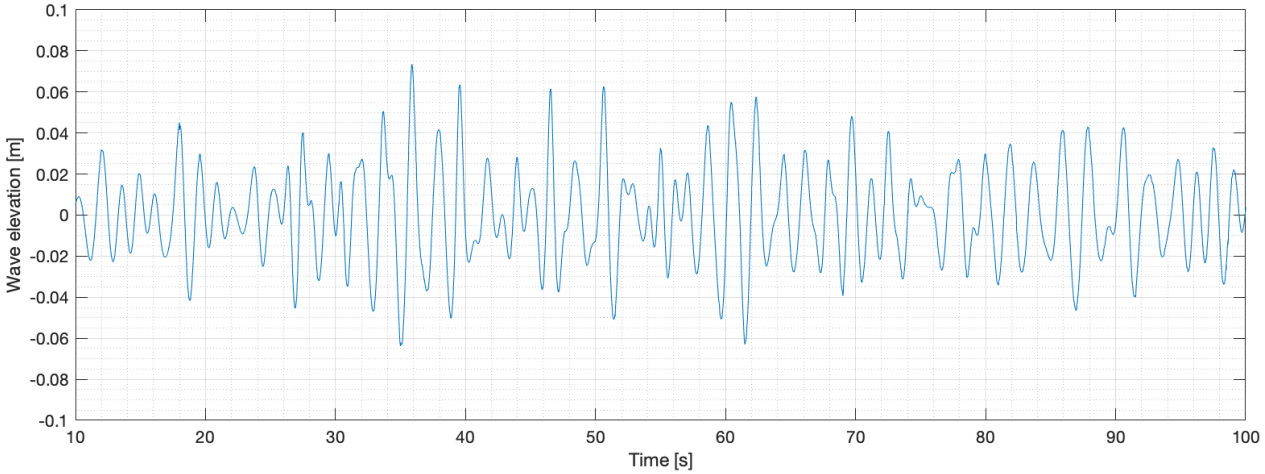


Figure 8.2.3: Time series of free surface elevation at wave probe 2 for irregular computation using WG for wave generation.

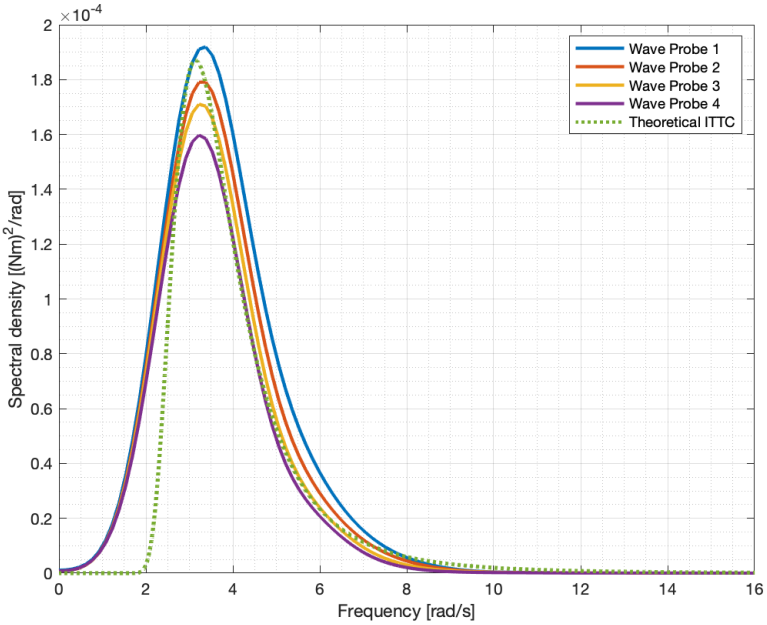


Figure 8.2.4: Energy spectrums for wave probes 1-4 for the irregular analysis using WG.

Energy spectrums for the computation using WG are presented in figure 8.2.4. The total energy decrease downstream the tank. Total energy at wave probe 1 is higher than the theoretical total energy. The low-frequency contributions in the range $0 - 2$ [rad/s] deviates from theory, and possible reasons are discussed in the next section. In figure 8.2.5 the free surface elevation at wave probe 5 for irregular waves generated by use of IWG and WG are plotted. These wave elevations gives low-frequency contributions to the energy spectrums, plotted in figure 8.2.6.

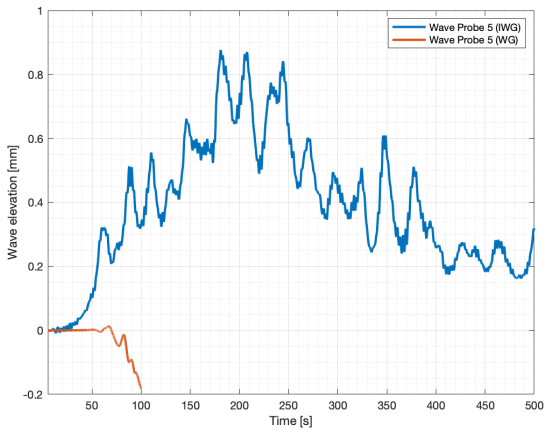


Figure 8.2.5: Time series of free surface elevation at wave probe 5 for the irregular computations using IWG and WG for wave generation.

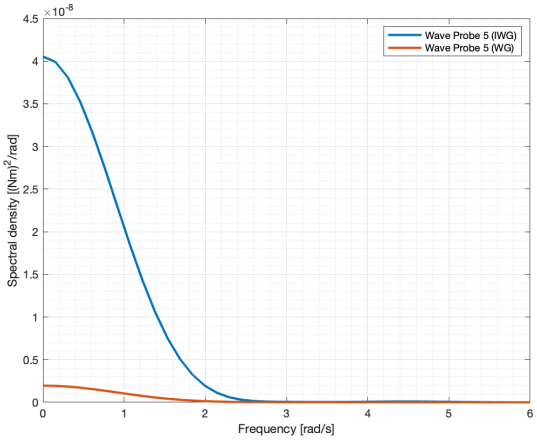


Figure 8.2.6: Energy spectrums for elevations at wave probe 5 using IWG and WG for wave generation.

During these analysis, time consumption of the computations were of essence. One aspect is that during the irregular analysis AGR was enabled, which is more time consuming than without since it re-generate the mesh multiple times during the computation. Additionally, longer overall simulation time was necessary to ensure accuracy. One of these simulations took week(s) using the computational resources available in this thesis. After a computation was done, the results were extracted and post-processed, which were time consuming. Computational power was a big limitation during the whole thesis, as discussed in the following section.

8.3 Discussion

Running an irregular wave analysis, the time step size is of essence. In the present thesis a time step of 0.01 [s] was used, raising the question if this contributed to uncertainty in the results. Zhi-Fu et al. [31] used a time step of $T/40$ and a simulation length of 200 [s] which gave good results. A time step of 0.01 [s] represents $T/200$ in comparison. This argues that the time step is sufficiently small. As previously mentioned, Muniyandy Elangovan [46] argued that a time step of 0.01 [s] was too coarse, and that 0.001 or 0.0001 [s] would be better. Furthermore, a study of the CFL number could give pointers regarding this. To conclude, the time step used in the present thesis arise an uncertainty, but it should be good enough, especially with regards to previous papers using coarser time steps.

Both time series presented are from simulations of relatively short duration. As mentioned in the introduction to this chapter, this is due to limited computational resources. However, it means that most probably not all wave contributions are generated during this time series. That again means that the wave spectrums will differ from theory. Secondly, the loss of energy downstream the wave tank are also discovered using the irregular waves. This is logical as wave spectrums contain of superpositioning many regular waves, and they loose energy as they propagate downstream the NWT.

Significant wave heights, H_S, for both simulations are approximately 10% lower than the target H_S of 0.1 [m]. Compared to Miquel et al. [28] this corresponds quite well for the same simulation length. However, an error of 10% is large if the purpose of the numerical wave tank is to study wave-structure interactions. A peak deviation of 7.53% below the expected, is not a huge error, but again this indicates that a longer simulation time should be applied to get more accurate results. Additionally, a $H_S = 0.1$ [m] is more close to model scale than reality, e.g. H_S is approximately 2.0 [m] at Ecofisk [47].

Low-frequency contributions to the energy spectrum in the range $0-2$ [rad/s] deviates from the theoretical ITTC spectrum. This happens for all irregular wave simulations. A check whether this was because of the time delay before the wave passed the wave probe was done, and it did not change the low-frequency contribution. One can speculate that the low-frequency contributions are present partly because of the energy loss in the waves as they propagate. However, this alone will probably not cause a significant contribution to the wave spectrum. Further work should include an investigation of possible reasons for these contributions, as they are an uncertainty source in this thesis.

By studying the wave energy in the damping zones for both calculations, it is possible to detect contributions in the $e-8$ [(Nm^2)/rad] range. Alone these will not contribute to the magnitude of the low-frequency contributions detected in the spectrums at wave probe 2. In this investigation the time series of the free surface elevation was plotted in figure 8.2.5, where it was seen that the elevation stays below 1% of the significant wave height calculated from the time series.

The irregular simulation using WG crashed after approximately 110 [s]. By observing the wave probe data it is clear that this is due to reflection and possibly numerical diffusion, as this was experienced during regular analysis with steep waves. When running the simulation a sponge layer of 3.5 wavelengths, with $T = 2$ [s] as reference, was used. Irregular waves consist of regular waves with different steepness, length, etc. which can be the main root of this challenge. Therefore, beginning further work with increasing the acceptable steepness would be a good place to start.

Regarding validation against the theoretical ITTC spectrum, the plotted spectrums in figures 8.2.2 and 8.2.4, assume that $\gamma = 1$. This means that the JONSWAP spectrum is equal to the PM spectrum as described in chapter 2, section 2.1.3. Further, it is assumed that $k = 1$, which indicate that the PM

spectrum is equal to the ITTC. Calculated values of k for the two calculations were $k_{IWG} = 1.3805$ and $k_{WG} = 1.3685$. With k-values so close to 1, it is sufficient to assume that they provide the same results. Statistical values of the two computations at wave probe 2, including the relation described in the theory section, are presented in table 8.3.1. Note that the mentioned k here is not the wave number, it is the relation described in section 2.1.3.

Table 8.3.1: Wave statistics for irregular waves using the relation between ITTC and PM.

Calc.	**k** [-]	ω_0 [rad/s]	$\mathbf{T_0}$ [s]	$\mathbf{H_S}$ [m]
IWG	1.3805	3.0319	2.0824	0.0901
WG	1.3685	3.0448	2.0636	0.0909

The game of balancing computational power and accuracy are important when using CFD software. A major limitation of this thesis was both computer and license resources which resulted in a lot of waiting. Therefore, choices have been made on the basis of this, regarding sacrificing accuracy for shorter computational times. This could be one of the important reasons for results. If any master thesis sould be done on this subject at a later time, it is recommended to use NTNU's supercomputer Vilje.

Modeling often involves simplifications of the real world, especially in a master thesis. To account for this, safety factors should be applied when using CFD to look at commercial wave-structure interaction problems. In comparison to experimental validation of models, numerical simulations can be both time- and cost-efficient. However, this require high computational resources and educated personell setting up the simulations. Experience is important running these simulations, as beginner mistakes often are made in the learning phase. Today, as there is high focus on developing better numerical models with more realistic modeling, CFD and NWTs are valueable for minimizing the need to redo model test in experiments. Additionally, the value that these simulations can be done without the need for scaling, increase the benefits of using numerical simulations.

9 | 3D Regular Waves

To further expand this thesis, a premature study of three-dimensional computations were performed. Once expanding into 3D, additional physical aspects can affect the result, e.g. reflections from side walls, turbulence, etc.

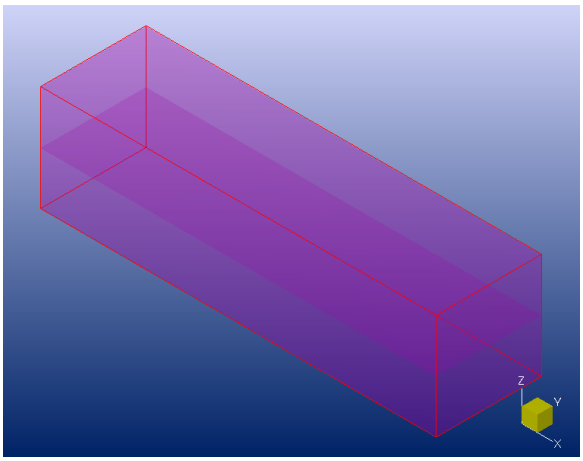


Figure 9.0.1: 3D computational domain with internal surface at z=0 [m].

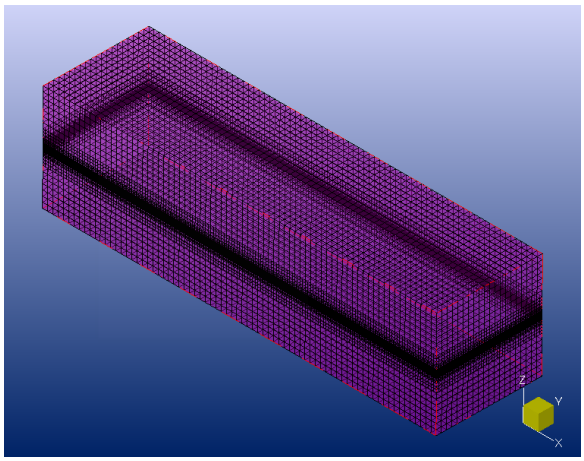


Figure 9.0.2: Mesh concentrated around the free surface.

Methodology to generate the 3D domain, illustrated in figure 9.0.1, mesh and set up the computations parameters were similar to the previously explained for two-dimensional scenarios. One difference was that the 2D grid generation mode was not used. The 3D mesh resulted in a *much* larger grid, and additionally longer computational time.

The wave tank dimensions used were 50x10x14 [m] (*length x width x height*). Damping zone length in x-direction was 3.5 wavelengths, and 1 [m] at each side wall. Figure 9.0.2 show how the mesh is concentrated around the internal surface, at $z = 0$ [m]. Settings used during mesh configuration were: surface refinement with target cell size of $dx = 0.1$, $dy = 0.1$ and $dz = 0.0039$ [m], maximum number of refinements as 10 with an aspect ratio of 100 and global refinement level. Resulting in 3 081 000 cells.

9.1 Results

In this section the results from the three-dimensional analysis with regular waves are presented. Figure 9.1.1 is a screenshot from HEXPRESS of the mass fraction at $t = 80$ [s] for a XZ-plane at $y = 0$. The wave probes used were placed along this plane, and the statistical data from the probes are used to present the results compared to the two-dimensional case.

A part of the time series of free surface elevation at wave probe 1 is plotted in figure 9.1.3. Theoretical and 2D waves are close to equal, where the 3D waves have higher peaks and lower thoughs, as quantified in table 9.1.1. The energy spectrum of both the 2D and 3D waves are plotted in figure 9.1.2. Peak frequency of the 3D computations read $\omega_P = 3.068$ [rad/s], which gives a period of $T_P = 2.048$ [s]. 2D waves contained considerably less total energy than the 3D waves.

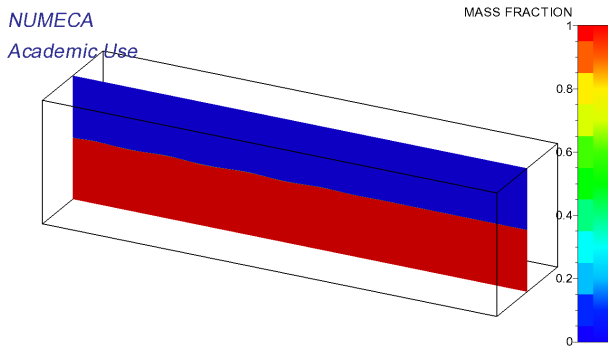


Figure 9.1.1: Mass fraction at time step 8000 for a XZ-plane at y=0.

