



university of
 groningen

faculty of science
 and engineering

The validation of wave overtopping over complex dikes using CFD

Witteveen

Bos

Master Project Applied Mathematics

July 2020

Student: S.I.O Scholte

First supervisor: Prof. Dr. Ir. R.W.C.P. Verstappen

Second supervisor: Dr. Ir. R. Luppens

External supervisor: Dr. Ir. A.C. de Niet

Abstract

Delta flume experiments are essential to confirm the safety of a dike. Building the design of the dike and making changes to it can be a slow and expensive process. The aim of this study was to investigate whether CFD can be used to assess how safe a structure is. This was done by comparing overtopping data of both CFD and delta flume experiments. From the mathematical notions on waves, structures and CFD, models have been setup in the software waves2Foam. Data of delta flume experiments were then used to validate and calibrate the models. However, focusing purely on the overtopping, results showed that CFD continuously produced a larger overtopping level than the physical experiments. The data displayed a development of high asymmetric waves in the CFD simulations which were assumed to be the main contributor to the high overtopping values. Although attempts for solving this problem failed, CFD did manage to show that some interesting conclusions and lessons were taken out of this research. Scaling and grid sizes did not result in different overtopping results and it showed that CFD was often in line with the reality regarding changes to waves and designs of structures.

Acknowledgements

This Master thesis marks the end of my study Applied Mathematics at the University of Groningen. I am thankful for the opportunities that I have gotten to grow in the field of Applied Mathematics and to grow as a person. It is strange to finish my Master's degree during a time where the university is not accessible due to the Corona virus pandemic. Nevertheless, I am grateful for the help that I've gotten from all the people at the University and Witteveen+Bos during my thesis.

At first, I would like to thank my first supervisor Roel Verstappen for his guidance during this project. Although it was not always easy to see each other during this strange time, the meetings were always helpful and gave new insights on how to proceed. He was also my supervisor during my Bachelor's thesis and internship and the collaboration was always good.

Secondly, my gratitude goes out to Arie de Niet, my supervisor from Witteveen+Bos, where I did my internship and Master's project. He offered always a helping hand and was often available for discussing new ideas or results. The regular meetings were helpful to reflect on the thesis. I would also thank him for his efforts in finding a thesis subject at Witteveen+Bos.

Next, many thanks go out to Coen Kuiper from Witteveen+Bos, who proposed the subject of the thesis. With his enthusiasm and positive feedback he added a lot of expertise in coastal engineering to the project. Together with Wim Ridderinkhof from Witteveen+Bos, they were my main contacts when I had specific questions about waves and coastal structures. Their knowledge helped a lot and I would thank both for their assistance.

I would also like to thank my second supervisor Roel Luppens for supervising my thesis. He always showed his willingness to help and gave a lot of valuable feedback.

At last, I want to thank my parents for their lifelong support. The ride towards this moment was not always easy, but I hope that I have made them proud.

List of Tables

2.1	Types of breaking waves.	7
2.2	Types of foreshores.	9
3.1	Input values for the Delta Flume experiment together with the measured overtopping.	24
5.1	The CFD modelling approach.	32
6.1	The boundary conditions of each variable at the different boundaries in the reference model.	34
6.2	Input parameters for the incoming waves.	38
8.1	Number of cells in each direction in the three different mesh sizes after using the <i>blockMesh</i> utility.	44
9.1	The wave and roughness parameters of the first model to imitate the conditions of the Delta Flume experiment.	53
9.2	Location of the gauges.	53
10.1	The wave and roughness parameters of the additional roughness models to see their effects on the overtopping.	56
10.2	The resulting overtopping of the CFD model with a lower sea water level.	59
10.3	Location of the gauges in the longer domain.	61
10.4	The resulting overtopping of the CFD model with the longer domain before the dike on the lower slope. Instead of $30m$, the length from the inflow boundary to the toe of the dike is $60m$	62
10.5	The resulting overtopping of the CFD model with a lower bottom of the domain, such that the water is deeper.	64
10.6	The resulting overtopping of the CFD model with the block profile on the lower slope.	66
10.7	The resulting overtopping of the CFD models with the steeper slope and the flatter slope. The toe of the lower slope is translated $\pm 0.5m$ in the y-direction.	67
10.8	The resulting overtopping of the CFD model on Delta Flume scale (1:2.95).	68
10.9	The input parameters and the resulting overtopping of the CFD models in this chapter. *The overtopping of this model is projected to prototype scale.	68
11.1	The resulting overtopping of the CFD model with ribs on the upper slope.	70

List of Figures

1.1	The new design of the Afsluitdijk.	1
2.1	Comparison of the JONSWAP and Pierson-Moskovitz spectra [1].	6
2.2	Definitions of some of the important parameters regarding waves and structures [13].	8
2.3	An example of a control volume, where the dashed lines are the boundaries of each control volume.	17
3.1	The Afsluitdijk, with the Wadden Sea on the right and the IJsselmeer on the left.	21
3.2	The profile of dike section 8a of the Afsluitdijk, seen from the side.	22
3.3	The Levvel-blocs on the lower part of the slope.	22
3.4	The upper slope seen from the side.	23
3.5	The upper slope seen from above.	23
3.6	The area behind the dike where the water that went over the dike is stored, such that the amount of the overtopping can be assessed.	25
4.1	An example of a domain (red area) where the overtopping is calculated by means of integration of the variable α	29
6.1	The geometry made with SALOME that is implemented in the OpenFOAM environment.	33
6.2	Initial blockmesh before the dike is excluded from the domain. The dimensions are 3.3m x 110m x 16m. It has 6 cells in the x-direction, 220 cells in the y-direction and 32 cells in the z-direction.	36
6.3	Complete mesh created by the snappyHexMesh tool of OpenFOAM.	36
6.4	The boundary layers around the dike.	37
6.5	The relaxation zone where the waves are generated. The blue plane resembles the sea level and the red part is the relaxation zone.	38
7.1	Surface elevation of the water at $y = 0\text{m}$	40
7.2	Surface elevation of the water at $y = 7.88\text{m}$	41
7.3	Relative overtopping of the turbulence models.	42
7.4	Cumulative overtopping of the turbulence models.	42
8.1	Surface elevation of the water at $y = 7.88\text{m}$ (Toe of lower slope) of the three different mesh sizes.	44
8.2	Total overtopping of the water of the three different mesh sizes.	45

8.3	The total overtopping of the simulations with $Co = 0.5$ and $Co = 1$. The grey area is a 10% deviation of the initial simulation with $Co = 1$	46
9.1	Estimation of the JONSWAP spectrum of the Delta Flume experiment.	49
9.2	A single XblocPlus (Levvel-bloc). They cover the entire lower talud of the dike.	50
9.3	side-view of the subdivision of the original blockmesh into 6 smaller meshes to define the porosity in the lower middle block.	52
9.4	The wave spectra measured at different points in the domain. a) Wave spectrum at inflow boundary ($y = -30$). b) Wave spectrum at $y = -20$. c) Wave spectrum at $y = -10$. d) Wave spectrum at the start of the dike structure ($y = 0$).	54
9.5	Overtopping of the first CFD simulation together with the overtopping of the Delta Flume experiment.	55
10.1	Overtopping of the models where the parameter K_s is increased on the slopes and the berm.	57
10.2	Comparison of the wave height from the CFD model and the Delta Flume experiment at a) $y = -30$ and b) $y = 0$	58
10.3	The water level at the different waves gauges in the domain with initial sea water level of $5.3m$	59
10.4	The water level at the different waves gauges in the domain with original sea level of $5m$	60
10.5	Overtopping of the model with the lower SWL, together with previous CFD models.	60
10.6	Water level at the different gauges in the domain of the CFD model with the longer domain before the dike.	61
10.7	Elevation of the water surface at the toe of the dike in the time interval (400s, 600s) of the CFD model with the longer domain before the dike. It is compared with the elevation of the flume experiment ('Schot').	62
10.8	Comparison of the wave height in the 8a_long model with and without dike structure at a) gauge 1, b) gauge 2, c) gauge 3 and d) gauge 4.	63
10.9	Sea water level of the 8a_long model without dike.	63
10.10	Sea water level of the 8a_deep model.	64
10.11	Comparison of the wave height from the CFD model with deep water and the Delta Flume experiment at a) $y = -30$ and b) $y = 0$	65
10.12	The pattern of the levvel-blocks is imitated with a 'stairs' profile.	65

10.13	The turbulence, visualized by the variable ω , that forms a layer onto the lower slope.	67
11.1	Top view of the ribs after meshing.	69
11.2	A cross-section of the mesh of the upper slope with the ribs.	70

List of Symbols

Symbol	Unit	Description
a	$[m]$	Amplitude of the wave
C_m	$[-]$	Mass coefficient
C_s	$[-]$	Roughness factor
Co	$[-]$	Courant number
d_{50}	$[m]$	Nominal diameter of the grain material
f	$[Hz]$	Frequency of the wave
f_p	$[Hz]$	Peak frequency of the wave
F	$[m]$	Distance of the fetch
F_p	$[kg\ m/s^2]$	Resistant force
g	$[m/s^2]$	Gravitational acceleration
h	$[m]$	Depth of the sea
h_{prot}	$[m]$	Protrusion height of the roughness elements
$H_{1/3}$	$[m]$	The average height of the highest one third of the waves
H_{m0}	$[m]$	Spectral wave height
H_s	$[m]$	Significant wave height
k	$[m^2/s^2]$	Turbulent kinetic energy
k	$[rad/m]$	Angular wave number
K_s	$[m]$	Roughness height
L_0	$[m]$	Wavelength at the toe of the structure
$L_{m-1,0}$	$[m]$	Spectral Wavelength at deep water
m_0	$[-]$	Zero-th moment of the spectral density
m_n	$[-]$	n-th moment of the spectral density
n_p	$[-]$	Porosity
q	$[m^3/s]$ per m	Overtopping discharge
p	$[kg/ms^2]$	Static pressure
p^*	$[kg/ms^2]$	Hydrostatic corrected pressure, defined as $p_{rgh} = p - \rho gh$
q	$[L/sperm]$	Overtopping
R_c	$[m]$	Crest board height relative to SWL
$R_{u2\%}$	$[m]$	Run up height of the waves that 2% of the waves exceed
$s'_{m-1,0}$	$[-]$	Local shallow water wave steepness
S	$[m^2 Hz^{-1}]$	Spectral density

SWL	$[m]$	Sea water level
t	$[s]$	Time
$T_{m-1,0}$	$[s]$	Spectral wave period
T_p	$[s]$	Peak period of the wave
u	$[m/s]$	Velocity
u_p	$[m/s]$	Porous velocity
u_τ	$[m/s]$	Shear velocity
U^+	$[m/s]$	Dimensionless velocity
y^+	$[m]$	Dimensionless distance between the boundary and adjacent cell center
α	$[^\circ]$	Angle of the slope of the talud
α	$[-]$	Phase of the fluid
γ_b	$[-]$	Influence factor for the berm
γ_f	$[-]$	Influence factor for roughness
$\gamma_{f,w}$	$[-]$	Total width of the exposed roughness elements per meter dike
γ_β	$[-]$	Influence factor for oblique wave attack
γ^*	$[-]$	Collection of all other influence factor
γ_p	$[-]$	Peak enhancement factor JONSWAP spectrum
ϵ	$[m^2/s^3]$	Kinetic energy dissipation rate
η	$[m]$	Surface elevation
κ	$[-]$	Von Kármán constant
μ	$[kg/m\ s]$	Dynamic viscosity
ν	$[m^2/s]$	Kinematic viscosity
ν_t	$[m^2/s]$	Turbulent viscosity
ν_w	$[m^2/s]$	Near wall viscosity
$\xi_{m-1,0}$	$[-]$	Breaker parameter
ρ	$[kg/m^3]$	Density
τ	$[kg/m\ s^2]$	Stress tensor
τ_ν	$[m^2/s^2]$	Reynolds tress tensor
φ	$[rad]$	Phase of the wave
ϕ	$[m^3/s]$	Flux
ϕ_α	$[m^3/s]$	Water flux
ϕ_ρ	$[Kg/s]$	Density flux
ω	$[Hz]$	Angular frequency
ω	$[s^{-1}]$	Rate of dissipation
ω_p	$[Hz]$	Peak angular frequency

Contents

1	Introduction	1
2	Preliminaries	4
2.1	Wave run-up and overtopping	4
2.1.1	Waves	4
2.1.2	Structures	8
2.1.3	Empirical research	9
2.2	Computational fluid dynamics	11
2.2.1	Free surface flow	12
2.2.2	Turbulence	13
2.2.3	Porous flow	15
2.2.4	Discretization	16
3	Experimental data	20
3.1	The Afsluitdijk	20
3.2	Dike profile	21
3.3	The delta flume experiments	24
4	Waves2Foam	26
4.1	Wave generation	26
4.1.1	Wave theory	26
4.1.2	Relaxation zones	28
4.2	Overtopping	28
4.2.1	Flux	28
4.2.2	Integration	30
4.3	Permeability	30
5	Modelling approach	31
6	Model 1: Reference model	33
6.1	Geometry	33
6.2	Initial and boundary conditions	33
6.3	Meshing	35
6.4	Wave input parameters	37
6.5	Results	39
7	Turbulence model	40

8	Mesh size	44
9	Model 3: Implementation of the Levvel-blocs	48
9.1	Wave input settings	48
9.2	Levvel-blocs	49
9.3	Results	53
10	Calibration	56
10.1	Roughness coefficient K_s	57
10.2	Wave parameters	57
10.3	Block profile	65
10.4	Slope sensitivity	66
10.5	Scale effects	67
10.6	Summary of the results	68
11	Ribs	69
11.1	Geometry	69
11.2	Results	70
12	Discussion	71
13	Conclusion	73
	References	74
A	Scripts	76
A.1	Spectrum	76
A.2	Overtopping	79
A.3	Overtopping data Delta Flume	81
A.4	Water level	83

1 Introduction

Allowable wave overtopping is one of the key parameters in the construction of coastal dikes. Too much overtopping leads to erosion of the land side of the dike, which will weaken the structure and eventually cause failure with possibly dramatic consequences. Therefore in the design phase of the dike the overtopping needs to be predicted accurately under different circumstances in order to determine the safety of the structure.

One can imagine that many parameters determine the amount of overtopping. The characteristics of the waves and the design of a dike structure play a big role in the actual overtopping. With help of model tests, both small scale and large scale, empirical formulas have been derived to calculate the wave overtopping. These empirical formulas are stated in the European Overtopping (EurOtop) manual. The uncertainty in wave overtopping calculations with these empirical formulas is quite large, because of the many parameters that are involved. Moreover these parameters are very location specific. Continuously improving CFD software might be able to compute overtopping more accurately than the empirical formulas. The computation of wave overtopping with CFD will be the main topic of the thesis.

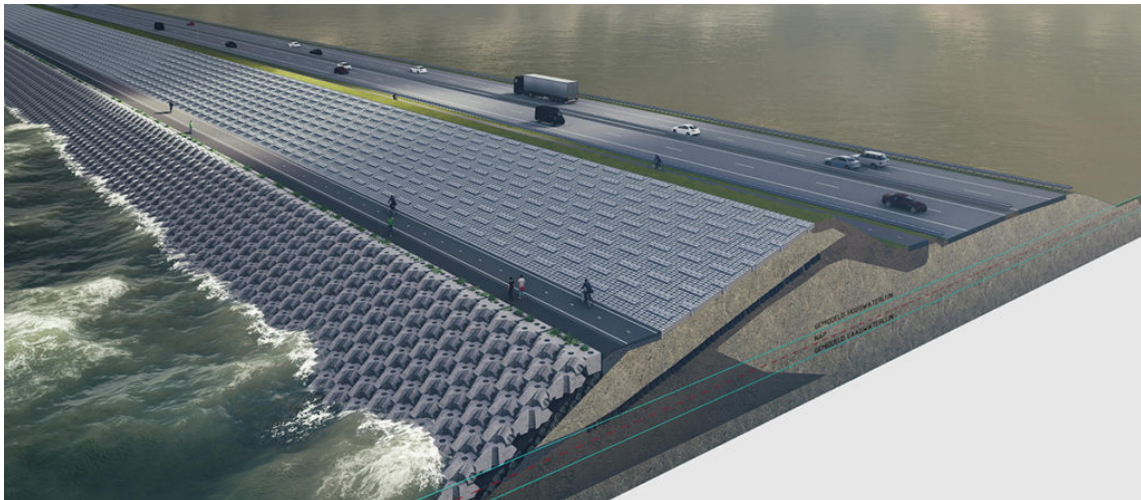


Figure 1.1: The new design of the Afsluitdijk.

The most famous coastal dike in the Netherlands is the Afsluitdijk. The dike is currently under renovation, as the present design fails to meet new safety standards. A new dike profile of the Afsluitdijk is tested in the largest Delta Flume in the Netherlands to verify the safety of its design¹. The flume is located in Delft and is courtesy to the company Deltares.

¹Source: www.rijkswaterstaat.nl

The data that is gathered with these tests is very useful to validate results of CFD models. The profile of the Afsluitdijk is quite complex, due to a combination of different types of slope, layers of armour protection elements and various roughness factors. The complexity of the dike makes the use of empirical formulas in assessing the safety of a design less suitable.

The main research question in this study is: Is it possible to calculate the amount of overtopping that occurs at the Afsluitdijk accurately with CFD? To answer this question, certain objectives are set along this research. Firstly, it is important to understand which parameters influence the amount of overtopping. Furthermore, a CFD model must be setup with the correct dike structure and wave parameters. At last, data of the CFD models and the Delta Flume experiments must be processed to compare them as good as possible.

The CFD software of choice is OpenFOAM. The main reasons to choose OpenFOAM is that there is experience with the software, it is open source and has been improved greatly over the past years. Most important, it is able to simulate a multiphase flow with a free surface and there exists an additional waves2Foam package that can generate and absorb many types of waves. The combination of OpenFOAM and waves2Foam has previously been used to assess the overtopping on dikes [9].

The thesis starts with a literature study on the hydraulic parameters that play a role in the actual overtopping (Ch. 2). This includes the characteristics belonging to a wave and the properties of the design of the dike. The second part of this chapter consists of the mathematical equations and tools that are needed in CFD to compute flows, like the Navier-Stokes equations and discretization methods.

The next Chapter (Ch. 3) consists of the practical information about the dike that will be investigated. The two Delta Flume experiments on this dike section are of great use to validate the CFD results. The only difference between the experiments is the design of the upper slope. In one experiment, the upper slope was smooth. The other experiment has ribs on the upper slope, as shown in Figure 1.1. The Chapter will dive deeper in the complexity of the dike and the data that is obtained from both experiments.

The last researching chapter is about the understanding and usage of the waves2Foam package (Ch. 4). The mathematical tools that are used specifically to generate the waves will be stated, as well as how waves are incorporated in a CFD model. Furthermore it will explain the mathematical background on different types of waves and how to calculate the overtopping.

With the practical and mathematical notions covered, the focus shifts to the modelling part of this thesis. The approach on which models will be conducted to eventually end up with

the model that imitates the Delta Flume experiment is briefly explained in Chapter 5. At first, a reference model will be set up to get a feeling of all the different steps that must be followed to get to a solution (Ch. 6). The steps include how to generate a good mesh with the correct dike profile in the geometry, the boundary conditions that are chosen and the parameters for the incoming waves. The reference model will incorporate a simplified form of the actual dike structure.

There are several important CFD properties which are tested in the next chapters. These are the choice of turbulence model (Ch. 7) and the independence of the grid size to the results (Ch. 8). The turbulence models tested are the $k - \epsilon$ and the $k - \omega$ SST model. After choosing one, different grid sizes are implemented to compare their results.

The final part of this research is to rebuild the dike, with the right wave parameters and roughness configurations (Ch. 9). It is a two-step process, where first the lower slope is recreated. The CFD model is then calibrated to the data of the experiment with a smooth upper slope in Chapter 10. Differences between the experiments and CFD are shown and discussed. Lastly, ribs on the upper slope are implemented in the model of Chapter 11. These ribs have a further reducing effect on the overtopping. The results will be compared to the other Delta Flume experiment.

2 Preliminaries

This chapter introduces the general definitions concerning waves and dikes that are needed in the research, as well as the CFD preliminaries for the computation of a free surface turbulent flow. The definitions will focus on the characteristics of waves and water structures that will partly determine the amount of run-up and overtopping a coastal structure can be subjected to. Moreover, the chapter presents an overview of empirical research on wave run-up and wave overtopping. This includes several formulas to calculate these two variables. The CFD section will consist of the governing equations that describe an incompressible free surface flow. The discretized equations are given and it is explained how turbulence is incorporated into the model.

2.1 Wave run-up and overtopping

The purpose of this section is to introduce the necessary hydraulic definitions that belong to a study on wave run-up and wave overtopping on coastal dikes. These definitions mainly come from the 2018 eurOtop manual [13], a manual based on European research which follows the latest results and conclusions on wave overtopping.

2.1.1 Waves

There are multiple wave properties that contribute to the amount of wave overtopping over a coastal structure. An overview of these properties is given in this paragraph.

Wave height

The first important wave parameter is the height of the wave. The wave height is often expressed as the significant wave height H_s [m]. There are multiple ways to define the significant wave height. The first definition is given by the spectral wave height:

$$H_s = H_{m0} = 4\sqrt{m_0}, \quad (1)$$

where m_0 is the zeroth moment of the spectral density of the wave. The n-th moment of the spectral density is calculated with the formula

$$m_n = \int_0^\infty f^n S(f) df, \quad (2)$$

with f [Hz] the frequency and $S(f)$ [$m^2 \text{ Hz}^{-1}$] the spectral density. The spectral density how the energy of the waves is distributed to the frequency in which the waves occur. The

interpretation of the zeroth moment of the spectral density is that it resembles the variance of the elevation of the free surface [5]. The spectral density will be discussed extensively later in this section.

The second definition for the significant wave height is the average of the highest one third of the waves, i.e. $H_s = H_{1/3}$. In many situations, these two approaches of deriving the wave height lead to similar values. In principal, the spectral wave height will be taken as H_s in this study, unless indicated otherwise.

Wave period

The next wave property is the wave period. There are several wave periods that are being used. The most common wave periods are the peak period and the spectral period. The peak period is defined by the peak frequency of the wave spectrum, calculated by $T_p = 1/f_p$ [s], where f_p is the peak frequency. By definition of the peak frequency,

$$S(f_p) \geq S(f)$$

for all possible frequencies in the domain of the spectrum.

The spectral period is calculated with the formula $T_{m-1,0} = m_{-1}/m_0$. When the peak period is not known, the relationship $T_p = 1.1 T_{m-1,0}$ is used. This relationship holds when the wave spectrum has a single peak.

Wave spectrum

Spectral wave height and period are obtained from an empirically determined spectrum that belong to different seas and oceans. This spectrum describes the relationship between the frequency of the waves and the wave spectral density. The spectrum that is used for the North Sea is the JONSWAP spectrum [5], which is an abbreviation for Joint North Sea Wave Observation Project. This is a so called single peak spectrum for which the previous relationship between the spectral and peak period is derived.

