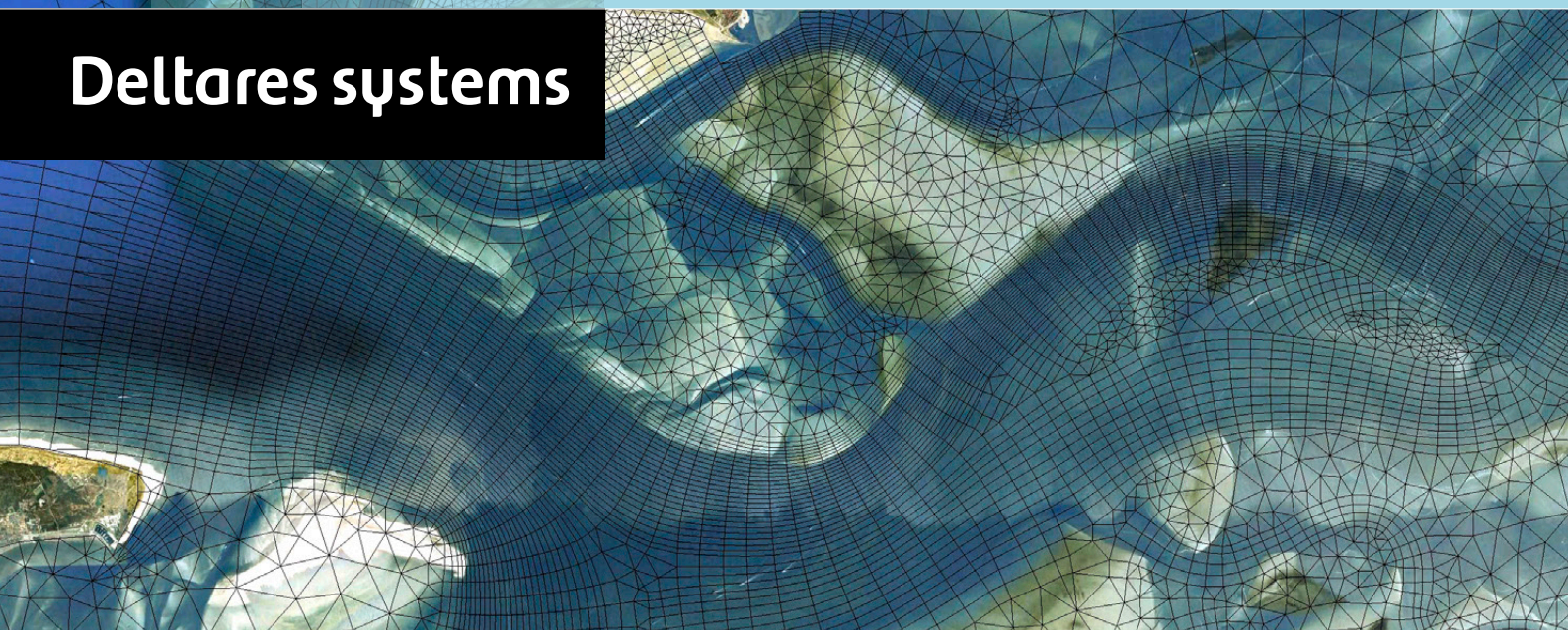
1D/2D/3D Modelling suite for integral water solutions

Delft3D Flexible Mesh Suite

Deltares systems



D-Flow Flexible Mesh

Deltares

Enabling Delta Life

Validation Document

D-Flow Flexible Mesh

Validation Document

Version: 1.4.6
SVN Revision: 51533

18 March 2020

D-Flow Flexible Mesh, Validation Document

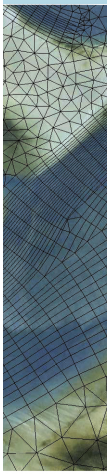
Published and printed by:
Deltares
Boussinesqweg 1
2629 HV Delft
P.O. 177
2600 MH Delft
The Netherlands

telephone: +31 88 335 82 73
fax: +31 88 335 85 82
e-mail: info@deltares.nl
www: https://www.deltares.nl

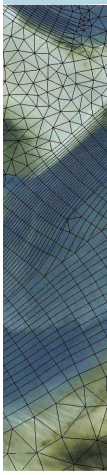
For sales contact:
telephone: +31 88 335 81 88
fax: +31 88 335 81 11
e-mail: software@deltares.nl
www: https://www.deltares.nl/software

For support contact:
telephone: +31 88 335 81 00
fax: +31 88 335 81 11
e-mail: software.support@deltares.nl
www: https://www.deltares.nl/software

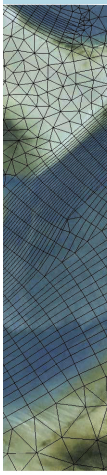
Contents



List of Figures



List of Tables



1 Introduction

1.1 Status of this document

This document has no official status yet, it is still in the review process. Distribution to others should always go via Deltares. Please contact sales@deltaressystems.nl for more information.

1.2 Introduction

D-Flow Flexible Mesh (or D-Flow FM, in short) is a 1D-2D-3D hydrodynamical simulation package that runs on flexible meshes. The term *flexible* concerns the familiar curvilinear meshes (like in WAQUA or Delft3D) combined with triangles, pentagons, hexagons and one-dimensional channel networks, all in one single mesh. The present document comprises the compilation of test documents related to the validation activities that have taken place throughout the course of development of D-Flow FM. This present chapter provides on overview of the scope and purpose of the present validation document.

1.3 Scope

The present validation document concerns the properties and validity of D-Flow FM; it focuses on the computational part of D-Flow FM. As a consequence, the contents of this document treat the aspects of the bare *flow motion simulation* capabilities of D-Flow FM. For example, the pre- and postprocessing, and the coupling with other modules around D-Flow FM, such as a wave module, a water quality module, a particle tracking module and an ecology module, are beyond the scope of this document.

The particular focus of this document is on *two-dimensional* flow simulation. The coupling with one-dimensional flow systems is taken care of within the Sobek development course, whereas three-dimensional flow systems (through σ- or z-layer type grids) are considered only loosely to be part of the scope of the project *Next Generation Hydro Software*. The particular focus is on the representation of Waqua functionalities within the modelling framework.

1.4 Purpose

The primary purpose of D-Flow FM is to solve the two-dimensional shallow water equations. These shallow water equations describe incompressible flow motion, based on the assumption that the vertical length scale is much smaller than the plane length scales. The equations are solved on a mesh that can contain triangles, quadrilaterals, pentagons and hexagons. The secondary purpose of D-Flow FM is to facilitate the coupling of one-, two- and three-dimensional grids.

The primary purpose of this validation document is to convey a representative image of the capabilities and quality of D-Flow FM when it comes to two-dimensional flow simulation. To that end, the various essential parts of the flow kernel are considered, such as advection, diffusion, coriolis, boundary conditions, etc. For this image, several types of validation models have been built.

1.5 Validation approach

The validation models can be of strict analytical nature (comparison with analytical results), of experiment nature (comparison with laboratory results), of schematic nature (hypothetical models with only focus on a considerably limited number of functionalities) or of real-world nature (practical engineering models). The validation approach is inspired by the Dee et al. (1994) guidelines for documenting the validity of computation modelling software. The key elements of this approach and the motivation for specific validation models are described in chapter 2.

1.6 Outline

The validation of the multiple aspects of the D-Flow FM package is treated component wise, in line with the several components of the shallow flow equations themselves. Thus, core components like advection and diffusion are treated in dedicated chapters: chapter 3 and chapter 5. Flooding and drying, a topic closely related to advection, is dedicated a separate chapter as well: chapter 4.

External forcings to the flow field can be exerted in different ways. In this document, three types of external forcings are considered: acceleration due to the rotation of the Earth (coriolis, treated in chapter 6), deceleration due to bed friction (treated in chapter 7) and acceleration due wind stresses (treated in chapter 8).

For the computation of typical Dutch river flows, the D-Flow FM package must be able to deal with weirs and barriers. The test reports for these functionalities are provided in chapter 9 and chapter 10, respectively.

Separate chapters are dedicated to specific technical but key elements of the software package. Within this context, multiple types of boundary conditions are considered (treated in chapter 11), as well as input and output functionality (in chapter 12). Some miscellaneous functionalities are treated in chapter 13.

After having discussed basic functionalities by means of basic test models, attention is paid to test models with a stronger link to the engineering practice. Within this context, several real-world models are highlighted in chapter 14, ranging from a medium-scale Frisian inlet case to a large-scale Californian coast case.

Practically all the test cases reported in this document are provided a relatively small document with a discussion of the quality assessment of the case under consideration. Some of these cases, however, are selected to be subjected to a more extensive quality assessment at a more profound academic level. Two cases — a dambreak case and a coastal wave run-up case — are considered for this purpose in chapter 15.

1.7 Disclaimer

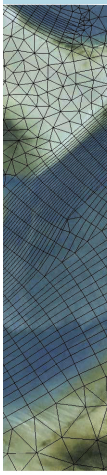
This document contains information about the quality of a complex modelling tool. Its purpose is to assist the user in assessing the reliability and accuracy of computational results, and to provide guidelines with respect to the applicability and proper use of the modelling tool. This document does not, however, provide mathematical proof of the correctness of results for a specific application. The reader is referred to the License Agreement for pertinent legal terms and conditions associated with the use of the software.

The contents of this validation document attest to the fact that computational modelling of complex physical systems requires great care and inherently involves a number of uncertain factors. In order to obtain useful and accurate results for a particular application, the use of high-quality modelling tools is necessary but not sufficient. Ultimately, the quality of the computational results that can be achieved will depend upon the adequacy of available data as well as a suitable choice of model and modelling parameters.

2 Model validity

This chapter reflects on the rationale for the validation of the software package D-Flow FM. Having a clear scope and purpose, set in the previous chapter, the testing philosophy is explained in more detail. For a full description of all the technical backgrounds, as well as all underlying assumptions and approximations, the reader is referred to the technical reference manual.

The IAHR/AIRH guidelines (Dee et al., 1994) are followed along the validation course, particularly regarding the several types of validation cases that have been developed. Some of the core aspects of the guidelines are highlighted in this chapter, including the types of validation cases and the claims and substations that corroborate the intented working of D-Flow FM.



2.1 Testing philosophy

The methodology for testing D-Flow FM has two main components: regression tests and validation tests. The regression tests aim at consistency in the D-Flow FM results in due course of the software developments. These tests particularly comprise functional elements of the software, for instance the proper working of the write routines and read routines for flow field data.

The validation tests aim at examining the D-Flow FM results at a conceptually higher level. These tests are intended to be provide an image of the actual quality of the results. To this end, multiple sources of data for comparison could be utilized, such as analytical solutions or data from experiments. In order to elucidate the functionalities of D-Flow FM and to offer a concise overview of the results of the validation efforts, a framework of *claims* and *substantiations* is adopted. The formulation of these claims and substantiations needs some clarification.

Within the context of this validation document, a claim should be perceived as a falsifiable statement intended to shape the expectations of D-Flow FM users. The adjective *falsifiable* does not indicate that the statement is false, but rather, it means that if the statement were false, then its falsehood could be demonstrated. If a statement is falsifiable, then it could be depicted as testable.

In line with the rationale behind the word *claim*, the word substantiation could merely be interpreted as a corroboration of a claim. Such a substantiation tends to be a confirmation of a claim posed, but cannot be interpreted as a proof of a claim. As a result, a substantiation in this report is merely intended to convey useful information on the testability of a claim posed.

2.2 Methodology

The IAHR/AIRH document (Dee et al., 1994) has formulated guidelines for documenting the validity of computation modelling software. Core elements of these guidelines comprise an overview of the assumptions and approximations that were made during the design and implementation of the model and contains a set of claims on the performance of the model as well as substantiations as a corroboration of the proper functioning of the model. For the validation of computational modelling software, multiple types of testcases are considered: analytical testcases, laboratory testcases, schematic testcases and real-world applications. These four types are briefly discussed.

Analytical testcases

Most ideally, an analytical solution is present for a certain flow case. The presence of an analytical solution obviates the comparison of D-Flow FM results with results from other software packages. Thus, the quality of the D-Flow FM results can be examined most objectively.

Laboratory testcases

Experimental data can be used to compare computational outcomes with. Affined to the analytical testcases, laboratory testcases thus exclude the stringent need for comparison with other software packages. Nonetheless, the comparison with other software packages does convey relevant additional insights, since experimental circumstances are always captured somehow in computational assumptions.

Schematic testcases

It is possible that for a relatively simple testcase no analytical solution is known. Nevertheless, such a testcase can actually succesfully be used for a qualitative analysis. Such a comparison can yield considerations from a rather academic point of view (for instance, consistency and expected behavior).

Real-world applications

D-Flow FM is intended to be used for applications in hydraulic engineering, coastal engineering and physical oceanography. From that perspective, the former three types of testcases are necessary but not sufficient. Real-world applications are added to the framework of testcases to examine the D-Flow FM performance. Real-world applications are more extensively dealt with within the framework of acceptance tests, not addressed in this document.

2.3 Claims and substantiations

In this section, the claims are formulated associated with the functionalities of D-Flow FM. The claims are ordered along the pathways of the core elements of D-Flow FM, such as advection, diffusion, friction, etc. The claims are accompanied by a substantiation in which a link is laid to specific testcases that corroborate the claim. A substantiation can comprise an analytical testcase, a laboratory testcase, a schematic testcase or a real-world application (see section 2.2).

2.3.1 General

General claims are formulated particularly for testcases with a strong focus on the choice of the grid type, refinement studies and the overall functioning of D-Flow FM. Moreover, claims are formulated for output files and parallel running.

Claim 2.3.1.1:	D-Flow FM can accurately simulate steady and unsteady hydrostatic flow on grids consisting of triangles, quadrilaterals, pentagons and hexagons.
Substantiation:	Cases 7.2, 7.3, 3.2, 3.3, 3.1, 3.4, 4.3, 3.5, 4.4, 4.1, 4.2, 7.4.
Remarks:	As such, the claims appears quite general. The focus of this claim, however, is mainly on the various types of grids that are supported, as well as the convergence behavior of error norms in refinement studies.

Claim 2.3.1.2: D-Flow FM is suitable for the prediction of the tidal dynamics in estuaries or coastal seas.

Substantiation: Cases 14.1, 14.2, 14.3, 14.4, 14.5, 14.6.

Claim 2.3.1.3: D-Flow FM is able to produce his-files, map-files and restart files with output timeseries and field output data.

Substantiation: Cases 12.1, 12.2, 12.3, 12.4, 12.5.

Claim 2.3.1.4: D-Flow FM can be run on parallel machines.

Substantiation: Cases.

2.3.2 Advection

Advection plays a role in each flow situation with non-zero velocity gradients. In that sense, many flow situation could serve as a test case in this respect. By the claims formulated below, the particular focus, however, is on dambreak-type problems and flooding/drying-type problems, since these cases contain significant advection aspects.

Claim 2.3.2.1: D-Flow FM can be used for an accurate prediction of flows resulting from dam breaks.

Substantiation: Cases 3.4, 4.3, 3.5, 4.4.

Claim 2.3.2.2: D-Flow FM can accurately simulate the propagation of long waves.

Substantiation: Cases.

Claim 2.3.2.3: D-Flow FM can accurately simulate the propagation of short waves.

Substantiation: Cases 4.1, 4.2.

Claim 2.3.2.4: D-Flow FM can accurately simulate drying and flooding of tidal areas.

Substantiation: Cases 4.3, 4.4, 4.1, 4.2.

2.3.3 Diffusion

As soon as a non-zero eddy viscosity is used in a flow simulation, velocity gradient will give rise to diffusion of momentum. The claims listed below focus on analytical testcases that provide an image of the effects of diffusion triggered by a velocity gradient induced by sidewalls.

Claim 2.3.3.1: D-Flow FM accurately uses the horizontal eddy viscosity concept for no-slip walls.

Substantiation: Cases 5.2.

Claim 2.3.3.2: D-Flow FM accurately uses the horizontal eddy viscosity concept for partial-slip walls.

Substantiation: Cases 5.1.

2.3.4 Friction

Bed friction is omnipresent in engineering applications. The friction can be prescribed by means of a couple of formulations, both in a non-linear framework (standard) and a linear framework (not standard).

Claim 2.3.4.1: D-Flow FM can take into account the impact of the space varying shear stress at the bottom. The input coefficients for bed friction can vary in space.

Substantiation: Cases.

Claim 2.3.4.2: D-Flow FM can take into account several friction formulations (Chezy, Manning, White-Colebrook or roughness height z_0).

Substantiation: Cases 7.1.

Claim 2.3.4.3: D-Flow FM is able to apply linearized friction in two-dimensional flow simulations.

Substantiation: Cases **??**.

2.3.5 External forcing

The claims listed below focus on external forcing that is exerted to the flow field. In this respect, the particular focus in on wind, but also on the Coriolis' force (due to the rotation of the earth).

Claim 2.3.5.1: D-Flow FM can take into account the impact of the Coriolis force associated with the rotation of the earth.

Substantiation: Cases 6.2, 6.3, 6.1.

Claim 2.3.5.2: D-Flow FM can be used for an accurate prediction of wind driven flow.

Substantiation: Cases 8.3, **??**.

Claim 2.3.5.3: D-Flow FM can be used for the prediction of storm surges due to cyclone winds.

Substantiation: Cases 14.6.

Claim 2.3.5.4: D-Flow FM can accurately simulate the effects of space and time varying wind stresses at the free water surface.

Substantiation: Cases 14.6.

2.3.6 Hydraulic structures

Various hydraulic structures can be dealt with by D-Flow FM. The claims below focus on the several types of weirs and barriers and related hydraulic aspects.

Claim 2.3.6.1: D-Flow FM can be used to investigate the hydrodynamic impact of hydraulic structures, such as gates, weirs and barrier.

Substantiation: Cases 9.1, 9.2, 9.4, 9.5.

Claim 2.3.6.2: D-Flow FM can accurately simulate subcritical and supercritical flows and the transition region when the flow changes from subcritical to supercritical or vice versa.

Substantiation: Cases 9.1, 9.2, 9.4, 9.5.

2.3.7 Constituents

D-Flow FM can deal with passive constituents (tracers) and active constituents (salinity).

Claim 2.3.7.1: D-Flow FM can accurately simulate the transport of dissolved material.

Substantiation: Cases 3.6.

Claim 2.3.7.2: D-Flow FM can be used for an accurate prediction of the density driven flow.

Substantiation: Cases 11.4.

2.3.8 Miscellaneous

Yet unmentioned functionality is partly considered in this section. The scope of this miscellaneous group is restricted to boundary conditions functionality, sources/sinks and dry points.

Claim 2.3.8.1: D-Flow FM can deal with boundary conditions that are prescribed as timeseries, as harmonic components and as astronomic components.

Substantiation: Cases 11.1, 11.2, 11.3, 11.6, 11.7, 11.8.

Claim 2.3.8.2: D-Flow FM can deal with boundary conditions for the water level, the velocity and the discharge.

Substantiation: Cases 11.1, 11.2, 11.3.

Claim 2.3.8.3: D-Flow FM can deal with boundary conditions for derived quantities by means of a Qh-table, a Neumann-type condition for the water level and a Riemann invariant.

Substantiation: Cases 11.6, 11.7, 11.8, 11.9, 11.5.

Claim 2.3.8.4: D-Flow FM can take into account time varying sources and sinks for e.g. river flows and discharges from outfalls.

Substantiation: Cases.

Claim 2.3.8.5: D-Flow FM can take into account the explicit specification of dry points.

Substantiation: Cases 13.1, 13.2.

3 Advection

3.1 Bélanger surface profile

Purpose

The purpose of this validation case is to examine the performance of D-Flow FM for a schematized channel flow simulation. For stationary flow through a river with a rectangular cross-section, the Bélanger surface profile equation can be utilized to compare the numerical solution with.

Linked claims

Claims that are related to the current test case are:

◇ claim 2.3.1.1: D-Flow FM can accurately simulate steady and unsteady hydrostatic flow on grids consisting of triangles, quadrilaterals, pentagons and hexagons

Approach

A straight channel with a rectangular cross-section is defined. Given an inflow discharge Q, a channel width B, a bottom slope i_b and a Chézy friction factor C, the distance between the free surface profile and the bed profile can be described by the Bélanger equation for d as the water depth:

$$\frac{\mathrm{d}d}{\mathrm{d}x} = i_b \frac{d^3 - d_e^3}{d^3 - d_g^3} \tag{3.1}$$

with d_e the equilibrium depth and d_g the limit depth (associated with $\mathrm{Fr} = 1$) following:

$$d_e = \left(\frac{Q^2}{B^2 g}\right)^{1/3} \qquad \text{and} \qquad d_g = \left(\frac{Q^2}{B^2 C^2 i_b}\right)^{1/3}. \tag{3.2}$$

Given a certain inflow discharge Q_{in} and a certain outflow water level condition h_{out}, the surface profile can hence numerically be estimated in the most simple way as:

$$\frac{d_i - d_{i-1}}{\Delta x} = i_b \frac{d_i^3 - d_e^3}{d_i^3 - d_g^3}, \tag{3.3}$$

having $d_i = h_{out} + i_b L$ at the outflow boundary. This, in fact *semi-analytical*, solution can be used for comparison.

Model description

For this test case, one single computational grid is generated. The computational domain has the sizes $L \times B = 100$ km $\times$ 20 m. The grid counts 200×1 cells. The cell size is 500×20 m^2 everywhere. The bed slope i_b is 10^{-4}. The inflow discharge is $Q = 600$ m^3/s. The Chézy coefficient is $C = 65$ m$^{1/2}$/s. The outflow water *level* is set equal to 0 m (w.r.t. reference), i.e. the water *depth* is hence equal to 10 meters.

Recall that the water depth is computed as the difference between the upstream water level (computed at the *cell center*) and the bed level at the velocity point (computed at the *cell face*), invoking a $\Delta x/2$ spatial shift. In the computational model, the bed level at the outflow boundary is equal to -10 m+NAP, whereas a water level equal to 0 m+NAP is imposed at

the outflow boundary. Since the location for the water level outflow boundary condition is also $\Delta x/2$ outside the grid (mirrored location), the water level boundary conditions yields a bed level equal to $b|_{x=L+\Delta x} = -i_b \cdot (L + \Delta x) = -10^{-4} \cdot 100500 = -10.05$ m (given a zero bed level at the entrance of the channel). Hence, the outflow water depth equals $d_{out} = h_{out} - b_{out} = 0 - (-10.05) = 10.05$ m, at $x = L + \Delta x$.

Results

The result from D-Flow FM for the water *depth* is shown in Figure 3.1 in combination with its semi-analytical equivalent. The semi-analytical solution is based on the equation for the Bélanger surface profile.

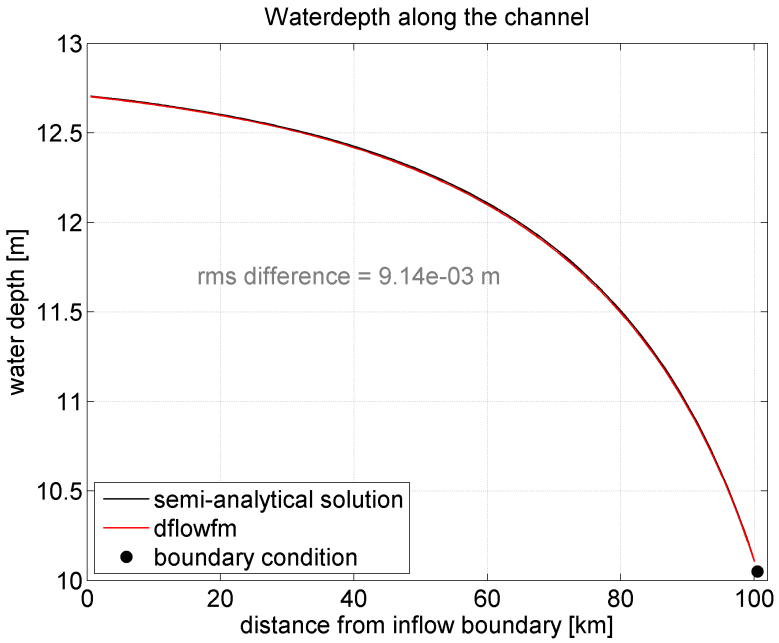


Figure 3.1: *Comparison of the numerical solution and the semi-analytical solution for the water depth.*

The root-mean-square difference between the numerical outcome from D-Flow FM and the semi-analytical solution is shown in Figure 3.1 as well. This rms-difference is of the order of 10^{-3} m.

Conclusion

For the Bélanger test case under consideration, the numerical solution approaches the semi-analytical solution up to an overall root-mean-square difference of the order of 10^{-3} m.

Version

This test has been carried out with version dflow-fm-x64-1.1.116.36629.

3.2 Refinement study for Bélanger channel flow using Cartesian grids

Purpose

The purpose of this validation case is to examine the performance of D-Flow FM for a schematized channel flow simulation. For stationary flow through a river with a rectangular cross-section, the Bélanger surface profile equation can be utilized to compare the numerical solution with. This validation case focuses on the effects of the refinement of a Cartesian grid. The topography comprises a linearly varying bathymetry with a constant slope.

Linked claims

Claims that are related to the current test case are:

◇ claim 2.3.1.1: D-Flow FM can accurately simulate steady and unsteady hydrostatic flow on grids consisting of triangles, quadrilaterals, pentagons and hexagons

Approach

A straight channel with a rectangular cross-section is defined. Given an inflow discharge Q, a channel width B, a bottom slope i_b and a Chézy friction factor C, the distance between the free surface profile and the bed profile can be described by the Bélanger equation for d as the water depth:

$$\frac{\mathrm{d}d}{\mathrm{d}x} = i_b \frac{d^3 - d_e^3}{d^3 - d_g^3} \tag{3.4}$$

with d_e the equilibrium depth and d_g the limit depth (associated with $\mathrm{Fr} = 1$) following:

$$d_e = \left(\frac{Q^2}{B^2 g}\right)^{1/3} \qquad \text{and} \qquad d_g = \left(\frac{Q^2}{B^2 C^2 i_b}\right)^{1/3}. \tag{3.5}$$

Given a certain inflow discharge Q and a certain outflow water level condition h_{out}, the surface profile can hence numerically be estimated in the most simple way as:

$$\frac{d_i - d_{i-1}}{\Delta x} = i_b \frac{d_i^3 - d_e^3}{d_i^3 - d_g^3}, \tag{3.6}$$

having $d_i = h_{out} + i_b L$ at the outflow boundary. This, in fact *semi-analytical*, solution can be used for comparison. If the grid is refined several times, the order of accuracy of the numerical integration routines can be assessed. For this purpose, only Cartesian grids are considered (triangular grids are considered in case e02-f01-c013).

Since grid refinement is one of the main aspects of this particular testcase, the boundary conditions are specified grid independent by means of a constant Dirichlet discharge boundary condition upstream and a constant Neumann water level boundary condition downstream. As Equation (3.4) shows, a fixed water level implies a fixed water level gradient (and *vice versa* as will be shown below).

Model description

For this test case, five computational grids are generated. The outflow parts of these grids are shown in Figure 3.2. The five grids are of Cartesian type. The longitudal size L of the domain is 10 km, whereas the lateral size B of the domain is 500 m.

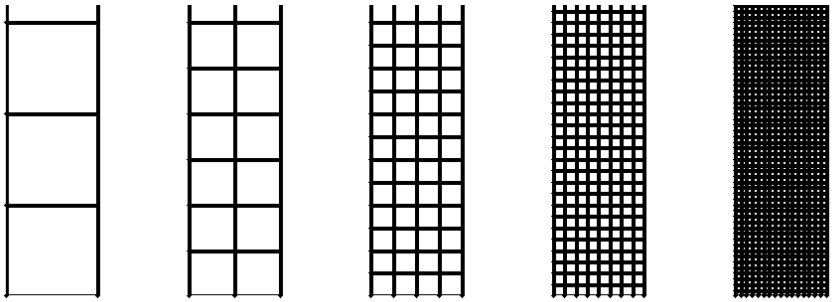


Figure 3.2: *From left to right: five Cartesian grids in order of refinement grade. Each refinement comprises twice as much cells in each direction. Only the outflow part of the grid is shown.*

A discharge Q equal to 2500 m^3/s is prescribed. The bottom slope i_b is prescribed[1] to be 10^{-4}. The bottom level at the inflow boundary is 0 m (w.r.t. reference) and at the outflow boundary -1 m (w.r.t. reference). The Chézy friction coefficient C is set to 65 m$^{1/2}$/s. At the output, a water level h_{out} equal to 1.5 m+NAP is aimed for. This water level implies a water depth (at the outflow boundary) equal to 2.5 m+NAP and hence a water level gradient equal to:

$$\frac{\mathrm{d}h}{\mathrm{d}x} = \frac{\mathrm{d}d}{\mathrm{d}x} + i_b = i_b \frac{d^3 - d_e^3}{d^3 - d_g^3} + i_b = -10^{-4} \frac{2.5^3 - 3.896767^3}{2.5^3 - 1.365915^3} - 10^{-4} = -4.330121 \cdot 10^{-4} \tag{3.7}$$

This water level *gradient* is imposed as the outflow boundary condition. To check whether this gradient implies a *unique* water level, Equation (3.4) is rewritten as:

$$d^3 = \left(d_e^3 - \frac{1}{i_b} \frac{\mathrm{d}d}{\mathrm{d}x} \cdot d_g^3 \right) \cdot \left(1 - \frac{1}{i_b} \frac{\mathrm{d}d}{\mathrm{d}x} \right)^{-1} = 15.6250 \text{ m}^3, \tag{3.8}$$

which implies $d = 2.5$ m as a unique solution and hence a water level $h_{out} = 1.5$ m+NAP.

The computational time step is set automatically being restricted by a CFL limit value equal to 0.5. Upstream, the keyword `jbasqbnddownwindhs` is set to `1` (relevant for the inflow boundary), whereas `izbndpos = 0` (relevant for the outflow boundary). For the bed friction, the option `Conveyance2D = 3` is set (i.e. an analytic expression is used for the hydraulic radius).

Results

First, recall the quantity `hu`, which represents the upstream water level (cell center) at the location of a velocity point (cell face). This quantity is used as the water depth within the computation of the bed friction at a velocity point. As a result, the water depth equals the water level at the upstream *cell center* minus the bed level at the *cell face*.

Hence, if a water depth is to be computed as part of the postprocessing, this water depth should be computed in analogy, namely as the difference between the bed level at a velocity point and the upstream water level. This value for `hu` should then be assessed against the backdrop of the semi-analytically computed water depth.

[1] The equations treat i_b by a positive value, whereas, however, this value is negative from a geometric point of view.