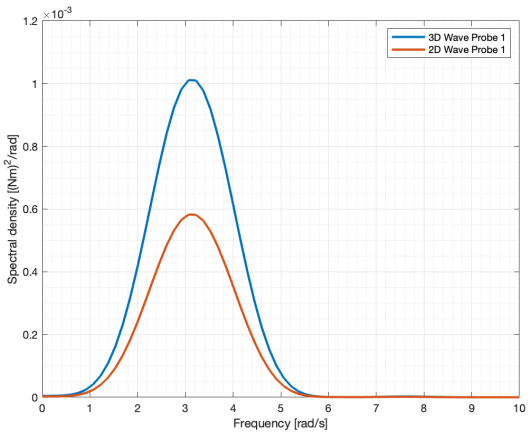


Figure 9.1.2: Energy spectrum for wave elvations captured at wave probe 1, plotted in figure 9.1.3, in 3D computation compared to 2D results.

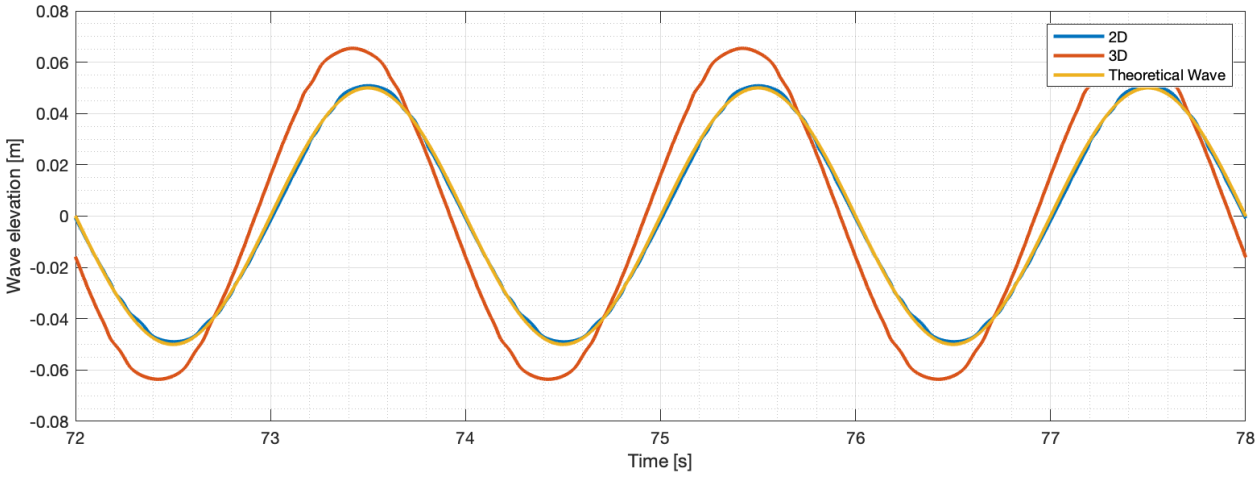


Figure 9.1.3: Time series of free surface elevation at wave probe 1 for regular computation using IWG for wave generation.

Table 9.1.1: Wave statistics for 3D regular waves calculated from time series captured at wave probe 1 during the simulation.

MWL [mm]	**Std** [-]	**Var** [-]	**Range** [m]	**Min** [m]	**Max** [m]
6.37	0.0449	0.0020	0.1432	-0.0652	0.0779

Free surface elevation at the end of each respective damping zone, are presented in figure 9.1.4 with the energy spectrum for the 3D computation plotted in figure 9.1.5. Fluctuations detected in the damping zone stays below 1% of the original wave height. Regardless, they contribute with some energy to the energy spectrum with peak frequency at $\omega_P = 0$ [rad/s]. Compared to the contributions from the original waves, these are of order $e-5$ [(Nm^2)/rad] lower.

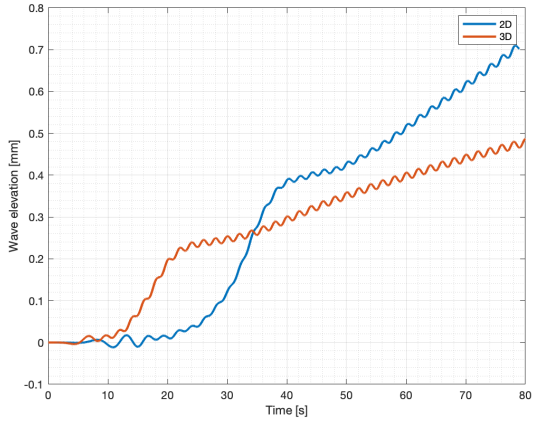


Figure 9.1.4: Time series of free surface elevation at wave probes placed at the end of the damping zone using IWG for wave generation.

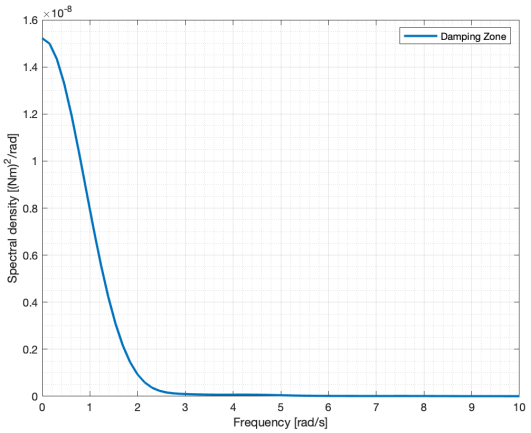


Figure 9.1.5: Energy spectrum for wave elvations in 3D illustrated in figure 9.1.4 to the left.

9.2 Discussion

As presented above, the three-dimensional wave generated with same properties as in 2D, does not give the same results. Several differences were made during the computation of the 3D setup. Firstly, the 3D domain is much shorter than the one in 2D. A study of how the length of the tank effects the 3D computation was not performed in this study, and new sensitivity studies should be performed on the 3D domain in further work. Secondly, the mesh in 3D was coarser than the one in 2D due to the size of the domain, where a coarser grid gave over 3 million cells in comparison the 2D with approximately 95% less cells.

The purpose of performing this 3D analysis was to get an indication whether the same settings could be applied in 3D giving similar results, which it cannot. As seen from the energy spectrums, the 3D waves contained almost double the amount of energy as the 2D. Additionally, the crest and through heights increased. However, the peak period of $T_P = 2.048$ [s] only deviate from theory with 2.4%. Investigations with wave probes placed several places both in y- and x-direction should be performed to see if there is some kind of increased energy in the middle of the tank. Generally, this emphasises that 3D simulations differs from 2D, and to understand the 3D phenomena, a thourogh 2D analysis should be performed before moving on to 3D.

Further work should include damping zone investigations, overall dimensions, mesh refinement and time step refinement for starters. Then, irregular waves should be included, and after they are validated structures can be included in the domain.

Generally, three-dimensional analysis are much more time consuming than two-dimensional, mainly because of a massive increase of cells in the computational domain. Therefore, more computational power than 32 cores should be available for these calculations. Not all CFD codes takes advantage of the number of cores, but this is a future development case within CFD codes [18].

10 Conclusions and Further Work

10.1 Conclusions

High cost and long time consumptions related to experimental testing of marine structures have led to the creation of Numerical Wave Tanks (NWTs). Lately, Computational Fluid Dynamics (CFD) have been applied to generate these NWTs in commercial optimization of maritime structures. Work done in this master thesis differs from other by focusing on creating guidelines for wave generation using two different wave generation methods in FINE/Marine. Dependency studies in 2D and thorough validation of both regular and irregular waves were performed. Additionally, a premature study of 3D regular waves was performed.

The main focus of this thesis was to build on the fundament of the project thesis, with the same theme, further exploring two-dimensional wave generation in FINE/Marine. This involved exploring new literature, experimenting with longer simulations, irregular waves and generating some general guidelines.

The analysis showed that FINE/Marine generate regular sinusoidal waves with high accuracy using both wave generation methods with the proposed guidelines, required a steepness below $1/33$. However, the guidelines showed some limitations using WG, where the wave damping was insufficient, resulting in reflections and numerical diffusion. This was connected with different damping behaviour using the two different wave generation methods, showing that the sponge layer works remarkably better using IWG. Loss of wave energy as the waves propagated down the NWT was experienced during all simulations. Lastly, using the proposed guidelines and theory on 3D waves, the simulated waves did not have the same wave properties as in 2D.

Further work should involve extensive studies of application of steep waves and damping zone efficiency. Then an investigation of the low-frequency contributions to the irregular spectrums should be performed. Following, dependency studies should be performed in 3D before eventually introducing irregular waves and a structure to study interaction effects.

10.2 Further Work

Several other aspects of the present study of numerical wave tanks in CFD are suggested for future work in this chapter. The first suggestion is to complete an extensive study of application of waves with a steepness higher than $1/33$. This is closely connected to a study of damping zone efficiency, especially using the WG at the boundary, e.g. by expanding the sponge layer to at least $5-6$ wavelenghts.

Additional studies of the pressure distribution around the waves could be interesting to see whether the physical aspects of experiments apply to the numerical simulations. This could be done by use of surface probes measuring the pressure.

As physical wave tanks often are longer than the ones used in the present thesis, expanding the dimensions to be more realistic could be interesting. Especially looking at how the wave energy decrease downstream the tank, and see if some numerical schemes could help on the energy dissipation. This could possibly be done in connection to the abovementioned, with regards to scaling effects.

Further, studying the the energy loss down the tank should be done. With a focus on where the additional energy in the low-frequency range of $0-2$ [rad/s] comes from. A possible way to begin this is to look at the damping zone. If any of the waves are reflecte back this can give low-frequency contributions. This is one way of validating where the contributions come from. Additionally, use of volume probes measuring the mass fraction can be interesting with regards to numerical diffusion, which is nearly impossible to see from the wave probe data alone.

Regarding post-processing, several measurements can be taken to make it more efficient. Generating a Python script which can both run a FINE/Marine analysis, extract the data once it is finished and later post-process it directly in Python would save time and be more efficient. One uncertainty in the present code is that it looks at the range and maximum values of surface elevation. Further work should therefore include looking at the mean wave height or similar.

As Miquel et al. [28] showed that 90% of the theoretical spectrum components were generated within the first 500 [s], a similar study could be performed for this scenario, as results now indicate the same conclusion applies. However, further analysis are required to draw this conclusion. Having quantified this number, makes it easier to apply right safety factors in simulations, and still saving time using shorter simulations than by example 3 [hours].

After finishing the two-dimensional analysis, further work should include a series of dependency studies performed in 3D, using regular waves, considering effects of variying mesh resolution, time step size, overall dimensions and damping zone lengths. Later, irregular waves should be applied and compared to wave statistics from realistic wave energy spectrums. The same analysis should be performed in model scale to look at scaling effects in the analysis. Eventually, an offshore structure should be introduced to study initial wave-impact loads, structure motions, etc.

Bibliography

[1] Fine/marine. https://www.numeca.com/product/finemarine. Accessed: 2018-12-03.

[2] Odd M. Faltinsen. *Sea Loads on Ships and Offshore Structures*. Cambridgre University Press, 1990.

[3] DNV GL. Modelling and analysis of marine operations. *Recommended Practice DNV-RP-H103*, 2011.

[4] Stanislaw R. Massel. *Ocean Waves Breaking and Marine Aerosol Fluxes*. Springer, Poland, 2007.

[5] NUMECA International. User manual fine/marine 7.2.

[6] Centrale Nantes. Theory guide fine/marine.

[7] Xiaojie Tian et al. Numerical and experimental studies on a three-dimensional numerical wave tank. *IEEE Journals & Magazines Access 2018 Volume 6*, 2016.

[8] Milovan Peric and Volker Bertram. Trends in industry applications of cfd for maritime flows. *10th International Conference on Computer and IT Applications in the Maritime Industries*, 2011.

[9] ITTC. The specialist committee on waves final report and recommendations to the 23rd ittc. *Proceedings of the 23rd ITTC*, 2002.

[10] Timothy E. Kendon Csaba Pakozdi and Carl-Trygve Stansberg. Breaking wave impact on a platform column: An introductory cfd study. *ASME 2011 30th International Conference on Ocean, Offshore and Arctic Engineering*, 2011.

[11] Ankit Aggarwal et al. Numerical study of irregular breaking wave forces on a monopile for offshore wind turbines. *Energy Procedia 14th Deep Sea Offshore Wind RD Conference*, 2017.

[12] Erik Lehn. *Litt om bølger*. MARINTEK, 2008.

[13] Bjørnar Pettersen. *TMR4247 Marin Teknikk 3 Hydrodynamikk*. Akademika Forlag, Norway, 2007.

[14] Øyvind A. Arntsen and Harald E. Krogstad. *Linear Wave Theory*. Norwegian University of Science and Technology, 2000.

[15] Dazheng Wang Limin Chen, Guanghua He and Jinsheng Zhang. Computation of wave-making resistance on high speed catamaran using fine/marine. *The Proceedings of The Twenty-fifth International Ocean and Polar Engineering Conference*, 2015.

[16] Dag Myrhaug. *TMR4230 Irregular Waves*. Akademika Forlag, 2006.

[17] Subrata Kumar Chakrabarti. *Hydodynamics of Offshore Strucutres*. WIT Press, 1987.

[18] Guan H. Yeoh and Anthony Yuen. *Lectures in MECH4620 Computational Fluid Dynamics*. 2018.

[19] Cfd online. `https://www.cfd-online.com`. Accessed: 2018-11-17.

[20] C. W. Hirt and B. D. Nicols. Volume of fluids (vof) method for the dynamic of free boundaries. *Journal of Computational Physics*, 1979.

[21] NUMECA International. User guide hexpress 7.2.

[22] Said Mazaheri et al. Wave generation in a numerical wave tank. *International Journal of Coastal Offshore Engineering*, 2015.

[23] Qingjie Du and Dennis Leung. 2d numerical simulation of ocean waves. *World Renewable Energy Congress 2011*, 2011.

[24] Chih-Ming Dong and Ching-Jer Huang. Generation and propagation of water waves in a two-dimensional numerical viscous wave flume. *Journal of Waterway, Port, Coastal and Ocean Engineering*, 2004.

[25] J.C. Park Moo-Hyun Kim and S.Y. Hong. Fully nonlinear multi-directional waves by a 3d viscous numerical wave tank. *The Ninth International Offshore and Polar Engineering Conference*, 1999.

[26] Weoncheol Koo Min Woo Kim and Sa Young Hong. Numerical analysis of various artificial damping schemes in a three-dimensional numerical wave tank. *Ocean Engineering 75*, 2013.

[27] Hans Bihs et al. A new level set numerical wave tank with improved density interpolation for complex wave hydrodynamics. *Computers and Fluids 140*, 2016.

[28] Adria Moreno Miquel et al. Analysis of different methods for wave generation and absorption in a cfd-based numerical wave tank. *Journal of Marine Science and Engineering 6*, 2018.

[29] Guillaume Ducrozet et al. A modified high-order spectral method for wavemaker modeling in a numerical wave tank. *European Journal of Mechanics B/Fluids 34*, 2012.

[30] Hu Zhe et al. Numerical wave tank based on a conserved wave-absorbing method. *China Ocean Engineering Volume 30*, 2014.

[31] Li Zhi-Fu et al. Simulation of irregular wave in a numerical wave tank. *Polish Maritime Research Special Issue 2015 S1 Volume 22*, 2015.

[32] M. J. Ketabdari H. Saghi and S. Booshi. Generation of linear and nonlinear waves in numerical wave tank using clustering technique-volume of fluid method. *Applied Mathematics and Mechanics English Edition 33*, 2012.

[33] Solomon C. Yim Seshu B. Nimmala and Stephan T. Grilli. An efficient three-dimensional fnpf numerical wave tank for large-scale wave basin experiment simulation. *Journal of Offshore Mechanics and Artic Engineering Volume 135*, 2013.

[34] Aldric Baquet and Jang Kim. Numerical modeling using cfd and potential wave theory for three-hour nonlinear irregular wave simulations. *Proceedings of the ASME 2017 36th International Conference on Ocean, Offshore and Arctic Engineering*, 2017.

[35] Star-ccm+. https://mdx.plm.automation.siemens.com/star-ccm-plus. Accessed: 2018-11-01.