The JONSWAP spectrum formula is deduced from the Pierson-Moskowitz spectrum (1964) given in [17]. The spectrum was derived from the assumption that a fully developed sea was created by steady blowing winds for several days. Then, for several wind speeds they made a spectrum and derived that the spectra were of the form

$$S(\omega) = \frac{\alpha g^2}{\omega^5} e^{(-\beta(\frac{\omega_0}{\omega})^4)}. \quad (3)$$

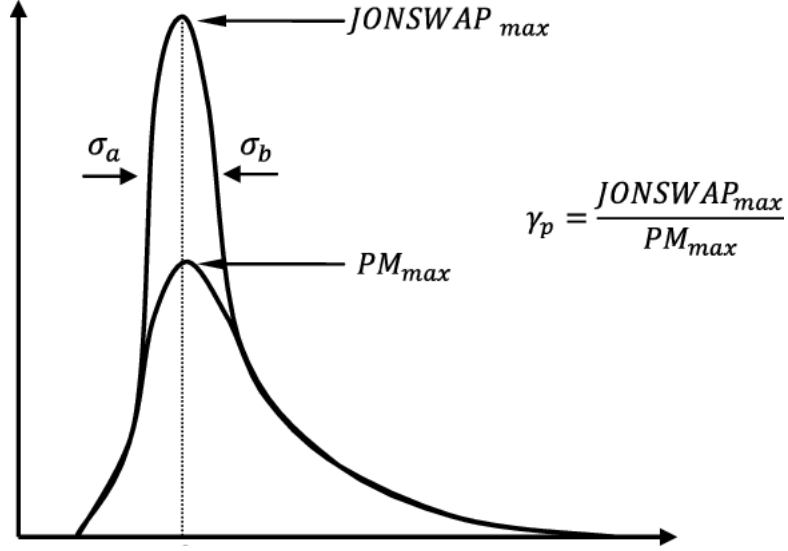


Figure 2.1: Comparison of the JONSWAP and Pierson-Moskovitz spectra [1].

Here, the spectral density $S(\omega)$ [m^2/Hz] is a function of the angular frequency $\omega = 2\pi f$ [Hz], the gravitational acceleration g [m/s^2] and the wind speed at a height of 19.5 meters above sea level $u_{19.5}$ [m/s], where $\omega_0 = g/u_{19.5}$. The measurement tools were located at this height in the research. The constants α and β are found to be 0.0081 and 0.74 respectively. The peak angular frequency ω_p was found by differentiating $S(\omega)$ with respect to ω and is equal to $\omega_p = 0.877 \frac{g}{U_{19.5}}$.

JONSWAP is based on another research on the data collected by Pierson and Moskowitz. The data fitted better with an amplification factor and therefore the relation between the JONSWAP and Pierson-Moskovitz spectra is a peak enhancement factor γ_p^r , with

$$r = e^{\left(-\frac{(\omega-\omega_p)^2}{2\sigma^2\omega_p^2}\right)},$$

see Figure 9.4. The value for σ is different on both sides of the peak. On the left, $\sigma = \sigma_a = 0.07$ and on the right, $\sigma = \sigma_b = 0.09$.

Different constant values were also found. They are

$$\alpha = 0.076 \left(\frac{u_{10}^2}{Fg}\right)^{0.22},$$

$$\gamma_p = 3.3,$$

$$\omega_p = 22 \left(\frac{g^2}{u_{10}F}\right)^{\frac{1}{3}},$$

with F [m] the distance of the fetch and u_{10} [m/s] the wind speed at a height of 10 meters

above sea level. The fetch is the area where the wind blows in a constant direction.

Wavelength

Another wave parameter that is important in the breaking of waves on a dike is the wavelength. In principle, two different types of wavelength are being used. These are the wavelength near the toe of the structure, L_0 [m], and the deep water spectral wavelength, $L_{m-1,0}$ [m]. The deep water spectral wavelength is given by

$$L_{m-1,0} = \frac{gT_{m-1,0}^2}{2\pi}. \quad (4)$$

The steepness of a wave [-] is defined as the ratio between the wave height and the wavelength, i.e.

$$s = \frac{H}{L}. \quad (5)$$

The calculated wave steepness depends mainly on which wavelength you choose. This can be deep water wavelengths like L_0 and $L_{m-1,0}$, but sometimes the wavelength in shallow water is used. The deep water wavelength is used in the formula for the breaker parameter $\xi_{m-1,0}$ [-]. The breaker parameter is used to describe whether a wave breaks and how it breaks on the structure. It is defined as

$$\xi_{m-1,0} = \frac{\tan \alpha}{\sqrt{s_{m-1,0}}}, \quad (6)$$

with α [°] the angle of the structure and $s_{m-1,0}$ corresponding to the wave steepness of the deep water spectral wavelength. The types of breaking can be characterized by the value of the breaker parameter and are shown in Table 2.1.

$\xi_{m-1,0}$	Type of breaking
$\xi_{m-1,0} < 0.2$	Spilling
$0.2 < \xi_{m-1,0} < 2$	Plunging
$\xi_{m-1,0} = \pm 2-3$	Collapsing
$\xi_{m-1,0} > 2-3$	Surging (no breaking)

Table 2.1: Types of breaking waves.

2.1.2 Structures

The water structure that is built to defend the land from flooding has many properties that will influence the amount of overtopping. Such a structure can have many designs and can become quite complex. The most common features of a coastal structure are discussed in this section.

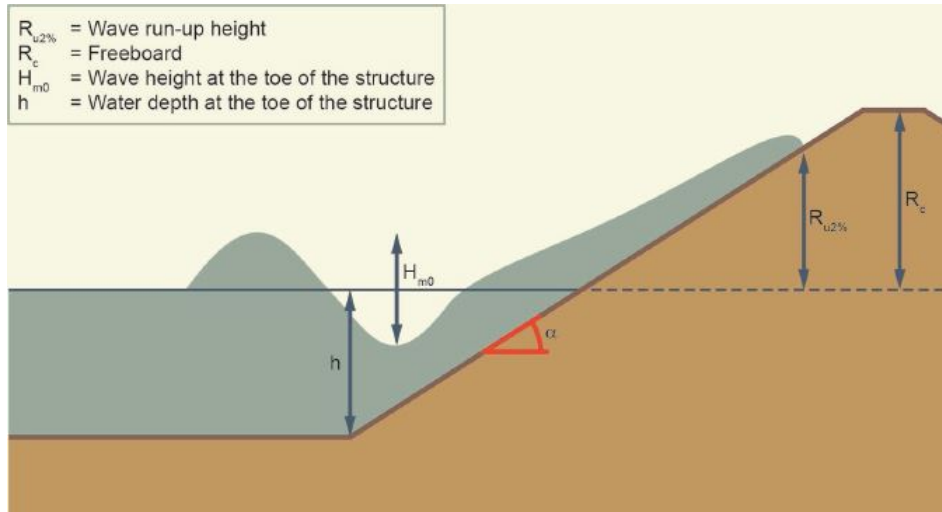


Figure 2.2: Definitions of some of the important parameters regarding waves and structures [13].

Foreshore

At first, the position of the structure is important. The toe of a structure is the position where the coastal structure begins. Some wave parameters are measured at the toe of the structures, such as the type of wavelength. The toe of the structure is the point where the foreshore meets the structure. The foreshore is defined as a gentle slope (a slope between a horizontal surface and a slope of 1:10) and has a minimum length of one wavelength.

A foreshore can be categorized in four different areas: deep, shallow, very shallow and extremely shallow. It depends on the relative depth of the foreshore, given by the ratio of the height of the sea water level (SWL) and the height of the wave in deep water: $h/H_{s,deep}$. The characterization of the four different foreshore areas is given in Table 2.2.

It is possible that a foreshore contains a multiple of those areas. The type of foreshore can have a significant influence on the wave height and therefore on the breaker parameter.

$h/H_{s,deep}$	Area
$h/H_{s,deep} < 0.3$	Extremely shallow
$0.3 < h/H_{s,deep} < 1$	Very shallow
$1 < h/H_{s,deep} < 4$	Shallow
$h/H_{s,deep} > 4$	Deep

Table 2.2: Types of foreshores.

Dike profile

The slope of the structure usually starts at the toe of the structure. This slope is steeper than the foreshore, usually between angles of 1:8 and 1:1. A structure may consist of multiple slopes. Roughness elements can be placed on the slopes to weaken the waves that run up the structure. Slopes could be (almost) horizontal. In that case, the slope is considered a berm. The purpose of the berm is often to create a bicycle path or side walk. Reducing effects from berms and roughness elements are captured with an influence factor γ [-], with $\gamma \in [0, 1]$. The influence factor for a berm is denoted by γ_b [-] and for roughness it is denoted by γ_f [-]. When an influence factor is equal to one, it is not present or it has no effect, e.g. because it lies under water. With γ^* [-] one denotes the product of the different influence factors.

The top of the structure is called the crest. From this point water can not flow back to the sea. The crest freeboard R_c [m] is the height of the crest relative to SWL. Overtopping discharge is usually calculated at the beginning of the crest.

With large-scale and small-scale tests in several delta flumes, research has been conducted about the relationship between the introduced parameters and the actual wave run-up and wave overtopping discharge. The most important results will be discussed in the next session.

2.1.3 Empirical research

Wave run-up height is the height that a wave reaches on a slope. In coastal engineering it is common to work with the wave run-up height that 2 percent of the breaking waves that are incoming at the toe of the structure [13] exceed. It is denoted by $R_{u2\%}$ [m]. This percentage is chosen arbitrary a long time ago and is still used to define the wave run-up level. Wave run-up is dependent on different parameters. These are mainly the wave height H_s , the breaker parameter $\xi_{m-1,0}$ and the influence factors γ^* . The basic formula that is derived empirically for wave run-up at relative gentle single slopes is

$$R_{u2\%} = 1.65 \cdot \xi_{m-1,0} \cdot \gamma^* \cdot H_s. \quad (7)$$

This is due to the relative small value of the breaker parameter. With steeper or more complex slopes, waves can surge and this effect on the wave run-up is found not to be linear. The formula for non-breaking waves by the eurOtop manual is

$$R_{u2\%} = \gamma_f \cdot \gamma_\beta \cdot H_s \cdot \left(4 - \frac{1.5}{\sqrt{\gamma_b \cdot \xi_{m-1,0}}}\right), \quad (8)$$

where γ_β [-] is the influence factor for oblique wave attack. Oblique wave attack has to do with the angle that a wave has collapsed on a dike, where a wave attack that is perpendicular to the dike has an angle of zero. In many studies, this study included, it is assumed that the effects of an oblique (not perpendicular) wave attack can be neglected, but it can influence the amount of overtopping [13].

A different study by Capel [3] derived another formula for the wave run-up level. This study was mainly about the influence factor of roughness on impermeable slopes and therefore a formula have been derived with only the roughness as an influence factor in it, although it was noted that the formula could be extended with other influence factors. The formula derived by Capel is

$$R_{u2\%} = 3.45 \cdot \tanh(0.65 \xi_{m-1,0}) \cdot \gamma_f \cdot H_s, \quad (9)$$

which showed a better fitting to the available data of the flume tests than the previous formulas of the eurOtop manual.

Wave overtopping is the discharge rate of a wave over the crest of the coastal structure. This discharge rate q is expressed in liter per second per meter dike [l/s m]. The discharge rate is often expressed in the dimensionless parameter $\frac{q}{\sqrt{g \cdot H_s^3}}$. The general formula of the wave overtopping has the form

$$\frac{q}{\sqrt{g \cdot H_s^3}} = a \cdot e^{-(b \frac{R_c}{H_s})^c}, \quad (10)$$

with a , b and c quantities that include wave or coastal parameters and influence factors. For average overtopping, eurOtop utilises the formula

$$\frac{q}{\sqrt{g \cdot H_s^3}} = \frac{0.023}{\sqrt{\tan \alpha}} \gamma_b \cdot \xi_{m-1,0} \cdot e^{-(2.7 \frac{R_c}{\xi_{m-1,0} \cdot H_s \cdot \gamma_b \cdot \gamma_f \cdot \gamma_\beta})^{1.3}}. \quad (11)$$

This formula is again for lower values of the breaker parameter, i.e. breaking waves.

In the research of the effects of roughness, Capel [3] also derived a new formula for the

average wave overtopping on impermeable slopes:

$$\frac{q}{\sqrt{g \cdot H_s^3}} = \frac{0.027}{\sqrt{\tan \alpha}} \cdot \xi_{m-1,0} \cdot e^{(-6.5 \frac{R_c}{R_{u2\%}})} \quad (12)$$

Because the present study is mainly about the influence factor roughness, which is incorporated in the wave run-up parameter in the formula of Capel, it is useful to state the formulas of the roughness influence factor that both eurOtop and Capel derived. Roughness is mostly applied by placing blocks on the slope of a dike in a particular setting. These characteristics come together in the parameter ρ_{γ_f} [-] and is called the roughness density. The formula for the roughness density is

$$\rho_{\gamma_f} = \frac{\gamma_{f,w} \cdot \sin \alpha \cdot h_{prot}}{R_c}, \quad (13)$$

with $\gamma_{f,w}$ [-] the total width of the exposed roughness elements per meter dike and h_{prot} [m] the protrusion height of the roughness elements. This parameter is present in the derivation of the roughness:

$$\gamma_{f,eurOtop} = 1 - [0.585 - \sqrt{(0.075 - s'_{m-1,0})\rho_{\gamma_f}} \cdot (-\ln \frac{q}{\sqrt{gH_s^3}})], \quad (14)$$

and

$$\gamma_{f,capel} = 1 - [0.585 - \sqrt{(0.075 - s'_{m-1,0})\rho_{\gamma_f}} \cdot (6.5 \frac{R_c}{R_{u2\%}} - \ln q_0)], \quad (15)$$

where $q_0 = \frac{0.027}{\sqrt{\tan \alpha}} \cdot \xi_{m-1,0}$.

To summarize, there are a lot of factors to take into account when calculating the overtopping. All this information helps to understand phenomena occurring in a CFD model. In the next section one sees how CFD simulates a flow and which mathematical equation play a role in the performance of a CFD model.

2.2 Computational fluid dynamics

The goal of this section is to recall the techniques that are used in Computational Fluid Dynamics (CFD) to simulate a free surface turbulent flow. This ranges from the equations that are needed to be solved and the discretization techniques that are used. All the methods described here are available in the OpenFOAM software that is used in this research.

2.2.1 Free surface flow

A free surface flow is characterized by computing two flows, often water and air, and the interaction between these two flows. The area of interaction is called the interface. The names free surface and interface can be used interchangeably. Capturing the location of the interface is the main challenge when it comes to simulating a free surface flow.

Any fluid flow is described by the Navier-Stokes equations. These equations have not been solved analytically, therefore they have to be solved numerically and this has given rise to a large field in numerical mathematics: CFD. The Navier-Stokes equations consist of an equation for the conservation of mass and an equation for the conservation of momentum. Those equations are given by

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (16)$$

and

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) = -\nabla p + \rho \mathbf{g} + \nabla \cdot \boldsymbol{\tau} \quad (17)$$

respectively. Here is ρ the density [kg/m^3], \mathbf{u} the velocity vector [m/s], t the time [s], p the pressure [$kg/m s^2$], \mathbf{g} the gravitational acceleration [m/s^2] and $\boldsymbol{\tau}$ the stress tensor [$kg/m s^2$].

Throughout the whole report, the flow is assumed to be incompressible. This implies that the density ρ of both fluids is considered constant. The assumption of an incompressible flow has some consequences for the above mentioned equations. At first, the conservation of mass equation simplifies to

$$\nabla \cdot \mathbf{u} = 0. \quad (18)$$

Moreover, the stress tensor $\boldsymbol{\tau}$ in the conservation of momentum equation is in the incompressible case given by the Stokes' stress constitutive equation

$$\boldsymbol{\tau} = \mu(\nabla \mathbf{u} + \nabla \mathbf{u}^T), \quad (19)$$

where μ [kg/ms] is the dynamic viscosity. This can be substituted in equation 17 to obtain

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) = -\nabla p + \rho \mathbf{g} + \mu \Delta \mathbf{u}. \quad (20)$$

An extra equation is needed to capture the interface between the two fluids. For this, a

variable α [-] is introduced that keeps track of the fluid. It is defined as

$$\alpha = \begin{cases} 0, & \text{if particle is in the other fluid} \\ 0 < \alpha < 1, & \text{at the interface} \\ 1, & \text{if particle is in the main fluid,} \end{cases} \quad (21)$$

where in this research, the main fluid is water and the other fluid is air. OpenFOAM uses a particular method to capture the interface between the fluids, which is called the Volume Of Fluid (VOF) method. The VOF method calculates the ratio of each cell filled with the liquid phase by solving

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{u}) = 0. \quad (22)$$

By solving for α , the density of the fluid is then simply calculated by

$$\rho = \alpha \rho_0 + (1 - \alpha) \rho_1, \quad (23)$$

where ρ_0, ρ_1 are the densities of the water and air respectively. This parameter is used to calculate the surface tension of the free surface.

Furthermore, OpenFOAM takes the hydrostatic pressure into account when solving for the pressure. This alternative pressure p^* is defined as

$$p^* = p - \rho g h,$$

where the hydrostatic part is subtracted from the static pressure. This relation is then used in equation 20 and cancels the ρg term to finally end up with

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) = -\nabla p^* + \mu \Delta \mathbf{u} \quad (24)$$

for the momentum equation.

2.2.2 Turbulence

The incorporation of turbulence is usually done with the Reynolds-Averaged Navier-Stokes (RANS) equations. These equations are obtained by splitting the variables in the instantaneous Navier-Stokes equations into a (time independent) mean component and a fluctuating component. For example the separation of the velocity is written as $u(x, t) = \bar{u}(x) + u'(x, t)$, where $\bar{u}(x)$ is the mean velocity and $u'(x, t)$ is the fluctuating velocity. Separation of all the

variables in the Navier-Stokes equations gives us

$$\nabla \cdot (\bar{\mathbf{u}} + \mathbf{u}') = 0, \quad (25)$$

and

$$\frac{\partial \rho(\bar{\mathbf{u}} + \mathbf{u}')}{\partial t} + \nabla \cdot (\rho(\bar{\mathbf{u}} + \mathbf{u}') \otimes (\bar{\mathbf{u}} + \mathbf{u}')) = -\nabla(p^* + p^{*'}) + \mu\Delta(\bar{\mathbf{u}} + \mathbf{u}'). \quad (26)$$

The next step is to take the mean over all terms and using the property $\bar{u}' = 0$. This yields

$$\nabla \cdot \bar{\mathbf{u}} = 0, \quad (27)$$

and

$$\frac{\partial \rho \bar{\mathbf{u}}}{\partial t} + \nabla \cdot (\rho \bar{\mathbf{u}} \otimes \bar{\mathbf{u}}) + \nabla \cdot (\overline{\rho \mathbf{u}' \otimes \mathbf{u}'}) = -\nabla \bar{p}^* + \mu \Delta \bar{\mathbf{u}}. \quad (28)$$

Notice that there is still a fluctuating term present in the momentum averaged equation due to the nonlinear convective term. Because these fluctuations are unknown, the problem arises how to approximate this term. This particular problem is called the closure problem. Luckily, there are several methods known that can deal with the nonlinear term to be able to solve the equations with only averaging terms. They are all based on the assumption that there exists an analogy between the molecular and turbulent viscosity. The nonlinear term $\overline{\mathbf{u}' \otimes \mathbf{u}'}$ is considered a stress term, also known as the Reynolds stress. With that assumption, the stress term can be expressed in a similar fashion as equation 19. Thus, the nonlinear term can be approximated as

$$\tau_\nu = -\overline{\mathbf{u}' \otimes \mathbf{u}'} \approx \nu_t(\nabla \bar{\mathbf{u}} + \nabla \bar{\mathbf{u}}^T), \quad (29)$$

where ν_t is the turbulent viscosity and τ_ν is the Reynolds stress. The methods to close the RANS equations are based on different ways to compute the turbulent viscosity. A few examples are the $k-\epsilon$ turbulence model and the $k-\omega$ turbulence model. These two methods are incorporated and commonly used in OpenFOAM.

The $k-\epsilon$ method uses two equations to calculate the turbulent viscosity that closes the RANS equations. The variable k resembles the turbulent kinetic energy and ϵ is the turbulent dissipation. The standard equations for k and ϵ formulated by Launder and Sharma [15] are given by

$$\frac{\partial k}{\partial t} + \nabla \cdot \bar{\mathbf{u}}k = \nu_t \tau_\nu - \epsilon + \Delta\left[\left(\frac{\nu + \nu_t}{\sigma_k}\right)k\right] \quad (30)$$

and

$$\frac{\partial \epsilon}{\partial t} + \nabla \cdot \bar{\mathbf{u}} \epsilon = \frac{\epsilon}{k} (\nabla \cdot C_{1\epsilon} \nu_t \tau_\nu \bar{\mathbf{u}} - C_{2\epsilon} \epsilon) + \Delta \left[\left(\frac{\nu + \nu_t}{\sigma_\epsilon} \right) \epsilon \right]. \quad (31)$$

The turbulent viscosity ν_t is then calculated with

$$\nu_t = C_\nu \frac{k^2}{\epsilon}. \quad (32)$$

The $k - \epsilon$ equations have five constant values that close the equations. The values of the five constants in the equations are $C_\nu = 0.09$, $C_{1\epsilon} = 1.44$, $C_{2\epsilon} = 1.92$, $\sigma_k = 1.0$ and $\sigma_\epsilon = 1.3$. The $k - \epsilon$ turbulence method is known to be very good at predicting the turbulence in the free stream of the flow, but behaves much worse near wall boundaries.

On the other hand, the $k - \omega$ turbulence model behaves in an opposite way. The model performs well near the wall, but has some weaknesses in predicting the turbulence in the free stream. The most used $k - \omega$ equations are derived by Wilcox [24], where ω is now the variable for the turbulent dissipation:

$$\frac{\partial k}{\partial t} + \nabla \cdot \bar{\mathbf{u}} k = \nabla \cdot \bar{\mathbf{u}} \tau_\nu - \beta^* k \omega + \Delta [(\nu + \sigma^* \nu_t) k] \quad (33)$$

and

$$\frac{\partial \omega}{\partial t} + \nabla \cdot \bar{\mathbf{u}} \omega = \nabla \cdot \alpha \frac{\omega}{k} \tau_\nu \bar{\mathbf{u}} - \beta \omega^2 + \Delta [(\nu + \sigma \nu_t) \omega], \quad (34)$$

with

$$\nu_t = \frac{k}{\omega}. \quad (35)$$

The coefficients that close the system are $\alpha = \frac{5}{9}$, $\beta = \frac{3}{40}$, $\beta^* = \frac{9}{100}$, $\sigma = 0.5$ and $\sigma^* = 0.5$. A combination of both methods is also available, called the $k - \omega$ Shear Stress Transport (SST) model. The exact equations and coefficients are explained in [16].

