Implementation of wave effects in the unstructured Delft3D suite

MSc Thesis

Delft, May 28, 2013

Author: B. Stengs
Email: basstengs@gmail.com

Committee:

Prof.dr.ir. M.J.F Stive	Delft University of Technology, HE-CE
ir. A.P. Luijendijk	Delft University of Technology /Deltares, HCO
dr.ir. M. Zijlema	Delft University of Technology, HE-EFM
Prof.dr.ir. J.A. Roelvink	Deltares/UNESCO-IHE
Prof.dr.ir. G.S. Stelling	Delft University of Technology, HE-EFM

Abstract

Recently an unstructured version of Delft3D, named D-Flow FM, has been developed at Deltares. This tool shows high potentials for coastal modelling due to the flexibility in domain resolution and unstructured grids. This MSc thesis focussed on implementing wave effects in D-Flow FM with satisfying results. In addition, a coupling system is built, enabling a flexible coupling between D-Flow FM and the wave module SWAN, for modelling the wave-current interaction. An innovative program-structure is presented which provides a dynamic coupling tool enabling interactive modelling.

This coupled tool shows large opportunities for coastal models where flexible resolution and geometry fitting are desired. Applications like the Sand Engine or the Wadden Sea modelling tidal velocity patterns at offshore regions and meandering gullies and rip channels at nearshore regions, all in one domain. These kinds of models are usually very computationally intensive, especially when morphodynamics are also included. With the introduction of unstructured modelling, coastal applications like the Sand Engine or the Wadden Sea, are modelled more efficiently.

The wave formulations implemented in D-Flow FM are state-of-the-art and correspond to the implementation as in Delft3D, which have been tested and validated with success. In the validation of wave modelling by D-Flow FM, four different test cases are considered. First, a simplified open channel flow is analysed, where the general performance of D-Flow FM is verified to the analytical solution. Second, an M2-tide for an alongshore uniform beach is modelled. Next, oblique incident waves are modelled on a planar beach. And finally, a non-uniform barred beach is modelled (Egmond case) with rip channels induced by waves. By means of these test cases the wave formulations in D-Flow FM are validated.

From the test cases, it is concluded that modelling waves in D-Flow FM generally gives good agreement with validated Delft3D models. The velocity patterns, both in magnitude and direction, show a good match, as confirmed by the Egmond case. The water levels modelled by the planar beach case match the Delft3D simulation as well. However, despite a good performance inside the surf zone, small model deviations are found outside the surf zone, at intermediate water depths. These model deviations should be reduced by correcting the bed shear stress for Stokes drift.

Past studies have proven the benefits of unstructured modelling over structured techniques, which shows high potential for D-Flow FM. With the inclusion of wave effects in D-Flow FM, improvement to nearshore modelling is made, but an extensive validation needs to be done. The dynamic software coupler provides an interactive modelling experience, and serves as a comprehensive tool for general model understanding. These developments have contributed significantly to improve nearshore modelling by D-Flow FM.

Acknowledgements

This MSc Thesis is made possible by the support and consult of many experts from Deltares. Thanks to the internship of eight months at Deltares prior to the graduation work I was able to push the limits of my MSc work. I would like to give special acknowledge to those who made this possible for me, especially Arjen Luijendijk.

The development of coupling D-Flow FM + SWAN was planned to be established as an alpha-version in November 2012 by DSC (Deltares Software Centre). Due to priority of developments in D-Flow FM (e.g. 3D-modelling, morphological modelling, parallelisation), wave modelling got postponed. Despite high workloads, the developers always managed to assist me when necessary. I want to thank Arthur van Dam for his advice and trust throughout the process. I give many thanks to Herman Kernkamp who has been patient with me for clarifying the Fortran code of D-Flow FM.

The Delft3D software has been an inspiration for the development of the wave formulation in D-Flow FM. I would like to give special thanks to Jan van Kester for his assist and knowledge contribution of the Delft3D applications. I want to thank Adri Mourtis, Jan Mooiman, Sander van der Pijl and Dano Roelvink for their support and consult during glitches and fatal errors.

Most of all I want to thank Fedor Baart for his assistance in programming and his large contribution to coupling D-Flow FM to SWAN, with innovative technics. Despite a doctoral thesis and becoming dad, he always managed to make time for (Fortran-, Python- or MATLAB-) programming both ends together. I have always enjoyed the endless sessions with Fedor Baart which were very pragmatic and enlightening. Many thanks.

Furthermore, I would like to thank my committee for their support and guidance throughout my MSc thesis. Arjen Luijendijk, my daily supervisor, who established the assignment and always managed to find time for bridging obstacles. Prof. Dano Roelvink, my expert at Deltares, for showing interest and giving consult in numerical modelling. Prof. Marcel Stive and Prof. Guus Stelling for supporting me on behalf of the TU Delft. Marcel Zijlema, expert at the TU Delft, for his review and assistance on behalf of the TU Delft.

Finally I thank my dear fellow student, Pieter Visser, for his review and his mental support throughout the curriculum. Last but not least, I thank my fellow students of the ‘Feestruimte’ for enjoying me with fascinating topics over a cup of coffee.

Contents

Chapter 1

Introduction

1.1 Background

In the study of hydrodynamic systems, the analytical solution of fluid motion is often unknown. This is where numerical models come into play by means of CFD modelling (Computational Fluid Dynamic). CFD models became invasive with the rise of computational power (e.g. Evans and Harlow, 1957). With modern technology the demand for high resolution model keeps rising. Structured grids are often used, based on quadrangles only. High resolution modelling is possible, but is less relevant at offshore regions. A rather new development is CFD modelling by unstructured grids, made from all kinds of polygons. This allows high flexibility for domain resolution which is much more efficient for CFD modelling (MacWilliams et al., 2007).

Deltares has recently developed the unstructured version of Delft3D, named D-Flow FM (D-Flow Flexible Mesh), for integrated modelling of rivers, coastal areas and estuaries. This program covers the modelling of flow patterns, morphological developments and ecology in an (unstructured) discretized way. Complex bottom patterns, such as shallow sandbanks, gullies and rip channels, can form difficulties for modelling with curvilinear grids due to the orthogonality requirements of the model. Whereas unstructured grids can be applied with much more flexibility and overcome these adverse effects relative easily (Kernkamp et al., 2011).

There is a strong demand in evaluating this expertise to a higher level of hydraulic engineering, as stated by Casulli and Walters (2000), Kernkamp et al. (2011) and Verwey et al. (2011). Coastal modelling belongs to one of these categories. For improving coastal models, an integration of wave modelling is required. Realizing this piece of engineering requires a validated coupling of D-Flow FM with SWAN (Simulating WAves Nearshore).

1.2 Research question and objectives

Unstructured modelling of coastal regions shows high potential in terms of computational efficiency. The flexibility in domain resolution enables a precise and flexible model focus. This has led to the following main research question:

“What developments could improve modelling nearshore hydrodynamics of complex coastal systems by D-Flow Flexible Mesh?”

The objectives of this research are:

- Analyzing applications of nearshore processes in numerical models
- Computational efficiency of numerical models
- Distinct and model tide-dominated systems versus wave-dominated systems
- Distinct and analyze fully unstructured grids versus curvilinear grids
- Analyze and develop improvements for comprehensive modelling
- Develop and extend D-Flow FM with wave effects
- Numerical accuracy with respect to the analytical solution

1.3 Approach

Essential elements for comprehensive numerical modelling are the physical interpretations of systems and the model response. A systematic approach is followed in this research which distinguishes four phases. At first, relevant nearshore processes are investigated. Next the general operation of Delft3D is analyzed and the implemented physical processes are studied. The same analysis is carried out for D-Flow FM and extended for numerical shortcomings. Finally, the extended formulations are tested relative to Delft3D and the analytical solution when possible.

The coherence of numerical modelling and physical modelling are a key element throughout the MSc work. These two lines of approach connect and collaborate via the model development and model validation. A fine collaboration of these two approaches is fundamental throughout the process.

1.4 Outline of thesis

This MSc thesis starts with a short introduction to relevant hydrodynamic processes for nearshore modelling in general. Hydrodynamics induced by tide, wind and waves are discussed. Next the numerical operation of D-Flow FM is highlighted and compared to Delft3D by means of two test cases.

After this research, D-Flow FM is extended with nearshore processes that were not included in the program. The robustness and performance of this piece of coding is tested by means of different test cases. Finally, conclusions and recommendations are drawn based on the performance of the test cases.

In the appendix of this report an introduction is given to the operation of Delft3D for nearshore modelling. Next a conceptual wave-coupling technique is discussed for full spectral modelling. Finally, a sensitivity analysis is carried out for modelling wave propagation in the nearshore.

Chapter 2

Modelling nearshore systems

2.1 Comprehensive modelling

Understanding system dynamics has been an endeavour for engineers since early history. Whether one considers a mechanical-, fluid dynamic-, metrological or electronic system – in theory the model schematization of these systems are similar. A fundamental aspect in this schematization is whether the representation of the model, all-embraces the physical processes of the system dynamics.

Here for, comprehensive knowledge is required of physical processes in the system before a model is set up. The relevant contribution of certain system processes determines whether or not such processes should be accounted for in the model. This schematization is an important procedure to achieve a sufficient model.

The response of physical systems can properly be described by mathematical expressions, especially in system dynamics. These systems are described by a rate of change through space or time. In mathematics this is described as a small unit change per small step through the considered dimension. By taking the limit to zero for this small step, gives a mathematical expression which is called a differential equation.

A relative simple physical system is described by a relative simple mathematical expression. For such systems it is sometimes possible to solve the expression in an analytical way. These analytical models are divided into elemental parts or basic principles. This allows a direct understanding of the systems behaviour in terms of dependency of parameters and processes. If a system is influenced by multiple (complex) relevant processes, it gets rather hard to find an analytical expression for the system. That is where the numerical models come into play.

A numerical model is a very handy tool for solving complex mathematical processes, but it does not give a comprehensive knowledge of the system at first sight. There is no one-to-one mapping possible of the system dependency. Another drawback is the numerical inaccuracies that arise due to truncation errors and convergence errors to the analytical solution (Zijlema, 2011). These inaccuracies require a certain tolerance of acceptance before making the numerical model operational. These drawbacks show the weakness and reliability of numerical models.

2.2 Physical processes in the nearshore

As stated in the previous section, a good understanding of relevant processes in general physical systems is of crucial importance for modelling. This section describes physical processes on the scale of the continental shelf, to nearshore hydrodynamics in general coastal regions. The contribution of tide, wind and waves to nearshore hydrodynamics are discussed.

2.2.1 Tide-induced hydrodynamics in the continental shelf and the nearshore

Tides in the nearshore are known as the rise and fall of coastal waters due to the gravitational pull of astronomical spheres. Both the rotation of the moon around the earth and the earth around the sun are the main driving forces of the astronomical tides. These two rotating systems both induce a centripetal and centrifugal force on the ocean waters. Considering the moon-earth system this induces two water bulges on either side of the earth. Due to the presence of landmasses and the tilting of the earth's axis (23.5° ± 5°), these two water bulges are not evenly distributed over the earth's surface. This could lead to observe twice a day high tide and low tide on the same spot (semi-diurnal tide), and on another spot only once a day the rise and fall of the tide (diurnal tide). Due to the presence of these landmasses resonance of the tidal components is also possible, in large shallow basins for example, which could introduce higher harmonics in the tidal signal (overtides) and/or amplification of the tide.

The relation to the frequency of the moon-earth system, the frequency of the earth-sun system and the frequency of the earth's rotation, determines the astronomical tide. When these frequencies align spring-tide will occur, and opposite when these frequencies are out of phase neap-tide occurs. There is a short phase shift of these spring-neap frequencies in relation to the astronomical frequencies due to the traveling time of the tidal wave from the free 'forcing tidal wave' near the Antarctic.

The rise and fall of the astronomical tide is called the vertical tide. The tide-induced hydrodynamics are called the horizontal tide. Considering a small tidal amplitude compared to the water depth and a more or less horizontal bottom slope leads to the following reduced shallow water equations:

$$\frac{\partial u}{\partial t} - fv = -g\frac{\partial \zeta}{\partial x} - \frac{\partial \tau_{bx}}{\rho h} \qquad \text{Eq. 2-1}$$

$$\frac{\partial v}{\partial t} + fu = -g\frac{\partial \zeta}{\partial y} - \frac{\partial \tau_{by}}{\rho h} \qquad \text{Eq. 2-2}$$

This expression describes a balance between inertia, Coriolis, the pressure gradient and bottom friction. The tidal wave deflects, in relative shallow basins, due to the Coriolis effect. In the Northern hemisphere this deflection is to the right due to the difference in the earth's orbital velocity as a function of latitude. Considering the North Sea basin, this results in a tidal wave originating from the Atlantic and traveling along the coastlines of the basin. This kind of wave is called a Kelvin wave.

Along the Dutch coast the water depths are considerably shallow, i.e. small depths compared to the length of the tidal wave. The phase velocity of the tidal wave becomes a direct function of the water

depth and bottom friction, according Eq. 2-1 and Eq. 2-2. Neglecting bottom friction for simplicity, the expression for wave propagation speed is as follows:

$$c = \sqrt{g(d+\zeta)} \qquad \text{Eq. 2-3}$$

The water depth is here expressed as $h = d + \zeta$. Considering a constant depth (d), shows a propagation speed which is directly related to the amplitude of the wave: $c = c(\zeta)$. Because the crest of the tidal wave travels faster than the trough, a pitched forward shape of the tidal wave results. This is translated (by means of Eq. 2-3) to an asymmetry in the horizontal tide with higher flood velocities (northward along the Dutch coast) and smaller ebb velocities (southward).

2.2.2 Wind-induced hydrodynamics in the continental shelf and the nearshore

Wind and low atmospheric pressure systems can be a very important factor to account for in coastal regions. A low atmospheric pressure system can locally 'tilt' (surge) the water level up to several meters. These scenarios are often associated with strong winds inducing an even higher water level setup near the coast. This makes storm surges important to account for in coastal modelling.

The wind exerts a shear stress on the water and induces a setup of the water level. This is expressed in the equilibrium situation as:

$$\rho g h \frac{\partial \bar{\zeta}}{\partial x} = \tau_{wind,x} \qquad \text{Eq. 2-4}$$

with:

$$\tau_{wind} = C_d \rho_a W^2 \qquad \text{Eq. 2-5}$$

In this equation W represents the wind velocity and C_d the wind-stress coefficient which is determined by Wu (1982). Eq. 2-4 shows that the water level setup due to wind shear stress is inversely proportional to the water depth, which gives larger water level setups in shallower regions (the nearshore). This water level difference is compensated by a return flow in the deeper parts of the water column.

2.2.3 Wave-induced hydrodynamics in the nearshore

"When I stand on the beach and peer at the waves, it amazes me the oceans-state I'm seeing is formed by storms many miles offshore." (Leo H. Holthuijsen)

Ocean waves formed by offshore storms and approaching the coast are influenced by many different processes. In this thesis the process of shoaling, refraction and wave breaking is highlighted in short for an understanding of the mean mechanism behind wave-induced hydrodynamics in the nearshore. The nearshore is considered to be the region from the beach to the offshore depth where the orbital wave motion of the water is not influenced by the bottom anymore. After a theoretical introduction of ocean waves, the nearshore hydrodynamics induced by wind-waves are divided into cross-shore and alongshore components.

Physics of waves

Ocean waves generated by a storm can be described by a statistical representation of the sea state by means of spectral analysis. Assuming an offshore stationary wave record (typically of 15-30min) of waves which are not too steep, allows a stochastic description of the surface elevation $\left(\underline{\eta}(t)\right)$. This is done by means of Fourier series which takes the sum of a large number of harmonic wave components each propagating with different amplitudes, frequencies and phases. From this random-phase/amplitude model the variance density spectrum for frequency and direction can be found, representing a statistical analysis of the sea state:

$$E(f,\theta) = \lim_{\Delta f \to 0} \lim_{\Delta\theta \to 0} \frac{1}{\Delta f\, \Delta\theta} E\{½\underline{a}^2\} = \left[{}^{m^2}/_{Hz}/_{rad} \right] \qquad \text{Eq. 2-6}$$

Integrating the two-dimensional spectrum $E(f,\theta)$ over all frequencies and over the full 2π directional domain, gives a the time-averaged surface elevation squared which is also known as the total variance:

$$\overline{\underline{\eta}^2} = \int_0^{\infty} \int_0^{2\pi} E(f,\theta)\, d\theta\, df \qquad \text{Eq. 2-7}$$

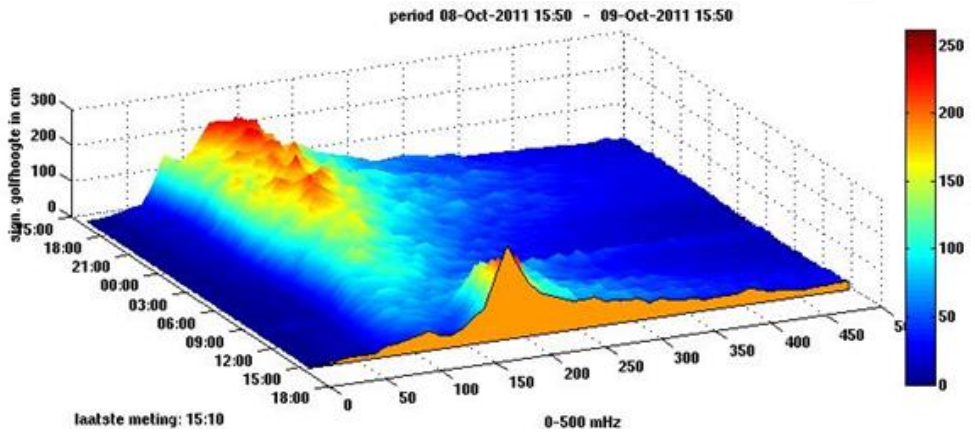


Figure 2-1: Measured wave spectrum near IJmuiden expressed in H_s and time. (Source: http://www.meetadviesdienst.nl)

The orbital motion of the water particles under a propagating deep water (linear) wave can be considered as closed circles from a Eulerian perspective. This circular motion decays exponentially downward over the water column. When waves approach the shoreface, the orbital motion of water particles under waves start to touch the seafloor and become closed ellipses instead of closed circles. Time averaging these orbital motions gives a net current velocity of zero.

The propagation speed of a surface wave slows down as it approaches the nearshore. This propagation speed is called the phase speed of a wave and is best described by the dispersion relationship for arbitrary depths:

$$c = \frac{g}{\omega}\tanh(kd) \qquad \text{Eq. 2-8}$$

with,

$$\omega = \frac{2\pi}{T} \qquad k = \frac{2\pi}{L} \qquad \text{Eq. 2-9}$$

where ω represents the radian frequency and *k* the wave number. From Eq. 2-8 one can concluded that the phase speed of a surface wave depends on the wave length and the frequency. For deep water ($kd \to \infty$) Eq. 2-8 becomes:

$$c_0 = \frac{g}{2\pi}T \approx 1.56T \qquad \text{Eq. 2-10}$$

And in very shallow waters ($kd \to 0$) then ($\tanh(kd) \to kd$) and Eq. 2-8 becomes:

$$c_{shallow} = \sqrt{gd} \qquad \text{Eq. 2-11}$$

In according to Eq. 2-10 the phase speed of a deep water wave is a function of the wave period only. By means of spectral analysis, a sea state can be considered as a large sum of harmonic waves, each with a different wave amplitude, wave period and phase. By Eq. 2-8, waves with a longer wave period travel faster than waves with shorter wave periods. In space, the longer waves (swell waves) filter from the short waves and are much more regular and long crested.

Thus, in according to the dispersion relationship, the wave frequencies sort and group together over space. Summating two of these waves with slightly different frequencies will reinforce at a certain moment and somewhat later dampen out (e.g. two people clapping non-synchronic). This forms a longer wave with its crest at the reinforced moments, and a trough when they cancel out. These long waves are called infra-gravity waves and are bound by the wave groups (phase locked), see Figure 2-2. Bound long waves travel with the speed of the group velocity, which is defined as:

$$c_g = nc \qquad \text{Eq. 2-12}$$

where *c* is the phase speed of the individual waves defined by Eq. 2-8 and,

$$n = \frac{1}{2}\left(1 + \frac{2kh}{\sinh 2kh}\right) \qquad \text{Eq. 2-13}$$

which is considered to be a relative deepness parameter of the incoming waves with wave number k.

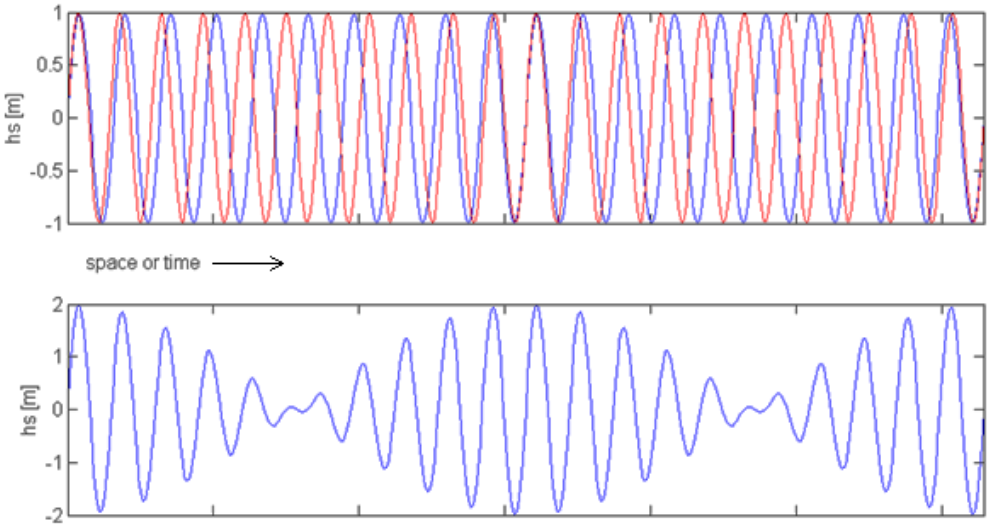


Figure 2-2: Superposition of two harmonics with slightly different frequencies

Inside the surf zone, where the short waves break, the bound long wave gets released from the group and is no longer phase locked. The bound long wave becomes a free long wave (Van Dongeren et al., 2003).

The wave period of infra-gravity waves vary from 30sec – 5min. Under its crest the wave energy at the coast is much larger than under the trough. This varying wave forcing over time in the nearshore induces a difference in water level setup, called surfbeat. The wave-induced forces are by definition related to the radiation stresses under the waves by:

$$F_x = -\frac{\partial S_{xx}}{\partial x} - \frac{\partial S_{xy}}{\partial y} \qquad \text{Eq. 2-14a}$$

$$F_y = -\frac{\partial S_{yy}}{\partial y} - \frac{\partial S_{yx}}{\partial x} \qquad \text{Eq. 2-14b}$$

Radiation stress is the wave-induced momentum flux, by the particle bodily motions and wave-induced pressures. Wave forces arise thus from a change in the transfer of momentum due to waves from one location to another (Longuet-Higgins and Stewart, 1964). The radiation stresses in Eq. 2-14) are defined as:

$$S_{xx} = \left(n - \frac{1}{2} + n\,cos^2\theta\right)E$$

$$S_{yy} = \left(n - \frac{1}{2} + n\,sin^2\theta\right)E \qquad \text{Eq. 2-15}$$

$$S_{xy} = S_{yx} = n\,cos\theta\,sin\theta\,E$$

with the time averaged wave-induced energy (E), defined as:

$$E = \frac{1}{8}\rho g H^2 \qquad \text{Eq. 2-16}$$

Cross-shore hydrodynamics

When waves travel from the deep waters to the nearshore they start to deform due to the influence of the bottom topography. The waves start to shoal in decreasing water depths, and non-linear effects force a skewness of the wave. Approaching even shallower waters, the crest of the wave starts to take over the trough of the wave, in according to the dispersion relationship of Eq. 2-8. This results in a pitched forward shape of the wave until the wave breaks. The kinematic and potential energy of the wave is dissipated by the roller energy and turbulence.

The wave-induced energy is directly related to the wave forces. One can expect that spatial varying wave energy in the cross-shore induces a difference in wave forcing. In the cross-shore, this spatial varying wave force finds equilibrium with the pressure gradient. This results in a water level set down just outside the surf zone, and a water level setup inside the surf zone.

To comply with mass conservation during the different stages of wave deformation, hydrodynamics in the nearshore are triggered in the water column (2DV effects). Outside the breaker zone wave skewness drives an onshore current near the bed. After wave breaking the mass of the rolling face is compensated by the undertow, which is an offshore directed current. Inside the breaker zone the infra-gravity waves drive an onshore current near the bed. Outside the breaker zone this current is offshore directed (Roelvink and Stive, 1989).

Alongshore hydrodynamics

Ocean waves that approach the coast under an angle (oblique incident waves), are an important driving force of alongshore hydrodynamics. Tides and wind-induced hydrodynamics also have their contribution, especially in the outlet of lagoons and shallow waters respectively. For the sake of simplicity an alongshore uniform coast is considered with oblique incident waves – the $\partial/\partial y$ terms in Eq. 2-14) become zero.