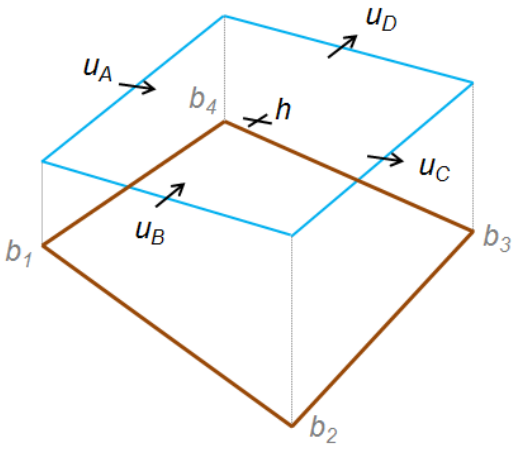


Figure 3.3: *Computational cell with surface level h, face velocities u_A, u_B, u_C and u_D and bed levels b_1, b_2, b_3 and b_4.*

The way of postprocessing the water depth can further be clarified by considering an arbitrary computational cell, as shown in Figure 3.3. Recall that the principal variables are the water level (located at the cell center) and the face-normal velocities (located at the center of the faces). The level of the bed is given in the cell corners. The value of `hu` at the location of, for instance, velocity u_C is then computed as $h-(b_2+b_3)/2$ (in case of `bedlevtyp = 3`; the options `4` and `5` take $\min(b_2, b_3)$ and $\max(b_2, b_3)$, respectively, to be subtracted from the water level). However, choosing for `Conveyance2D = 3` implies an approach according to `bedlevtyp = 4`.

Since the analytical solution is known for this flow situation, the deviations of the computed results from the analytical results can be measured exactly. As a measure, the L_2-norm of the residual Res (i.e. the difference between the analytical and the numerical solution) is taken, defined as

$$\mathrm{Res}_{L_2} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\mathrm{Res}_i^2} \tag{3.9}$$

with N the number of evaluated grid locations. Differences with the analytical solution for the five different grids are shown in Figure 3.4 through this measure.

Figure 3.4 reveals the convergence behavior of the water depth (left panel) and the water level (right picture). For the water depth, the values of the quantity `hu` is used (i.e. both at flowlinks parallel to the flow direction and flowlinks perpendicular to the flow direction). For the water level, the actual cell centered water level itself is considered, obviously.

Figure 3.4 reveals first order convergence for both the water depth and the water level. However, the absolute errors for the water level are larger than for the water depth. For the context of this difference, a closer look is taken at the values at an arbitrary location in the flow domain, provided by Figure 3.5. This figure shows the water depth at the flowlink (yellow circle), the bed level at the netnodes (red circle) and the water level at the flownodes (blue circle). As explained above, the water depth at the flowlink is the difference between the *upstream* water

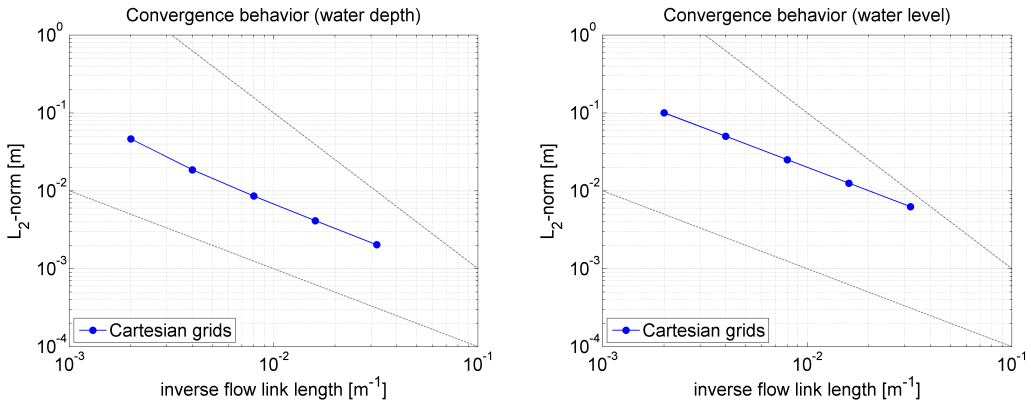


Figure 3.4: *Rate of convergence, measured by means of the L_2-norm using the analytical solution and the computational output. Left panel: convergence of the computed water depth; right panel: convergence of the computed velocity.*

level and the *lowest* bed level adjacent to the flowlink, i.e.:

$$\begin{aligned} \text{water depth} &= \text{water level} - \text{bed level} \\ &= 1.818 - \text{-}0.900 \\ &= 2.718 \text{ m} \end{aligned}$$

which is a direct result of opting for `Conveyance2D = 3`: in this approach, a spatial shift equal to $\Delta x/2$ dictates the rate of convergence.

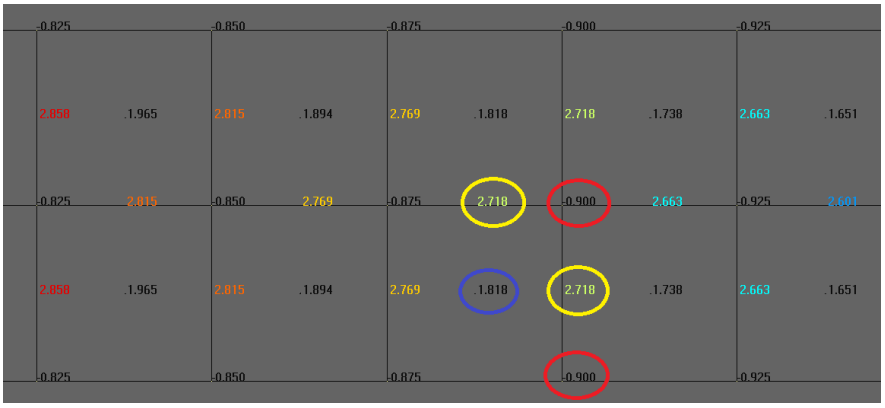


Figure 3.5: *Selected part of one of the computational grids, revealing outcomes at specific locations. The water depth at a flowlink is highlighted by a yellow circle, the water level at a flownode by a blue circle and the bed level at a netnode by a red circle.*

Conclusion

For the two-dimensional, unidirectional channel flow simulations, the following conclusions can be drawn:

1 on all the five grids (of different grade of refinement), the computational outcomes approximate the analytical outcomes at an accuracy that convergence at a rate of order 1,
2 the absolute residuals for the water level (at flownodes) are generally larger than for the water depth (at flowlinks).

Version

This test has been carried out with version dflow-fm-x64-1.1.116.36629.

3.3 Refinement study for Bélanger channel flow using triangular grids

Purpose

The purpose of this validation case is to examine the performance of D-Flow FM for a schematized channel flow simulation. For stationary flow through a river with a rectangular cross-section, the Bélanger surface profile equation can be utilized to compare the numerical solution with. This validation case focuses on the effects of the refinement of a triangular grid. The topography comprises a linearly varying bathymetry with a constant slope.

Linked claims

Claims that are related to the current test case are:

◇ claim 2.3.1.1: D-Flow FM can accurately simulate steady and unsteady hydrostatic flow on grids consisting of triangles, quadrilaterals, pentagons and hexagons

Approach

A straight channel with a rectangular cross-section is defined. Given an inflow discharge Q, a channel width B, a bottom slope i_b and a Chézy friction factor C, the distance between the free surface profile and the bed profile can be described by the Bélanger equation for d as the water depth:

$$\frac{\mathrm{d}d}{\mathrm{d}x} = i_b \frac{d^3 - d_e^3}{d^3 - d_g^3} \tag{3.10}$$

with d_e the equilibrium depth and d_g the limit depth (associated with $\mathrm{Fr} = 1$) following:

$$d_e = \left(\frac{Q^2}{B^2 g}\right)^{1/3} \qquad \text{and} \qquad d_g = \left(\frac{Q^2}{B^2 C^2 i_b}\right)^{1/3}. \tag{3.11}$$

Given a certain inflow discharge Q and a certain outflow water level condition h_{out}, the surface profile can hence numerically be estimated in the most simple way as:

$$\frac{d_i - d_{i-1}}{\Delta x} = i_b \frac{d_i^3 - d_e^3}{d_i^3 - d_g^3}, \tag{3.12}$$

having $d_i = h_{out} + i_b L$ at the outflow boundary. This, in fact *semi-analytical*, solution can be used for comparison. If the grid is refined several times, the order of accuracy of the numerical integration routines can be assessed. For this purpose, only triangular grids are considered (Cartesian grids are considered in case e02-f01-c012).

Since grid refinement is one of the main aspects of this particular testcase, the boundary conditions are specified grid independent by means of a constant Dirichlet discharge boundary condition upstream and a constant Neumann water level boundary condition downstream. As Equation (3.10) shows, a fixed water level implies a fixed water level gradient (and *vice versa* as will be shown below).

Model description

For this test case, three computational grids are generated. The outflow parts of these grids are shown in Figure 3.6. The five grids are of Cartesian type. The longitudal size L of the domain is 10 km, whereas the lateral size B of the domain is 500 m.

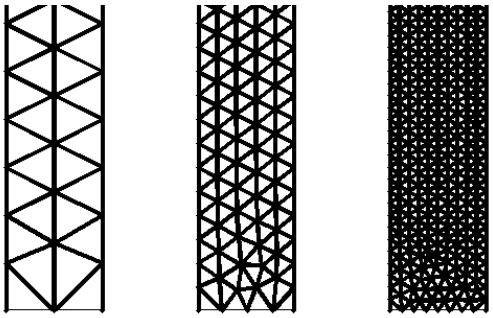


Figure 3.6: *From left to right: three triangular grids in order of refinement grade. Each refinement comprises twice as much cells in each direction. Only the outflow part of the grid is shown.*

A discharge Q equal to 2500 m^3/s is prescribed. The bottom slope i_b is prescribed[2] to be 10^{-4}. The bottom level at the inflow boundary is 0 m (w.r.t. reference) and at the outflow boundary -1 m (w.r.t. reference). The Chézy friction coefficient C is set to 65 m$^{1/2}$/s. At the output, a water level h_{out} equal to 1.5 m+NAP is aimed for. This water level implies a water depth (at the outflow boundary) equal to 2.5 m+NAP and hence a water level gradient equal to:

$$\frac{\mathrm{d}h}{\mathrm{d}x} = \frac{\mathrm{d}d}{\mathrm{d}x} + i_b = i_b \frac{d^3 - d_e^3}{d^3 - d_g^3} + i_b = -10^{-4} \frac{2.5^3 - 3.896767^3}{2.5^3 - 1.365915^3} - 10^{-4} = -4.330121 \cdot 10^{-4} \tag{3.13}$$

This water level *gradient* is imposed as the outflow boundary condition. To check whether this gradient implies a *unique* water level, Equation (3.10) is rewritten as:

$$d^3 = \left(d_e^3 - \frac{1}{i_b} \frac{\mathrm{d}d}{\mathrm{d}x} \cdot d_g^3 \right) \cdot \left(1 - \frac{1}{i_b} \frac{\mathrm{d}d}{\mathrm{d}x} \right)^{-1} = 15.6250 \text{ m}^3, \tag{3.14}$$

which implies $d = 2.5$ m as a unique solution and hence a water level $h_{out} = 1.5$ m+NAP.

The computational time step is set automatically being restricted by a CFL limit value equal to 0.5. Upstream, the keyword `jbasqbnddownwindhs` is set to `1` (relevant for the inflow boundary), whereas `izbndpos = 0` (relevant for the outflow boundary). For the bed friction, the option `Conveyance2D = 3` is set (i.e. an analytic expression is used for the hydraulic radius).

Results

First, recall the quantity `hu`, which represents the upstream water level (cell center) at the location of a velocity point (cell face). This quantity is used as the water depth within the computation of the bed friction at a velocity point. As a result, the water depth equals the water level at the upstream *cell center* minus the bed level at the *cell face*.

Hence, if a water depth is to be computed as part of the postprocessing, this water depth should be computed in analogy, namely as the difference between the bed level at a velocity point and the upstream water level. This value for `hu` should then be assessed against the backdrop of the semi-analytically computed water depth.

[2] The equations treat i_b by a positive value, whereas, however, this value is negative from a geometric point of view.

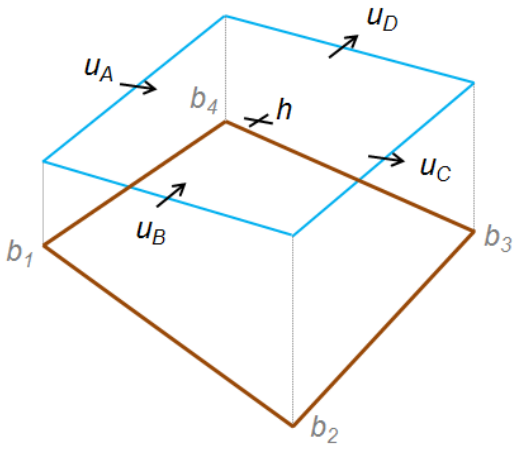


Figure 3.7: *Computational cell with surface level h, face velocities u_A, u_B, u_C and u_D and bed levels b_1, b_2, b_3 and b_4.*

The way of postprocessing the water depth can further be clarified by considering an arbitrary computational cell, as shown in Figure 3.7. Recall that the principle variables are the water level (located at the cell center) and the face-normal velocities (located at the center of the faces). The level of the bed is given in the cell corners. The value of `hu` at the location of, for instance, velocity u_C is then computed as $h-(b_2+b_3)/2$ (in case of `bedlevtyp = 3`; the options `4` and `5` take $\min(b_2, b_3)$ and $\max(b_2, b_3)$, respectively, to be subtracted from the water level). However, choosing for `Conveyance2D = 3` implies an approach according to `bedlevtyp = 4`.

Since the analytical solution is known for this flow situation, the deviations of the computed results from the analytical results can be measured exactly. As a measure, the L_2-norm of the residual Res (i.e. the difference between the analytical and the numerical solution) is taken, defined as

$$\mathrm{Res}_{L_2} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\mathrm{Res}_i^2} \tag{3.15}$$

with N the number of evaluated grid locations. Differences with the analytical solution for the three different grids are shown in Figure 3.8 through this measure.

Figure 3.8 reveals the convergence behavior of the water depth (left panel) and the water level (right picture). For the water depth, the values of the quantity `hu` is used (i.e. both at flowlinks parallel to the flow direction and flowlinks perpendicular to the flow direction). For the water level, the actual cell centered water level itself is considered, obviously.

Figure 3.8 reveals first order convergence for both the water depth and the water level. However, the absolute errors for the water level are larger than for the water depth. For the context of this difference, a closer look is taken at the values at an arbitrary location in the flow domain, provided by Figure 3.9. This figure shows the water depth at the flowlink (yellow circle), the bed level at the netnodes (red circle) and the water level at the flownodes (blue circle). As explained above, the water depth at the flowlink is the difference between the *upstream* water

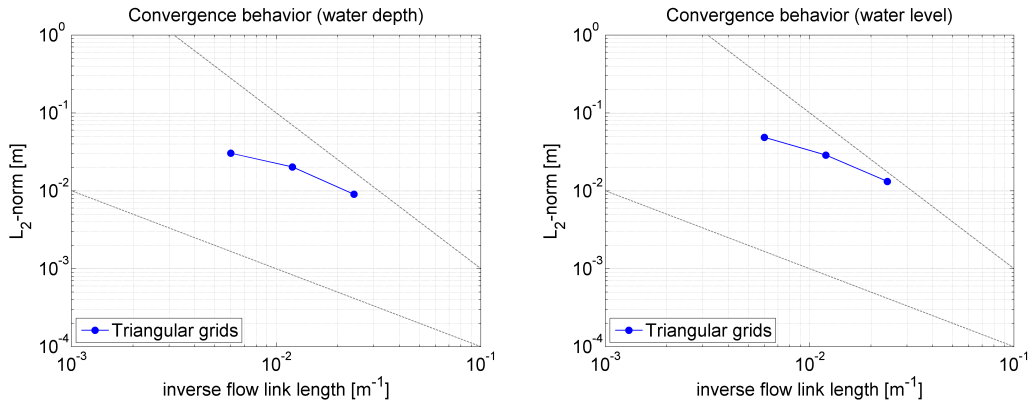


Figure 3.8: *Rate of convergence, measured by means of the L_2-norm using the analytical solution and the computational output. Left panel: convergence of the computed water depth; right panel: convergence of the computed velocity.*

level and the *lowest* bed level adjacent to the flowlink, i.e.:

$$\begin{aligned} \text{water depth} &= \text{water level} - \text{bed level} \\ &= 1.865 - \text{-}0.888 \\ &= 2.753 \text{ m} \end{aligned}$$

which is a direct result of opting for `Conveyance2D = 3`: in this approach, a spatial shift equal to $\Delta x/2$ dictates the rate of convergence.

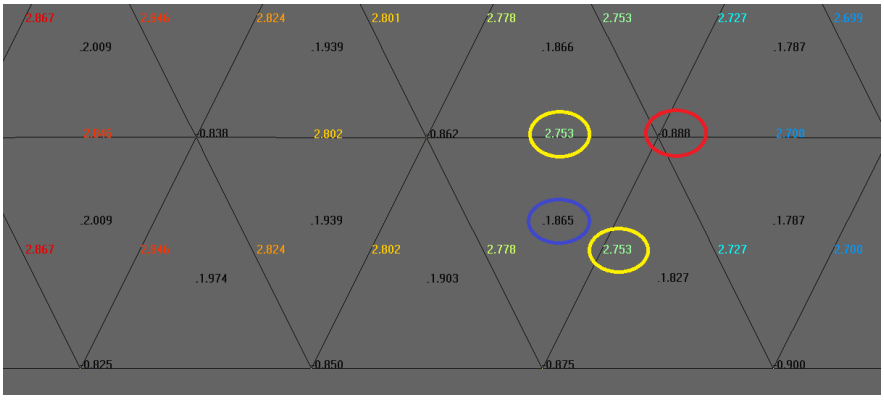


Figure 3.9: *Selected part of one of the computational grids, revealing outcomes at specific locations. The water depth at a flowlink is highlighted by a yellow circle, the water level at a flownode by a blue circle and the bed level at a netnode by a red circle.*

Conclusion

For the two-dimensional, stationary, homogeneous and uniform channel flow simulations, the following conclusions can be drawn:

1 on all the three grids (of different grade of refinement), the computational outcomes approximate the analytical outcomes very closely, both regarding the water levels and the velocities,
2 the differences between the computational and analytical outcomes converge at rate 0 with decreasing cell size.

Version

This test has been carried out with version dflow-fm-x64-1.1.116.36629.

3.4 One-dimensional dambreak over a wet bed

Purpose

The purpose of this validation study is to test the D-Flow FM modelling accuracy for flooding over both a wet bed, resulting from a dam break. The D-Flow FM model results are compared with the analytical solution. In this validation study, multiple grids are considered.

Linked claims

Claims that are related to the current test case are:

- ◇ 2.3.1.1
- ◇ 2.3.1.1: D-Flow FM can accurately simulate steady and unsteady hydrostatic flow on grids consisting of triangles, quadrilaterals, pentagons and hexagons
- ◇ claim 2.3.2.1: D-Flow FM can be used for an accurate prediction of flows resulting from dam breaks

Approach

For one-dimensional dambreak flow over a wet bed, analytical expressions are available to compare the numerical solution with. The initial state is a stepwise water level with $h(x) = h_l$ for $0 \leq x \leq x_0$ and $h(x) = h_r$ for $x_0 < x < L$, with the discontuinity at $x = x_0$ and a computational domain from $x = 0$ to $x = L$.

The analytical solution, for a wet bed case, can be prescribed for four regions, marked by three relevant locations, namely at $x = x_A$, $x = x_B$ and x_C, with:

$$\begin{aligned} x_A &= x_0 - t\sqrt{gh_l} \\ x_B &= x_0 + 2t\sqrt{gh_l} - 3tc_m \\ x_C &= x_0 + t\frac{2c_m^2\left(\sqrt{gh_l} - c_m\right)}{c_m^2 - gh_r} \end{aligned}$$

The analytical solution for the water level along the three parts reads:

$$h(x,t) = \begin{cases} h_l, & \text{if} \quad x < x_A(t), \\ \dfrac{4}{9g}\left(\sqrt{gh_l} - \dfrac{x - x_0}{2t}\right)^2, & \text{if} \quad x_A(t) \leq x \leq x_B(t), \\ \dfrac{c_m^2}{g}, & \text{if} \quad x_B(t) \leq x \leq x_C(t), \\ h_r, & \text{if} \quad x > x_C(t). \end{cases} \tag{3.16}$$

The analytical solution for the velocity along the three parts reads:

$$u(x,t) = \begin{cases} 0 \text{ m/s}, & \text{if} \quad x < x_A(t), \\ \dfrac{2}{3}\left(\dfrac{x - x_0}{t} + \sqrt{gh_l}\right), & \text{if} \quad x_A(t) \leq x \leq x_B(t), \\ 2\left(\sqrt{gh_l} - c_m\right), & \text{if} \quad x_B(t) \leq x \leq x_C(t), \\ 0 \text{ m/s}, & \text{if} \quad x > x_C(t). \end{cases} \tag{3.17}$$

Compared to the dry bed case, an additional variable is part of the solution, namely $c_m = \sqrt{gh_m}$. This variable c_m can numerically be derived, through Newton iteration, from the

expression:

$$-8gh_r c_m^2 \left(\sqrt{gh_l} - c_m\right)^2 + \left(c_m^2 - gh_r\right)^2 \left(c_m^2 + gh_r\right) = 0. \tag{3.18}$$

Model description

For this testcase, several grids are generated (see Figure 3.10). The computational settings and parameters are:

- ◇ domain length $L = 60$ km,
- ◇ time step $\Delta t = 1.0$ seconds,
- ◇ $h_l = 2$ m and $h_r = 0.1$ m,
- ◇ no horizontal viscosity and no bottom friction.

The computational grids developed for this case are shown in Figure 3.10. Grids 1, 2 and 3 contain square cells with increasing refinement, grid 4 contains triangular cells, grid 5 is an equidistant with elongated cells, grid 6 is a variant of grid 5 with a smoothly varying cell size, grid 7 has refinements in two directions.

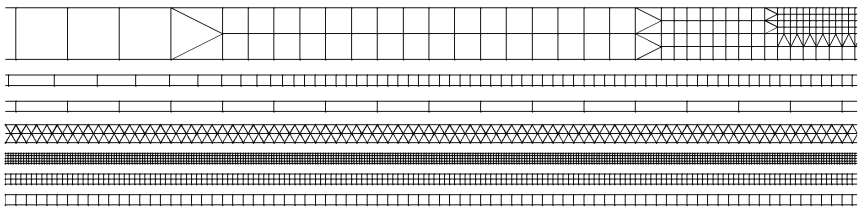


Figure 3.10: *Seven grids for the one-dimensional dambreak case. From bottom to top (numbered from 1 to 7): grids 1, 2 and 3 contain square cells with increasing refinement, grid 4 contains triangular cells, grid 5 is an equidistant with elongated cells, grid 6 is a variant of grid 5 with a smoothly varying cell size, grid 7 has refinements in two directions.*

Results

The results of the one-dimensional dambreak are considered after 3600 seconds, when the waves have considerably propagated without having been reflected by the boundaries. Three issues are highlighted:

1. the effects of refinement in two directions are discussed (grids 1, 2 and 3),
2. the effects of refinement in one direction are considerend (grids 5 and 6),
3. the effects of the irregularity of the grid are discussed (grids 4 and 7).

Considering issue 1, it turned out that the three squared cell grids show decreasing deviations from the exact solution with decreasing cell size, as is to be expected. Since the mutual differences are relatively small, only the result for the finest grid are shown: see Figure 3.11.

Considering issue 2, it turned out that elongation of the cells considerably detoriates the result. The computation on the grid with the equidistantly distributed cells (see Figure 3.12) slightly underestimates the velocities, but reproduces the shock location relatively well, as is also the case for the grid with smoothly varying cell size (see Figure 3.13).

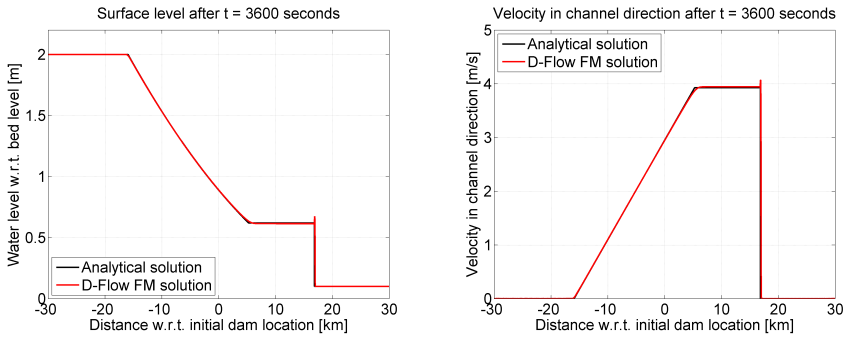


Figure 3.11: *Results for the most refined Cartesian grid (grid 3). Left panel: the water level (m); right panel: the velocity (m/s).*

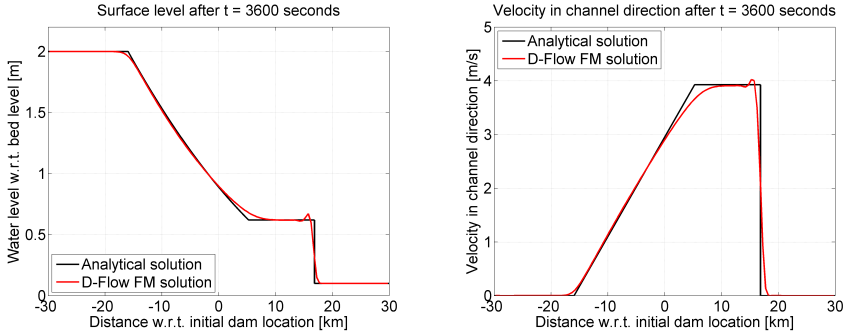


Figure 3.12: *Results for the grid containing equidistant, elongated rectangular cells (grid 5). Left panel: the water level (m); right panel: the velocity (m/s).*

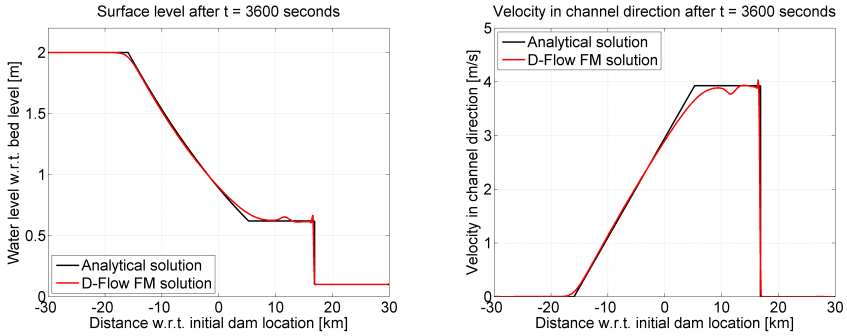


Figure 3.13: *Results for the grid containing rectangular cells with smoothly varying size (grid 6). Left panel: the water level (m); right panel: the velocity (m/s).*

Considering issue 3, it can be seen that the grid with triangular cells (see Figure 3.14) performs comparable with the squared cells grid. The grid with refinement in two directions (see Figure 3.15) captures the shock locations less accurately (pronounced wiggles).

Conclusion

For a one-dimensional dambreak case over a wet bed, D-Flow FM performs fairly well regarding the prediction of the water level and the velocity. The best result are obtained using square cell grids; deviations from the exact solutions become more pronounced as soon as the grid is more irregular.

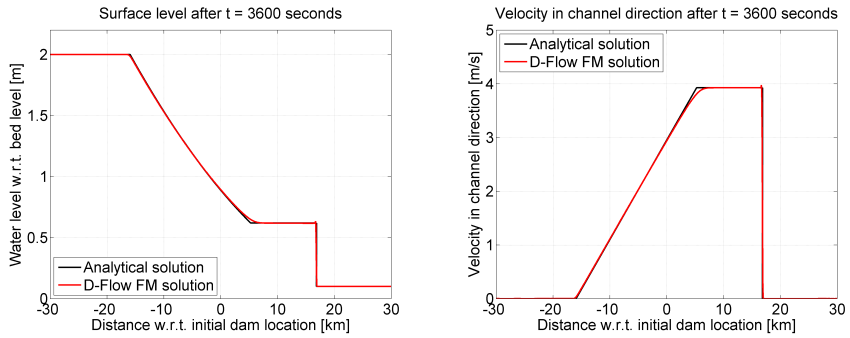


Figure 3.14: *Results for the triangular grid (grid 4). Left panel: the water level (m); right panel: the velocity (m/s).*

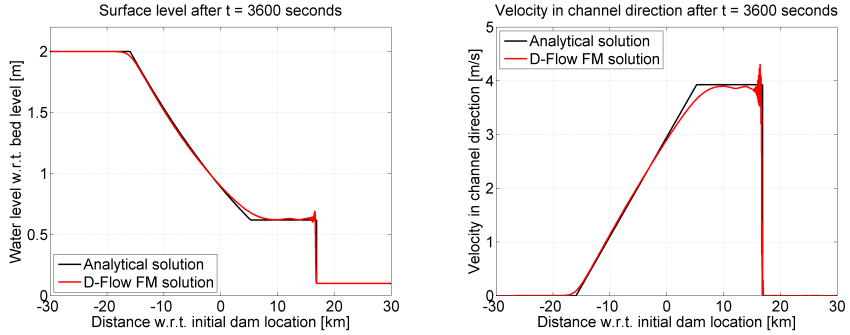


Figure 3.15: *Results for the grid with refinement in two directions (grid 7). Left panel: the water level (m); right panel: the velocity (m/s).*

Version

This test has been carried out with version dflow-fm-x64-1.1.116.36629.

3.5 Two-dimensional dambreak over a wet bed

Purpose

This validation study investigates the flow resulting from a dam break including the interaction of reflecting bores in a 2D model. The case investigated is based on the experiments by Stelling & Duinmeijer from 2001. Tests are conducted on Cartesian as well as on triangular grids.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.1.1: D-Flow FM can accurately simulate steady and unsteady hydrostatic flow on grids consisting of triangles, quadrilaterals, pentagons and hexagons
- ◇ claim 2.3.2.1: D-Flow FM can be used for an accurate prediction of flows resulting from dam breaks

Approach

For this validation study a model is applied that is described in Stelling & Duinmeijer (2001). The geometry of the experimental setup is shown in Figure 3.16. The experiments are performed in a closed domain.

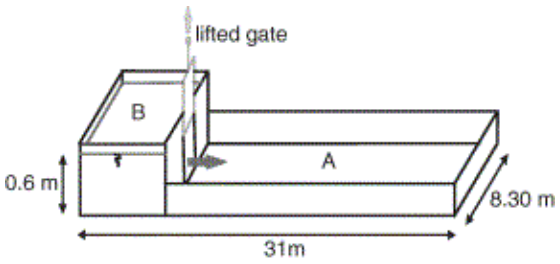


Figure 3.16: *Experimental setup of the Stelling & Duinmeijer experiment.*

Two experiments are considered by Stelling & Duinmeijer:

- ◇ a dam break with an initially wet region; and
- ◇ a dam break with an initially dry region.

Through the geometry of the experimental setup, reflecting bores interact with each other, resulting into quite a challanging validation case compared to canonical one-dimensional dam break problems. In the present validation case, the wet bed case is considered.