2.2.3 Porous flow

For permeable structures such as breakwaters, the momentum and continuity equations of the flow are modified as shown in [10]:

$$\nabla \cdot \mathbf{u} = 0, \quad (36)$$

$$\frac{1 + C_m}{n_p} \left(\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) \right) = -\nabla p^* + \frac{1}{n_p} \mu \Delta \mathbf{u} - \mathbf{F}_p. \quad (37)$$

here, the \mathbf{u} is now defined as the filter velocity, which relates to the porous velocity \mathbf{u}_p as

$$\mathbf{u} = n \mathbf{u}_p, \quad (38)$$

with n [-] the porosity. The variable \mathbf{F}_p [$kg\ m/s^2$] is the resistance force, which is given by

$$\mathbf{F}_p = a \rho \mathbf{u} + b \rho \mathbf{u} \|\mathbf{u}\|_2, \quad (39)$$

where a [-] and b [-] are resistance coefficients. The values of the coefficients depend on the chosen resistance formulations. At last, the mass coefficient C_m [-] appearing in equation 37 is defined as

$$C_m = \gamma_p \frac{1 - n}{n}. \quad (40)$$

The coefficient γ_p [-] is for the closure of the equations and has the value of 0.34.

2.2.4 Discretization

The existence and uniqueness of an analytical solution to the Navier-Stokes equations in three dimensions is still one of the unsolved Millennium Prize Problems. Discretization of the (Reynolds-Averaged) Navier-Stokes equations is therefore needed to solve the problem numerically. The spatial discretization method used is the finite volume method. This method divides the whole domain into control volumes around each grid point. Then the integral in each direction is taken and Gauss's theorem is applied to the divergence terms in the Navier-Stokes equations. In 3D, Gauss's theorem states that

$$\int_V (\nabla \cdot \mathbf{F}) dV = \int_{\partial V} (\mathbf{F} \cdot \mathbf{n}) d\partial V,$$

with V the control volume, ∂V the boundary of the volume and \mathbf{n} the normal direction pointed outwards of the control volume. When the finite volume method is applied to equation 24, one obtains

$$\int_V \frac{\partial \mathbf{u}}{\partial t} dV \int_V \nabla \cdot (\mathbf{u} \otimes \mathbf{u}) dV = \int_V -\nabla \frac{p^*}{\rho} dV + \int_V \nu \Delta \mathbf{u} dV, \quad (41)$$

where we divided by ρ and used that $\nu = \frac{\mu}{\rho}$, called the dynamic viscosity. Applying Gauss's theorem gives

$$\frac{\partial \mathbf{u}}{\partial t} V + \int_S (\mathbf{u} \otimes \mathbf{u}) \mathbf{n} dS = \int_S -\frac{p^*}{\rho} \mathbf{n} dS + \int_S \nu \nabla \mathbf{u} \mathbf{n} dS, \quad (42)$$

with $S = \partial V$ the surface of the control volume. From here, difference schemes are used to discretize the surface integrals. There are several methods to discretize in space, with the most common methods being the central and upwind discretization.

As an example, the different methods are explained with the two-dimensional control volumes as in Figure 2.3. The dashed lines are the faces of the control volumes. The solution ϕ of each control volume is stored in the center of each cell, i.e for the middle control volume in the point P. Here, ϕ could be any variable that is solved at each iteration such as the pressure or the velocity. To avoid confusion, the variables u and v in Figure 2.3 denote the flux of the flow, which is defined at the boundaries of each cell.

The discretized forms of ϕ , $\frac{d\phi}{dx}$, $\frac{d^2\phi}{dx^2}$ etc. follow from the Taylor expansion around the central points of the grid cells. The Taylor expansions for points E and W in Figure 2.3 are

$$\phi_E = \phi_P + \Delta x \frac{d\phi}{dx} \Big|_{x=x_p} + \frac{(\Delta x)^2}{2} \frac{d^2\phi}{dx^2} \Big|_{x=x_p} + h.o.t. \quad (43)$$

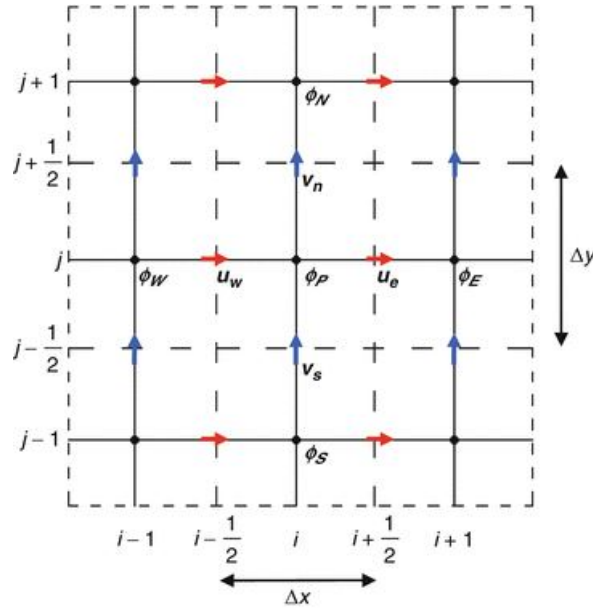


Figure 2.3: An example of a control volume, where the dashed lines are the boundaries of each control volume.

and

$$\phi_W = \phi_P - \Delta x \frac{d\phi}{dx} \Big|_{x=x_p} + \frac{(\Delta x)^2}{2} \frac{d^2\phi}{dx^2} \Big|_{x=x_p} + h.o.t., \quad (44)$$

with $\phi_P = \phi(x_P, y_P)$. The central discretization of the first derivative is obtained by subtracting equation 44 from 43:

$$\frac{d\phi}{dx} \Big|_{x=x_p} = \frac{\phi_E - \phi_W}{2\Delta x} + \mathcal{O}(h^2) \quad (45)$$

The same can be done for the y-direction. It will follow that

$$\frac{d\phi}{dy} \Big|_{y=y_p} = \frac{\phi_N - \phi_S}{2\Delta y} + \mathcal{O}(h^2). \quad (46)$$

The upwind method is obtained by rewriting the Taylor expansion. It can be used in a forward scheme, using equation 43, such that

$$\frac{d\phi}{dx} \Big|_{x=x_p} = \frac{\phi_E - \phi_P}{\Delta x} + \mathcal{O}(h), \quad (47)$$

or as a backward scheme, using equation 44

$$\frac{d\phi}{dx} \Big|_{x=x_p} = \frac{\phi_P - \phi_W}{\Delta x} + \mathcal{O}(h). \quad (48)$$

It is seen that the advantage of central discretization is achieving a higher order of accuracy. On the other hand, the upwind method is overall considered to be more stable. Central discretization can lead to so called wiggles [22] that blow up the simulation.

The second derivative can only be discretized in a central way, namely by adding equation 43 and 44. Rewriting this gives

$$\frac{d^2\phi}{dx^2} \Big|_{x=x_p} = \frac{\phi_E - 2\phi_P + \phi_W}{(\Delta x)^2} + \mathcal{O}(h^2). \quad (49)$$

Time integration is mostly done by an implicit integration scheme such as the backward Euler method. Again, taking equation 24 as an example, the term with the time derivative is expressed as a function of the solution at the next time iteration, i.e.

$$\frac{\mathbf{u}^{(n+1)} - \mathbf{u}^{(n)}}{\delta t} = -\nabla \cdot (\mathbf{u}^{(n+1)} \otimes \mathbf{u}^{(n+1)}) - \frac{1}{\rho} \nabla p^{*(n+1)} + \nu \Delta \mathbf{u}^{(n+1)}. \quad (50)$$

The advantage of an implicit scheme is that the solution is always stable, although solving it has a higher complexity than an explicit scheme such as the forward Euler method.

The discretized equations are ready to be solved. Multiple algorithms have been created over the past decades to optimize numerical qualities like stability and computation time. Two commonly used algorithms for coupling the discretized equations are the SIMPLE algorithm and the PISO-method [11]. A combination of the two lead to the PIMPLE method, which is explained in [7], which is used by OpenFOAM when computing a free surface flow.

3 Experimental data

The purpose of this section is to discuss the information regarding the dike section of the Afsluitdijk that has been experimented in the Delta Flume in Delft. This dike section will be modelled in the CFD calculations to eventually compare it with the experimental data.

Firstly, the section of the Afsluitdijk itself will be investigated. This starts with background information on the Afsluitdijk as well as the current renovation plan. The focus will then be on the dike section itself, where all the important geometrical parameters are explained. The dike section that plays the central role in the present study is dike section 8a, which is located on the Wadden Sea side. Together with the overall sketch of the dike profile, there is a deeper look into the overtopping reduction elements.

As already discussed, the design of a coastal structure is tested before it is actually built. Both small-scale tests as large-scale tests in delta flumes at several locations are performed for this purpose. One of the largest delta flumes is built in Delft and is courtesy of Deltares. In the Delta Flume, Deltares has imitated the conditions at sea and have recreated the profile of dike section 8a to obtain a lot of data with regard to wave speed, forces on the dike and the average overtopping. The parameters that Deltares has used, together with the resulting overtopping is described to use later on for the OpenFOAM simulation and its validation.

3.1 The Afsluitdijk

The Afsluitdijk is a 32 kilometers long dike in the Netherlands that separates the Wadden Sea and the IJsselmeer. It is built in 1932 and was part of the Zuiderzee Works, with as main goal to create extra land for agriculture. This led eventually to the rise of the province Flevoland. The dike is responsible for the fresh water in the IJsselmeer and it has a highway connecting the provinces Noord Holland and Friesland.

In 2017, new safety standards for water structures have been formed. The current Afsluitdijk fails to meet these new norms and therefore a renovation was needed. At the end of 2018, Rijkswaterstaat (In English: Directorate-General for Public Works and Water Management) of the Netherlands have begun with the renovation of the Afsluitdijk. Because the Afsluitdijk plays a pivotal role in the safety of the Netherlands, the dike must be resistant to the most severe conditions, even if the chances that these conditions occur are extremely small (once every ten thousand years). The main parts of the renovations are that the dike will be heightened, newly created overtopping reduction blocks will be placed along the whole Afsluitdijk, more sluices to pump water out of the IJsselmeer are constructed and a new migration river for fish is placed. The renovation is carried out by the consortium Level, a



Figure 3.1: The Afsluitdijk, with the Wadden Sea on the right and the IJsselmeer on the left.

group consisting of three contractors, and the process lasts approximately 4 years.

3.2 Dike profile

A dike is designed such that the amount of overtopping satisfies the norms to guarantee the safety of the dike. Additional requirements can be set, depending on the location of the dike and its purposes. The purposes of the Afsluitdijk mentioned in the previous paragraph led also to a quite complex dike profile with many ways to reduce the amount of overtopping. Also the aesthetics of the dike is very important to Rijkswaterstaat, where simplicity is highly valued.

A cross section of the whole profile dike section 8a is shown in Figure 3.2. The overtopping reduction parts of this dike are the lower slope, consisting of so called Levvel-blocs, the berm in the middle and the upper slope with a block revetment in a special pattern. These parts are the main wave reduction factors that must protect the dike and everything that is behind the crest from overtopping waves.

The lower part of the dike consists of special blocks that have been designed very recently. They are placed at the whole length of the Afsluitdijk. These blocks are called Levvel-blocs

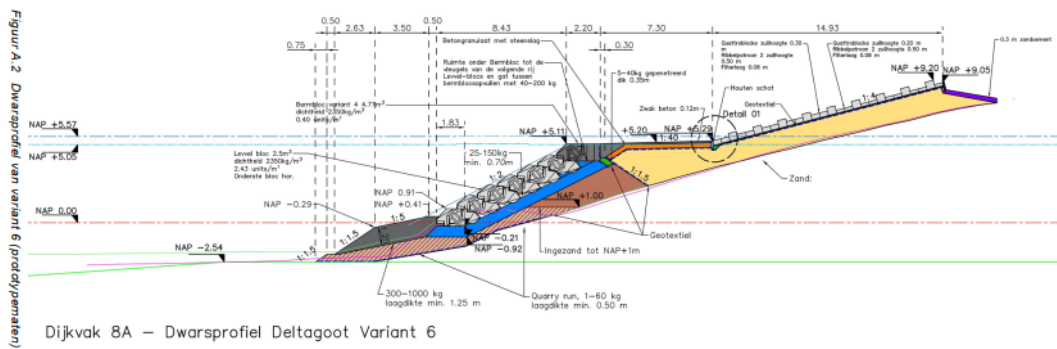


Figure 3.2: The profile of dike section 8a of the Afsluitdijk, seen from the side.

and their pattern can be seen in Figure 3.3. The block itself is 2.8 meter long, 2.3 meter wide and 1.13 meter high. Their particular shape makes it easy to overlap the blocks and this makes the construction easier. Each block weighs around sixty-five hundred kilograms. The blocks are named after the Level consortium. The type of block is called an XblocPlus and is designed to reduce the CO₂ emission and the use of concrete. Around seventy-five thousand Level-blocs will be placed on the Afsluitdijk. The mean slope of this part of the structure is around 1:2.

Although most of the waves will be damped by the Level-blocs on the lower talud, there are times that those blocks are not enough to prevent the waves from overtopping. The berm

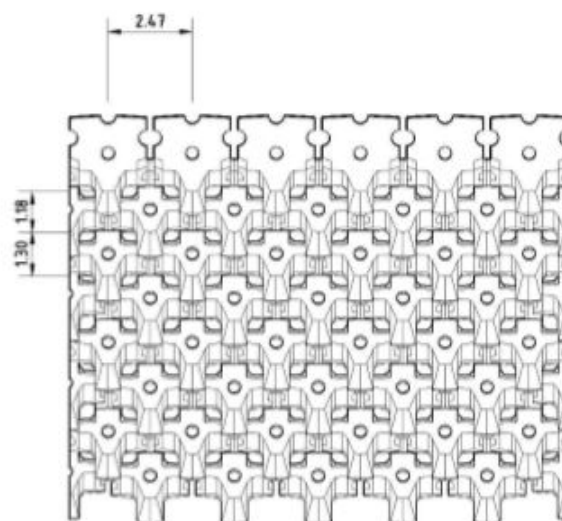


Figure 3.3: The Level-blocs on the lower part of the slope.

that is used as a bicycle path and side walk reduces the overtopping as well. The width of the berm is 9.5 meters which slopes upwards a little (1:40) for most part of the berm.

The upper part of the dike has a gentle slope (1:4) which consists of a block revetment with a certain pattern. The height of those blocks vary, as can be seen in Figure 3.4. The standard blocks on the first 15 rows lie 10 centimeters higher than the last 20 rows. The ribs that are on each third row are 50 cm high and there are 11 of those rows in total. Figure 3.5 shows that the ribs are 4 block elements wide with 2 standard blocks in between each rib.

These three regions together are there to make sure that even the most severe weather conditions possible will keep the total overtopping inside the new safety norms. These

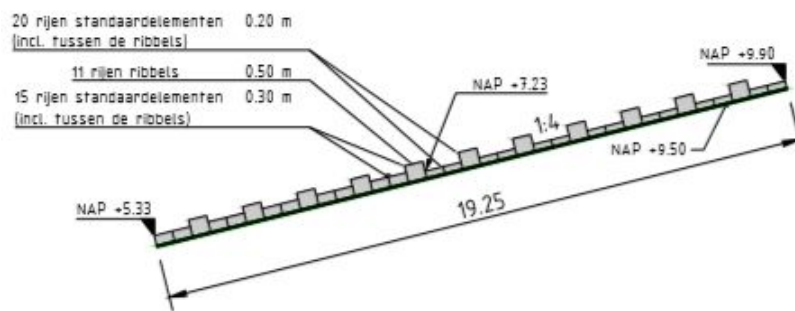


Figure 3.4: The upper slope seen from the side.

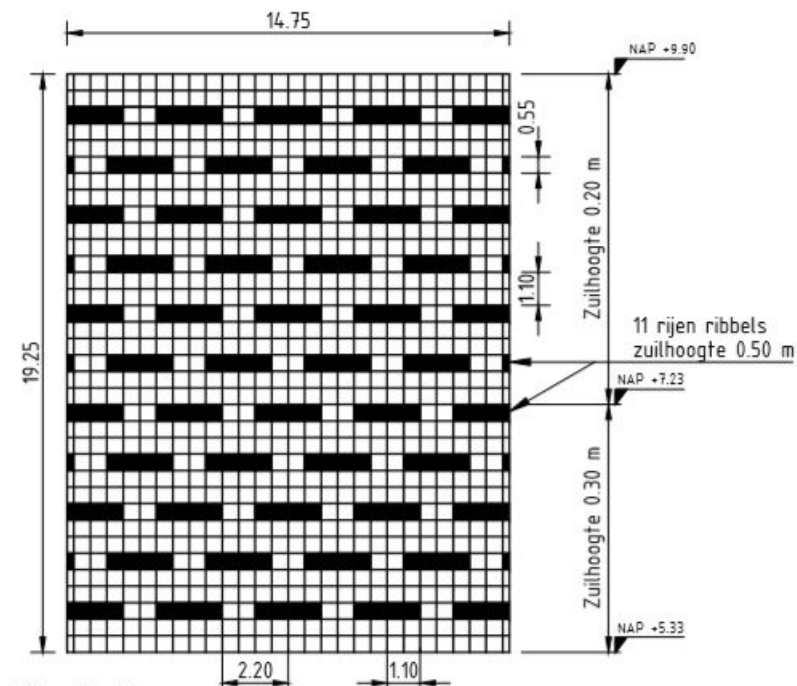


Figure 3.5: The upper slope seen from above.

norms are set to a maximum of 10 liter per second per meter dike. The severe conditions are extensively tested for this dike profile in the Delta Flume of Deltares. The experiments are discussed in the next part of this section, where the data is presented that is needed for the CFD model.

3.3 The delta flume experiments

The new Delta Flume in Delft is built in 2015 and is one of the largest delta flumes in the world. A wide variety of coastal structures can be tested here to assess their safety. The dimensions of the Delta Flume are 300m x 5m x 9.5m and it can generate waves with a maximum height of approximately 4.5 meter. Although the Delta Flume is very large, the experiments can not be executed at prototype scale. The tests with the dike profile of dike section 8a were done at a scale of 1:2.95.

The waves in the Delta Flume are generated with a wave board. It can generate regular and irregular waves. Regular waves are waves with the same frequency and amplitude during the whole experiment. Irregular waves are the opposite, i.e. there is no pattern in the amplitude and frequency of the waves. The latter waves are clearly the type of waves that occur at sea and oceans. In dike testing it is therefore more realistic to test with irregular waves. The irregular waves occurring at the North Sea are captured with the JONSWAP spectrum. The JONSWAP spectrum is therefore mainly used in the Delta Flume to test the Dutch coastal structures.

Many experiments have been executed on dike section 8a. There are two experiments from which the data is available to validate the CFD computations. The difference between the two experiments is that one has a smooth upper slope, whereas the other has the wave reduction ribs shown before (figures 3.4 and 3.5).

The input parameters for the waves are the peak period, water level, spectral wave height and the spectral wave period. Their values as well as the measured overtopping for both experiments are given in Table 3.1.

Besides the measured overtopping at the end of the simulation, one is interested in the

Experiment	$SWL (m)$	$H_{m0} (m)$	$T_p (s)$	$T_{m-1,0} (s)$	$q (L/s \text{ per } m)$
8a without ribs	5.3	4.53	7.71	7.73	72
8a with ribs	5.3	4.3	7.66	7.13	9.7

Table 3.1: Input values for the Delta Flume experiment together with the measured overtopping.

4 Waves2Foam

The waves2Foam package [9] is an extension of the CFD software OpenFOAM. It is developed in 2011 with the goal to generate and absorb waves at the free surface. This toolbox was eventually extended to modelling the interaction of waves with coastal structures. The OpenFOAM version v1812 with the plugin waves2Foam 2134 is used in this research.

The waves2Foam plug-in together with the already extensively developed OpenFOAM is able to set up a similar numerical experiment as done in the Delta Flume. OpenFOAM takes care of the basis of CFD modelling, i.e. the meshing, boundary and initial conditions, discretization methods and solving the discretized equations. These CFD actions have already been introduced in Section 2.2. Waves2foam is purely to generate and absorb the waves and has also some helpful utilities in assessing the total overtopping for example. The underlying mathematical notions of waves2Foam and how this is implemented will be discussed in this section.

4.1 Wave generation

The main use of waves2Foam is generating waves. Waves have many properties and most of them vary for different oceans and seas. Therefore experiments and CFD models must implement the correct wave theory that is applicable in their real-life purpose. The type of wave and the standard characteristics like the wave height and wave period are the main concerns. In waves2Foam, this is all implemented by a single input file for the wave properties.

4.1.1 Wave theory

The choice of wave theory is not always trivial. Not every wave theory is applicable to a certain situation. In this study it is clear to use a JONSWAP spectral wave theory. Besides JONSWAP, waves2Foam presents many possibilities concerning wave theories that one wants to use. JONSWAP is important at the point where results from the CFD simulation will be compared with the Delta Flume experiment. For reproducibility purposes however, it might be convenient to use an easier wave theory with regular waves instead of the irregular JONSWAP waves.

Regular waves are mainly based on the Stokes' wave theorem [6]. Sir George Stokes used a potential flow to solve motions of waves on the free surface, with the assumption that the flow was inviscid and irrotational. His research was a continuation of the Airy wave theory, formed by George Biddell Airy [6]. The Airy wave theory was made for a linear wave motion and Stokes was able to extend this theory for non-linear waves. He used perturbation series

to define different order approximate solutions for the wave motion at the water level.

The solution of the free surface elevation η [m] of the first order Stokes' theory is

$$\eta = a \cos(\omega t - \mathbf{k} \cdot \mathbf{x} + \varphi), \quad (55)$$

with a [m] the amplitude of the wave, ω [rad s⁻¹] the angular frequency, \mathbf{k} [rad m⁻¹] the angular wave number vector, \mathbf{x} [m] the position relative to the points where the waves are generated and $\varphi \in (0, 2\pi)$ [rad] the phase of the wave. The angular wave number k is related to the wavelength by the relation

$$k = \frac{2\pi}{L}. \quad (56)$$

In deep water, the dispersion relation [6] between the wave number and wave frequency is

$$\omega^2 = gk. \quad (57)$$

Furthermore, the velocity in the horizontal and vertical direction is given by

$$u_x = \frac{\omega H_s}{2} \cdot \frac{\cosh \|k\|(z+h)}{\sinh \|k\|h} \cdot \cos(\omega t - \mathbf{k} \cdot \mathbf{x} + \varphi) \quad (58)$$

and

$$u_z = \frac{\omega H_s}{2} \cdot \frac{\sinh \|k\|(z+h)}{\sinh \|k\|h} \cdot \sin(\omega t - \mathbf{k} \cdot \mathbf{x} + \varphi) \quad (59)$$

respectively.

Irregular waves are a linear combination of Stokes waves. This results in the equation for the surface elevation η that is used in waves2Foam to generate irregular waves, i.e.

$$\eta = \sum_i^N a_i \cos(\omega_i t - \mathbf{k}_i \cdot \mathbf{x} + \varphi_i), \quad (60)$$

with N is the number of wave components that are considered to create the wave. Every wave has their own frequency, amplitude etc.

For spectral wave theories, the values in equation 60 depend on the wave spectrum. Many of the values are dependent on the spectral shape, i.e. the wave spectrum. The input values for a JONSWAP spectrum in waves2Foam are therefore the peak period T_p , the spectral wave height H_s and the peak enhancement factor γ_p , as given in Section 2.1.1.