[36] Drobyshevski Yuriy Thomas Giles Abdussamie Nagi, Ojeda Roberto and Amin Walid. The impact of extreme wave events on a fixed multicolumn offshore platform. *International Journal of Offshore and Polar Engineering*, 2017.

[37] Hans Bihs Ankit Aggarwal, Mayilvahanan Alagan Chella and Øivind Asgeir Arntsen. Irregular wave forces on a large vertical circular cylinder. *Energy Procedia 13th Deep Sea Offshore Wind RD Conference*, 2017.

[38] Lei Gao Cheng-sheng Wu, De-cai Zhou and Quan ming Miao. Cfd computation of ship motions and added resistance for a high speed trimaran in regular head waves. *International Journal of Naval Architecture and Ocean Engineering*, 2011.

[39] Zhirong Shen Haixuan Ye and Decheng Wan. Numerical prediction of added resistance and vertical ship motions in regular head waves. *Journal of Marine Science and Application*, 2012.

[40] Sakir Bal. Performance prediction of surface-piercing bodies in numerical towing tank. *International Journal Of Offshore And Polar Engineering*, 2008.

[41] Hideo Orihara Hideaki Miyata and Yohei Sato. Nonlinear ship waves and computational fluid dynamics. *The Japan Academy Series B-Physical And Biological Sciences*, 2014.

[42] Dazheng Wang Limin Chen, Guanghua He and Jinsheng Zhang. Computation of wave-making resistance on high speed catamaran using fine/marine. *The Proceedings of The Twenty-fifth International Ocean and Polar Engineering Conference*, 2015.

[43] A. Clément C. Kim and K. Tanizawa. Recent research and development of numerical wave tanks - a review. *Offshore Polar Engineering 9*, 1999.

[44] Weather statistics for stavanger. https://www.yr.no/place/Norway/Rogaland/Stavanger/Stavanger/statistics.html. Accessed: 2018-11-01.

[45] Stavanger average sea temperature. https://www.seatemperature.org/europe/norway/stavanger-january.htm. Accessed: 2018-11-01.

[46] Muniyandy Elangovan. Simulation of irregular waves by cfd. *International Journal of Mechanical and Mechatronics Engineering Vol. 5*, 2011.

[47] Magnar Reistad et al. A high-resolution hindcast of wind and waves for the north sea, the norwegian sea and the barents sea. *Norwegian Meteorological Institute met.no*, 2009.

A Results

A.1 Mass Fraction 0.5 of 2D Regular Wave Computations

This section plots the isolines of mass fraction 0.5, illustrating the free surface captured at time step 8 000 or $t = 80$ [s].

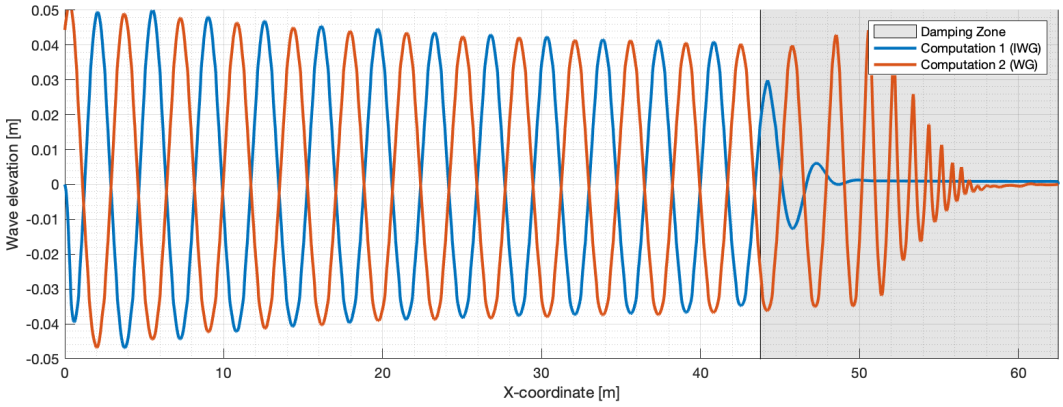


Figure A.1.1: Free surface downstream tank for computations 1 and 2.

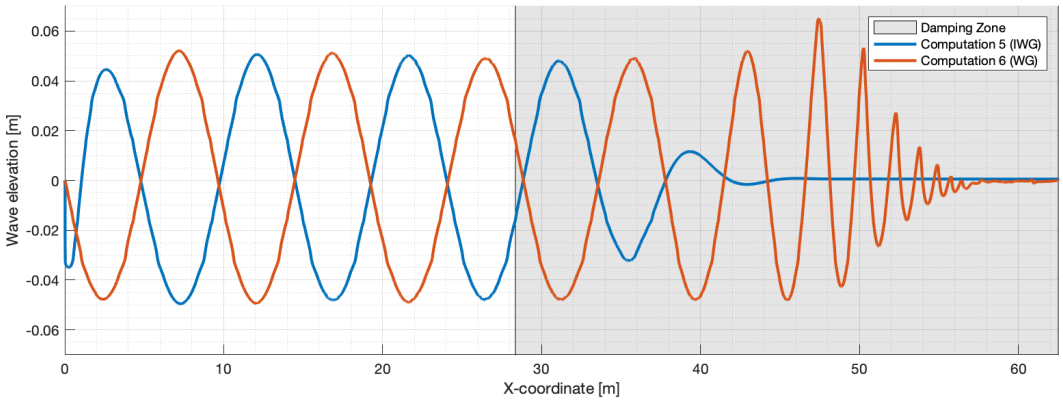


Figure A.1.2: Free surface downstream tank for computations 5 and 6.

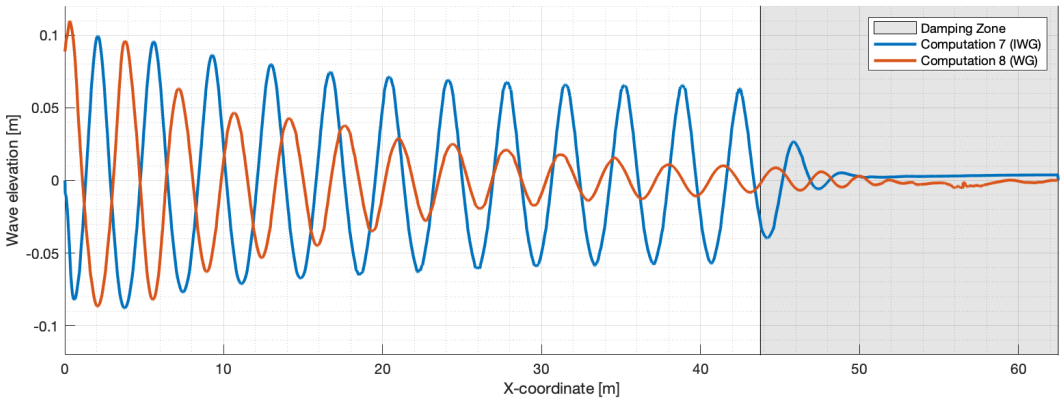


Figure A.1.3: Free surface downstream tank for computations 7 and 8.

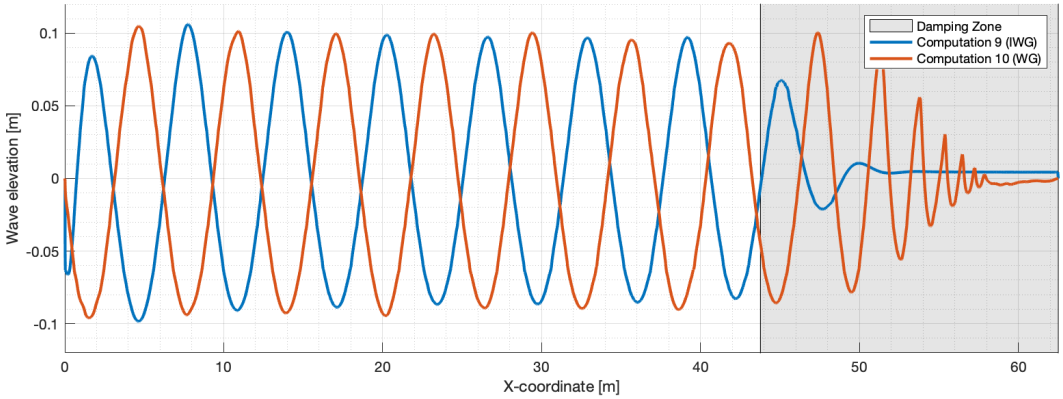


Figure A.1.4: Free surface downstream tank for computations 9 and 10.

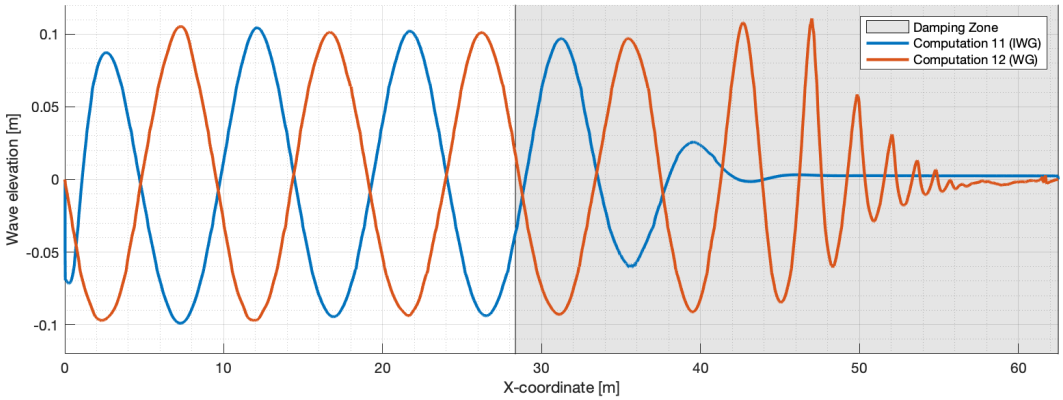


Figure A.1.5: Free surface downstream tank for computations 11 and 12.

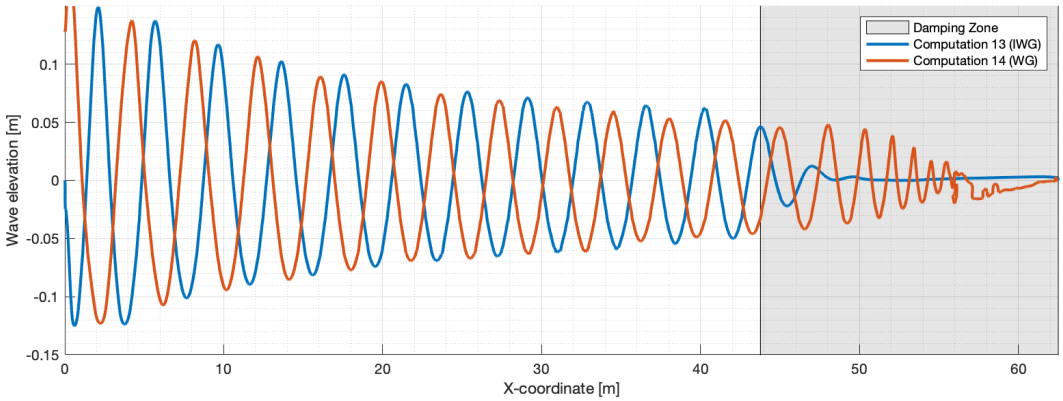


Figure A.1.6: Free surface downstream tank for computations 13 and 14.

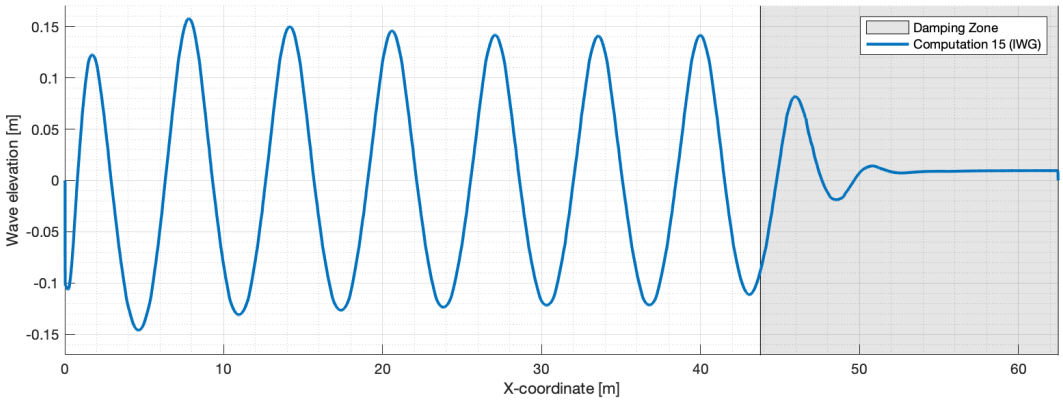


Figure A.1.7: Free surface downstream tank for computation 15.

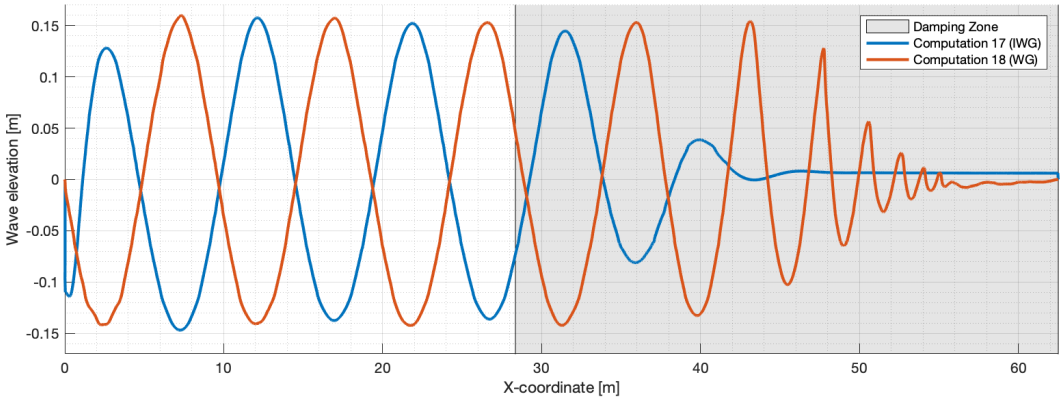
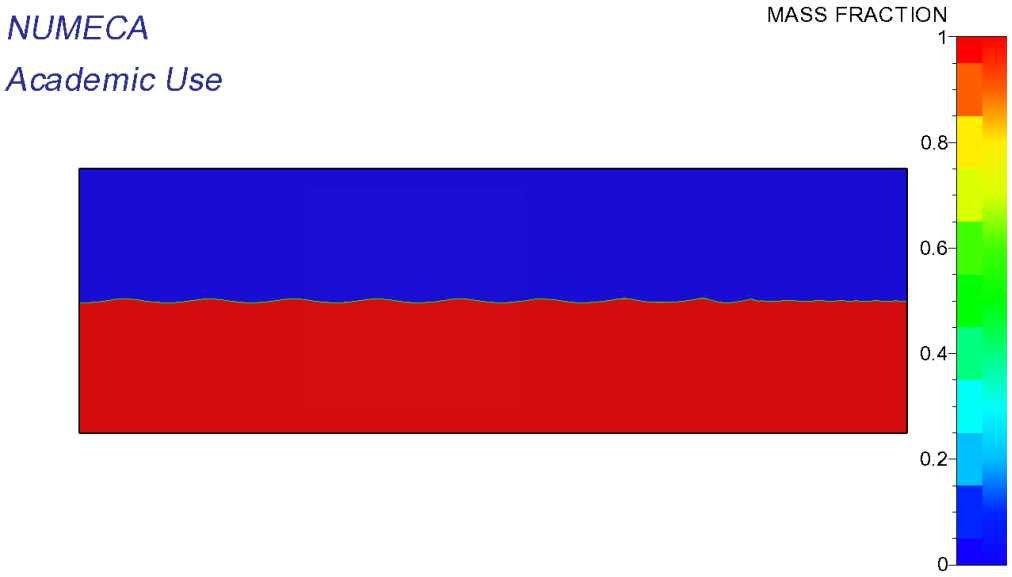
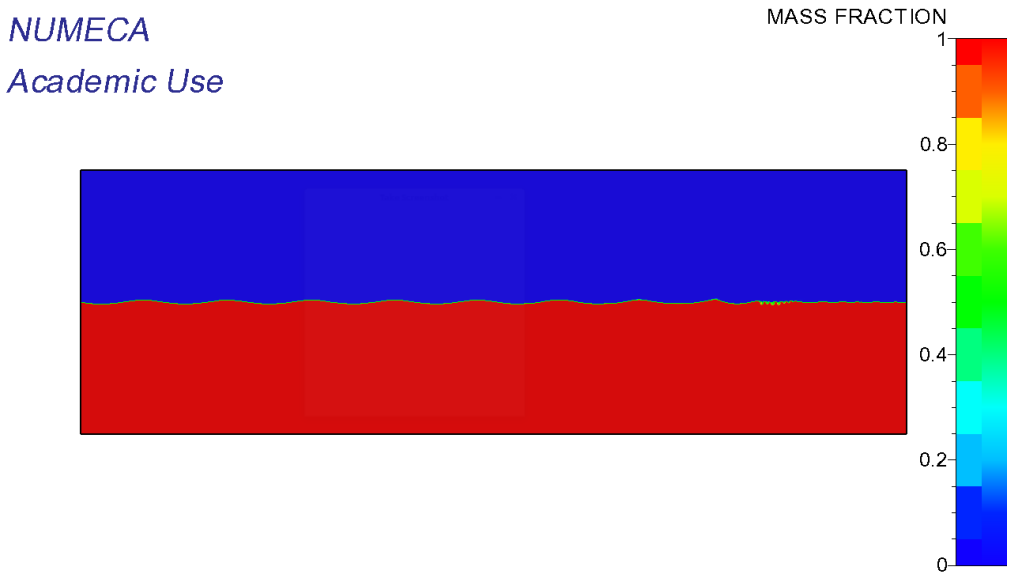