The experimental set-up consists of two reservoirs, A and B, separated by a wall with a gate of width 0.4 m that can be lifted. Reservoir B is initially filled with water of height 0.6 m. Two experiments are performed. The first experiment contains a thin layer of water with a depth of 0.05 m downstream of the gate and in the second experiment, the reservoir A is initially dry. The gate is then lifted with a speed of 0.16 m/s and the subsequent flooding in reservoir A is studied.

Model description

For this testcase, a Cartesian grid and a triangular grid are adopted. The wall that divides the two basins is represented through thin dams for the Cartesian case. For the triangular case,

this wall is represented directly in the grid domain boundaries. The computational settings and parameters are:

- ◇ domain length $L = 31$ m,
- ◇ domain width $D = 8.3$ m,
- ◇ the Cartesian grid has sizes $\Delta x = \Delta y = 0.1$ m and hence 310 $\times$ 73 cells,
- ◇ the triangular grid consists of triangular cells with a typical edge length of 0.1 m.
- ◇ time step $\Delta t = 0.006$ seconds,
- ◇ Manning bottom roughness = 0.012 $m^{-1/3}$/s,
- ◇ the horizontal eddy viscosity $\nu_h = 5.0 \cdot 10^{-4}$ m^2/s.

Results

An artist's impression of the result of the D-Flow FM computation on the Cartesian grid is shown in Figure 3.17. In this figure, the bore has already been reflected from the boundary at $x = L = 31$ m, turning in a backwards propagating front.

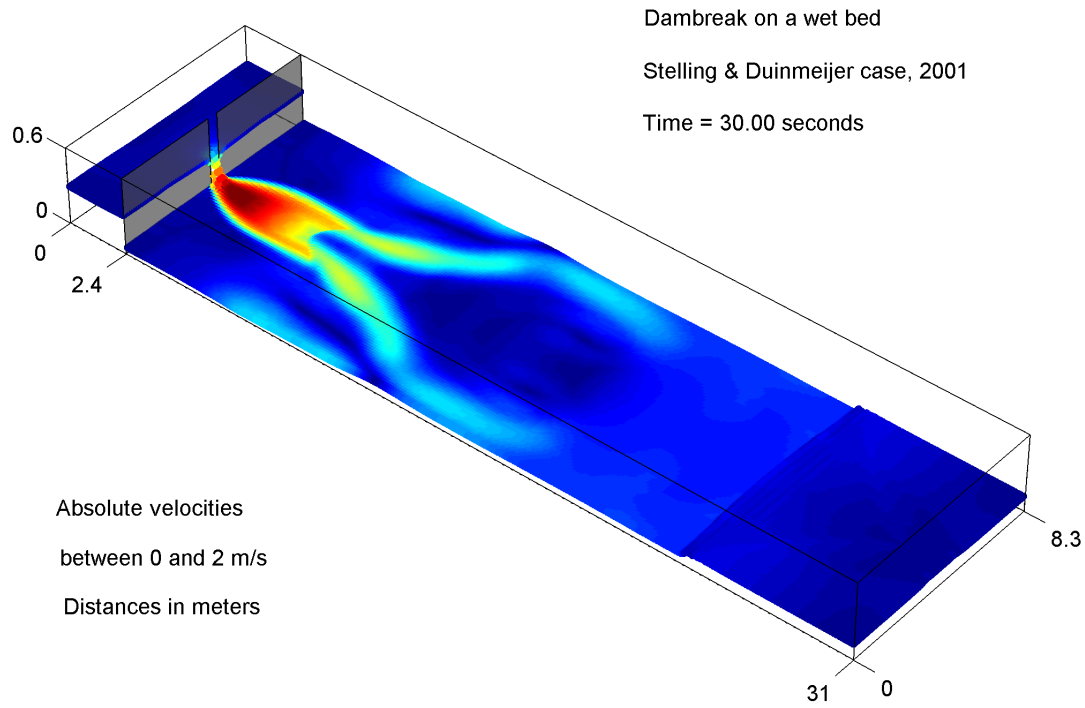
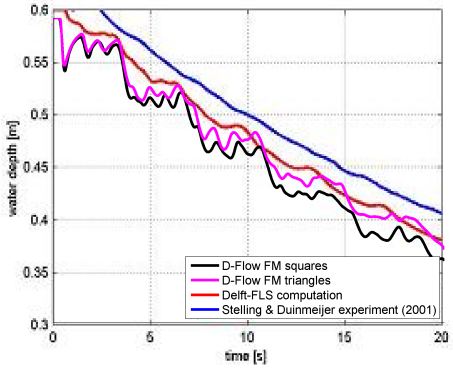


Figure 3.17: *Computational result after 30 seconds, showing the water level and the absolute velocities.*

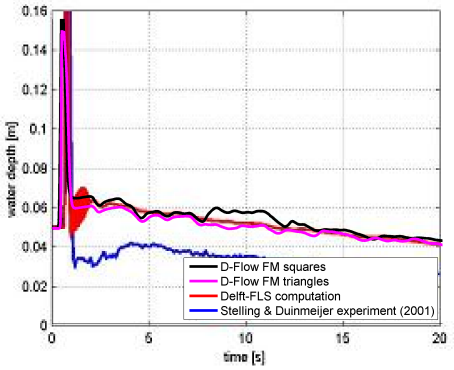
Timeseries for the water level are available at multiple locations along the centerline of the flume. Six of these locations are highlighted in Figure 3.18. With reference to the gate location, these six gauges are located at -1 m, +1 m, 6 m, 9 m, 13 m, 17 m and 21 m. The results for the simulation on the Cartesian grid and triangular grid are shown in black and magenta, respectively.

Figure 3.18 shows that in the D-Flow FM simulation, the reservoir empties quicker compared to the experiment. At later stages, the D-Flow FM results are comparable to the Delft-FLS results. Variants of the D-Flow FM computation can now be undertaken to study the influences of the gridtype, the gridsize, the value of the bottom friction and the horizontal eddy-viscosity.

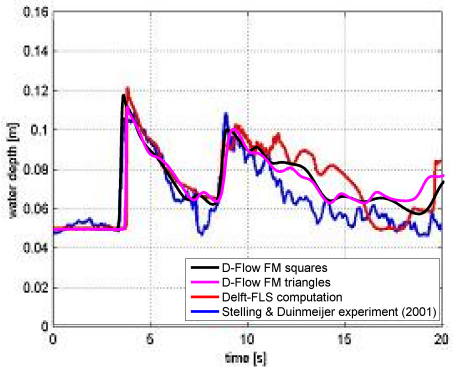
Mutual differences between the results obtained on the Cartesian grid and the triangular grid



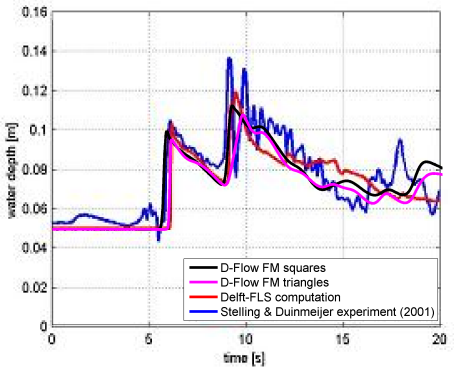
(a) *Results at 1 m upstream of the gate.*



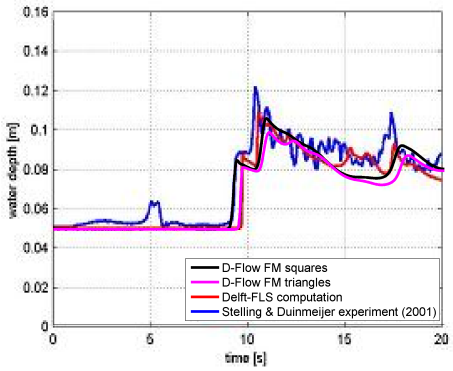
(b) *Results at 1 m downstream of the gate.*



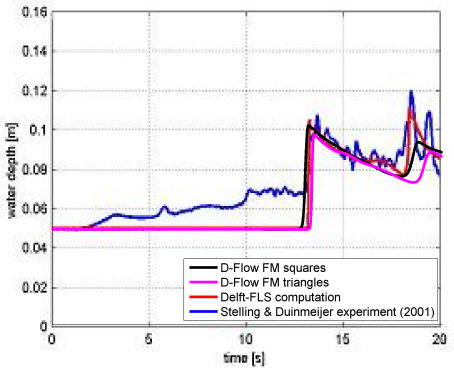
(c) *Results at 6 m downstream of the gate.*



(d) *Results at 9 m downstream of the gate.*



(e) *Results at 13 m downstream of the gate.*



(f) *Results at 17 m downstream of the gate.*

Figure 3.18: *Results for the two-dimensional dambreak on a wet bed. In lexicographic ordering: water depth in meters at -1 m, +1 m, 6 m, 9 m, 13 m and 17 m downstream of the gate.*

are difficultly interpretable. On the Cartesian grid, the reservoir seems to empty faster that in the triangular case. Moreover, the wave front due to the dam break appears to move somewhat faster in the Cartesian case compared to the triangular case. However, strict conclusions that address the actual performance difference between the two simulations cannot be drawn unequivocally.

Conclusion

D-Flow FM provides fairly good results for the two-dimensional dambreak on a wet bed. The D-Flow FM results are close to the Delft-FLS results, but should further be explored to investigate what the main causes are regarding the differences with the experiment.

Version

This test has been carried out with version dflow-fm-x64-1.1.116.36629.

3.6 Advection of a passive tracer

Purpose

D-Flow FM utilizes transport routines that both serve the computation of salt transport and passive tracers. In the present validation case, the advection of a passive tracer is considered.

Linked claims

Claims that are related to the current test case are:

◇ claim 2.3.7.1: D-Flow FM can accurately simulate the transport of dissolved material

Approach

A quasi one-dimensional flow model is constructed with flow under steady state conditions. For this flow model, three grids are developed: one coarse Cartesian grid, one refined Cartesian grid and a triangular grid. The goal is to achieve a flow state that is steady at a constant water level and a constant velocity, in the absence of friction. Once this state is achieved, a tracer is released at the inflow boundary during a finite period. The tracer substance thus travels along the domain, leaving the domain at the outflow boundary. In the absence of friction, diffusion and Earth rotation, the problem can either be solved analytically.

Model description

The flow model has sizes $L \times B$ (length times width) equal to 300 m $\times$ 1 m. Three grids are considered, from which a part is shown in Figure 3.19.

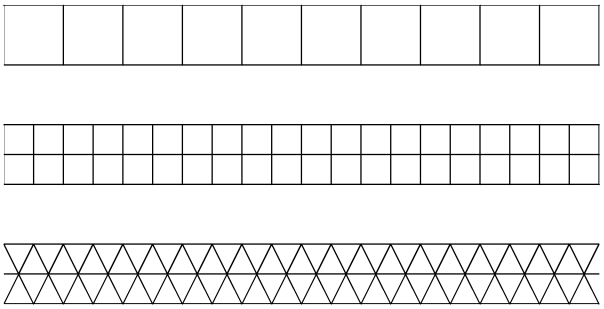


Figure 3.19: *From top to bottom: parts of the developed grids, namely one coarse Cartesian grid, one refined Cartesian grid and a triangular grid.*

The bed level is set to 0 with respect to the reference level. At the inflow boundary, a water level equal to 1 m (w.r.t. reference) is prescribed, as well as at the outflow boundary. At the inflow boundary, a fixed normal velocity equal to 0.1 m/s is imposed. Friction, diffusion and Earth rotation are switched off. The expected (trivial) solution is, hence, a steady flow state at a water level equal to 1 m w.r.t. reference and a constance flow velocity of 0.1 m/s.

The model is given a cold start, hence demanding a certain spin-up time. The model is given a spin-up time equal to 600 minutes which is proven to be a sufficiently long period. After 600 minutes, a passive tracer is released at the inflow boundary during a period of 10 minutes.

Results

The analytical solution for this advection problem is straightforward: just a block travelling throughout the domain at a fixed velocity. The analytical solution is shown in black in Figure 3.20 for the situation after 2500 seconds after the release of the tracer at the inflow boundary. In addition, Figure 3.20 shows the results

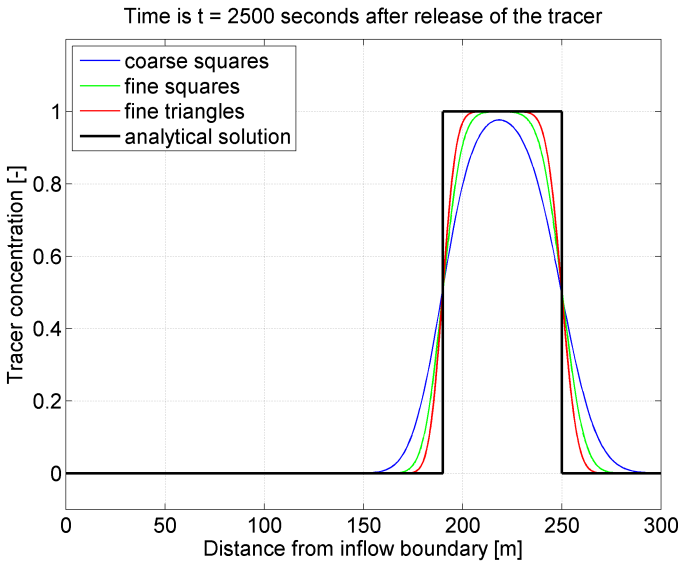


Figure 3.20: *Numerical solution after 2500 seconds after release of the passive tracer at the inflow boundary.*

The results show fair agreement of the numerical solutions with the analytical one. Obviously, numerical diffusion plays a clear role; a role that gains influence with decreasing cell size.

Conclusion

D-Flow FM appears to be able to compute the transport of a passive scalar. Since the calculation of salt transport utilizes the same code routines, a similar conclusion can be drawn for the computation of salinity.

Version

This test has been carried out with version dflow-fm-x64-1.1.137.40169.

4 Flooding and drying

4.1 One-dimensional planar free surface oscillations

Purpose

Flooding and drying are key phenomena from the hydraulic engineering practice. This test investigates the performance of D-Flow FM for a case for which an exact solution is available.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.1.1: D-Flow FM can accurately simulate steady and unsteady hydrostatic flow on grids consisting of triangles, quadrilaterals, pentagons and hexagons
- ◇ claim 2.3.2.3: D-Flow FM can accurately simulate the propagation of short waves
- ◇ claim 2.3.2.4: D-Flow FM can accurately simulate drying and flooding of tidal areas

Approach

For one-dimensional frictionless flow in a parabolic shaped basin, an exact solution is available for the evolution of the water level and the velocity, in case the initial surface elevation is a linearly increasing across the bathymetry. The analytical solution reads, with x the coordinate across the bathymetry:

$$\begin{aligned} h(x,t) &= \frac{\eta h_0}{r_0}\left(2x\cos\omega t - \eta r_0 \cos^2\omega t\right) \\ u(x,t) &= -\eta r_0 \omega \sin\omega t, \end{aligned}$$

with frequency $\omega = \sqrt{2gh_0}/r_0$ and period $2\pi/\omega$. This flow is simulated in D-Flow FM and compared with the analytical solution. Basically, the only input parameters are h_0 as a vertical measure [m], r_0 as a horizontal measure [m] and η a parameter [-]. The shape of the bathymetry is prescribed as:

$$z(x) = -h_0\left(1 - \frac{x^2}{r_0^2}\right).$$

Model description

The model is set up as a one-dimensional case. Three grids are built: (1) a one-dimensional network (i.e. no cells, just linear pieces), (2) a one-dimensional grid (i.e. a row of cells) and (3) a two-dimensional grid, including squares of several sizes, connected by means of triangles. The three grids are shown in Figure 4.1.

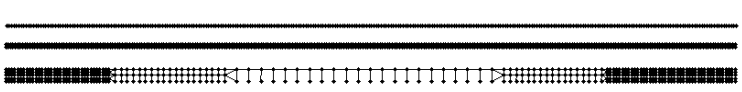


Figure 4.1: *The three computational grids for the flooding and drying testcase.*

The input settings h_0, r_0 and η are 10 m, 120 m and 0.23 respectively. Friction and horizontal viscosity are turned off. The runtime comprises 600 seconds. Given the above parameters, the period $2\pi/\omega$ yields 53.8284 seconds. For the parameter `chkadvd`, the value of 0 m is set.

Results

The water level after 260 seconds (i.e. 4.83 periods) is shown in Figure 4.2 against the backdrop of the bathymetry as well as the analytical solution. The figure visualizes deviations between the computed solutions and the exact solution. The deviation is the most pronounced for the one-dimensional network. The one-dimensional grid and the two-dimensional grid show negligible differences mutually.

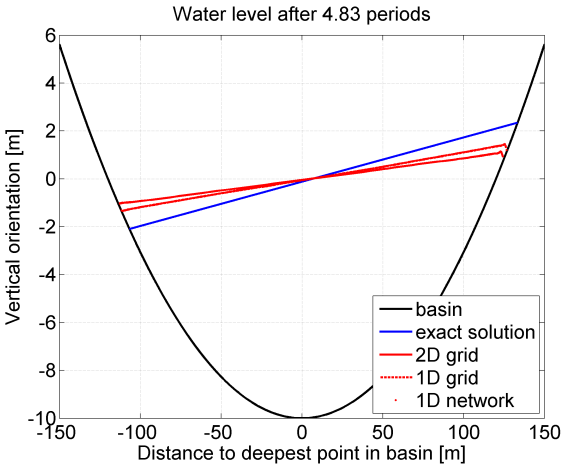


Figure 4.2: *Computational results obtained on the one-dimensional network, the one-dimensional grid and the two-dimensional grid, drawn against the backdrop of the bathymetry.*

Figure 4.2 is provided a context through Figure 4.3. In Figure 4.3, the water level and the velocity are shown as varying in time. The location chosen for the comparison is 61.5 m right of the deepest point in the basin.

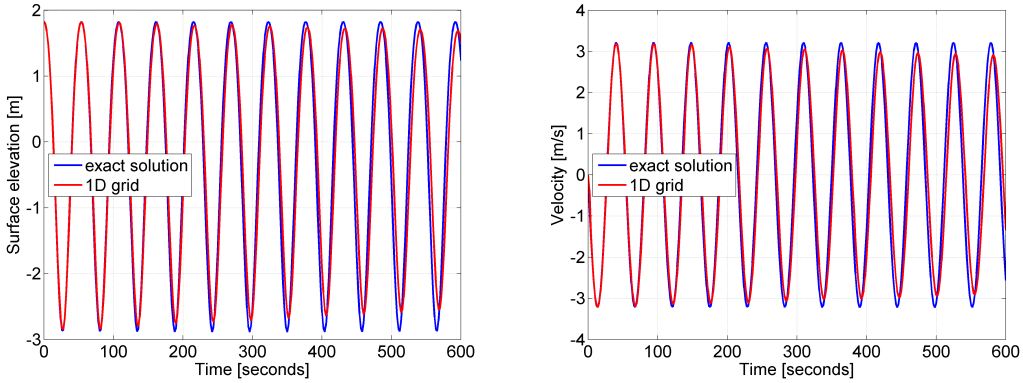


Figure 4.3: *Temporal evolution of the water level (left panel) and the velocity (right panel) following from the computation on the one-dimensional grid and from the exact solution.*

Figure 4.3 shows that the amplitude of the computed signals decreases in time. Moreover, a phase lag appears with increasing magnitude in the course of time. Besides these general computational scheme related issues, the flooding and drying scheme appears to work properly in the sense that dry areas can flood and dry at all.

As a variation on the same theme, grids are generated in addition with either square cells or triangular cells. In case of the use of triangular cells, it has been experienced that the choice on either `conveyance2D = 3` or `conveyance2D = -1` make quite a difference. In case of `conveyance2D = 3`, extreme velocities come into existence, whereas these stay absent in case of `conveyance2D = -1`.

As an illustration, Figure 4.4 which displays the cell-center velocities in x-direction at 20 seconds after the start of the simulation. In these 20 seconds, some drying has taken place. The area over which this drying had taken place is marked by the arrow in Figure 4.4. During these 20 seconds, high velocities have appeared on the interface dry/wet in case of `conveyance2D = -1`. However, extremely high velocities have appeared in the area that fell dry during these 20 seconds, in case of `conveyance2D = 3`.

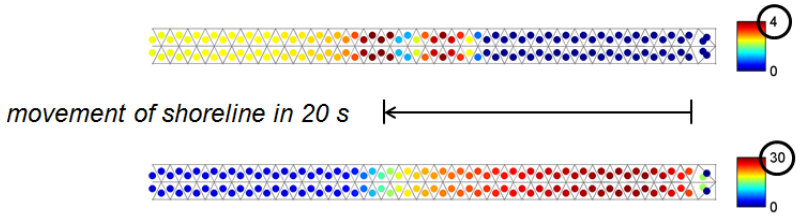


Figure 4.4: *Cell-center x-velocities at 20 seconds after the start of the simulation. For the top panel, `conveyance2D = -1` is used, for the bottom panel `conveyance2D = 3` is used. Notice the difference in color bar values (given in m/s).*

The extremely high velocities have only appeared on triangular grids. Although the simulation have been carried out without friction, the extremely high velocities did not vanish after applying some amount of friction.

Conclusion

D-Flow FM is able to computationally deal with flooding and drying, although a certain amount of diffusion takes place causing a phase lag with respect to the analytical solution. It is found that high velocities are found on the interface dry/wet. Moreover, it is found that the combination `conveyance2D = 3`, triangular grid and flooding/drying is a risky one.

Version

This test has been carried out with version dflow-fm-x64-1.1.148.41435.

4.2 Two-dimensional radial free surface oscillations

Purpose

In the test case *One-dimensional planar free surface oscillations*, the key phenomena of flooding and drying are investigated for a one-dimensional model. In the present test case, a two-dimensional case is investigated. In the two-dimensional case, free surface oscillations will be considered for a radially symmetric case with a curved free surface. An analytical solution is available to compare the numerical results with (William Thacker, *Some exact solutions to the nonlinear shallow-water wave equations*, J. Fluid Mech. (1981), vol. 107, pp. 499-608).

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.1.1: D-Flow FM can accurately simulate steady and unsteady hydrostatic flow on grids consisting of triangles, quadrilaterals, pentagons and hexagons
- ◇ claim 2.3.2.3: D-Flow FM can accurately simulate the propagation of short waves
- ◇ claim 2.3.2.4: D-Flow FM can accurately simulate drying and flooding of tidal areas

Approach

A test model setup is set up to match with the assumptions associated with the analytical solution. The setup yields a paraboloid topography, prescribed as:

$$z(r) = -D_0 \left(1 - \frac{r^2}{a^2}\right)$$

with $-D_0$ the lowest bed level of the geometry and a a measure for the steepness of the bed. Thacker has figured out multiple analytical solutions for two-dimensional cases in the absence of bed friction. One of them comprises the description of oscillations for which the surface is curved. Three grids are generated to represent the paraboloid on: a grid containing hexagon cells, a grid containing square cells and a grid containing triangular cells (shown in Figure 4.5).

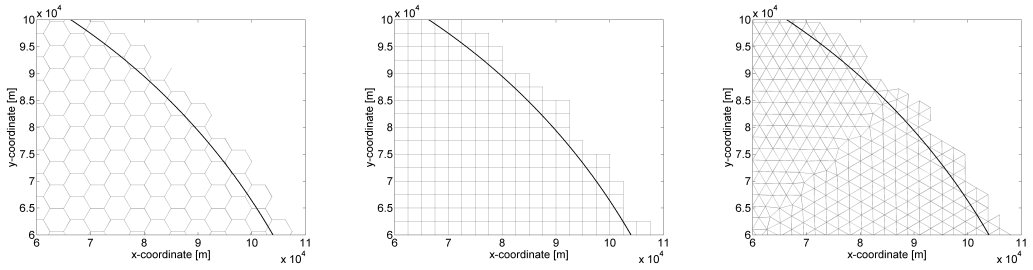


Figure 4.5: *Computational grids for the testcase (only a detail is shown). From left to right: grid with hexagon cells, square cells and triangular cells. The curved black line marks the circular area of interest (radius $R = 120$ km). Notice that the grids become finer from left to right.*

The analytical solution comprises a curved free surface and an initial zero-velocity field. The solution, derived by Thacker, describes the water level and velocity magnitude as a function of the radius r and the time t. Although acceleration due to Coriolis' forces are present in the solution, the Coriolis' effect is disregarded in the present test case. Without the Coriolis'

effect, the solution reads:

$$
\begin{aligned}
h(r,t) &= D_0 \left\{ \frac{\sqrt{1-A^2}}{1-A\cos\omega t} - 1 - \frac{r^2}{L^2}\left[\frac{1-A^2}{(1-A\cos\omega t)^2} - 1 \right] \right\} \\
u(r,t) &= \frac{1}{2} r\omega \left| \frac{A\sin\omega t}{1-A\cos\omega t} \right|
\end{aligned}
$$

for the water level h and the velocity magnitude u in which:

$$
\begin{aligned}
A &= \frac{(D_0+\eta)^2 - D_0^2}{(D_0+\eta)^2 + D_0^2} \\
\omega^2 &= \frac{8gD_0}{L^2}
\end{aligned}
$$

in which $L = a = 102$ km (geometry of the paraboloid), $D_0 = 10$ m (distance from deepest point to the reference level) and $\eta = 2$ m the largest positive surface elevation (w.r.t. the reference level). The parameter $g = 9.81$ m/s^2 represents the gravitation acceleration.

Model description

The used grids are (partially) shown in Figure 4.5. Friction and diffusion are switched off. As already mentioned, the following parameter values are used: $L = a = 102$ km, $D_0 = 10$ m and $\eta = 2$ m. Hence, via ω, one period $T = 2\pi/\omega = 22877$ s ≈ 6.35 hours. The run time yields two periodic cycles. Advection scheme `33` is used. The parameter `chkadvd = 0`.

Results

The final result for the grid containing hexagons is provided in Figure 4.6.

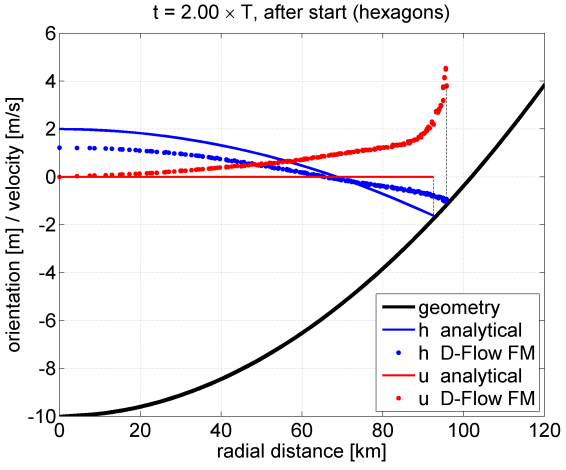


Figure 4.6: *Computational results for the testcase on the hexagonal grid. The values on the vertical axis represent water levels [m w.r.t. reference] related to the data represented in blue and represent velocities [m/s] related to the data represented in red.*

This figure shows the analytical and numerical solutions for h (in blue) and u (in red) at $t = 2T$. The computational results are projected onto the one axis of symmetry. The point at which the blue graphs and black parabola intersect can be perceived as the shoreline. The difference between the locations of the shoreline from the analytical and numerical solution is marked by two vertical black dashed lines. Analagous figures are shown in Figure 4.7 for all

the three grids (varying from top to bottom) and two points in time: $t = T$ (left panels) and $t = 2T$ (right panels).

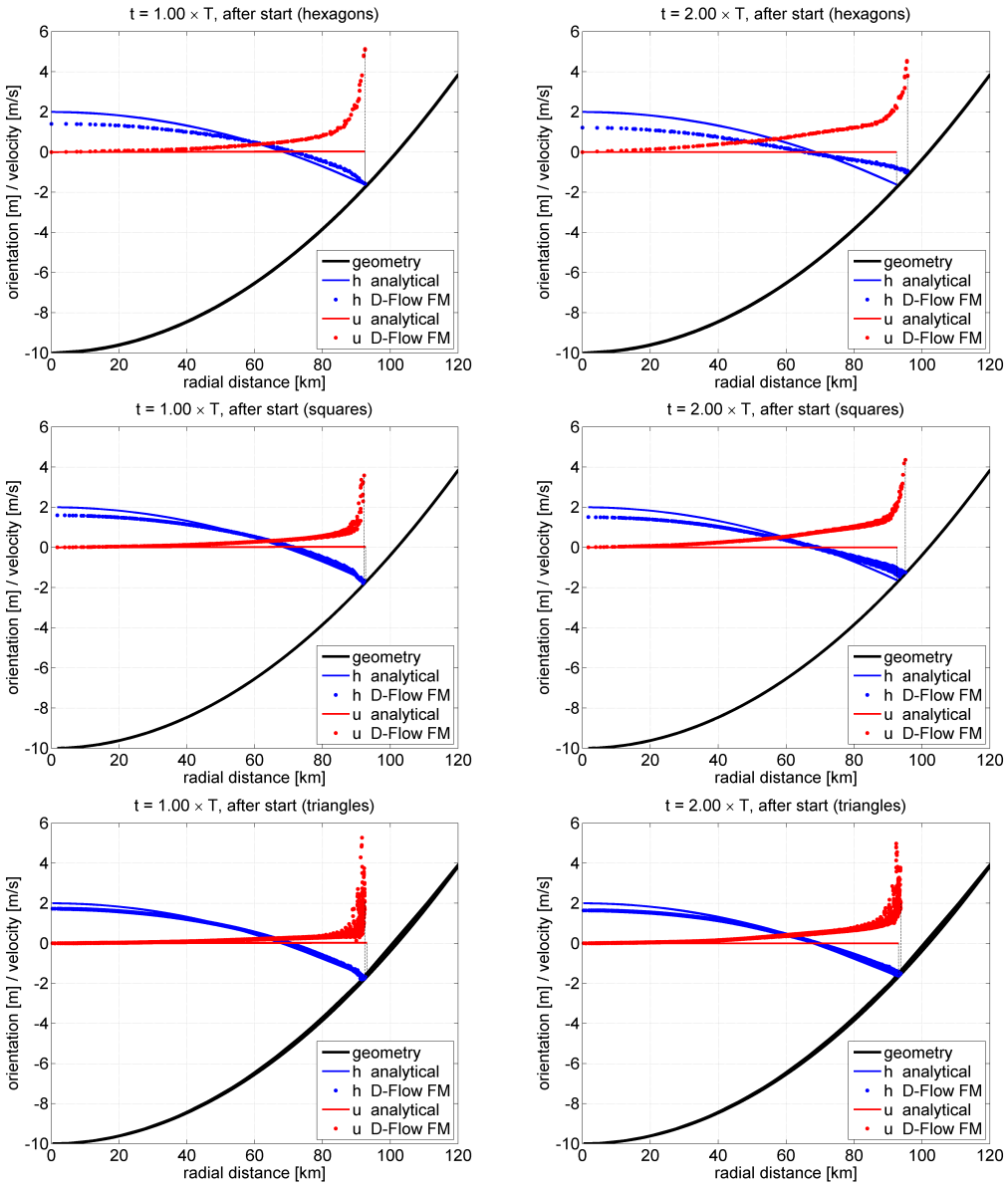


Figure 4.7: *Computational results for the testcase on the hexagonal grid (upper panels), square grid (center panels) and triangular grid (bottom panels) after one period cycle (left panels) and after two period cycles (right panels).*

It is relevant to notice that the three grids, shown in Figure 4.5, contain cells of different size. The grid containing hexagons is the coarsest, the triangular grid is the finest. This different grade of refinement is also reflected in the numerical results: the top panels in Figure 4.7 reveal the largest deviations from the analytical solution, whereas the bottom panels show the smallest deviations from the analytical solution.