4.1.2 Relaxation zones

To generate the waves, waves2Foam makes use of a relaxation zone technique. The purpose of this technique is to create a boundary condition that avoids internally reflecting waves at the wave generating boundary. The technique can also be used at other boundaries to avoid the reflection at those boundaries. The relaxation is explicitly used for the indicator function for the fluid fraction α and the velocity \mathbf{u} . A zone has to be chosen where this relaxation takes place and the velocity and indicator function are determined by a weighing function that corrects the value for the given parameters, i.e.

$$\phi = (1 - w_R) \phi_{target} + w_R \phi_{computed}. \quad (61)$$

So ϕ is either the velocity or the phase fraction. The variable w_R is the weighted function and is exponentially distributed by

$$w_R = 1 - \frac{e^{\theta^{3.5}} - 1}{e - 1}, \quad \theta \in (0, 1). \quad (62)$$

The value of θ is dependent on the location in the relaxation zone. For example, at points laying on the interconnection between the relaxation zone and the computing zone, $\theta = 0$ such that $w_R = 1$. At the inlet boundary $\theta = 1$ such that $w_R = 0$.

There are also other ways to distribute the weighted function, mainly via polynomial functions. The exponential distribution is the default distribution, however.

4.2 Overtopping

Overtopping will be the key parameter to validate the results of the CFD model with the experimental data. Hereby both the mean discharge rate over the whole simulation and the overtopping of the individual overtopping waves are studied. There are two common ways to assess the amount of overtopping.

4.2.1 Flux

The first method is to use the overtopping function of waves2Foam itself. It calculates the overtopping at each time instance of the simulation with the available flux fields. The two available fluxes from the simulation are the normal flux ϕ [$m^3 s^{-1}$] of the fluid as well as the flux multiplied by the density $\phi_\rho = \phi \cdot \rho$ [$kg s^{-1}$] of that particular fluid. By using Equation 23, the density flux can be expressed as

$$\phi_\rho = \phi \cdot \rho = \phi(\rho_{\alpha=0}(1 - \alpha) + \rho_{\alpha=1}\alpha) \quad (63)$$

$$= (\rho_{\alpha=1} - \rho_{\alpha=0})\phi_\alpha + \rho_{\alpha=0}\phi. \quad (64)$$

Here, a third flux $\phi_\alpha = \phi \cdot \alpha$ [$m^3 s^{-1}$] is introduced. Note that this ϕ_α represents the water flux. Solving for ϕ_α gives

$$\phi_\alpha = \frac{\phi_\rho - \rho_{\alpha=0}\phi}{\rho_{\alpha=1} - \rho_{\alpha=0}}. \quad (65)$$

With the two available fluxes, the flux of water can thus be calculated.

The next step to assess the total overtopping is to define the set of faces where the flux needs to be evaluated. For overtopping, one needs to select all the faces of the cells directly above the end of the crest of the dike. The total overtopping is then calculated with

$$\mathbf{q} = \sum_{f \in \mathcal{F}} \phi_{\alpha,f} \frac{\mathbf{S}_f}{\|\mathbf{S}_f\|_2}. \quad (66)$$

So \mathbf{q} [$m^3 s^{-1}$] is the volume flux over all the faces defined in the set \mathcal{F} and \mathbf{S}_f is the normal vector to each face that is then normalized.

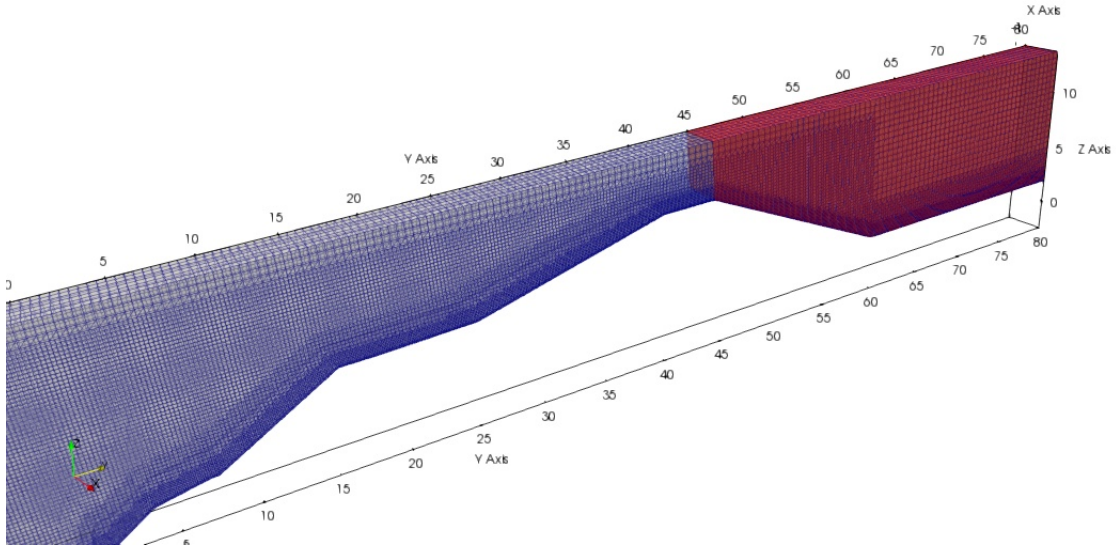


Figure 4.1: An example of a domain (red area) where the overtopping is calculated by means of integration of the variable α .

4.2.2 Integration

The second method to calculate the overtopping is to use the post-processing software par-aView. One can select part of the domains and integrate all the simulated variables over that domain. The integration of the variable α gives the amount of water in that domain. So one could select the area behind the dike such as shown in Figure 4.1 and integrate. The integrated data at each time step can then be exported to use for further analysis.

Due to issues with the installation and compiling errors, the overtopping function of waves2Foam discussed in Section 4.2 did not work. Therefore, the method of integration is used to calculate the overtopping.

4.3 Permeability

When implementing the permeable lower slope, one needs the porous flow equations as given in Section 2.2.3. There are several methods to close the system for computing the resisting force. One of the formulations is stated by Van Gent (2015) in [19]. The parameters a and b have the form

$$a = \alpha \frac{(1-n)^2 \nu}{n^3 d_{50}^2}, \quad b = \beta \left(1 + \frac{7.5}{KC}\right) \frac{1-n}{n^3 d_{50}}. \quad (67)$$

Here, d_{50} is the nominal diameter of the grain material. The constants α , β and KC are closure coefficients, with KC being the Keulegan-Carpenter number.

5 Modelling approach

To get to the final goal, which is modelling a replica of the Delta Flume experiment, several intermediate steps must be taken. The purpose of this section is to give a brief overview of the modelling scheme to get to the final goal.

To get familiar with the software, a simplified model of the Afsluitdijk will be implemented. This model will serve as a reference model. In this model, the necessary settings in the CFD model will be implemented such as the mesh and the boundary conditions. The geometry of the dike profile as seen in Figure 3.2 will already be implemented without the roughness elements, i.e. the dike is completely smooth.

Next, the role of turbulence will be investigated. The turbulence models $k-\epsilon$ and $k-\omega$ SST are tested and compared with the reference model to visualize their behaviour. Furthermore, for CFD simulations it is important to check whether the grid size does not influence the convergence of the solution. This will be checked by choosing two different grid sizes and comparing the results with the original grid size.

From here, the dike will be developed further with the introduction of the roughness elements on the lower and upper slope of the dike. Firstly, the Level-blocs will be implemented on the lower talud. The blocks are permeable and have particular shapes that must be taken care of. This model will then be calibrated with the Delta Flume experiment with the smooth upper slope. At last, the ribs on the upper talud will be placed onto the dike to create the same conditions as the Delta Flume experiment.

A schematic overview of the approach that has been conceptualized is given in Table 5.1.

Chapter	Mesh size	Turbulence model	Wave theory	Dike profile
6	N	-	First order Stokes	Smooth slopes
7	N	$k - \epsilon$, $k - \omega$ SST	First order Stokes	Smooth slopes
8	$\frac{1}{2}N$, $\frac{3}{4}N$, N	$k - \epsilon$ or $k - \omega$ SST	First order Stokes	Smooth slopes
9 & 10	N	$k - \epsilon$ or $k - \omega$ SST	JONSWAP	Level-blocs lower slope, upper slope smooth
11	N	$k - \epsilon$ or $k - \omega$ SST	JONSWAP	Level-blocs lower slope, ribs upper slope

Table 5.1: The CFD modelling approach.

6 Model 1: Reference model

This section describes the first CFD model that will be used as a basis for all the upcoming simulations. The goal of this section is to give an overview of the settings that lead to a working simulation that generates waves and calculates the overtopping over a dike. This includes the choice of boundary conditions, generating the mesh and settings regarding discretizing the NS equations.

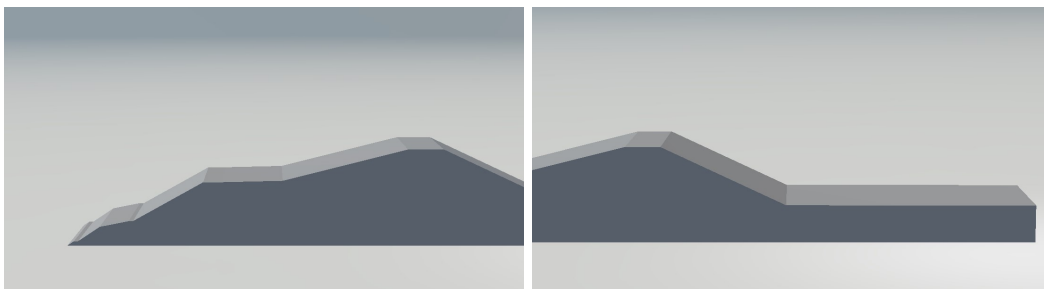
6.1 Geometry

The domain of the CFD simulation is basically a box from which the dike is cut out. Whereas the box is easily constructed in OpenFOAM itself (as will be seen in Section 6.3), the dike profile is more complicated to build. An external CAD program is therefore needed and SALOME, version 9.3.0 is chosen for this purpose. The geometry of the dike will resemble dike section 8a that is shown in Section 3.2. No roughness elements are considered in this reference model. All surfaces are therefore smooth and this simplifies the geometry greatly. A box to capture the overtopping water is made behind the crest to assess the amount of overtopping by simple integration over that part of the domain.

6.2 Initial and boundary conditions

Boundary conditions are necessary to solve the discretized partial differential equations. The choice of boundary condition determines the well-posedness of the problem, i.e. whether there exists a unique solution to the set of equations.

A separate folder in OpenFOAM is dedicated to initiate the boundary conditions of each variable at each boundary. The three variables that the PIMPLE method solves are U ,



(a) The profile of the dike, with the same dimensions corresponding to Figure 3.2.

(b) The area behind the dike where the overtopping water is stored.

Figure 6.1: The geometry made with SALOME that is implemented in the OpenFOAM environment.

p_{rgh} and α . Therefore three files, one for each variable, are needed to specify the boundary conditions. OpenFOAM has a wide variety of many different boundary conditions that can be implemented. In Table 6.1 an overview is given of the boundary conditions that have been chosen.

Some of these boundary conditions have a clear name that describes their mathematical meaning perfectly. For example the *zeroGradient*, *slip* and *noSlip* boundary conditions are the constraints

$$\frac{d\phi}{dn} = 0 \quad (68)$$

$$\phi \cdot n = 0 \quad (69)$$

$$\phi = \mathbf{0} \quad (70)$$

respectively. Other names need to be explained a bit more. The *inletOutlet* conditions are a mixed boundary condition that depends on the direction of the flux. The standard *inletOutlet* condition behaves like a *zeroGradient* boundary condition when the flux goes out of the domain, but it changes to a *fixedValue* condition when the flux moves into the domain:

$$\phi = \mathbf{a}, \quad (71)$$

where \mathbf{a} can be a scalar or a vector, depending on the variable that ϕ represents. The *inletOutlet* boundary condition is applied to the upper boundary, often called the atmosphere boundary, for the variable α in this case. The value of α is set to zero, i.e. only air moves into the domain at that boundary. The *pressureInletOutletVelocity* condition is a special *InletOutlet* condition, but works almost similar. If the flux goes out of the domain the

	U	p_{rgh}	α
Inlet	WaveVelocity	ZeroGradient	WaveAlpha
Bottom	Slip	ZeroGradient	ZeroGradient
Dike	NoSlip	ZeroGradient	ZeroGradient
Outlet	NoSlip	ZeroGradient	ZeroGradient
Atmosphere	pressureInletOutletVelocity	TotalPressure	InletOutlet
Front	Slip	ZeroGradient	ZeroGradient
Back	Slip	ZeroGradient	ZeroGradient

Table 6.1: The boundary conditions of each variable at the different boundaries in the reference model.

velocity has a *zeroGradient* condition, but for incoming flux, the velocity gets its value from the neighbouring internal cells. This boundary condition only works when the value of the pressure is known. Therefore a *totalPressure* boundary condition is used, which gives the value of the static pressure by means of fixing the total pressure. In the incompressible case this leads to the condition

$$p = p_0 - \frac{1}{2}|\mathbf{u}|^2, \quad (72)$$

where the last term is the kinematic pressure. The value of p_0 is then prescribed in the file for the pressure p_{rgh} . The only two boundary conditions left to explain are the *waveVelocity* and *waveAlpha* conditions. Those conditions depend on the incoming waves and will be treated later on in paragraph 6.4.

6.3 Meshing

A good quality mesh is essential for the accuracy and computation of the solution at each time step during the simulation. The size of the mesh is always a trade-off between computation time and accuracy. A very fine mesh with small cells has a higher accuracy, but takes more time to compute than a coarse mesh with lower accuracy. The computation time is mainly dependent on the time step between iterations. The time step is controlled by the Courant-Friedrichs-Lewy condition that is defined as

$$Co = \frac{\|\mathbf{u}\| \cdot \Delta t}{\Delta x} \leq C_{\max}. \quad (73)$$

This constraint will adjust the timestep accordingly when the value of the Courant number Co gets too big. Consequently, a finer mesh results in smaller time steps. In explicit time integration schemes, it is necessary to assure that $Co \leq 1$ for numerical stability. For implicit time schemes however, the constraint is not necessary to attain a stable solution [22]. Although all the models will be using an implicit time scheme, the Courant number constraint is still set to 1. The reason to still use this constraint is for accuracy purposes. The accuracy could be suffering when using time steps that are too large. A formal definition of the Courant number is the amount of information that passes through a grid cell during one time step. Setting $C_{\max} = 1$ assures that the flow will never pass more than one cell during one timestep. Particles jumping over more cells results in a lower accuracy.

The first step of the meshing process is to define a box which will be the basis of the computational domain. The box is created with a dictionary file called *blockMeshDict* where the eight vertices of the box are initialized and every face of the box is named. Furthermore, the mesh of this block is immediately made. This mesh is uniform and the cells that are

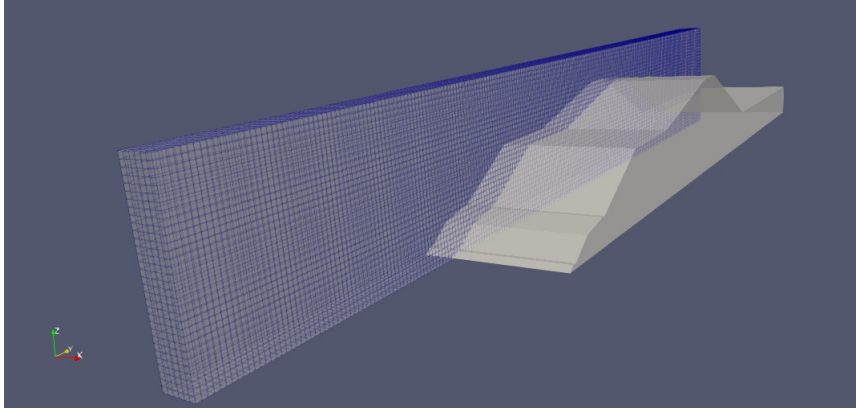


Figure 6.2: Initial blockmesh before the dike is excluded from the domain. The dimensions are 3.3m x 110m x 16m. It has 6 cells in the x-direction, 220 cells in the y-direction and 32 cells in the z-direction.

created will be the largest cell size present in the final mesh after the refinements have been executed. The mesh is seen in Figure 6.2, with the dimensions and number of cells in each direction. It is chosen to keep the domain close to a 2D simulation to reduce the computation time.

The following step is to extract the geometry of the dike. OpenFOAM has the implemented utility *snappyHexMesh* for this purpose. The utility will also refine the mesh where necessary and can add boundary layers. The resulting mesh after extracting the geometry of the dike and implementing refinements is seen in Figure 6.3.

The refinements have been implemented in several ways. At first, two refinement boxes are initialized that are refined one level. A level of refinement means that the amount of cells

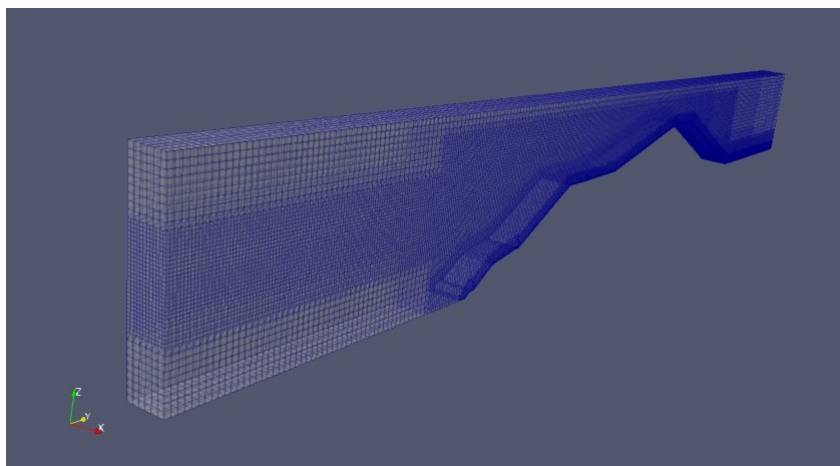


Figure 6.3: Complete mesh created by the *snappyHexMesh* tool of OpenFOAM.

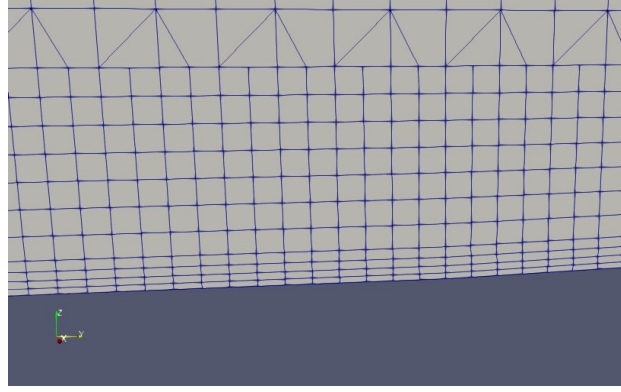


Figure 6.4: The boundary layers around the dike.

in that area are doubled, i.e. the cell size in each dimension gets halved. Those boxes are around the interface and around the dike, the main area of influence.

The other area where refinement is applied is at the surface of the dike. The refinement consists of 2 normal levels of refinement and the implementation of four boundary layers (Figure 6.4). Every level of refinement around the surface consist of seven cells in the normal direction inside the domain. The boundary layers expand with a ratio of 1.2 seen from the boundary layer lying against the surface.

6.4 Wave input parameters

Every setting that has been discussed until now were necessary settings conform an OpenFOAM model. This chapter will be all about the additional waves2Foam package and its settings to generate the waves.

The contribution of the waves2Foam plugin is mainly merged in a single input file, which contains the wave generation and absorption parameters.

Relaxation zone

The relaxation zone is the part of the domain where waves are generated or get damped. In the current project, only one relaxation zone at the inflow boundary that generates the waves is needed. The overtopping water will be captured in the domain behind the dike and does not need to be damped.

The relaxation zone is a rectangular shape in the x- and y-direction. It is placed at the height of the sea level and the length of 10 meters is chosen. This length should be at least one wavelength [8]. The relaxation zone is shown in Figure 6.5.

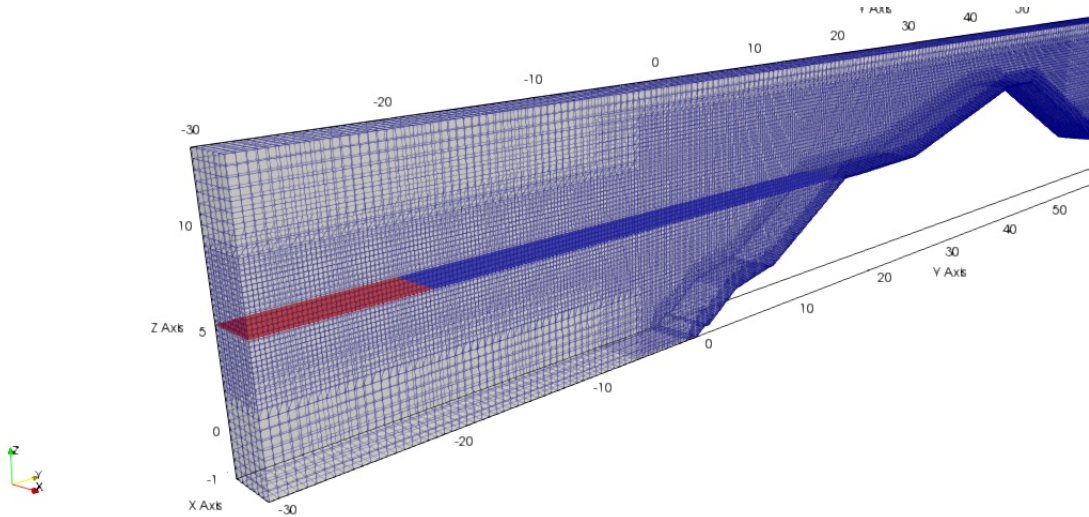


Figure 6.5: The relaxation zone where the waves are generated. The blue plane resembles the sea level and the red part is the relaxation zone.

Wave characteristics

There are a lot of wave types that can be implemented in waves2Foam. For the first simulations where the software is tested and several settings are analyzed, an easily reproducible and uniform Stokes wave is chosen. The wave height and period will be chosen similar to the experiment, discussed in Section 3.3. Therefore the wave height $H_s = 4.2$ and wave period $T_{m-1} = 7.73$. Other settings are schematically given in Table 6.2.

The ramping time is a build-up period for the wave to fully develop. It is needed to get a stable solution at the beginning of the simulation. Otherwise, shocks may appear. Waves2Foam couples this information with the boundary conditions *waveAlpha* and *waveVelocity* to set and update these boundary conditions for α and U at the inflow boundary.

Parameter	Value
Relaxation zone	$(0,-30,5.3) \times (3.3,-20,5.3)$
Sea level	5.3m
Wave type	Stokes first
H_s	4.2m
T_{m-1}	7.73s
Direction	$(0 \ 1 \ 0)$
Ramping time	5s

Table 6.2: Input parameters for the incoming waves.

6.5 Results

The simulation is performed for a duration of 120 seconds. During this time, 15 waves collapsed against the dike. Although this simulation time is not that much, the computation time of this model lasted already around 6 hours. This model will be used as a reference in the upcoming chapters, where the turbulence model will be chosen and the grid size independency are tested. The trade-off between simulation time and computation time is a hard one. With 15 waves the analysis in the next chapter can be performed.