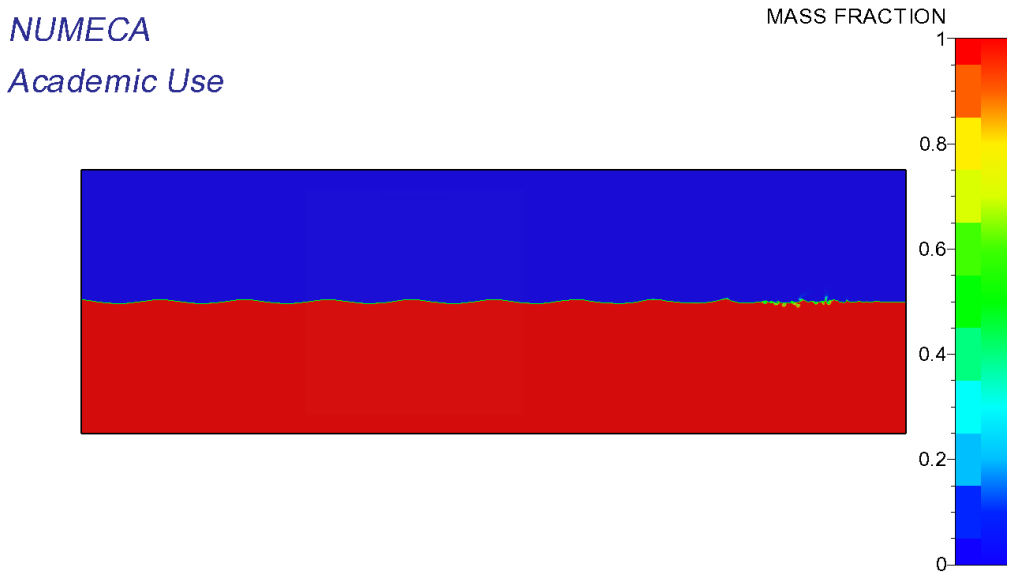
Figure A.1.8: Free surface downstream tank for computations 17 and 18.

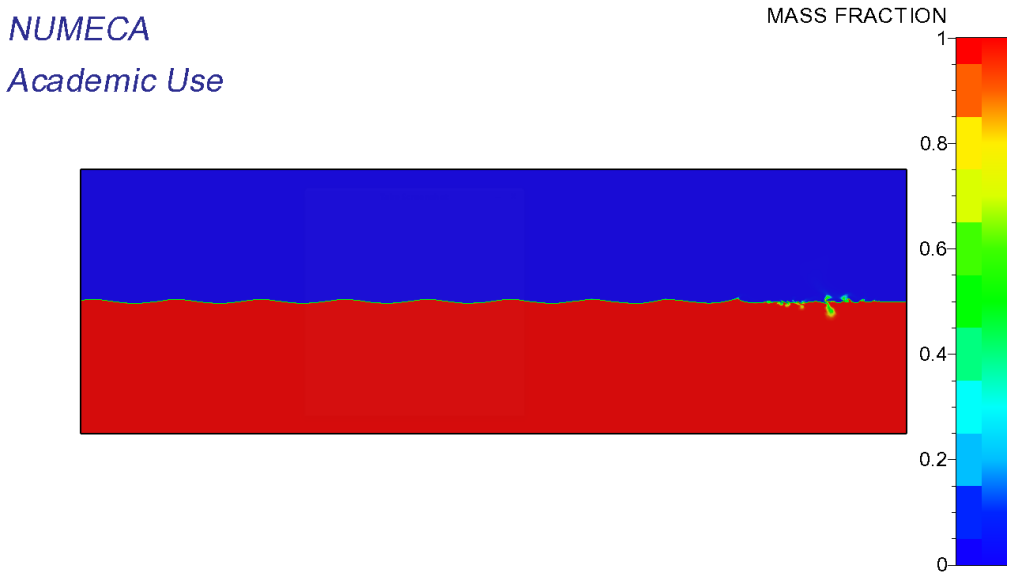
A.2 2D Regular Waves, Computation 16

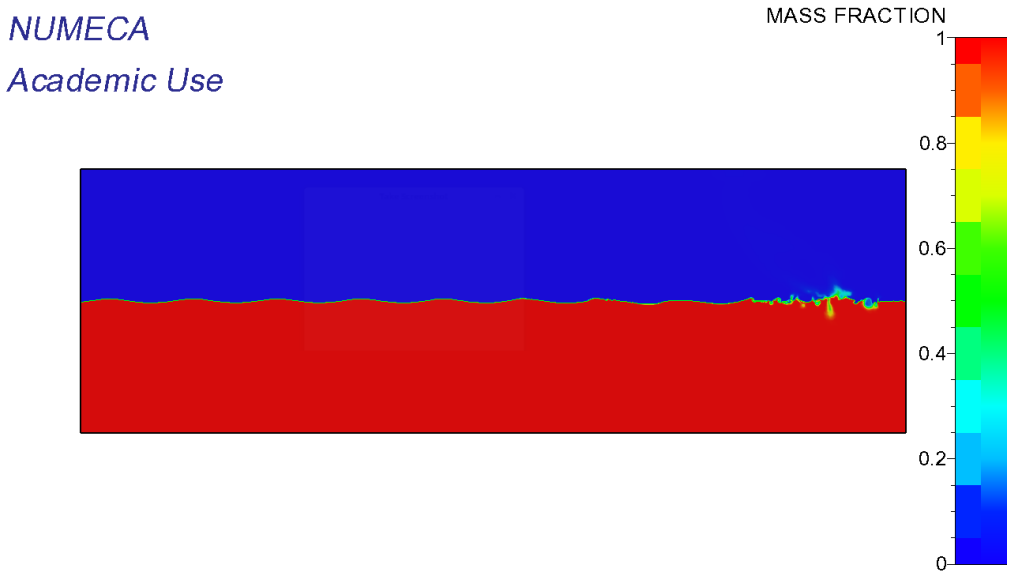
The following seven figures illustrate the mass fraction of computation 16 from $t = 35.6$ to 38.0 [s] with a time step value of 0.4 [s].

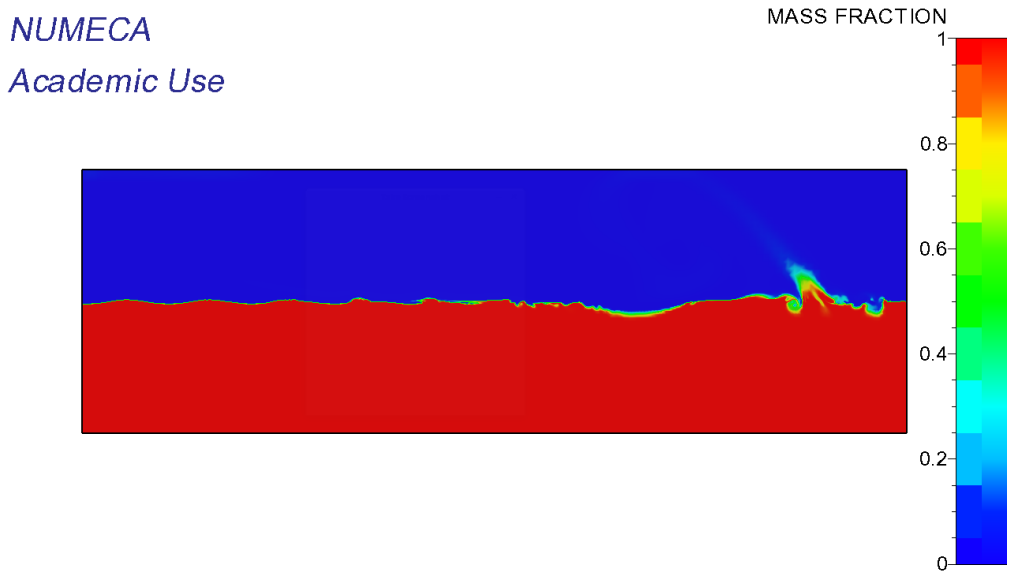


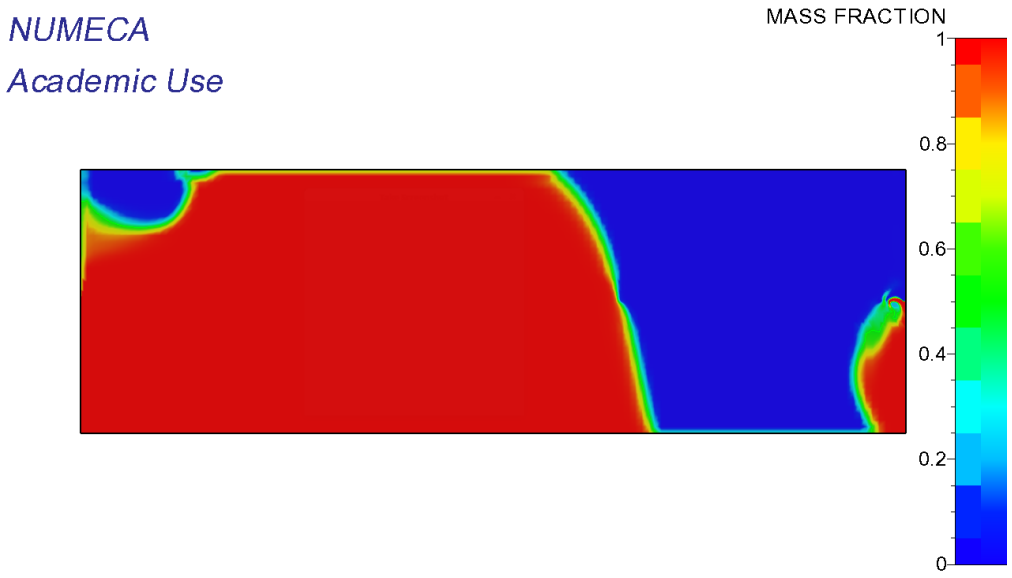



NUMECA
Academic Use
MASS FRACTION
1
0.8
0.6
0.4
0.2
0


NUMECA
Academic Use
MASS FRACTION
1
0.8
0.6
0.4
0.2
0


NUMECA
Academic Use
MASS FRACTION
1
0.8
0.6
0.4
0.2
0


NUMECA
Academic Use
MASS FRACTION
1
0.8
0.6
0.4
0.2
0


NUMECA
Academic Use
MASS FRACTION
1
0.8
0.6
0.4
0.2
0

B | Master Poster

As a part of the master thesis a poster for an exhibition was made and attatched on the following page. The main objective of the exhibition was to present to fellow students, faculty, SINTEF Ocean and others visiting MTS the topics researched by students graduating each year.

The poster were evaluated by the following criterias:

- Scientific level
- Description of problem/topic researched
- Completedness and clarity of presentation with regard to description of:
  - Motivation and problem definition
  - How the problem was solved
  - Who has contributed and what is your contribution
  - Results
  - Conclusions
  - References
- Layout related to the balance between figures, text, animations, etc.

Instructions on content were as following in this paragraph. "The poster should include master thesis title, your name, the name of the primary and secondary advisors and poster sections: objective and scope, introduction, methods or modeling, simulations/experiments, results and conclusions."

A CFD Study of Numerical Wave Tanks

Mari T. Storsul

Supervisor: Kourosh Koushan

Secondary supervisor: Eloïse Croonenborghs

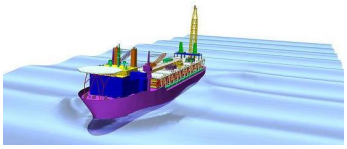
NTNU – Trondheim
Norwegian University of Science and Technology

SINTEF

Introduction

Ships, cage aquaculture, offshore wind turbines and offshore platforms are examples of structures that operate in oceanic environments, constantly under influence of complex forces from wind, current and waves. In particular, waves are important during a design phase, as it causes motions and loads. Experiments in wave tanks and flumes are among the most commonly used methods for wave research today. As this is time consuming and expensive, this has led to the creation of Numerical Wave Tanks (NWTs). Lately, Computational Fluid Dynamics (CFD) has been applied to generate NWTs with a various of different numerical modeling techniques [1].

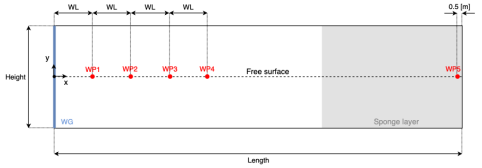
Illustrative photo from [2].



Stability and accuracy are essential to the performance of NWTs, and accurate wave generation are among the existing technical difficulties. Therefore, it is necessary to explore appropriate wave generation methods, that can reproduce actual marine conditions. The aim of the present work are, therefore (1) Performing a series of dependency studies considering the effect of varying setup parameters of importance in full scale. (2) Develop guidelines that will allow wave-capturing presented with alternative approaches for wave generation. (3) Perform a validation of the simulated waves against numerical results to analytical solutions.

Modeling and Software

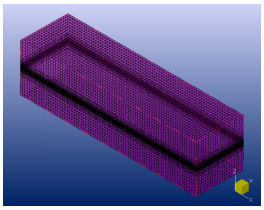
This thesis mainly performed 2D simulations, and additionally a preliminary 3D investigation. Two methods of generating waves were studied: (1) Internal Wave Generator (IWG) and (2) Wave Generator (WG). The applied CFD software, FINE/Marine, is based on Reynolds Averaged Navier-Stokes Equations, and it applies the Volume of Fluid method to model the free surface.



Schematics of the NWT are presented in the above figure for the case using IWG. Wave Probes (WP) were used to capture the free surface elevation in time. Boundary conditions applied were updated hydrostatic pressure for the lower and upper boundary, external far field condition for tank end and either far field or WG at the inlet. Sponge layers were used for wave damping.

Simulations of regular sinusoidal waves were simulated in 80 [s], and irregular simulations in 500 [s]. All computations used a relatively coarse time step size of 0.01 [s].

Mesh generation was performed using an initial mesh, and applying additional refinement around an internal surface representing the free surface. For the irregular analysis, Adaptive Grid Refinement (AGR) procedures were used. The figure to the left illustrates how the mesh was concentrated around the free surface.