A remarkable aspect of the output results is the presence of relatively high velocities near the shoreline. These high velocities appear particularly during drying. For the triangular grid (bottom panels in Figure 4.7), these high velocities are accompanied with quite some scatter, which suggests that the output results are not perfectly radialsymmetric.

Conclusion

D-Flow FM is able to computationally deal with flooding and drying in two-dimensional model configurations. For a relatively coarse grid, considerable deviations are accounted. In the shoreline area, considerable velocities are computed which are likely to be due to relative small water depths.

Version

This test has been carried out with version dflow-fm-x64-1.1.132.38471.

4.3 One-dimensional dambreak over a dry bed

Purpose

The purpose of this validation study is to test the D-Flow FM modelling accuracy for flooding over both a dry bed, resulting from a dam break. The D-Flow FM model results are compared with the analytical solution. In this validation study, multiple grids are considered.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.1.1: D-Flow FM can accurately simulate steady and unsteady hydrostatic flow on grids consisting of triangles, quadrilaterals, pentagons and hexagons
- ◇ claim 2.3.2.1: D-Flow FM can be used for an accurate prediction of flows resulting from dam breaks
- ◇ claim 2.3.2.4: D-Flow FM can accurately simulate drying and flooding of tidal areas

Approach

For one-dimensional dambreak flow over a dry bed, analytical expressions are available to compare the numerical solution with. The initial state is a stepwise water level with $h(x) = h_l$ for $0 \leq x \leq x_0$ and $h(x) = 0$ for $x_0 < x < L$, with the discontuinity at $x = x_0$ and a computational domain from $x = 0$ to $x = L$.

The analytical solution, for a dry bed case, can be prescribed for three regions, induced by two fronts, namely at $x = x_A$ and $x = x_B$, with $x_A = x_0 - t\sqrt{gh_l}$ and $x_B = x_0 + 2t\sqrt{gh_l}$. The analytical solution for the water level along the three parts reads:

$$h(x,t) = \begin{cases} h_l, & \text{if} \quad x < x_A(t), \\ \dfrac{4}{9g}\left(\sqrt{gh_l} - \dfrac{x - x_0}{2t}\right)^2, & \text{if} \quad x_A(t) \leq x \leq x_B(t), \\ 0 \text{ m}, & \text{if} \quad x > x_B(t). \end{cases} \tag{4.1}$$

The analytical solution for the velocity along the three parts reads:

$$u(x,t) = \begin{cases} 0 \text{ m/s}, & \text{if} \quad x < x_A(t), \\ \dfrac{2}{3}\left(\dfrac{x - x_0}{t} + \sqrt{gh_l}\right), & \text{if} \quad x_A(t) \leq x \leq x_B(t), \\ 0 \text{ m/s}, & \text{if} \quad x > x_B(t). \end{cases} \tag{4.2}$$

This solution shows if the scheme is able to locate and treat correctly the wet/dry transition. It also emphasizes whether the scheme preserves the positivity of the water height.

Model description

For this testcase, several grids are generated (see Figure 4.8). The computational settings and parameters are:

- ◇ domain length $L = 60$ km,
- ◇ time step $\Delta t = 1.0$ seconds,
- ◇ $h_l = 2$ m and $h_r = 0$ m,
- ◇ no horizontal viscosity and no bottom friction.

The computational grids developed for this case are shown in Figure 4.8. Grids 1, 2 and 3 contain square cells with increasing refinement, grid 4 contains triangular cells, grid 5 is an equidistant with elongated cells, grid 6 is a variant of grid 5 with a smoothly varying cell size, grid 7 has refinements in two directions.

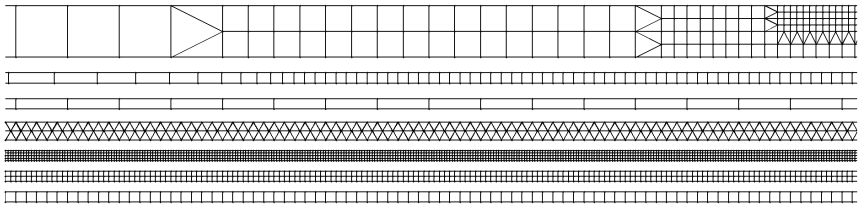


Figure 4.8: *Seven grids for the one-dimensional dambreak case. From bottom to top (numbered from 1 to 7): grids 1, 2 and 3 contain square cells with increasing refinement, grid 4 contains triangular cells, grid 5 is an equidistant with elongated cells, grid 6 is a variant of grid 5 with a smoothly varying cell size, grid 7 has refinements in two directions.*

For the parameter `chkadvd`, the value 0 m is chosen.

Results

The results of the one-dimensional dambreak are considered after 3200 seconds, when the waves have considerably propagated without having been reflected by the boundaries. Three issues are highlighted:

1 the effects of refinement in two directions are discussed (grids 1, 2 and 3),
2 the effects of refinement in one direction are considerend (grids 5 and 6),
3 the effects of the irregularity of the grid are discussed (grids 4 and 7).

Considering issue 1, it turned out that the three squared cell grids show decreasing deviations from the exact solution with decreasing cell size, as is to be expected. Since the mutual differences are relatively small, only the result for the finest grid are shown: see Figure 4.9. The location of the shock deviates from the exact solution by about 4 km, after 3200 seconds.

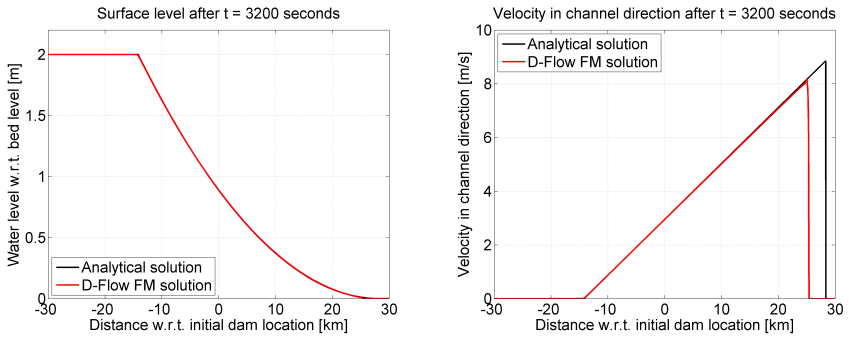


Figure 4.9: *Results for the most refined Cartesian grid (grid 3). Left panel: the water level (m); right panel: the velocity (m/s).*

Considering issue 2, it turned out that elongation of the cells considerably detoriates the result. The computation on the grid with the equidistantly distributed cells (see Figure 4.10) underestimates the velocities, but reproduces the shock location relatively well, whereas the

computation on the grid with smoothly varying cell size (see Figure 4.11) relatively well predicts the velocities but does a less good job in capturing the shock location.

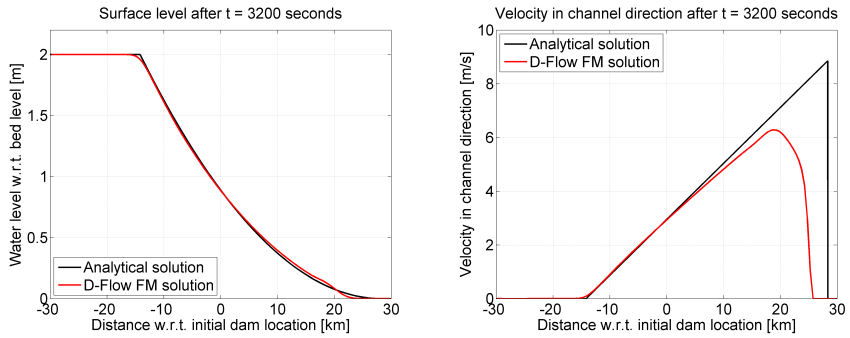


Figure 4.10: *Results for the grid containing equidistant, elongated rectangular cells (grid 5). Left panel: the water level (m); right panel: the velocity (m/s).*

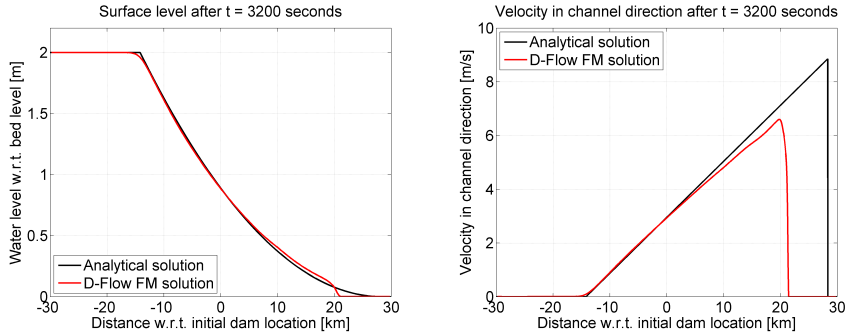


Figure 4.11: *Results for the grid containing rectangular cells with smoothly varying size (grid 6). Left panel: the water level (m); right panel: the velocity (m/s).*

Considering issue 3, it can be seen that the grid with triangular cells (see Figure 4.12) performs comparable with the squared cells grid. The grid with refinement in two directions (see Figure 4.13) captures the shock locations less accurately.

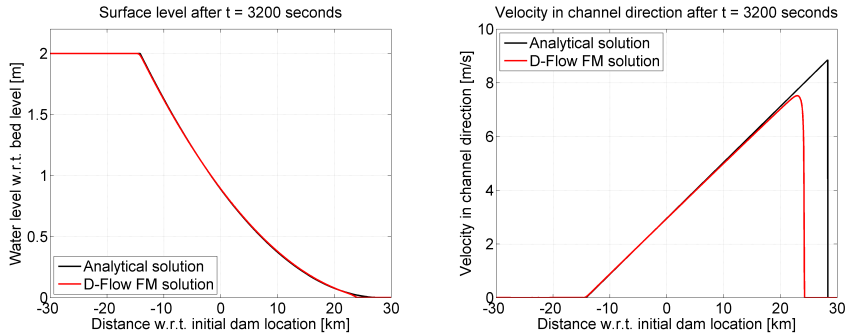


Figure 4.12: *Results for the triangular grid (grid 4). Left panel: the water level (m); right panel: the velocity (m/s).*

Conclusion

For a one-dimensional dambreak case over a dry bed, D-Flow FM performs fairly well regarding the prediction of the water level and the velocity. The prediction of the actual location of the propagating shock appeared to be relatively difficult. The best result are obtained using

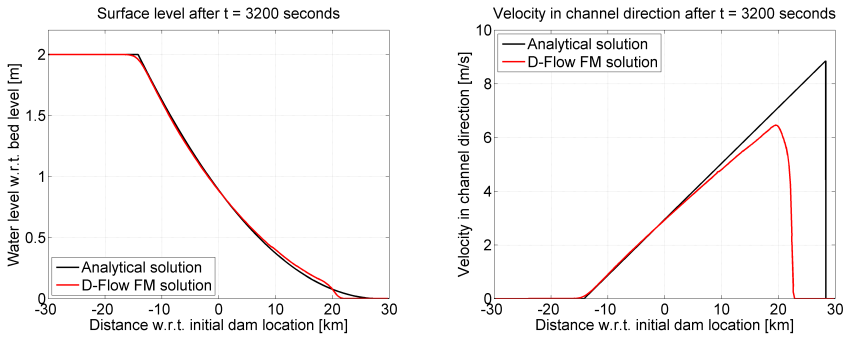


Figure 4.13: *Results for the grid with refinement in two directions (grid 7). Left panel: the water level (m); right panel: the velocity (m/s).*

square cell grids; deviations from the exact solutions become more pronounced as soon as the grid is more irregular.

Version

This test has been carried out with version dflow-fm-x64-1.1.116.36629.

4.4 Two-dimensional dambreak over a dry bed

Purpose

This validation study investigates the flow resulting from a dam break including the interaction of reflecting bores in a 2D model. The case investigated is based on the experiments by Stelling & Duinmeijer from 2001. Tests are conducted on Cartesian as well as on triangular grids.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.1.1: D-Flow FM can accurately simulate steady and unsteady hydrostatic flow on grids consisting of triangles, quadrilaterals, pentagons and hexagons
- ◇ claim 2.3.2.1: D-Flow FM can be used for an accurate prediction of flows resulting from dam breaks
- ◇ claim 2.3.2.4: D-Flow FM can accurately simulate drying and flooding of tidal areas

Approach

For this validation study a model is applied that is described in Stelling & Duinmeijer (2001). The geometry of the experimental setup is shown in Figure 4.14. The experiments are performed in a closed domain.

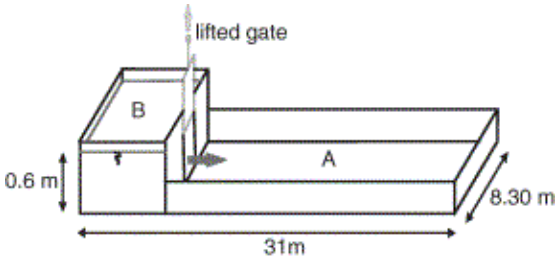


Figure 4.14: *Experimental setup of the Stelling & Duinmeijer experiment.*

Two experiments are considered by Stelling & Duinmeijer:

- ◇ a dam break with an initially wet region; and
- ◇ a dam break with an initially dry region.

Through the geometry of the experimental setup, reflecting bores interact with each other, resulting into quite a challanging validation case compared to canonical one-dimensional dam break problems. In the present validation case, the dry bed case is considered.

The experimental set-up consists of two reservoirs, A and B, separated by a wall with a gate of width 0.4 m that can be lifted. Reservoir B is initially filled with water of height 0.6 m. Two experiments are performed. The first experiment contains a thin layer of water with a depth of 0.05 m downstream of the gate and in the second experiment, the reservoir A is initially dry. The gate is then lifted with a speed of 0.16 m/s and the subsequent flooding in reservoir A is studied.

Model description

For this testcase, a Cartesian grid is adopted. The wall that divides the two basins is represented through thin dams. The computational settings and parameters are:

- ◇ domain length $L = 31$ m,
- ◇ domain width $D = 8.3$ m,
- ◇ the Cartesian grid has sizes $\Delta x = \Delta y = 0.1$ m and hence 310 $\times$ 73 cells,
- ◇ the triangular grid consists of triangular cells with a typical edge length of 0.1 m.
- ◇ time step $\Delta t = 0.006$ seconds,
- ◇ Manning bottom roughness = 0.012 $m^{-1/3}$/s,
- ◇ the horizontal eddy viscosity $\nu_h = 5.0 \cdot 10^{-4}$ m^2/s.

For the parameter `chkadvd`, the value of 0 m is set.

Results

An artist's impression of the result of the D-Flow FM computation is shown in Figure 4.15. In this figure, the bore has already been reflected from the boundary at $x = L = 31$ m, turning in a backwards propagating front.

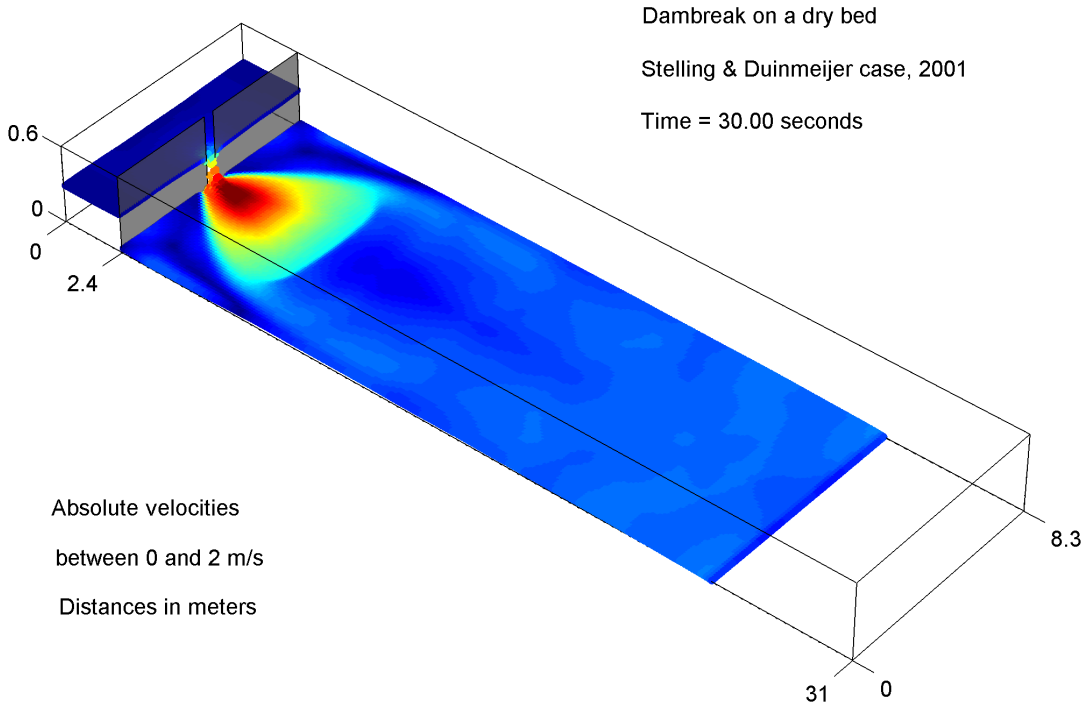
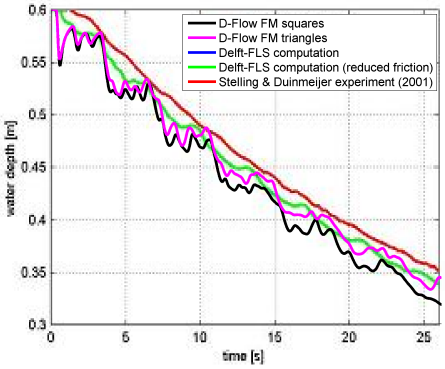


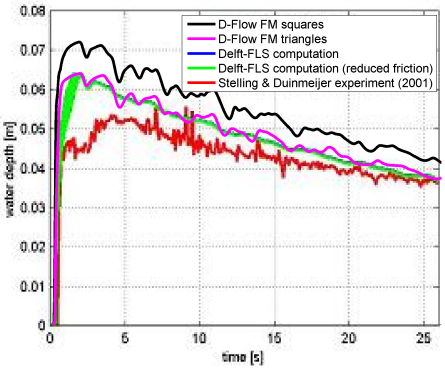
Figure 4.15: *Computational result after 30 seconds, showing the water level and the absolute velocities.*

Timeseries for the water level are available at multiple locations along the centerline of the flume. Six of these locations are highlighted in Figure 4.16. With reference to the gate location, these six gauges are located at -1 m, +1 m, 6 m, 9 m, 13 m and 17 m.

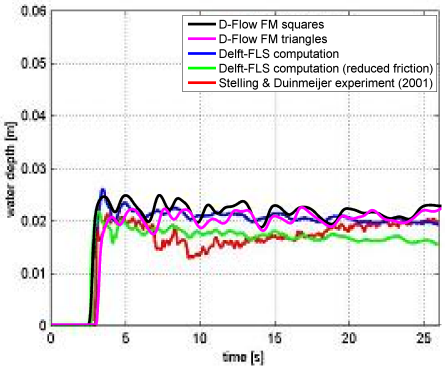
Figure 4.16 shows that in the D-Flow FM simulation, the reservoir empties quicker compared to the experiment. At later stages, the D-Flow FM results are comparable to the Delft-FLS results. A remarkable difference is seen at 6 m downstream of the gate: the D-Flow FM computations reveals a backwards reflected bore, induced by the sidewalls, whereas this bore



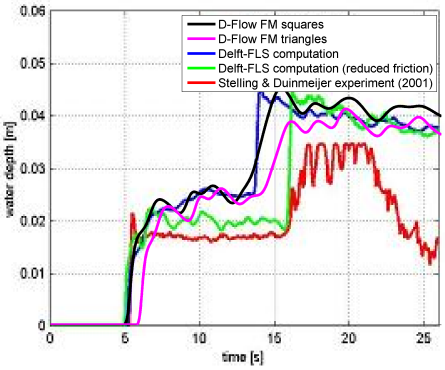
(a) *Results at 1 m upstream of the gate.*



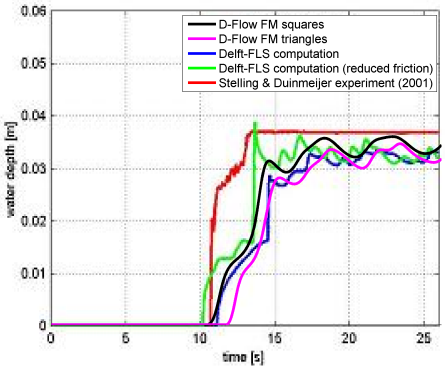
(b) *Results at 1 m downstream of the gate.*



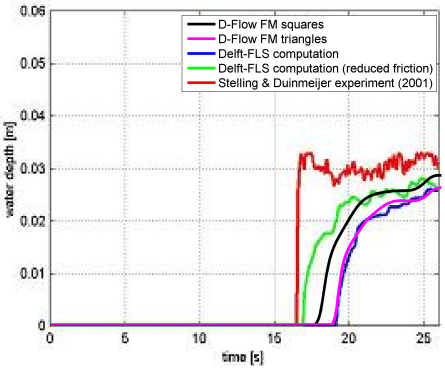
(c) *Results at 6 m downstream of the gate.*



(d) *Results at 9 m downstream of the gate.*



(e) *Results at 13 m downstream of the gate.*



(f) *Results at 17 m downstream of the gate.*

Figure 4.16: *Results for the two-dimensional dambreak on a dry bed. In lexicographic ordering: water depth in meters at -1 m, +1 m, 6 m, 9 m, 13 m and 17 m downstream of the gate.*

is not seen in the experiment and Delft-FLS results. From the pictures at 9 m downstream of the gate, this bore is again visible, proving that it travels the most rapidly in the D-Flow FM computation. Variants of the D-Flow FM computation can now be undertaken to study the influences of the gridtype, the gridsize, the value of the bottom friction and the horizontal eddy-viscosity.

Mutual differences between the results obtained on the Cartesian grid and the triangular grid are difficultly interpretable. On the Cartesian grid, the reservoir seems to empty faster that in

the triangular case. Moreover, the wave front due to the dam break appears to move somewhat faster in the Cartesian case compared to the triangular case. However, strict conclusions that address the actual performance difference between the two simulations cannot be drawn unequivocally.

Conclusion

D-Flow FM provides fairly good results for the two-dimensional dambreak on a dry bed. The D-Flow FM results are close to the Delft-FLS results, but should further be explored to investigate what the main causes are regarding the differences with the experiment.

Version

This test has been carried out with version dflow-fm-x64-1.1.116.36629.

5 Diffusion

5.1 Poiseuille flow with partial slip sidewalls

Purpose

The purpose of this validation case is to examine the performance of D-Flow FM for a plane Poiseuille flow simulation with a partial slip condition at the sidewalls. By 'partial slip', it is meant that the streamwise velocities at the wall are not zero though dictated by friction. D-Flow FM treats this kind of boundaries by imposing wall friction based on the logarithmic law-of-the-wall.

Poiseuille flow is pressure-induced flow in a long duct, in which case the flow is confined by two sidewalls. Poiseuille flow is distinguished from drag-induced flow such as Couette flow. Furthermore, it is assumed that there is laminar flow of an incompressible Newtonian fluid of dynamic viscosity μ) induced by a constant positive pressure difference or pressure drop Δp in a channel of length L and width $B \ll L$.

In fact, a Poisseuille flow with partial slip sidewalls comprises an internal contradiction, since Poisseuille flow is inherently laminar, whereas partial slip is based on a turbulent wall-law. Nonetheless, this hypothetical testcase facilitates useful comparison from an academic point of view.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.3.2: D-Flow FM accurately uses the horizontal eddy viscosity concept for partial-slip walls

Approach

For this two-dimensional test case, x and y are defined as the longitudinal and lateral coordinates, respectively, with respective velocity components u and v. The origin of the coordinate system is put at the center of the entrance of the channel. The flow is driven by a pressure gradient $\nabla p = (-P, 0) = (-\rho g \Delta H/L, 0)$ with ΔH the water level difference between the outflow boundary and inflow boundary, prescribing a flat bottom. The kinematic viscosity $\nu = \mu/\rho$ is used. Given the context of the shallow water equations, a uniform horizontal eddy viscosity ν_h is used instead.

At the sidewalls, i.e. at $y = \pm B/2$, a partial slip condition is prescribed. This condition is implemented on the basis of a factor α, being a constant of proportionality between the friction velocity u_* and the actual velocity u, such that $u_* = \alpha u$. This factor α is prescribed as:

$$\alpha = \kappa \Big/ \ln\left(1 + \frac{\Delta y}{2y_0}\right), \tag{5.1}$$

in which κ Von Kármán's constant equal to 0.41, $y_0 = k_s/30$ and Δy the grid size of a cell adjacent to the wall. The parameter k_s is specified in the `mdu`-file. Using the coefficients c_0 and c_1:

$$c_0 = -g\frac{\Delta H}{2L} \qquad \text{and} \qquad c_1 = \frac{\nu_h}{\alpha}\sqrt{B \cdot c_0} + c_0\left(\frac{B - \Delta y}{2}\right)^2, \tag{5.2}$$

the velocity profile can now be predicted by:

$$u(y) = \frac{c_1 - c_0 y^2}{\nu_h} \qquad \text{and} \qquad v = 0. \tag{5.3}$$

Model description

A Cartesian grid consisting of 200×20 cells is established, covering a domain of sizes $L \times B$ equal to 10 km $\times$ 1 km. A flat bottom is used. The following flow settings are chosen:

- ◇ constant horizontal eddy viscosity $\nu_h = 0.1$ m^2/s,
- ◇ Nikuradse roughness $k_s = 0.1$ m,
- ◇ water level drop $\Delta H = -1.0 \cdot 10^{-4}$ m.

Through these parameters, the analytical velocity profile is fully determined. The advection scheme nr. 3 (assigned as `Perot q(uio-u)`) is used.

Results

First, the solution has to reach a steady state. To check if the steady state has been reached, the criterion $|\partial u/\partial t|_\infty < 10^{-10}$ m/s^2 is used. The computed velocity profile at the center of the domain is shown in comparison with the analytical solution in Figure 5.1.

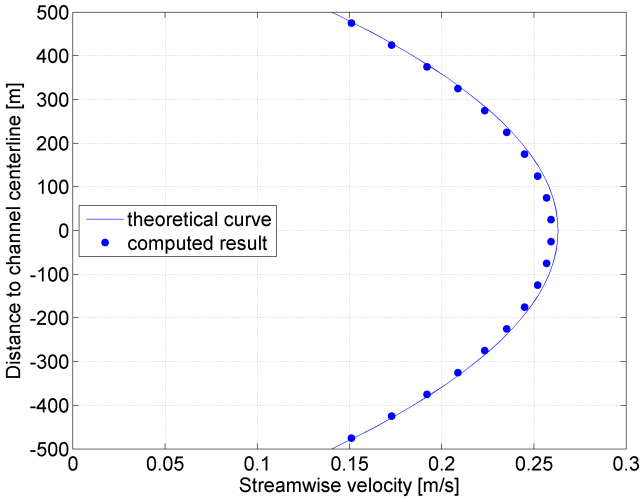


Figure 5.1: *Computed profile and analytical profile for the plane Poiseuille flow case with partial slip sidewalls.*

The computed results tend to slightly underestimate the analytical results. The computed results near the centerline (not *at* the centerline, due to staggering) differs -0.7261% from its analytical counterpart. This issue is currently under investigation by Mart Borsboom. The maximum absolute lateral velocity is of the order of 10^{-8} m/s.

Conclusion

For plane Poisseuille flow with partial slip sidewalls, D-Flow FM approximates the analytical solution fairly well: the parabolic profile is reproduced and the maximum velocity is approximated within 1% accuracy. The numerical results tend to underestimate the analytical results.

Version

This test has been carried out with version dflow-fm-x64-1.1.116.36629.

5.2 Poiseuille flow with no-slip sidewalls

Purpose

The purpose of this validation case is to examine the performance of D-Flow FM for a plane Poiseuille flow simulation with a no-slip condition at the sidewalls. This no-slip condition inheres zero-velocities at the sidewalls. Poiseuille flow is pressure-induced flow in a long duct, in which case the flow is confined by two sidewalls. Poiseuille flow is distinguished from drag-induced flow such as Couette flow. Furthermore, it is assumed that there is laminar flow of an incompressible Newtonian fluid of dynamic viscosity μ) induced by a constant positive pressure difference or pressure drop Δp in a channel of length L and width $B \ll L$.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.3.1: D-Flow FM accurately uses the horizontal eddy viscosity concept for no-slip walls

Approach

For this two-dimensional test case, x and y are defined as the longitudinal and lateral coordinates, respectively, with respective velocity components u and v. The origin of the coordinate system is put at the center of the entrance of the channel. The flow is driven by a pressure gradient $\nabla p = (-P, 0) = (-\rho g \Delta H/L, 0)$ with ΔH the water level difference between the outflow boundary and inflow boundary, prescribing a flat bottom. The kinematic viscosity $\nu = \mu/\rho$ is used. Given the context of the shallow water equations, a uniform horizontal eddy viscosity ν_h is used instead. The velocities u and v can be shown to satify:

$$u(y) = \frac{g\Delta H}{8\nu L}\left(B^2 - 4y^2\right) \qquad \text{and} \qquad v = 0. \tag{5.4}$$

Model description

A Cartesian grid consisting of 200×20 cells is established, covering a domain of sizes $L \times B$ equal to 10 km $\times$ 1 km. A flat bottom is used. The following flow settings are chosen:

- ◇ constant horizontal eddy viscosity $\nu_h = 0.1$ m^2/s,
- ◇ water level drop $\Delta H = -1.0 \cdot 10^{-4}$ m.

Through these parameters, the analytical velocity profile is fully determined. The advection scheme nr. 3 (assigned as `Perot q(uio-u)`) is used.

Results

First, the solution has to reach a steady state. To check if the steady state has been reached, the criterion $|\partial u/\partial t|_\infty < 10^{-10}$ m/s^2 is used. The computed velocity profile at the center of the domain is shown in comparison with the analytical solution in Figure 5.2.

The computed results tend to slightly underestimate the analytical results. The computed results near the centerline (not *at* the centerline, due to staggering) differs -0.3507% from its analytical counterpart. This issue is currently under investigation by Mart Borsboom. The maximum absolute lateral velocity is of the order of 10^{-8} m/s.