Oblique incident waves refract when entering shallow waters, in according to Snell's law. The wave rays bend towards shallow bottom contours. The angle of incidence (θ) thus converges to zero when waves enter shallow waters.

From Eq. 2-15 is found that the radiation stresses due to waves are a function of the angle of incidence, the relative depth and the wave height squared. These parameters become an important measure for alongshore wave forcing. This alongshore force finds equilibrium with the bed shear stress outside the surf zone for an infinitely long coastline. Inside the surf zone can be found, by means of Snel's Law and conservation of energy principle, that the alongshore driving force becomes a function of the wave energy dissipation, or wave breaking. By i.a. (Reniers, 1999) was found that the maxima alongshore current velocities for oblique incident waves are found at the breaker line(s), i.e. near the bars crest and another maxima close to the shore. These results are shown in Figure 2-3.

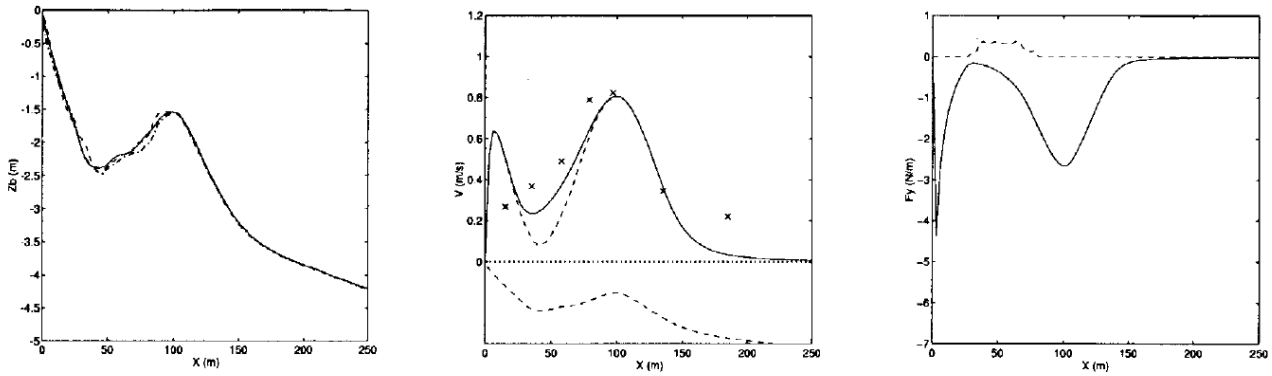


Figure 2-3: Alongshore forcing over a uniform barred beach, with local pressure gradient (solid line) and without (dashed line). Measured data is denoted by 'x' (Reniers, 1999)

Modelling wave-driven alongshore currents on a uniform sloping beach, generally show good agreements with laboratory data and field data. For a barred beach this is not generally the case, i.e. shear instabilities of the alongshore currents show difficulties in modelling (Reniers, 1999). This is especially the case with alongshore inhomogeneities like coastal structures, bars, rip channels, and gullies. Due to these 3D features, complex current patterns may arise which are challenging to translate into the laws of physics.

2.3 Introduction to numerical modelling

In the expression of the already simplified shallow water equations, Eq. 2-1 and Eq. 2-2, the relative important term of the bed shear stress is included. The bed shear stress is defined as:

$$\tau_b = c_f \rho U^2 \qquad \text{Eq. 2-17}$$

This expression is proportional to the velocity squared, which makes the differential equations Eq. 2-1 and Eq. 2-2 non-linear. This makes solving Eq. 2-1 and Eq. 2-2 analytically nearly impossible without simplifications. Solving non-linear equations like Eq. 2-1 and Eq. 2-2 efficiently requires numerical analysis.

The conventional numerical discretization is based on quadrangles (structured), where a rectangular or curvilinear grid is fitted to the area of interest. Examples of structured models are Delft3D, WAQUA (deGoede, 2011) and TRIWAQ. Despite all their capabilities, modelling meandering river channels or convex coastlines (for example) is not an easy task, from the users' perspective. In addition, the flexibility of domain resolution in terms of detailed modelling is limited. These drawbacks are often overcome with techniques like domain-decomposition, nesting routines, quadtree cells or special "cut-cell" treatments (Kirkpatrick et al., 2003).

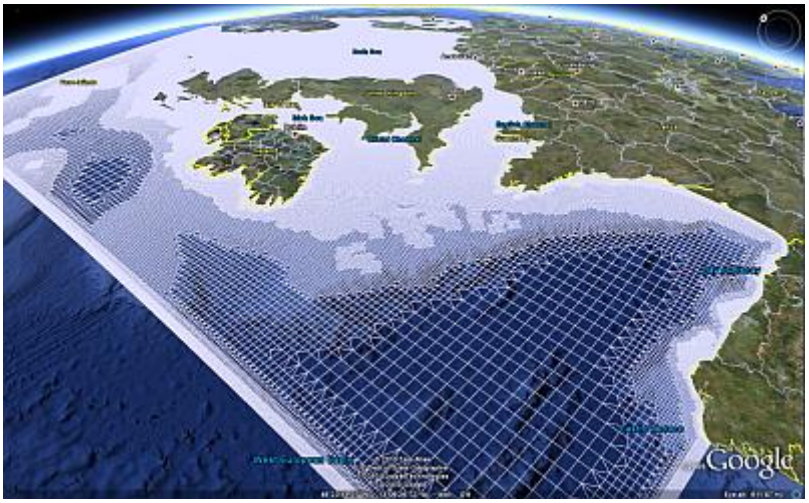


Figure 2-4: Example of a flexible mesh from the Continental Shelf Model

A rather new development is CFD modelling of unstructured grid, also called a flexible mesh (FM), made from polygons instead of only quadrangles. This enables much more flexibility for refining the resolution within one domain (Casulli and Walters, 2000). If desired, locally high resolution can be applied in modelling vast areas, as shown in Figure 2-4.

These advantages of unstructured models where recognized by many researchers whom contributed to the development of unstructured CFD modelling. Examples of unstructured models are ADCIRC (Dietrich et al., 2011), TELEMAC (Jones and Davies, 2006), UnTRIM (Casulli and Zanolli, 2002; MacWilliams et al., 2007) and MIKE Flexible Mesh (Sørensen, 2004). Deltares has recently been working on an unstructured model of Delft3D, named D-Flow FM. Primary test cases found positive results with a need to further development (Kernkamp et al., 2011).

The objective of this thesis is to extend flexible modelling, by means of D-Flow FM, to coastal regions with wave-dominated systems.

Chapter 3

Operation of D-Flow Flexible Mesh

3.1 Introduction

Since 2010, Deltares has been working on a new computational solver for unstructured grids. This numerical engine is called D-Flow FM and is mainly based on the original Delft3D module. Primary tests show promising results for the development of computational modelling. In according to Casulli and Walters (2000), Kernkamp et al. (2011) and Verwey et al. (2011), there is a strong demand in evaluating the expertise of unstructured modelling to higher levels of hydraulic engineering.

3.2 Spatial integration

The governing model equations of D-Flow FM are of the same type as Delft3D, namely the shallow water equations (SWEs). The computational solver however, is in a total different form. The integration of the governing equations is over the flow links, rather than in the conventional m-, n-dimensions. The velocities are expressed by the depth-averaged horizontal velocity vector $(\vec{u})$. This makes the 2DH (2 Dimensional in the Horizontal) spatial integration one-dimensional. The SWEs of Eq. A-6 until Eq. A-8 are rewritten to:

$$\frac{\partial \zeta}{\partial t} + \nabla \cdot (H\vec{u}) = q \qquad \text{Eq. 3-1}$$

$$\frac{\partial \vec{u}}{\partial t} + adv(\vec{u}) + g\nabla\zeta + \frac{1}{\rho}\nabla\langle\overrightarrow{\tau_b}\rangle + 2\Omega \times \vec{u} = S \qquad \text{Eq. 3-2}$$

In which $\nabla \equiv \left[\partial_x, \partial_y\right]^T$ is the horizontal gradient operator, q and S representing the source term and external forcing respectively, $adv(\vec{u})$ is the advection term, Ω is the angular velocity of rotation of the earth and H is defined as,

$$H = \zeta - d \qquad \text{Eq. 3-3}$$

with ζ the water level relative to a reference plane and d the bottom level positively upward, in contrast to Delft3D (see Eq. A-14). The SWEs of Eq. 3-1 and Eq. 3-2 express conservation of mass and momentum (Kernkamp et al., 2011).

The same staggered grid principle as Delft3D is used, but now in terms of flow nodes and net nodes (Casulli and Zanolli, 2002). The nodes are connected by links. The net nodes are defined on the corners of the grid cell. The flow nodes are defined at the *cell circumcenter*; the intersection of all perpendicular bisectors of the grid cell. A perpendicular bisector is found by drawing two circles with the net nodes as center and connecting the two intersections of the circles (see Figure 3-1).

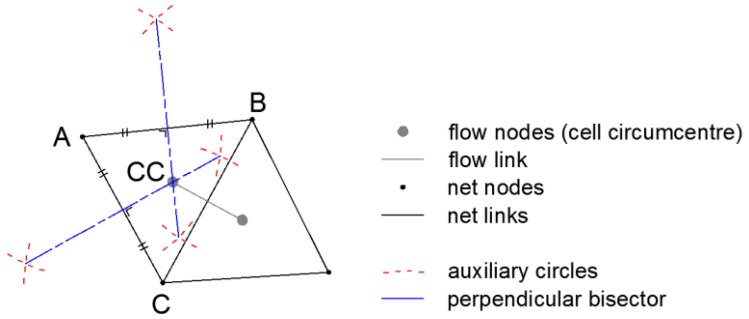


Figure 3-1: Orthogonal grid definition in D-Flow FM

Mathematically the coordinates of the cell circumcenter $[CC_x; CC_y]$ is found by the expression:

$$CC_x = 2\{(A_x^2 + A_y^2)(B_y - C_y) + (B_x^2 + B_y^2)(C_y - A_y) + (C_x^2 + C_y^2)(A_y - B_y)\}/D \quad \text{Eq. 3-4a}$$

$$CC_y = 2\{(A_x^2 + A_y^2)(C_x - B_x) + (B_x^2 + B_y^2)(A_x - C_x) + (C_x^2 + C_y^2)(B_x - A_x)\}/D \quad \text{Eq. 3-4b}$$

with;

$$D = 2\{A_x(B_y - C_y) + B_x(C_y - A_y) + C_x(A_y - B_y)\} \quad \text{Eq. 3-4c}$$

At the flow nodes the water levels are defined and at the net nodes the depth points are imposed. The velocity points are defined at the cell faces, along the flow links. This gives a similar staggered grid principle as Delft3D but now the spatial integration is in one-dimension, namely along the flow links.

In Delft3D, the water level in a single cell and depth values at the cell's interface are determined by the DPSOPT and DPUOPT respectively. In D-Flow FM this option is determined by choosing a BotLevType. The bottom level at the velocity points ($d_j^{\vec{u}}$) are found by the minimum-, mean- or maximum depth value of the net nodes (d_j). And then (by default) the bottom level at the water level points (ζ_j) are determined by the lowest connected link of $d_j^{\vec{u}}$.

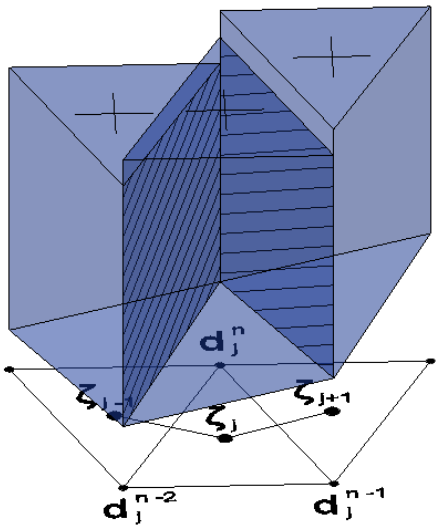


Figure 3-2: Example of 2DH cells defined on an unstructured grid with a sloping bathymetry

3.3 Time integration

The Delft3D time integrator uses the ADI principle over n- and m-space. Due to the one-dimensional space integration of D-Flow FM, the ADI principle cannot be applied. The implicit θ-method is used instead for both Eq. 3-1 and Eq. 3-2, which follows the approach of Kramer and Stelling (2008) except the advection terms are solved explicitly (upwind). This finite volume approach guarantees the conservation of momentum.

To assure numerical stability, the Courant number (CFL condition) may not be chosen too large due to the explicit part of the numerical solver. Defining the numerical time step by means of the CFL-criteria, shows to be efficient for unstructured grids due to the varying flow links (Kernkamp et al., 2011). With λ as a dimension of space, the Courant number is defined as:

$$CFL = \sqrt{gH}\frac{\Delta t}{\Delta \lambda} \qquad \text{Eq. 3-5}$$

After discretizing Eq. 3-2, u^{t+1} can be expressed in terms of the water level (ζ) and substituted back into Eq. 3-1, leaving a system of equations with only the water levels as unknown. Eq. 3-1 transforms into a matrix vector system of the type $\mathbf{Ax} = \mathbf{b}$, with $\mathbf{x}$ representing the unknown water level vector at the new time level and matrix $\mathbf{A}$ a sparse symmetric diagonally dominant matrix (Casulli and Walters, 2000). The full discretization of Eq. 3-1 and Eq. 3-2 to a system of equations in matrix form, can be found in Kernkamp et al. (2011).

Per time step the computation is divided into sections and solved parallel, through the use of OpenMP. In this sense the computational workload is divided over multiple cores of the processor, providing high efficiency to the available computational capacity. Conventional models like Delft3D are not supported with OpenMP and are thus based on single core computations.

3.4 Mesh (Unstructured)

Complex bottom patterns, such as shallow sandbanks, gullies and rip channels, can form difficulties for modelling with curvilinear grids due to the orthogonality requirements of the model. Structured modelling of deltas and estuaries are even more challenging. Applying unstructured grids for such models allows much more flexibility to the user, which may result in higher model-accuracy, as stated by Verwey et al. (2011).

A flexible mesh is made of polygons like triangles, pentagons, etc. Rather than conventional models whom are limited by rectangles. Due to the flexibility in resolution, nested models are no longer required. Here for, points of interest with different length scales are simulated with one single mesh. This is economically interesting due to a reduction in computation time. Another benefit of FM modelling is the ability to merge several grids (3D, 2D and 1D) to one mesh. For example a delta network with many branches and junctions as shown in Figure 3-3.

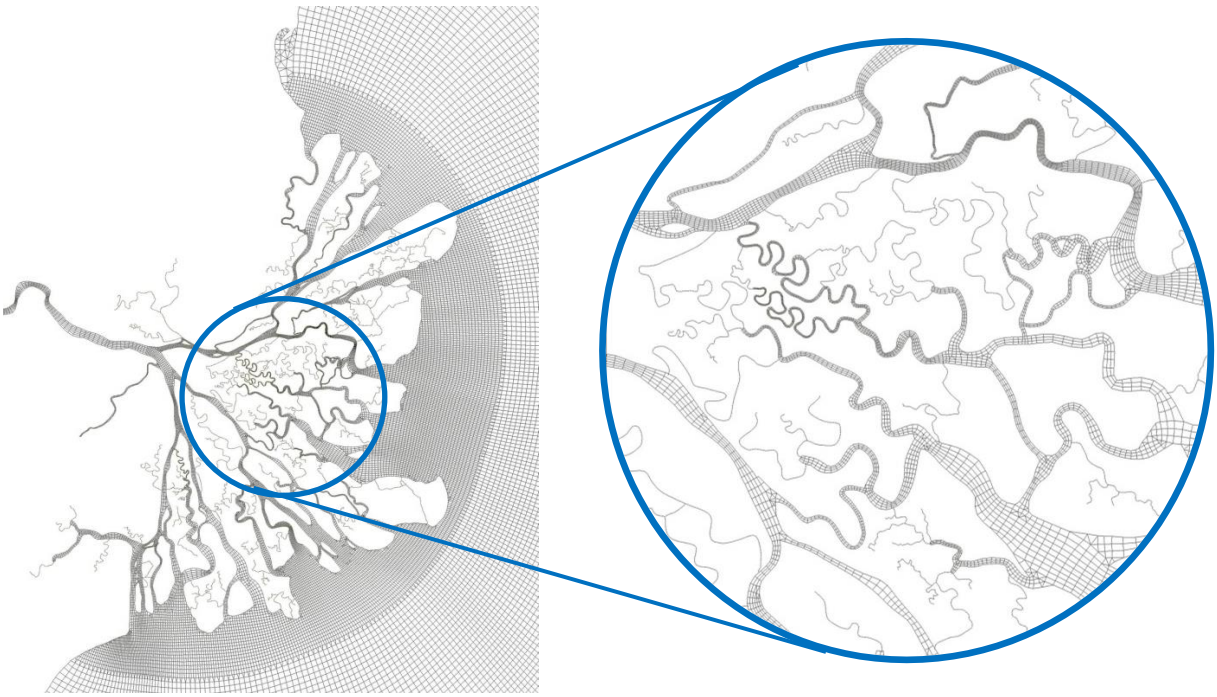


Figure 3-3: Example of an unstructured grid for the Mahakam Delta (Borneo)

There are two design requirements for creating an unstructured grid, similar as Delft3D. It needs to be smooth and orthogonal. The smoothness of a cell is defined by the relative cell surface area to adjacent cells. This ratio ideally equals unity from which an example is shown in Figure 3-1. The orthogonality is the degree of perpendicularity the flow link intersects the net link (cell face). This measure is defined by the cosine of the angle φ between both links, ideally being perpendicular (Casulli and Zanolli, 2002). To comply with these design criteria, D-Flow FM has an 'Orthogonalise/Smooth' feature which replaces the net nodes and flow nodes for finding a better network criteria.

To guaranty numerical stability the length of the flow links needs to be sufficient to comply with the CFL criteria of Eq. 3-5 and to avoid small denominators. This means that the cell circumcenter needs to lie within the grid cell. Here for grid cells need to have acute angles. Figure 3-4 gives an example of a wrong computational cell in red, but with still an acceptable flow link due to the geometry of the adjacent cell. It would be better to correct this cell by avoiding the obtuse angle and matching the surface areas to the surrounding cells (see green cell).

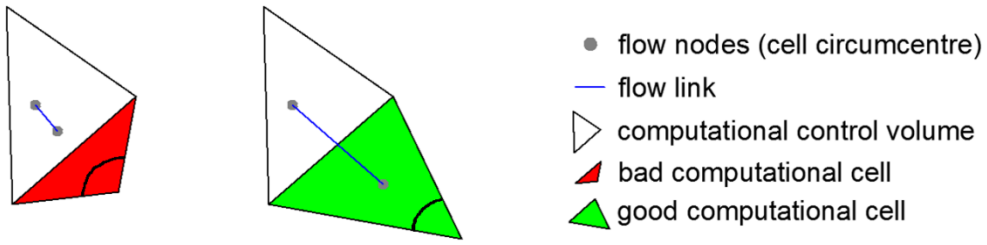


Figure 3-4: Example of a bad computational cell in red due to the obtuse angle. The green cell corrected the obtuse angle which positions the cell circumcenter inside the cell.

Although a flexible mesh allows all kinds of polygons in the domain, mostly curvilinear grid cells are used. Unstructured cells are mainly used in transition areas with different grid resolution, grid connections and meandering channels. As demonstrated by Verwey et al. (2011), a curvilinear grid more or less in line with the main flow pattern is preferred over a fully unstructured grid, due to the reduced velocity fluxes in space. Relative large velocity gradients make the non-linear advection terms of Eq. A-6 until Eq. A-8 relative dominant, introducing relative large numerical errors. This theory suggests an application of grid cells that are elongated and aligned in the flow direction, as illustrated in Figure 3-3. Models like ADCIRC (Dietrich et al., 2011), TELEMAC (Jones and Davies, 2006) and MIKE Flexible Mesh (Sørensen, 2004), are less flexible in that perspective due to their fully unstructured grid restrictions.

3.5 Open boundary conditions

To limit the computational area, but by still including relative large spatial physical processes, open boundaries are applied where necessary. Different type of 'virtual' water-to-water boundaries are available depending on:

- The type of forcing; e.g. tidal elevation and river discharge
- Relevant processes; e.g. salinity and sediment intrusion
- Numerical stability; e.g. well-posed systems with dampening of the eigen-modes

The following boundary conditions are considered:

- Water level: $\zeta = F_{\zeta}(t)$
- Discharge: $Q = F_{Q}(t)$

- Neumann: $\frac{\partial \zeta}{\partial n} = f(t)$
- Riemann invariant: $U \pm \zeta\sqrt{\frac{g}{d}} = F_R(t)$

The implementation of the harmonic water level boundaries in D-Flow FM are verified to the WAQUA model by Kernkamp et al. (2011). An application of the Continental Shelf Model (CSM) is used for a tidal simulation with astronomical boundary components. A uniform grid with spherical coordinates is applied in the WAQUA model. D-Flow FM used the unstructured grid illustrated in Figure 2-4. In according to Kernkamp et al. (2011), the computed water levels of both models match well, which verifies the implementation of the harmonic boundary condition in D-Flow FM.

Water level boundary:

The water level boundary is widely applied in modelling nearshore hydrodynamics. For relative large models, astronomical water level boundaries are used to represent the tidal forcing. This harmonic forcing can have an asymmetric shape due to relative large amplitude to depth ratios (by means of Eq. 2-3). Such a non-linear signal is modeled by means of Fourier series. This is a summation of harmonic wave components each propagating with different amplitudes, frequencies and phases. This is described by the function:

$$\zeta = \sum_{j=1}^{N} \hat{\zeta}_j \cos(\omega_j t - k_j x) = \sum_{j=1}^{N} \hat{\zeta}_j \cos(\omega_j t - \varphi_j) \qquad \text{Eq. 3-6}$$

where $\hat{\zeta}_j$ represents the amplitude of the harmonic signal in [m], ω_j and k_j the radian frequency and wave number respectively determined by Eq. 2-9 and the phase difference φ_j over a distance of $d_{1,2}$ is determined by:

$$\varphi_j = \frac{2\pi}{L_j} d_{1,2} \qquad \text{Eq. 3-7}$$

The numerical stability is enhanced by adding some numerical damping. Stelling (1983) added the time-derivative of the Riemann invariant multiplied with a reflection coefficient α, to Eq. 3-6. This gives the following water level boundary:

$$\zeta + \alpha \frac{\partial}{\partial t}\left(U \pm 2\sqrt{gH}\right) = F_\zeta(t) \qquad \text{Eq. 3-8}$$

The implementation of this formulation in Delft3D is validated by Gerritsen et al. (2007). In D-Flow FM the formulation of Eq. 3-6 has been verified by Kernkamp et al. (2011) in terms of astronomical components. However, the formulation of Eq. 3-8 has yet (2013) to be implemented in D-Flow FM.

Neumann boundary:

A Neumann boundary type is often used for relative small nearshore models but should always be applied with (at least) one water level boundary to make the system numerically well-posed (Zijlema, 2011). For example, a tidal forcing is simulated by one harmonic offshore water level boundary and two harmonic lateral Neumann boundaries. In this sense, the Neumann boundaries represent an alongshore water level gradient, enabling a discharge and a change in water level at the boundary. Roelvink and Walstra (2004) showed the relevance of implementing the Neumann formulation in Delft3D.

The Neumann boundary of the type $\frac{\partial \zeta}{\partial n} = f(t)$ is found by taking the spatial derivative of Eq. 3-6, which gives the following expression:

$$\frac{\partial \zeta}{\partial x} = \sum_{j=1}^{N} k_j \hat{\zeta}_j \sin(\omega_j t - k_j x) = \sum_{j=1}^{N} k_j \hat{\zeta}_j \cos\left(\omega_j t - \left(\varphi_j + \frac{\pi}{2}\right)\right) \qquad \text{Eq. 3-9}$$

The amplitude of the Neumann boundary is thus a function of the wave number times the wave amplitude. Due to the cosine formulation an extra phase shift of $\pi/2$ is required. The water level gradient is implicitly implemented in de numerical solver. Expression Eq. 3-9 is first multiplied with ∂x and then placed on the right hand side of the matrix solver.

For both Delft3D as D-Flow FM the post-processing of the water level gradient have great similarities. Only D-Flow FM requires a positive slope for model inflow. An application of the Neumann boundary is given in the test case: Application to a tidal model (Duck schematisation).

3.6 Application to a uniform flow (river case)

3.6.1 Introduction

Relative simple but well understood systems help us to understand the behaviour of numerical models. In this case a river flow debouching in a large open lake is considered with a constant river discharge, based on the validation case of Gerritsen et al. (2007). The backwater curve finds equilibrium between the hydrostatic pressure and the bed shear stress. This river case is well understood in terms of analytical modelling. This allows a direct understanding of system behaviour in terms of dependency of parameters and processes. This kind of comprehensive modelling is preferred over many other kinds of modelling techniques.