The simulation is visualized with the open source software paraView, version 5.6.0. Paraview is able to visualize the values of the different variables to see the patterns in the flow. The data can be analyzed and exported to excel files. Overtopping is calculated by integrating the variable α over all the cells in the domain behind the crest, as shown in Figure 4.1. ². The resulting overtopping and wave height will be shown in the next chapter.

²Due to issues with the installation and compiling errors, the overtopping function of waves2Foam discussed in Section 4.2 did not work.

7 Turbulence model

Turbulence effects will have an influence on the flow pattern and eventually the amount of overtopping that will occur. Especially the turbulence effects that will come from the roughness elements. The two turbulence models that are considered are the $k - \epsilon$ turbulence model and the $k - \omega$ SST model, described in Section 2.2.2. The expectation is that the $k - \omega$ SST will be the most preferable turbulence model. This assumption comes from the experience that $k - \epsilon$ will perform less near walls. The $k - \omega$ SST model uses for the near wall domain the $k - \omega$ model and makes use of the $k - \epsilon$ model in the free stream. Although this choice might look obvious, it is good to see both their performances for this particular model.

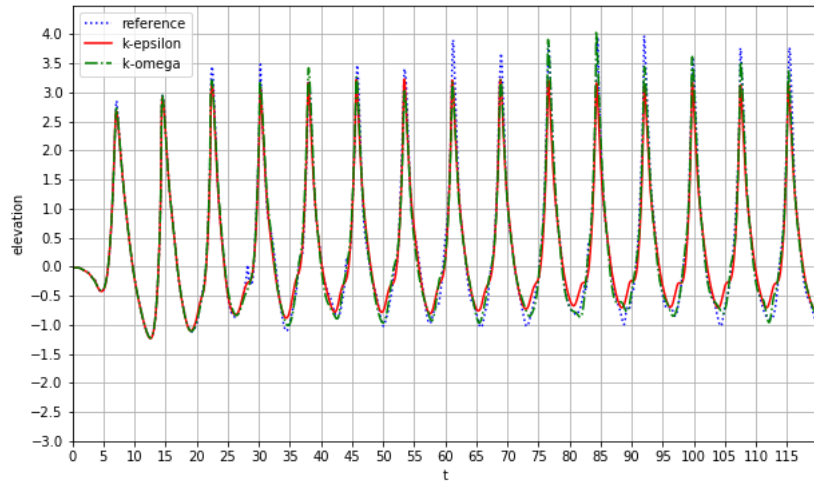


Figure 7.1: Surface elevation of the water at $y = 0\text{m}$.

Boundary conditions

With the implementation of a turbulence model, the new variables k , ν_t and ϵ or ω , depending on which turbulence model is used, are introduced. These variables also need prescribed boundary conditions. The most important boundary condition are wall functions, that can model the effects of roughness elements on the mean flow field. The wall functions will not have many effects in the current models, with smooth slopes. It will be of importance later in this study though, when the roughness elements are considered.

Surface elevation

The surface elevation is measured at multiple points in the domain. The surface elevation is a `waves2Foam` utility that measures the height of the free surface at a point in the (x, y) -plane. This point is defined as a gauge, that collects the data and writes the surface elevation at each iteration in a separate file. Two gauges are placed at the coordinates $(1.65, 0)$ and $(1.65, 7.88)$. The first point is located at the start of the dike structure and the second gauge is at the toe of the lower slope of the dike. The two turbulence models together with the reference model from Chapter 6 are then compared at those points, to see the differences in surface height of the water. The wave characteristics are copied straight from the reference model and parameters like computation time are obviously taken the same. The resulting surface elevation can be observed in Figure 7.1 and Figure 7.2.

The figures show that there are some minor differences in the elevation, at both gauges. From the first gauge at $y = 0m$, it can be seen that the solid red line resembling the $k - \omega$ model produces the lowest amplitude in the waves. This is quite good to explain, because this turbulence model is performing well in the free stream. The damping from this turbulence model results in energy dissipation in the waves, although in this particular model the differences are small, but not negligible.

Figure 7.2 also shows some small differences between the different turbulent models. Besides that, the periodic elevation converges to a steady wave after approximately five waves in all models.

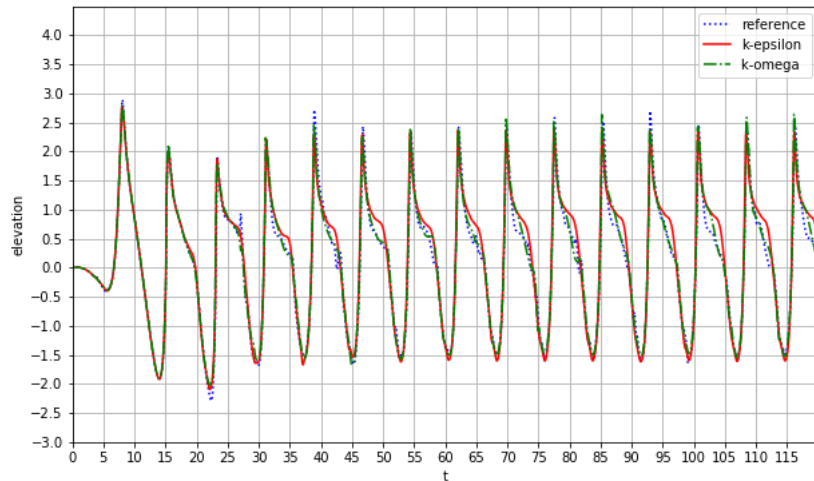


Figure 7.2: Surface elevation of the water at $y = 7.88m$.

Overtopping

Although the wave propagation looks quite similar while looking to the surface elevation at different points, the effects of those minor differences can have a large influence in the measured overtopping. The overtopping is again measured by means of integration of the variable α . In Figures 7.3 and 7.4, the relative overtopping and the total overtopping are shown.

The relative overtopping shows that every wave produces overtopping in all models. The amount of overtopping from every wave is different however. There is no clear pattern on which turbulence model produces the most overtopping. A better overview of how the overtopping develops is shown in Figure 7.4.

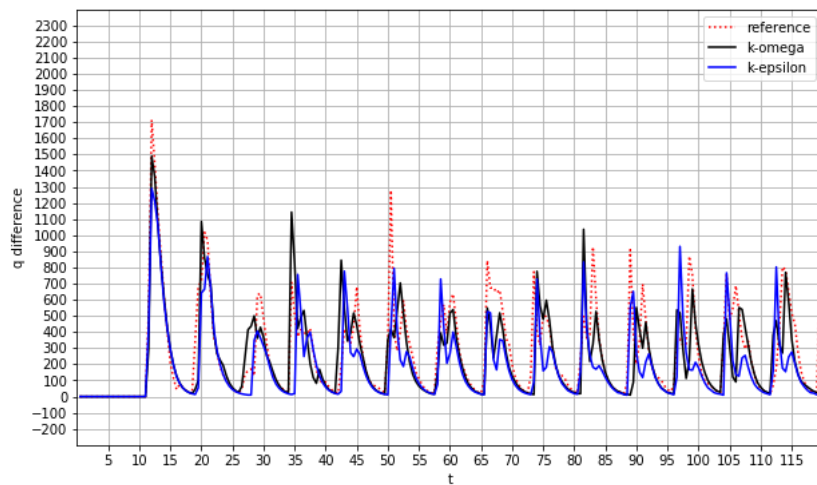


Figure 7.3: Relative overtopping of the turbulence models.

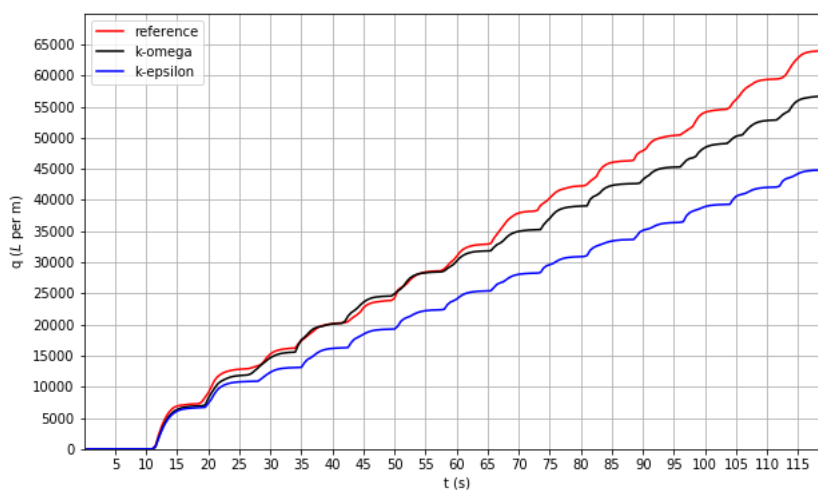


Figure 7.4: Cumulative overtopping of the turbulence models.

As expected, the standard CFD model without turbulence is responsible for the most overtopping. Although at the first half of the simulation the difference with the $k - \omega$ SST model is minimal, the two lines diverge. The $k - \epsilon$ model however consistently has the lowest overtopping during the whole simulation. In the end it has approximately 30 percent less overtopping than the standard model.

Figures 7.3 and 7.4 show that the choice of turbulence model has a significant impact on the results. Although the slopes are smooth, turbulent effects occur in these simulations. For example breaking waves onto the dike lead to a turbulent flow in that area. So it is not clear to state which turbulence model produces more accurate results. From [15] [16], the $k - \epsilon$ model behaves not well near the wall. It over-predicts the turbulent kinetic energy at cells close to the wall. The $k - \omega$ SST model therefore looks more realistic, as turbulence is modelled well at those points [16]. Furthermore, the $k - \epsilon$ works similar as the $k - \omega$ model in the free stream and the free surface. Moreover, the accurate prediction of the turbulence becomes more important when roughness elements are implemented in the model. So based on the theoretical knowledge, the $k - \omega$ SST is used for the upcoming CFD models.

8 Mesh size

With the turbulence model chosen, it is time to look at the grid independency of the CFD simulations. Three different initial mesh sizes are considered. This initial mesh size is the mesh that is created with the *blockMesh* utility in OpenFOAM. The three mesh sizes are defined as **Standard**, **Coarse** and **Very coarse** and their respective amount of cells can be seen in Table 8.1. The standard mesh is the mesh that has been used initially in the previous models. The other two meshes have more cells. The number of boundary layers and their respective sizes are the same in each model.

	x	y	z
Standard	6	220	32
Coarse	5	165	24
Very coarse	3	110	16

Table 8.1: Number of cells in each direction in the three different mesh sizes after using the *blockMesh* utility.

Surface elevation

The surface elevation of the three models is examined first. In Figure 8.1 the elevation at the toe of the lower slope is shown. The three lines are overlapping almost exactly, which

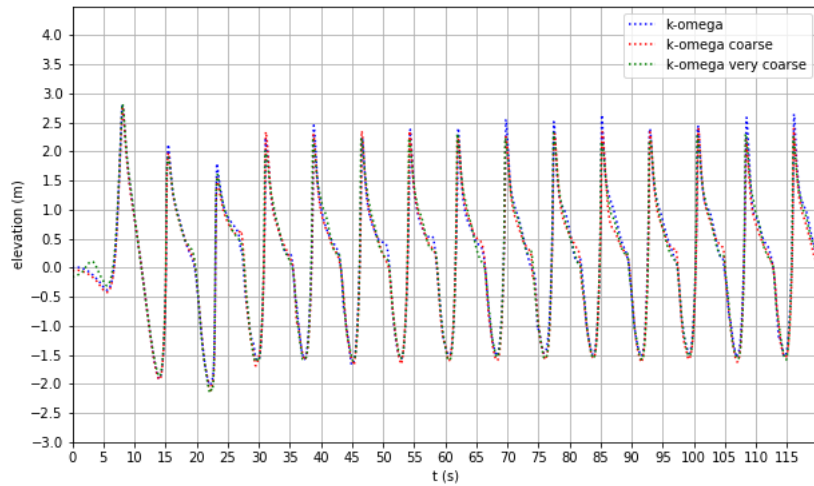


Figure 8.1: Surface elevation of the water at $y = 7.88\text{m}$ (Toe of lower slope) of the three different mesh sizes.

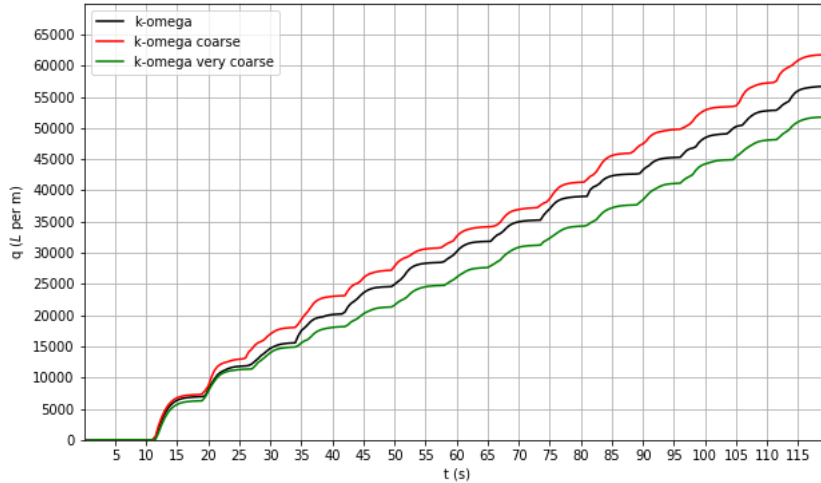


Figure 8.2: Total overtopping of the water of the three different mesh sizes.

was expected. There are some minor differences as in the comparison between the turbulence models. These differences could have a significant impact in the amount of overtopping.

Overtopping

The obtained overtopping in Figure 8.2 shows strange behaviour. At first, the amount of overtopping in the two coarser grids have approximately a difference of around 10 percent. Those percentages are not the most worrying however. It was expected that the overtopping showed a convergent behaviour when the grid size becomes finer, e.g. a finer mesh result in consistently more or less overtopping. Instead, the line of the finest grid lies in the middle of the three lines.

To come back to the first observation, the graph shows that the grid size is quite important. Especially when dealing with a free surface flow, the sharpness of the free surface can have a significant influence in the accuracy of the solution. Fortunately, in constructing dikes they mainly look at the overtopping per second. The current safety rules are that dikes may have a maximum overtopping rate of 5-10 L/s per meter dike. So a 10 percent deviation in the results will rarely be the difference between a safe structure and a weak structure. Only when the computed overtopping comes close to the passable overtopping rate, a detailed study to the maximum error due to grid size and other factors are needed. For this study, a 10 percent error deviation in the results is for now chosen as acceptable.

Courant number

The second observation about the finest grid not resulting in the most or least overtopping of the three grid sizes is of more concern. This strange behaviour could have been caused by the choice of the Courant number. Recall that the Courant number has the physical interpretation of how many cells fluid particles (information) travels during one time step. If information travels more than one cell during one time step, i.e. $Co > 1$, errors could occur and even accumulate. Currently, the Courant limit is set to precisely one. Due to round-off errors, the Courant number is almost never precisely one and it is therefore likely that this number exceeds one slightly. This should not be a problem straight away. The PIMPLE algorithm is known for its capability to perform accurate for $Co > 1$ as well [7]. Also, if the grid sizes are fine enough the errors should be small.

Although it seems that the choice of Courant number should not give much of a difference, the models with the three different grid sizes have been run with the restriction of $Co < 0.5$. The results are shown in Figure 8.3. The grey area resembles a $\pm 10\%$ deviation of the original measured overtopping when the Courant limit was equal to one. The results are clear. In the very coarse grid, the Courant restriction makes much of a difference and it

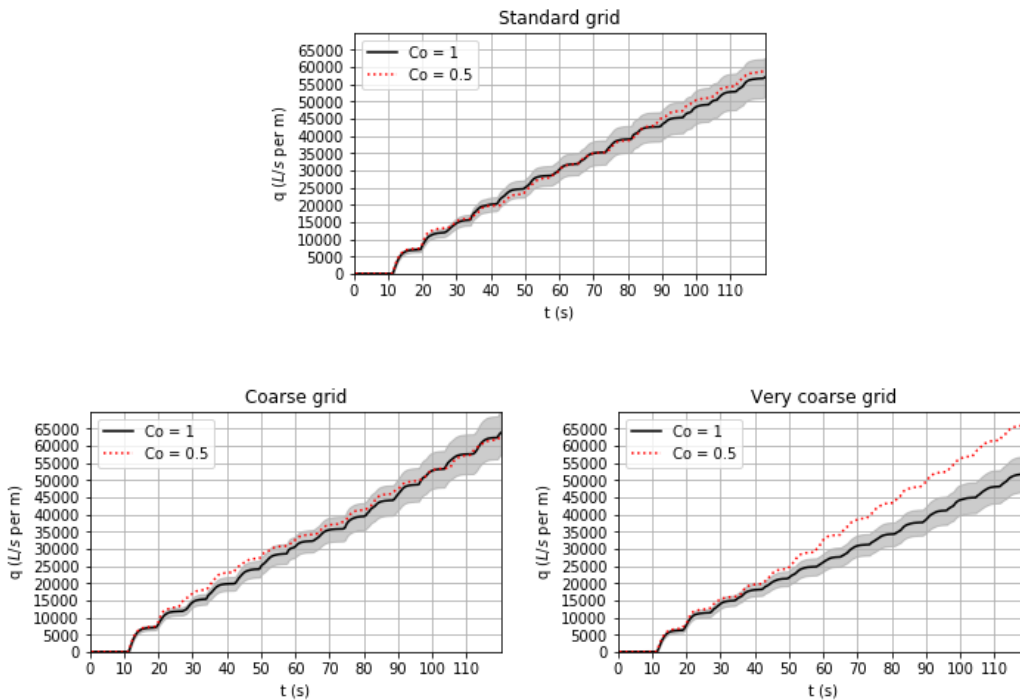


Figure 8.3: The total overtopping of the simulations with $Co = 0.5$ and $Co = 1$. The grey area is a 10% deviation of the initial simulation with $Co = 1$.

now produces the most overtopping. For the other two grids there is not a major difference between both simulations and both lines spiral around each other.

The convergence pattern when using a finer grid is now more logical, with the amount of overtopping decreasing when the grid cells get smaller. Although it might be convenient to take $Co < 0.5$ as a standard restriction, the computation time of the simulation will be around twice as long and the very coarse grid will not be used in the future CFD models. So the Courant restriction will be put back to 1.

To conclude, the solution is not totally independent on grid size. However, the differences that occur in the total overtopping (especially) are on the long term not significant enough. Hence, when the CFD models are validated with the experimental data, it will not lead to different conclusions regarding our main research questions.

9 Model 3: Implementation of the Levvel-blocs

All preceding CFD models served the purpose of understanding the software waves2Foam and to examine the performance of standard CFD features like the grid size and the turbulence model.

The next goal is to imitate the Delta Flume experiment and compare the results between CFD and the physical experiment. There are two experiments where the data have been obtained from, which have been already briefly discussed in Section 3.3. The difference between the two experiments is the upper slope, with one of them being smooth and the other containing ribs. In this model it will be all about the first Delta Flume experiment.

Firstly, the wave settings have to be in line with the Delta Flume experiment. They have used a JONSWAP spectrum and it would be far better if the signal that is put into the experiment can serve as input signal for the CFD model as well. Then the Levvel-blocs have to be considered, although they are submerged and their roughness effect ought to be small [13]. Several attempts are done to implement them as good as possible. Their shape and size is time-consuming to implement in the domain, however. A roughness variable K_s is used as an alternative to imitate the roughness. Including permeability on the lower slope is investigated as well.

9.1 Wave input settings

The two major differences in the upcoming models compared to the model that already have been carried out are different incoming waves and the changing geometry. Those changes will be mentioned in this paragraph.

JONSWAP spectrum

The experiments in the Delta Flume have been performed with a JONSWAP spectrum. This spectrum is given as input in the wave paddle at the start of the flume that generates the waves. This exact signal is not in our possession, but the data of the surface elevation of the water at the foreshore is available. Using a Fourier transformation, implemented in the code given in appendix A.1, the spectrum that belongs to this signal is seen in Figure 9.1.

The goal is to match this spectrum in our CFD model. The peak period and peak wave height of the experiment are provided and are respectively $T_p = 7.33$ and $H_s = 4.22$.

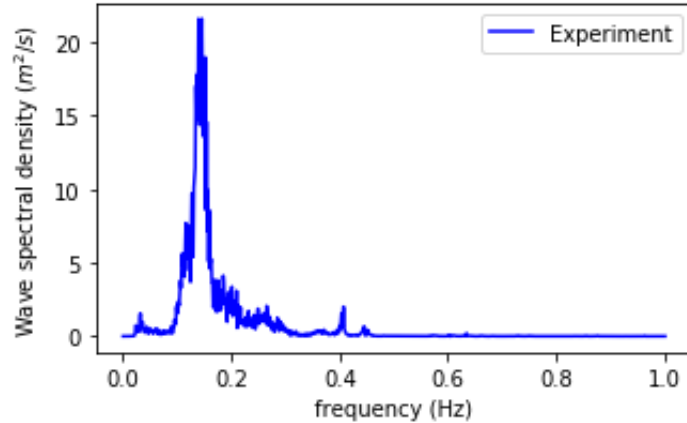


Figure 9.1: Estimation of the JONSWAP spectrum of the Delta Flume experiment.

Wave paddle signal

It would be great to use the exact same signal of the surface elevation of the experiment in the Delta Flume in the CFD model. The package of waves2Foam itself is not able to do this however. Luckily, there are other options for this job.

The first option is the usage of the additional package OceanWave3D that comes together with the installation of waves2Foam. This utility uses the potential of the flow to generate the waves [9]. Although it can take an input signal from a wave paddle, it was not clear how this input must be implemented in the case directory of OpenFOAM. Although there was an example of an OceanWave3D model available, it did not work and due to time limitations it was decided not to investigate this further.

Another software package that is capable of using experimental wave paddle signals is olaFlow. The software is an extension of OpenFOAM similar to waves2Foam, but developed by another organisation. All the source codes of olaFlow are also freely available. This software has also a nice Graphical User Interface for the input of all the settings of the CFD model. This option is also not investigated further, as the present research focuses on waves2FOAM. But it is good to know that this option exists and is suggested to look into for further research.

9.2 Levvel-blocs

The second big change in the model is the geometry. The roughness elements of the dike need to be implemented. This will be done step by step, to see the effects of the respective changes and to compare with the corresponding flume experiment. The first roughness elements that are implemented are the Levvel-blocs. Their particular shape and size make it

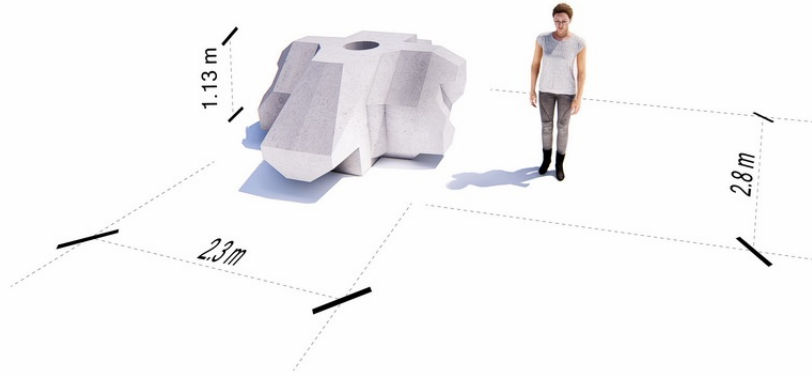


Figure 9.2: A single XblocPlus (Levvel-bloc). They cover the entire lower talud of the dike.

very hard to implement in the geometry. Therefore it is chosen to still consider the lower slope of the dike as a single smooth surface, but to play with the in-built roughness parameter and porosity settings. Roughness and porosity are considered the most important elements of the Levvel-blocs to reduce the overtopping. Although the lower talud is submerged due to the high water and therefore the overtopping effects have become much less significant, they cannot be neglected completely.