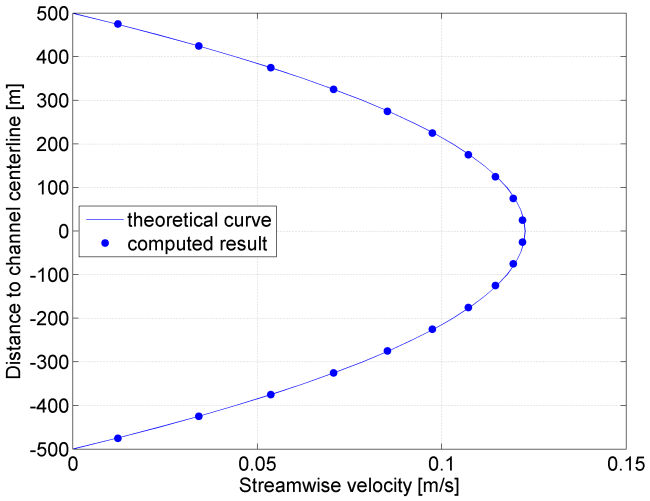


Figure 5.2: *Computed profile and analytical profile for the plane Poiseuille flow case with no-slip sidewalls.*

Conclusion

For plane Poisseuille flow with no-slip sidewalls, D-Flow FM approximates the analytical solution fairly well: the parabolic profile is reproduced and the maximum velocity is approximated within 1% accuracy. The numerical results tend to underestimate the analytical results.

Version

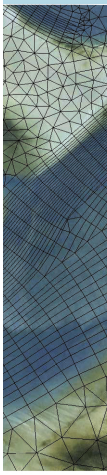
This test has been carried out with version dflow-fm-x64-1.1.116.36629.

6 Coriolis

6.1 Effects of Coriolis' force on the flow in a straight channel

Purpose

The purpose of this validation case is to examine the balance between the Coriolis force and the slope of the water surface in a long straight channel with a uniform depth. This is a non-trivial case, because the equation for u is defined in different points than v, consequently an interpolation scheme is needed to give an estimate of v to evaluate the contribution from the Coriolis force to the momentum balance.



Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.5.1: D-Flow FM can take into account the impact of the Coriolis force associated with the rotation of the earth

Approach

The model runs are run without advection, without diffusion and without friction. To achieve that, the following parameters are set:

- ◇ `AdvecType` is set to `0`;
- ◇ `UnifFrictCoef` is set to `0.0`;
- ◇ `Vicouu` is set to `0.0`;
- ◇ `Vicouv` is set to `0.0`;
- ◇ `Smagorinsky` is set to `0.0`;
- ◇ `Elder` is set to `0.0`;
- ◇ `irov` is set to `0`;
- ◇ `Vicoww` is set to `0.0`;
- ◇ `TidalForcing` is set to `0`

The inlet boundary is set to a constant velocity of 0.1 m/s over the entire cross section. The computational domain is 500 km long and 300 km wide.

The outlet boundary condition applies the Riemann invariant type boundary condition, which takes the form

$$\zeta = 2R_i - \sqrt{\frac{H}{g}}u \tag{6.1}$$

on the boundary. The Riemann invariant (R_i) is given by the user on the outlet boundary. Note that u is positive inward. Here H is the total water depth, ζ is the surface elevation and g is the acceleration due to gravity. The surface elevation in the steady-state situation is given as

$$\zeta(y) = -\frac{f(y-y_0)}{g}u \tag{6.2}$$

where f is the Coriolis constant and $y_0 = 150$ km is taken as the middle of the channel. The Coriolis constant is set to constant in the computational domain by use of the `AngLat`

keyword, and the latitude is taken for 45° North. The Coriolis constant is given as

$$f = 2\Omega \sin\varphi \quad \text{where} \quad \Omega = \frac{2\pi}{T} \quad \text{[Hz]} \tag{6.3}$$

with T equal to 23 hr. 56 m and 4.1 s, in seconds. φ is the latitude. Notice that the analytical solution for this test case is independent of the lateral bed level variations as long as the corresponding Riemann invariant is properly applied at the outflow boundary.

Several variations of the subject are considered in this test case:

- ◇ variation of the gridcell size,
- ◇ variation of the type of grid: quadrilateral or triangular,
- ◇ variation of the shape of the bathymetry,
- ◇ variation of the way the Riemann boundary condition is imposed.

Model description

To investigate whether or not the numerical solution depends on lateral bed level variations, four bed configurations are considered in this test case (also see Figure 6.1 for the visualization):

1. a flat bed at a level of 500 m below the reference level,
2. a linearly varying bed, from 50 m to 500 m below the reference level,
3. a piecewise linearly varying bed, with flat areas at a level of 50 m below the reference level and at a level of 500 m below reference level, respectively,
4. a bed level that varies according to the the shape of a cosine, varying from 50 m to 500 m below the reference level.

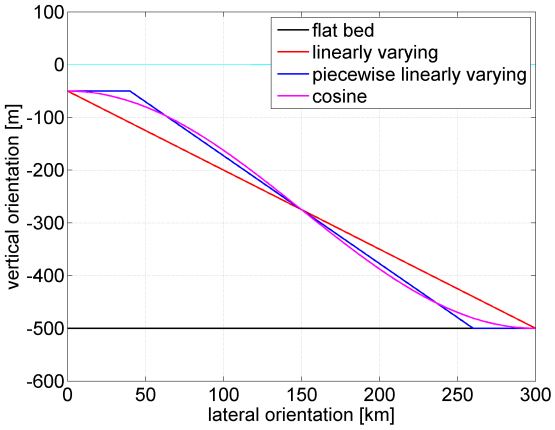


Figure 6.1: *Various bed level variation approaches for the Coriolis test case. The topography only varies in lateral direction; the bed level is constant in longitudinal (streamwise) direction.*

Multiple grids are utilized for this test case. In the basis, distinction is made between grids with quadrilateral cells and with triangular cells. For both types of grids, several grades of refinement are generated. The refined Cartesian grids are generated by means of refining the grid, shown in the left panel of Figure 6.2 once and twice with a factor of 2. The refined triangular grid is generated by means of refining the grid, shown in the right panel of Figure 6.2, once with a factor of 2. The setups shown in Figure 6.2 have a cosine-shape for the bed, in lateral direction.

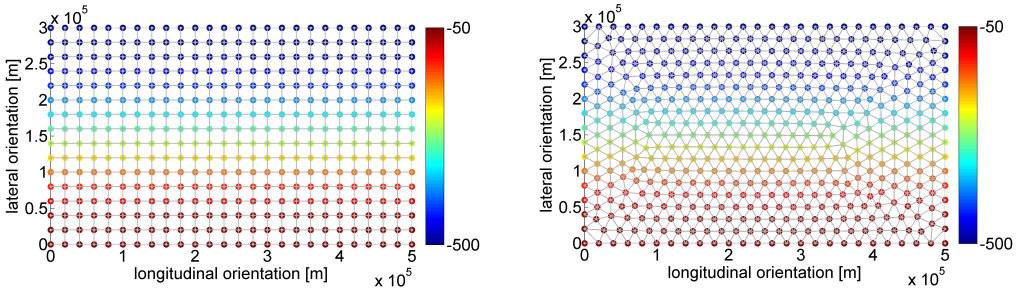


Figure 6.2: *Grids used as basis for the test case. Multiple refinement grades are generated as well. The colors indicate the bed level. For the bed level, the cosine-shaped bathymetry is visualized here as an example.*

Results

In this section, the results are addressed from multiple viewpoints. First, the way of imposing the Riemann invariant is addressed. Second, the effects of grid refinement are investigated for the quadrilateral grids. Third, aspects of the bed level treatment are briefly commented on. Fourth, the grid type (quadrilateral versus triangular) is addressed.

Spatially varying bathymetry: aspects of prescribing the Riemann invariant
Prescribing a Riemann invariants invokes the need to know the bed level at the boundary under consideration. Suppose, the gridcell as visualized in Figure 6.3 represents a gridcell at the edge of the grid.

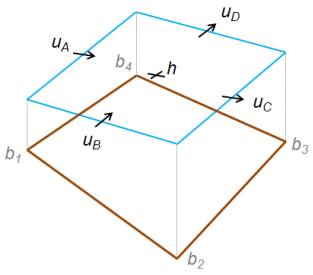


Figure 6.3: *Computational cell with surface level h, face velocities u_A, u_B, u_C and u_D and bed levels b_1, b_2, b_3 and b_4.*

Think of a boundary at the face with u_C as the face-normal velocity and b_2 and b_3 as the bed levels along the boundary. The aim is to prescribe the Riemann invariant at the cell center of the ghost-cell outside the boundary, i.e. the mirrored equivalent of the cell shown. As an actual depth, necessary to prescribe the Riemann invariant, the value $(b_2 + b_3)/2$ might be the most intuitive choice.

Two approaches to specify the Riemann invariant are followed and mutually compared:

1 use $(b_2 + b_3)/2$ for computing the local water depth,
2 use the lowest value, in casu b_2, for computing the locat water depth.

The results for both approaches are visualized in Figure 6.4 for each bathymetry (flat bed, linear bed, piecewise linear bed and cosinal bed). Notice that for the flat bed, no effect of either the one choice or the other is expected (constant depth). Moreover, notice that taking the

lowest bed level is equivalent to shift the Riemann boundary data series along the boundary over half a grid cell.

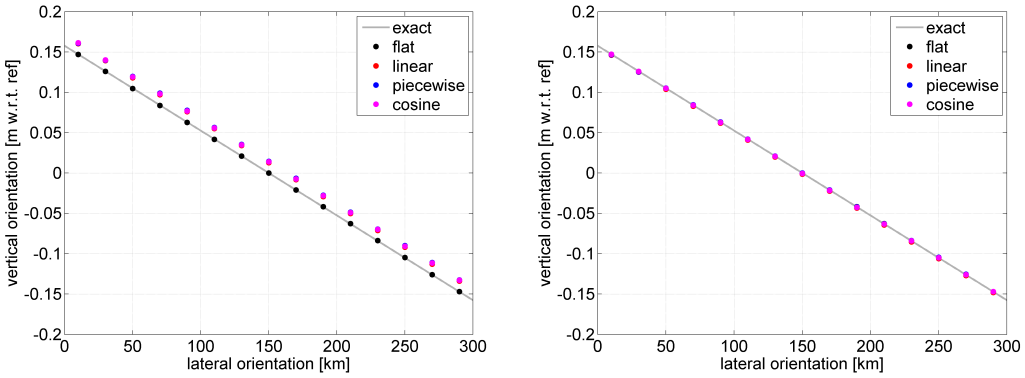


Figure 6.4: *Numerical solution for the various bathymetry configuration in relation to the exact solution. Left panel: standard way of imposing a Riemann invariant; right panel: with a Riemann invariant, imposed with a shift of half a gridcell.*

The results, shown in Figure 6.4, show that taking the lowest bed level for the computation of the local water depth returns a good result, whereas this is not the case for the approach in which we use the mean bed level. For all the bathymetry configurations, the results coincide.

Spatially varying bathymetry: aspects of the bed level treatment
For the treatment of the bed level (cf. Figure 6.3), the user can use the mdu-keywords `conveyance2d` and `bedlevtyp` to make D-Flow FM handle according to the user's demands. In order to explore the effects of the joint choice for both keywords, the options `conveyance2d = -1/3` and the options `bedlevtyp = 1/2/3/4/5` are varied in all possible combinations.

The results of these additional computations are as follows:

- ◇ the options `bedlevtyp = 1/2` returned poor results, regardless of the choice for `conveyance2d`,
- ◇ the options `bedlevtyp = 3/4/5` returned mutual differences that are of an order of magnitude smaller than the difference with the exact solution as shown in the left panel of Figure 6.4, regardless of the choice for `conveyance2d`.

Flat bathymetry: effects of grid refinement
The flat bed case is used for a grid convergence study. For this purpose, only the three Cartesian grids are considered: the first with cells with an edge length of 20 km, the second 10 km and the third 5 km. The differences with the exact solution are measured with an L_2-norm. For each simulation, the timestep $\Delta t = 50$ seconds is chosen. The simulations cover 10 days of simulated time, which has proven to be sufficient to reach a steady state.

The L_2-norm of the difference between the exact solution and the numerical solution is shown in Figure 6.5 against the typical grid cell size. Figure 6.5 reveals zeroth order convergence of the L_2-norm measure.

On the type of the grid: quadrilateral or triangular cells
The simulations for the flat bed setup and the cosine bed setup have been conducted on both a quadrilateral (or Cartesian) grid and a triangular grid. This enables a comparison of the accuracy of both approaches. Using the L_2-norm of the differences between the exact water levels and the computed water levels, the results read:

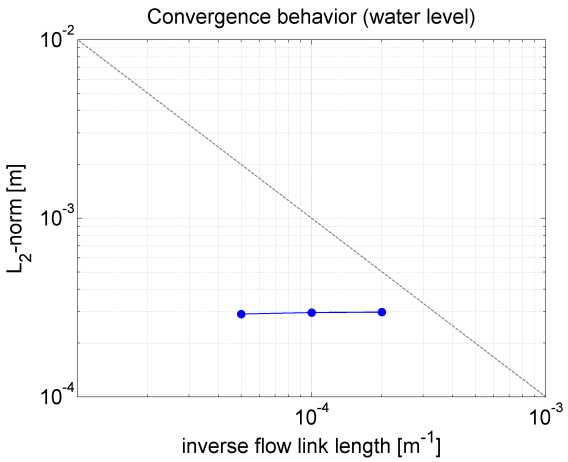


Figure 6.5: *Convergence behavior of the L_2-norm of the water level difference (the numerical solution minus the exact solution). The dashed line represents first order behavior.*

- ◇ Flat bed bathymetry on a Cartesian grid: 2.963195095572729e-04 m,
- ◇ Flat bed bathymetry on a triangular grid: 4.813599995505768e-02 m,
- ◇ Cosine shaped bathymetry on a Cartesian grid: 1.405800785225406e-02 m,
- ◇ Cosine shaped bathymetry on a triangular grid: 8.379261007472463e-03 m.

For the flat bed case, the triangular grid returns less accurate results compared to the quadrilateral grid. For the cosine shaped bathymetry case, the opposite appears to hold. Hence, no clear general conclusion can be drawn on the basis of these results.

Conclusion

From the above considerations, the following conclusions can be drawn:

- ◇ D-Flow FM is able to reproduce the analytical solution for the balance between the Coriolis force and the slope of the water surface in a long straight channel, regardless of the bathymetry shape,
- ◇ the way of prescribing a Riemann invariant at the outflow boundary could be perceived of as confusing regarding the value of the bed level to use for the water depth definition,
- ◇ for the flat bed case, zeroth order convergence behavior is measured for the computed water level with decreasing grid cell size,
- ◇ no clear conclusion can be drawn regarding the accuracy of quadrilateral grids versus triangular grids.

Version

This test has been carried out with version dflow-fm-x64-1.1.136.39235.

6.2 Coriolis in a frictionless basin

Purpose

The purpose of this testcase is to assess the ability of D-Flow FM to compute geophysical flows under influence of the Earth's rotation through the Coriolis force. In some cases, the computational results can be compared with (semi-)analytical solutions. This case comprises propagating waves in a rectangular, semi-closed basin for which a solution can be derived semi-analytically: the classical Taylor problem from 1921.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.5.1: D-Flow FM can take into account the impact of the Coriolis force associated with the rotation of the earth

Approach

A rectangular basin as given in Figure 6.6 is adopted. The basin is closed at three edges and open at the boundary at $x = L$, with L the length of the basin. For this setup, Taylor (1921) has figured out that the solution of the shallow water equations then yields the forcing Kelvin wave, a reflected Kelvin wave and an infinite number of so-called Poincaré waves. This solution can be derived semi-analytically using the *collocation method*.

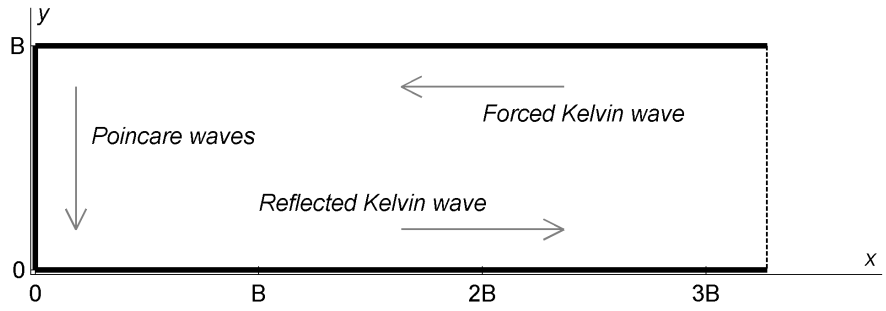


Figure 6.6: *The classical Taylor problem: propagation of waves in a semi-closed basin of rectangular shape.*

To mimic the semi-analytical solution of Taylor, the water level elevation at the open boundary at $x = L$ is prescribed. This elevation, denoted by ζ, is the sum of the incoming and outgoing Kelvin wave, based on the solution for an infinitely long channel bounded by sidewalls at $y = 0$ and $y = B$. These so-called pseudo-standing Kelvin waves are described by the following expression:

$$\zeta(y) = \frac{2c}{g}U_0 \exp\left(-\frac{fB}{2c}\right) \cdot \cosh\left(\frac{f}{c}\left(\frac{B}{2} - y\right)\right) \cdot \cos kx \cos \omega t \quad (6.4)$$

$$+ \frac{2c}{g}U_0 \exp\left(-\frac{fB}{2c}\right) \cdot \sinh\left(\frac{f}{c}\left(\frac{B}{2} - y\right)\right) \cdot \sin kx \sin \omega t \quad (6.5)$$

in which B is the width of the basin, g the gravitational accelaration, f the Coriolis parameter being $f = 2\Omega \sin \varphi$, with Ω the Earth's rotation $2\pi/24/3600 = 7.2722 \cdot 10^{-5}$ rad/s and φ the latitude, c the phase speed of the forcing's wave equal to $\sqrt{gH}$, with H the water depth. The parameters k and ω are the wavenumber and wavefrequency, respectively, and U_0 is the forcing's amplitude.

The expression for the surface elevation ζ can be rewritten as:

$$\zeta_{in}(y) = \frac{c}{g}U_0 \cdot \quad \exp\left(-\tfrac{fB}{c}\right) \cdot \quad \exp\left(+\frac{fy}{c}\right) \cdot (\cos kx \cos\omega t - \sin kx \sin\omega t) \quad (6.6)$$

$$\zeta_{out}(y) = \frac{c}{g}U_0 \cdot \qquad\qquad\qquad \exp\left(-\frac{fy}{c}\right) \cdot (\cos kx \cos\omega t + \sin kx \sin\omega t) \quad (6.7)$$

which enables a prescription of the boundary conditions by means of two signals with different amplitude and with a mutual phase difference equal to the phase difference between $\cos kx \cos\omega t - \sin kx \sin\omega t$ and $\cos kx \cos\omega t + \sin kx \sin\omega t$.

Model description

The computational domain is the rectangular basin as shown in Figure 6.6. The chosen input for the domain is as follows:

- ◇ length L × width B is 1800 × 550 km^2,
- ◇ the water depth H is equal to 80 m (and hence, the phase speed c is determined),
- ◇ the latitude $\varphi = 52°$ (and hence, the Coriolis parameter f is determined),
- ◇ the forcing amplitude U_0 is chosen equal to c/g,
- ◇ the waveperiod T of the forcing wave is equal to 745 minutes (and hence, the wave-frequency ω is determined as $\omega = 2\pi/T$ as well as the wavenumber k, namely as $k = \omega/c = \omega/\sqrt{gH}$).

The friction is turned off; the horizontal eddy viscosity has the uniform value of 0 m^2/s. The sidewalls are impermeable, but frictionless (normal velocities equal zero). At $x = L$ the phase lag between ζ_{in} and ζ_{out} can be computed as $45.12°$.

Three computational grids are examined:

1. a coarse Cartesian grid consisting of 72×22 cells of 25.0×25.0 km^2,
2. a fine Cartesian grid consisting of 144×44 cells of 12.5×12.5 km^2,
3. a triangular grid with a typical cell edge size comparable to the typical grid size of the fine Cartesian grid.

Results

The results for each of the three grids are shown in Figure 6.7 in comparison with the semi-analytical solution. Maximum surface elevations are shown.

The bottom panel (semi-analytical solution) shows a symmetric image of the absolute surface elevations. The solution, basically a superposition of the two Kelvin waves and a large number of Poincaré waves) consists of three amphidromic points.

The computational results (the three top panels) show that D-Flow FM has difficulties in reproducing the amplitudes of the surface elevation. The location of the amphidromic points seems to agree with the semi-analytical solution. However, the image is diffuse. It should further be investigated what the explanation for this diffuse picture is.

Conclusion

For geophysical flow under influence of the Coriolis force, but in the absence of friction and horizontal eddy viscosity, D-Flow FM experiences difficulties in reproducing the amplitudes of the surface elevation. The location of the amphidromic points seems to agree with the semi-analytical solution.

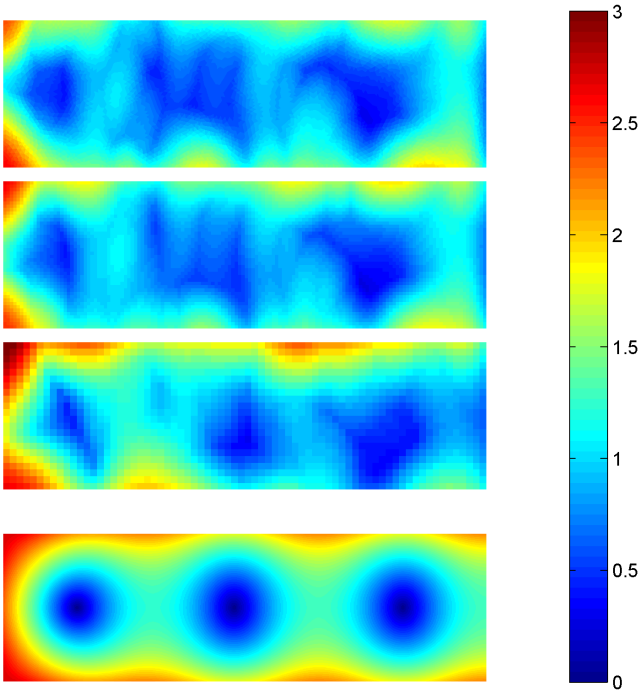


Figure 6.7: *Semi-analytical solution (bottom panel) for the Coriolis test without friction. The upper three panels show the computational result from D-Flow FM on the triangular grid (upper panel), the fine Cartesian grid (second panel) and the coarse Cartesian grid (third panel). The colors span surface elevations from 0 m to 2.8 m.*

Version

This test has been carried out with version dflow-fm-x64-1.1.136.39235.

Acknowledgement

The writer is thankful to mr. Koen Berends for providing Matlab-scripts to compute the semi-analytical solution. These Matlab-scripts are based on: Roos, P.C., Velema, J.J., Hulscher, S.J.M.H., Stolk, A. (2011), An idealized model of tidal dynamics in the North Sea: resonance properties and response to large-scale changes, Ocean Dynamics, 61 (12), 2019-2035.

6.3 Coriolis in a basin with bottom friction

Purpose

The purpose of this testcase is to assess the ability of D-Flow FM to compute geophysical flows under influence of the Earth's rotation through the Coriolis force. In some cases, the computational results can be compared with (semi-)analytical solutions. This case comprises propagating waves in a rectangular, semi-closed basin for which a solution can be derived semi-analytically: the classical Taylor problem from 1921.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.5.1: D-Flow FM can take into account the impact of the Coriolis force associated with the rotation of the earth

Approach

A rectangular basin as given in Figure 6.8 is adopted. The basin is closed at three edges and open at the boundary at $x = L$, with L the length of the basin. For this setup, Taylor (1921) has figured out that the solution of the shallow water equations then yields the forcing Kelvin wave, a reflected Kelvin wave and an infinite number of so-called Poincaré waves. This solution can be derived semi-analytically using the *collocation method*.

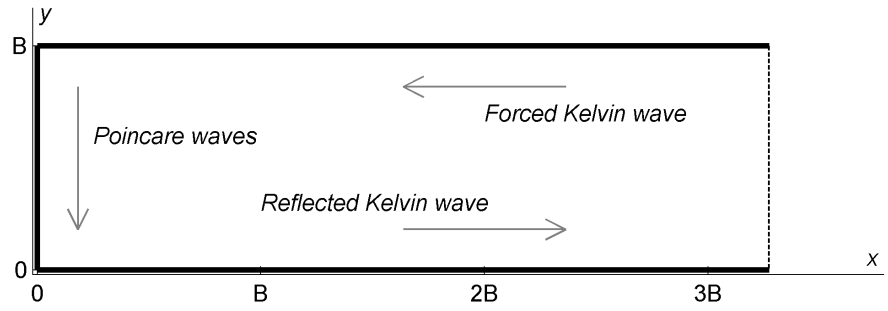


Figure 6.8: *The classical Taylor problem: propagation of waves in a semi-closed basin of rectangular shape.*

To mimic the semi-analytical solution of Taylor, the water level elevation at the open boundary at $x = L$ is prescribed. This elevation, denoted by ζ, is the sum of the incoming and outgoing Kelvin wave, based on the solution for an infinitely long channel bounded by sidewalls at $y = 0$ and $y = B$. These so-called pseudo-standing Kelvin waves are described by the following expression:

$$\zeta(y) = \frac{2c}{g}U_0 \exp\left(-\frac{fB}{2c}\right) \cdot \cosh\left(\frac{f}{c}\left(\frac{B}{2} - y\right)\right) \cdot \cos kx \cos \omega t \quad (6.8)$$

$$+ \frac{2c}{g}U_0 \exp\left(-\frac{fB}{2c}\right) \cdot \sinh\left(\frac{f}{c}\left(\frac{B}{2} - y\right)\right) \cdot \sin kx \sin \omega t \quad (6.9)$$

in which B is the width of the basin, g the gravitational accelaration, f the Coriolis parameter being $f = 2\Omega \sin \varphi$, with Ω the Earth's rotation $2\pi/24/3600 = 7.2722 \cdot 10^{-5}$ rad/s and φ the latitude, c the phase speed of the forcing's wave equal to $\sqrt{gH}$, with H the water depth. The parameters k and ω are the wavenumber and wavefrequency, respectively, and U_0 is the forcing's amplitude.

The expression for the surface elevation ζ can be rewritten as:

$$\zeta_{in}(y) = \frac{c}{g}U_0 \cdot \quad \exp\left(-\tfrac{fB}{c}\right) \cdot \quad \exp\left(+\frac{fy}{c}\right) \cdot (\cos kx \cos \omega t - \sin kx \sin \omega t) \tag{6.10}$$

$$\zeta_{out}(y) = \frac{c}{g}U_0 \cdot \qquad\qquad \exp\left(-\frac{fy}{c}\right) \cdot (\cos kx \cos \omega t + \sin kx \sin \omega t) \tag{6.11}$$

which enables a prescription of the boundary conditions by means of two signals with different amplitude and with a mutual phase difference equal to the phase difference between $\cos kx \cos \omega t - \sin kx \sin \omega t$ and $\cos kx \cos \omega t + \sin kx \sin \omega t$.

Model description

The computational domain is the rectangular basin as shown in Figure 6.8. The chosen input for the domain is as follows:

- ◇ length L × width B is 1800 × 550 km^2,
- ◇ the bottom friction is given by Chézy's coefficient C, equal to 62.64 m$^{1/2}$/s, corresponding with a drag coefficient $c_d = 2.5 \cdot 10^{-3}$,
- ◇ the water depth H is equal to 80 m (and hence, the phase speed c is determined),
- ◇ the latitude $\varphi = 52^\circ$ (and hence, the Coriolis parameter f is determined),
- ◇ the forcing amplitude U_0 is chosen equal to c/g,
- ◇ the waveperiod T of the forcing wave is equal to 745 minutes (and hence, the wave-frequency ω is determined as $\omega = 2\pi/T$ as well as the wavenumber k, namely as $k = \omega/c = \omega/\sqrt{gH}$).

The horizontal eddy viscosity has the uniform value of 0 m^2/s. The sidewalls are impermeable, but frictionless (normal velocities equal zero). At $x = L$ the phase lag between ζ_{in} and ζ_{out} can be computed as 45.12°.

Three computational grids are examined:

1. a coarse Cartesian grid consisting of 72×22 cells of 25.0×25.0 km^2,
2. a fine Cartesian grid consisting of 144×44 cells of 12.5×12.5 km^2,
3. a triangular grid with a typical cell edge size comparable to the typical grid size of the fine Cartesian grid.

Results

The results for each of the three grids are shown in Figure 6.9 in comparison with the semi-analytical solution. Maximum surface elevations are shown.

The bottom panel (semi-analytical solution) shows an asymmetric image of the absolute surface elevations. This asymmetry is caused by friction. The solution, basically a superposition of the two Kelvin waves and a large number of Poincaré waves) consists of three amphidromic points. The surface elevation amplitudes across the domain (ranging from 0 to 1.6 m) are smaller compared to the case without friction (ranging from 0 to 2.8 m) because of energy dissipation due to friction.

The computational results (the three top panels) show that D-Flow FM has difficulties in reproducing the amplitudes of the surface elevation, particularly in the corners of the domain. The location of the amphidromic points agree well with the semi-analytical solution. However, the image is rather diffuse. It should further be investigated what the explanation for the deviations in surface elevation amplitude is.

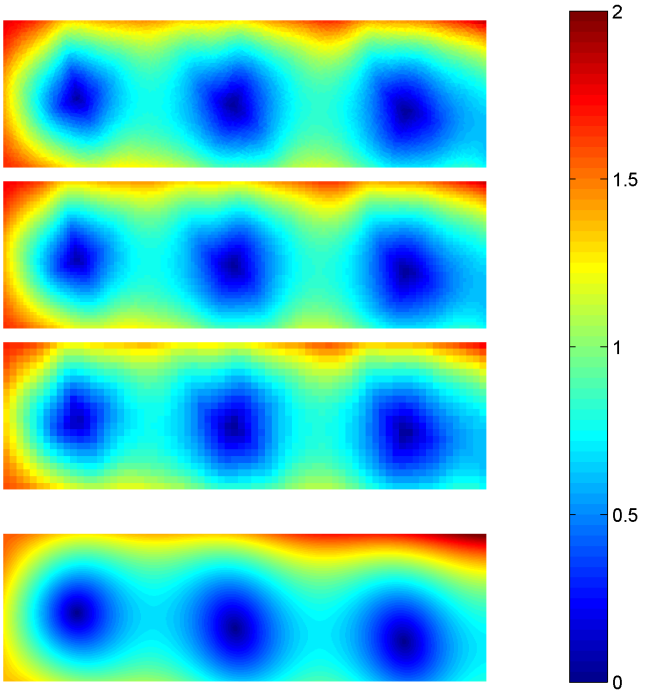


Figure 6.9: *Semi-analytical solution (bottom panel) for the Coriolis test with friction. The upper three panels show the computational result from D-Flow FM on the triangular grid (upper panel), the fine Cartesian grid (second panel) and the coarse Cartesian grid (third panel). The colors span surface elevations from 0 m to 1.6 m.*

Conclusion

For geophysical flow under influence of the Coriolis force and bottom friction, but under the absence of horizontal eddy viscosity, D-Flow FM experiences difficulties in reproducing the amplitudes of the surface elevation. The location of the amphidromic points agree well with the semi-analytical solution.