3.6.2 Model schematisation

A uniform river is considered with a stationary upstream discharge debouching in a large open lake with a constant water level. The water level of the lake coincides with the equilibrium height (h_e) of the river. Along the river, both the bed friction and the sloping bathymetry are considered to be constant. This gives in theory a uniform river flow with parallel stream lines. For the numerical cases the grid sizes and time steps are kept the same and only the steady state solution is considered. The following model parameters are used:

Table 3-1: Parameters for modelling the river case

Length L (m)	10 000
Constant slope i_b (-)	0.0001
Discharge q (m^2/s)	5
Chézy coefficient C_D ($m^{1/2}/s$)	65
Grid size dx=dy (m)	500
Time step dt (s)	60

3.6.3 Analytical approach

The effect of back water curves in river flows are extensively treated by Havinga et al. (2006). Schematising the model such that the water level coincides with the equilibrium depth provides a rather simple expression for the river system. The model schematisation can be described by the following analytical expression:

$$h_e = \left(\frac{q}{C_D\sqrt{i_b}}\right)^{2/3} \qquad \text{Eq. 3-10}$$

Solving Eq. 3-10 for the parameters of Table 3-1 gives an equilibrium depth of:

$h_e = 3.89677m$

3.6.4 Numerical approach, Delft3D

Delft3D FLOW uses a staggered grid solver with the depth point defined on the nodes of this network and the water level defined at the cell centres. The water level boundaries are defined just outside the computational control volume due to this staggered grid solver. For the river case, the equilibrium depth is considered by Eq. 3-10 at the end of the domain (at the cells face). Finding the right $\zeta_{boundary}$ means that the equilibrium depth needs to be correct by the bed slope times the length of half a grid cell. With a bed slope of 10^{-4} and a grid size of 500m one finds: $10^{-4} \cdot \left(\frac{500}{2}\right) = 0.025m$. Figure 3-5 clarifies this subtraction by means of the discretized bed level.

$$\zeta_{boundary} = 3.89677 - 0.025 = 3.87177m \qquad \text{Eq. 3-11}$$

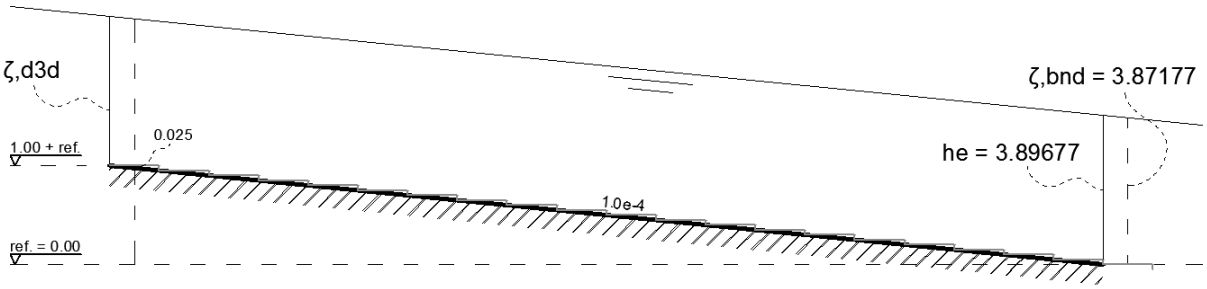


Figure 3-5: Open channel sketch including the water level definitions

Consider a domain of 20 rectangular grid cells. This returns a staggered grid of mxn equals 21x2 grid points. The grid enclosure, in terms of the grid point (array) number, equals 22x3 as clarified in Figure 3-6. Applying no boundary condition coincides with a 'closed boundary' where the normal velocities are zero. For the river case this coincides with the river banks. Both upstream and downstream open boundaries are applied with respectively a constant discharge and a fixed water level. The discharge boundary is positioned on the cell face. The water level boundary is located outside the computational control volume; in Figure 3-6 this is positioned on array m=22. The observation point 10.000m upstream is at the array m=2.

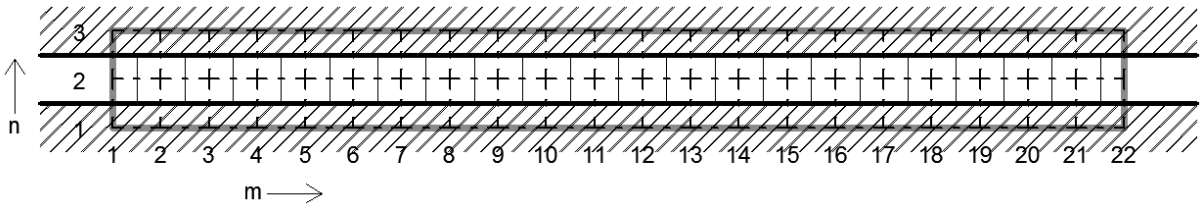


Figure 3-6: Open channel definition sketch of staggered grid with grid enclosure in grey

With a subtraction of $10^{-4} \cdot 10\,000 = 1.0m$ and again the 0.025m due to the staggered grid solver, one finds:

$$\zeta_{upstream,D3D} = 4.87120 - 1.0 + 0.025 = 3.89620\text{m} \quad \text{Eq. 3-12}$$

3.6.5 Numerical approach, D-Flow FM

The D-Flow FM model settings are exactly copied from the Delft3D settings, as prescribed by Table 3-1. The grid is converted to a network, but with the same staggered grid principle as in Delft3D: the velocity definition on the cell face, the water level defined on the cell circumcenter (cell centre here) and the depth point defined on the nodes of the network.

h_e defined on cells face:

The D-Flow FM simulation is schematized in Figure 3-5 and Figure 3-6, similar as the Delft3D simulation. The downstream water level is defined as Eq. 3-11 ($\zeta_{boundary} = 3.87177\text{m}$) in such a way that the equilibrium depth is reached on the cell face. In Figure 3-7 the $\zeta_{boundary}$ is defined in the most right flow node (white node). The observation point is defined 20 cells upstream, in the second flow node from left. The water level at the observation point needs to be subtracted by $10^{-4} \cdot 10\,000 = 1.0m$ and corrected by 0.025 due to the staggered grid solver. One finds:

$$\zeta_{upstream,FM1} = 4.85718 - 1.0 + 0.025 = 3.88218\text{m} \quad \text{Eq. 3-13}$$

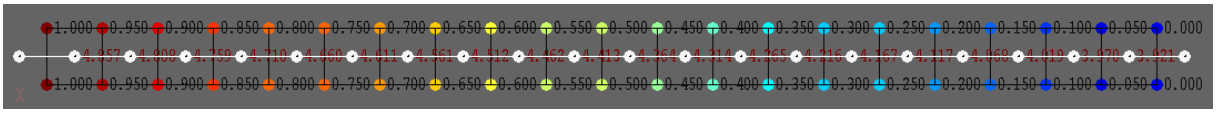


Figure 3-7: Open channel definition with flow nodes in white and bottom level dotted on network nodes

h_e defined on cells circumcenter:

The result found by Eq. 3-13 can be more accurate when investigating the numerical solver of D-Flow FM more closely. In the D-Flow FM model relative large errors are found near the upstream boundary. This can be explained by the upwind solver (backward in space) of the advection terms of Eq. A-7 and Eq. A-8. As described according the discretised Eq. A-12 and similar to the D-Flow FM solver, communicating information upstream could influence numerical outcomes, especially with Chezy flows.

The condition of the upstream discharge boundary at the cells face, determines the upstream information of the ghost cell. This is where numerical errors arise. For calibration matters, the D-Flow FM model is provided with a new feature which does one forward step in space, only at the discharge boundary. The performance of D-Flow FM with this new feature seems to be stable for this case study, although the robustness of this solution remains questionable.

Another D-Flow FM model is built, by taking this matter into account. All water levels are now directly determined on the water level points (the cell circumcenter) instead of the cell face, so no corrections need to be applied for the so-called 'staggered grid solver'. The $\zeta_{boundary}$ is determined by the condition of the second last flow node of Figure 3-7. One assumes that the condition in this

particular node equalizes the equilibrium depth. The depth information in the flow node comes from the right network node:

$$h_{i+1/2} = \zeta_{i+1/2} + d_{i+1} \qquad \text{Eq. 3-14}$$

In the second last flow node to be considered, the depth of Eq. 3-14 equals zero. Due to this matter the downstream boundary conditions equals the equilibrium depth minus the vertical difference over the length of one grid cell:

$$\zeta_{boundary} = 3.896766938 - 0.05 = 3.846766938m \qquad \text{Eq. 3-15}$$

The observation point is found in the second first flow node of Figure 3-7, that is 20 flow links upstream of $\zeta_{boundary}$. Applying the correction of the bed slope and adding the vertical difference over one grid cell to obtain the computed equilibrium depth over the full 20*500=10.000m, gives the normative computational output:

$$\zeta_{upstream,FM2} = 4.846766941 - 1.0 + 0.05 = 3.896766941\text{m} \qquad \text{Eq. 3-16}$$

As one may notice, the accuracy compared to the analytical solution is much higher compared to the outcome of Eq. 3-13. Solving Eq. 3-10 with an accuracy of 9 decimals for the parameters of Table 3-1 gives an equilibrium depth of:

$h_e = 3.896766938m$

which makes the numerical outcome of Eq. 3-16 accurate by O(e-09) compared to the analytical solution.

3.6.6 Model findings

From the model results elaborated above different results are found. Table 3-2 summarizes these model results with the difference to the analytical solution found by Eq. 3-10. The relative difference is the difference divided by the length of the domain (10.000m), which gives a dimensionless number.

Table 3-2: Model results river case

Model	$\zeta_{upstream}$[m]	Difference [m]	Relative diff [-]
Analytic	3.89677	-	-
Delft3D	3.89620	0.00057	5.7e-08
D-Flow FM1	3.88218	0.01459	1.459e-06
D-Flow FM2	3.896766941	3e-09	3e-13

3.6.7 Conclusion river case

A relative simple case study is investigated for validating both Delft3D-FLOW and D-Flow FM. The equilibrium depth of an open channel flow is considered with a sloping bathymetry. Attention is paid to the staggered grid solver, which determines the water level boundaries of both numerical models.

The steady state solution of both models show acceptable accurate results. Although both model settings are kept the same, the Delft3D model gives a smaller relative difference to the analytical solution than D-Flow FM. The ratio of the order of magnitude of this relative difference is O(e-08) to O(e-06). In D-Flow FM the largest errors were found upstream of the flow model, near the discharge boundary, due to the upwind solver.

A feature is built for D-Flow FM which does one forward step in space only at the boundary. Although the robustness of this solution still needs to be investigated, the numerical outcome is much better. The relative error for this specific case is reduced to O(e-13).

These findings are of interest for further model comparisons to more complex case studies.

3.7 Application to a tidal model (Duck schematisation)

3.7.1 Introduction

A model simulation is carried out in D-Flow FM to verify the nearshore hydrodynamics induced by a tidal forcing for the Duck model. Special attention is given to the flooding/drying principle near the waterline. The emphasis of this test case is on the implementation of the boundary conditions for nearshore models. For the sake of simplicity, a stationary moment in time is modelled. These performances are tested against the Delft3D model. Both model settings are kept similar.

3.7.2 Duck model

The Duck model is more or less an alongshore uniform nearshore model with 1200 x 4200m dimensions. Duck is located in the state of North Carolina, US. Extensive research is done along this coastal strip by the US Army Engineer WES, since 1979. Three of these projects are: DELILAH (Van Dongeren et al., 2003), DUCK94 and SandyDuck '97. Field data of DUCK94 and SandyDuck '97 are used by a research of Hsu et al. (2006) for validating and calibrating a nearshore Delft3D model. The emphasis for the research was on the effect of bottom friction and the consideration of including the surfbeat model. The results showed robust and accurate predictions of the nearshore flows. Model settings used for this test case are based on the field research of 1997.

In the simulation, wave effects and morphodynamics are excluded to achieve the main objective of the test case. The obstruction of the pier is removed from the model in order to make the model quasi alongshore uniform. The model bathymetry is based on the field data from SandyDuck '97, which shows more or less an alongshore uniform bathymetry.

US Army Engineers Field Research Facility, site of the SandyDuck '97 experiment

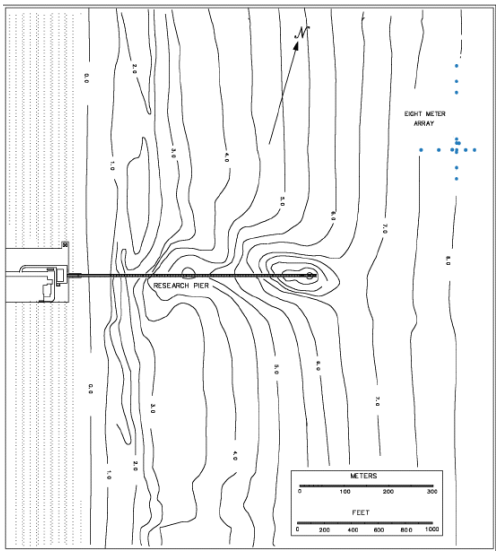


Figure 3-8: SandyDuck '97, North Carolina, US (source: http://frf.usace.army.mil)

3.7.3 Bottom level definition

The models are built with special attention to the bottom level definition. In D-Flow FM the bottom level (defined at the water level points) are (by default) set to the lowest connected link, as explained in the model theory. To match these settings, the DPSOPT in Delft3D is put from MEAN to MAX (lowest point). The DPUOPT is set from MOR to MIN which is (in case of excluding morphology) the same.

The threshold depth, above which a grid cell is considered to be wet, is set from e-01m to e-04m. These settings are of main interest near the waterline, where the flooding/drying principle plays a dominant role. For a rising tide (flooding of the cells), locally high velocities at the shoreline could appear due to sudden flooding of the cells. By lowering the threshold depth, this instant flooding is prevented. Flooding of the cells happen much smoother in this sense and prevents numerical instabilities near the waterline.

3.7.4 Derivation of harmonic boundary conditions

The Delft3D Duck model is conducted with a harmonic water level boundary at the sea side and in the two lateral harmonic Neumann boundaries. This gives a simulation of the M2-tide. In the model comparison to D-Flow FM, the harmonic boundaries are replaced by stationary boundaries, for the sake of simplicity. The maximum ebb velocities are simulated. Before building this model, a derivation is given of the harmonic boundary conditions:

Period tidal wave:

T = M2
= 44676 [sec]
= 744.6 [min]
= 12.41 [h]
T = 12h 24min 36sec

Amplitude Neumann boundary:

$$k\hat{\zeta} = \frac{2\pi}{L}\hat{\zeta}$$
$$= \frac{2\pi}{T \cdot c}\hat{\zeta}$$
$$= \frac{2\pi}{T\sqrt{gH}}\hat{\zeta}$$
$$= \frac{2\pi \cdot 0.5}{44676\sqrt{9.81 \cdot 13}}$$
$$k\hat{\zeta} = \text{ 6.23e-006 [rad]}$$

Phase difference:

$$\varphi = 360 \cdot \frac{x}{L}$$
$$= 360 \cdot \frac{x}{T \cdot c}$$
$$= 360 \cdot \frac{x}{T\sqrt{gH}}$$
$$= 360 \cdot \frac{4204}{44676\sqrt{9.81 \cdot 13}}$$
$$\varphi = 3°$$

3.7.5 Stationary boundary conditions

With the derivation of the harmonic boundary conditions the stationary case is built. By means of Eq. 3-9 the Southern boundary has a phase delay of 3° with respect to the Northern boundary. A simple hand computation is made, based on Eq. 3-9, to deduce the water level gradient for the Southern boundary during maximum ebb velocities, at $t = \pi/2 = 44\,676/4 = 11\,169\ sec$:

$$\frac{\partial \zeta}{\partial x} = \sum_{j=1}^{1} k_j \hat{\zeta}_j \cos\left(\omega_j t - \left(\varphi_j + \frac{\pi}{2}\right)\right)$$

$$= k\hat{\zeta} \cos\left(\omega t - \left(\varphi + \frac{\pi}{2}\right)\right)$$

$$= 6.23\text{e-}006 \cdot \cos\left(\frac{2\pi}{44676} \cdot 11169 - \left(3 \cdot \frac{2\pi}{360} + \frac{\pi}{2}\right)\right)$$

$$\frac{\partial \zeta}{\partial x} = 6.22146\text{e-}006\ [-]$$

Similar the water level boundary and the Northern Neumann boundary are deduced. These amplitudes are used as model input for both models. In D-Flow FM, the sign of the water level gradient determines model inflow or outflow. A positive specified Neumann amplitude leads to inflow. The ebb velocities of the Duck model are towards the north, which holds for a model outflow at the Northern boundary and thus a negative sign. The applied boundary settings for the D-Flow FM model are presented below.

Table 3-3: Boundary settings D-Flow FM for stationary Duck model

	Neumann north	Water level north	Water level south	Neumann south
Period (min)	0.0	0.0	0.0	0.0
Amplitude (ISO)	-6.23000e-006	0.0	0.02617	6.22146e -006
Phase (deg)	0.0	0.0	0.0	0.0

3.7.6 Model performance of stationary case

The river case showed a relative error to the analytical solution of O(e-08) for Delft3D and O(e-13) for D-Flow FM. This was for a uniform flow in equilibrium with the bed shear stress, simulated by a discharge boundary and a water level boundary respectively. The Duck schematisation is much similar apart from the boundary conditions. The flow velocities are (also) balanced by the bed shear stress. This ratifies the emphasis on the performance of the stationary boundary conditions in D-Flow FM for this case.

The model settings of both Delft3D and D-Flow FM are kept similar for a one-to-one model comparison. The computational time step is fixated and kept similar to Duck Delft3D ($\Delta t = 6sec$). As well as the initial conditions, friction definition ($Chezy = 65$), threshold depth ($Epshu$), grid resolution and depth file.

The relative performance of the model forcing is tested by means of the Root Mean Square Error (RMSE). This parameter shows the overall model performance relative to the Delft3D model. The model parameters to be considered are the water level and current velocity:

- RMSE water level: 0.00027 [m]
- RMSE current velocity: 0.01027 [m/s]

Both the water levels and the current velocities show a relative good match to the Delft3D model. For a further analysis, the current velocities of both models are investigated. The results are plotted in Figure 3-9.

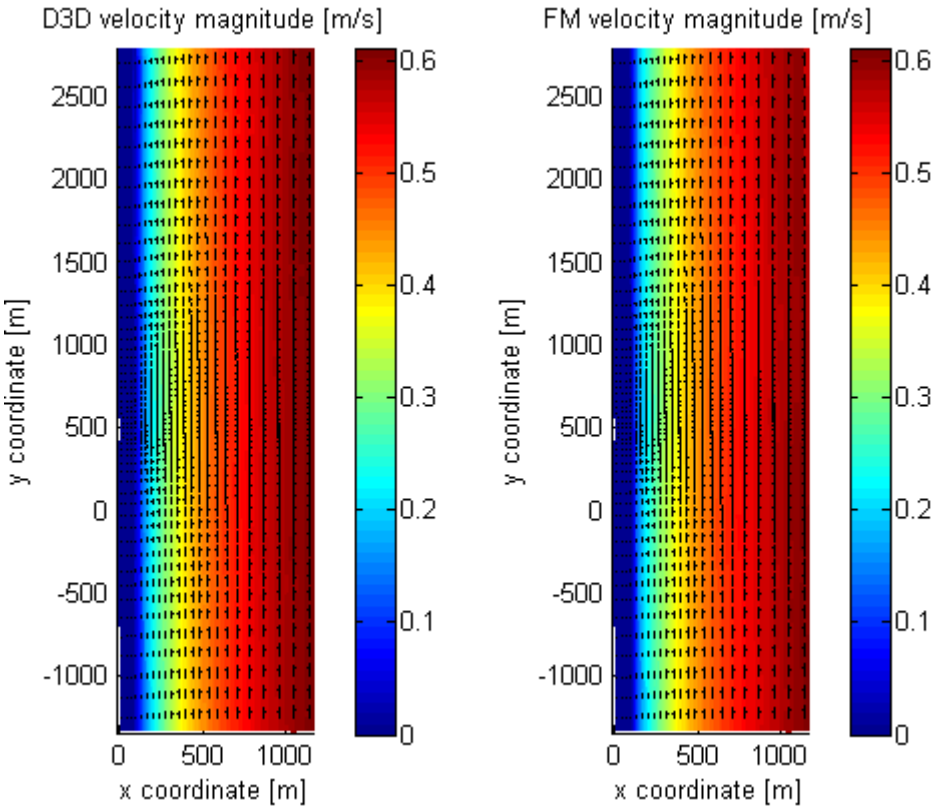


Figure 3-9: Depth-averaged velocities of the Duck model with 2x Neumann and one offshore water level boundary.

These results show qualitative similarities, which verify stationary Neumann boundaries in D-Flow FM. The model differences are analysed in more detail by subtracting both models from one another. These results are shown in Figure 3-10, by means of a plan view and a cross section at y = 2000m.

Two locations of interest are found from the difference plot; near the waterline and at the offshore boundary. Despite tuning the flooding/drying settings, numerical differences are found at the waterline. Offshore, the differences between both models are rather significant. Near the Southern boundary (at y < -1000m) this difference is hardly noticeable but reinforces moving north. Both these model deviations should be taken into account for further developments.

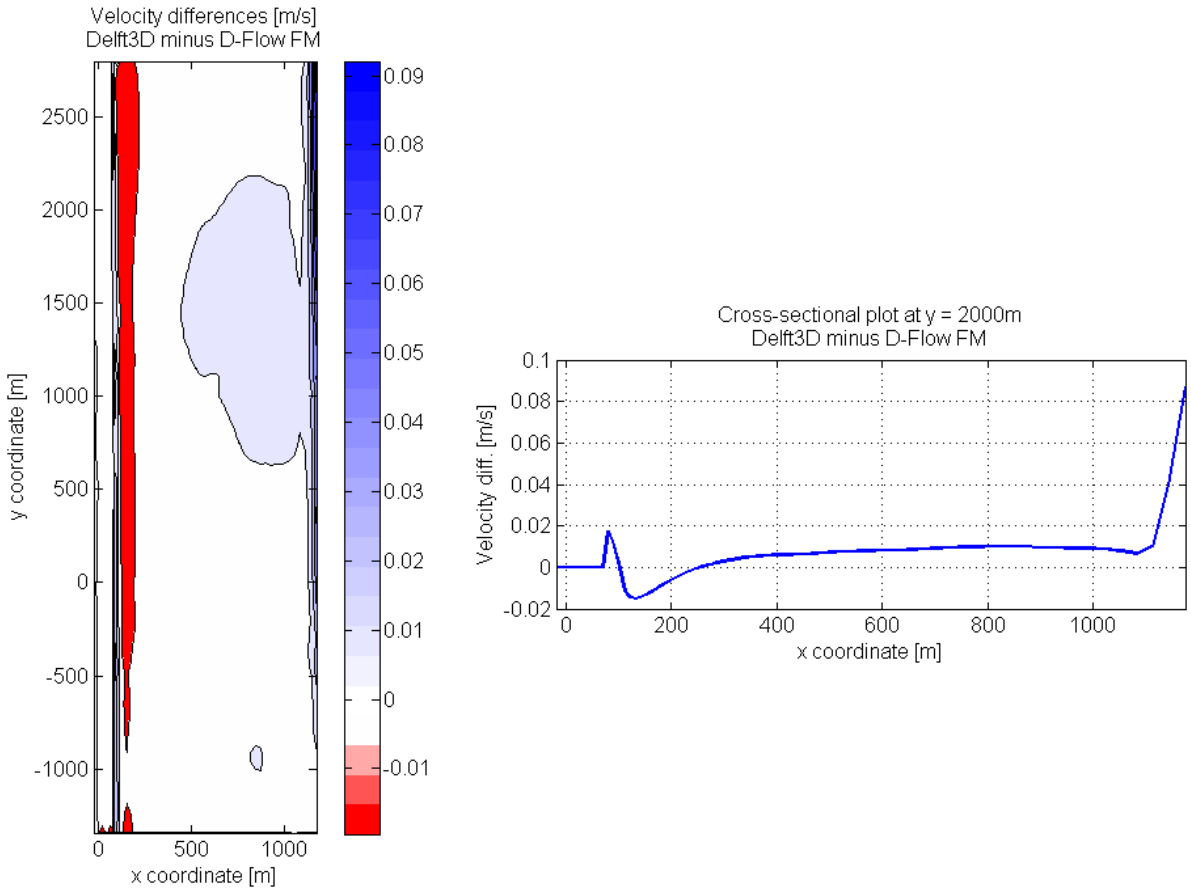


Figure 3-10: Velocity differences of the Duck model with Delft3D minus D-Flow FM

3.7.7 Computational performance of Delft3D versus D-Flow FM

In addition the performance of D-Flow FM is tested relative to Delft3D by means of the computation time. The simulations are run on a dual core processor of 2.69GHz with 2,00GB of RAM. The number of computational grid cells are similar, as well as the number of time integrations. The models are rerun several times on different moments to minimize secondary effects, but no significant differences were found between the reruns.

Table 3-4: Computational performance of the stationary Duck model

	Delft3D	D-Flow FM
Duration simulation [hours]	24h	24h
Averaged time step [seconds]	6s	5.999s
Spin-up time [hours]	4h	4h
Computation time [minutes]	15min	25min

The spin-up time for both models shows to be more or less the same which based on the velocity fields. The significant difference in computation time could mainly be induced by the visualizer of the graphical user interface of D-Flow FM. In according to Kernkamp et al. (2011), computation times of D-Flow FM are in the same order of magnitude as WAQUA, TRIWAQ and Delft3D-FLOW.