Roughness

The shape of the Levvel-blocs are mainly responsible for a great reduction in the amount of overtopping. This shape, as seen in Figure 9.2³, is time-consuming to recreate and also their uniting shape when the blocks lay over each other will be an exhaustive task. For those reasons, the first attempt to capture the roughness of the blocks is via a roughness parameter.

In CFD, roughness on a surface can be implemented in the variable ν_t via a wall function. This wall function is needed because the calculation of ν_t is not valid in the region close to the boundary. It is replaced by the near-wall viscosity ν_w . The two parameters that are important are the roughness height K_s and the roughness factor C_s , which account for the shape of the roughness elements.

For smooth surfaces, the velocity profile of the flow in the normal direction of the wall from

³Source: <https://www.xbloc.com/en/news/introduction-of-the-new-xblocplus>

the boundary to the center of the adjacent cell is given by

$$U^+ = \frac{1}{\kappa} \log(Ey^+), \quad (74)$$

where κ is the Von Kármán constant, E is another constant and y^+ is the dimensionless distance between the boundary and the adjacent cell center. The velocity is also made dimensionless. This formula is called the law of the wall and fits for values of $30 < y^+ < 200$. Theodore von Kármán (1903) [23] was the first to publish this result.

Rough surfaces need an adjustment of the law of the wall to account for the shape and size of the rough elements. This is captured in the a single function ΔB . The law of the wall becomes

$$U^+ = \frac{1}{\kappa} \log(Ey^+) - \Delta B. \quad (75)$$

The variable ΔB is thus a function of K_s and C_s . The function is split into three regions of K_s :

$$\Delta B = \begin{cases} 0, & K_s^+ < 2.25 \\ \frac{1}{\kappa} \left(\frac{K_s^+ - 2.25}{87.75} + C_s K_s^+ \right) \cdot \sin(0.4258 [\log K_s^+ - 0.811]), & 2.25 \leq K_s^+ \leq 90 \\ \frac{1}{\kappa} \log(1 + C_s K_s^+), & K_s^+ > 90 \end{cases} \quad (76)$$

The dimensionless variables are given by

$$U^+ = \frac{U}{u_\tau}, \quad y^+ = \frac{\rho y_p u_\tau}{\mu}, \quad K_s^+ = \frac{\rho K_s u_\tau}{\mu},$$

with u_τ the shear velocity. The near-wall viscosity is then calculated with

$$\nu_w = \frac{\nu y^+}{\frac{1}{\kappa} \log(Ey^+) - \Delta B} \quad (77)$$

The roughness parameters K_s and C_s in OpenFOAM can be adjusted with the OpenFOAM wall function *nutkRoughWallFunction*. The strategy is to set the C_s value for the shape of the roughness elements to the value 0.5, which corresponds to a uniform distribution of the elements. The parameter K_s , the size of the elements will be played with. The strategy is to calibrate K_s such that the overtopping of the CFD simulation matches the actual overtopping of the experimental data.

Porosity

The next step is to implement porosity on the lower talud. In Section 2.2.3 it is explained which changes in the mathematical framework has been done in `waves2Foam` to account for permeable structures. The documentation on the implementation of the porosity is quite minimal however. This made it a long and intensive search on how to define the porosity on the surface of the dike. The possibility exists however, as is seen in [9].

There is a single `waves2Foam` tutorial that shows that porosity is defined on a domain (block) in the `blockMesh` file. In this domain, the porous equations of Section 2.2.3 hold, where in the other parts of the domain, the standard RANS equations hold (i.e. $n = 1$). The problem with this mesh is that the blockmesh needs to be divided into 6 separate meshes (Figure 9.3) in order to define a porous domain below the lower slope of the Levvel-blocs. Note that the sloped red line in Figure 9.3 coincides with the lower slope of the dike. This option has a large effect on the grid cells, with many different shapes and large ratios of the volumes between neighbouring cells in some parts of the domain. Especially the result that many cells are not close to squares have consequences for the `snappyHexMesh` tool, that needs uniform, close to square cells to refine the mesh properly.

The incorporated quality check at the end of the `snappyHexMesh` process failed consequently and as a result, the CFD simulation could not be performed. There have been attempts to solve this matter, but without success.

There is also an option to create a geometry in STL-extension in OpenFOAM itself. How

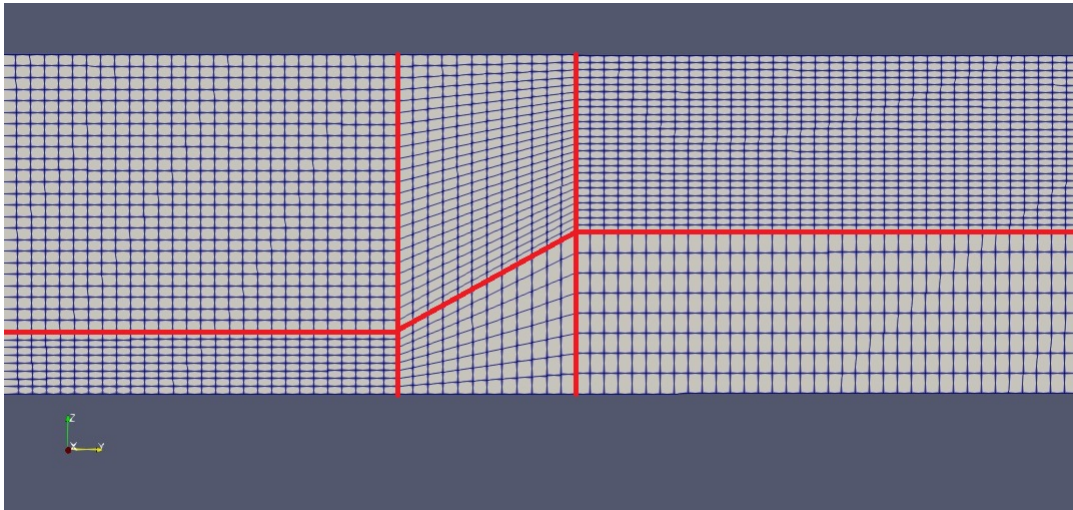


Figure 9.3: side-view of the subdivision of the original blockmesh into 6 smaller meshes to define the porosity in the lower middle block.

to process it as a porous domain was unclear, however. Moreover, another difficulty came to mind when considering a porous volume below the lower slope: where is the roughness defined? The lower slope will not be a boundary anymore and therefore it is unclear how to implement the roughness on that surface. These questions could not be answered in this research unfortunately. Due to time issues it was chosen to keep the porosity out.

9.3 Results

All the methods described above are implemented in the CFD model. Below is a small summary of the important parameters that are described in the previous paragraph.

Name	H_s	T_p	SWL	K_s lower slope	K_s berm	K_s upper slope
8a_rough	4.53	7.71	5.3	0.005	0.0005	0.0005

Table 9.1: The wave and roughness parameters of the first model to imitate the conditions of the Delta Flume experiment.

The simulation time of this model is 960s, which is much longer than all the previous models. With this simulation time, around 120 waves will be generated and therefore the overtopping is calculated more accurately.

Wave spectrum

Wave gauges are placed at 5 locations in the domain. See Table 9.2 for their specific locations.

From the first four gauges, data of the surface elevation is used in a Fourier transformation to obtain the wave spectrum (appendix A.1). This lead to the wave spectra in Figure 9.4 together with the wave spectrum from the Delta Flume experiment. The fifth gauge is used later for other purposes.

Gauge #	Location	Interpretation
Gauge 1	$y = -30$	Inflow boundary (paddle)
Gauge 2	$y = -20$	End of relaxation zone
Gauge 3	$y = -10$	Foreshore
Gauge 4	$y = 0$	Toe of structure
Gauge 5	$y = 7.88$	Toe of lower slope

Table 9.2: Location of the gauges.

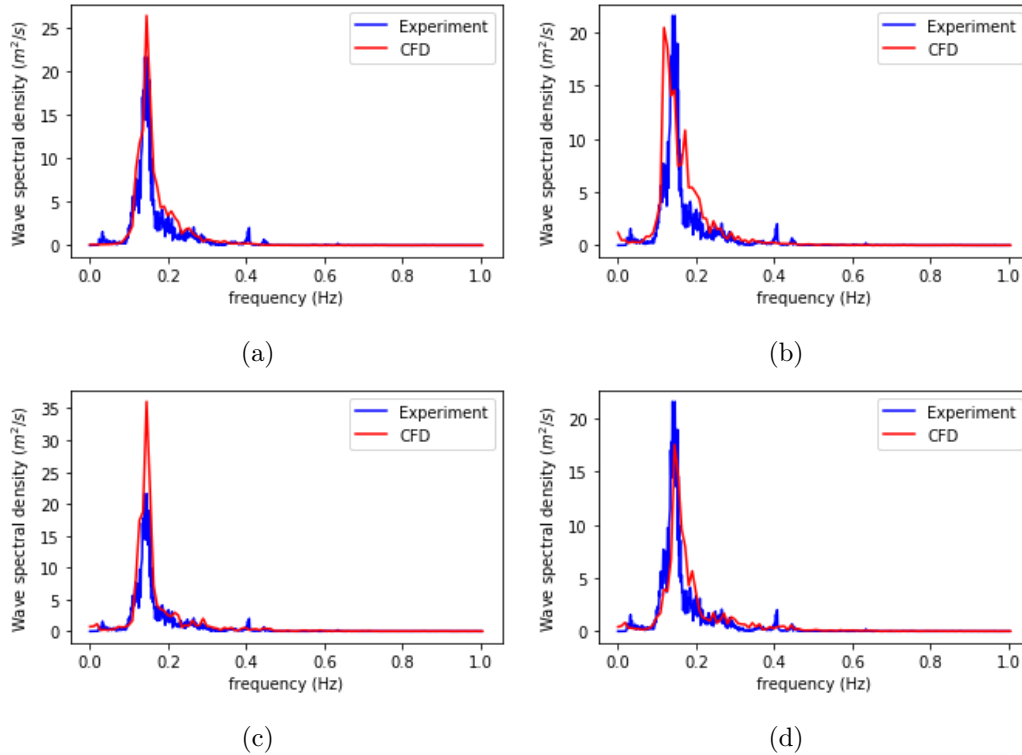


Figure 9.4: The wave spectra measured at different points in the domain. **a)** Wave spectrum at inflow boundary ($y = -30$). **b)** Wave spectrum at $y = -20$. **c)** Wave spectrum at $y = -10$. **d)** Wave spectrum at the start of the dike structure ($y = 0$).

The shape of the wave spectra of the experiment and CFD simulations are quite similar in all cases. The odd one is the spectrum at $y = -20$, where the main peak occurs at the lower frequency and a second peak originates. Also, the peak at $y = -10$ contains a lot more energy than in the experiment.

Besides that, there are a lot of resemblances between both spectra. It is hard to say solely on the wave spectrum whether the wave conditions are similar enough to expect similar overtopping. For example, the differences in spectra in Figure 9.4b and Figure 9.4c can have multiple causes. Maybe the reflection of the dike plays a larger role, and maybe the shape of the level blocks is very important.

Overtopping

The overtopping could give a good indication whether the incoming waves produce the same results. In Figure 9.5 the overtopping over time is shown, together with the overtopping of the experiment.

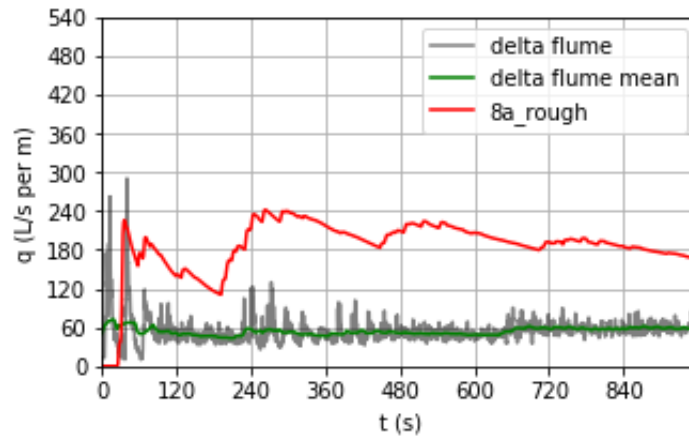


Figure 9.5: Overtopping of the first CFD simulation together with the overtopping of the Delta Flume experiment.

The amount of water over the dike is substantially higher than what is measured in the Delta Flume. The reasons why this occurs is not clear at this point. There could be multiple causes, like a difference in incoming waves, the influence of the lower slope or the amount of roughness in general. In the next chapter, a widespread search is conducted regarding the major phenomena that lead to the large difference and it is tried to get the overtopping in CFD simulations further down.

10 Calibration

The first CFD model was not able to result in similar overtopping as the experiment. This is of course not surprising. There are a lot of parameters to tweak and look further into. Although the difference in overtopping between CFD and the experiment is approximately a factor 3, there is confidence to find the right parameters that cause this difference.

The first thing that was looked into is the roughness on the dike. The parameter K_s is varied on several surfaces to investigate any reducing effects. There is no explicit value available for the material concrete for example, but it mostly ranges from 0.3 – 5 mm⁴.

Next, the height of the waves will be analyzed as well as the water level during the whole simulation. Also, the domain is made longer to see whether the relative short domain in which the waves can develop has an effect.

The profile of the lower slope is changed to incorporate the shape of the level blocks better, to investigate whether the submerged lower talud has maybe more impact than expected. In addition, the slope of the lower talud is changed slightly to see whether that will have a significant effect on the overtopping.

Lastly, possible scale effects that could occur will be investigated. The Delta Flume experiments have been performed on a scale 1 : 2.95. The CFD calculations are on prototype scale⁵ and scaling could have some effect, as can be seen in for example [12].

Name	H_s	T_p	SWL	K_s lower slope	K_s berm	K_s upper slope
8a_rough	4.53	7.71	5.3	0.005	0.0005	0.0005
8a_Ks0005	4.53	7.71	5.3	0.005	0.005	0.005
8a_Ks005	4.53	7.71	5.3	0.5	0.05	0.05

Table 10.1: The wave and roughness parameters of the additional roughness models to see their effects on the overtopping.

⁴Values from https://www.engineeringtoolbox.com/surface-roughness-ventilation-ducts-d_209.html

⁵It is on prototype scale in 2 directions. The width of the dike is in the CFD computation smaller, such that the computational time could be kept lower.

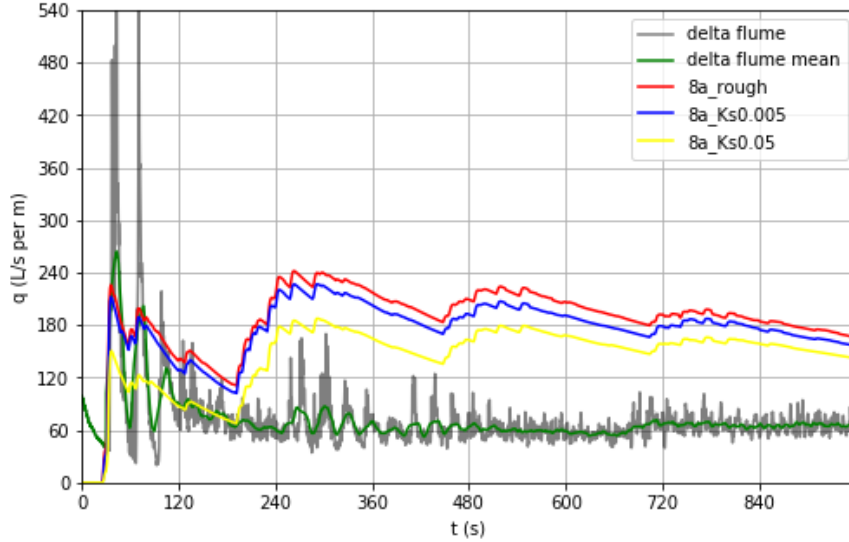


Figure 10.1: Overtopping of the models where the parameter K_s is increased on the slopes and the berm.

10.1 Roughness coefficient K_s

The roughness on the dike could be one of the main parameters that leads to more overtopping. The exact values of K_s on the dike are not known. Therefore it is important to know whether a change in K_s results in a large difference in overtopping. Especially the value of K_s on the lower slope is hard to determine. However, the value of K_s on the upper slope can be between 0.3 – 5 mm. Two additional models have been setup to investigate the effects of varying the roughness parameter. The goal was to come close to the experimental results with only varying the parameter K_s .

The models have led to the overtopping results in Figure 10.1. The roughness parameter on the berm and upper slope is multiplied with a factor ten in the two subsequent models as shown in Table 10.1. Although the value of $K_s = 0.05$ is not likely on those surfaces, to come close to the experimental overtopping the choice was made to try higher values for K_s on those surfaces as well.

10.2 Wave parameters

The wave spectra of the simulations do not give enough information to state that the incoming waves are similar enough. The height of individual waves and the water height are examined. The differences between the experimental waves and CFD simulations that occur lead to new models to find the main cause of the large difference in overtopping.

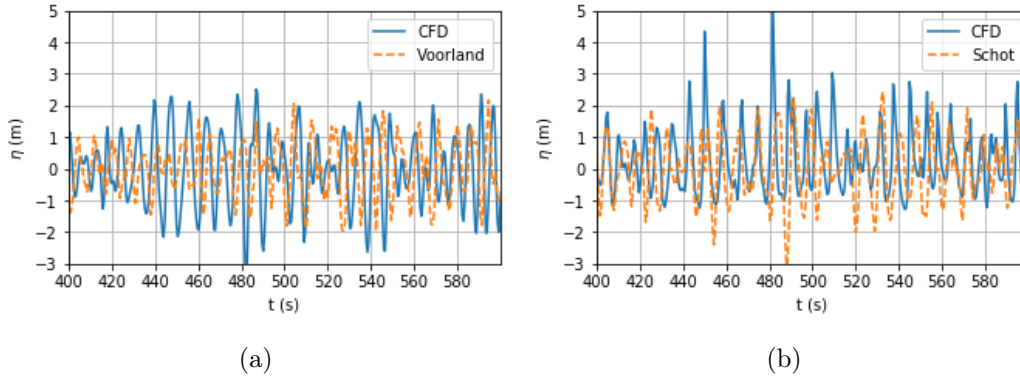


Figure 10.2: Comparison of the wave height from the CFD model and the Delta Flume experiment at **a)** $y = -30$ and **b)** $y = 0$.

Significant wave height

First, the significant wave height of the CFD model is examined and compared with the Delta Flume waves. There are two water height signals available from the Delta Flume. One positioned at deep water, the other close to the toe of the dike. The signals are called **Voorland** and **Schot** respectively. The CFD signal at $y = -30$ will be compared with the **Voorland** signal and the **Schot** signal will be compared with the $y = 0$ signal. These locations are chosen because at $y = -30$, reflection of the dike should not influence this signal. On the other hand, at $y = 0$, the effects of the dike should be visible in the data. In Figure 10.2, two graphs of the difference in surface elevation η in the interval (400s, 600s) are shown. Note again that different random waves are used in the CFD and Delta Flume experiments. Therefore, a comparison of individual waves is of no use.

The surface elevations in Figure 10.2a seem to be quite similar, although it is still hard to interpret them due to the difference in the generated waves. Figure 10.2b shows a different elevation profile. Especially the high peaks of the CFD waves that are also asymmetrical around the original sea water level are not present in the Delta Flume.

Sea water level

From the previous paragraph, the presumption exists that the water level rises in the domain in front of the dike. At the wave gauges in the domain, the running mean of height of the water surface is calculated. The results are shown in Figure 10.3. In figures about the sea water level, the running mean of the water level is shown. It is taken at an interval $[t - 125s, t + 125s]$. Therefore the first and last 125 seconds have been omitted from the figure. Thus $t = 0s$ in the figure shows the running mean at $t = 125s$. Furthermore, due to creating

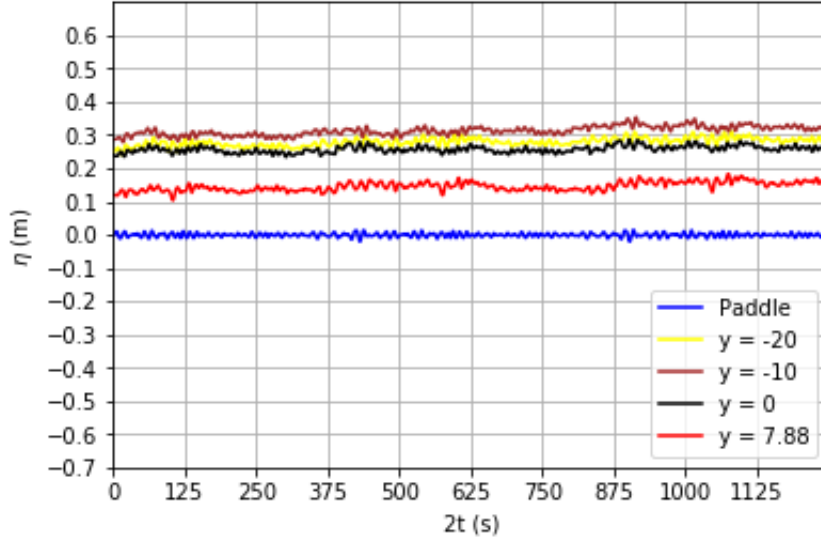


Figure 10.3: The water level at the different waves gauges in the domain with initial sea water level of $5.3m$.

comparable datasets between the Delta Flume and CFD experiments, the time on the x-axis is approximately twice as large as it should be.

Figure 10.3 clearly shows that the sea water level lies approximately 30 centimeters higher than the original sea water level. This phenomenon is called wave setup. Wave setup is a common effect of the wave interaction with the dike. However, the water surface in the whole domain lies approximately 30 centimeters higher and not only around the toe of the dike.

To observe the effects of a higher water level, it is decided to let the initial water level drop to 5 meters. The effects on the water level as well as the resulting overtopping is then visualised in two graphs. In the graph of Figure 10.4, the value of $\eta = 0$ corresponds to the original water level of 5.3 meter. The resulting overtopping is given in Figure 10.5 and Table 10.2.

Name	t	H_s	T_p	SWL	K_s lower slope	K_s rest	q
8a_lowSWL	960 s	4.53 m	7.71 s	5 m	0.5 m	0.05 m	119.5 L/s/m

Table 10.2: The resulting overtopping of the CFD model with a lower sea water level.