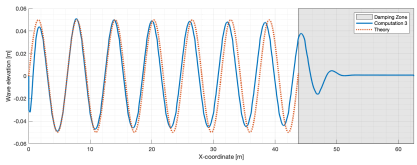


Abbreviations

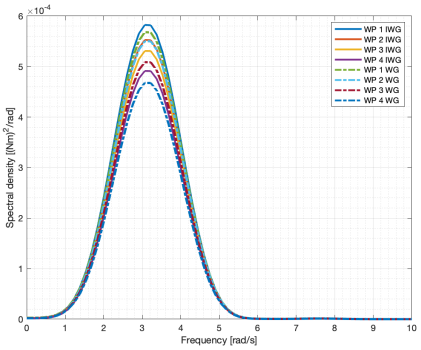
AGR	Adaptive Grid Refinement
CFD	Computational Fluid Dynamics
IWG	Internal Wave Generator
NWT	Numerical Wave Tank
WG	Wave Generator
WP	Wave Probe

Results

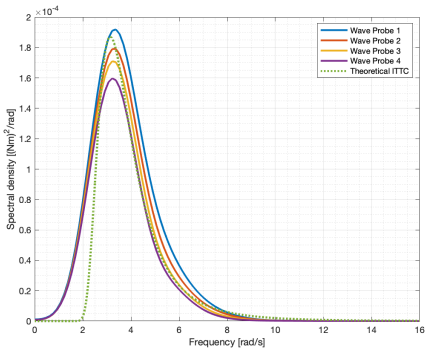
In this poster a small part of the results are presented. The first two figures are from a regular analysis comparing the two wave generation methods. In the figure below and to the left, the isoline representing the free surface downstream the NWT are presented compared to theory. Theoretical height is 0.1 [m], and the simulation wave height error stays below 1.1%.

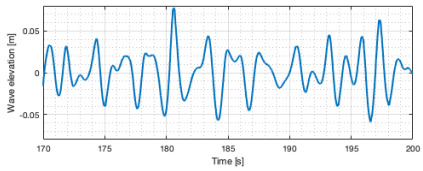


In the figure to the right, wave spectrums of computations using IWG (whole line) and WG (dotted line). Total wave energy decrease downstream the tank, and are higher by use of IWG than WG. Peak frequency of the simulated waves deviates with -2% of the intended theoretical frequency.



The next two figures are from an irregular analysis using the WG. To the right a figure of the free surface elevation at WP 2 is presented. Energy spectrums from each WP are plotted in the figure to the left, showing relatively good results compared to theory. Some low-frequency waves give a deviation from theory from $0-2$ [rad/s]. These irregular analysis also show that some energy losses down the tank are experienced.





Conclusions

The analysis showed that FINE/Marine generate regular sinusoidal waves with high accuracy using both wave generation methods with the proposed guidelines. However, the guidelines showed some limitations with steep waves using WG, where the damping zone was insufficient. Additionally, different damping behaviour for the two wave generation methods were detected. Finally, loss of wave energy downstream the NWT were experienced during all simulations.

Further work should involve an extensive damping zone efficiency study in 2D. Then an investigation of the low-frequency contributions to the irregular spectrums should be performed. Following, the domain should be extended into 3D and compared to the validated 2D waves.

Acknowledgements

This thesis would not have been possible without the support of my supervisor Kourosh Koushan and secondary supervisor Eloïse Croonenborghs. My sincere thanks to Fengjian Jiang for contact with NUMECA's support services.

References

[1] Xiaojie Tian et al. Numerical and experimental studies on a three-dimensional numerical wave tank. *IEEE Journals & Magazines Access 2018*, 6, 2018.

[2] Stephen Ferguson. Numerical towing tanks, a practical reality? *The Maritime Executive*, 2015.

C | Python Script

```
#   Pre-processing in FINE/Marine
#
#   This scripts does all pre-processing in FINE/Marine,
#   and is opened inside FM. It handles CAD modeling,
#   BC definition, wave generation, damping zone setup,
#   solver setup with numerical schemes.
#
#   This script was made for use of two-dimensional analysis
#   using regular waves.
#
#   Last edited: 27.05.19
#   Written by: Mari T. Storsul

script_version(2.2)

#   Predefining variables

height = 10
length = 62.5
name = "name"
path = "/home/finemarine/Documents/MariStorsul/name"
dom_path = "/home/finemarine/Documents/MariStorsul/name/_mesh/name
    .dom"
save_path = "/home/finemarine/Documents/MariStorsul/name/_mesh/
    name.igg"
int_surf_name = "name_internal_surface_6"
init_mesh_dx = length*2
init_mesh_dy = height*4
is_target_dx = 0.0625
is_target_dy = 0.0039
wave_length = 6.245240
wave_height = 0.1
wave_period = 2
source_loc = wave_length/2
wp5 = length - 0.5
sponge_layer_placement = length - wave_length*3.5

#   Creating project and open HEXPRESS
```

```
37
38 FM.create_project("/home/finemarine/Documents/MariStorsul/waves/
       wave1")
39 FM.switch_to_HEXPRESS()
40 HXP.close_project()
41 HXP.close_project()
42
43 #    Creating CAD model
44
45 HXP.create_cube("B1",Point(0,-height,0),Point(length,height,0.001)
       )
46 HXP.delete_stl_triangulation ()
47 HXP.delete_stl_triangulation ()
48 HXP.delete_stl_triangulation ()
49 HXP.create_domain(dom_path,["B1"],-1,-1,-1,2,-1,2)
50 HXP.delete_stl_triangulation ()
51 HXP.delete_bodies(HXP.get_all_bodies())
52 HXP.import_domain(dom_path)
53 HXP.set_mesh_generation_mode("3D")
54
55 #    Predefine boundary conditions
56
57 HXP.domain("wave1").get_face(0).set_name("front_face")
58 HXP.domain("wave1").get_face(0).set_type("MIR",0)
59 HXP.domain("wave1").get_face(1).set_name("y_min")
60 HXP.domain("wave1").get_face(1).set_type("EXT",0)
61 HXP.domain("wave1").get_face(2).set_name("x_min")
62 HXP.domain("wave1").get_face(2).set_type("EXT",0)
63 HXP.domain("wave1").get_face(3).set_name("y_max")
64 HXP.domain("wave1").get_face(3).set_type("EXT",0)
65 HXP.domain("wave1").get_face(4).set_name("back_face")
66 HXP.domain("wave1").get_face(4).set_type("MIR",0)
67 HXP.domain("wave1").get_face(5).set_name("x_max")
68 HXP.domain("wave1").get_face(5).set_type("EXT",0)
69 HXP.set_mesh_generation_mode("2D")
70
71
72 #    Create internal surface
73
74 run_script("/usr/numeca/finemarine72/_python/_hexpress_plugins/
       Marine/Internal surface creation.py")
75 curve_13 = new_polyline("curve_13")
76 curve_13.insert_point(1,Point(0,0,0))
77 curve_13.insert_point(2,Point(0,0,0.001))
78 curve_14 = new_cspline("curve_14")
79 curve_14.insert_point(1,Point(length,0,0))
80 curve_14.insert_point(2,Point(length,0,0.001))
81 HXP.lofted_surface([curve_13,curve_14],int_surf_name)
82
83 #    Meshing
```

```
84
85 HXP.init_cartesian_mesh(init_mesh_dx,init_mesh_dy,1)
86 HXP.set_global_number_of_refinements(10)
87 HXP.domain_face(7).enable_adaptation(True)
88 HXP.domain_face(7).set_number_of_refinements(10)
89 HXP.domain_face(7).set_max_aspect_ratio(12.5)
90 HXP.domain_face(7).set_adaptation_criteria(0,0,1)
91 HXP.domain_face(7).set_target_sizes(is_target_dx,is_target_dy,0)
92 HXP.generate_initial_mesh()
93 HXP.adapt_mesh()
94 HXP.snap_mesh()
95 HXP.regularize_mesh()
96 HXP.set_optimization_params(0,4,100,7,3,0,10)
97 HXP.insert_viscous_layers()
98
99 #   Save project and mesh, and return to FM
100
101 HXP.save_project(save_path)
102 HXP.save_project()
103 FM.link_mesh_file(save_path)
104 FM.switch_HEXPRESS_to_FM()
105
106 #   Simulation setup
107
108 FM.set_time_configuration(FM.UNSTEADY)
109 FM.set_fluid_viscosity(1,0.001397)
110 FM.set_fluid_density(1,1027)
111 FM.set_fluid_viscosity(2,1.77E-05)
112 FM.set_fluid_density(2,1.25)
113
114 #   BC definition
115
116 FM.set_bc_value(1,1,2,27)
117 FM.set_bc_value(1,1,4,27)
118 jet = FM.get_jet_by_name('Jet')
119 FM.update_jets()
120
121 #   Wave generation using WG
122 #   Apply the folloing before jet = FM.get_jet_by_name('Jet'):
123 #   FM.set_bc_value(1,1,3,45)
124 #   FM.set_WaVe_params("REGULAR",1,10,1,0,0,0,0.1,1.5,3.3)
125
126 #   Wave generation using IWG
127
128 FM.set_internal_wave_usage(1)
129 parameters = FM.get_internal_wave_parameters()
130 parameters.set_depth(height)
131 parameters.set_period(wave_period)
132 parameters.set_height(wave_height)
133 parameters.set_source_x(source_loc)
```

```
parameters.set_direction(FM.POSITIVE_X)
FM.set_internal_wave_parameters(parameters)

#    Wave damping

FM.set_wave_damping_usage(1)
FM.set_sponge_layer(0,sponge_layer_placement,-height,height)

#    Computational parameters

FM.set_urex_velocity(0.5,0.5,0)
FM.set_traveling_shot_parameters(0,0,0,0,0,0,0)
FM.set_control_variables(20,2,8000,4000)

#    Volume and wave probes

FM.set_volume_probe(101,1)
FM.set_probe_interval(40,"timesteps")
FM.set_point_probe(1, "WAVE PROBE", 0, wave_length, 0, 0)
FM.set_point_probe(2, "WAVE PROBE", 0, wave_length*2, 0, 0)
FM.set_point_probe(3, "WAVE PROBE", 0, wave_length*3, 0, 0)
FM.set_point_probe(4, "WAVE PROBE", 0, wave_length*4, 0, 0)
FM.set_point_probe(5, "WAVE PROBE", 0, wp5, 0, 0)

#    Save computation and run script

FM.save_active_computation()
FM.save_project()
run_script("/usr/numeca/finemarine72/_python/_fine/_marine/
   Predefined/Wave_generation_infos.py")
```

D | Matlab Scripts

D.1 Main Postprocessing Script: post_processing.m

```
%   POST PROCESSING
%
%   This script is written as a part of a Master Thesis about Numerical
%   Wave Tanks in Computational Fluid Dynamics.
%
%   This is the main script of all post_processing in Matlab.
%   It calls to many subscripts handling different parts of the analysis.
%   The idea behind it is to make it easy to present some results without
%   plotting all at the same time.
%
%   Throughout the script several functions are used. import_isoline.m and
%   import_probe_data.m are two of them, and they are used to import the
%   .dat file and make it into readable files.
%
%   Latest edited: 27.05.2019
%   Written by: Mari T. Storsul

%%  Sensitivity Analysis in 2D

LR_sensitivity  %   Length                      % Figures nb. start w/ 10
HS_sensitivity  %   Height                      % Figures nb. start w/ 20
DZ_sensitivity  %   Length of Damping Zone      % Figures nb. start w/ 30
BC_sensitivity  %   Lower Boundary Condition    % Figures nb. start w/ 40
MR_sensitivity  %   Mesh Refinement             % Figures nb. start w/ 50
DS_sensitivity  %   Discretization Scheme       % Figures nb. start w/ 60

%%  2D Regular Waves Analysis

regular_waves   %   Regular analysis with 16 different cases (8 IWG, 8 WG)
                %   Figures nb. start w/80

%%  2D Irregular Waves Analysis

irregular_analysis  %   Irregular Analysis      % Figures nb. start w/ 70

%%  3D Regular Wave Analysis

regular_waves_3D    %   Regular analysis with 3D wave probe data. Same
                    %   wave characteristics as comp. 3 and 4 in 2D
                    %   analysis.
                    %   Figures nb. start w/ 90
```

D.2 Subscript: LR_sensitivity.m

```
%   LR Sensitivity

%%   Import Data
iso_LR_37 = import_isoline('isoline_2D_IWG_LR_37.dat',4,inf);
iso_LR_50 = import_isoline('isoline_2D_IWG_LR_50.dat',4,inf);
iso_LR_62 = import_isoline('isoline_2D_IWG_LR_62.dat',4,inf);
wp_LR_37 = import_probe_data('points_probe_2D_IWG_LR_37.dat',16,inf);
wp_LR_50 = import_probe_data('points_probe_2D_IWG_LR_50.dat',16,inf);
wp_LR_62 = import_probe_data('points_probe_2D_IWG_LR_62.dat',16,inf);

%%  Defining Theoretical Values
t = 0:0.0025:80;
sine_wave = 0.05*sin(pi*t-pi);

x = 0:0.0025:62.5;
y = 0.05*sin((2*pi/6.24524)*x-pi);

%%  Calculating Theoretical Values From Time Series

mean_LR_37 = mean(wp_LR_37(:,3));
std_LR_37 = std(wp_LR_37(:,3));
var_LR_37 = var(wp_LR_37(:,3));

mean_LR_50 = mean(wp_LR_50(:,3));
std_LR_50 = std(wp_LR_50(:,3));
var_LR_50 = var(wp_LR_50(:,3));

mean_LR_62 = mean(wp_LR_62(:,3));
std_LR_62 = std(wp_LR_62(:,3));
var_LR_62 = var(wp_LR_62(:,3));

%%  Plots

%   Wave Probe 2
figure(101)
plot(wp_LR_37(:,1),wp_LR_37(:,3) ,...
    wp_LR_50(:,1),wp_LR_50(:,3) ,...
    wp_LR_62(:,1),wp_LR_62(:,3) ,...
    t,sine_wave,':','LineWidth',2)
legend('L = 37.5 [m]','L = 50.0 [m]','L = 62.5 [m]','Theory')
legend('Location','southeast')
grid on
grid minor
set (gcf,'position',[10,10,900,300])
xlabel 'Time [s]'
ylabel 'Wave elevation [m]'
xlim([76 80])
saveas(figure(101),'Figures/SA/LSA_wp2.png')

%   Plot of Isoline 0.5
%{
figure(102)
plot(iso_LR_37(:,1),iso_LR_37(:,2),'*' ,...
    iso_LR_50(:,1),iso_LR_50(:,2),'*' ,...
    iso_LR_62(:,1),iso_LR_62(:,2),'*' ,...
    x,y,'-')
```

```
57 legend('L = 37.5 [m]','L = 50.0 [m]','L = 62.5 [m]','Theory')
58 grid on
59 grid minor
60 set(gcf,'position',[10,10,900,300])
61 xlabel 'X-coordinate [m]'
62 ylabel 'Wave elevation [m]'
63 xlim([0 62.5])
64 saveas(figure(102),'Figures/SA/LSA_iso.png')
65 %}
```