Version

This test has been carried out with version dflow-fm-x64-1.1.136.39235.

Acknowledgement

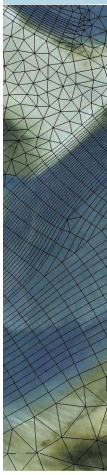
The writer is thankful to mr. Koen Berends for providing Matlab-scripts to compute the semi-analytical solution. These Matlab-scripts are based on: Roos, P.C., Velema, J.J., Hulscher, S.J.M.H., Stolk, A. (2011), An idealized model of tidal dynamics in the North Sea: resonance properties and response to large-scale changes, Ocean Dynamics, 61 (12), 2019-2035.

7 Bed friction

7.1 Bed friction formulations

Purpose

The purpose of this validation case is to examine the working of different bed friction formulations. D-Flow FM facilitates bed friction values for Chézy, Manning and White-Colebrook formulations. For White-Colebrook, two versions are available: one that is compliant with Delft3D and one that is compliant with WAQUA.



Linked claims

Claims that are related to the current test case are:

◇ claim 2.3.4.2: D-Flow FM can take into account several friction formulations (Chezy, Manning, White-Colebrook or roughness height z_0)

Approach

For this test case, a straight, stationary, homogeneous channel flow is chosen. Given a bottom that varies linearly with a slope i_b, a discharge Q and a certain friction coefficient, an equilibrium depth h_e can be computed. Given a Chézy friction value C, the water depth h_e is given by:

$$h_e = \left(\frac{Q}{BC\sqrt{i_b}} \right)^{2/3}. \tag{7.1}$$

In this test, four friction formulations are tested:

1 Chézy, with a friction factor C,
2 Manning, with a friction factor m, related to C through:

$$m = \frac{1}{C} h_e^{1/6}, \tag{7.2}$$

in which the constant '1' has a unit such that the coefficient m has no unit,
3 White-Colebrook (in Delft3D-formulation), with a friction factor w_d, related to C through:

$$w_d = \frac{30 \cdot h_e}{\exp\left(1 + \frac{\kappa}{\sqrt{g}} C\right)}, \tag{7.3}$$

in which κ (in the formulation of White-Colebrook for Delft3D) represents Von Kármán's constant equal to 0.41,
4 White-Colebrook (in WAQUA-formulation), with a friction factor w_q, related to C through:

$$w_q = \frac{12 \cdot h_e}{10^{C/18}}, \tag{7.4}$$

in which the constant '18' has the unit $\mathrm{m}^{1/2}$/s.

In principle, the h_e represents the hydraulic radius. However, in the current testcase, the water depth is considered the hydraulic radius.

Given these formulations, a certain Chézy friction factor can be chosen and, thereafter, be translated into other friction factor types through the above expressions.

Model description

For this test case, one particular computational grid is generated. The grid is of Cartesian type. The longitudinal size L of the domain is 10 km, whereas the lateral size B of the domain is 500 m. In longitudinal direction 20 cells are deployed, whereas in lateral direction only one grid cell is present.

A discharge Q equal to 2500 m^3/s is prescribed at the inflow boundary. The bottom slope i_b is prescribed to be 10^{-4}. The bottom level at the inflow boundary is 0 m (w.r.t. reference) and at the outflow boundary -1 m (w.r.t. reference). The Chézy friction coefficient C is set to 65 m$^{1/2}$/s. Given the above values, the equilibrium water depth h_e is 3.8967669 m. The other friction factors can hence be computed as:

1 Chézy factor (see e02-f04-c010) – $C = 65$ m$^{1/2}$/s,
2 Manning factor (see e02-f04-c011) – $m = 0.019299114465780$,
3 White-Colebrook factor (Delft3D) (see e02-f04-c012) – $w_d = 0.008674691060828$ m,
4 White-Colebrook factor (WAQUA) (see e02-f04-c013) – $w_q = 0.011449184837085$ m,

Results

The actual water level differences (numerical versus analytical) at cell face points along the channel are computed as the root-mean-square difference of the computed water depths and the analytical equilibrium water depth. The results are:

1 Chézy – $4.97119 \cdot 10^{-8}$ m
2 Manning – $2.07954 \cdot 10^{-6}$ m
3 White-Colebrook (Delft3D) – $8.13436 \cdot 10^{-9}$ m
4 White-Colebrook (WAQUA) – $5.62659 \cdot 10^{-9}$ m

Conclusion

For a stationary hydraulic computation of flow over an inclined bottom (with friction, a discharge inflow boundary and a Neumann outflow boundary), D-Flow FM is able to accurately treat friction formulations according to Chézy, Manning, White-Colebrook (Delft3D) and White-Colebrook (WAQUA).

Version

This test has been carried out with version dflow-fm-x64-1.1.116.36629.

7.2 Refinement study for channel flow using Cartesian grids

Purpose

The purpose of this validation case is to examine the performance of D-Flow FM for the simulation of a schematized uniform and homogeneous channel flow. This validation case focuses on the effects of the refinement of a Cartesian grid. The topography comprises a linearly varying bathymetry with a constant slope.

Linked claims

Claims that are related to the current test case are:

◇ claim 2.3.1.1: D-Flow FM can accurately simulate steady and unsteady hydrostatic flow on grids consisting of triangles, quadrilaterals, pentagons and hexagons

Approach

To enable academic comparison of the computational output with a known analytical solution, the topography of the bed varies linearly with a slope i_b. Given a discharge Q and a friction coefficient (in this case, Chézy's factor C is used), an equilibrium water depth d_e can be computed as:

$$d_e = \left(\frac{Q}{BC\sqrt{i_b}} \right)^{2/3} . \tag{7.5}$$

If the grid is refined several times, the order of accuracy of the numerical integration routines can be assessed. For this purpose, only Cartesian grids are considered (triangular grids are considered in case e02-f01-c011).

Model description

For this test case, five computational grids are generated. The outflow parts of these grids are shown in Figure 7.1. The five grids are of Cartesian type. The longitudal size L of the domain is 10 km, whereas the lateral size B of the domain is 500 m.

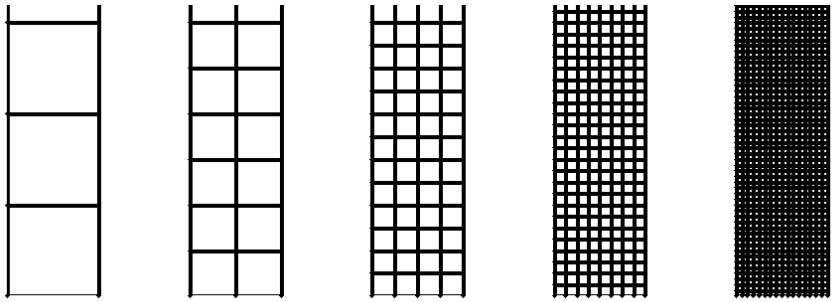


Figure 7.1: *From left to right: five Cartesian grids in order of refinement grade. Each refinement comprises twice as much cells in each direction. Only the outflow part of the grid is shown.*

A discharge Q equal to 2500 m^3/s is prescribed. The bottom slope i_b is prescribed to be 10^{-4}. The bottom level at the inflow boundary is 0 m (w.r.t. reference) and at the outflow boundary -1 m (w.r.t. reference). The Chézy friction coefficient C is set to 65 m$^{1/2}$/s. Given

the above values, the equilibrium water depth d_e is computed to be 3.89677 m:

$$d_e = \left(\frac{Q}{BC\sqrt{i_b}}\right)^{2/3} = \left(\frac{2500}{500 \cdot 65 \cdot \sqrt{10^{-4}}}\right)^{2/3} = 3.89677\text{m}. \tag{7.6}$$

In order to properly impose an outflow boundary that is grid independent, a Neumann-boundary is prescribed at the outflow: $\partial h/\partial x_n = -i_b$, with h the water level. The computational time step is set automatically being restricted by a CFL limit value equal to 0.5. Upstream, the keyword `jbasqbnddownwindhs` is set to `1` (relevant for the inflow boundary), whereas `izbndpos = 0` (relevant for the outflow boundary). For the bed friction, the option `Conveyance2D = 3` is set (i.e. an analytic expression is used for the hydraulic radius).

Results

First, recall the quantity `hu`, which represents the upstream water level (cell center) at the location of a velocity point (cell face). This quantity is used as the water depth within the computation of the bed friction at a velocity point. As a result, the water depth equals the water level at the upstream *cell center* minus the bed level at the *cell face*.

Hence, if a water depth is to be computed as part of the postprocessing, this water depth should be computed in analogy, namely as the difference between the bed level at a velocity point and the upstream water level. This value for `hu` should then be assessed against the backdrop of the equilibrium depth d_e (see Equation (7.5)) being equal to 3.89677 m, in this case.

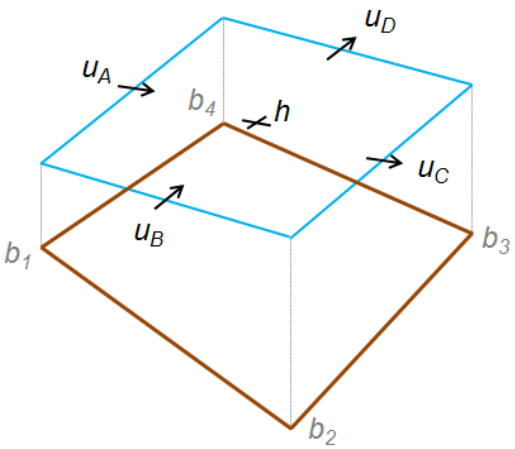


Figure 7.2: *Computational cell with surface level h, face velocities u_A, u_B, u_C and u_D and bed levels b_1, b_2, b_3 and b_4.*

The way of postprocessing the water depth can further be clarified by considering an arbitrary computational cell, as shown in Figure 7.2. Recall that the principal variables are the water level (located at the cell center) and the face-normal velocities (located at the center of the faces). The level of the bed is given in the cell corners. The value of `hu` at the location of, for instance, velocity u_C is then computed as $h-(b_2+b_3)/2$ (in case of `bedlevtyp = 3`; the options `4` and `5` take $\min(b_2, b_3)$ and $\max(b_2, b_3)$, respectively, to be subtracted from the water level). However, choosing for `Conveyance2D = 3` implies an approach according to `bedlevtyp = 4`.

Since the analytical solution is known for this flow situation, the deviations of the computed results from the analytical results can be measured exactly. As a measure, the L_2-norm of the

residual Res (i.e. the difference between the analytical and the numerical solution) is taken, defined as

$$\mathrm{Res}_{L_2} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\mathrm{Res}_i^2} \tag{7.7}$$

with N the number of evaluated grid locations. Differences with the analytical solution for the five different grids are shown in Figure 7.3 through this measure.

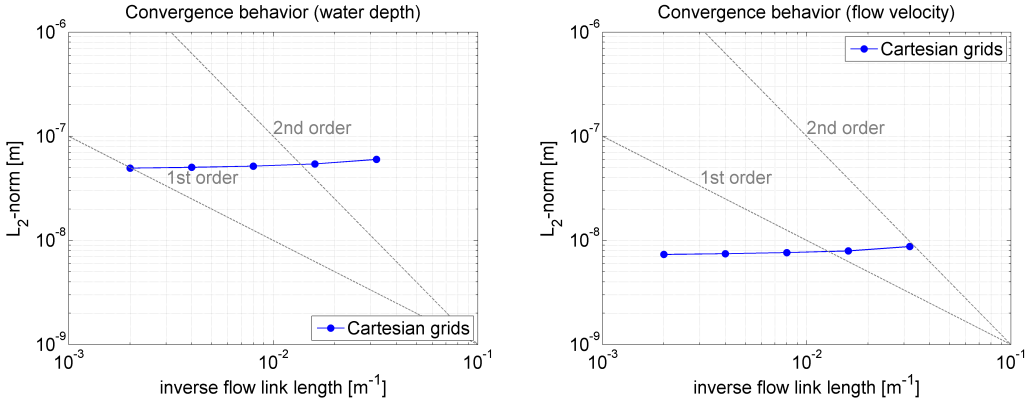


Figure 7.3: *Rate of convergence, measured by means of the L_2-norm using the analytical solution and the computational output. Left panel: convergence of the computed water depth; right panel: convergence of the computed velocity.*

Figure 7.3 reveals that the actual differences between the computational and analytical outcomes are very small (both for the water levels and the velocities). However, the rate of the convergence tends to be of order 0 (both for the water levels and the velocities).

Conclusion

For the two-dimensional, stationary, homogeneous and uniform channel flow simulations, the following conclusions can be drawn:

1. on all the five grids (of different grade of refinement), the computational outcomes approximate the analytical outcomes very closely, both regarding the water levels and the velocities,
2. the differences between the computational and analytical outcomes converge at rate 0 with decreasing cell size.

Version

This test has been carried out with version dflow-fm-x64-1.1.116.36629.

7.3 Refinement study for channel flow using triangular grids

Purpose

The purpose of this validation case is to examine the performance of D-Flow FM for the simulation of a schematized uniform and homogeneous channel flow. This validation case focuses on the effects of the refinement of a grid with only triangular cells. The topography comprises a linearly varying bathymetry with a constant slope.

Linked claims

Claims that are related to the current test case are:

◇ claim 2.3.1.1: D-Flow FM can accurately simulate steady and unsteady hydrostatic flow on grids consisting of triangles, quadrilaterals, pentagons and hexagons

Approach

To enable academic comparison of the computational output with a known analytical solution, the topography of the bed varies linearly with a slope i_b. Given a discharge Q and a friction coefficient (in this case, Chézy's factor C is used), an equilibrium water depth d_e can be computed as:

$$d_e = \left(\frac{Q}{BC\sqrt{i_b}} \right)^{2/3} . \tag{7.8}$$

If the grid is refined several times, the order of accuracy of the numerical integration routines can be assessed. For this purpose, only triangular grids are considered (Cartesian grids are considered in case e02-f01-c010).

Model description

For this test case, trhee computational grids are generated. The outflow parts of these grids are shown in Figure 7.4. The three grids are of triangular type. The longitudal size L of the domain is 10 km, whereas the lateral size B of the domain is 500 m.

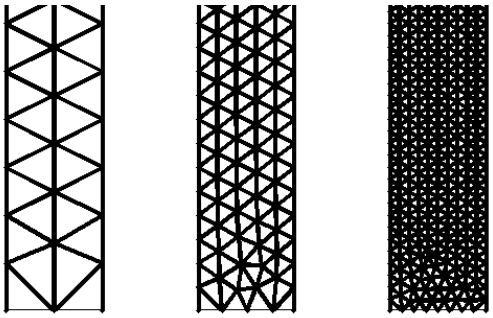


Figure 7.4: *From left to right: three triangular grids in order of refinement grade. Each refinement comprises twice as much cells in each direction. Only the outflow part of the grid is shown.*

A discharge Q equal to 2500 m^3/s is prescribed. The bottom slope i_b is prescribed to be 10^{-4}. The bottom level at the inflow boundary is 0 m (w.r.t. reference) and at the outflow boundary -1 m (w.r.t. reference). The Chézy friction coefficient C is set to 65 m$^{1/2}$/s. Given

the above values, the equilibrium water depth d_e is computed to be 3.89677 m:

$$d_e = \left(\frac{Q}{BC\sqrt{i_b}}\right)^{2/3} = \left(\frac{2500}{500 \cdot 65 \cdot \sqrt{10^{-4}}}\right)^{2/3} = 3.89677\text{m}. \tag{7.9}$$

In order to properly impose an outflow boundary that is grid independent, a Neumann-boundary is prescribed at the outflow: $\partial h/\partial x_n = -i_b$, with h the water level. The computational time step is set automatically being restricted by a CFL limit value equal to 0.5. Upstream, the keyword `jbasqbnddownwindhs` is set to `1` (relevant for the inflow boundary), whereas `izbndpos = 0` (relevant for the outflow boundary). For the bed friction, the option `Conveyance2D = 3` is set (i.e. an analytic expression is used for the hydraulic radius).

Results

First, recall the quantity `hu`, which represents the upstream water level (cell center) at the location of a velocity point (cell face). This quantity is used as the water depth within the computation of the bed friction at a velocity point. As a result, the water depth equals the water level at the upstream *cell center* minus the bed level at the *cell face*.

Hence, if a water depth is to be computed as part of the postprocessing, this water depth should be computed in analogy, namely as the difference between the bed level at a velocity point and the upstream water level. This value for `hu` should then be assessed against the backdrop of the equilibrium depth d_e (see Equation (7.8)) being equal to 3.89677 m, in this case.

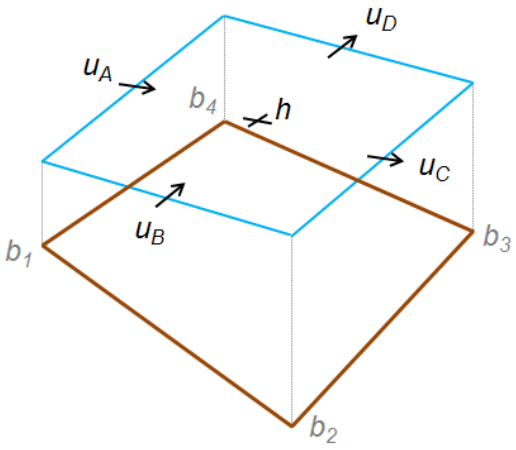


Figure 7.5: *Computational cell with surface level h, face velocities u_A, u_B, u_C and u_D and bed levels b_1, b_2, b_3 and b_4.*

The way of postprocessing the water depth can further be clarified by considering an arbitrary computational cell, as shown in Figure 7.5. Recall that the principal variables are the water level (located at the cell center) and the face-normal velocities (located at the center of the faces). The level of the bed is given in the cell corners. The value of `hu` at the location of, for instance, velocity u_C is then computed as $h-(b_2+b_3)/2$ (in case of `bedlevtyp = 3`; the options `4` and `5` take $\min(b_2, b_3)$ and $\max(b_2, b_3)$, respectively, to be subtracted from the water level). However, choosing for `Conveyance2D = 3` implies an approach according to `bedlevtyp = 4`.

Since the analytical solution is known for this flow situation, the deviations of the computed results from the analytical results can be measured exactly. As a measure, the L_2-norm of the

residual Res (i.e. the difference between the analytical and the numerical solution) is taken, defined as

$$\mathrm{Res}_{L_2} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\mathrm{Res}_i^2} \tag{7.10}$$

with N the number of evaluated grid locations. Differences with the analytical solution for the five different grids are shown in Figure 7.6 through this measure.

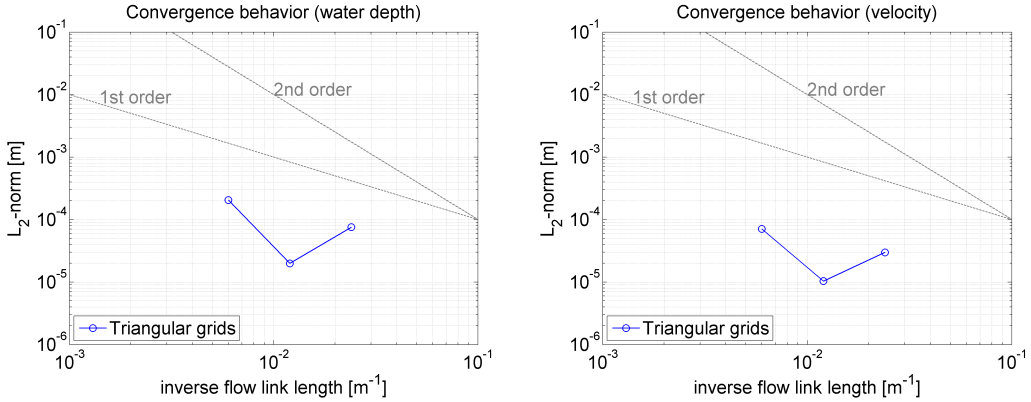


Figure 7.6: *Rate of convergence, measured by means of the L_2-norm using the analytical solution and the computational output. Left panel: convergence of the computed water depth; right panel: convergence of the computed velocity.*

Figure 7.6 reveals that the actual differences between the computational and analytical outcomes are significantly larger compared to the ones found for the Cartesian grids (both for the water levels and the velocities, see the documentation for case e02-f01-c010). The residual is assessed with the entire domain taken into account. Morever, the deviations from the analytical solution appear not to converge (both for the water levels and the velocities). The same has appeared to hold for the medium part of the domain, intending to leave potential boundary effects out of consideration.

Conclusion

For the two-dimensional, stationary, homogeneous and uniform channel flow simulations, the following conclusions can be drawn:

1. on all the three grids (of different grade of refinement), the computational outcomes approximate the analytical outcomes with an accuracy being significantly coarse compared to the simulations on Cartesian grids, both regarding the water levels and the velocities,
2. the differences between the computational and analytical outcomes appear not to converge.

Version

This test has been carried out with version dflow-fm-x64-1.1.116.36629.

7.4 Channel with curvilinear grid distortion

Purpose

The purpose of this validation study is to investigate the effect of a misaligned grid, i.e. a grid that makes an angle with the flow direction, on the accuracy of the model results. This validation study is directly derived from the validation document for Delft3D.

Linked claims

Claims that are related to the current test case are:

◇ claim 2.3.1.1: D-Flow FM can accurately simulate steady and unsteady hydrostatic flow on grids consisting of triangles, quadrilaterals, pentagons and hexagons

Approach

A uniform flow in a simple straight channel over a sloping topography is considered. In the steady state, the slope of the water level should equal the slope of the bed. In this test case, it is examined to what extent the numerical solution is dependent on the grid it is computed on. For this purpose, the outcomes obtained on a Cartesian grid are compared with the outcomes obtained on a sinusoidally distorted grid. This sinusoidally distorted grid is shown in Figure 7.7. A computation on an equivalent, strictly Cartesian grid of $M \times N = 10 \times 80$ cells is carried out for comparison.

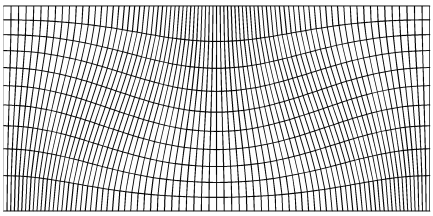


Figure 7.7: *The computational grid for curvilinearly distorted grid case. The bed level ranges from -4 m (left) to -4.5 m (right) w.r.t. the reference level. The flow is from left to right.*

To enable academic comparison of the computational output with a known analytical solution, the slope of the bed is set varying linearly with a slope i_b. Given a discharge Q and a friction coefficient (in this case, Chézy's factor C is used), an equilibrium water depth d_e can be computed as:

$$d_e = \left(\frac{Q}{BC\sqrt{i_b}} \right)^{2/3}. \tag{7.11}$$

Model description

The settings for the computation are (taken from the Delft3D validation document):

◇ length $L = 500$ m and width $B = 240$ m,
◇ bed slope $i_b = 10^{-3}$,
◇ inflow velocity $U = 2.933654$ m/s or inflow discharge $Q = 1434.2047$ m^3/s,
◇ Chézy's coefficient $C = 65$ m$^{1/2}$/s.

With this parameters, the equilibrium depth is $d_e = 2.037$ m. Including the $\Delta x/2$ distance between the outflow boundary and the cell centers of the ghost cells, the output water level is

set equal to:

$$\begin{aligned} h_{out} &= \text{bed level} + \text{equilibrium depth} - i_b \cdot \Delta x/2 \\ &= -4.5 + 2.037 - 10^{-3} \cdot 500/80/2 \\ &= -2.466125 \text{ m+NAP} \end{aligned}$$

Hence, the output water level is set equal to -2.466125 m w.r.t. the reference level. Two specific settings to be mentioned are:

1 the location of the outflow water levels is set on the cell center of the ghost cell outside the grid, being mirrored due to grid staggering (i.e. `izbndpos = 0`),
2 at the inflow boundary, no specific corrections are made for the interpolation of the water levels, based on the downwind water level (i.e. `jbasqbnddownwindhs = 0`).

The two computations are carried out twice: one time with a **velocity** boundary condition upstream, and one time with a **discharge** boundary condition upstream.

Results

Figure 7.8 shows the water depth at the end of the simulation time in case a **velocity** condition is applied at the inflow boundary. One can see that deviations from the analytical equilibrium depth are present. The closer to the outflow boundary, the more the depth approaches the analytical equilibrium depth of 2.037 m.

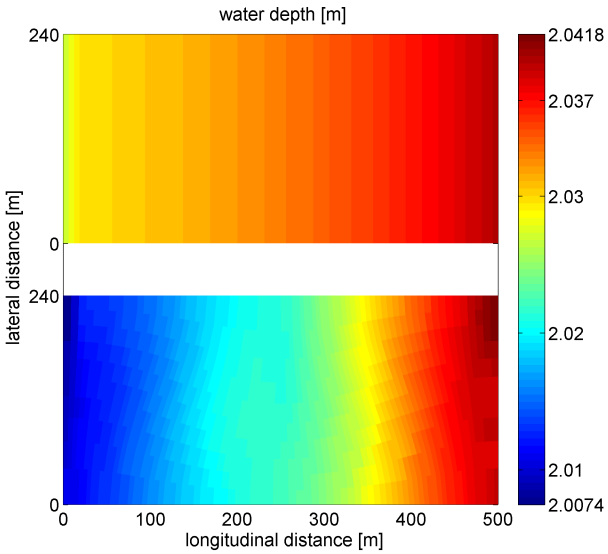


Figure 7.8: *The water depth at the end of the simulation time (steady state solution). Upper panel: results on the Cartesian grid; lower panel: results on the curvilinearly distorted grid. For this result, a **velocity** condition is applied at the inflow boundary. The analytical equilibrium depth is 2.037 meters.*

For this case with curvilinear distortion of the grid, the deviations from the analytical solution are significantly larger for the distorted grid compared to the rectangular grid. It should, moreover, be remarked that this two-dimensional case does not attain the same accuracy for this type of case as the one-dimensional case does (cf. case e02-f05-c020).

Figure 7.9 shows the water depth at the end of the simulation time in case a **discharge** condition is applied at the inflow boundary.

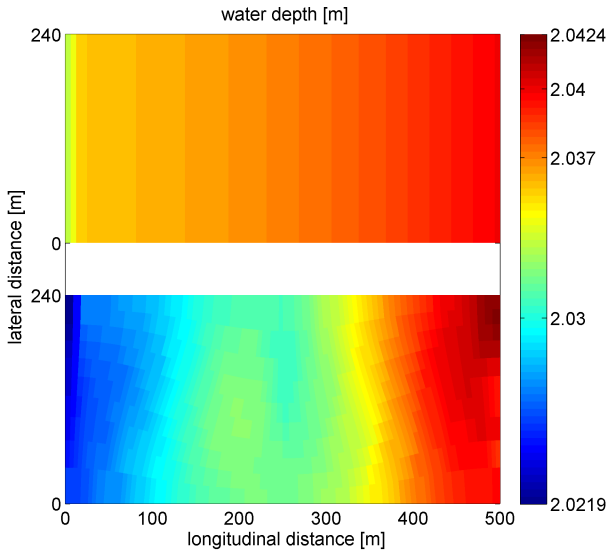


Figure 7.9: *The water depth at the end of the simulation time (steady state solution). Upper panel: results on the Cartesian grid; lower panel: results on the curvilinearly distorted grid. For this result, a* ***discharge*** *condition is applied at the inflow boundary. The analytical equilibrium depth is 2.037 meters.*

As Figure 7.9 shows, the deviations in the outflow region are quite comparable with their equivalents from Figure 7.8. However, it is seen that the deviations become smaller the more the inflow boundary is approached. From this it could be concluded that the discharge inflow boundary condition is favorable over the velocity inflow boundary condition.

Remark that the timestep Δt is computed automatically, based on a CFL limit value of 0.7. If `jbasqbnddownwindhs = 1` is chosen in the mdu-file, the computation on the curvilinearly distorted grid starts wiggling.

Conclusions

Misalignment of the grid has a clear effect on the accuracy of the model results. For the curvilinearly distorted grid, the maximum deviation from the analytical solution for the equilibrium depth is significantly larger compared to the equivalent Cartesian grid case.

Version

This test has been carried out with version dflow-fm-x64-1.1.116.36629.

8 Wind

8.1 Sudden uniform wind forcing in a partially open basin

Purpose

Wind is present in many applications requiring simulation of surface waters. Rodney Sobey published a set of analytical solutions for the fluid flow in shallow-water situations with wind forcing in the paper Analytical Solutions for Storm Tide Codes in *Coastal Engineering* (2002, 46: 213-231). Here, the case "ST3: Sudden uniform wind on a partially open basin" has been selected for benchmarking. One more case from the same suite is presented in the following section.

Conveniently, analytical solutions enable us to know in advance the flow field that follows from specific initial conditions, boundary conditions and external forcing in a certain basin. Both the governing equations for fluid flow and the shape of the basin need simplification in order to provide exact solutions by means of analytical integration.

Upon reproducing the cases for which analytical solutions are available, the attending formulations and algorithms of D-Flow FM are subject to an objective, although not exhaustive, critical exercise.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.4.3: D-Flow FM is able to apply linearized friction in two-dimensional flow simulations
- ◇ claim 2.3.5.2: D-Flow FM can be used for an accurate prediction of wind driven flow

Approach

A basin with rectangular shape and uniform depth is considered. The reference level for vertical distances is the quote of the undisturbed water surface. Water levels are indicated as η. The basin depth is H. The wind blows parallel to one pair of sides, taken as the x-direction. Free mass exchange with external water is allowed at the upwind side (x =0), while the other boundaries are closed vertical walls. Depending on scale, such a basin can be seen as a crude representation of anything from a harbour to a marginal sea.

The equations for which the analytical solution is derived differ from the general shallow-water equations on the following counts:

- ◇ the advective accelerations are neglected;
- ◇ the Coriolis force is neglected;
- ◇ the viscosity effects are neglected;
- ◇ the surface elevations are small with respect to the undisturbed water depth ($\eta/H \ll 1$);
- ◇ bed friction is a linear function of the depth-averaged velocity;
- ◇ vertical walls exert no friction.

Under these assumptions, the governing equations are linearised; the motion is one-dimensional and always aligned with the wind direction; and, finally, the water body is in dynamical equilibrium, acted on by free-surface gradients, surface shear and bed friction.

Sobey develops his treatment in terms of the net mass flux in the x-direction per unit of

basin's width q (discharge per unit width, for short). Here, rather, the formalism is in terms of the depth-averaged velocity u, following the implementation of D-Flow FM. By virtue of the linearisation, the two quantities are related by the relationship $q = uH$, which holds between the solutions of q and H.