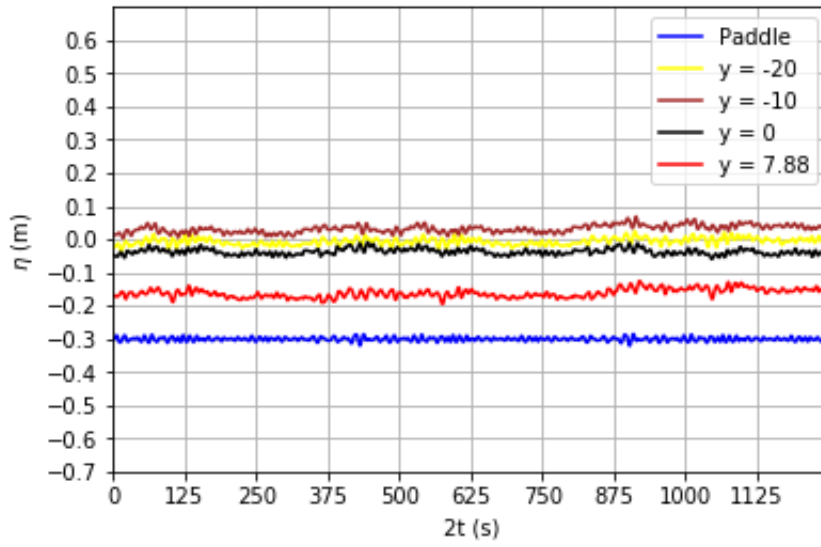


Figure 10.4: The water level at the different waves gauges in the domain with original sea level of 5m.

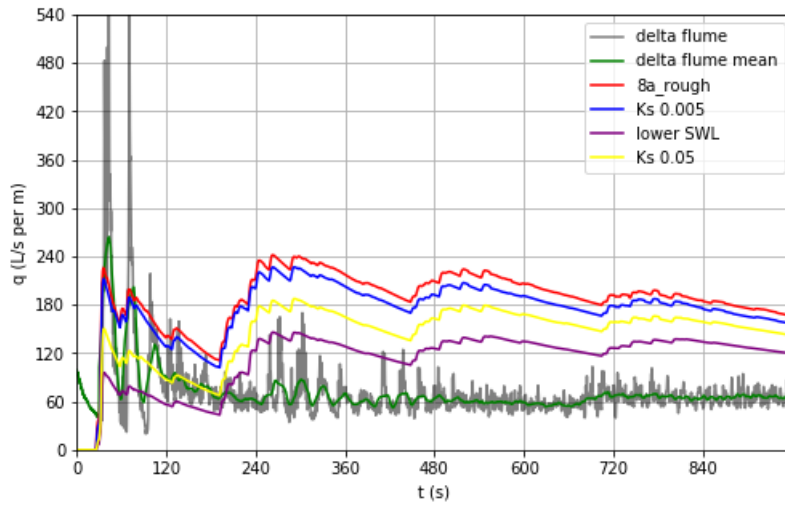


Figure 10.5: Overtopping of the model with the lower SWL, together with previous CFD models.

Gauge #	Location	Interpretation
Gauge 1	$y = -60$	Inflow boundary (paddle)
Gauge 2	$y = -40$	End of relaxation zone
Gauge 3	$y = -20$	Foreshore
Gauge 4	$y = 0$	Toe of structure
Gauge 5	$y = 7.88$	Toe of lower slope

Table 10.3: Location of the gauges in the longer domain.

Interaction with dike structure

The interaction with the dike influences the incoming waves, mainly by reflection and water that flows back from the dike. The toe of the dike lies 30 meters from the inflow boundary. After the waves develop in the relaxation zone, they are already 20 meters before the dike. This could be a rather short distance, as they might be directly influenced by the interaction of the dike.

The short domain should be enough, as the waves are fully created in the relaxation zone. Nevertheless, it is good to confirm this. So the length of the domain before the dike was doubled to 60 meters in the upcoming model. The gauges were now placed in the positions shown in Table 10.3. At this point, it was found that the parameters H_s and T_p were not set correctly. The values were changed to $H_s = 4.22m$ and $T_p = 7.69s$. The resulting sea water level over the domain (Figure 10.6), the wave height (Figure 10.7) and the amount of overtopping water (Table 10.4) are shown.

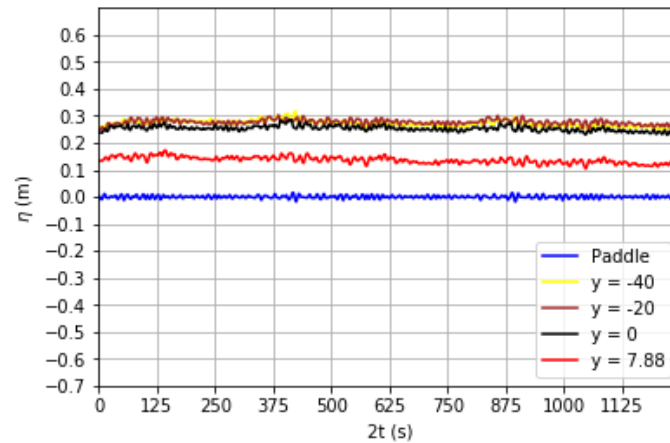


Figure 10.6: Water level at the different gauges in the domain of the CFD model with the longer domain before the dike.

Name	t	H_s	T_p	SWL	K_s lower slope	K_s rest	q
8a_long	960 s	4.22 m	7.69 s	5.3 m	0.5 m	0.05 m	154.1 L/s/m

Table 10.4: The resulting overtopping of the CFD model with the longer domain before the dike on the lower slope. Instead of 30m, the length from the inflow boundary to the toe of the dike is 60m.

The water level is still high in the whole domain, almost as high as in the original CFD simulation. The waves are also asymmetric with high peaks. These similar results lead to comparable overtopping as in the model with the shorter domain. Computationally, this is good to know as this saves a lot of extra grid cells.

To visualize the effects of the interaction with the dike, a model without dike was made. A simple *blockMesh* was sufficient for this purpose. A second relaxation zone was created at the end of the domain to absorb the waves. With the exact same random waves generated, the difference in wave height and sea water level is seen in Figure 10.8.

Figure 10.8a confirms that the same random waves were generated. In Figures 10.8b - 10.8d, the peaks of the waves seem higher almost at every wave. Furthermore, the water level in Figure 10.9 shows a small rise regarding the models with a dike structure, instead of the almost 30 centimeters which was observed earlier.

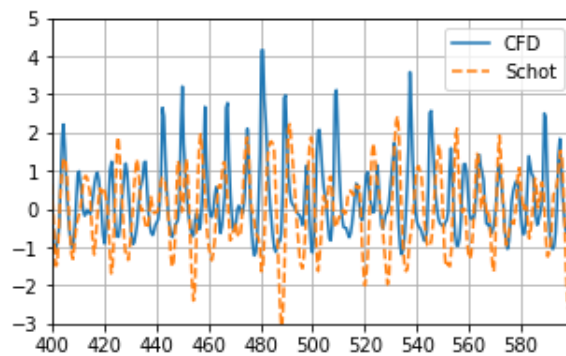


Figure 10.7: Elevation of the water surface at the toe of the dike in the time interval (400s, 600s) of the CFD model with the longer domain before the dike. It is compared with the elevation of the flume experiment ('Schot').

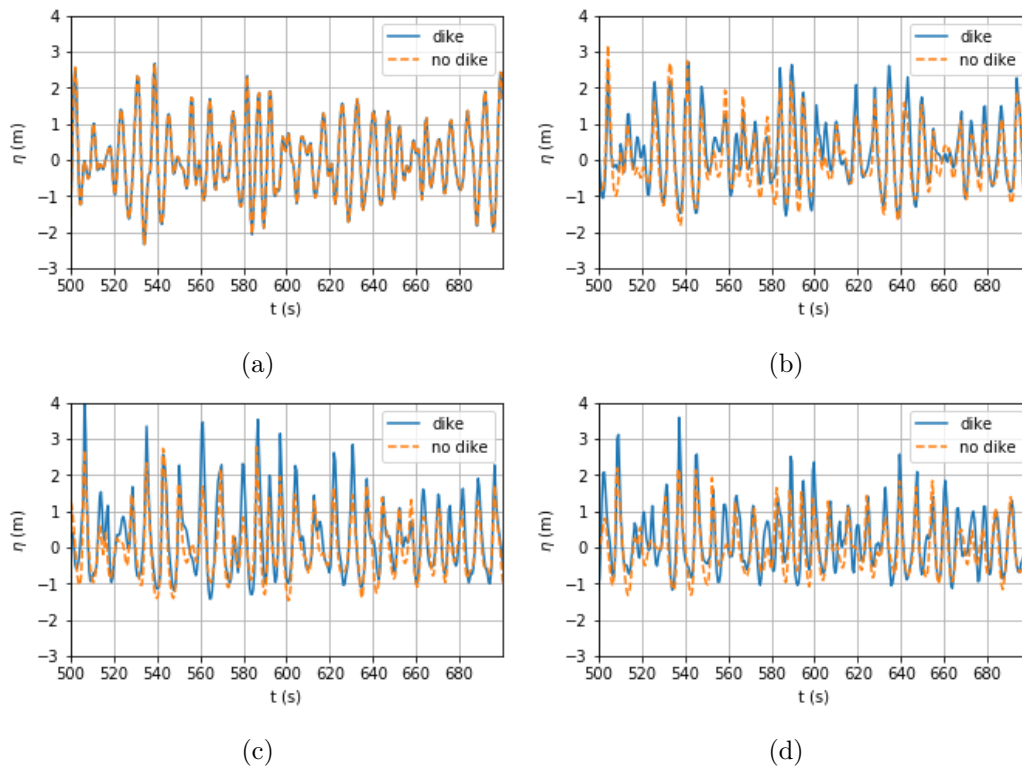


Figure 10.8: Comparison of the wave height in the **8a_long** model with and without dike structure at **a)** gauge 1, **b)** gauge 2, **c)** gauge 3 and **d)** gauge 4.

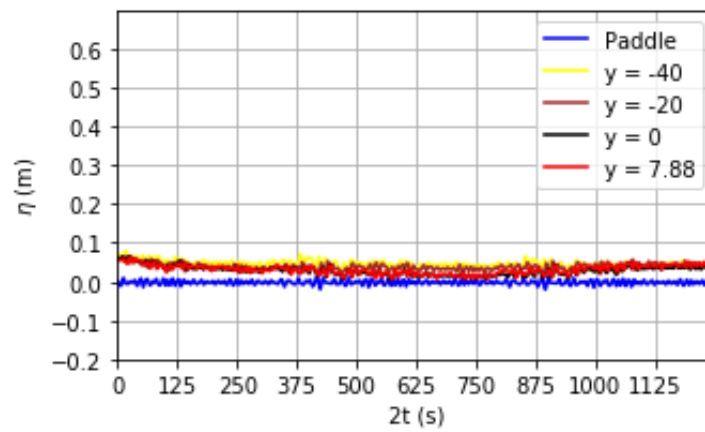


Figure 10.9: Sea water level of the **8a_long** model without dike.

Deep water

The most notable difference in the waves were the asymmetric high peaks. A possible explanation could be that the water is not deep enough. In the upcoming model, the bottom of the domain is lowered with 8 meters.

The graphs in Figure 10.10 and 10.11 show the new water level and wave height. Especially the sea water level has changed compared to the previous models. The wave setup is lower, but the waves still look very asymmetric. The overtopping is a somewhat higher than what is seen in previous models.

Name	t	H_s	T_p	SWL	K_s lower slope	K_s rest	q
8a_deep	960 s	4.22 m	7.69 s	5.3 m	0.5 m	0.05 m	170.2 L/s/m

Table 10.5: The resulting overtopping of the CFD model with a lower bottom of the domain, such that the water is deeper.

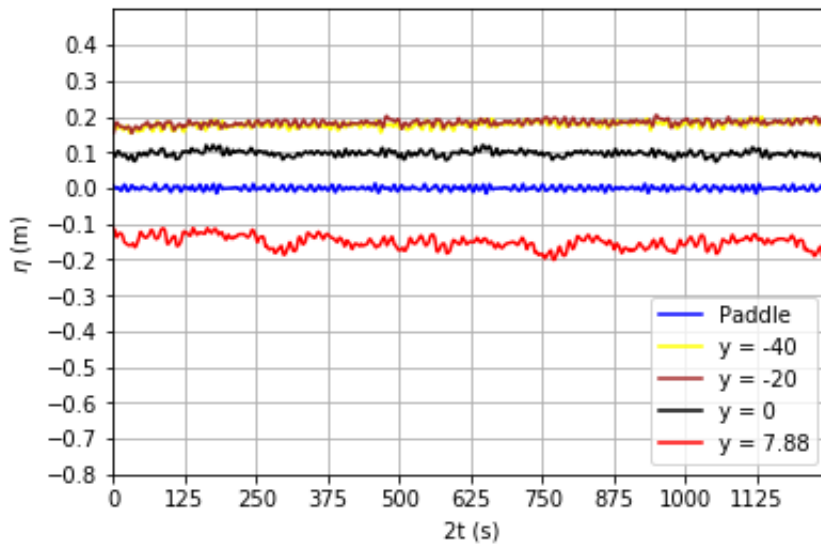


Figure 10.10: Sea water level of the **8a_deep** model.

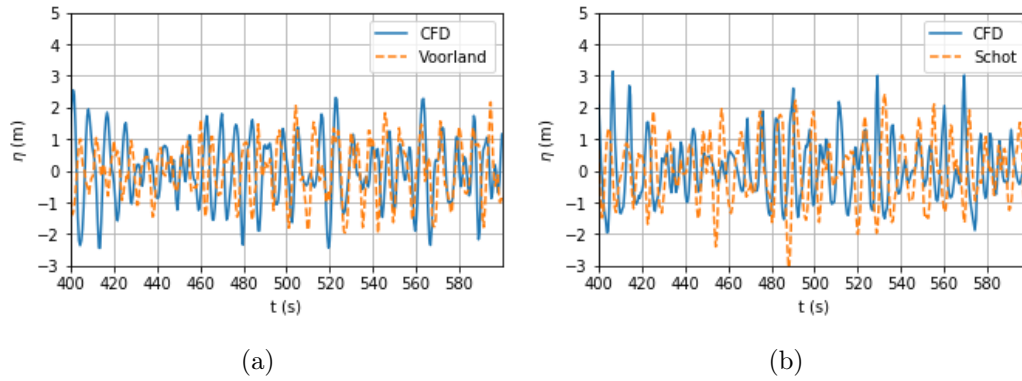


Figure 10.11: Comparison of the wave height from the CFD model with deep water and the Delta Flume experiment at **a)** $y = -30$ and **b)** $y = 0$.

10.3 Block profile

The shape of the level-blocks plays an important role in reducing the wave overtopping under normal conditions. There is a chance that the significance of the level-blocks, although submerged under the current conditions, is high as well. The pattern as shown in Figure 9.2 takes too much time to implement, so it has to be decided how to capture the particular shape of the blocks as good as possible.

Looking to the pattern of the level-blocks, their shape seems like stairs. Each block has a horizontal, vertical and slope surface, which are placed onto each other. The 'stairs' of level blocks that has been made in SALOME is shown in Figure 10.12.

The stairs profile replaces the smooth lower slope. The new surface carries a lot more

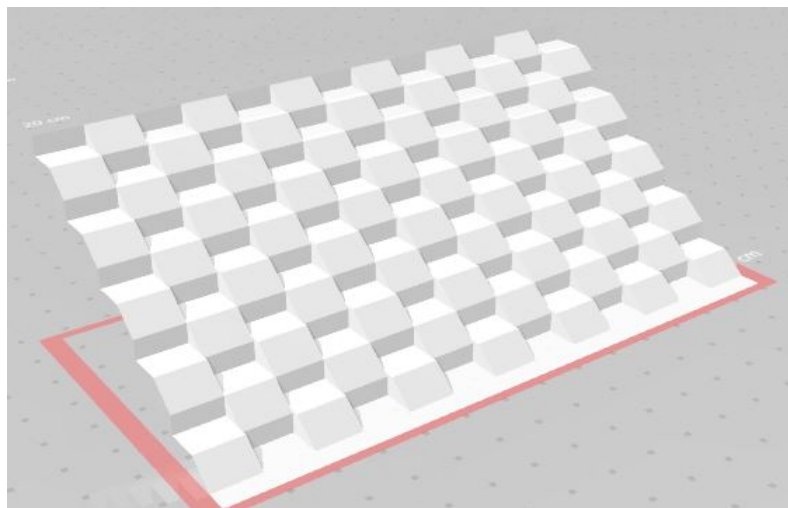


Figure 10.12: The pattern of the level-blocks is imitated with a 'stairs' profile.

complexity. The single sloped boundary is now a structure with many small boundaries. The resulting mesh will be affected by this, as more refinement is needed. Consequently, there were no boundary layers present on those small boundaries. The roughness parameter K_s will therefore not play a significant role in this model, so it will purely be the shape of the lower talud that causes the turbulence.

Besides that, the new lower slope is implemented without many disturbances. The resulting overtopping is given in Table 10.6.

Name	t	H_s	T_p	SWL	K_s lower slope	K_s rest	q
8a_levvelblocks	960 s	4.22 m	7.69 s	5.3 m	0.5 m	0.05 m	142.7 L/s/m

Table 10.6: The resulting overtopping of the CFD model with the block profile on the lower slope.

The overtopping still changes very little. It was expected from the literature like the EurOtop manual that submerged roughness elements will not have much effect, i.e. the roughness factor $\gamma_f = 1$. This single CFD model with the improvised rough lower talud shows similar results as the model with the high roughness parameter K_s , where $q = 142$ L/s per meter, although the significant wave height was higher in that model ($4.53m$ vs $4.22m$). This may mean that the roughness parameter K_s is too high on the lower slope, or that this simplified level-block structure is far from reality.

In the visualization of the turbulence around the lower slope, a new influence factor of the overtopping might have been found. In Figure 10.13, it is seen that the turbulence mainly stays around the lower slope. It looks like the turbulence acts as a wall and the flow must pass around it. As can be seen, this wall of turbulence has a slightly other slope than the original slope. This could be another reason why the simulation with the level-blocks produces similar results as the model with the high roughness factor K_s , although the wave height was higher on the latter model. Therefore it is worthwhile to investigate the effects of a slightly different angle on the lower slope.

10.4 Slope sensitivity

The previous models have created the suspicion that the angle of the lower slope has a large influence on the amount of water that flows over the dike. To investigate this, there have been performed two simulations with slightly different angles than the original angle of the lower slope. The toe of the lower slope has been translated $\pm 0.5m$ in the y-direction.

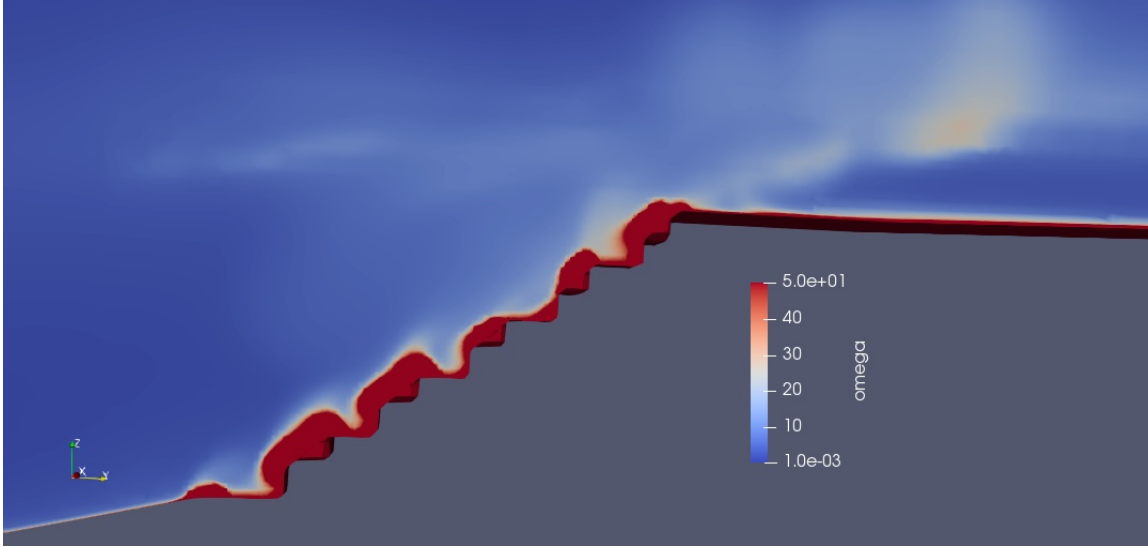


Figure 10.13: The turbulence, visualized by the variable ω , that forms a layer onto the lower slope.

Name	t	H_s	T_p	SWL	K_s lower slope	K_s rest	q
8a_steep	960 s	4.22 m	7.69 s	5.3 m	0.5 m	0.05 m	154 L/s/m
8a_flat	960 s	4.22 m	7.69 s	5.3 m	0.5 m	0.05 m	152.6 L/s/m

Table 10.7: The resulting overtopping of the CFD models with the steeper slope and the flatter slope. The toe of the lower slope is translated $\pm 0.5m$ in the y-direction.

Despite the suspicion, small changes in the angle of the lower slope do not lead to big differences in the overtopping. Although it is still not the breakthrough that has been looked for, another possibility is ruled out. Changes in the angle of the slope definitely affect the results, but the lower slope being submerged or the changes in slope are too small to explain why the overtopping does not change.

10.5 Scale effects

The last option for investigating the big difference in overtopping is to bring our model down to Delta Flume scale. Therefore, a change in variables must be made. The length and the time variables will be divided by 2.95 and $\sqrt{2.95}$ respectively. This leads to the parameters in Table 10.9.

The resulting q is also on Delta Flume scale. Therefore it must be multiplied with $2.95^{1.5}$ to compare with all the models on prototype scale. This results in an overtopping of 153.5 L/s per meter, similar to the prototype models.

Name	t	H_s	T_p	SWL	K_s lower slope	K_s rest	q
8a_scaled	558 s	1.53 m	4.49 s	1.8 m	0.17 m	0.017 m	30.3 L/s/m

Table 10.8: The resulting overtopping of the CFD model on Delta Flume scale (1:2.95).

10.6 Summary of the results

In this summary, a brief overview of all results in this chapter is given. The main results will be discussed in the discussion session (Chapter 12).

At first, several roughness configurations on the surfaces on the dike gave been tested. This was done by varying the parameter K_s (**8a_Ks0005** and **8a_Ks005**). Then, a closer look was taken to the waves that are generated in the CFD model. Here, the sea water level and the wave heights were compared with the Delta Flume experiment. This lead to some new CFD models: **8a_lowSWL**, **8a_long** and **8a_deep**. Here, it was also found that the wrong values for the wave height H_s and wave period T_p have been taken.

The structure was investigated next. Several changes were made in the geometry to see their effects on the overtopping. The lower slope has been replaced with a rough pattern to resemble the level blocks (**8a_levelblocks**) and the angle of that slope has been altered slightly (**8a_steep** and **8a_flat**). At last, the CFD model has been brought back to the scale of the Delta Flume. This is model **8a_scaled**.