3.7.8 Conclusion Duck model

For coastal research purposes, the site of Duck is extensively investigated with validated models as a result. A validated Delft3D model is used as a setup to validate tidal modelling by D-Flow FM. For the sake of simplicity, a stationary moment in time is considered for modelling the highest ebb velocities. Primary results show a good performance of D-Flow FM, for both water levels and velocities.

The model forcing is simplified to an offshore sloping water level boundary and two lateral Neumann boundaries. The settings are derived from the validated Delft3D model and tuned for a stationary case. Attention is given to the numerical implementation of the boundary settings for both models. Other model settings are kept similar as well, with special attention to the flooding/drying settings.

Based on the model comparison, the results of the water levels and velocities show realistic and accurate results. This concludes a qualitative correct formulation of stationary Neumann boundaries in D-Flow FM. Though, predominated model differences are found at the waterline and at the offshore region. These differences should be taken into account for using Neumann boundaries for further developments.

In addition, further research is required for validating harmonic Neumann boundaries in D-Flow FM.

Chapter 4

Wave application in D-Flow Flexible Mesh

4.1 Introduction

Modelling the evolution of wind-generated waves in D-Flow FM requires a coupling to a wave module. In Delft3D the standardized wave module is the third-generation SWAN model (Ris et al., 1999; Booij et al., 1999). For the online coupling of Delft3D + SWAN, the communication file is used for data transfer between both modules. By means of this report a new couple technique is presented for D-Flow FM + SWAN. With the data transfer through the internal memory, a much faster coupling tool is established which allows interactive modelling.

The layout of this chapter starts with an overview of the applied software methodology. The dynamic coupling tool of D-Flow FM + SWAN is highlighted with the use of innovative platforms. To establish this coupling system, the encoding of D-Flow FM is adapted by the inclusion of wave effects. This is done by means of this MSc work and mainly based on the Delft3D encoding. These adaptions are tested and validated by means of two nearshore schematisations.

4.2 Dynamic coupling of D-Flow FM with SWAN

4.2.1 Program layout

Conventionally, both FLOW and WAVE module of Delft3D are run by an executable, and the wave computations are performed with a separate SWAN executable. This software methodology showed to be robust, but not so flexible in terms of debugging computations and interim adjustments to simulations. A software design which allows the user to be much more interactive during the simulation is the 'Hollywood principle' which basically says: "Don't call us, we'll call you". D-Flow FM and SWAN are run as dynamic-link libraries (or dll's), directed by one main program (an executable). Only when the framework (the main program) sends a call to the dynamic-link library a task or computation is performed.

An overview of the software methodology used for the D-Flow FM + SWAN coupling is given in Diagram 1. The layout of this structure will be discussed in the following paragraphs. On the top, the main program (framework) is positioned as an executable. Next the ESMF (Earth System Modelling Framework) platform is built, where the main model loop through time is tracked and data

regridding is performed. Next, the Basic Model Interface (or BMI) is used to enable model communication between D-Flow FM and SWAN, but also with DeltaShell for example. And finally, by the use of an API (Application Programming Interface), the dynamic-link libraries are run. Note in the diagram the colored arrows, which represent the allocated variables through the online communication.

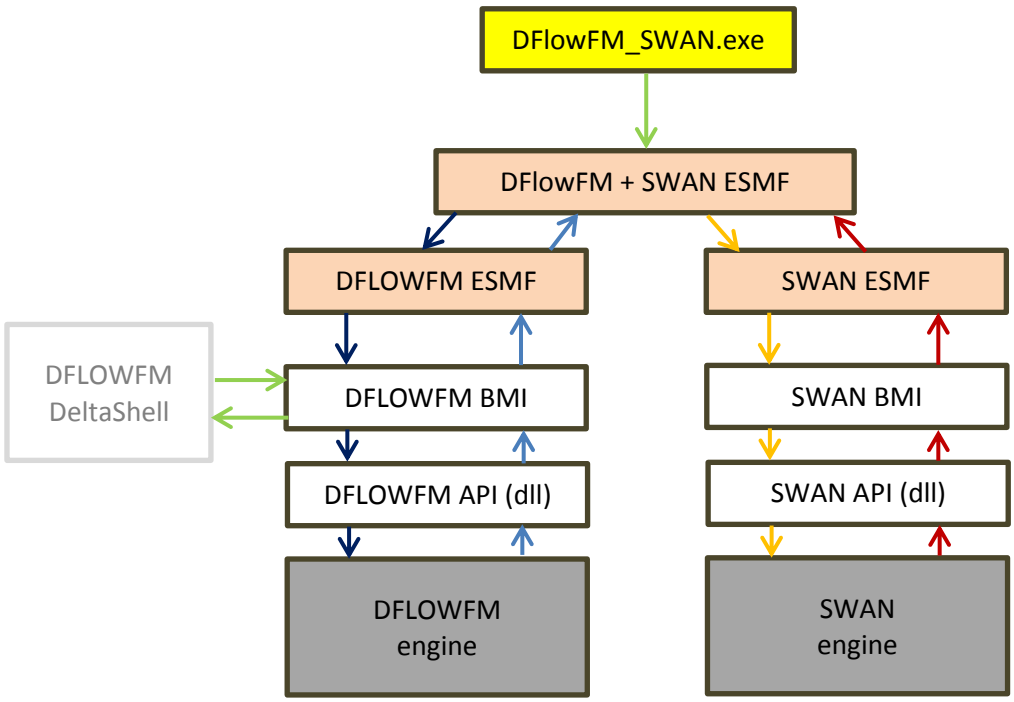


Diagram 1: D-Flow FM + SWAN coupling with 'dynamic' online communication

By means of this architecture multiple programs (or modules) can 'communicate' with one another, which could be interesting for modelling other physical processes like ecology for example. In addition, this architecture enables dynamic modelling. The engines do not run the model from start to end, but can be run per time step with intermediate model adjustments to the bathymetry or wave conditions for example. Here for the model is only initialization once, namely at the start of the time loop. Within the time loop the results are directly visualized to the user.

4.2.2 Improvements from dynamic coupling system

With this architecture, both engines can run parallel with different computational time steps. On the level of 'DFlowFM + SWAN ESMF' the variables merge and are set for the new couple interval. Here for, D-Flow FM uses wave-information which is one time step lagging, due to the variable-dependency of both modules. For relative small temporal variations of the wave forcing, these delayed responses are nil. If required, both models can run in series with no lagging information.

Another advantage of this dynamic layout is the data transfer via the internal memory. In contrast to data transfer via disk (or file), data transfer by the internal memory is much faster, but somewhat more complicated to realize. Encoded variable-positions are allocated after each computational

performance of the module. With modern technology (2013) this speeds up the process with a factor of 250 to more or less 50 GB/s. With relative short coupling intervals, this difference becomes significant.

This makes the D-Flow FM + SWAN coupling efficient from a computational perspective, but also from the users' perspective. By the work of Donchyts et al. (2013) the user can run the model, visualize the model and edit the model all at the same time. This interactive way of modelling gives a whole new nature to coastal modelling which promotes general model understanding.

These improvements of the dynamic coupler show high potentials for the field of coastal engineering and oceanography. The interactive coupling tool provides a much shorter feedback loop of the model response, promoting the model understanding. This is illustrated by Figure 4-1 which shows the interactive improvements.

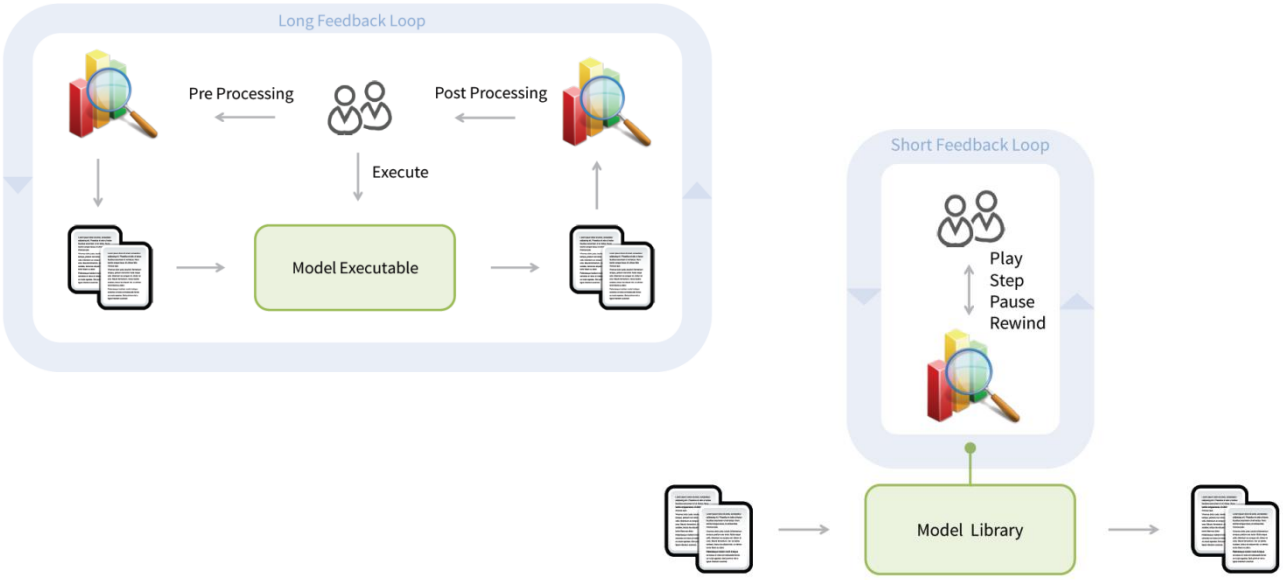


Figure 4-1: Left the conventional run by an executable. Right an 'interactive' run by means of a dynamic-link library.

4.2.3 ESMF platform

For the software development there has been chosen to use the ESMF framework by the work of Hill et al. (2004). ESMF is an open source software platform with extensive Earth-science applications like numerical weather predictions and data assimilation. The main functionality used from the ESMF architecture is the 'Coupler Component Class' with the regridding functionalities.

The choice for using the ESMF framework is based on the following aspects:

- Good support for coupling structured and unstructured models
- Written in a clear Fortran interface; (Delft3D and D-Flow FM are both written in this same programming language).
- Supports MPI's (Message Passing Interfaces), e.g. the Linux cluster of Deltares
- In- and output files of ESMF regridding application are also in netCDF format.

Main functionalities of the ESMF framework are tracking the model time-loop, managing when and how individual models are run and data interpolation. Both D-Flow FM and SWAN are managed at once on this ESMF platform. Both modules can run parallel with different time steps. When both flow- and wave computations are finished, the relevant data is interpolated by the ESMF regridding and set as model input for the new computations. In this sense the online coupling is managed by the ESMF framework. Other applications of the ESMF framework are discussed by the work of Collins et al. (2005) and Killeen et al. (2006), and are used for climate models like NOAA GFDL model and NASA GEO-5 model.

4.2.4 BMI platform

By means of the BMI application, coupling multiple (existing) libraries or frameworks has made possible. This has been implemented in D-Flow FM by the work of Donchyts et al. (2013) with interesting applications. For example, D-Flow FM communicating with hardware which traces hand movements for modelling real-time water displacements. This shows the capability of the dynamic software coupler.

The development of the BMI application is mainly the work of Peckham et al. (2012), which distinguishes three main parts in a model run. At the start of the time loop the model is initialized, next a model update is performed within the time loop and finishes with a finalize step. An example of a BMI application processed with MATLAB is given below:

```
% Load the library
[bmidll] = bmi_new(unstruc.dll);
bmi_initialize(bmidll, input.mdu);

% Online water level update
dt = 1.0;
for i = 1:100
   bmi_update(bmidll, dt);
   s1 = bmi_var_get(bmidll, 's1');
end
bmi_finalize(bmidll);
```

Example 1: BMI functionality with 'Hollywood principle'

This example describes the essence of the BMI functionality. The dynamic-link library of D-Flow FM (unstruc.dll) is first initialized by using the model input file (input.mdu). The model is updated every time step (dt) and retrieves (in this example) the allocated water levels (s1) for that time step. In the end of the time loop the model is finalized. This is a classic example of the Hollywood principle. The dynamic-link library of SWAN is called in a similar way.

Note that the model initialization is only performed once; namely in the beginning of the computation. Now that the model initialization of both D-Flow FM and SWAN is only performed

once, the computational efficiency is significantly increased. Rather than the conventional Delft3D routine, where an initialization is performed every time a module is started.

In the BMI-example the water level parameter is retrieved (bmi_var_get) after each model update. Similar, parameters are set in the model (bmi_var_set). This setting- and getting of model parameters is shown by the colored arrows in Diagram 1, and is the fundamental essence in the online coupling technique.

By means of this BMI platform a dynamic communication with SWAN is realized. With this interface, coupling to other (open source) software packages like DELWAQ or XBeach is made possible. The modelling network can be managed with a graphical user interface (GUI) like DeltaShell (Donchyts and Jagers, 2010).

4.3 Wave formulations in D-Flow FM

4.3.1 Introduction

The wave parameters computed by SWAN give rise to a change in the nearshore hydrodynamics. Under the 2DH assumption, the bed shear stresses are enhanced and the wave forces trigger alongshore currents and a water level change in the nearshore. These effects can be significant, depending on the amount of wave energy. For a planar beach, current velocities of 0.3m/s arise with waves of (only) 1.0m high. For a beach with rip channels these velocities can be much higher.

The wave formulations implemented in D-Flow FM are based on the Delft3D formulations. Main differences are the method of integration and the data interpolation. This section discusses the numerical implementation of the wave formulation in D-Flow FM.

4.3.2 Adapted momentum equation

By the theory of physics of waves was found that the wave forces and shear stresses give an important contribution to nearshore hydrodynamics. In Delft3D the wave-induced forces are applied in the momentum equations as source terms (Deltares, 2011, Delft3D-FLOW). This essence of FLOW + WAVE coupling is described in Longuet-Higgins and Stewart (1964).

A similar procedure is used for the depth-averaged model of D-Flow FM. Considering the wave-averaged momentum equation of D-Flow FM and substituting the wave-induced forces as a source term, gives the following expression:

$$\frac{\partial\langle\vec{u}\rangle}{\partial t} + adv(\langle\vec{u}\rangle) + g\nabla\langle\zeta\rangle + \frac{1}{\rho}\nabla\langle\overrightarrow{\tau_b}\rangle + 2\Omega \times \langle\vec{u}\rangle = \vec{F} \qquad \text{Eq. 4-1}$$

where $\langle\cdot\rangle$ denotes a time averaging over the wave motion, $\overrightarrow{\tau_b}$ the vectorial bottom shear stress and $\vec{F} = [F_x, F_y]^T$ representing the substituted wave forces. The bed shear stresses ($\overrightarrow{\tau_b}$) from Eq. 4-1

consists of a wave related part and a current related part. Various wave-current interaction models are built, to account for this phenomenon. Soulsby et al. (1993) developed a parameterization of these models with one standard formula, each model having its own fitting coefficients. This formulation is adopted in the D-Flow FM code, based on the Delft3D formulations.

The bed shear stress due to the current related part ($|\vec{\tau}_c|$) is a main function of the current velocity squared and some kind of friction coefficient, depending on the kind of formulation. The expression for the wave related part ($|\vec{\tau}_w|$) is proportional to the orbital velocity squared and to a wave friction factor (f_w) given by Swart (1974). The orbital velocity is computed from the linear wave theory and is given by [Deltares, 2011, Delft3D-FLOW]:

$$u_{orb} = \frac{1}{4}\sqrt{\pi}\frac{H_{rms}\omega}{\sinh(kH)} \qquad \text{Eq. 4-2}$$

where H_{rms} represents the Root-Mean-Square wave height, ω and k the radian frequency and wave number respectively, and H the water depth. The radian frequency is defined according to Eq. 2-9, by 2π over the wave period. In this definition the smoothed peak wave period (T_{ps}) is used from SWAN instead of the standard peak wave period (T_p). The T_{ps} is continuous in the frequency domain, rather than the discrete T_p. There is also an option to retrieve orbital velocities by SWAN, which is not in the scope of this research. Summarized, the following parameters from SWAN are used to determine the enhanced bed shear stresses due to waves ($|\vec{\tau}_w|$):

- Root-mean-square wave height (H_{rms})
- Peak wave period (T_{ps})
- Wave direction (θ)
- (Orbital velocity (u_{orb}))

The bed shear stresses are computed in a separate OpenMP section to distribute the overall computational workload over multiple cores of the processor. First the orbital velocities are computed by Eq. 4-2. Next the enhanced bed shear stresses are computed based on the Soulsby formulation. And finally, the friction term of Eq. 4-1 is updated. The wave forces are interpolated to the velocity points and substituted as explicit source terms in Eq. 4-1. Next Eq. 4-1 is solved for the new computational time step.

4.3.3 Flow – wave interaction

By the physics of waves was found that the flow conditions influence the wave field. The phase speed of the wave for example, shows a dependency of the water depth, as was found by the dispersion relationship. Thus a change in water level setup gives a response on the wave propagation and wave breaking. In addition, the current velocities change the wave spectrum by a frequency shift (Holthuijsen, 2007).

Delft3D takes account for these phenomena by the online coupling method to SWAN. Updated flow conditions are sent to SWAN on a certain interval. With these updated flow conditions a new SWAN computation is performed. The following parameters are updated in SWAN:

- Water level
- Bottom level
- GLM velocity in x-direction
- GLM velocity in y-direction
- Wind velocity in x-direction
- Wind velocity in y-direction

4.4 Dynamic SWAN module

4.4.1 Introduction to SWAN

SWAN is a third generation phase-averaged wave model, for modelling random short crested waves (Ris, 1997). Fields of application are coastal areas, lakes and estuaries with given wind-, bottom-, and current conditions. Interesting applications of SWAN, besides (Deltares, 2011, Delft3D-WAVE), is the work of Dietrich et al. (2011) by coupling SWAN to ADCIRC in the application of modelling hurricane waves on fully unstructured grids.

Validation and verification of SWAN is extensively carried out by Delft University of Technology. Different test cases are validated by the work of Ris et al. (1999) and Booij et al. (1999), which are in agreement with Gautier (2010). The following aspects are considered:

- Verification of the proper implementation of formulations for physical processes that are implemented in SWAN.
- Determine the numerical accuracy and robustness of the SWAN model.
- Verification of the input and output commands of SWAN.
- Validation and verification of the performance of the SWAN model in idealized-, laboratory- and field cases (with added winds and currents).

4.4.2 Operation of SWAN

Considering a wave field with the presence of a current, wave energy is no longer conserved since energy can be transferred between the current and the waves (Bosboom and Stive, 2011). In SWAN, waves are described with the wave action density spectrum:

$$N(\sigma,\theta) = \frac{E(\sigma,\theta)}{\sigma} \qquad \text{Eq. 4-3}$$

Through the wave action density spectrum, wave action is conserved. In SWAN the action balance equation is solved which for Cartesian coordinates is (Hasselmann et al., 1973):

$$\frac{\partial}{\partial t}N + \frac{\partial}{\partial x}c_x N + \frac{\partial}{\partial y}c_y N + \frac{\partial}{\partial \sigma}c_\sigma N + \frac{\partial}{\partial \theta}c_\theta N = \frac{S}{\sigma} \qquad \text{Eq. 4-4}$$

The first term on the left-hand side represents the inertia or 'local rate of change'. The second and third terms represent the rate of change by transportation or advection. The fourth term represents the frequency shifting of the waves due to ambient currents. Here σ is defined as the radian frequency in a system moving with the current (relative radian frequency) and is a function of depth and ambient currents (Holthuijsen, 2007). The fifth term represents depth-induced and current-induced refraction. The source term on the right of Eq. 4-4 consists of the contribution of winds, non-linear wave-wave interaction and dissipation.

SWAN can be applied in a stationary mode so time has been omitted from Eq. 4-4. A wave record is assumed to be stationary (Gaussian) over a period of 15–30minutes (Holthuijsen, 2007).The interval of FLOW + WAVE coupling by the communication file is, for stationary assumptions, consequently set to 30minutes. Non-stationary situations, i.e. coupling intervals smaller than 30minutes, are simulated as quasi-stationary with repeated model runs. This requires a relative large computational domain, i.e. large compared to the phase velocity of the waves through the computational domain (Deltares, 2011, Delft3D-WAVE).

The SWAN simulation is run on the computational grid for the x-, y-space, and a separate grid for the θ-, σ-space. In geographic space an implicit upwind scheme is used as numerical solver with an iterative four-sweep technique. A spatial discretization is used with the information on the vertices. In spectral space the numerical schemes are implicit mixed upwind/central order schemes. These implicit schemes in the four-dimensional propagation space ensure unconditionally stable wave propagation (Booij et al., 1999).

4.4.3 SWAN BMI

In the dynamic coupler of D-Flow with SWAN, SWAN is used as a library rather than an executable. In this sense both modules communicate by means of the BMI application. The functionality of the BMI platform is to couple multiple libraries to an interactive system (Peckham et al., 2012). The setting- and getting of data is now possible through the internal memory, rather than the exchange by files.

The model initialization and model computation of SWAN are separated by means of SWAN BMI. The SWAN model is only initialization once, namely at the start of the time loop. This is where the input files of SWAN are defined, which are used throughout the main time loop. By initializing the model only once, the computational efficiency is increased, compared to the conventional Delft3D-WAVE simulation.

The wave parameters in SWAN are stored in one large output array (SWAN team, 2013). This variable is adopted in the internal memory of the whole framework. This data is internally transferred via the ESMF regridding to the internal memory of D-Flow FM. This online communication is schematized in Diagram 1 by the colored arrows. An interesting concept with this configuration is the routine to which D-Flow FM receives wave information. A SWAN computation is not *called* (as with the executable configuration of Delft3D) but *received* when the main program says so. This is a classic application of the Hollywood principle. These commands are managed on the ESMF platform.

4.5 Data regridding

Prior to the wave computation the data of D-Flow FM needs to be interpolated (or mapped) to the computational grid of SWAN. After the SWAN run, the data is remapped for the next D-Flow FM computation. In general, interpolations induce errors with possible loss of (energy- and/or mass-) conservation as a consequence. The ESMF regridding functionality is used for these interpolations (Hill et al., 2004).

The relevant hydrodynamic conditions from D-Flow FM are all known on the water level points of the grid, apart from the wind velocities. In addition, the grid cell administration of D-Flow FM is kept at the water level points (flow nodes). This is an essential difference compared to Delft3D, due to the different polygonal cell shapes of the unstructured grid. Communicating the model parameters at the water level points to the ESMF regridding, is in that sense obvious. By the ESMF regridding the data is mapped to the vertices of the computational grid used by SWAN.

In general the SWAN domain is much larger than the flow domain due to implications from the wave boundaries in SWAN. For a successful SWAN computation the flow conditions in the non-overlapping areas must be known by data extrapolation. A robust and straightforward method is to copy and extent the condition at the boundaries of the inner domain to the non-overlapping area.

An interesting functionality of ESMF is the 'ESMF_RegridWeightGen'. This is a regridding application by means of a weight matrix, identifying the relative parameter contribution to a mapped grid cell. This weight matrix is multiplied by the values of the source grid to produce the values of the destination grid. There are three different types of interpolation methods: patch, first-order conservative and bilinear interpolation. The latter two are briefly presented in this thesis.

First-order conservative interpolation:

The first-order conservative interpolation method typically has a larger interpolation error but performs much better in preserving the data. This conservation check is done by subtracting both the integral of the data over the source grid and destination grid. The weight matrix of this interpolation method is determined by the intersecting area of the source grid cells with the destination grid cells, divided by the area of the destination grid cell. This is schematically drawn in the figure below.

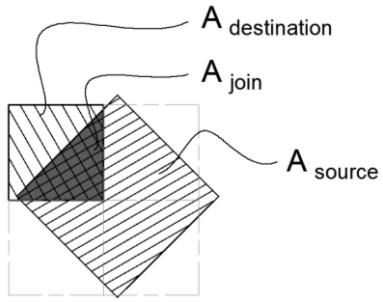


Figure 4-2: Definition sketch first-order conservative interpolation

$$W(m,n) = \frac{A_{join}}{A_{destination}(m,n)} \qquad \text{Eq. 4-5}$$

Bilinear interpolation:

The default interpolation method of the ESMF regridding application is bilinear interpolation. The weighted value of the destination grid point is determined by a linear interpolation in one direction, and then again in the other direction. Rewriting these terms for a structured grid gives the following algorithm to generate the bilinear weight matrix:

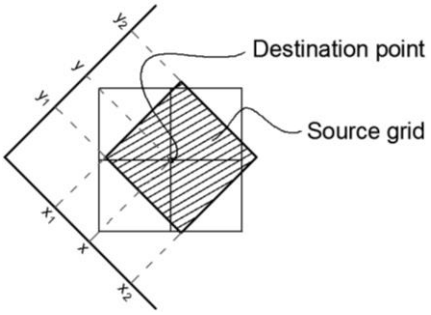


Figure 4-3: Definition sketch bilinear interpolation

$$W(x,y) = \frac{1}{(x_2 - x_1)(y_2 - y_1)} \cdot \{ f(x_1,y_1)(x_2 - x)(y_2 - y) + f(x_2,y_1)(x - x_1)(y_2 - y) + f(x_1,y_2)(x_2 - x)(y - y_1) + f(x_2,y_2)(x - x_1)(y - y_1) \}$$

Eq. 4-6

4.6 Application of wave-driven alongshore currents (planar beach)

4.6.1 Introduction

The formulation of wave-induced bed shear stresses in according to Soulsby et al. (1993) is built in the D-Flow FM code. The wave forces are included in the code by implementing an extra source term in the momentum equation Eq. 4-1. The preliminary tests of this Fortran code is executed by the use of spatial varying wave parameters from a Delft3D run with oblique incident waves. The wave parameters are read from the communication file, converted to the correct grid administration of D-Flow FM and placed as arrays in the code. In this sense the implemented wave formulation in D-Flow FM is tested for primary purposes. The results are compared to Delft3D and a 1D wave energy balance model.