D.3 Subscript: HS_sensitivity.m

```
%   HS Sensitivity

%%   Import data
iso_HS_3 = import_isoline('isoline_2D_IWG_HS_3.dat',4,inf);
iso_HS_7 = import_isoline('isoline_2D_IWG_HS_7.dat',4,inf);
iso_HS_9 = import_isoline('isoline_2D_IWG_HS_9.dat',4,inf);
wp_HS_3 = import_probe_data('points_probe_2D_IWG_HS_3.dat',16,inf);
wp_HS_7 = import_probe_data('points_probe_2D_IWG_HS_7.dat',16,inf);
wp_HS_9 = import_probe_data('points_probe_2D_IWG_HS_9.dat',16,inf);

%%  Calculating Theoretical Values From Time Series

mean_HS_3 = mean(wp_HS_3(:,3));
std_HS_3 = std(wp_HS_3(:,3));
var_HS_3 = var(wp_HS_3(:,3));

mean_HS_7 = mean(wp_HS_7(:,3));
std_HS_7 = std(wp_HS_7(:,3));
var_HS_7 = var(wp_HS_7(:,3));

mean_HS_9 = mean(wp_HS_9(:,3));
std_HS_9 = std(wp_HS_9(:,3));
var_HS_9 = var(wp_HS_9(:,3));

%%  Plots

%   Wave Probe 2
figure(201)
plot(wp_HS_3(:,1),wp_HS_3(:,3),...
    wp_LR_62(:,1),wp_LR_62(:,3),...
    wp_HS_7(:,1),wp_HS_7(:,3),...
    wp_HS_9(:,1),wp_HS_9(:,3),...
    t,sine_wave,':','LineWidth',2)
legend('h = 3.0 [m]','h = 5.0 [m]','h = 7.0 [m]','h = 9.0 [m]','Theory')
legend('Location','southeast')
grid on
grid minor
set (gcf,'position',[10,10,900,300])
xlabel 'Time [s]'
ylabel 'Wave elevation [m]'
xlim([76 80])
saveas(figure(201),'Figures/SA/HSA_wp2.png')
```

D.4 Subscript: DZ_sensitivity.m

```
%   DZ Sensitivity

%%  Import Data
%   All isoline data is taken at time step 200 in GLView
iso_DZ_3 = import_isoline('isoline_2D_IWG_DZ_3.dat',4,inf);
iso_DZ_25 = import_isoline('isoline_2D_IWG_DZ_25.dat',4,inf);
iso_DZ_35 = import_isoline('isoline_2D_IWG_DZ_35.dat',4,inf);
iso_DZ_4 = import_isoline('isoline_2D_IWG_DZ_4.dat',4,inf);
wp_DZ_3 = import_probe_data('points_probe_2D_IWG_DZ_3.dat',16,inf);
wp_DZ_25 = import_probe_data('points_probe_2D_IWG_DZ_25.dat',16,inf);
wp_DZ_35 = import_probe_data('points_probe_2D_IWG_DZ_35.dat',16,inf);
wp_DZ_4 = import_probe_data('points_probe_2D_IWG_DZ_4.dat',16,inf);

%   Wave Generator data
wp_DZ_WG = import_probe_data('points_probe_2D_WG_Reg_Waves1.dat',16,inf);

%%  Defining Theoretical Values
x = 0:0.0025:62.5;
y = 0.05*sin((2*pi/6.24524)*x+pi*4.66);

%%  Calculating Theoretical Values From Time Series

mean_DZ_25 = mean(wp_DZ_25(:,3));
std_DZ_25 = std(wp_DZ_25(:,3));
var_DZ_25 = var(wp_DZ_25(:,3));

mean_DZ_3 = mean(wp_DZ_3(:,3));
std_DZ_3 = std(wp_DZ_3(:,3));
var_DZ_3 = var(wp_DZ_3(:,3));

mean_DZ_35 = mean(wp_DZ_35(:,3));
std_DZ_35 = std(wp_DZ_35(:,3));
var_DZ_35 = var(wp_DZ_35(:,3));

mean_DZ_4 = mean(wp_DZ_4(:,3));
std_DZ_4 = std(wp_DZ_4(:,3));
var_DZ_4 = var(wp_DZ_4(:,3));

%%  Plots

%   Wave Probe 5
figure(301)
plot(wp_LR_62(:,1),wp_LR_62(:,6),...
    wp_DZ_25(:,1),wp_DZ_25(:,6),...
    wp_DZ_3(:,1),wp_DZ_3(:,6),...
    wp_DZ_35(:,1),wp_DZ_35(:,6),...
    wp_DZ_4(:,1),wp_DZ_4(:,6),'LineWidth',2)
legend('DZ = 2.0 x wave length','DZ = 2.5 x wave length', ...
    'DZ = 3.0 x wave length','DZ = 3.5 x wave length',...
    'DZ = 4.0 x wave length')
grid on
grid minor
set (gcf,'position',[10,10,900,300])
xlabel 'Time [s]'
ylabel 'Wave elevation [m]'
saveas(figure(301),'Figures/SA/DSA_wp5.png')
```

```
57
58 %    Isoline
59 figure(302)
60 plot(iso_LR_62(:,1),iso_LR_62(:,2),...
61      iso_DZ_25(:,1),iso_DZ_25(:,2),...
62      iso_DZ_3(:,1),iso_DZ_3(:,2),...
63      iso_DZ_35(:,1),iso_DZ_35(:,2),...
64      iso_DZ_4(:,1),iso_DZ_4(:,2),...
65      x,y,':','LineWidth',2)
66 legend('DZ = 2.0 x wave length','DZ = 2.5 x wave length', ...
67      'DZ = 3.0 x wave length','DZ = 3.5 x wave length',...
68      'DZ = 4.0 x wave length','Theory')
69 grid on
70 grid minor
71 set(gcf,'position',[10,10,900,300])
72 xlabel 'X-coordinate [m]'
73 ylabel 'Wave elevation [m]'
74 xlim([40 62.5])
75 saveas(figure(302),'Figures/SA/DSA_iso.png')
76
77 %%  Plots from WG - used for validation
78 %   WP 5
79 figure(303)
80 plot(wp_DZ_WG(:,1),wp_DZ_WG(:,6),'LineWidth',2)
81 legend('Wave Generator')
82 grid on
83 grid minor
84 set (gcf,'position',[10,10,900,300])
85 xlabel 'Time [s]'
86 ylabel 'Wave elevation [m]'
87 saveas(figure(303),'Figures/SA/DSA_WG_wp5.png')
```

D.5 Subscript: BC_sensitivity.m

```
%   BC Sensitivity

%%  Import Data
iso_BC_hydropress = import_isoline('isoline_2D_IWG_BC_lower_hydropress_2.dat',4,
    inf);
wp_BC_hydropress = import_probe_data('points_probe_2D_IWG_BC_lower_hydropress_2.
    dat',16,inf);

iso_BC_slipwall = import_isoline('isoline_2D_IWG_BC_lower_slipwall.dat',4,inf);
wp_BC_slipwall = import_probe_data('points_probe_2D_IWG_BC_lower_slipwall.dat'
    ,16,inf);

%%   Defining theoretical values
%   Phase difference w*t where t is +4.66 sec.
%   For isolines, time step 7961
x = 0:0.0025:62.5;
y = 0.05*sin((2*pi/6.24524)*x+pi*4.66);
%   For wave probes
t = 0:0.0025:80;
sine_wave = 0.05*sin(pi*t-pi);

%%  Calculating Theoretical Values From Time Series

mean_BC_hydropress = mean(wp_BC_hydropress(:,3));
std_BC_hydropress = std(wp_BC_hydropress(:,3));
var_BC_hydropress = var(wp_BC_hydropress(:,3));

mean_BC_slipwall = mean(wp_BC_slipwall(:,3));
std_BC_slipwall = std(wp_BC_slipwall(:,3));
var_BC_slipwall = var(wp_BC_slipwall(:,3));

mean_BC_farfield = mean(wp_DZ_35(:,3));
std_BC_farfield = std(wp_DZ_35(:,3));
var_BC_farfield = var(wp_DZ_35(:,3));

%%   Plot of wave probe 2
figure(401)
plot(wp_BC_hydropress(:,1),wp_BC_hydropress(:,3),...
    wp_BC_slipwall(:,1),wp_BC_slipwall(:,3),...
    wp_DZ_35(:,1),wp_DZ_35(:,3),...
    t,sine_wave,':','LineWidth',2)
legend('Hydrostatic Pressure','Slip Wall','Far Field','Theory')
legend('Location','southeast')
grid on
grid minor
set (gcf,'position',[10,10,900,300])
xlim([76 80])
xlabel 'Time [s]'
ylabel 'Wave elevation [m]'
saveas(figure(401),'Figures/SA/BC_wp2.png')

%%  Plot of wave probe 5 (in connection to damping sensitivity)
figure(402)
plot(wp_BC_hydropress(:,1),wp_BC_hydropress(:,6),...
    wp_BC_slipwall(:,1),wp_BC_slipwall(:,6),...
    wp_DZ_35(:,1),wp_DZ_35(:,6),'LineWidth',2)
```

```
legend('Hydrostatic Pressure','Slip Wall','Far Field')
legend('Location','southwest')
grid on
set (gcf,'position',[10,10,900,300])
xlabel 'Time [s]'
ylabel 'Wave elevation [m]'
saveas(figure(402),'Figures/SA/BC_wp5.png')

%%  Detailed plot of the first 20 sec
figure(403)
plot(wp_BC_hydropress(:,1),wp_BC_hydropress(:,6),...
    wp_BC_slipwall(:,1),wp_BC_slipwall(:,6),...
    wp_DZ_35(:,1),wp_DZ_35(:,6),'LineWidth',2)
legend('Hydrostatic Pressure','Slip Wall','Far Field')
legend('Location','southwest')
grid on
grid minor
set (gcf,'position',[10,10,900,300])
xlim([0 20])
xlabel 'Time [s]'
ylabel 'Wave elevation [m]'
saveas(figure(403),'Figures/SA/BC_wp5_detailed.png')

%%   Plot of isoline
figure(404)
plot(iso_BC_hydropress(:,1),iso_BC_hydropress(:,2),...
    iso_BC_slipwall(:,1),iso_BC_slipwall(:,2),...
    iso_DZ_35(:,1),iso_DZ_35(:,2),...
    x,y,':','LineWidth',2)
legend('Hydrostatic Pressure','Slip Wall','Far Field','Theory')
grid on
grid minor
set(gcf,'position',[10,10,900,300])
xlabel 'X-coordinate [m]'
ylabel 'Wave elevation [m]'
xlim([0 62.5])
saveas(figure(404),'Figures/SA/BC_iso.png')
```

D.6 Subscript: MR_sensitivity.m

```
% MR Sensitivity

%%  Import Data
%   All isoline data is taken at time step XXXX in GLView

iso_AR_200 = import_isoline('isoline_2D_IWG_AR_200.dat',4,inf);
iso_AR_50 = import_isoline('isoline_2D_IWG_AR_50.dat',4,inf);
iso_AR_12 = import_isoline('isoline_2D_IWG_AR_12.dat',4,inf);
iso_AGR = import_isoline('isoline_2D_IWG_AGR_1.dat',4,inf);
wp_AR_200 = import_probe_data('points_probe_2D_IWG_AR_200.dat',16,inf);
wp_AR_50 = import_probe_data('points_probe_2D_IWG_AR_50.dat',16,inf);
wp_AR_12 = import_probe_data('points_probe_2D_IWG_AR_12.dat',16,inf);
wp_AGR = import_probe_data('points_probe_2D_IWG_AGR_1.dat',16,inf);

% Results from project analysis
import_project_thesis_result    % variables IWGfinalISO and IWGfinalWP

%%  Calculating Theoretical Values From Time Series

mean_AGR = mean(wp_AGR(:,3));
std_AGR = std(wp_AGR(:,3));
var_AGR = var(wp_AGR(:,3));

%   Plot of wave probe 2 for Increasing Aspect Ratios
%{
figure(501)
plot(wp_AR_200(:,1),wp_AR_200(:,3),'*',...
    wp_AR_50(:,1),wp_AR_50(:,3),'*',...
    wp_AR_12(:,1),wp_AR_12(:,3),'*',...
    t,sine_wave,'-')
legend('AR = 200','AR = 50','AR = 12.5','Theory')
legend('Location','southeast')
grid on
grid minor
set(gcf,'position',[10,10,900,300])
xlim([76 80])
xlabel 'Time [s]'
ylabel 'Wave elevation [m]'
saveas(figure(501),'Figures/SA/MR_wp2.png')
%}

%% Plot of wave probe 2
figure(502)
plot(wp_AGR(:,1),wp_AGR(:,3),...
    wp_BC_hydropress(:,1),wp_BC_hydropress(:,3),...
    t,sine_wave,':','LineWidth',2)
legend('With AGR','Without AGR','Theory')
legend('Location','southeast')
grid on
grid minor
set(gcf,'position',[10,10,900,300])
xlim([76 80])
xlabel 'Time [s]'
ylabel 'Wave elevation [m]'
saveas(figure(502),'Figures/SA/MR_wp2.png')

```

```
%% Plot of wave probe 3
figure(503)
plot(wp_AGR(:,1),wp_AGR(:,4),...
    IWGfinalWP(:,1)-0.3202,IWGfinalWP(:,3),'LineWidth',2)
legend('With AGR','From Project Thesis')
legend('Location','southeast')
grid on
grid minor
set(gcf,'position',[10,10,900,300])
xlim([76 80])
xlabel 'Time [s]'
ylabel 'Wave elevation [m]'
saveas(figure(503),'Figures/SA/MR_wp.png')
%   0.3202 is the time it takes for the wave to pass from wave probe 3
%   in the master setup to wave probe 2 in the project setup

%%   Plot of isoline
figure(504)
plot(iso_AGR(:,1),iso_AGR(:,2),...
    iso_BC_hydropress(:,1),iso_BC_hydropress(:,2),...
    x,y,':','LineWidth',2)
legend('With AGR','Without AGR','Theory')
grid on
grid minor
set(gcf,'position',[10,10,900,300])
xlabel 'X-coordinate [m]'
ylabel 'Wave elevation [m]'
xlim([0 62.5])
saveas(figure(504),'Figures/SA/MR_iso.png')
```

D.7 Subscript: DS_sensitivity.m

```
%   DS Sensitivity

%%  Import Data

iso_DS_AVLSMART = import_isoline('isoline_2D_IWG_DS_AVLSMART.dat',4,inf);
iso_DS_BICS = import_isoline('isoline_2D_IWG_DS_BICS.dat',4,inf);
iso_DS_BRICS = import_isoline('isoline_2D_IWG_DS_BRICS.dat',4,inf);

wp_DS_AVLSMART = import_probe_data('points_probe_2D_IWG_DS_AVLSMART.dat',16,inf)
    ;
wp_DS_BICS = import_probe_data('points_probe_2D_IWG_DS_BICS.dat',16,inf);
wp_DS_BRICS = import_probe_data('points_probe_2D_IWG_DS_BRICS.dat',16,inf);

%%  Theoretical Result
t = 0:0.0025:80;
sine_wave = 0.05*sin(pi*t-pi);

%%  Calculating Theoretical Values From Time Series

mean_DS_AVLSMART = mean(wp_DS_AVLSMART(:,3));
std_DS_AVLSMART = std(wp_DS_AVLSMART(:,3));
var_DS_AVLSMART = var(wp_DS_AVLSMART(:,3));

mean_DS_BICS = mean(wp_DS_BICS(:,3));
std_DS_BICS = std(wp_DS_BICS(:,3));
var_DS_BICS = var(wp_DS_BICS(:,3));

mean_DS_BRICS = mean(wp_DS_BRICS(:,3));
std_DS_BRICS = std(wp_DS_BRICS(:,3));
var_DS_BRICS = var(wp_DS_BRICS(:,3));

%%  Plots

%   Wave Probe 2
figure(601)
plot(wp_DS_AVLSMART(:,1),wp_DS_AVLSMART(:,3),...
    wp_DS_BICS(:,1),wp_DS_BICS(:,3),...
    wp_DS_BRICS(:,1),wp_DS_BRICS(:,3),...
    t,sine_wave,':','LineWidth',2)
legend('AVLSMART','BICS','BRICS','Theory')
legend('Location','southeast')
grid on
grid minor
set(gcf,'position',[10,10,900,300])
xlim([74 80])
xlabel 'Time [s]'
ylabel 'Wave elevation [m]'
saveas(figure(601),'Figures/SA/DS_wp2.png')

%   Isoline 0.5
figure(602)
plot(iso_DS_AVLSMART(:,1),iso_DS_AVLSMART(:,2),...
    iso_DS_BICS(:,1),iso_DS_BICS(:,2),...
    iso_DS_BRICS(:,1),iso_DS_BRICS(:,2),'LineWidth',2)
legend('AVLSMART','BICS','BRICS')
grid on
```

```
grid minor
set(gcf,'position',[10,10,900,300])
xlim([0 62.5])
xlabel 'X-coordinate [m]'
ylabel 'Wave elevation [m]'
saveas(figure(602),'Figures/SA/DS_iso.png')
```