The following equations for the mass and x-momentum conservation are solved for:

$$\frac{\partial \eta}{\partial t} = -H\frac{\partial u}{\partial x} \tag{8.1}$$

$$\frac{\partial u}{\partial t} = -g\frac{\partial \eta}{\partial x} - \lambda u + \frac{\mathcal{F}_0}{H} \tag{8.2}$$

where λ is the linear friction coefficient and $\mathcal{F}_0 = \tau_s/\rho_w$ is a shorthand for the wind forcing term, with τ_s being the surface shear stress and ρ_w the water density.

Linear friction is used at times in ocean modelling, while a quadratic parametric law is used in shallow waters (fully-developed turbulent friction). Translating a linear friction coefficient into an equivalent roughness parameter for turbulent flows requires the estimation of some typical value for the flow velocity, $\hat{u}$. This scale is arbitrary and its choice depends on judgement. For the Manning friction law (with roughness coefficient m), the following correspondence holds:

$$\lambda = \frac{gm^2}{H^{\frac{7}{3}}}\hat{u} \tag{8.3}$$

Finally, the steady and uniform wind forcing could be seen as an approximation for an air stream getting to blow above the basin after a rapid sideward displacement, parallel with the sea entrance.

An exact solution is determined for initial conditions representing a body accelerating from rest:

$$\eta(x,0) = u(x,0) = \left.\frac{\partial \eta}{\partial t}\right|_{(x,0)} = 0 \quad , \quad \left.\frac{\partial u}{\partial t}\right|_{(x,0)} = \mathcal{F}_0 \, ;$$

for a constant water level at the upwind open boundary (x = 0):

$$\eta(0,t) = \left.\frac{\partial u}{\partial x}\right|_{(0,t)} = 0 \, ;$$

and for an impervious wall at the downwind boundary ($x = L$):

$$u(L,t) = 0 \quad , \quad \left.\frac{\partial \eta}{\partial x}\right|_{(L,t)} = \frac{\mathcal{F}_0}{C^2} \, .$$

The analytical solutions for η and u contain a steady-state solution plus a transient summation of standing waves (free modes) damped by bed friction:

$$\eta(x,t) = \frac{\mathcal{F}_0}{C^2}x + e^{-\frac{\lambda}{2}t}\sum_{n=1}^{\infty}\left(\cos\omega_n t + \frac{\lambda}{2}\frac{\sin\omega_n t}{\omega_n}\right)\hat{f}_n X_n(x) \tag{8.4}$$

$$u(x,t) = 0 + e^{-\frac{\lambda}{2}t}\frac{1}{H}\sum_{n=1}^{\infty}\frac{\sin\omega_n t}{\omega_n}\,\hat{g}_n\,Y_n(x) \tag{8.5}$$

where $C^2 = gH$ is the squared long-wave celerity. To compute the transient contributions in Equation (8.4) and Equation (8.5), the following space-dependent quantities need be determined from the basin length L for each oscillating mode n:

$$\begin{aligned} \beta_n &= \left(n - \frac{1}{2}\right)\frac{\pi}{L} \\ X_n &= \left(\frac{2}{L}\right)^{\frac{1}{2}} \sin \beta_n x \\ Y_n &= \left(\frac{2}{L}\right)^{\frac{1}{2}} \cos \beta_n x \\ \hat{f}_n &= -\frac{\mathcal{F}_0}{C^2}\int_0^L x X_n(x)\,dx \\ \hat{g}_n &= \mathcal{F}_0 \int_0^L Y_n(x)\,dx \end{aligned} \tag{8.6}$$

The corresponding angular velocities are then calculated with the long-wave dispersion relationship in a damped system:

$$\omega_n = \sqrt{\beta_n^2 C^2 - \left(\frac{\lambda}{2}\right)^2} \tag{8.7}$$

Only the combinations of L, H (via C) and λ giving a positive radicand for Equation (8.7) lead to the development of the transient oscillating phase. In particular, in Equation (8.7) one can recognize the period of inertial waves, $2\pi/\beta_n C$, and the $1/e$-decay time, $2/\lambda$, associated with exponential damping. The dispersion relationship, thus, indicates how the time scales of standing waves and of friction combine to produce the basin's response (e.g. how many oscillations can take place before the transient phase is damped down to the steady-state solution).

Lastly, the solution $\eta(x,t)$ scales with $\mathcal{F}_0 L/C^2 = \hat{\eta}$, the steady-state wind set-up at the downwind end. The solution of $u(x,t)$ scales with $\mathcal{F}_0 L/CH = \hat{u}$, the maximum depth-averaged velocity at the open boundary in case of frictionless oscillations. Locally, the corresponding scales at each position are

$$\hat{\eta}_x = \hat{\eta}\frac{x}{L} \quad , \quad \hat{u}_x = \hat{u}\frac{L-x}{L}\,, \tag{8.8}$$

which will be used later on to normalise the results.

In spite of the linearity requirements, the analytical solution describes a non-trivial behaviour of the water levels and depth-averaged velocities, with regular variations of sign and rates of change, that makes it well-suited for testing numerical codes.

A free surface tilted with constant slope and no net mass flux everywhere is the eventual steady-state balance (wind set-up). However, such a wind set-up arrangement can be preceded by a transient phase with oscillating water elevations and with alternate water exchange at the open boundary.

The wind onset is sudden. Its shearing action over the surface of the entire basin drives a current against the closed boundary downwind. Hence, the surge starts off as a heave of water at the downwind end. While the water keeps on piling up there, the raised waters increase in height and longitudinal extent upwind.

At the same time, these raising water levels cause an increasing free-surface slope that promotes a return flow in competition with the wind-driven flow. Once a net outflow is established at the upwind open boundary, emptying begins, which then favours the wind-driven surge once again.

This alternating prevalence of wind-shear and surface-slope forcing is damped by bed friction exponentially, until the steady wind set-up is reached. Interesting for guiding intuition, the basin's peak response to the wind forcing can occur in the transient phase.

These mechanisms are of fundamental nature and expectedly blueprint those occurring with non-linear physics at play, such as neglected terms (e.g. advection) or more complex parametrizations (e.g. turbulent bed friction).

Model description

Here Sobey's indications have been followed, unless otherwise stated. No sensitivity analysis to parameters range has been carried out. The domain has a length of L = 5000 m and a depth of H = 2 m. The width, immaterial for the one-dimensional flow pattern, has been set to W = 250 m.

When comparing the analytical and numerical solutions, care is needed in handling the staggered-grid arrangement implemented by D-Flow FM. In the solver's representation, open boundaries are drawn onto the cells faces, that is to say *velocity* grid points. The *water-level* grid points where the boundary conditions are applied, however, are actually located at 'ghost' stations half cell-size off the open boundary. Therefore, the analytical solution has effectively been worked out for a basin of effective length L = 5000 m + $\Delta x/2$, where 5000 m is the distance between the boundaries as a user would normally define it in D-Flow FM, while Δx is the cell size.

The atmospheric forcing is equal to $\mathcal{F}_0 = 1.5 \times 10^{-3}$ m^2/s^2. This can be obtained from the classic wind-drag parametrization with wind velocity of 25 m/s (Bft 10), a drag coefficient of C_d = 0.002 and air density ρ_a = 1.2 kg/m^3. These values are realistic. As a result, C = 4.429 m/s, $\hat{\eta}$ = 0.3832 m and $\hat{u}$ = 0.8487 m/s.

The linear friction coefficient is $\lambda = 10^{-4}$ s^{-1}. Based on Equation (8.3), using the above velocity scale $\hat{u}$, this λ corresponds to a Manning coefficient m = 0.0055 s/m$^{1/3}$, representing an unusually smooth bed. On the other hand, for the purpose of validation, this choice of λ appears to lead to all terms of the governing equations being equally relevant. On so doing, the assessment of the flow solver's attending formulations is more comprehensive than a stereotyped field case.

The initial conditions are a horizontal water level at η = 0 and quiescent water. Based on the relationship $t_{1/e} = 2/\lambda$, a simulation of 6 hours will include the amplitude exponential decay down to $1/e$, 37%, of the undamped value.

Both a Cartesian grid and a structured triangular grid are used to discretize the domain. The condition η = 0 is implemented at the upwind open boundary. The upwind parts of these grids are shown in Figure 8.1.

The cells in the Cartesian grid are squares with side of Δx = 25 m (user-defined). The shortest wavelength determinable on a grid is $2\Delta x$ (Nyquist criterion), whereby the upper-bound wave number is $\pi/\Delta x$. This restriction holds for reasons inherent to space discretization only. As a result, the flow solver can only resolve the transient components having wave number $\beta_n \leq \pi/\Delta x$. Since the solutions ((8.4)) and ((8.5)) contain infinite summations of standing

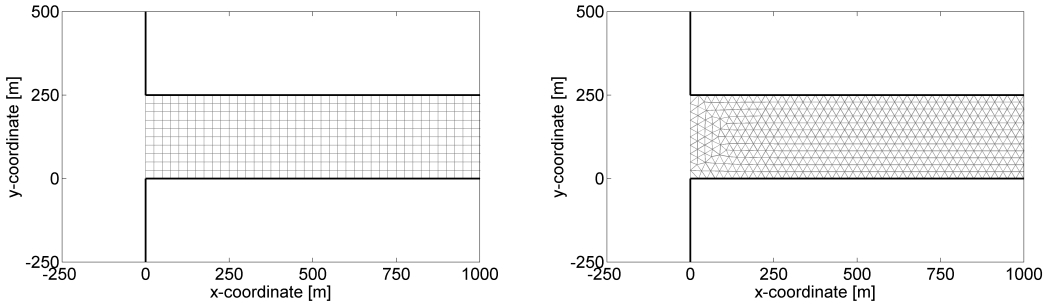


Figure 8.1: *Test case 'Sudden uniform forcing in a partially open basin'. Computational grids: Cartesian-type (left) and structured with triangular cells (right). Part view near the upwind open boundary: the domain has a closed end at x = 5000 m.*

waves, for a fair comparison, the number of modes in the exact solution can be limited to $n = n_x + 1$, where n_x is the number of cells in the direction of wave propagation.

Further, the resolved mode with the smallest wavelength $2\Delta x$ also dictates the smallest oscillation period that should be computed. Since ω_{n_x+1} = 0.5566 rad/s after the dispersion relationship ((8.7)), the fastest period of all resolvable components is T_{n_x+1} = 11.28 s. Again for the Nyquist criterion, a time step less than the half of this period suffices to resolve all temporal oscillations of interest. Here, Δt = 5 s.

Therefore, with this choice of grid spacing and time step, the transient phase in Equation (8.4) and Equation (8.5) is fully resolved in both space and time within the accuracy allowed by the domain discretization. It could be shown that, on so doing, the unity Courant number criterion is also fulfilled.

The cells in the triangular grid are equilateral with side of Δx = 25 m.

The numerical results are compared with the same evaluation of the analytical solution as the Cartesian grid. In order to linearise the governing equations, the following settings have been specified in the mdu file:

- `AdvecType = 0`
- `Lincontin = 1`
- `Umodlin = 1`
- `UnifFrictCoefLin =` $\langle \lambda H \rangle$

Additionally, an ad hoc modification was applied to the source code to linearise the wind forcing term by dividing the shear stress by H rather than by $\eta + H$. The following flags have been modified with respect to default settings:

- `Teta0 = 0.5`
- `CFLMax = 1`

in order to integrate the algebraic water-level gradients consistently with the acceleration term (through the implicitness factor θ) and to avoid automatic changes of the time step, respectively. The Technical Reference Manual provides further context to this information. The other

settings take default values.

Results

The simulated histories of water level and depth-averaged velocity have been compared with the analytical counterparts at selected observation points on the basin's middle longitudinal axis. The corresponding profiles along the basin at selected time instants are not shown, as they do not add essential information.

In the following plots, using Equation (8.8), the water levels have been normalised with $\hat{\eta}_x$, the expected local elevation at steady state, and the depth-averaged velocities with $\hat{u}_x$, the expected local maximum velocity of frictionless oscillations. Seen that the flow field is spatially variable, the results are thus presented in proportion to the expected local values.

Each plot displays the exact values (in red), the computed value (in blue or green, dashed). The deviation of the latter from the former (in black) is then plotted with a secondary axis. Dashed red lines support the interpretation, showing the steady-state situation and the exponential damping function of Equation (8.4) and Equation (8.5). The results for the Cartesian and triangular grid are given in separate columns. For ease of interpretation with regard to the staggered-grid arrangement, from here onwards x denotes the co-ordinate value that a D-Flow FM user would normally deal with, as in Figure 8.1.

Water levels are shown in Figure 8.2 for three stations at mid length (x = 2512.5 m, $\hat{\eta}_x$ = 0.1930 m), three-quarter length (x = 3762.5 m, $\hat{\eta}_x$ = 0.2866 m) and at the basin's closed end (x = 5000 m, $\hat{\eta}_x$ = 0.3832 m). These observation points are located in the downwind half of the basin, where water levels are the largest.

The water level history is a damped square wave, where periods of rapid change alternate with periods of almost horizontal level. Over each cycle, at each location, the final water elevation and the amplitude of the ensuing cycle decrease exponentially, eventually tending to the steady-state wind set-up.

As for the hydrodynamic processes as seen at fixed stations, rapid rise indicates the arrival of the water already piled up by the wind against the downwind end, as the heave grows and moves upwind. There, in the net, wind-driven flux and return flux cause convergence in the water column. Rapid fall indicates the onset of a flux-divergence situation as the return flow, driven by the surface slope, prevails over the wind-driven flow, causing emptying and lowering of the water column. The nearly flat intervals are a reflection of a time lag while these waves of change travel from the downwind end to the location in point with celerity C. Consistently, the water-level history has a saw-tooth pattern at the downwind end itself.

The agreement between the numerical and analytical solutions is such that the two curves can hardly be distinguished. The deviations are well less than 2% of the local scales for both the Cartesian grid and the triangular grid at each station. The box-and-whisker plots of Figure 8.3 show the magnitude of the deviations occurring during the simulations. The red stripes indicate the median value, the upper and lower ends of the box delimit the values capping 25% and 75% of the data points, and the whiskers mark the entire value range. The deviations magnitude is in the order of millimetres, for both grids at each observation point.

The largest discrepancies closely correspond to changes in the flow development, as described above, and are quickly reduced, repeating a regular pattern specific to location and grid type. In all cases, the deviations show an oscillating value with no noticeable trend and remain bounded for more than 4000 time steps.

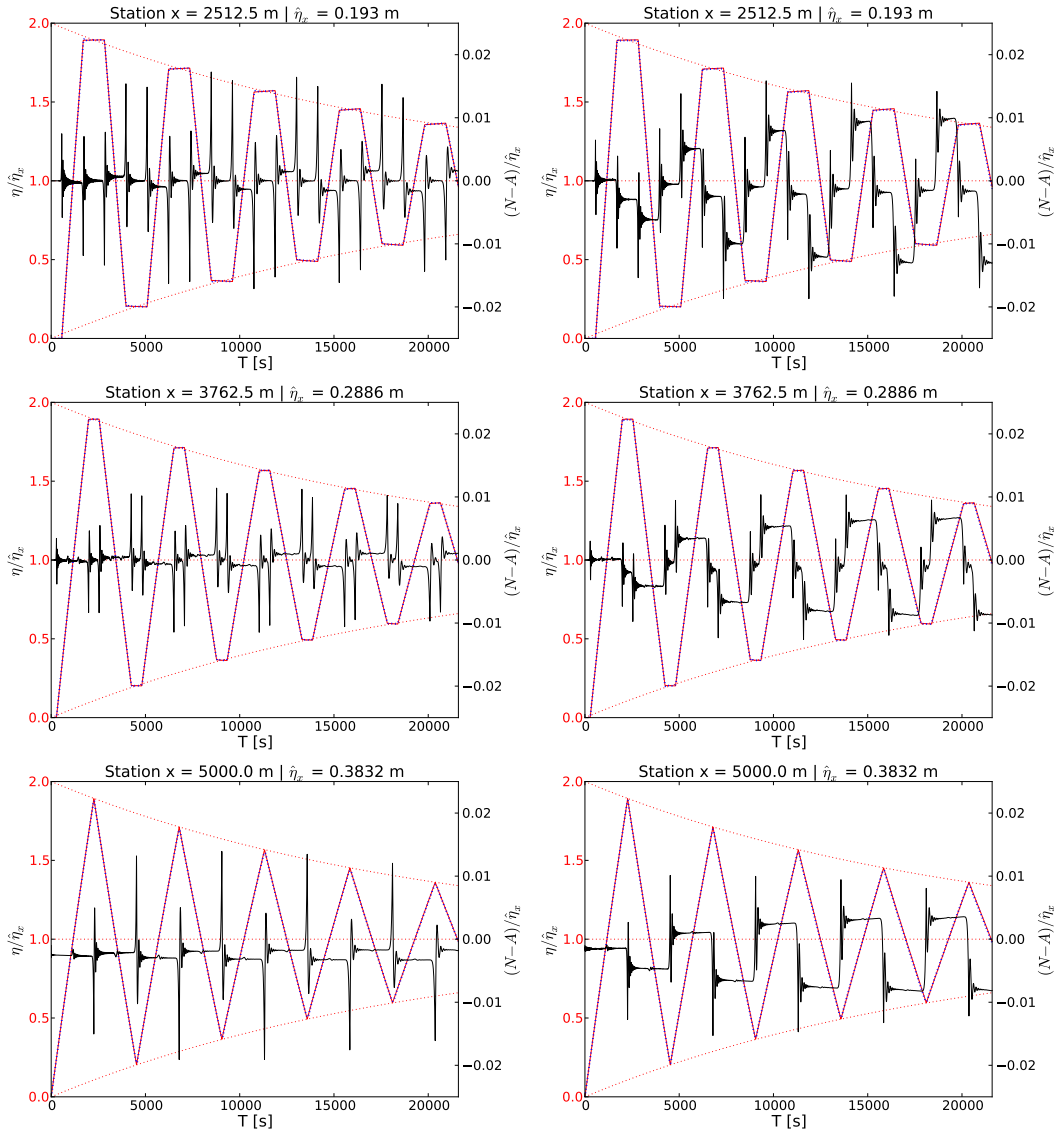


Figure 8.2: *Water levels. History of normalised values at mid (top), three-quarter (middle) and full basin length (bottom). Cartesian grid (left), triangular grid (right). Primary y-axis: analytical solution (A, red); D-Flow FM solution (N, blue dashed). Secondary y-axis: residuals (N-A, black). Dashed red lines: steady-state wind set-up at location and exponential decay law.*

The equivalent plots for the depth-averaged velocity are shown in Figure 8.4, for three stations at the open boundary (x = 0 m, $\hat{u}_x$ = 0.8466 m/s) and at one-quarter (x = 1250 m, $\hat{u}_x$ = 0.6350 m/s) and one-half basin length (x = 2500 m, $\hat{u}_x$ = 0.4233 m/s). These observation points are located in the upwind half of the basin, where the flow velocities are the largest. Positive values indicate flow landwards, negative seawards.

The exact solution for the depth-averaged velocity has the same pattern as the water levels in a mirror-like fashion, with a saw-tooth pattern at the open boundary (representing phases of uniformly accelerating and retarded flow) and damped square waves in the interior domain. The effect of bed friction is clearly seen from the exponentially decreasing values.

Here too, numerical and analytical results are in close agreement, with deviations from the local scales contained in the $\pm$2% range. The magnitude of the deviations observed in the

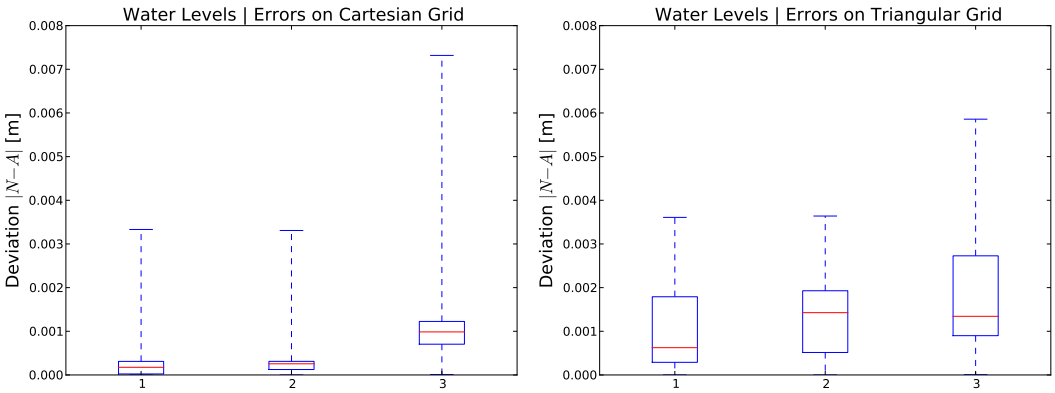


Figure 8.3: *Water levels. Deviations in absolute value between numerical and analytical histories in Figure 8.2. Cartesian grid (left), triangular grid (right). Station 1: mid basin. Station 2: three-quarter basin. Station 3: full basin length.*

simulation is summarized in Figure 8.5 in the box-and-whisker plots. The deviations are in the order of mm/s in most cases. Only the deviations at the open boundary station in the triangular grid are in the order of cm/s, with a bias towards underestimation. This peculiarity is plausibly caused by the more complex cell arrangement in that area (Figure 8.1) and has negligible impact at the downstream locations.

Conclusion

This test concerned the temporal and spatial variations of water levels and depth-averaged velocities in a partially-open rectangular basin with uniform depth, acted on by a steady, uniform and suddenly applied wind. The resulting flow is one-dimensional.

This test case is a partial but objective verification of the D-Flow FM skills for computing storm surges. Both partiality and objectivity reside in the analytical solution, whose existence requires a simplification of the complete shallow-water governing equations. However, once D-Flow FM is set to solve the same simplified equations, the comparison between the exact and simulated flow quantities assesses whether the computations are correctly devised and implemented.

In the governing equations, the inertial acceleration and surface gradient terms have been considered in the general form. In line with the assumption of small water-level changes with respect to the undisturbed water depth, the flux-convergence term in the continuity equation and the wind-shear term in the momentum equations have been linearised based on scaling arguments. Bed friction has been linearised by means of a first-order, laminar-flow type parametrization of the shear stress. Finally, advection, horizontal viscous friction and the Coriolis force have been neglected altogether.

The terms retained after these simplifications are, or closely represent, driving mechanisms essential to the storm-surge generation process. D-Flow FM has been adapted to solve the linearised equations by means of available input keywords in the driver file and of one modification to the source code of little generality. The time step has been chosen to resolve the flow field down to the temporal resolution consequent on the user-defined grid spacing.

Moreover, the numerical treatment of the simplified equations still couples the discrete continuity and momentum equations one to another upon performing the same algorithms as the general formulation. In spite of specifics, the flow solver still aims to work out water levels

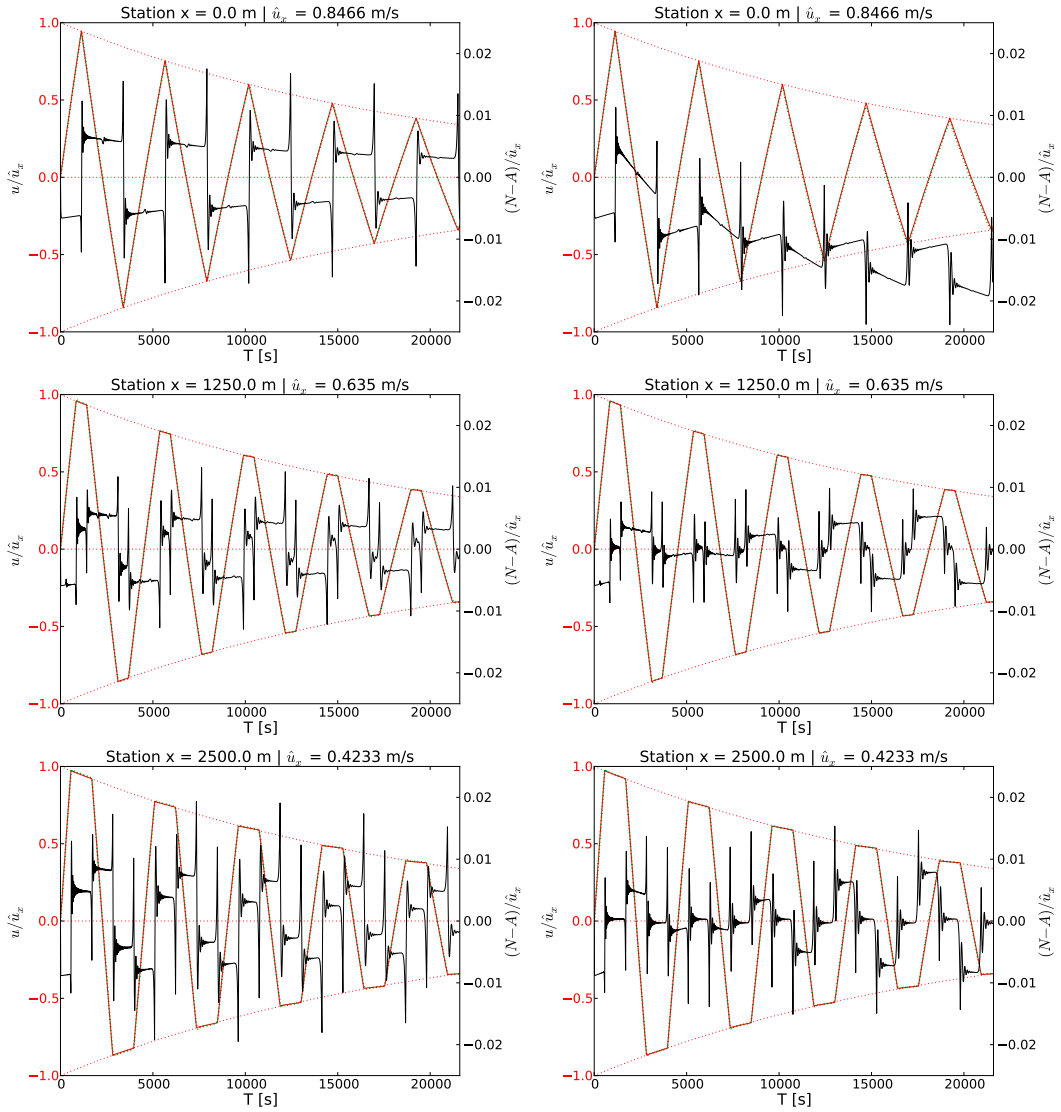


Figure 8.4: *Depth-averaged velocity. History of normalised values at open boundary (top), one-quarter (middle) and mid basin length (bottom). Cartesian grid (left), triangular grid (right). Primary y-axis: exact solution (A, red); D-Flow FM solution (N, green dashed). Secondary y-axis: deviation (N-A, black). Dashed red lines: steady-state rest and exponential decay law.*

and velocities that are nearly mass- and momentum-conserving at each grid location and time level, as a result of the evolving inner flow field and of the (time-steady) initial conditions, boundary conditions and external forcing. Therefore, a central part of the coding in D-Flow FM is put to the test.

In summary, the agreement of the time-varying water levels and depth-averaged velocities at selected stations in the basin is very high, whether the basin is represented with a mesh of rectangular or triangular elements. The deviations are consistently small, in both absolute and relative sense, and show a bounded, periodic behaviour in computations of the order of 4000 time steps. Such deviations do not taint the flow prediction with spurious effects of practical significance.

The flow at the open boundary is solved accurately upon prescribing a fixed water level alone.

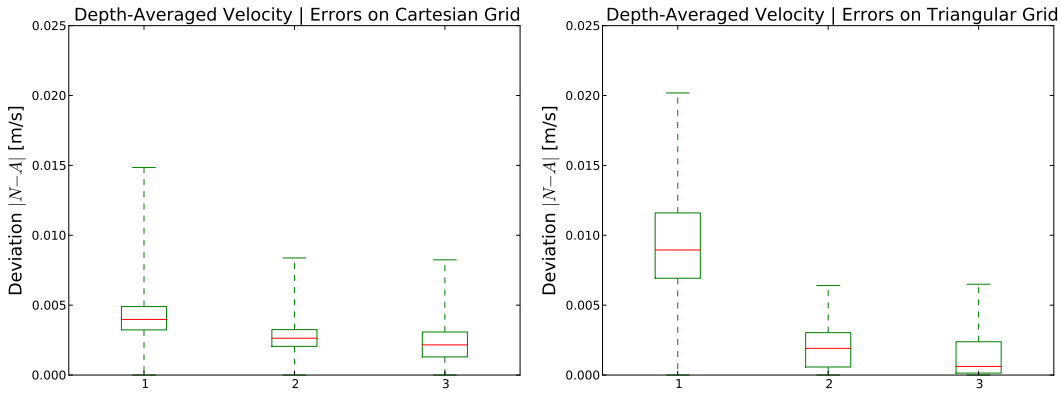


Figure 8.5: *Depth-averaged velocity. Deviations in absolute value between numerical and analytical histories in Figure 8.4. Cartesian grid (left), triangular grid (right). Station 1: open boundary. Station 2: basin first quarter. Station 3: mid basin.*

The standing-wave contributions that constitute the transient flow are handled correctly without causing spurious reflection in the inner domain.

In view of the necessary assumptions, anticipated behaviours and chosen settings, D-Flow FM appears to compute the expected flow field with highly accurate and stable results on a Cartesian and a structured triangular grid. Subject to the additional consideration of non-linear processes, whether neglected or simplified here, D-Flow FM is expected to correctly simulate the long-wave component in storm-surge situations acted on by steady and uniform winds.

Acknowledgements

This validation case has been conducted and commented by Giordano Lipari of Waterbeweging | Watermotion as a pilot user of D-Flow FM. Neither monetary nor non-monetary requital have been awarded for this work. Wim van Balen of Deltares provided the grids and has controlled the quality of results and conclusions.

Version

This test has been carried out with D-Flow FM version 1.1.116-36532.

8.2 Travelling wind pulse in a partially open basin

Purpose

This validation exercise implements the test "ST4: Moving pulse across a partially open basin" published by Rodney Sobey in the paper Analytical Solutions for Storm Tide Codes in *Coastal Engineering* (2002, 46: 213-231). The paper presents a set of analytical solutions for the fluid flow in shallow-water situations with wind forcing.

This section is companion to the previous one that considered a sudden uniform wind. This case rather addresses the effect of a wind pulse of limited spatial extent that travels over the basin. A wind pulse could be seen as an approximation for short-lived intensifications of the air stream, such as those caused by squall lines. Much of the explanation presented here draws from the companion section. Shared features are only repeated concisely for ease of reading.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.4.3: D-Flow FM is able to apply linearized friction in two-dimensional flow simulations
- ◇ claim 2.3.5.2: D-Flow FM can be used for an accurate prediction of wind driven flow

Approach

The basin has rectangular shape and uniform depth, crudely representing anything from a harbour to a marginal sea. The quote of the undisturbed water surface is the reference level for vertical positions. Water levels are indicated as η, the basin depth with H. The wind pulse blows parallel to one pair of sides, taken as the x-direction. The upwind side of the basin is open to a larger external water basin (at x = 0), while the other boundaries are closed vertical walls.