4.6.2 Schematisation wave model

A planar beach profile is assumed for an alongshore uniform coast. Offshore waves approach the coast under an angle of 30° with a Hrms of 1.0m and a Tpeak of 7.0s. These wave conditions are simulated by Delft3D, D-Flow FM and a 1D wave energy balance model.

The 1D wave energy balance model solves the equation in the form of Eq. B-2. This model is MATLAB-based and is established by Deltares for primary purposes. For the sake of accuracy the model is extended with the formulation of Soulsby et al. (1993) and filled by the Fredsøe 1984 fitting coefficients.

The Delft3D model built for the specific case contains a rectangular grid with 50x50 cells on a resolution of 50x50m. A linear sloping bathymetry is used similar to the 1D model, as schematised in Figure 4-4. In order to make the flow stationary, the SWAN domain is 4 times larger than the FLOW domain but has a similar resolution. In this sense the numerical disturbances from the lateral wave boundaries do not influence the FLOW model. The directional spreading in SWAN is put to $s = 4$ and the bottom friction definition of JONSWAP is used with a default coefficient of 0.067m^2/s^3. An offshore stationary water level condition is imposed with a reflection parameter alpha set to zero. In the cross-shore two Neumann boundaries are imposed, with a water level gradient set to zero. In this sense a stationary steady state solution is found for wave-induced hydrodynamics in the nearshore.

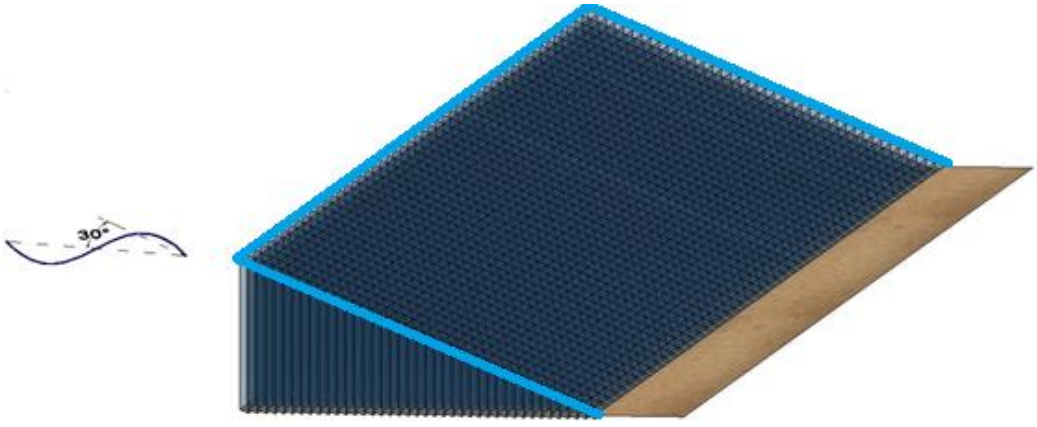


Figure 4-4: Planar beach schematised by 50x50 grid cells, 2 lateral Neumann conditions, one stationary offshore water level condition and oblique incident waves.

The wave parameters are read from the communication file and pre-processed in MATLAB in according to the grid administration of D-Flow FM. For this routine the feature 'renumber (internal) flow nodes' is switched off, which normally serves for a faster matrix solver. The wave parameters on the boundaries are defined by a copy of the nearest neighbour, similar as in Delft3D.

The same grid, bathymetry and boundary conditions are used from the Delf3D simulation. The alongshore water level gradients on the lateral boundaries are put to zero due to the alongshore uniform flow of the steady state solution. Stationary Neumann boundaries in D-Flow FM show realistic results as was shown by the test case: Application to a tidal model (Duck schematisation). The post-processing of the model results are in GLM-velocities (General Lagrangian Mean), which are Eulerian velocities compensated by Stokes drift. The GLM-velocity is a measure for the velocities observed at the surface, also called the total velocity, and is related to the Eulerian velocity by:

$$\vec{u}^L = \vec{u}^E + \vec{u}^S \qquad \text{Eq. 4-7}$$

where $\vec{u}^L$ is the GLM-velocity vector, $\vec{u}^E$ is the Eulerian velocity vector and $\vec{u}^S$ is the Stokes drift vector (Bosboom and Stive, 2011).

4.6.3 Model theory

A theoretical model is assumed with a stationary flow in time. For this case the wave transformation induces a spatial varying wave field over the cross-shore direction. The cross-shore forcing finds equilibrium with the pressure gradient, which triggers a water level change in the nearshore. In the alongshore direction this pressure gradient cannot be developed due to the alongshore uniformity of the bathymetry. These wave forces trigger an alongshore current which makes equilibrium with the bed shear stress. In the steady state solution Eq. 4-1 reduces to:

$$F_x = g\frac{\partial \zeta}{\partial x} \qquad \text{Eq. 4-8}$$

$$F_y = \frac{\partial \tau_{by}}{\rho H} \qquad \text{Eq. 4-9}$$

The bed shear stress ($\overrightarrow{\tau_b}$) is in general an important model parameter. The enhanced bed shear stress due to waves modelled by D-Flow FM is validated for the stationary alongshore uniform current.

4.6.4 Model performance

Despite the optimisation of the 1D wave energy balance model and an extensive sensitivity analysis of the Delft3D simulation, a shift in wave energy dissipation between both models remains. In SWAN the directional spreading, horizontal eddy viscosity, horizontal eddy diffusivity, breaker index, bottom friction formulation and bottom friction coefficient are investigated, but with no better results compared to the default settings (see Appendix C). For the sack of verification, the 1D model serves well. The alongshore velocities of both models give a qualitative match. These Delft3D-WAVE results are used as model input for the D-Flow FM simulation. The following results are found:

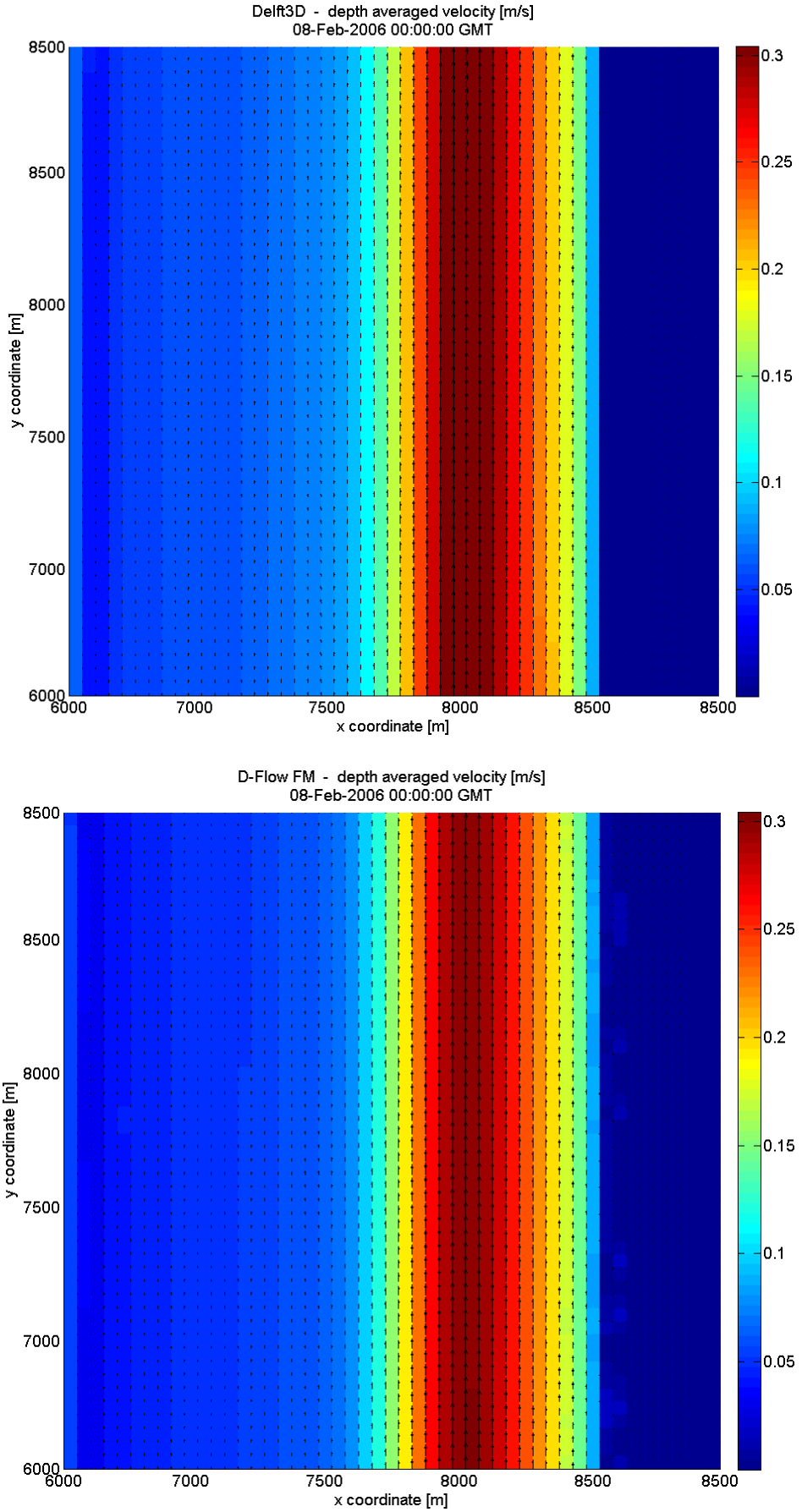


Figure 4-5: Alongshore velocities due to oblique incident waves on a planar beach. Upper panel are the Delft3D results and lower panel the D-Flow FM results.

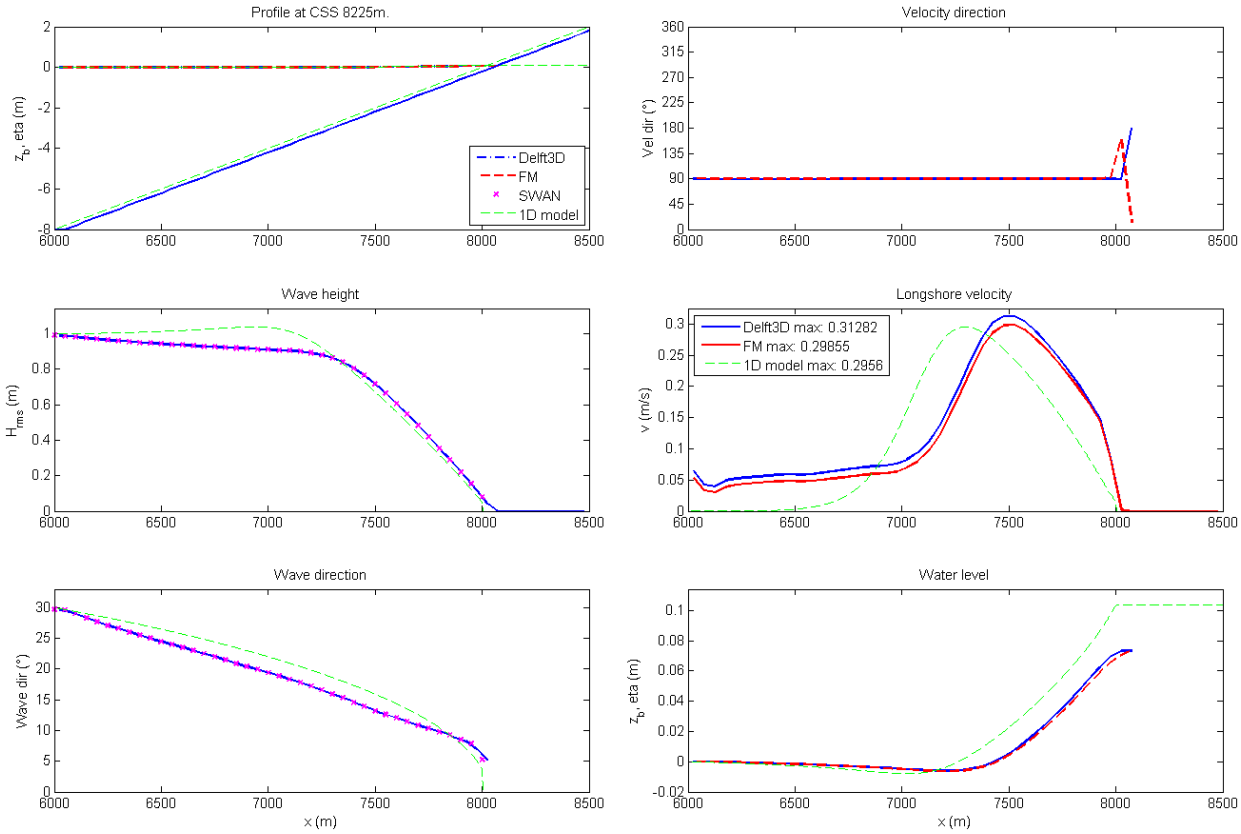


Figure 4-6: Cross-section of oblique incident waves approaching a planar beach. Delft3D model in blue and D-Flow FM model in red.

The velocity patterns of Figure 4-5 show a general good agreement, which gives a primary confirm of a correct implementation of the 2DH wave formulation in the D-Flow FM code. The cross-sectional plot of Figure 4-6 shows a good match of the alongshore velocity profile, considering the primary purposes of this test case. The alongshore velocities of Delft3D and the 1D model are of the same order of magnitude.

The wave propagation modelled by the 1D model and Delft3D show striking results. The difference in wave shoaling induces an offshore shift of wave energy dissipation. An extensive analysis is carried out in Appendix C, but with no better results. To verify the Delft3D model, a stand-alone SWAN computation is performed with the exact same input variables. The wave height and wave direction of this SWAN model is also plotted in Figure 4-6, and coincides with the Delft3D model.

The wave forcing in cross-shore direction is verified by the water level plot of Figure 4-6, in according to Eq. 4-8. The wave-induced water level setup in the nearshore of D-Flow FM show qualitative similarities to the Delft3D simulation. The alongshore velocity profile of D-Flow FM shows a general good match to the Delft3D velocity profile. These results are most accurate in shallow waters. At the offshore regions differences between both models arise. At this region the effect of wave forces decrease and bed shear stresses become dominant. From [Deltares, 2011, Delft3D-FLOW] is given that in Delft3D the bed shear stresses are computed with the Eulerian velocities. In D-Flow FM the GLM velocities are used, which is a plausible cause for the model differences. This Stokes drift correction by means of Eq. 4-7 is yet (2013) to be implemented in D-Flow FM.

4.6.5 Verification Soulsby formulation

The Soulsby formulation consists of fitting coefficients for different wave models, modelling the wave-current interaction. The commonly used expression is the Fredsøe (FR84) wave model and is here considered as the default case. To verify the Soulsby implementation in D-Flow FM, different wave formulations are compared to Delft3D simulations. These results are elaborated in Table 4-1. The Root Mean Square (RMS) velocities of both models are compared, followed by the Root Mean Square Error (RMSE) between both models.

Table 4-1: Velocity values between Delft3D and D-Flow FM with different formulations of Soulsby et al. (1993):

Model	RMS(U) D3D [m/s]	RMS(U) FM [m/s]	RMSE(U) [m/s]
Fredsøe (FR84)	0.14646	0.13843	0.13783
Myrhaug and Slaattelid (MS90)	0.14384	0.13499	0.12701
Huynh-Thanh and Temperville (HT91)	0.12855	0.12062	0.13174
Grant and Madsen (GM79)	0.10474	0.09695	0.09746
Davies et al. (DS88)	0.14878	0.14009	0.14029
Bijker (BK67)	0.09303	0.08539	0.09943
Christoffersen and Jonsson (CJ85)	0.12495	0.11683	0.11476
O' Connor and Yoo (OY88)	0.18509	0.16443	0.18595

In the post-processing of the Delft3D communication-files, the wave parameters are identical due to the non-extension of the hydrodynamic FLOW results to WAVE. The wave parameters used as model input for the D-Flow FM simulations, have thus not been changed. The results of Table 4-1 show a correlation between the RMS values and the model deviation. A linear up scaling of the model deviation can be recognized with respect to the velocity magnitude.

4.6.6 Conclusion planar beach case

A simplified wave model in D-Flow FM is verified by a Delft3D wave simulation and a 1D wave energy balance model. The model is schematised to an alongshore uniform coast with oblique incident waves. This gives a model response with an alongshore uniform velocity profile and a water level setup in the cross-shore.

The water level setup and velocity profile of D-Flow FM shows a qualitative good match to the Delft3D simulation and the 1D model. This suggests a correct implementation of the wave formulations in the D-Flow FM.

From a closer perspective is found that the effect of Stokes drift is not neglectable at offshore regions. The enhanced bed shear stress computed by D-Flow FM is yet (2013) to be extended with the Stokes drift correction. Despite this effect, the overall model performance relative to Delft3D is considered good.

4.7 Nearshore hydrodynamics for a barred beach (Egmond case)

4.7.1 Introduction

The wave formulations in D-Flow FM are tested for the Egmond case. This barred beach has been a study-ground for years now (Elias, 1999; Luijendijk et al., 2010; Winter, 2012). The coastline of Egmond attracts many bathing people in summer time but is also prone to strong rip currents. This combination makes this coastline potentially dangerous for swimming, which is a strong motivation for coastal research.

Models are built to give insight in the hydrodynamic patterns in the nearshore and to predict the possible occurrence of rip currents. Different model packages are used, but a widely applied tool is Delft3D. These settings are calibrated and validated with different field data applications (i.e. wave buoys, drifter data, Argus images) by i.a. Elias (1999) and Winter (2012).

A replica of the hindcast model from Winter (2012) is built in Delft3D, for the field survey period of 22-8-2011 until 26-8-2011. The model includes nearshore hydrodynamics induced by wind, waves and tides for a non-uniform barred coastline. The model results are compared to field measurements by means of drifter data. This validated Delft3D model forms the basis of this test case. Next this Delft3D model is simplified by considering wave-induced hydrodynamics only. A similar D-Flow FM model is built and is tested to the Delft3D model.

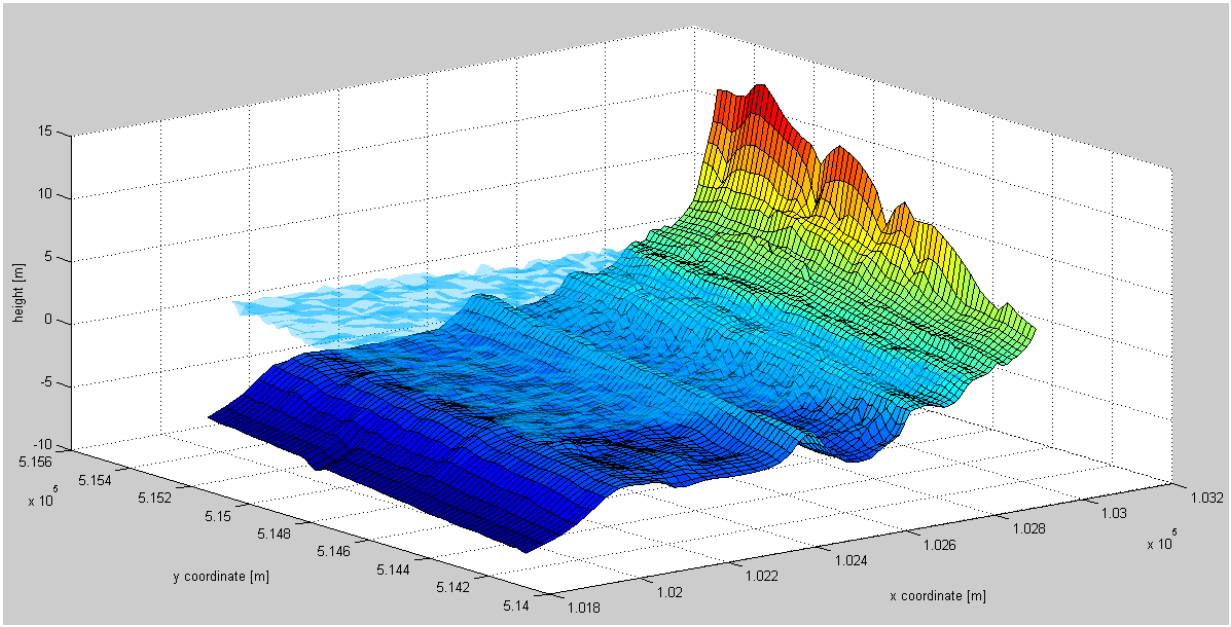


Figure 4-7: Bathymetry plot of Egmond model on a rectangular mesh

4.7.2 Modelling approach

A stationary moment in time is considered for the analysis to focus on wave-induced hydrodynamics only. Winter (2012) found by literature that the strongest rip velocities have been measured in the field at low tide. The limited water depth in the rip channel is an important characteristic of the rip strength. At high tide the waves passes over the bar without breaking and as a consequence the rip current is weaker or neglectable. This has been confirmed by the field measurements.

Clearly developed rip channels have been reported and captured on Thursday 25-8-2011 around 18.00GMT. This was during falling tide, 1hour prior low tide. The offshore wave conditions obtained are a significant wave height of 0.59m from the south-west, a peak wave period of 5.1sec, an angle of incidence of 48° and neglectable winds.

This stationary moment in time is used to hindcast the field measurements and to validate a D-Flow FM wave model of the Egmond coastline. A time independent model is used where tidal influences are neglected. The lateral boundary conditions are two zero-Neumann boundaries and an offshore water level boundary of -0.50m. The wave condition implemented in Delft3D is a stationary 2D spectrum file (*.sp2) for the wave conditions of 18.00GMT. The spectrum file is obtained by a nesting routine from a Dutch coastline model, nested in a continental shelf model, nested in a global model. These models are forced by astronomical tidal forcing and wind fields, based on metrological files.

The wave forcing in the D-Flow FM model is built with the same routine as the test case: Application of wave-driven alongshore currents (planar beach). The wave parameters are retrieved from the communication-file of a previous Delft3D simulation, and put as parameter-arrays in the D-Flow FM code. In the online coupling of Delft3D-FLOW with SWAN, the hydrodynamics results from Delft3D-FLOW are not used as model input for SWAN. In this sense it represents the D-Flow FM simulation with the stationary wave parameters as model input.

The same model settings as the Delft3D simulation are used in D-Flow FM, which gives a one-to-one model comparison. The simulations are run on the same rectangular grids. Though, the powerful feature of D-Flow FM is to apply high resolutions in areas of interest. Unstructured grids show high potentials for models like the Egmond case, especially for capturing rip currents on high resolution. The rectangular grid (which is used) has a grid resolution of 5x10m in the rip channels and 20x10m offshore. Here for 131x138 grid cells are used, 17 938 cells in total. This is a computational extensive job, given that the high resolution at offshore regions is not necessary.

4.7.3 Model validation

A Delft3D model is built based on the XBeach model of Winter (2012). The model parameters are tuned and compared to the drifter measurements. Two clear rip channels are developed in the model, approximately 400m apart. The drifter measurements focus on the structure of the Northern rip channel, and the offshore velocities of the Southern rip channel. The objective of this hindcast is simulating nearshore hydrodynamics from a qualitative perspective. In the following figure the results are shown.