D.8 Subscript: regular_waves.m

```
%   Regular wave analysis
%
%   Note that the variables called "wave" here is referring to the
%   computations mentioned throughout the thesis. Computations 1 and 2 are
%   performed with wave A, 7 and 8 with wave B, etc. Computations 3 and 4
%   are performed with wave D.

%%  Import Data

for i = 1:1:3
    iso_wave{i} = import_isoline(['iso_wave' num2str(i) '.dat'],4,inf);
end

for i = 5:1:13
    iso_wave{i} = import_isoline(['iso_wave' num2str(i) '.dat'],4,inf);
end

for i = 17:1:18
iso_wave{i} = import_isoline(['iso_wave' num2str(i) '.dat'],4,inf);
end

%   The generalized function for importing isolines did not work for wave
%   4 and 14, so two autogenerated script were used. Wave 16 exploded so
%   no isoline exist.
iso_wave4_script
iso_wave14_script
iso_wave{15} = import_isoline('iso_wave15.dat',4,inf);

for i = 1:1:18
wp_wave{i} = import_probe_data(['points_probe_wave' num2str(i) '.dat'],16,inf);
end

%%  Wave Statistics For Wave Probe 2

%   Preallocating space for matrices
mean_waves = zeros(1,18);
std_waves = zeros(1,18);
var_waves = zeros(1,18);

for  i = 1:1:18                         %   Goes through all waves
    data = cell2mat(wp_wave(i));        %   Retrieving wave probe data
    mean_waves(i) = mean(data(:,3));    %   Calculating mean water level
    std_waves(i) = std(data(:,3));      %   Calculating standard deviation
    var_waves(i) = var(data(:,3));      %   Calculating variance
end

%%  Theoretical Wave Statistics

t = 0:0.0025:80;
sine_wave = 0.05*sin(pi*t);

x_IWG = 0:0.0025:43.7643;
y_IWG = 0.05*sin((2*pi/6.24524)*x_IWG);

x_WG = 0:0.0025:43.7643;
y_WG = 0.05*sin((2*pi/6.24524)*x_WG-pi);
```

```
57
58 %%  Plots of Wave Elevation
59
60 %   Free surface elevation at wave probe 2
61 %   For both IWG and WG with same wave statistics in the same plot
62 wp_wave3 = cell2mat(wp_wave(3));
63 wp_wave4 = cell2mat(wp_wave(4));
64
65 figure(801)
66 plot(wp_wave3(:,1)-1,wp_wave3(:,3),...  %   -1 due to phase difference
67     wp_wave4(:,1),wp_wave4(:,3),...
68     t,sine_wave,'LineWidth',2)
69 legend('Internal Wave Generator','Wave Generator','Theoretical Wave')
70 grid on
71 grid minor
72 set(gcf,'position',[10,10,900,300])
73 xlim([72 78])
74 xlabel 'Time [s]'
75 ylabel 'Wave elevation [m]'
76 saveas(figure(801),'Figures/Regular/wave34_wp2.png')
77
78 %   Free surface elevation at wave probe 1,2,3 and 4 for wave 3
79 figure(802)
80 plot(wp_wave3(:,1),wp_wave3(:,2),...
81     wp_wave3(:,1),wp_wave3(:,3),...
82     wp_wave3(:,1),wp_wave3(:,4),...
83     wp_wave3(:,1),wp_wave3(:,5),'LineWidth',2)
84 legend('WP 1','WP2','WP3','WP4')
85 legend('Location','southeast')
86 grid on
87 grid minor
88 set(gcf,'position',[10,10,900,300])
89 xlim([72 78])
90 xlabel 'Time [s]'
91 ylabel 'Wave elevation [m]'
92 saveas(figure(802),'Figures/Regular/wave3_wps.png')
93
94 %   Free surface elevation at wave probe 1,2,3 and 4 for wave 4
95 figure(803)
96 plot(wp_wave4(:,1),wp_wave4(:,2),...
97     wp_wave4(:,1),wp_wave4(:,3),...
98     wp_wave4(:,1),wp_wave4(:,4),...
99     wp_wave4(:,1),wp_wave4(:,5),'LineWidth',2)
100 legend('WP 1','WP2','WP3','WP4')
101 grid on
102 grid minor
103 set(gcf,'position',[10,10,900,300])
104 xlim([72 78])
105 xlabel 'Time [s]'
106 ylabel 'Wave elevation [m]'
107 saveas(figure(803),'Figures/Regular/wave4_wps.png')
108
109 %%  Plot of Wave Spectrums for the Above Elevations
110
111 %   Spectrums Wave 3
112 S_wave3_wp1 = dat2spec([wp_wave3(800:8000,1) wp_wave3(800:8000,2)]);
113 S_wave3_wp2 = dat2spec([wp_wave3(800:8000,1) wp_wave3(800:8000,3)]);
114 S_wave3_wp3 = dat2spec([wp_wave3(800:8000,1) wp_wave3(800:8000,4)]);
115 S_wave3_wp4 = dat2spec([wp_wave3(800:8000,1) wp_wave3(800:8000,5)]);
116
```

```
%   Plot of Wave Spectrums for Wave 3
%{
figure(804)
plot(S_wave3_wp1.w,S_wave3_wp1(:,1).S,...
    S_wave3_wp2.w,S_wave3_wp2(:,1).S,...
    S_wave3_wp3.w,S_wave3_wp3(:,1).S,...
    S_wave3_wp4.w,S_wave3_wp4(:,1).S,'LineWidth',2)
legend('WP 1','WP 2','WP 3','WP 4')
grid on
grid minor
xlim([0 10])
xlabel 'Frequency [rad/s]'
ylabel 'Spectral density [(Nm)^2/rad]'
saveas(figure(804),'Figures/Regular/wave3_spec.png')
%}

%   Spectrums Wave 4
S_wave4_wp1 = dat2spec([wp_wave4(800:8000,1) wp_wave4(800:8000,2)]);
S_wave4_wp2 = dat2spec([wp_wave4(800:8000,1) wp_wave4(800:8000,3)]);
S_wave4_wp3 = dat2spec([wp_wave4(800:8000,1) wp_wave4(800:8000,4)]);
S_wave4_wp4 = dat2spec([wp_wave4(800:8000,1) wp_wave4(800:8000,5)]);

%   Plot of Wave Spectrums for Wave 4
%{
figure(805)
plot(S_wave4_wp1.w,S_wave4_wp1(:,1).S,...
    S_wave4_wp2.w,S_wave4_wp2(:,1).S,...
    S_wave4_wp3.w,S_wave4_wp3(:,1).S,...
    S_wave4_wp4.w,S_wave4_wp4(:,1).S,'LineWidth',2)
legend('WP 1','WP 2','WP 3','WP 4')
grid on
grid minor
xlim([0 10])
xlabel 'Frequency [rad/s]'
ylabel 'Spectral density [(Nm)^2/rad]'
saveas(figure(805),'Figures/Regular/wave4_spec.png')
%}

%   Both of all Wave Spectrums in the same Plot
figure(806)
plot(S_wave3_wp1.w,S_wave3_wp1(:,1).S,...
    S_wave3_wp2.w,S_wave3_wp2(:,1).S,...
    S_wave3_wp3.w,S_wave3_wp3(:,1).S,...
    S_wave3_wp4.w,S_wave3_wp4(:,1).S,...
    S_wave4_wp1.w,S_wave4_wp1(:,1).S,'-.',...
    S_wave4_wp2.w,S_wave4_wp2(:,1).S,'-.',...
    S_wave4_wp3.w,S_wave4_wp3(:,1).S,'-.',...
    S_wave4_wp4.w,S_wave4_wp4(:,1).S,'-.','LineWidth',2)
legend('WP 1 (Comp. 3: IWG)','WP 2 (Comp. 3: IWG)','WP 3 (Comp. 3: IWG)',...
    'WP 4 (Comp. 3: IWG)', 'WP 1 (Comp. 4: WG)','WP 2 (Comp. 4: WG)',...
    'WP 3 (Comp. 4: WG)','WP 4 (Comp. 4: WG)')
grid on
grid minor
xlim([0 10])
xlabel 'Frequency [rad/s]'
ylabel 'Spectral density [(Nm)^2/rad]'
saveas(figure(806),'Figures/Regular/wave34_spec.png')

%%  Maximum of Spectral Density

```

```
max_spec_wave3(1) = max(S_wave3_wp1.S);
max_spec_wave3(2) = max(S_wave3_wp2.S);
max_spec_wave3(3) = max(S_wave3_wp3.S);
max_spec_wave3(4) = max(S_wave3_wp4.S);

max_spec_wave4(1) = max(S_wave4_wp1.S);
max_spec_wave4(2) = max(S_wave4_wp2.S);
max_spec_wave4(3) = max(S_wave4_wp3.S);
max_spec_wave4(4) = max(S_wave4_wp4.S);

%% Plot of Mass Fraction (Isoline 0.5)

mf_wave3 = cell2mat(iso_wave(3));
mf_wave4 = cell2mat(iso_wave(4));

%   Damping Zone Patch Definition
vertex = [43.7643 -0.06; 62.5 -0.06; 62.5 0.06; 43.7643 0.06];
faces = [1 2 3 4];

%   Wave 3 Mass Fraction 0.5
figure(807)
patch('Faces',faces,'Vertices',vertex,'FaceColor','k','FaceAlpha',.1);
hold on
plot(mf_wave3(:,1),mf_wave3(:,2),...
    x_IWG, y_IWG,':','LineWidth',2)
legend('Damping Zone','Computation 3','Theory')
grid on
grid minor
set(gcf,'position',[10,10,900,300])
xlabel 'X-coordinate [m]'
ylabel 'Wave elevation [m]'
xlim([0 62.5])
saveas(figure(807),'Figures/Regular/wave3_iso.png')
hold off

%   Wave 4 Mass Fraction 0.5
figure(808)
patch('Faces',faces,'Vertices',vertex,'FaceColor','k','FaceAlpha',.1);
hold on
plot(mf_wave4(:,1),mf_wave4(:,2), ...
    x_WG, y_WG,':','LineWidth',2)
legend('Damping Zone','Computation 4','Theory')
grid on
grid minor
set(gcf,'position',[10,10,900,300])
xlabel 'X-coordinate [m]'
ylabel 'Wave elevation [m]'
xlim([0 62.5])
saveas(figure(808),'Figures/Regular/wave4_iso.png')
hold off

%   Wave 3 & 4 Mass Fraction 0.5 plot for discussion
figure(809)
patch('Faces',faces,'Vertices',vertex,'FaceColor','k','FaceAlpha',.1);
hold on
plot(mf_wave3(:,1),mf_wave3(:,2),...
    mf_wave4(:,1),mf_wave4(:,2),'LineWidth',2)
legend('Damping Zone','Computation 3 (IWG)','Computation 4 (WG)')
grid on
grid minor
```

```
xlabel 'X-coordinate [m]'
ylabel 'Wave elevation [m]'
xlim([40 62.5])
saveas(figure(809),'Figures/Regular/wave34_iso_discussion.png')
hold off

%% Addidional checks of the isolines
%   Especially how the sponge layer behaves

mf_wave1 = cell2mat(iso_wave(1));
mf_wave2 = cell2mat(iso_wave(2));

mf_wave5 = cell2mat(iso_wave(5));
mf_wave6 = cell2mat(iso_wave(6));
mf_wave7 = cell2mat(iso_wave(7));
mf_wave8 = cell2mat(iso_wave(8));
mf_wave9 = cell2mat(iso_wave(9));
mf_wave10 = cell2mat(iso_wave(10));
mf_wave11 = cell2mat(iso_wave(11));
mf_wave12 = cell2mat(iso_wave(12));
mf_wave13 = cell2mat(iso_wave(13));
mf_wave14 = cell2mat(iso_wave(14));
mf_wave15 = cell2mat(iso_wave(15));

mf_wave17 = cell2mat(iso_wave(17));
mf_wave18 = cell2mat(iso_wave(18));

%   Damping Zone Patch Definition
vertex2 = [43.7643 -0.2; 62.5 -0.2; 62.5 0.2; 43.7643 0.2]; % T = 2,1.5
vertex4 = [28.347 -0.2; 62.5 -0.2; 62.5 0.2; 28.347 0.2];   % T = 2.5
faces = [1 2 3 4];

%   Comp. 1&2
figure(810)
patch('Faces',faces,'Vertices',vertex2,'FaceColor','k','FaceAlpha',.1);
hold on
plot(mf_wave1(:,1),mf_wave1(:,2),...
    mf_wave2(:,1),mf_wave2(:,2),'LineWidth',2)
legend('Damping Zone','Computation 1 (IWG)','Computation 2 (WG)')
grid on
grid minor
set(gcf,'position',[10,10,900,300])
xlabel 'X-coordinate [m]'
ylabel 'Wave elevation [m]'
xlim ([0 62.5])
ylim ([-0.05 0.05])
saveas(figure(810),'Figures/Regular/wave12_iso.png')
hold off

%   Comp. 5&6
figure(811)
patch('Faces',faces,'Vertices',vertex4,'FaceColor','k','FaceAlpha',.1);
hold on
plot(mf_wave5(:,1),mf_wave5(:,2),...
    mf_wave6(:,1),mf_wave6(:,2),'LineWidth',2)
legend('Damping Zone','Computation 5 (IWG)','Computation 6 (WG)')
grid on
grid minor
set(gcf,'position',[10,10,900,300])
xlabel 'X-coordinate [m]'
```

```
ylabel 'Wave elevation [m]'
xlim ([0 62.5])
ylim ([-0.07 0.07])
saveas(figure(811),'Figures/Regular/wave56_iso.png')
hold off

%   Comp. 7&8
figure(812)
patch('Faces',faces,'Vertices',vertex2,'FaceColor','k','FaceAlpha',.1);
hold on
plot(mf_wave7(:,1),mf_wave7(:,2),...
    mf_wave8(:,1),mf_wave8(:,2),'LineWidth',2)
legend('Damping Zone','Computation 7 (IWG)','Computation 8 (WG)')
grid on
grid minor
set(gcf,'position',[10,10,900,300])
xlabel 'X-coordinate [m]'
ylabel 'Wave elevation [m]'
xlim ([0 62.5])
ylim ([-0.12 0.12])
saveas(figure(812),'Figures/Regular/wave78_iso.png')
hold off

%   Comp. 9&10
figure(813)
patch('Faces',faces,'Vertices',vertex2,'FaceColor','k','FaceAlpha',.1);
hold on
plot(mf_wave9(:,1),mf_wave9(:,2),...
    mf_wave10(:,1),mf_wave10(:,2),'LineWidth',2)
legend('Damping Zone','Computation 9 (IWG)','Computation 10 (WG)')
grid on
grid minor
set(gcf,'position',[10,10,900,300])
xlabel 'X-coordinate [m]'
ylabel 'Wave elevation [m]'
xlim ([0 62.5])
ylim ([-0.12 0.12])
saveas(figure(813),'Figures/Regular/wave910_iso.png')
hold off

%   Comp. 11&12
figure(814)
patch('Faces',faces,'Vertices',vertex4,'FaceColor','k','FaceAlpha',.1);
hold on
plot(mf_wave11(:,1),mf_wave11(:,2),...
    mf_wave12(:,1),mf_wave12(:,2),'LineWidth',2)
legend('Damping Zone','Computation 11 (IWG)','Computation 12 (WG)')
grid on
grid minor
set(gcf,'position',[10,10,900,300])
xlabel 'X-coordinate [m]'
ylabel 'Wave elevation [m]'
xlim ([0 62.5])
ylim ([-0.12 0.12])
saveas(figure(814),'Figures/Regular/wave112_iso.png')
hold off

%   Comp. 13&14
figure(815)
patch('Faces',faces,'Vertices',vertex2,'FaceColor','k','FaceAlpha',.1);
```

```
hold on
plot(mf_wave13(:,1),mf_wave13(:,2),...
    mf_wave14(:,1),mf_wave14(:,2),'LineWidth',2)
legend('Damping Zone','Computation 13 (IWG)','Computation 14 (WG)')
grid on
grid minor
set(gcf,'position',[10,10,900,300])
xlabel 'X-coordinate [m]'
ylabel 'Wave elevation [m]'
xlim ([0 62.5])
ylim ([-0.15 0.15])
saveas(figure(815),'Figures/Regular/wave1314_iso.png')
hold off

%   Comp. 17&18
figure(816)
patch('Faces',faces,'Vertices',vertex4,'FaceColor','k','FaceAlpha',.1);
hold on
plot(mf_wave17(:,1),mf_wave17(:,2),...
    mf_wave18(:,1),mf_wave18(:,2),'LineWidth',2)
legend('Damping Zone','Computation 17 (IWG)','Computation 18 (WG)')
grid on
grid minor
set(gcf,'position',[10,10,900,300])
xlabel 'X-coordinate [m]'
ylabel 'Wave elevation [m]'
xlim ([0 62.5])
ylim ([-0.17 0.17])
saveas(figure(816),'Figures/Regular/wave1718_iso.png')
hold off

%   Comp. 15
figure(817)
patch('Faces',faces,'Vertices',vertex2,'FaceColor','k','FaceAlpha',.1);
hold on
plot(mf_wave15(:,1),mf_wave15(:,2),'LineWidth',2)
legend('Damping Zone','Computation 15 (IWG)')
grid on
grid minor
set(gcf,'position',[10,10,900,300])
xlabel 'X-coordinate [m]'
ylabel 'Wave elevation [m]'
xlim ([0 62.5])
ylim ([-0.17 0.17])
saveas(figure(817),'Figures/Regular/wave15_iso.png')
hold off
%%  Other Necessary Wave Statistics

%   Evolution of wave 3
mean_wave3 = zeros(1,4);
std_wave3 = zeros(1,4);
var_wave3 = zeros(1,4);

for i = 2:1:5
    mean_wave3(i-1) = mean(wp_wave3(:,i));
    std_wave3(i-1) = std(wp_wave3(:,i));
    var_wave3(i-1) = var(wp_wave3(:,i));
end

%   Evolution of wave 4
```

```
417 mean_wave4 = zeros(1,4);
418 std_wave4 = zeros(1,4);
419 var_wave4 = zeros(1,4);
420
421 for i = 2:1:5
422     mean_wave4(i-1) = mean(wp_wave4(:,i));
423     std_wave4(i-1) = std(wp_wave4(:,i));
424     var_wave4(i-1) = var(wp_wave4(:,i));
425 end
426
427 %   Range of chosen wave probe data files
428
429 bounds_wave3 = zeros(3,4);  %   Matrix description:
430 bounds_wave4 = zeros(3,4);  %   Row 1: Range, Row 2: Min, Row 3: Max
431                             %   With columns describing which WP (1-4)
432 for i = 1:1:4
433     bounds_wave3(1,i) = range(wp_wave3(:,i+1));
434     bounds_wave4(1,i) = range(wp_wave4(:,i+1));
435     [bounds_wave3(2,i), bounds_wave3(3,i)] = bounds(wp_wave3(:,i+1));
436     [bounds_wave4(2,i), bounds_wave4(3,i)] = bounds(wp_wave4(:,i+1));
437 end
```