Name	t	H_s	T_p	SWL	K_s lower slope	K_s rest	q
8a_rough	960 s	4.53 m	7.71 s	5.3 m	0.005 m	0.0005 m	165.9 L/s/m
8a_Ks0005	960 s	4.53 m	7.71 s	5.3 m	0.005 m	0.005 m	156.3 L/s/m
8a_Ks005	960 s	4.53 m	7.71 s	5.3 m	0.5 m	0.05 m	142 L/s/m
8a_lowSWL	960 s	4.53 m	7.71 s	5 m	0.5 m	0.05 m	119.5 L/s/m
8a_long	960 s	4.22 m	7.69 s	5.3 m	0.5 m	0.05 m	154.1 L/s/m
8a_deep	960 s	4.22 m	7.69 s	5.3 m	0.5 m	0.05 m	170.2 L/s/m
8a_levelblocks	960 s	4.22 m	7.69 s	5.3 m	0.5 m	0.05 m	142.7 L/s/m
8a_steep	960 s	4.22 m	7.69 s	5.3 m	0.5 m	0.05 m	154 L/s/m
8a_flat	960 s	4.22 m	7.69 s	5.3 m	0.5 m	0.05 m	152.6 L/s/m
8a_scaled	558 s	1.53 m	4.49 s	1.8 m	0.17 m	0.017 m	153.5* L/s/m

Table 10.9: The input parameters and the resulting overtopping of the CFD models in this chapter. *The overtopping of this model is projected to prototype scale.

11 Ribs

The implementation of ribs on the upper slope is the focus of this chapter. It contains the new geometry and the resulting level of overtopping. The original aim of the research was to eventually model this dike profile and to validate overtopping results of CFD with the corresponding Delta Flume experiment. When it became clear that the overtopping did not match, research shifted to understanding and solving the differences. Therefore this chapter will only briefly describe a CFD model that is executed with ribs.

11.1 Geometry

The geometry slightly changes with the incorporation of ribs. The ribs will be placed as shown in Figure 11.

A small change is made to the upper slope. In Figure 11 it is seen that the ribs are all $0.5m$ high and the base blocks of the lower slope are $0.3m$ high on the first rows and later $0.2m$ high. Instead of changing the height of the base of the lower slope, it is chosen to differ the height of the ribs. The first five rows of ribs were then placed at a height of $0.2m$ high and the all further ribs were $0.3m$ high. This way, the geometry was a little simpler to make.

In Figure 11.1 is shown how the rib pattern looks like after meshing with the *snappyHexMesh* tool. With the initially chosen width of $3.3m$, precisely one rib ($2.2m$) plus the area between each rib ($1.1m$) was present on each row. The ribs are not very sharp around the edges. The resulting mesh around the ribs seen from the side is shown in Figure 11.2. The boundary layers are present between the ribs, but not on the ribs itself. Further refinement to increase

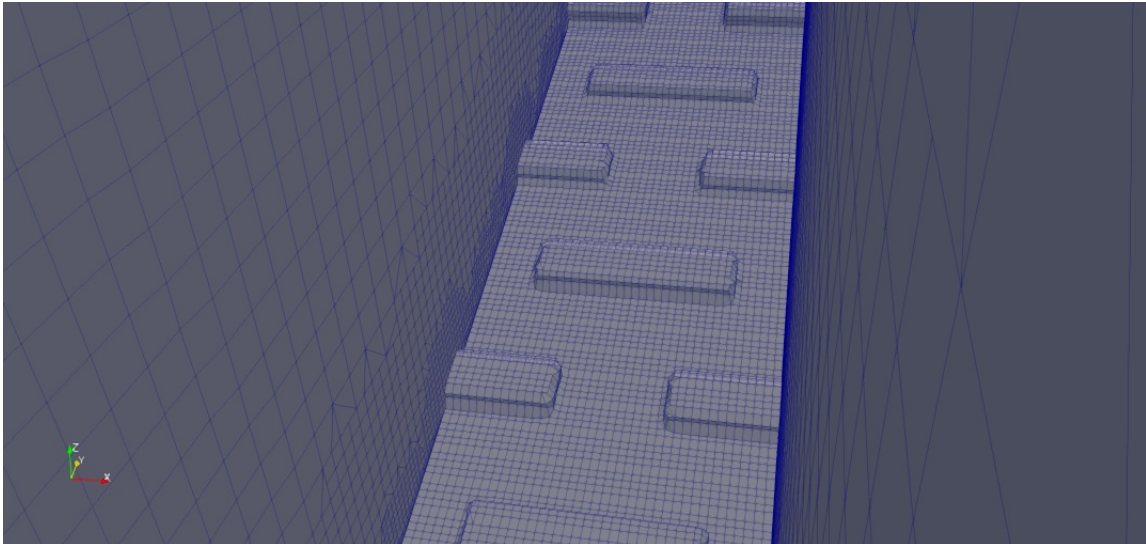


Figure 11.1: Top view of the ribs after meshing.



Figure 11.2: A cross-section of the mesh of the upper slope with the ribs.

the sharpness of the edges and to get boundary layers on the ribs did not take place. Because there was limited time and further refinement could increase the computation time greatly, it was chosen to leave the mesh in this state.

11.2 Results

In the Delta Flume experiments, one saw a very large reduction of the level of overtopping when the ribs were present. The ribs lead to a decrease in overtopping of approximately 85%. It is hard to say beforehand what level of overtopping is expected in CFD. Given the previous results however, it is not reasonable to think that the overtopping will decrease as drastic as in the Delta Flume, .

Name	t	H_s	T_p	SWL	K_s lower slope	K_s rest	q
8a_ribs	960 s	4.22 m	7.69 s	5.3 m	0.5 m	0.05 m	122.5 L/s/m

Table 11.1: The resulting overtopping of the CFD model with ribs on the upper slope.

The input parameters for the model are chosen similarly as in the last CFD models of the previous chapter. They are given in Table 11.1. The overtopping of $122.5L/s/m$ is close to the overtopping of the model with the lower sea water level.

In the upcoming discussion section, the results of the models will be discussed.

12 Discussion

The models have shown that CFD can implement a lot of complex structures and waves. Regarding overtopping however, CFD did not manage to produce comparable results to the Delta Flume experiments. Possible causes have been investigated that lead to the difference in overtopping. It seemed that the high, asymmetric waves that the results have shown are the main cause of the different results.

From the changes that were made, reducing the sea water level has a large impact on the total overtopping. A water level drop of 30 centimeters lead to an overtopping reduction of approximately 20% compared to all the other CFD models. Nevertheless, even this model produces two times as much overtopping than what was found in the Delta Flume.

The origin of the asymmetric waves are still unclear. In the generated waves at the inflow boundary of the CFD model, the asymmetry seemed not present yet. Further in the domain however, the peaks became much higher and the trough of the wave was close to the original sea water level. It is normal for waves to be a little asymmetric with a slightly higher peak compared to the trough, but the results show that especially the higher waves were also the most asymmetric ones.

The dike certainly had an effect on the incoming waves. Direct reflection and water that flowed back from the dike seem to have a big influence. When compared two identical models with and without dike structure, almost every peak of the waves with dike lay higher than the waves without dike. Sometimes the peaks differed more than one meter. It is hard to relate this to the Delta Flume experiment. Only data from two locations regarding the wave height were available. The data were compared with two locations in the CFD model, but the locations did not match perfectly.

Besides the drop of the sea water level, other changes in the original model had a less significant effect. As expected, increasing the roughness on the lower slope did not result in much less overtopping. The decreasing overtopping seems to be caused mainly by increasing the roughness K_s on the berm and upper slope as well. Also other changes to the lower slope did not result in less overtopping.

In the Delta Flume, ribs on the upper slope lead to a strong reduction of the level of overtopping. In CFD, the reduction was significant, but not nearly as strong as in the Delta Flume. The result fits in earlier conclusions that the waves behave different compared to the Delta Flume and this leads to the large overtopping value in CFD. Changes to the dike to reduce the overtopping seem inferior to the influence of the high, asymmetric waves.

The generalizability of the results is limited however by the number of CFD simulations

that have been executed. The impact of varying a parameter in the CFD model requires more proof to reliably state the precise effects of the changes. Also, the lack of implementing some features in waves2Foam have led to more complexity in obtaining and validating results. Moreover, valuable time has been invested in making these implementations work. Features such as imitating the waves from the experiment, the use of the overtopping function and using a permeable structure would have made this study better. In particular, using the same waves as the Delta Flume experiment would have given the possibility to compare individual waves.

Nevertheless, ruling out possibilities that lead to a higher overtopping gives also confidence that CFD is a very capable method to test designs of a dike. Especially, confirming that scaling and changes to submerged parts of the dike have no significant effect on the overtopping gives more value to CFD simulations. Furthermore, one can argue whether a factor of two to three above the actual overtopping is bad for the design of the dike. It would have been much worse if CFD models show a smaller amount of overtopping and one should design the dike on those overtopping values. The purpose of using CFD in dike testing should be to use it in the early stages, to get an indication on which overtopping to expect. Furthermore, effects of (small) changes in the dike could be tested faster and cheaper with CFD, instead of rebuilding the dike in the Delta Flume frequently.

From this study, one could expand it in several directions. At first, it is important to use identical waves as a flume experiment. Then it is clear whether the asymmetric waves come from the wave settings or that there is another cause that is related to the CFD configurations. When asymmetric waves are still present, a study could be conducted on where they come from. Is it caused by the dike or do the CFD features like the grid or boundary conditions play a more important role. One could also think of focusing only on implementing an exact copy of the dike in a CFD model. However, it is quite hard to implement all the features on the dike and to mesh it properly after.

13 Conclusion

This thesis aimed to show that CFD can be used for correctly estimate the overtopping of waves running up to a complex dike. The CFD software OpenFOAM together with the waves2Foam plugin showed to be a powerful tool in generating waves and research their behaviour. Although CFD did not succeed in producing similar results as in Delta Flume experiments, it gave a lot of insight in where the differences might come from.

The presented methods in this thesis helped to get a good understanding on many aspects in modelling waves. The main focus hereby lied on the mathematics and less on the hydraulics. The available data of Delta Flume experiments of the Afsluitdijk gave a big opportunity to test the capability and the focus points of using CFD to assess dike safety. Waves2Foam was the appropriate CFD software for this purpose, as it was open source and has proved its capability in generating waves and modelling free surface flows.

It was expected that the overtopping of the CFD model and the Delta Flume would agree a lot better. Although the input waves between both experiments were not identical, the irregular JONSWAP waves used in the two experiments had the same parameters. The difference in overtopping has shifted the focus at the end of the research. The original goal was to eventually analyze the roughness effects of ribs on the upper slope and use the particular Delta Flume experiment for validation. Instead, only the experiment with a smooth upper slope was used to calibrate the CFD model and the research focused on finding where the differences in overtopping came from.

Asymmetric waves around the sea water level seemed to be the main cause of a much higher overtopping in CFD simulations. Research has been conducted on how some waves evolved to extreme asymmetric high waves. Although there was no success in finding the cause, several new suggestions arose to focus on in further research. The main starting point is to produce the same waves as the ones generated in a flume experiment. Comparing individual waves could give a lot more valuable information, something that limited this study. Other recommendations would be to produce more models that focus on a few specific characteristics of waves and/or dikes, instead of a large variety of characteristics.

Although there is still a lot to research regarding the posed problem, this study gave insight on where to focus on when using CFD for assessing overtopping over a complex dike. It showed some main contributors to the level of overtopping and also showed which parameters have only minimal effect when one changes these. To conclude, this research took a step in the right direction for using CFD to determine the safety of a dike structure. Hopefully it motivates other researchers to improve CFD results in this type of hydraulics and share the results with the world.

References

- [1] Mohd Zaki Noor Irza Najafian Gholamhossein Abu Husain, Mohd Khairi. *Prediction of Extreme Offshore Structural Response: An Efficient Time Simulation Approach*. 07 2013.
- [2] J.A. Battjes. *Vloeistofmechanica b70*. TU Delft, 1990.
- [3] Alex Capel. *Wave run-up and overtopping reduction by block revetments with enhanced roughness*, volume 104. Elsevier, 2015.
- [4] N.C.Markato C.D.Argyropoulos. *Recent advances on the numerical modelling of turbulent flows*, volume 39(2). Elsevier, 2015.
- [5] E.B.L.Mackay. *Resource Assessment for Wave Energy*, volume 8. Elsevier, 2012.
- [6] L. H. Holthuijsen. *Waves in Oceanic and Coastal Waters*. Cambridge University Press, 2007.
- [7] Tobias Holzmann. *Mathematics, Numerics, Derivations and OpenFOAM®*. 2019.
- [8] Luofeng Huang. An opensource solver for wave-induced fsi problems. 2017.
- [9] N G Jacobsen, D R Fuhrman, and J Fredsøe. A Wave Generation Toolbox for the Open-Source CFD Library: OpenFoam®. *International Journal for Numerical Methods in Fluids*, 70(9):1073–1088, 2012.
- [10] B Jensen, N G Jacobsen, and E D Christensen. Investigations on the porous media equations and resistance coefficients for coastal structures. *Coastal Engineering*, 84:56–72, 2014.
- [11] Milovan Peric Joel H. Ferziger. *Computational Methods for Fluid Dynamics*. Springer-Verlag Berlin Heidelberg, 2002.
- [12] Lars Bergdahl Rickard E. Bensow Johannes Palm, Claes Eskilsson. Assessment of scale effects, viscous forces and induced drag on a point-absorbing wave energy converter by cfd simulation. *Journal of Marine Science and Engineering*, 6, 10 2018.
- [13] T. Bruce J. De Rouck A. Kortenhuis T. Pullen H. Schüttrumpf P. Troch B. Zanuttigh J.W. van der Meer, N.W.H. Allsop. *EurOtop Manual*. 2018.
- [14] Ingrid Koenen. Bam en van oord onthullen magisch concept robuuste afsluitdijk. 2018.
- [15] D.B. Launder, B.E.; Spalding. *The numerical computation of turbulent flows*, volume 3(2). Elsevier, 1974.
- [16] F.R. Menter. Two-equation eddy-viscosity turbulence models for engineering applica-

- tions. *AIAA Journal*, 32(8):1598–1605, 1994.
- [17] Moskowitz. Lionel Pierson, Williard J. Jr. *A Proposed Spectral Form for Fully Developed Wind Seas based on the similarity theory of S.A. Kitaigorodskii*, volume 69. 1964.
- [18] Ronald van Bochove. Vier jaar renoveren van de afsluitdijk. 2019.
- [19] Marcel van Gent, P Tonjes, H Petit, and Peter van den Bosch. Wave action on and in permeable structures. *Coastal Engineering Proceedings*, pages 1739–1753, 01 1994.
- [20] Marcel R.A. van Gent. *Oblique wave attack on rubble mound breakwaters*, volume 88. Elsevier, 2014.
- [21] Marcel R.A. van Gent. The new delta flume for large-scale testing. *Hydrolink*, (2):48–50, 2015.
- [22] A.E.P. Veldman. *Computational Fluid Dynamics*. Rijksuniversiteit Groningen, 2012.
- [23] Th. von Kármán. *Mechanische Ähnlichkeit und Turbulenz*.
- [24] David C. Wilcox. Formulation of the k-w turbulence model revisited. *AIAA Journal*, 46(11):2823–2838, 2008.

A Scripts

This appendix contains the base of the scripts that have been used to read and analyse the overtopping and surface elevation data.

A.1 Spectrum

```
#Import the necessary packages
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import scipy.signal
from scipy.fftpack import fft
from scipy.stats import chi2
from scipy.interpolate import interp1d
from scipy.signal import argrelextrema

#Navigate to the folder of the files
path = "D:/Users/SCHI6/Documents/Afstuderen/Data/"

#Import the surface elevation data
data_JONSWAP = np.loadtxt(path+"Waves2Foam/elevations/no_dike2.dat",
                          unpack = True, skiprows = 4)

#Reshape the surface elevation data
data_JONSWAP= data_JONSWAP.transpose()
data1 = pd.DataFrame(data = data_JONSWAP[0:,0:],
                    index = data_JONSWAP[0:,0],
                    columns = ["time", "y=-20", "y=-10",
                              "y=-30", "y=0", "y=7"])

sealevel = 5.13
x = np.linspace(0.013, 960, num=1770, endpoint=True)

#Substract the sea level from the water height
data1['y=-30'] = data1['y=-30'] - sealevel
data1['y=-20'] = data1['y=-20'] - sealevel
data1['y=-10'] = data1['y=-10'] - sealevel
```

```

data1['y=0'] = data1['y=0'] - sealevel
data1['y=7'] = data1['y=7'] - sealevel

#Divide the data into multiple blocks. Fs is the sampling frequency
Fs = 2
n = len(data1)
nfft = n/8

#Reshape the data
data_2 = scipy.signal.detrend(data1["y=-30"])
nBlocks = int(n/nfft)
data_new = data_2[0:nBlocks*nfft]

dataBlock = np.reshape(data_new, (nBlocks,nfft))

#Define the frequency axis
df = Fs/float(nfft)
f = np.arange(0,Fs/2+df,df)
fId = np.arange(0,len(f))

#Calculate the variance for each block and for each frequency using Fourier
#transformation
fft_data = fft(dataBlock,n = nfft,axis = 1)
fft_data = fft_data[:,fId]
A = 2.0/nfft*np.real(fft_data)
B = 2.0/nfft*np.imag(fft_data)

#Calculate the energy
E = (A**2 + B**2)/2
E = np.mean(E, axis = 0)/df

#Define the limits of the 90% confidence interval
edf = round(nBlocks*2)
alpha = 0.1
confLow = edf/chi2.ppf(1-alpha/2,edf)
confUpper = edf/chi2.ppf(alpha/2,edf)

```



```
#Plot the spectrum
plt.figure(figsize=(5,3))
plt.plot(f,E, color = 'blue', label = 'Experiment')
plt.title('Estimate of the spectrum')
plt.xlabel('frequency (Hz)')
plt.ylabel(r'Wave spectral density ( $m^2/s$ )')
plt.legend()
plt.show()
```

A.2 Overtopping

```
#Import the necessary packages
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

#Navigate to the folder of the files
path = "D:/Users/SCHI6/Documents/Afstuderen/Data/"

#Load the overtopping data at the first time step
overtopping = pd.read_csv(path + "Waves2Foam/overtopping/overtopping.0.csv")

#Select only the column that contains the overtopping
overtopping = overtopping[['alpha.water']]

#Load the overtopping at all the other timesteps and merge them
for x in range(1,960):
    overtopping2 = pd.read_csv(path+"Waves2Foam/overtopping/overtopping."+str(x)+".csv")
    overtopping = pd.concat([overtopping,overtopping2[['alpha.water']]],
                            ignore_index=True, sort = False)

#Calculate the overtopping
overtopping = overtopping.rename(columns = {'alpha.water':'alpha'})

overtopping['time'] = overtopping.index + 1
overtopping['q_total'] = 1000 * overtopping.alpha / 3.3
overtopping['q'] = overtopping.q_total / overtopping.time

#Plot the overtopping
plt.figure(figsize=(8,5))
plt.plot(overtopping.time, overtopping['q'],linestyle = 'solid',
         color = 'yellow', label = 'overtopping')
plt.xlabel(r't (s)')
plt.ylabel(r'q (L/s per m)')
plt.axis([0, 960, 0,1])
plt.xticks(np.arange(0,960,step = 120))
```

```
plt.yticks(np.arange(0,600,step = 60))  
plt.grid()  
plt.legend()  
plt.show()
```

A.3 Overtopping data Delta Flume

```
#Import the necessary packages
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

#Navigate to the folder of the files
path = "D:/Users/SCHI6/Documents/Afstuderen/Data/"

#Constant values
bodemWHM21 = 0.057
WHM21nul = -0.4795
l_bak1 = 12.4
b_bak1 = 5
helling = 3
l_bak2 = 5.005
b_bak2 = 5
schaal = 2.95

#Read the data from the Delta Flume experiments
overtopping = pd.read_csv(path + "Golfgoot/Overslag deel 1.asc",
                          delimiter="\t", header = 0)

#Substract the height of the water height meters
overtopping["waterstand1"] = overtopping.WHM21 - WHM21nul + bodemWHM21
overtopping["waterstand2"] = overtopping.WHM22 - WHM21nul + bodemWHM21

#Rescale the data and calculate the overtopping
overtopping["t"] = overtopping["T(ms)"] * np.sqrt(schaal) / 1000
overtopping["q1"] = (l_bak1 + 0.5 * helling * overtopping.waterstand1) * overtopping.waters
overtopping["q2"] = l_bak2 * overtopping.waterstand2
overtopping["totaal"] = 1000 * (overtopping.q1 + overtopping.q2) / (overtopping.t - 100) *
overtopping.iloc[0,[38]] = 0

#Calculate the running mean of the overtopping
overtopping["mean"] = overtopping["totaal"]
```

```
for row in range(5240,100000):  
    overtopping['mean'].iloc[row] = overtopping['totaal'].iloc[row-300:row+301].sum()/601
```

A.4 Water level

```
#Import the necessary packages
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
from scipy.interpolate import interp1d

#Navigate to the folder of the files
path = "D:/Users/SCHI6/Documents/Afstuderen/Data/"

#Import the surface elevation data
data_JONSWAP = np.loadtxt(path+"Waves2Foam/elevations/no_dike2.dat",
                          unpack = True, skiprows = 4)

#Reshape the surface elevation data
data_JONSWAP= data_JONSWAP.transpose()
data1 = pd.DataFrame(data = data_JONSWAP[0:,0:],
                    index = data_JONSWAP[0:,0],
                    columns = ["time", "y=-20",
                              "y=-10", "y=-30", "y=0", "y=7"])

sealevel = 5.3
x = np.linspace(0.013, 960, num=1770, endpoint=True)

#Substract the sea level from the water height
data1['y=-30'] = data1['y=-30'] - sealevel
data1['y=-20'] = data1['y=-20'] - sealevel
data1['y=-10'] = data1['y=-10'] - sealevel
data1['y=0'] = data1['y=0'] - sealevel
data1['y=7'] = data1['y=7'] - sealevel

#Interpolate the water level data
f30 = interp1d(data1['time'], data1['y=-30'], kind='cubic')
f20 = interp1d(data1['time'], data1['y=-20'], kind='cubic')
f10 = interp1d(data1['time'], data1['y=-10'], kind='cubic')
f0 = interp1d(data1['time'], data1['y=0'], kind='cubic')
```

```

f7 = interp1d(data1['time'], data1['y=7'], kind='cubic')

#Use the interpolated functions to find the elevation at the x-values
elevation20 = f20(x)
elevation10 = f10(x)
elevation30 = f30(x)
elevation0 = f0(x)
elevation7 = f7(x)

#Rename the columns
data1 = pd.DataFrame({'time': x,"y=-30": elevation30,"y=-20": elevation20,
                    "y=-10": elevation10,"y=0": elevation0,
                    "y=7": elevation7})

#Calculate the running mean of the water height
plt.plot(np.convolve(elevation20, np.ones((500,))/500, mode='valid'),
         color = 'blue', label = 'Paddle');
plt.plot(np.convolve(elevation10, np.ones((500,))/500, mode='valid'),
         color = 'yellow', label = 'y = -40');
plt.plot(np.convolve(elevation30, np.ones((500,))/500, mode='valid'),
         color = 'brown', label = 'y = -20');
plt.plot(np.convolve(elevation0, np.ones((500,))/500, mode='valid'),
         color = 'black', label = 'y = 0');
plt.plot(np.convolve(elevation7, np.ones((500,))/500, mode='valid'),
         color = 'red', label = 'y = 7.88');

```