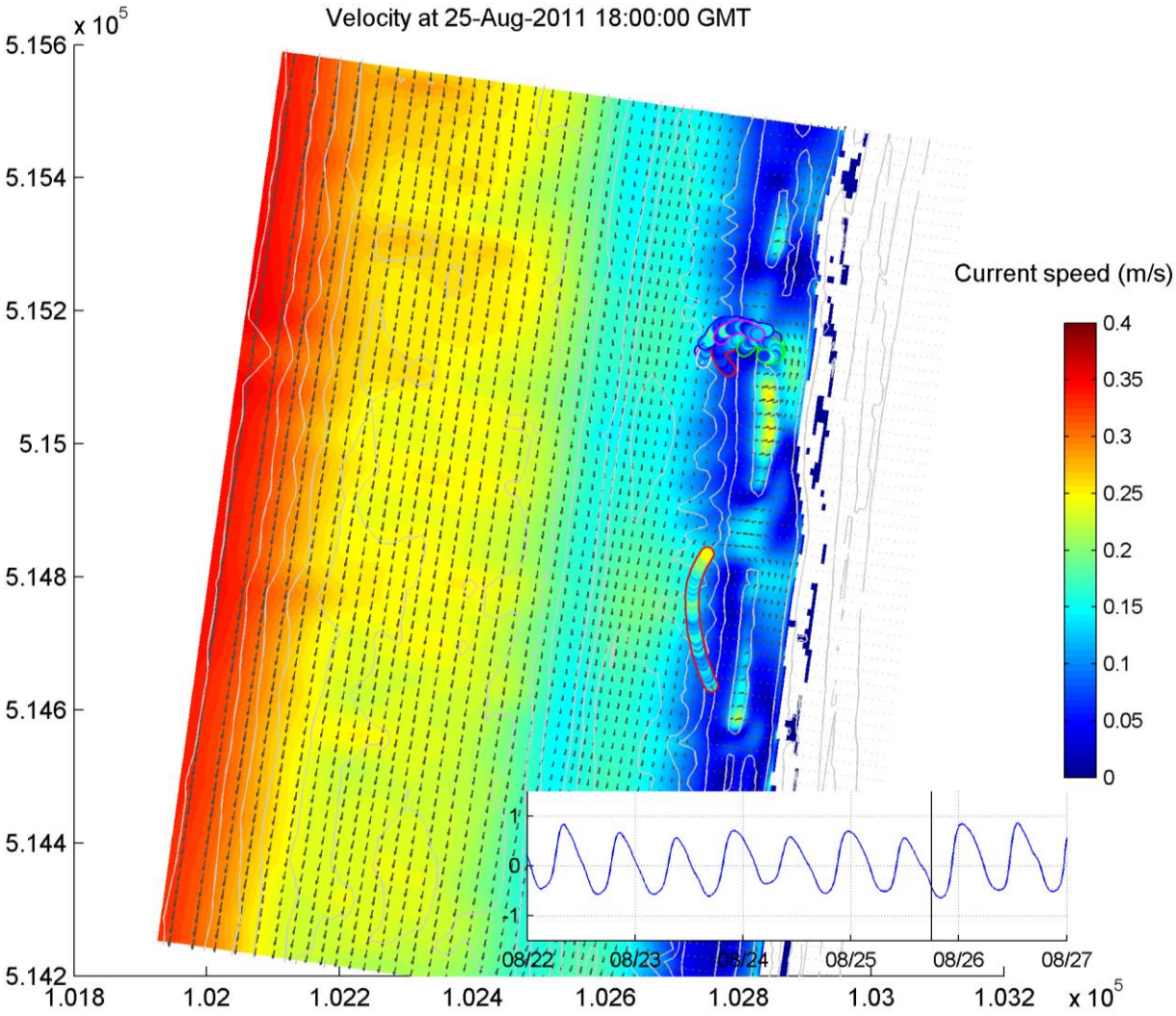


Figure 4-8: Delft3D Egmond model compared to drifter measurements. Inclusion of wave-, tide- and wind effects.

The nearshore velocity patterns of the Delf3D model shows a general good match to the drifter data. The velocity field of the rip channels match the trajectories of the drifters rather well. On the sandbanks the increase in wave action is clearly modelled by an increase in the velocity field, feeding the rip channel. Local velocity differences are likely caused by effects on a much smaller scale than a 5x10m resolution or 3D effects. One can conclude a realistic model representation of the 2DH Delft3D Egmond model.

4.7.4 Stationary model

The Delft3D Egmond model is schematized into a D-Flow FM model and forced by wave action only. The wave parameters are retrieved from the communication-file of a previous Delft3D simulation, and put as parameter-arrays in the D-Flow FM code. The grid resolution has been de-refined by a factor 2, due to the restrictions of the implemented arrays' length in the D-Flow FM code. The Delft3D-FLOW model has also been de-refined to a resolution of 66x70 grid cells (4548 cells in total)

for a one-to-one model comparison. The SWAN computation is performed on the same computational grid as the reference case. Other flow-model settings are kept similar, with special attention to the following settings:

- boundary conditions
- initial conditions
- friction definition
- horizontal eddy viscosity
- depth file
- depth definition
- threshold depth
- Soulsby formulation

4.7.5 Nearshore model performance

With the tuned model settings and the horizontal eddy viscosity set to zero, the overall results show a relative good match regarding the nearshore velocities. For the sake of maintaining numerical stability, the computational time step in D-Flow FM is determined by the CFL-criteria of Eq. 3-5 ($CFL = 0.7$). The computational time step in Delft3D is kept similar to the reference case, which gives a stable solution ($\Delta t = 6sec$). Results of the nearshore GLM-velocities of both simulations are shown in Figure 4-9 with two points of interest indicated with white arrows; the rip channel and the nearshore velocity extreme.

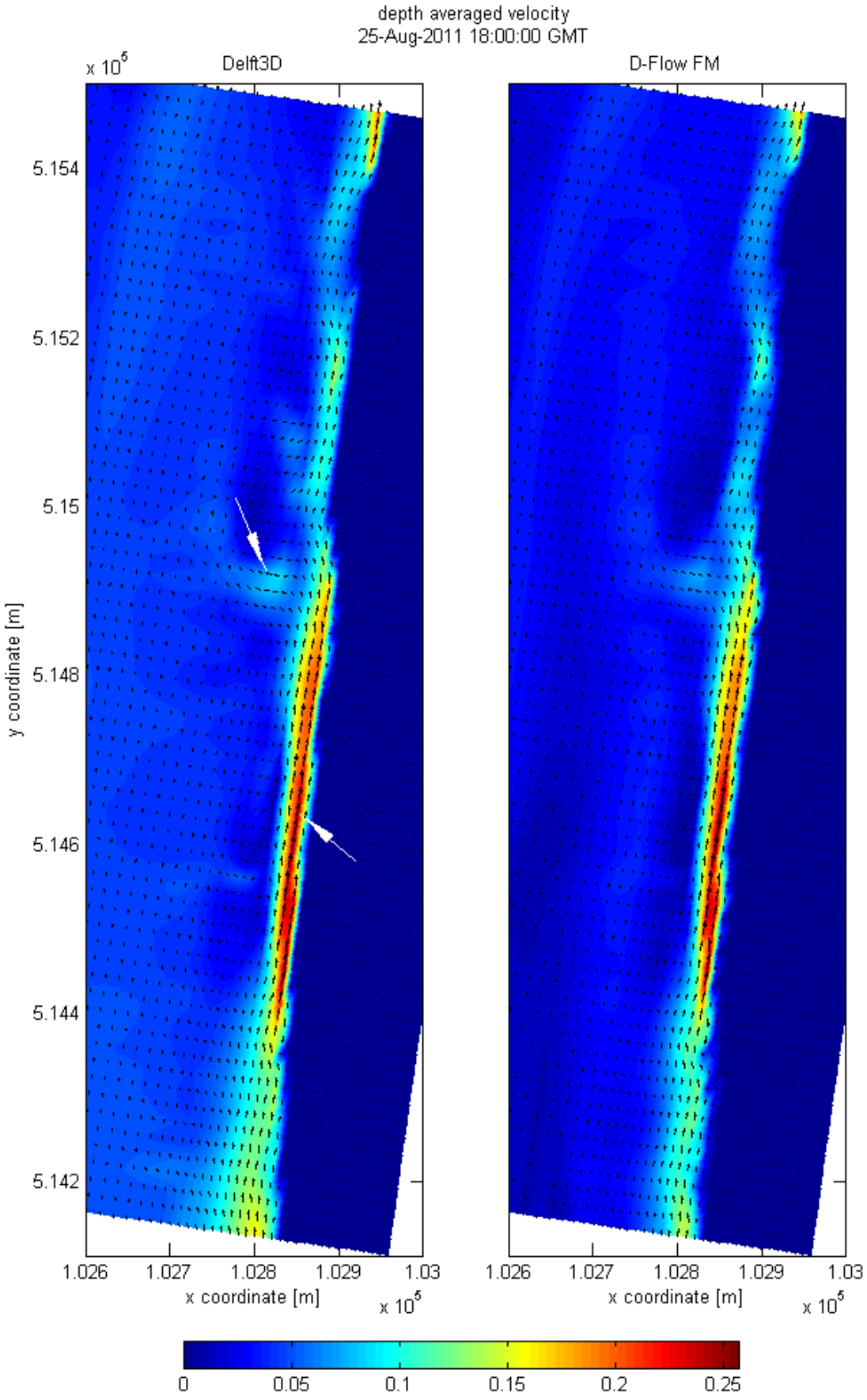


Figure 4-9: Wave simulation of the Egmond model. Nearshore GLM-velocities of the Delft3D model (left) compared to the D-Flow FM model (right). Points of interest are the rip channel and the nearshore velocity extreme.

Figure 4-9 highlights the nearshore hydrodynamics of both models, solely induced by a stationary wave field. A clear rip channel is shown which is driven by a high alongshore velocity in the nearshore. The location of this rip channel coincides with the Southern rip channel of Figure 4-8. A time-series plot is made of the absolute velocities at this location (102 818; 514 920) and at the nearshore extreme (102 850; 514 630).

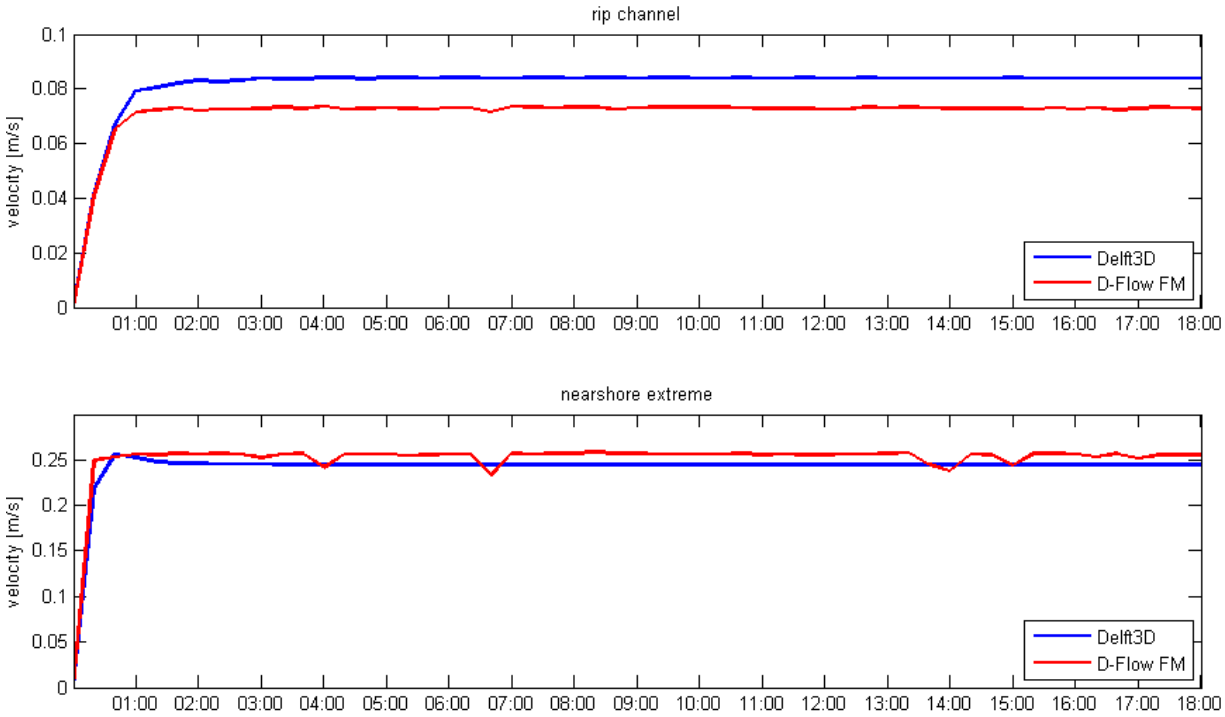


Figure 4-10: Time-series plot of the depth-averaged velocity; inside the rip channel and at the nearshore extreme.

Despite the fact that this computation is stationary in time, this time-series plot shows some interesting results. The spin-up between Delft3D and D-Flow FM is more or less the same. After the spin-up time, D-Flow FM still shows distortions in the velocity signal. The lower panel of Figure 4-10 clearly shows these distortions, while the Delft3D model shows a smooth stationary solution. This clarifies the constraint for using the CFL-criteria for the D-Flow FM simulation to make the model numerically stable. These instabilities are most likely induced by the explicit solver of the advection term.

From the lower panel of Figure 4-10 is shown that the nearshore velocities are slightly overestimated by D-Flow FM, relative to Delft3D. These nearshore velocities 'feed' the rip channel by means of advected flow. This is clearly visual in Figure 4-9. Despite the overestimate in the nearshore extreme, the velocities inside the rip channel are underestimated by 10-15%, as shown in the upper panel of Figure 4-10. These model differences are likely induced by the different advection solver of D-Flow FM.

4.7.6 Overall model performance

For the overall model assessment, the velocity fields of both models are subtracted from one another and plotted in Figure 4-11. The locations of the time-series plots are indicated with dark arrows in the nearshore.

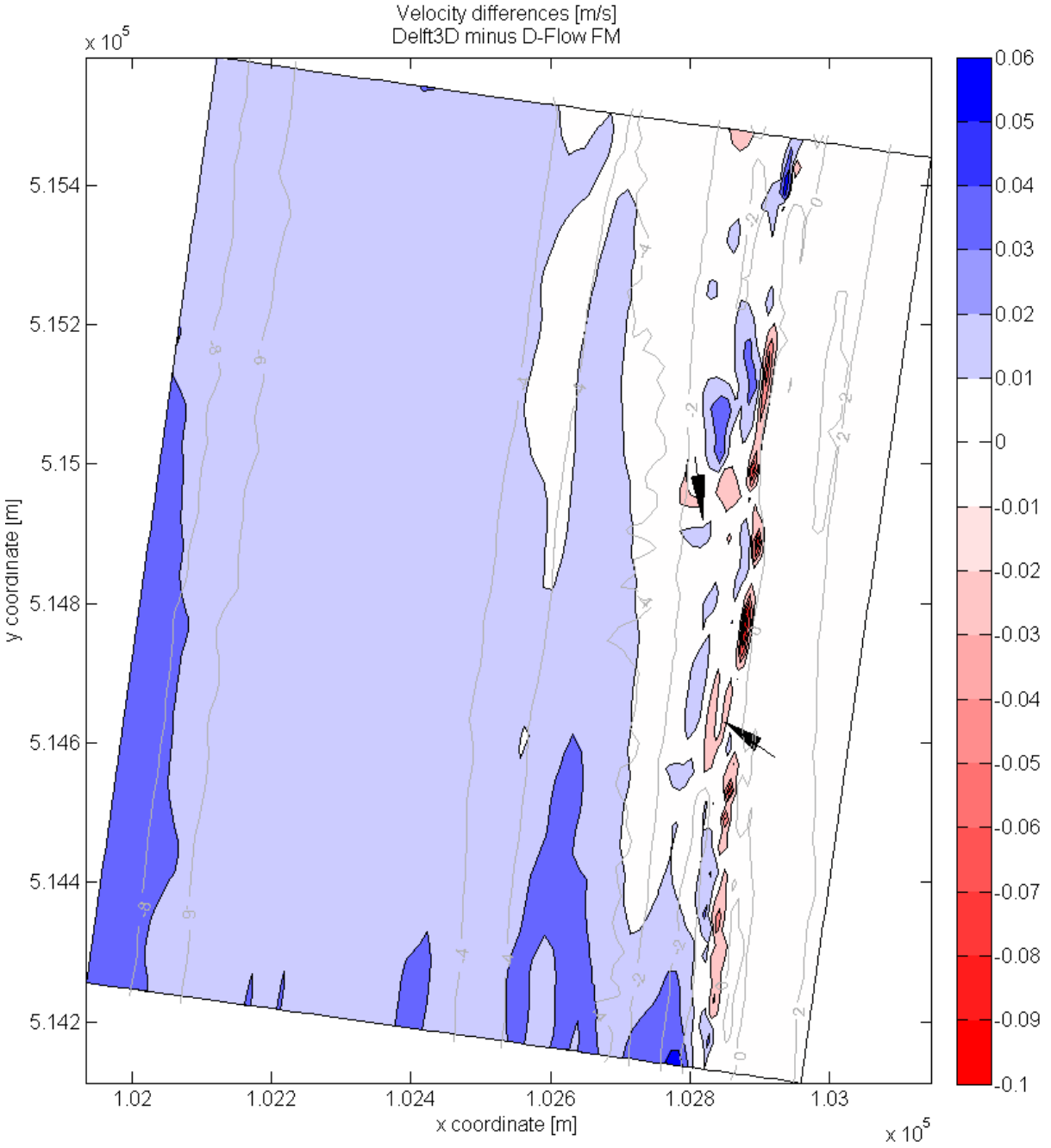


Figure 4-11: Velocity difference of the Egmond model; Delft3D minus D-Flow FM.

These results validate the results of the test case: Application of wave-driven alongshore currents (planar beach). At the offshore region effects of wave-induced forces are small. The enhanced bed shear stresses are dominant at these regions. These differences are likely caused due to the exclusion of Stokes drift in D-Flow FM. In Delft3D the enhanced bed shear stresses are computed by means of the Eulerian velocities, which are corrected for Stokes drift by Eq. 4-7 (Deltares, 2011,Delft3D-FLOW).

Other striking results from Figure 4-11 are the local velocity differences in the breaker zone and at the shoreline. Model differences at the shoreline were also found in the test case: Application to a tidal model (Duck schematisation), where this is explained by the flooding/drying principle. These numerical differences are in the same order of magnitude as Figure 4-11. In addition, the velocity differences in the breaker zone are likely caused by the advection solver of D-Flow FM which is different compared to Delft3D. This effect is clearly noticeable inside the rip channel, where advection is dominant.

The overall model performance of D-Flow FM compared to Delft3D is assessed by means of the Root Mean Square Error (RMSE). The model parameters to be considered are the water level and the absolute current velocity. The Root Mean Square (RMS) of the model parameters is also shown which is aggregated over space:

Table 4-2: Results stationary Egmond model aggregated over space:

Parameter	RMS D3D	RMS FM	RMSE
Water level [m]	0.49951	0.49981	0.00041
Current velocity [m/s]	0.04830	0.03546	0.02055

The water level differences are some order of magnitude smaller than the velocity differences. The underestimated velocities of D-Flow FM are predominated at intermediate water depths, as shown in Figure 4-11. The model differences inside the surf zone are much smaller.

4.7.7 Conclusion Egmond case

The Delft3D Egmond model is validated by means of drifter measurements. From this validated Delft3D model, a simplified stationary model is built to focus on wave modelling only. The effect of tide and wind are excluded and the wave field is kept stationary in time. This simplified model is replicated in D-Flow FM and validated to the Delft3D model.

From the model comparison is concluded that the velocities of both models (in magnitude and direction) show good similarities inside the surf zone. The spatial varying velocities in the nearshore are in agreement and the rip channel modelled by D-Flow FM is captured well with velocity fields that match the Delft3D simulation accurate by 85-90%.

The results presented are in agreement with the test case: Application of wave-driven alongshore currents (planar beach). The effect of excluding Stokes drift in D-Flow FM is not neglectable at offshore regions where predominated model differences are found. In the nearshore, the effect of the different advection solver is clearly captured. D-Flow FM gives critical velocities at the nearshore due to the explicit scheme. These nearshore velocities (of the advected flow) are slightly underestimated by D-Flow FM, which is noticeable inside the rip channels.

From this qualitative assessment is concluded that the implemented wave formulations in D-Flow FM give accurate results for primary purposes.

Chapter 5

Conclusions and recommendations

5.1 Introduction

The fundaments of an innovative coupling tool are built to run a D-Flow FM + SWAN computation efficiently. In the development of this coupling tool, the code of D-Flow FM is adapted by the inclusion of wave effects. This is done by means of this MSc work and mainly based on the Delft3D encoding. The adapted D-Flow FM model is tested and validated by means of two nearshore models.

Improvements are made for the D-Flow FM + SWAN coupling, compared to the conventional coupling of Delft3D + SWAN. Parallel computing, combined with communication via internal memory and a single initialization of the modules, makes the D-Flow FM + SWAN coupling significantly faster. In addition, the coupling tool is made interactive for the user with a much shorter feedback loop of the model response. The user can run the model, visualize the model and edit the model all at the same time. These improvements allow great opportunities in the fields of coastal engineering and oceanography.

Four test cases are considered. The first two test cases validate solely current related hydrodynamics for stationary Chézy flow and tidal forcing, respectively. Next the wave formulations in D-Flow FM are validated for a planar beach case and the Egmond case, respectively. Oblique incident waves are considered for an alongshore uniform coast and for a non-uniform barred beach. The performance of D-Flow FM is analysed by these test cases and based on this analysis, recommendations are presented for improving nearshore modelling with D-Flow FM.

5.2 Conclusion test cases

The first test case concerns the equilibrium depth of an open channel flow with sloping bathymetry. The analytical solution is compared to the numerical solutions of both Delft3D and D-Flow FM. The order of magnitude of this relative difference is O(e-08) for the Delft3D model. In D-Flow FM a feature is built (to increase numerical accuracy) which does one forward step in space only at the boundary. This decreased the relative difference from O(e-06) to O(e-13).

The second test case simulates the tidal forcing for an alongshore uniform beach. The M2-tide is modelled by means of a stationary water level boundary and two stationary Neumann boundaries. Based on the model comparison, the resulting water levels and velocities show realistic and accurate results, leading to the conclusion that the formulation of stationary Neumann boundaries in D-Flow

FM is qualitatively correct. However, further research is required for validating harmonic Neumann boundaries in D-Flow FM. Here for, only stationary boundary conditions are used for the wave simulations.

The wave formulations implemented in D-Flow FM are validated by considering two wave simulations: an alongshore uniform model (planar beach case) and a non-uniform barred beach (Egmond case). Both models are forced by a stationary wave field, which triggers a stationary velocity field and water level difference in the nearshore. The model performances are tested against validated Delft3D models.

Wave parameters are retrieved from a Delft3D + SWAN model and used as model input for the D-Flow FM simulation. The wave parameters are read from the communication file and are mapped to the D-Flow FM model by studying the grid administrations. Next, the parameter arrays are copied into the D-Flow FM code to mimic the SWAN coupling. The following wave parameters are used as model input:

- Root-mean-square wave height (H_{rms})
- Peak wave period (T_p)
- Wave direction (θ)
- Wave forces in x-direction (F_x)
- Wave forces in y-direction (F_y)

The model settings and grid resolution of D-Flow FM and Delft3D are kept similar. In this sense a one-to-one model comparison is made. The model response is tested by analyzing the following parameters:

- Water levels (ζ)
- Absolute velocities ($\left|\vec{U}\right|$)
- Flow directions (u,v)

Based on the results of both wave cases, it can be concluded that wave-driven hydrodynamics modelled by D-Flow FM are accurate within 85-90% with respect to nearshore Delft3D models. The water levels show a good match, as confirmed by the planar beach case. The wave-induced velocity patterns, both in magnitude and direction, are in agreement with Delft3D, as demonstrated by the Egmond case. The wave-driven hydrodynamics in the nearshore are captured well, including the rip channel. Based on these conclusions is found that the implemented wave formulations in D-Flow FM are successfully validated.

Based on the results, it is concluded that predominant model differences are found at intermediate water depths. From the planar beach case it is found that the effect of Stokes drift is not neglectable and may give a significant contribution to these model deviations. In Delft3D, the enhanced bed shear stresses are computed with Eulerian velocities, rather than GLM-velocities. This essential difference is confirmed by the Egmond model. In addition, the difference between both advection solvers is evident in nearshore regions for the Egmond case, where the absolute velocities inside the rip channel were slightly underestimated by D-Flow FM.

5.3 Recommendations

Wave modelling by D-Flow FM:

- Include depth-averaged Stokes drift. The enhanced bed shear stresses are computed with GLM-velocities in D-Flow FM, rather than Eulerian velocities as in Delft3D.
- Validate the model response for time-varying wave fields. In the test cases, only stationary wave fields are considered where inertial effects of flow are neglected.
- Verify SWAN BMI in the D-Flow FM + SWAN coupling. The model initialization and model computation of SWAN are separated by means of the BMI application. In addition, SWAN is called by means of a dynamic library rather than an executable, as in Delft3D. Variables are set to- and retrieved from the internal memory at each SWAN computation. These modifications to SWAN need to be verified.
- Build a feature to run multiple nested SWAN models. D-Flow FM is able to model large coastal areas with a single mesh, but with the inclusion of waves, a nesting routine for the SWAN domains is required.
- Couple D-Flow FM to UNSWAN. This is the unstructured module of SWAN (Zijlema, 2010), which shows high potential for coastal modelling in combination with D-Flow FM.
- Verify the ESMF regridding functionality with the different interpolation methods.
- 3D implementation of the wave formulations:
    - Vertical mass flux due to Stokes drift.
    - Streaming near the bottom, as a consequence of a wave-induced currents in the wave boundary layer.
    - Turbulent effects.
    - Adjustment of bottom shear stress formulation.

Nearshore modelling by D-Flow FM:

- Correct the implementation of the harmonic Neumann boundaries in D-Flow FM. Extensive research has been done to deduce the mismatching results of the harmonic Neumann boundaries. From the debugging analysis, it is concluded that the numerical differences occur in the phase definition of D-Flow FM. Obviously, this requires further analysis.
- Include the Roller model for modelling infra-gravity waves.