D.9 Subscript: irregular_analysis.m

```
%%  Irregular Analysis

irr = import_probe_data('points_probe_irr.dat',16,inf);          % used IWG
irr2 = import_probe_data('points_probe_irr_wg.dat',16,inf);      % used WG

%   IWG simulation
figure(701)
plot(irr(:,1),irr(:,3))
grid on
grid minor
set (gcf,'position',[10,10,900,300])
xlim ([5 500])
ylim ([-0.1 0.1])
xlabel 'Time [s]'
ylabel 'Wave elevation [m]'
saveas(figure(701),'Figures/Irregular/irr_wp2.png')

%   WG simulation - crashed after approx 100 seconds
figure(702)
plot(irr2(:,1),irr2(:,3))
grid on
grid minor
set (gcf,'position',[10,10,900,300])
xlim ([10 100])
ylim ([-0.1 0.1])
xlabel 'Time [s]'
ylabel 'Wave elevation [m]'
saveas(figure(702),'Figures/Irregular/irr2_wp2.png')

%   IWG detailed for poster
figure(708)
plot(irr(:,1),irr(:,3),'LineWidth',2)
grid on
grid minor
xlim ([170 200])
ylim ([-0.08 0.08])
xlabel 'Time [s]'
ylabel 'Wave elevation [m]'
saveas(figure(708),'Figures/Irregular/irr_wp2_poster.png')

%   Plot of WP 3 (irr)
%{
figure(703)
plot(irr(:,1),irr(:,4))
grid on
set (gcf,'position',[10,10,900,300])
title('wp3')
xlim([0 500])
xlabel 'Time [s]'
ylabel 'Wave elevation [m]'
saveas(figure(703),'Figures/irr_wp3.png')
%}

%   Plot of WP 4 (irr)
%{
figure(704)
```

```
plot(irr(:,1),irr(:,5))
grid on
set (gcf,'position',[10,10,900,300])
title('wp4')
xlim([0 500])
xlabel 'Time [s]'
ylabel 'Wave elevation [m]'
saveas(figure(704),'Figures/irr_wp4.png')
%}

% Stoppet etter 220.291 time steps = 2202 sek = 36.7 min

%% Wave Spectrums
irr_wp1 = dat2spec([irr(200:50000,1) irr(200:50000,2)]);
irr_wp2 = dat2spec([irr(400:50000,1) irr(400:50000,3)]);
irr_wp3 = dat2spec([irr(600:50000,1) irr(600:50000,4)]);
irr_wp4 = dat2spec([irr(800:50000,1) irr(800:50000,5)]);
irr_wp5 = dat2spec([irr(1000:50000,1) irr(1000:50000,6)]);

irr2_wp1 = dat2spec([irr2(1000:10000,1) irr2(1000:10000,2)]);
irr2_wp2 = dat2spec([irr2(1000:10000,1) irr2(1000:10000,3)]);
irr2_wp3 = dat2spec([irr2(1000:10000,1) irr2(1000:10000,4)]);
irr2_wp4 = dat2spec([irr2(1000:10000,1) irr2(1000:10000,5)]);
irr2_wp5 = dat2spec([irr2(1000:10000,1) irr2(1000:10000,6)]);

%   Defining variables for the JONSWAP spectrum
%   JONSWAP = ITTC for gamma = 1
Ohm = linspace(0,33,10000);
Hs = 0.1;
Tp = 2;

[S, Amp, Phase] = JONSWAP( Ohm, Hs, Tp);

%   Plot the calculated spectrums at WP 1-4 against theoretical JONSWAP
figure(705)
plot(irr_wp1.w,irr_wp1.S,...
    irr_wp2.w,irr_wp2.S,...
    irr_wp3.w,irr_wp3.S,...
    irr_wp4.w,irr_wp4.S,...
    Ohm,S,':','LineWidth',2)
xlabel  'Frequency [rad/s]'
ylabel  'Spectral density [(Nm)^2/rad]'
legend ('Wave Probe 1','Wave Probe 2','Wave Probe 3','Wave Probe 4','Theoretical
    ITTC')
ylim ([0 2e-4])
xlim ([0 16])
grid on
grid minor
saveas(figure(705),'Figures/Irregular/irr_spec.png')

%   Plot the calculated spectrums at WP 1-4 against theoretical JONSWAP
figure(706)
plot(irr2_wp1.w,irr2_wp1.S,...
    irr2_wp2.w,irr2_wp2.S,...
    irr2_wp3.w,irr2_wp3.S,...
    irr2_wp4.w,irr2_wp4.S,...
    Ohm,S,':','LineWidth',2)
xlabel  'Frequency [rad/s]'
ylabel  'Spectral density [(Nm)^2/rad]'
```

```
legend ('Wave Probe 1','Wave Probe 2','Wave Probe 3','Wave Probe 4','Theoretical
    ITTC')
ylim ([0 2e-4])
xlim ([0 16])
grid on
grid minor
saveas(figure(706),'Figures/Irregular/irr2_spec.png')

%   Plot for master exhibition poster
figure(707)
plot(irr_wp2.w,irr_wp2.S,irr2_wp2.w,irr2_wp2.S,Ohm,S,'LineWidth',2)
xlabel  'Frequency [rad/s]'
ylabel  'Spectral density [(Nm)^2/rad]'
legend ('Internal Wave Generator','Wave Generator','Theoretical ITTC')
ylim ([0 2e-4])
xlim ([0 16])
grid on
grid minor
saveas(figure(707),'Figures/Irregular/irr_spec_poster.png')

%%  Damping Zone Spectrums and Calculations

figure(710)
plot(irr_wp5.w,irr_wp5.S,irr2_wp5.w,irr2_wp5.S,'LineWidth',2)
xlabel  'Frequency [rad/s]'
ylabel  'Spectral density [(Nm)^2/rad]'
legend ('Wave Probe 5 (IWG)','Wave Probe 5 (WG)')
xlim ([0 6])
grid on
grid minor
saveas(figure(710),'Figures/Irregular/irr_wp5_spec.png')

figure(711)
plot(irr(:,1),irr(:,6)*1000,...
    irr2(1:10000,1),irr2(1:10000,6)*1000,'LineWidth',2)
grid on
grid minor
legend ('Wave Probe 5 (IWG)','Wave Probe 5 (WG)')
xlim ([5 500])
xlabel 'Time [s]'
ylabel 'Wave elevation [mm]'
saveas(figure(711),'Figures/Irregular/irr_wp5.png')

mean_wp4_IWG = mean(irr(:,6))*1000;      % [mm]
mean_wp4_WG = mean(irr2(1:10000,6))*1000;      % [mm]

%%  Statistics Calculated From Time Series

%   Irregular analysis 1
[irr_m,irr_mtext] = spec2mom(irr_wp2);
irr_Hs = 4*sqrt(irr_m(1));
irr_std = sqrt(irr_m(1));

%   Irregular analysis 2
[irr2_m,irr2_mtext] = spec2mom(irr2_wp2);
irr2_Hs = 4*sqrt(irr2_m(1));
irr2_std = sqrt(irr2_m(1));

%   Total energy
%   Calculated by integrating the spectrum with the trapezoid method
```

```
irr_tot_energy = trapz(irr_wp2.S);                  %   2050 sampling points
irr2_tot_energy = trapz(irr2_wp2.S);                %   2050 sampling points
theoretical_tot_energy = trapz(S(2:5:10000));       %   2000 sampling points

%   Connection between PM and ITTC
    %   Independent variables
    g = 9.81;     % [m/s]

    %   Irregular computation 1
    m0 = irr_m(1);
    m2 = irr_m(2);
    sd = sqrt(m0);
    wz = sqrt(m2/m0);
    k(1) = (sqrt(g/sd)) / (3.54*wz);
    w0(1) = 0.710*wz;   % Relation to PM
    T0(1) = (2*pi)/w0(1);
    Hs(1) = 4*sqrt(m0);

    %   Irregular computation 2
    m0 = irr2_m(1);
    m2 = irr2_m(2);
    sd = sqrt(m0);
    wz = sqrt(m2/m0);
    k(2) = (sqrt(g/sd)) / (3.54*wz);
    w0(2) = 0.710*wz;   % Relation to PM
    T0(2) = (2*pi)/w0(2);
    Hs(2) = 4*sqrt(m0);

function [ S, Amp, Phase ] = JONSWAP( Ohm, Hs, Tp)
% JONSWAP - Calculates the wave spectrum values for a JONSWAP spectrum
% Taken from MathWorks

wp = 2*pi/Tp;
Gamma = 1;
 for x = 1:length(Ohm)
     if Ohm(x)<wp
         Sigma = 0.07;
     else
         Sigma = 0.09;
     end
     A = exp(-((Ohm(x)/wp-1)/(Sigma*sqrt(2)))^2);
     S(x) = 320*Hs^2*Ohm(x)^-5/Tp^4*exp(-1950*Ohm(x)^-4/Tp^4)*Gamma^A;
 end

 % Determine the frequency step from the frequency vector. Note that the
 % highest frequency step is extrapolated.
 domg = zeros( size(Ohm) );
 domg(1:end-1) = diff( Ohm );
 domg(end) = domg(end-1);

 % Determine the amplitudes from the spectral values
 Amp = sqrt( 2 * S .* domg );

 % Random phases
 Phase = rand(1,length(Ohm))*2*pi;

end
```

D.10 Subscript: regular_waves_3D.m

```
%   3D Regular Wave Analysis

%%   Import Data
import_wp_3D    %   Import wave probe data for 3D computation
                %   C1: time, C2: WP1, C3: WP2
iso_3D = import_isoline('isoline_3D_Reg.dat',4,inf);

wp_wave3 = cell2mat(wp_wave(3));

%%  Theoretical Wave Statistics
t = 0:0.0025:80;
sine_wave = 0.05*sin(pi*t-pi);

%%  Wave Elevation

%   Free surface elevation 1 wavelength downstrem the wavetank
%   3D (IWG), 2D (IWG) and theory with equal wave stats. in the same plot
figure(901)
plot(wp_wave3(:,1),wp_wave3(:,2),...  %   -1 due to phase difference
    wp_3D(:,1),wp_3D(:,2),...
    t,sine_wave,'LineWidth',2)
legend('2D','3D','Theoretical Wave')
grid on
grid minor
set(gcf,'position',[10,10,900,300])
xlim([72 78])
xlabel 'Time [s]'
ylabel 'Wave elevation [m]'
saveas(figure(901),'Figures/Regular/wave3D_wp1.png')

%   Free surface elevation in the damping zone
%   3D (IWG), 2D (IWG)
figure(902)
plot(wp_wave3(:,1)-1,wp_wave3(:,6)*1000,...
    wp_3D(:,1),wp_3D(:,3)*1000,'LineWidth',2)
legend('2D','3D')
grid on
grid minor
xlim ([0 80])
xlabel 'Time [s]'
ylabel 'Wave elevation [mm]'
saveas(figure(902),'Figures/Regular/wave3D_dz.png')

%%  Wave Spectrums

S_3D_wp1 = dat2spec([wp_3D(800:8000,1) wp_3D(800:8000,2)]);
S_3D_dz = dat2spec([wp_3D(800:8000,1) wp_3D(800:8000,3)]);

%   Wave probe 1
figure(903)
plot(S_3D_wp1.w,S_3D_wp1(:,1).S,...
    S_wave3_wp1.w,S_wave3_wp1(:,1).S,'LineWidth',2)
legend('3D Wave Probe 1','2D Wave Probe 1')
grid on
grid minor
xlim([0 10])
```

```
xlabel  'Frequency [rad/s]'
ylabel  'Spectral density [(Nm)^2/rad]'
saveas(figure(903),'Figures/Regular/wave3D_spec.png')

%   Wave probe damping zone
figure(904)
plot(S_3D_dz.w,S_3D_dz(:,1).S,'LineWidth',2)
legend('Damping Zone')
grid on
grid minor
xlim([0 10])
xlabel  'Frequency [rad/s]'
ylabel  'Spectral density [(Nm)^2/rad]'
saveas(figure(904),'Figures/Regular/wave3D_spec_dz.png')

%%  Calculation of Wave Statistics

%   Peak frequency and period
[pks_3D locs_3D] = findpeaks(S_3D_wp1.S,S_3D_wp1.w);
Wp_3D = 1/(2*pi/locs_3D(1));
Tp_3D = 1/Wp_3D;

%   Mean, Std, Var, Bounds and Range
mean_3D = mean(wp_3D(:,2));
std_3D = std(wp_3D(:,2));
var_3D = var(wp_3D(:,2));
[lb_3D, ub_3D] = bounds(wp_3D(:,2));
range_3D = range(wp_3D(:,2));
```

Mari Tåranrød Storsul

A CFD Study of Numerical Wave Tanks

NTNU
Kunnskap for en bedre verden

SINTEF