General modelling by D-Flow FM:

- Export the bottom friction parameter for combined current-wave effects to the output-file. Bottom friction is an important model parameter for analyzing the model response, especially for comparisons with Delft3D. The output of this parameter should be set to the water level points for a one-to-one model comparison. Because the momentum equation is solved on the velocity points, a post-processing of the friction parameter is required.
- Allow the variables of water density ($rhomean$) and threshold depth ($Epshu$) to be defined as input variables.

- Verify implemented viscosity parameter in D-Flow FM. Model runs performed in debug mode with tuned viscosity parameters caused program errors; thus, the viscosity parameter is set to zero for the wave simulations.

Wave modelling by Delft3D:

- A derivation of the orbital velocity computed from the linear wave theory, expressed by Eq. 4-2, is missing. The orbital velocity computed by Delft3D, as formulated in [Deltares, 2011, Delft3D-FLOW], is expressed differently than vanRijn (2007a) and Soulsby (1997). The Delft3D expression differs by a factor $0.5 \cdot \sqrt{\pi/2}$.

References

Booij N., Ris, R.C., Holthuijsen, L.H., (1999). *A third-generation wave model for coastal regions. Part I: Model description and validation.* Journal of Geophysical Research: Oceans (1978-2012), Volume 104, No. C4, pp. 7649-7666.

Borsboom, M.J.A., Doorn, N., Groeneweg, J., van Gent, M.R.A., (2000). *A Boussinesq-type wave model that conserves both mass and momentum.* Proc. 27th, International Conference on Coastal Engineering, Sydney, Australia.

Bosboom, J. and Stive, M.J.F., (2011). *Coastal Dynamics I*. Delft, VSSD.

Broomans, P., (2003). *Numerical accuracy in solutions of the shallow-water equations.* MSc Thesis, Delft University of Technology, EWI, Numerical Analysis.

Casulli, V. and Walters, R.A., (2000*). An unstructured grid, three-dimensional model based on the shallow water equations*. Int. J. Numer. Meth. Fluids, 32:331–348.

Casulli, V. and Zanolli, P., (2002). *Semi-implicit numerical modeling of nonhydrostatic free-surface flows for environmental problems.* Mathematical and Computer Modelling 36, 1131-1149.

Collins, N., Theurich, G. DeLuca, C., Trayaonv, A. Li, P., Yang, W., Hill, C. (2005). *Design and implementation of Earth System Modeling Framework components.* International Journal of High Performance Computing Applications.

Deltares (2011). *Delft3D-FLOW Simulation of multi-dimensional hydrodynamic flows and transport phenomena, including sediments*. Delft, Deltares.

Deltares (2011). *Delft3D-QUICKPLOT Visualisation and animation program for analysis of simulation results*. Delft, Deltares.

Deltares (2011). *Delft3D-WAVE Simulation of short-crested waves with SWAN*. Delft, Deltares.

Dietrich, J.C., Zijlema, M., Westerink, J.J., Holthuijsen, L.H., Dawson, C., Luettich Jr., R.A., Jensen, R.E., Smith, J.M., Stelling, G.S., Stone, G.W., (2011). *Modeling hurricane waves and storm surge using integrally-coupled, scalable computations.* Coastal Engineering, 58, 45-65.

Dingemans, M.W., Radder, A.C., de Vriend, H.J., (1987). *Computation of the driving forces of wave-induced currents.* Coastal Engineering 11, pp. 539-563.

Donchyts, G.V., Baart, F., van Dam, A., Jagers, H.R.A., (2013). *The joy of interactive modelling.* European Geoscience Union, Vienna.

Donchyts, G.V. and Jagers, H.R.A., (2010). *DeltaShell – an open modelling environment.* International Congress on Environmental Modelling and Software (iEMSs), Ottawa, Intario, Canada.

Dykes, J.D., Hsu, Y.L., Kaihatu, J.M., (2003). *Application of Delft3D in the nearshore zone.* Proc. 5th AMS Coastal Conference, Seattle, Washington, 57-61.

Elias, E.P.L., (1999). *The Egmond mode – Calibration, validation and evaluation of Delft3D-MOR with field measurements.* Delft University of Technology, Institutional Repository, Z2394.20/Z2396.20.

Evans, M.W., Harlow, F.H., (1957). *The Particle-In-Cell method for hydrodynamic calculations.* Los Alamos Scientific Laboratory report, LA-2139, University of California.

Gautier, C., (2010). SWAN calibration and validation for HBC2011. Delft, Deltares.

Gerritsen, H., de Goede, E.D., Platzek, F.W., Genseberger, M., van Kester, J.A. Th.M, Uittenbogaard, R.E., (2007). *Validation document Delft3D-FLOW – a software system for 3D flow simulations.* Delft Hydraulics [s.I.], 266 pp.

Goede, E. de, (2011). *Validatie van Villemonte overlaatformulering in WAQUA met praktijkmetingen.* Rijkswaterstaat – Deltares, Institutional Repository, Delft.

Hasselmann, K., Bernett, T.P., Bouws, E., Carlson, H., Cartwright, D.E., Enke, K., Ewing, J., Gienapp, H., Hasselmann, D.E., Kruseman, P., Meerburg, A., Müller, P., Olbers, D.J., Richter, K., Sell, W., Walden, H., (1973). *Measurements of wind wave growth and swell decay during the Joint North Sea Wave Project (JONSWAP).* Deutsches Hydrographisches Institut Hamburg, Ergänzungsheft zur Deutschen Hydrographischen Zeitschrift Reihe A (8), Nr. 12.

Havinga, H., Visser, P.J., de Vriend, H.J., Wang, Z.B., (2006). *River Engineering.* Reader, Delft University of Technology, Institutional Repository.

Hill, C., DeLuca, C., Balaji, V., Suarez, M., da Silva, A., (2004). *Architecture of the Earth System Modeling Framework.* Computing in Science and Engineering, Vol. 6, Nr. 1, pp. 18-28.

Holthuijsen, L.H. (2007). *Waves in Oceanic and Coastal Waters.* Cambridge University Press.

Hsu, Y.L., Dykes, J.D., Allard, R.A., Kaihatu, J.M., (2006). *Evaluation of Delft3D performance in nearshore flows.* NRL Memorandum Report, Naval Research Laboratory, Stennis Space Center, MS 39529-5004.

Jones, J.E. and Davies, A.M., (2006). *Application of a finite element model (TELEMAC) to computing the wind induced response of the Irish Sea.* Continental Shelf Research, 26, 1519-1541.

Kernkamp, H.W.J., van Dam, A., Stelling, G.S., de Goede, E.D., (2011). *Efficient scheme for the shallow water equations on unstructured grids with applications to the Continental Shelf*. Ocean Dynamics, 61:1175-1188.

Killeen, T., DeLuca, C., Gombosi, T., Toth, G., Stout, Q., Goodrich, C., Sussman, A., Hesse, M., (2006). *Integrated Frameworks for Earth and Space Weather Simulation.* American Meteorological Society Meeting, Atlanta, GA.

Kirkpatrick, M.P., Armfield, S.W., Kent, J.H., (2003). *A representation of curved boundaries for the solution of the Navier-Stokes equations on a staggered three-dimensional Cartesian grid.* Journal of Computational Physics 184, 1-36.

Kramer, S.C., Stelling, G.S., (2008). *A conservative unstructured scheme for rapidly varying flows*. J Numer Meth Fluids 58:183–1212.

Longuet-Higgins, M. S. and R.W. Stewart, (1964). *Radiation stresses in water waves; a physical discussion, with applications.* Deep-Sea Research, 11, 4, 529–562.

Luijendijk, A.P., Henrotte, J., Walstra, D.J.R., van Ormondt, M., (2010). *Quasi-3D modelling of surf zone dynamics.* Delft University of Technology, Institutional Repository.

MacWilliams, M.L., Gross, E.S., DeGeorge, J.F., Rachiele, R.R., (2007). *Three-dimensional hydrodynamic modeling of the San Francisco Estuary on an unstructured grid.* IAHR, 32nd Congress, Venice, Italy.

Peckham, S.D., Hutton, E.W.H., Norris, B., (2012). *A component-based approach to integrated modeling in the geosciences: The design of CSDMS.* Computers & Geosciences: Modeling for Environmental Change, 53:3 – 12.

Reniers, A.J.H.M., (1999). *Longshore current dynamics*. Delft University of Technology, Institutional Repository.

Rijn, L.C. van, (2007a). *Unified view of sediment transport by currents and waves. I: initiation of motion, bed roughness and bed-load transport.* Journal of Hydraulic Engineering, vol. 133, no. 6.

Ris, R. C., (1997). *Spectral Modelling of Wind Waves in Coastal Areas*. Delft University of Technology, Institutional Repository.

Ris, R.C., Holthuijsen, L.H., Booij, N., (1999). *A third generation wave model for coastal regions. Part II: Verification.* Journal of Geophysical Research: Oceans (1978-2012), Volume 104, No. C4, pp. 7667-7681.

Roelvink, J.A., Petit, H.A.H., Kostense, J.K., (1992). *Verification of a one-dimensional surfbeat model against laboratory data.* Coastal Engineering. Part II: Long Period Waves, Storm Surges and Wave Groups, pp. 960-973.

Roelvink, J.A., Stelling, G.S., Hoonhout, B.M., Risandi, J., Jacobs, W., (2012). *Development and field validation of a 2DH curvilinear storm impact model.* ICCE 2012: Proceedings of the 33rd International Conference on Coastal Engineering, Santander, Spain, 1-6 July 2012.

Roelvink, J.A. and Stive, M.J.F., (1989). *Bar-generating cross-shore flow mechanisms on a beach*. Journal Geophysical. Research., Vol. 94, pp. 4785-4800.

Roelvink, J.A. and Walstra, D.J.R., (2004) *Keeping it simple by using complex models.* In Proceedings of the 6th International Conference on Hydro-Science and Engineering. Advances in Hydro-Science and Engineering, vol. VI, page p. 12. Brisbane, Australia. 217-219.

Sørensen, O., (2004). *Floodplain modelling using unstructured finite volume technique.* DHI Technical Note, January.

Soulsby, R.L., (1997). *Dynamics of marine sands: A manual for practical applications*. Thomas Telford Publications, London.

Soulsby, R.L., Hamm, L., Klopman, G., Myrhaug, D., Simons, R.R., Thomas, G. P., (1993). *Wave-current interaction within and outside the bottom boundary layer.* Coastal Engineering 21: 41-69. 243, 245, 246, 247.

Stelling, G.S., (1983). *On the construction of computational methods for shallow water flow problems*. Delft University of Technology, Institutional Repository.

SWAN team, (2013). *SWAN Cycle III version 40.91AB.* User manual, Delft University of Technology, Faculty of Civil Engineering and Geosciences Environmental Fluid Mechanics Section.

Swart, D.H., (1974). *Offshore sediment transport and equilibrium beach profiles.* Delft University of Technology, Institutional Repository. Delft Hydraulics Publ. 131. 248, 366.

Van Dongeren, A., Reniers, A., Battjes, J., and Svendsen, I., (2003). *Numerical modeling of infragravity wave response during DELILAH*. J. Geophys. Res., 108(C9), 3288, doi:10.1029/2002JC001332.

Verwey, A., Kernkamp, H.W.J., Stelling, G.S., Tse, M.L., Leung, W.C., (2011). *Potential and application of hydrodynamic modelling on unstructured grids.* Asian and Pacific Coasts 2011: pp. 1254-1261.

Winter, G., (2012). *Rip current characteristics at the Dutch coast: Egmond aan Zee.* MSc Thesis, Delft University of Technology, Institutional Repository, 1204386-000-HYE-0004.

Wu, J., (1982). *Wind-stress coefficients over sea surface from breeze to hurricane*, J. Geophys. Res., 87, C12, 9704–9706.

Zijlema, M., Stelling, G.S., Smit, P., (2011). *SWASH: An operational public domain code for simulating wave fields and rapidly varied flows in coastal waters.* Coastal Engineering, 58: 992-1012.

Zijlema, M., (2010). *Computation of wind–wave spectra in coastal waters with SWAN on unstructured grids.* Coastal Engineering 57, 267-277.

Zijlema, M., (2011). *Computational modelling of flow and transport.* Reader, Delft University of Technology.

Appendix A

Applied numerics of nearshore modelling with Delft3D

A.1 Introduction

Deltares has developed Delft3D which is a 3D computational model for rivers, estuaries and coastal regions. It can simulate hydrodynamic flows, sediment transport, waves, water quality, morphological developments and ecology. This program has several modules with amongst others: open channel flows (FLOW), wave-driven hydrodynamics (WAVE) and water quality (WAQ). These modules cover an integrated 3D modelling of specific flow patterns, morphological developments and ecological changes.

A.2 Integration of FLOW – WAVE module

Modelling nearshore hydrodynamics requires an integration of the FLOW and WAVE module. This allows an integrated modelling of tides, wave-driven hydrodynamics, enhanced turbulence and bed shear stresses. For the interaction of the FLOW – WAVE module the online coupling of Delft3D is considered. This means that during the simulation data is transferred between the different modules by a communication file (com-****.dat).

A.3 Equations FLOW

Delft3D-FLOW solves the Navier-Stokes equations for an incompressible fluid, under the shallow water assumption (hydrostatic pressure) and the Boussinesq assumptions (ρ = constant). This is valid for relative long waves which are weakly non-linear. The equations are given in the form:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} = -\frac{1}{\rho_0}\frac{\partial p}{\partial x} + \upsilon\Delta u - f_x \qquad \text{Eq. A-1}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z} = -\frac{1}{\rho_0}\frac{\partial p}{\partial y} + \upsilon\Delta v - f_y \qquad \text{Eq. A-2}$$

$$\frac{\partial w}{\partial t} + u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z} = -\frac{1}{\rho_0}\frac{\partial p}{\partial z} + \upsilon\Delta w - f_z - \frac{\rho}{\rho_0}g \qquad \text{Eq. A-3}$$

where *u*, *v* and *w* denote the velocity components in *x*-, *y*- and *z*-direction respectively, ρ the density, ρ_0 the initial density, p the pressure, υ the kinematic viscosity and f_x, f_y and f_z represent the components of the Coriolis force. The equations can be recognized by the momentum balance equations on the left hand side and the forcing on the right hand side. These can be derived based on conservation of momentum under the assumption of an incompressible fluid:

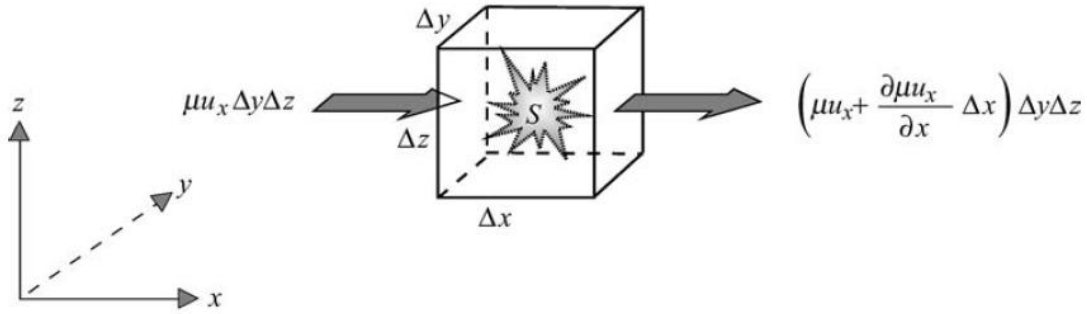


Figure A-1: Balance equation based on conservation

Similarly, the continuity equation can be derived under the assumption of conservation of mass:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \qquad \text{Eq. A-4}$$

In the 2DH approach, the assumption is made that the vertical accelerations are small enough to be neglected. Then Eq. A-3 describing the rate of change in the vertical, reduces to the hydrostatic pressure distribution over the vertical:

$$\frac{\partial p}{\partial z} = -\rho g \qquad \text{Eq. A-5}$$

Substituting Eq. A-5 into Eq. A-1 and Eq. A-2, and elaborating the viscous stresses yields the 2DH SWEs:

$$\frac{\partial \zeta}{\partial t} + \frac{\partial Hu}{\partial x} + \frac{\partial Hu}{\partial y} = 0 \qquad \text{Eq. A-6}$$

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + g\frac{\partial \zeta}{\partial x} - fv + \frac{1}{\rho_0}\left(\frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} + \frac{\partial \tau_{bx}}{H}\right) = 0 \qquad \text{Eq. A-7}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + g\frac{\partial \zeta}{\partial y} + fu + \frac{1}{\rho_0}\left(\frac{\partial \tau_{yx}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{by}}{H}\right) = 0 \qquad \text{Eq. A-8}$$

These three equations contain three unknown parameters, namely the water level (ζ) and the velocities in two directions (*u* and *v*). However, the analytical solution of these equations in a more simplified form can only be calculated in a very limited number of cases, which is very unlikely to occur in nature. A numerical approximation, by means of e.g. Delft3D-FLOW, is therefore required.

A.4 Time integration FLOW

In Delft3D-FLOW a finite difference scheme on a staggered grid in space is used with an ADI solver in time. ADI stands for Alternating Direction Implicit scheme which splits one time step into two stages. The idea is to use implicit numerical approximations in one spatial direction and explicit ones in the other spatial direction. The main advantage of ADI is that the computation time is much faster rather than solving implicit schemes only, as was found by Stelling (1983).

For example the numerical scheme of the two-dimensional heat equation (diffusion equation):

$$\frac{\partial T}{\partial t} = K\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right) \qquad \text{Eq. A-9}$$

would look like:

$$\frac{T_{m,n}^{t+½} - T_{m,n}^{t}}{\Delta t/2} = K\left(\frac{T_{m-1,n}^{t+½} - 2T_{m,n}^{t+½} + T_{m+1,n}^{t+½}}{\Delta x^2} + \frac{T_{m,n-1}^{t} - 2T_{m,n}^{t} + T_{m,n+1}^{t}}{\Delta y^2}\right) \qquad \text{Eq. A-10a}$$

$$\frac{T_{m,n}^{t+1} - T_{m,n}^{t+½}}{\Delta t/2} = K\left(\frac{T_{m-1,n}^{t+½} - 2T_{m,n}^{t+½} + T_{m+1,n}^{t+½}}{\Delta x^2} + \frac{T_{m,n-1}^{t+1} - 2T_{m,n}^{t+1} + T_{m,n+1}^{t+1}}{\Delta y^2}\right) \qquad \text{Eq. A-10b}$$

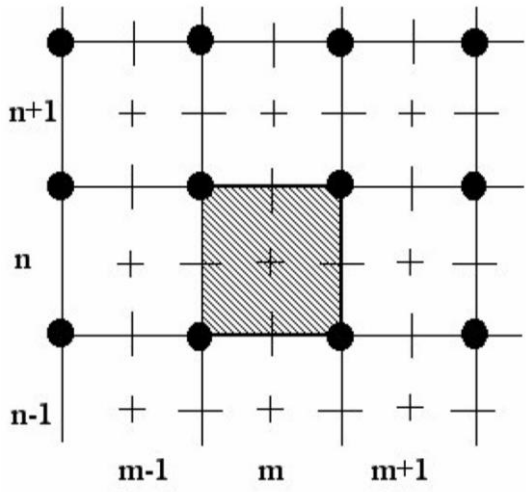
Note that in Eq. A-10a the second order derivative in x-direction is evaluated implicitly and the second order derivative in y-direction is evaluated explicitly. For the next time stage Eq. A-10b) this is vica versa. The SWEs in Delft3D-FLOW are solved in a similar way. A full description of the discretized SWEs is given by Broomans (2003).

To assure numerical stability, the Courant number (CFL condition) may not be too large due to the ADI-effect. The numerical solver turns out to be robust for a CFL upper bound of $4\sqrt{2}$ (Deltares, 2011, Delft3D-FLOW). With λ as a dimension of space, the Courant number is defined as:

$$CFL = \sqrt{gH}\frac{\Delta t}{\Delta \lambda} \leq 4\sqrt{2} \qquad \text{Eq. A-11}$$

A.5 Spatial integration FLOW

In space, a staggered grid is used which means that the water level and flow velocities are not computed on the same grid point. The arrangement of this staggered grid is shown by Figure A-2 (Stelling, 1983).



Legend:

full lines	The numerical staggered grid.
grey area	Computational control volume.
+	Water level (ζ), concentration of constituents, salinity, temperature.
-	Horizontal velocity component in x-, ξ-, u- or m-direction.
|	Horizontal velocity component in y-, η-, v- or n-direction.
	Depth

Figure A-2: Staggered grid

The continuity equation of Eq. A-6 is solved on the water level points and the momentum equations Eq. A-7 and Eq. A-8 on the velocity points. The choice of the spatial discretization of the advection terms had great influence on the accuracy, monotony and efficiency of the computational method, as shown in the river case. The discretisation of the advection term of Eq. A-6 in x-direction at (*m,n,k*) for example is (Broomans, 2003):

$$\frac{\partial Hu}{\partial x} = \left(H^{t+½}_{m+½,\mathrm{n},\mathrm{k}} u^{t+½}_{m+½,\mathrm{n},\mathrm{k}} - H^{t+½}_{m-½,\mathrm{n},\mathrm{k}} u^{t+½}_{m-½,\mathrm{n},\mathrm{k}}\right)/\Delta x \qquad \text{Eq. A-12}$$

with:

$$H^{t+½}_{m+½,\mathrm{n},\mathrm{k}} = \frac{1}{2}\left(\zeta^{t+½}_{m,n,k} + \zeta^{t+½}_{m+1,n,k} + d^{t+½}_{m+½,\mathrm{n}-½,\mathrm{k}} + d^{t+½}_{m+½,\mathrm{n}+½,\mathrm{k}}\right) \qquad \text{Eq. A-13}$$

Eq. A-12 is solved for both time stages. Note that in the first stage the expression is implicit and it is explicit in the second stage. For the explicit expression, the velocity on a cell face is partly determined by upstream information. In case of modelling a Chezy flow this effect may have large consequences near the upstream boundary, as found in the river case.

In Eq. A-13 the water levels are defined regarding a certain reference level, here MSL or z=0.00m. In Delft3D, the coordinate system of the vertical is positively upwards for the water level and for the bottom level downward (see Figure A-3). Eq. A-13 can also be rewritten in a more general form:

$$H = \zeta + d \qquad \text{Eq. A-14}$$

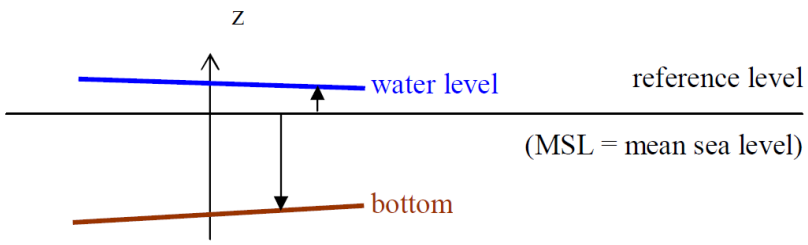


Figure A-3: Coordinate system of the vertical plane

In 2DH the total water level in a single cell ($H_{m,n}$) is determined by the four surrounding depth points ($d_{m\pm½,n\pm½}$). In Delft3D this is defined with the option DPSOPT; taking the minimum-, mean- or maximum value of the surrounding depth points. In a similar way depth values need to be determined at the cell interfaces (e.g. $d_{m+½,n}$), as the velocities are determined by the discharge times the interface area. This is done with the option DPUOPT; taking the minimum-, mean- or upwind value of the depth defined at the water level points ($d^{\zeta}_{m,n}$ and $d^{\zeta}_{m+1,n}$).

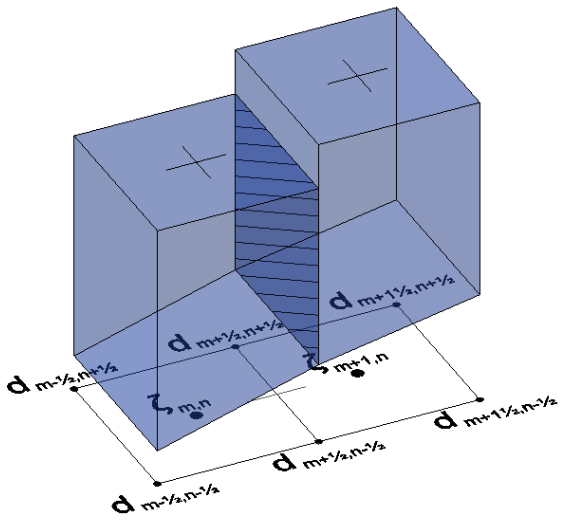


Figure A-4: Example of 2DH cells defined on a staggered grid with a sloping bathymetry

A.6 Mesh (structured)

Figure A-2 is an example of a perfect quadrangular grid. This is convenient when velocity gradients and water level gradients in x- and y-direction are of the same order of magnitude. On the other hand, relative small rates of change in one direction desires from an economic point of view lower model resolution. A rectangular grid is then preferred over the perfect quadrangular mesh. If these rectangular grid cells are more or less in line with the flow patterns, gradients are kept to a minimum. Meandering flow patterns thus desire a meandering mesh. A curvilinear grid is then preferred. Another aspect is that within one model, physical processes with different length scales can be of interest, e.g. Coriolis force and bathymetry-induced flow diversion. The limitations of computational extent can be overcome by (multiple) nested grids in one another. An example of a nested curvilinear grid is shown in Figure A-5.

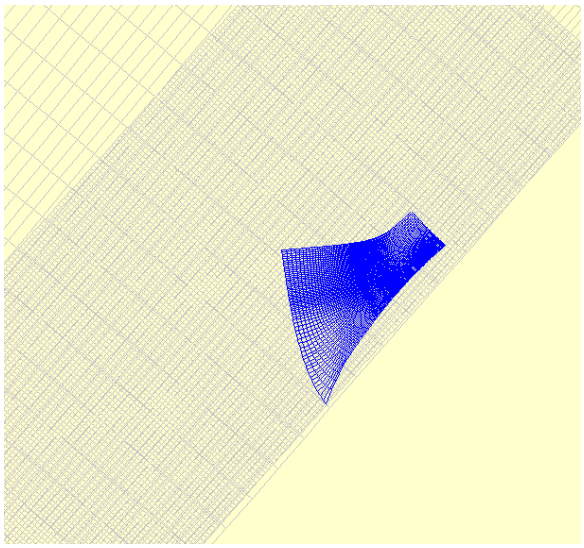


Figure A-5: Nested curvilinear grid

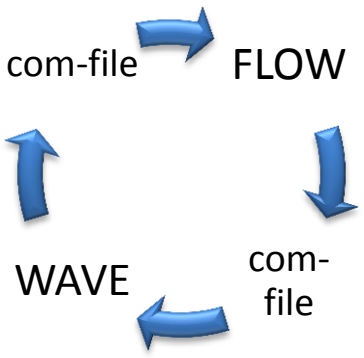
The flexibility of modelling with curvilinear grids in Delft3D is limited by the orthogonality requirement. To reduce the numerical error to a minimum, the grid cells need to satisfy a certain order of orthogonality in order to comply with the formulations of Eq. A-6 until Eq. A-8. Building complex models (e.g. deltas or estuaries with different length scales) is challenging and could take weeks due to these limiting factors.

A.7 Delft3D WAVE

The WAVE module uses an input grid, computational grid and output grid. The user defines the bathymetry, current fields and wind fields (if present) on the input grid. Delft3D-WAVE directs SWAN to perform the computations on the computational grid. The output grid is used in Delft3D-WAVE for the required output. Diagram 2 displays in essence which parameters are used for both FLOW and WAVE module:

Input FLOW parameters

- Wave forces ($F_{x,y}$)
- Shear stresses



- water level
- currents
- bathymetry
- (wind)

Input WAVE parameters

Diagram 2: Delft3D FLOW + WAVE coupling with the updated model parameters

The wave forces are by definition related to the radiation stresses under the waves, in according to Eq. 2-14). This is an important physical mechanism for wave-driven hydrodynamics in the nearshore. According to Dingemans et al. (1987), the wave forces formulated in terms of gradients of the radiation stresses could give rise to spurious currents. A more robust formulation is an expression in terms of wave energy dissipation. Dingemans et al. (1987) found a dissipation formulation with qualitatively correct current patterns:

$$F_x = -\frac{\partial S_{xx}}{\partial x} - \frac{\partial S_{xy}}{\partial y} = D\frac{k_x}{\omega} \qquad \text{Eq. A-15a}$$

$$F_y = -\frac{\partial S_{yy}}{\partial y} - \frac{\partial S_{yx}}{\partial x} = D\frac{k_y}{\omega} \qquad \text{Eq. A-15b}$$

where k and ω represent the wave number and radian frequency respectively. The rate of wave energy dissipation (D) is computed in SWAN, from the right hand side of Eq. 4-4, and consists of a contribution due to wave breaking and bottom friction.

Table A-2 describes which parameters are stored in the com-file, with the specific units and the location on the grid. The abbreviations used for the locations on the grid are clarified in Table A-1 (Deltares, 2011, Delft3D-QUICKPLOT). For a clarification of the staggered grid one is referred to Figure A-2.

Table A-1: Abbreviations for locations on grid

Abbreviation	Meaning
D	depth points, corners of grid cells
Z	water level points, cell centers of grid
UV	velocity points
Z (UV)	data averaged to water level points (data on file defined at velocity points)
c	defined at cell centers in vertical direction
i	defined at cell interfaces in vertical direction

File <com-runid.dat> and <com-runid.def>
File format NEFIS

Table A-2: Parameters stored in com-file

Quantity	Unit	Location
morphologic grid		D(1)
hydrodynamic grid		Z
inactive water level points		Z
thin dams		UV
temporarily inactive velocity points		UV
wind velocity	m/s	Z
hrms wave height	m	Z

hrms wave vector	m	Z
tp wave period	s	Z
wave dissipation	W/m2	Z
wave force	N/m2	Z (UV)
mass flux	m3/sm	Z (UV)
water level	m	Z
velocity (horizontal)	m/s	Z (UV), c
velocity in depth-averaged flow direction	m/s	Z (UV), c
velocity normal to depth-averaged flow direction	m/s	Z (UV), c
Depth-averaged velocity	m/s	Z (UV)
unit discharge (horizontal)	m2/s	Z (UV), c
Depth-averaged unit discharge	m2/s	Z (UV)
discharge potential	m3/s	D
spiral flow intensity	m/s	Z
u roughness parameter(2)	(2)	Z (U)
v roughness parameter(2)	(2)	Z (V)
max. bottom friction	N/m2	Z
salinity	ppt	Z, c
temperature	C	Z, c
vertical eddy diffusivity	m2/s	Z, c
initial bedload transport	m3/m	Z
avg bedload transport	m3/sm	Z
initial susp. transport	m3/m	Z
avg susp. transport	m3/sm	Z
initial bed level	m	D
fixed layer	m	D
sed. layer above fixed bed	m	D
time-varying bed level	m	D
cum. erosion/sedimentation	m	D
transport layer thickness	m	D
transp. l. thickness (wlvl)	m	Z
median grainsize	m	Z
transport layer	-	D
transport layer (wlvl)	-	Z
exchange layer	-	D
GUU grid distance	m	Z (U)
GVU grid distance	m	Z (U)
GVV grid distance	m	Z (V)
GUV grid distance	m	Z (V)
cell area water level point	m2	Z
cell area bottom point	m2	D
other fields(3)		Z(3)

(1) In the recent version of the program, the hydrodynamic grid connects the 3D co-ordinates of the cell centers of the computation grid points. This may change in a future release to represent the bounding boxes of the grid cells.

(2) The names and units of these fields are adjusted according to the roughness formulation used in the simulation. Units: Manning n in $s/m^{1/3}$, White-Colebrook/Nikuradse k in m, Chezy in $m^{1/2}/s$, z0 in m.

(3) New 2D and 3D fields of the same dimension as the grid are automatically detected. They are assumed to be located at water level points.

A.8 Interpolations Delft3D + SWAN

During the SWAN computations (on the computational grid) water levels and current velocities are obtained from Delft3D by bilinear interpolation of the Delft3D-FLOW grid (input grid). The output on the output grid is in turn obtained in Delft3D from bilinear interpolation of the computational grid from SWAN. If the grids are not overlapping the last known values on the domain are copied and extended to the non-overlapping areas. This remapping routine causes some loss in accuracy.

Input grid → Computational grid → Output grid

An example case can be as following:

- Input grid = output grid = FLOW.grd
- Computational grid = SWAN.grd

The input grid equals the output grid and is directly used in the FLOW module. The computational grid is used for the SWAN computation. In according to Deltares (2011, Delft3D-WAVE) and SWAN team (2013), there are no interpolation errors when *Input grid = computational grid = output grid*.

Appendix B

Conceptual wave coupling

B.1 Introduction

From a research of Luijendijk et al. (2010) has been concluded that the conventional wave modelling of Delft3D shows a good performance and is computational efficient. Also for the vertical velocity distribution with hydrodynamics induced by wave breaking, dissipation due to bottom friction, wind shear stresses acting on the surface and the tidal forcing. Modelling these nearshore hydrodynamics are based on the Delft3D FLOW + WAVE coupling. The coupling between both modules is a function of:

$$f\left(\zeta, \vec{u}, d, W, F_{x,y}, \tau\right) \qquad \text{Eq. B-1}$$

One desire is to include the modelling of both short-wave (gravity wave) and long wave (infra-gravity wave) effects. The stationary JONSWAP spectrum showed to be a realistic representation of a young sea state but is not applicable to swell (Holthuijsen, 2007). Nevertheless, the low frequency waves introduce the phenomena of surfbeat, as discussed in chapter 2. This is an important mechanism for sediment transport in the nearshore, as was found by Roelvink and Stive (1989).

Different models already exist which account for this phenomena. A short introduction for three of these models is given as a study to find a possible integrated wave model for D-Flow FM.

B.2 Surfbeat model

A surfbeat model (roller model) does not model the individual short waves (gravity waves or carrier waves), but models the forcing caused by the short waves. The varying wave forcing is called bound long waves. The bound long waves induce a difference in water level setup at the nearshore, called surfbeat. In the assumption of a stationary wave record, the mechanism of bound long waves is not accounted for. These processes are modeled separately by e.g. the surfbeat model (Dykes et al., 2003; Hsu et al., 2006). This is done by solving the short wave energy balance:

$$\frac{\partial E}{\partial t} + \frac{\partial}{\partial x}\left(E c_g \cos\theta\right) + \frac{\partial}{\partial y}\left(E c_g \sin\theta\right) = -D_w \qquad \text{Eq. B-2}$$

Here E represents the short wave energy defined by Eq. 2-16 and D_w represents the dissipation of wave energy. The model is phase-averaged and contains a phase-randomizer with a uniform distribution on $[0, 2\pi]$ (Deltares, 2011, Delft3D-FLOW).

A limitation of this surfbeat model is the restrict application for narrow-banded wave spectra, both in frequency and directional space. A separate wave model is needed to compute the wave directions and wave periods of the short waves. This could be the phase-averaged wave model SWAN for example.

B.3 Boussinesq and non-hydrostatic models

A non-hydrostatic model is SWASH for example, developed by Zijlema et al. (2011) at the Delft University of Technology. A Boussinesq type of wave models is TRITON (Borsboom et al., 2000) for example. Both Boussinesq and non-hydrostatic models are phase resolving which accounts for all relevant nearshore processes on large detail (e.g. shoaling, refraction, reflection and triad interactions). These models are potentially accurate but computational consuming compared to hydrostatic models.

B.4 Integrated wave modelling

Investigating the previous mentioned models, desires a combination of all models except the non-hydrostatic part. In the primary 2DH assumptions, hydrostatic computations will satisfy. An integration of short wave modelling combined with the modelling of the infra-gravity waves is desired. Here for, non-stationary and phase-resolved computations are required. To realise this, the stationary phased-averaged SWAN computation somehow need to become phase-resolved and non-stationary.

An interesting configuration is by making the input spectrum for SWAN time-dependent. After each wave computation the action density spectrum can externally be stored, becoming time-dependent $N(\sigma, \theta, t)$. This can be transferred into a wave-record which is quasi-phase-resolved by imposing a similar phase-randomizer as for the surfbeat model (Deltares, 2011, Delft3D-FLOW). Only to take account for the low frequency waves, the phase-randomizer needs to be correlated to the historical wave record in order to find a smooth overlap of e.g. the infra-gravity waves. This non-stationary wave record is then stored and expands over time, with the phases of the bound long waves being locked. An example of a time expanding wave record is shown in the figure below.

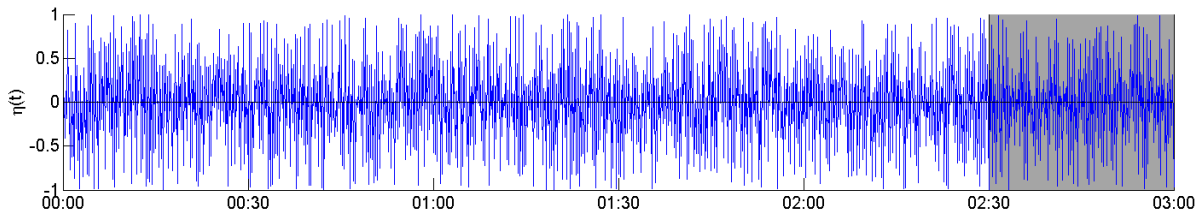


Figure B-1: Example a non-stationary wave record with a statistical analysis over the grey region

For the next wave computation, a spectral analysis is done over the last 30minutes of the wave record in order to make the SWAN computation stationary. In this spectral analysis the wave phases

are again made uniformly distributed. Now the amplitudes and frequencies of the infra-gravity waves are included for the new wave run.

The inclusion of the wave-induced hydrodynamics with the 'flow' induced hydrodynamics is now a function of:

$$f\left(N, \zeta, \vec{u}, d, W, F_{x,y}, \tau\right) \quad \text{Eq. B-3}$$

where N represents the action density spectrum defined as:

$$N = N(\sigma, \theta, x, y, t) \quad \text{Eq. B-4}$$

The time variable in this expression is related to the wave record of Figure B-1. This wave record varies over space. Translating this into a model would contain a space varying spectral condition in the form of Eq. B-4. The input file for SWAN would then be a 2D spectrum file (*.sp2) for example.

B.5 Discussion

Although this conceptual wave model takes account for the full frequency domain, it will most likely be computational intensive. The relevance of developing this methodology in a most efficient way, outweighed to the conventional wave modelling packages, depends on the model developers. A presumable extension to modelling infra-gravity waves will be the inclusion of the surfbeat model in D-Flow FM. This model has been verified in according to Roelvink et al. (1992), and is (also) successfully integrated in the Delft3D-FLOW module.

Another development within Deltares, for morphological modelling purposes, is the research of implementing the XBeach formulation directly in D-Flow FM. This formulation uses the action balance equation in the form of Eq. 4-4. Recent developments of XBeach are curvilinear grid models (see Roelvink et al., 2012) which showed good morphological predictions to field data.

A plausible configuration of modelling full spectrum waves in D-Flow FM will be a selection of both of these options. For hydrodynamic purposes SWAN combined with the surfbeat model, and for morphological purposes the XBeach model.

Appendix C

Sensitivity analysis planar beach case

C.1 Modelling approach

A sensitivity analysis is carried out of a Delft3D model to find a better match with the 1D wave energy balance model. These results are of interest for the case: Application of wave-driven alongshore currents (planar beach). The wave propagation with respect to shoaling and refraction is investigated by tuning the Delft3D settings. The model response is investigated by expressing the alongshore current velocities.

The original settings of the planar beach case are presented at first. Followed by a sensitivity analysis to the following parameters:

- Directional spreading
- Horizontal eddy viscosity
- Horizontal eddy diffusivity
- Breaker index
- Bottom friction formulation SWAN
- Bottom friction coefficient SWAN

For the sack of accuracy the 1D model is extended with the Soulsby fitting coefficients of Fredsøe 1984 (Soulsby et al., 1993). The breaker index parameter of the 1D model is set to 0.73. The wave parameters of the 1D model are kept constant. The results are shown in Figure C-1 through Figure C-6.

Case 0:

Directional spreading	4
Horizontal eddy viscosity	0
Horizontal eddy diffusivity	0
Soulsby fitting coefficients	Fredsøe 1984
Breaker index	0.73
Bottom friction formulation SWAN	JONSWAP
Bottom friction coefficient SWAN	0.067

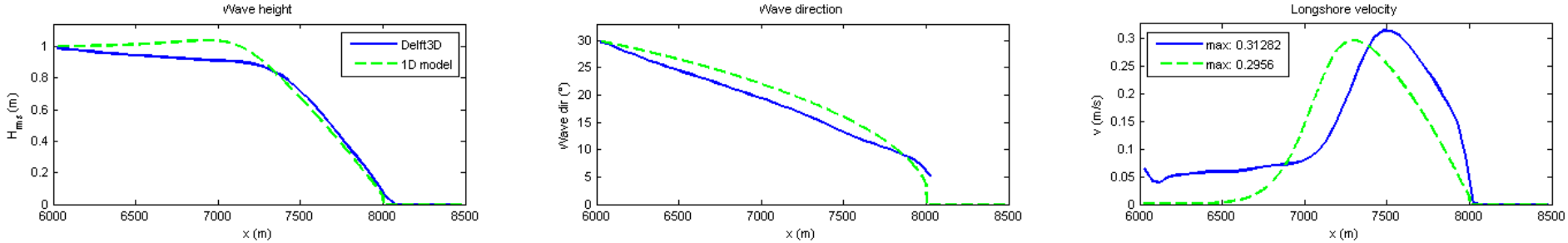


Figure C-1: Case 0 with the original model setting

Case 1:

Directional spreading	4
Horizontal eddy viscosity	**1**
Horizontal eddy diffusivity	**10**
Soulsby fitting coefficients	Fredsøe 1984
Breaker index	0.73
Bottom friction formulation SWAN	JONSWAP
Bottom friction coefficient SWAN	0.067

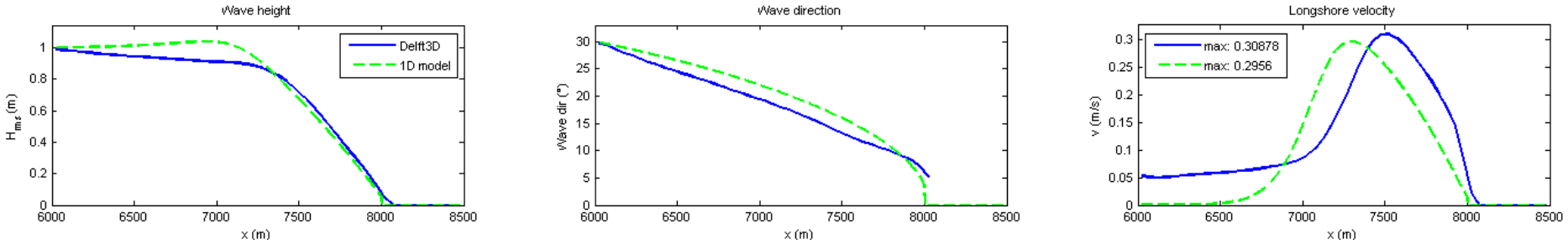


Figure C-2: Case 1 with a modified viscosity parameter

Case 2:

Directional spreading	**60**
Horizontal eddy viscosity	0
Horizontal eddy diffusivity	0
Soulsby fitting coefficients	Fredsøe 1984
Breaker index	0.73
Bottom friction formulation SWAN	JONSWAP
Bottom friction coefficient SWAN	0.067

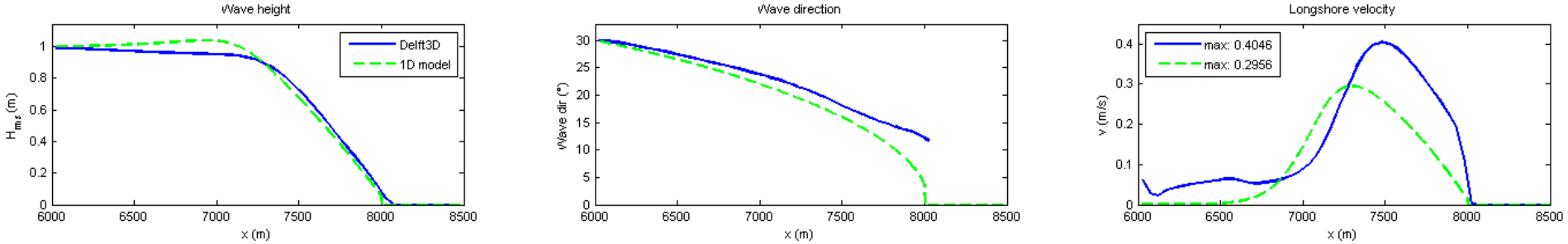


Figure C-3: Case 2 with a modified directional wave spreading

Case 3:

Directional spreading	4
Horizontal eddy viscosity	0
Horizontal eddy diffusivity	0
Soulsby fitting coefficients	Fredsøe 1984
Breaker index	0.73
Bottom friction formulation SWAN	**None**
Bottom friction coefficient SWAN	**-**

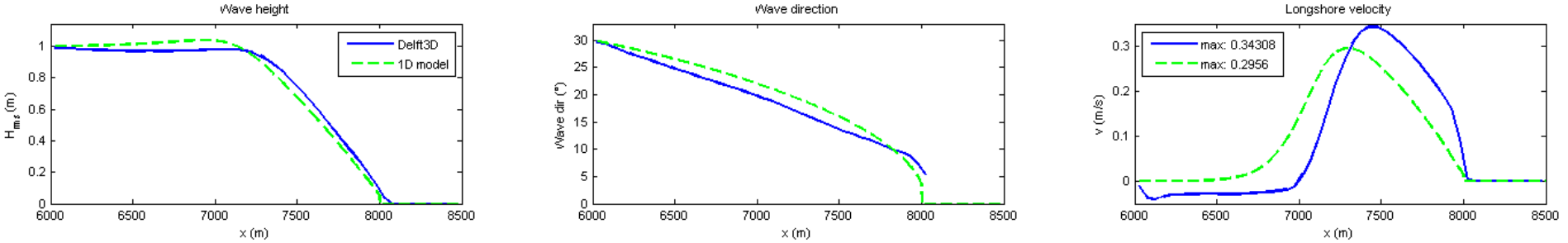


Figure C-4: Case 3 with no bottom friction in SWAN

Case 4:

Directional spreading	4
Horizontal eddy viscosity	0
Horizontal eddy diffusivity	0
Soulsby fitting coefficients	Fredsøe 1984
Breaker index	**0.60**
Bottom friction formulation SWAN	JONSWAP
Bottom friction coefficient SWAN	0.067

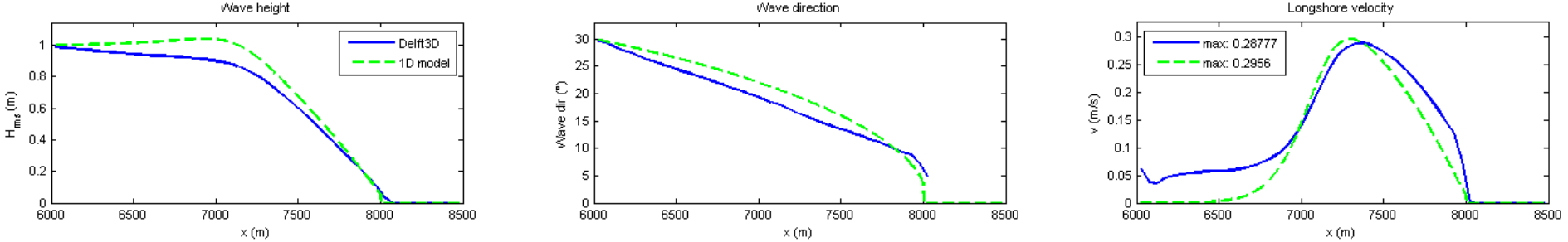


Figure C-5: Case 4 with a modified breaker index

Case 5:

Directional spreading	**10**
Horizontal eddy viscosity	0
Horizontal eddy diffusivity	0
Soulsby fitting coefficients	Fredsøe 1984
Breaker index	**0.60**
Bottom friction formulation SWAN	JONSWAP
Bottom friction coefficient SWAN	**0.08**

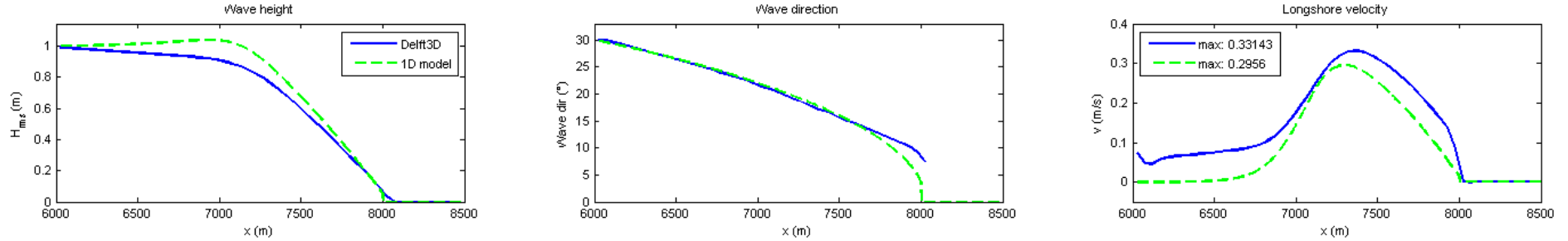


Figure C-6: Case 5 with a modified directional wave spreading, breaker index and bottom friction coefficient

C.2 Model results

From these results is found that directional spreading and bottom friction in SWAN are important model parameters for wave propagation in the nearshore. Increasing the directional spreading shows a latency of the wave refraction. Decreasing the bottom friction coefficient induces a larger shoaling effect. However, both parameters give larger alongshore velocities as a response.

Tuning the breaker index parameter, results in an offshore shift of wave breaking. This matches the 1D velocity profile closely, but mismatches the wave propagation. Combining all three wave parameters (directional spreading, bottom friction, and the breaker index) does not make things better. The shoaling effect is incorrect and the offshore velocities are too high.

The model settings of case 0 are a compromise between realistic absolute velocities and reasonable wave propagation in the nearshore. An interesting notice from these test cases is the effect of horizontal eddy viscosity, between case 0 and case 1. In case 1 this effect smoothes the velocity profile at the offshore region.