The wind pulse is represented by a stretch of length inside which the surface shear on the water increases sharply from the background zero value to a uniform value τ_s. The longitudinal extent of the pulse region is l, unchanging in time. The wind pulse has forward speed V and the head of pulse enters the basin at $t = 0$. Therefore, the wind forcing term acting on each water column is

$$\mathcal{F}(x,t) = \mathcal{F}_0 \left[\mathcal{H}\left(t - \frac{x}{V}\right) - \mathcal{H}\left(t - \frac{l+x}{V}\right) \right] \tag{8.9}$$

where $\mathcal{F}_0 = \tau_s/\rho_w$ is the surface shear inside the pulse, normalized with the water density ρ_w, while the Heaviside functions of time

$$\mathcal{H}(\tau) = \begin{cases} 1 & \tau \geq 0 \\ 0 & \tau < 0 \end{cases}$$

describe a step change from 0 to 1 at time τ.

The simplifications to the general shallow-water equations to achieve an analytical solution consist of neglecting advective accelerations, the Coriolis force and the horizontal effects of viscosity. Furthermore, it must be assumed that the surface elevations are small with respect to the undisturbed water depth ($\eta/H \ll 1$), that bed friction is a linear function of the depth-averaged velocity and that vertical walls exert no friction.

As a result, indicating the depth-averaged velocity as u, the following equations for the mass and x-momentum conservation are solved for:

$$\frac{\partial \eta}{\partial t} = -H\frac{\partial u}{\partial x} \tag{8.10}$$

$$\frac{\partial u}{\partial t} = -g\frac{\partial \eta}{\partial x} - \lambda u + \frac{\mathcal{F}}{H} \tag{8.11}$$

where λ is the linear friction coefficient and $\mathcal{F}$ is the wind forcing term of Equation (8.9).

The exact solution is elaborated for initial conditions representing a body at rest:

$$\eta(x,0) = u(x,0) = \left.\frac{\partial \eta}{\partial t}\right|_{(x,0)} = \left.\frac{\partial u}{\partial t}\right|_{(x,0)} = 0\,;$$

for a constant water level at the upwind open boundary (x = 0):

$$\eta(0,t) = \left.\frac{\partial u}{\partial x}\right|_{(0,t)} = 0\,;$$

and for an impervious wall at the downwind boundary ($x = L$):

$$u(L,t) = 0 \quad , \quad \left.\frac{\partial \eta}{\partial x}\right|_{(L,t)} = \frac{\mathcal{F}}{C^2}\left[\mathcal{H}\left(t - \frac{L}{V}\right) - \mathcal{H}\left(t - \frac{l+L}{V}\right)\right] ,$$

where $C^2 = gH$ is the squared long-wave celerity.

The analytical solutions for η and u are given by an infinite summation of modes. These read:

$$\eta(x,t) = \sum_{n=1}^{\infty} X_n(x) \int_0^t e^{-\frac{\lambda}{2}(t-\tau)} \frac{\sin[\omega_n(t-\tau)]}{\omega_n} \left[A_n(\tau) + B_n(\tau)\right] d\tau \tag{8.12}$$

$$u(x,t) = \frac{1}{H}\sum_{n=1}^{\infty} Y_n(x) \int_0^t e^{-\frac{\lambda}{2}(t-\tau)} \sin[\omega_n(t-\tau)]\, C_n(\tau)\, d\tau \tag{8.13}$$

The space-dependent functions in Equation (8.12) and Equation (8.13) are defined from the basin length L through:

$$\beta_n = \left(n - \frac{1}{2}\right)\frac{\pi}{L} \tag{8.14}$$

$$X_n = \left(\frac{2}{L}\right)^{\frac{1}{2}} \sin \beta_n x$$

$$Y_n = \left(\frac{2}{L}\right)^{\frac{1}{2}} \cos \beta_n x$$

The time-dependent parts in Equation (8.12) and Equation (8.13) are given by convolution integrals representing the memory effect of the wind-pulse passage. Their determination requires knowing firstly the angular velocities ω_n, which can be computed from the long-wave dispersion relationship in a damped system:

$$\omega_n = \sqrt{\beta_n^2 C^2 - \left(\frac{\lambda}{2}\right)^2} , \tag{8.15}$$

and, secondly, evaluating the functions:

$$A_n(t) = \mathcal{F}_0 \left(\frac{2}{L}\right)^{\frac{1}{2}} \{[\mathcal{H}_1(t) - \mathcal{H}_2(t)] \sin(\beta_n V t) - [\mathcal{H}_3(t) - \mathcal{H}_4(t)] \sin[\beta_n (V t - l)]\} \quad (8.16)$$

$$B_n(t) = \mathcal{F}_0 \left(\frac{2}{L}\right)^{\frac{1}{2}} \{-(-1)^n [\mathcal{H}_2(t) - \mathcal{H}_4(t)]\} \quad (8.17)$$

$$C_n(t) = \mathcal{F}_0 V \left(\frac{2}{L}\right)^{\frac{1}{2}} \{[\mathcal{H}_1(t) - \mathcal{H}_2(t)] \cos(\beta_n V t) - [\mathcal{H}_3(t) - \mathcal{H}_4(t)] \cos[\beta_n (V t - l)]\} \quad (8.18)$$

The following shorthand notations have been used for the Heaviside functions:

$$\mathcal{H}_1(t) = \mathcal{H}(t)\,, \mathcal{H}_2(t) = \mathcal{H}(t - \frac{L}{V})\,, \mathcal{H}_3(t) = \mathcal{H}(t - \frac{l}{V})\,, \mathcal{H}_4(t) = \mathcal{H}(t - \frac{l+L}{V})$$

which signal important moments of change such as the present time, the time to cover the basin length at speed V, the times when the pulse has entered the basin in full and has left the basin behind, respectively.

Like in the case of the sudden uniform wind, the dispersion relationship ((8.15)) indicates how inertial oscillations and friction combine and produce the basin's response (e.g. how effectively friction hampers the changes in free-surface elevation). In addition, the functions ((8.16)), ((8.17)), ((8.18)) describe a complex time development associated with the lasting effect of the wind-pulse transit.

Lastly, the solution $\eta(x,t)$ scales with $\mathcal{F}_0 L/C^2 = \hat{\eta}$. The solution $u(x,t)$ scales with $\mathcal{F}_0 LV/C^2 H = \hat{u}$, which is V/C times the same scale in the sudden uniform wind. Unlike the sudden uniform wind, these scales only anticipate the order of magnitude of the solution, since the steady-state condition is merely the return to a quiescent basin here. They will be used later on to normalise the results.

In terms of hydrodynamic processes on the basin as a whole, as the wind pulse enters the basin, the water columns underneath are sheared and raised while the flow begins to move leeward. The flow response, then, depends on the comparison between the pulse's forward speed and the basin's long-wave celerity (in essence, its depth).

In the case worked out here, the wind pulse travels with a velocity that is very close to the long-wave celerity. Hence, the free-surface disturbance is a bulge of water that travels along the basin without spreading too far away from the pulse region overhead, bearing some resemblance with a solitary progressive wave. When such raised waters reach the basin's end while the wind pulse leaves the basin behind, the sudden piling-up against the wall occurs and the deformed free surface is reflected.

The surface slope resulting from the piling-up at the wall promotes the emptying of the basin, now in the absence of wind forcing. The reflected bulge of water then travels back towards the sea entrance where it is reflected back again into the inner basin, in compliance with the standing wave behaviour consequent on the boundary condition.

Eventually, this repeating bouncing behaviour will die off on account of friction and of the mass exchange with the sea, which gradually bring the basin back to rest.

Therefore, in spite of the linearity requirements on the governing equations, the analytical solutions ((8.12)) and ((8.13)) describe a non-trivial behaviour of the water levels and depth-averaged velocities. The significant spatial and temporal variability of flow direction and intensity makes this test case very well-suited for assessing numerical codes.

Model description

In the test model set-up, Sobey's indications have been followed unless otherwise stated. No sensitivity analysis to parameters range has been carried out. The domain has a length of L = 5000 m and a depth of H= 2 m. The width, immaterial for the one-dimensional flow pattern, has been set to W = 250 m.

The atmospheric forcing is equal to $\mathcal{F}_0 = 1.5 \times 10^{-3}$ m^2/s^2. This can be obtained realistically from the classic wind-drag parametrization with wind velocity of 25 m/s (Bft 10), a drag coefficient of C_d = 0.002 and air density ρ_a = 1.2 kg/m^3. As a result, C = 4.429 m/s, $\hat{\eta}$ = 0.3832 m and $\hat{u}$ = 0.8487 m/s.

The wind pulse travels with a forward speed V= 5 m/s, which is higher than Sobey's original setting of 1 m/s. This speed results in a pulse advance of one grid cell per time step according to the specifications presented later on. It has been chosen to rule out from the outset that the interpolation of the discrete wind field representing the pulse interferes in the numerical solution.

The linear friction coefficient is λ = 10^{-4} s^{-1}. Using the above velocity scale $\hat{u}$, λ corresponds to a Manning coefficient m = 0.0055 s/m$^{1/3}$, representing an unusually smooth bed. On the other hand, for the purpose of validation, this choice of λ appears to lead to all terms of the governing equations being equally relevant. Therefore, the flow solver's formulations for all terms in the governing equations (8.10) and (8.11) are assessed comprehensively.

The initial conditions are a horizontal water level at η = 0 and quiescent water. The condition η = 0 is implemented at the upwind open boundary.

When comparing the analytical and numerical solutions, care is needed in handling the staggered-grid arrangement implemented by D-Flow FM. In the solver's representation, open boundaries are drawn onto the cells faces, that is velocity grid points. The water-level grid points where the boundary conditions are applied, however, are actually located at 'ghost' stations half cell-size off the open boundary. Therefore, the analytical solution has effectively been worked out for a basin of effective length L = 5000 m + $\Delta x/2$, where 5000 m is the distance between the boundaries as a user would normally define it in D-Flow FM, while Δx is the cell size.

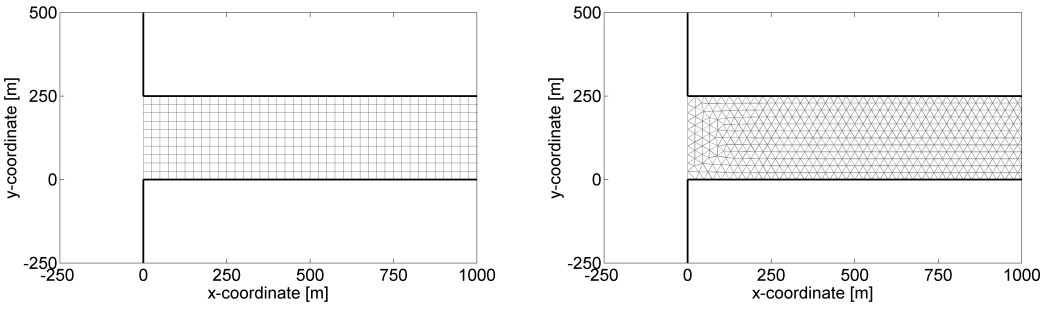


Figure 8.6: *Test case 'Travelling wind pulse in a partially open basin'. Computational grids: Cartesian-type (left) and structured with triangular cells (right). Part view near the upwind open boundary: the domain has a closed end at x = 5000 m.*

Both a Cartesian grid and a structured triangular grid are used to discretize the domain. The upwind parts of these grids are shown in Figure 8.6.

The cells in the Cartesian grid are squares with side of Δx = 25 m (user-defined).

As detailed in the previous section, the analytical solutions ((8.12)) and ((8.13)) contain infinite summations of inertial waves, while the grid spacing limits the possibility of reproducing the entire spectrum of oscillations. Therefore, in compliance with the Nyquist criterion, the number of modes in the exact solution has been limited to $n = n_x + 1$, where n_x is the number of cells in the direction of wave propagation. Likewise, a time step Δt = 5 s suffices to resolve all temporal oscillations that result from the dispersion relationship ((8.15)). As a result of this handling, the unity Courant number criterion is fulfilled too.

The cells in the triangular grid are equilateral with side of Δx = 25 m. The numerical results are compared with the same evaluation of the analytical solution as the Cartesian grid.

Concerning the time-dependent part in Equations ((8.12)) and ((8.13)), inconveniently, the convolution integrals have no straightforward solution in closed form. Therefore, they have been evaluated numerically with the so-called trapezoidal rule

$$\int_0^t f(\tau)d\tau \approx \Delta t \sum_{n=0}^{t/\Delta t} \frac{f\left((n+1)\Delta t\right) + f(n\Delta t)}{2}$$

where f is a generic integrand. This estimate is second-order accurate in time and compatible with the accuracy of the numerical solution.

In order to linearise the governing equations, the following settings have been specified in the mdu file:

- `AdvecType = 0`
- `Lincontin = 1`
- `Umodlin = 1`
- `UnifFrictCoefLin =` $\langle \lambda H \rangle$

The implicitness factor θ and the maximum Courant number have been modified with respect to default settings into

- `Teta0 = 0.5`
- `CFLMax = 1`

Additionally, one ad hoc modification was applied to the source code to linearise the wind forcing term by dividing the shear stress by H rather than by $\eta + H$. The other settings take default values.

Results

The simulated histories of water level and depth-averaged velocity have been compared with the analytical counterparts at selected observation points on the basin's middle longitudinal axis. In the following plots, the water levels and the depth-averaged velocities have been normalised with $\hat{\eta}$ and $\hat{u}$, respectively.

Each plot displays the exact values (in red), the computed value (in blue or green, dashed). The residuals of the computed values with respect of the exact ones (in black) are also plotted with a secondary axis. The results for the Cartesian and triangular grid are given in separate columns. For ease of interpretation with regard to the staggered-grid arrangement, from here onwards x denotes the co-ordinate value that a D-Flow FM user would normally deal with, as in Figure 8.6.

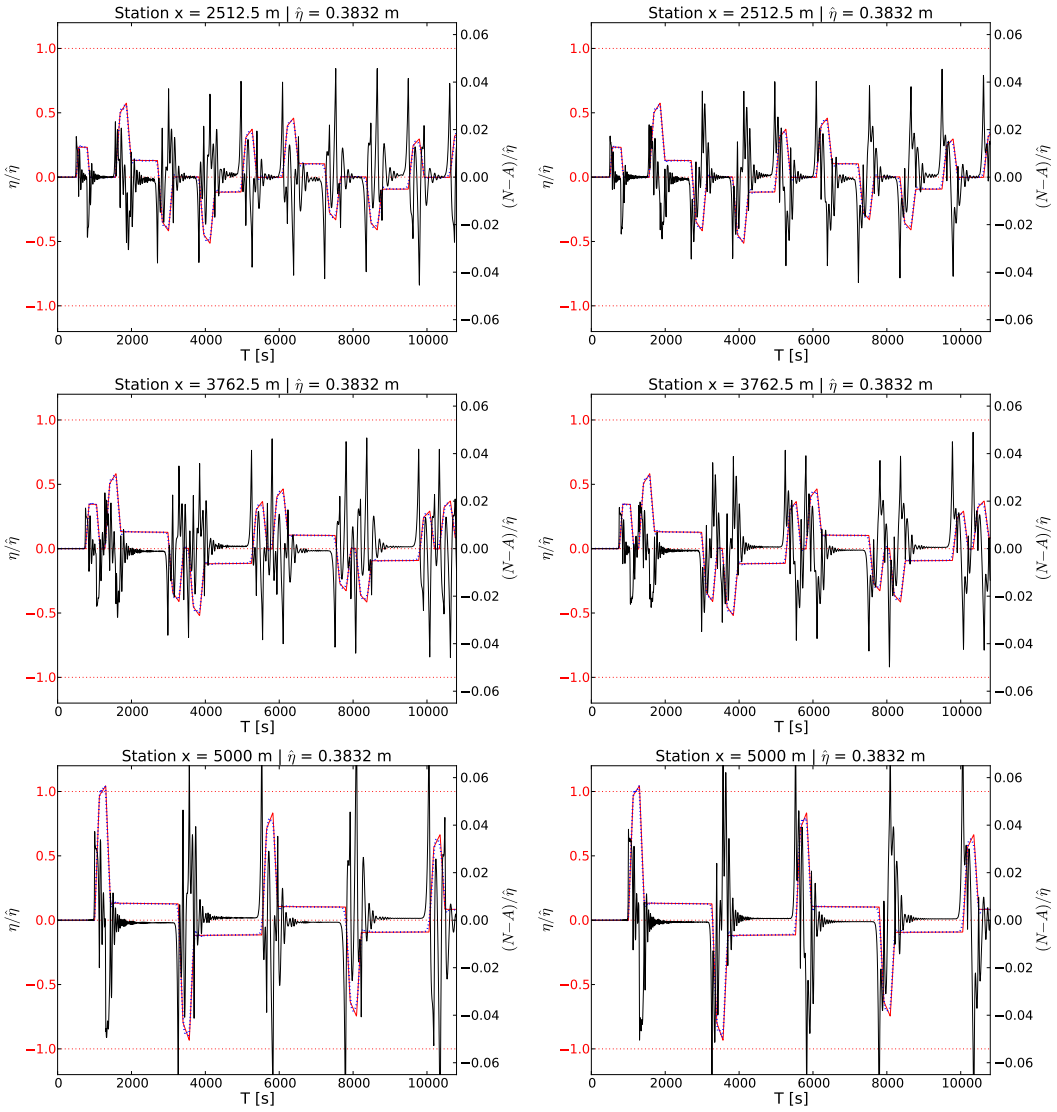


Figure 8.7: *Water levels. History of normalised values at mid (top), three-quarter (middle) and full basin length (bottom). Cartesian grid (left), triangular grid (right). Primary y-axis: analytical solution (A, red); D-Flow FM solution (N, blue dashed). Secondary y-axis: residual (N-A, black). Dashed red lines: steady-state wind set-up at location and exponential decay law.*

Like for the case of uniform and sudden wind forcing, the water levels are shown in Figure 8.7 for three stations closely at mid length (x = 2512.5 m), three-quarter length (x = 3762.5 m) and at the closed end of the basin (x = 5000 m). These observation points are located in the downwind half of the basin, where the largest water levels are expected owing to the water piling up against the wall.

As for the hydrodynamic processes as seen at fixed stations, the still-water level holds until the arrival of the wind-sheared water. The first sudden increase and fall of the water level corresponds to the passage of the water directly raised by the pulse overhead. Then, the subsequent positive surge is the trace of the water bulge while it returns after having bounced against the downwind wall (where, obviously, the raising and bouncing occur in a single surge peak).

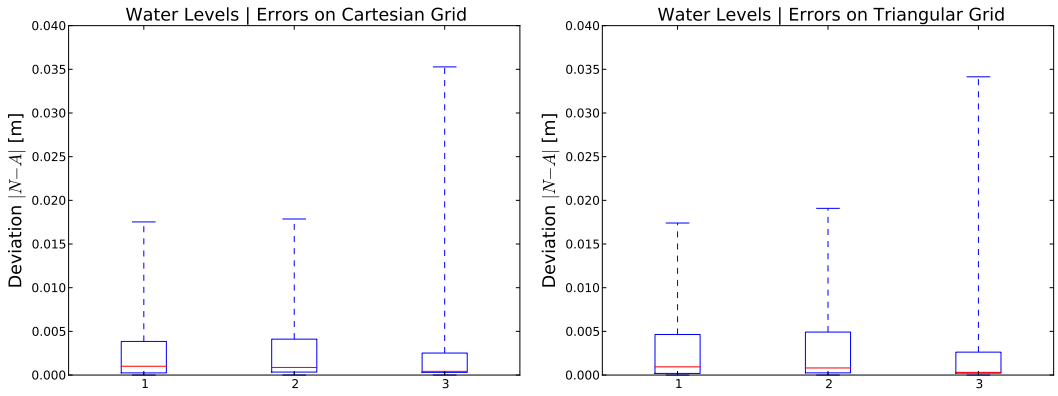


Figure 8.8: *Water levels. Deviations in absolute value between numerical and analytical histories in Figure 8.7. Cartesian grid (left), triangular grid (right). Station 1: mid basin. Station 2: basin third quarter. Station 3: full basin length.*

The ensuing negative surge is the result of the free-surface depression travelling landwards after the reflection, like a standing wave, at the sea entrance. A second pass of a negative surge then occurs on the way back of the same depression that, after having been reflected at the wall, is also enhanced due to concomitant flow divergence. This sequence of events then repeats itself and decays at a slow pace.

The numerical solution approaches the analytical solution very closely. The patterns of the residuals are rugged and indicate that the numerical solution deviates the most at times of sharp changes. However, such spikes are rapidly reduced as the computation proceeds, and do not appear to build up while the quasi-periodic pattern repeats itself. Recalling that the scale $\hat{\eta}$ only relates to the situations at the downwind wall, these plots provide qualitative information elsewhere. The maximum relative error at the downwind wall is $\pm$9.2%.

Turning to the box-and-wisker plots of Figure 8.8, the entire variability of the differences of the numerical and analytical solutions is summarised in absolute values. Such differences are at most 1.75 cm, 1.78 cm and 3.52 cm in the stations at middle, three-quarter and full length, against expected water-elevation ranges of -19 cm through 22 cm in both inner stations and -35 through 40 cm at the wall.

The results obtained with the structured triangular grid are similar to those from the Cartesian grid.

The equivalent plots for the depth-averaged velocity are shown in Figure 8.9, for three stations at the open boundary (x = 0 m) and at one-quarter (x = 1250 m) and one-half basin length (x = 2500 m). Positive values indicate flow landwards, negative seawards.

Here too, numerical and analytical results are in close qualitative agreement, with similar results for the Cartesian and structured triangular grid alike. The magnitude of the residuals observed in the simulation is summarized in Figure 8.10 as box-and-whisker plots, where the peak deviations for the Cartesian grid are in the order of 7 cm/s at the open boundary and 4 cm/s at the other stations. The expected ranges of the depth-averaged velocity at these stations are -84 cm/s through 75 cm/s at the open boundary, -46 cm/s through 41 cm/s at the other stations.

The results obtained with the structured triangular grid show slightly lower residuals.

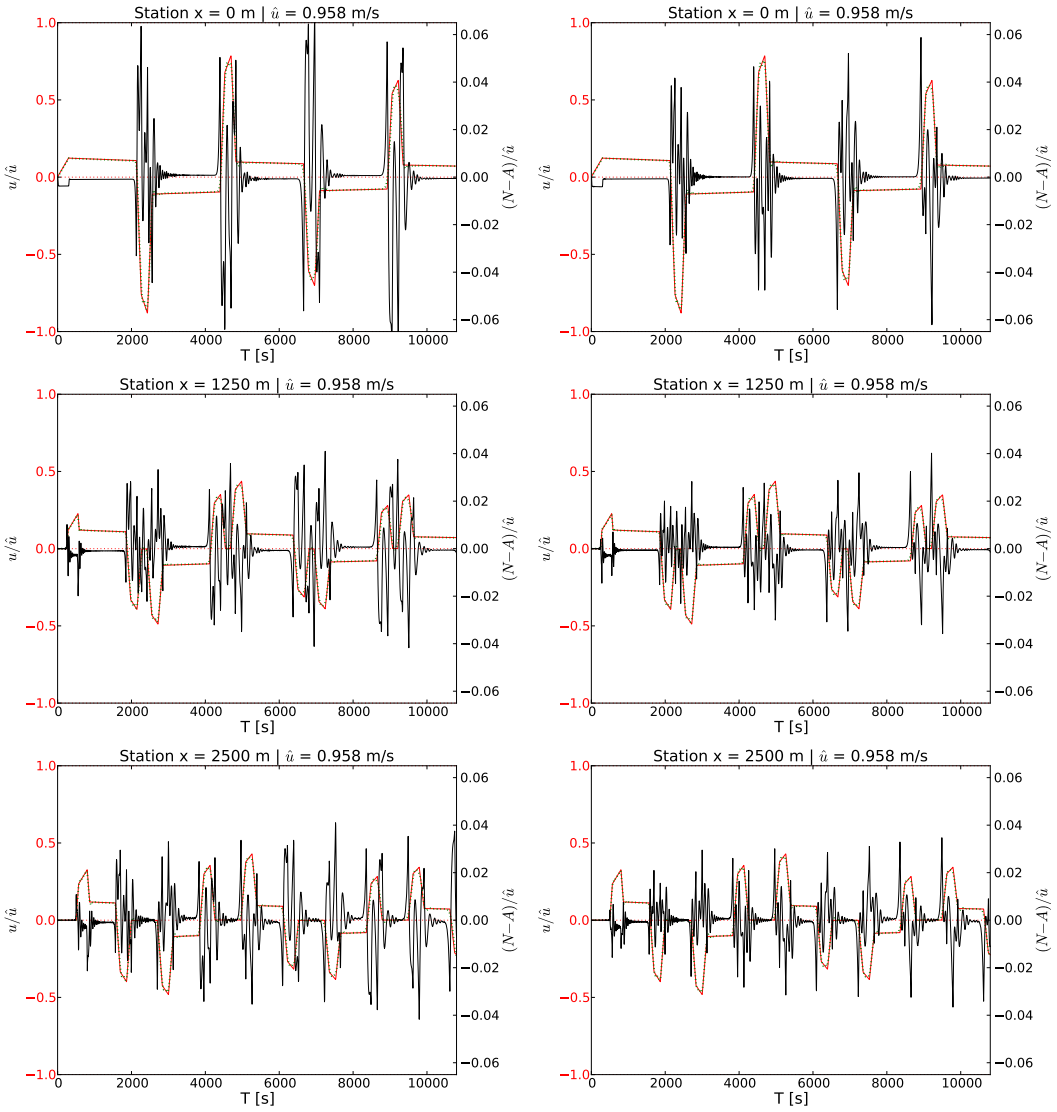


Figure 8.9: *Depth-averaged velocity. History of normalised values at open boundary (top), one-quarter (middle) and one-half basin length (bottom). Cartesian grid (left), triangular grid (right). Primary y-axis: exact solution (A, red); D-Flow FM solution (N, green dashed). Secondary y-axis: deviation (N-A, black).*

These observations support the conclusion that the largest relative deviations between numerical and analytical flow fields are within the $\pm$10% range. Interesting enough, the results obtained with a forward wind speed of 1 m/s (Sobey's original value, not discussed here) showed smaller deviations, arguably because of the more gradual development of the flow response when the pulse speed is lower than the long-wave celerity. The considerations drawn here can thus be regarded to be on the side of caution.

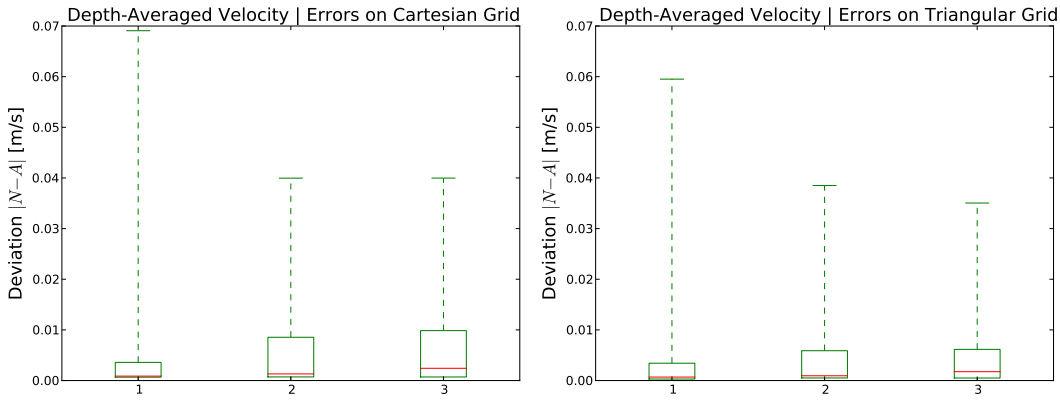


Figure 8.10: *Depth-averaged velocity. Deviations in absolute value between numerical and analytical histories in Figure 8.9. Cartesian grid (left), triangular grid (right). Station 1: open boundary. Station 2: basin first quarter. Station 3: middle basin.*

Conclusions

This test concerned the temporal and spatial variations of water levels and depth-averaged velocities in a partially-open rectangular basin with uniform depth, acted on by a wind pulse of limited extent travelling along its length. The resulting flow is one-dimensional and, with the parameters selected here, shows considerable spatial and temporal variability.

D-Flow FM has been adapted to solve the linearised equations of the analytical solution by means of available input keywords in the driver file and of one modification to the source code of little generality. The time step has been chosen to resolve the flow field down to the temporal resolution consequent on the user-defined grid spacing.

In spite of this simplification, a central part of the coding in D-Flow FM is put to the test. Not only are the terms retained in the equations driving mechanisms for the storm-surge generation process, but also working out the numerical solution activates the same algorithms as for the general governing equations. Given the objectivity of the analytical solution, it can then be checked how closely D-Flow FM solves for water levels and velocities that are mass- and momentum-conserving at each grid location and time level, as a result of the evolving inner flow field and of the initial conditions, boundary conditions and external forcing.

In this test, the agreement of the time-varying water levels and depth-averaged velocities at selected stations in the basin appeared to be high, whether the basin is represented with a mesh of rectangular or triangular elements. The residuals are consistently within $\pm$10% of the expected flow scales and show a bounded behaviour in computations in the order of 2000 time steps. Such residuals peak at times of sharp changes of the flow condition and do not taint the flow prediction with spurious effects of practical significance.

Therefore, in view of the necessary assumptions, anticipated behaviours and chosen settings, D-Flow FM has computed the expected flow field with fairly accurate and stable results on a Cartesian and a structured triangular grid.

Subject to the additional consideration of non-linear processes neglected or simplified here, these results lend a high degree of confidence that D-Flow FM can correctly simulate the long-wave component of storm surges acted on by spatially and rapidly variable wind fields.

Acknowledgements

This validation case has been conducted and commented by Giordano Lipari of Waterbeweging | Watermotion as a pilot user of D-Flow FM. Neither monetary nor non-monetary requital have been awarded for this work. Wim van Balen of Deltares provided the grids and has controlled the quality of results and conclusions.

Version

This test has been carried out with D-Flow FM version 1.1.116-36532.

8.3 Wind over a schematized lake

Purpose

The present test case investigates the flow that is induced by a uniform wind field. This case has been taken from the Delft3D validation document. Solutions obtained on several grids (Cartesian/triangular, coarse/fine) can be compared with the solution that is obtained using Delft3D.

Linked claims

Claims that are related to the current test case are:

◇ claim 2.3.5.2: D-Flow FM can be used for an accurate prediction of wind driven flow

Approach

This validation study represents a schematised lake. It is a closed basin in which a (constant) northwestern wind forcing is applied, under an angle of 315° with the north-direction and with a velocity of 5 m/s. Thus, the wind direction is not aligned along the grid lines.

A space varying depth is applied. The schematised lake is the deepest in the center of the basin, see Figure 8.11. Due to the constant wind forcing, a steady state solution is reached. The prescription of the bathymetry is available on a resolution of 1 km $\times$ 1 km.

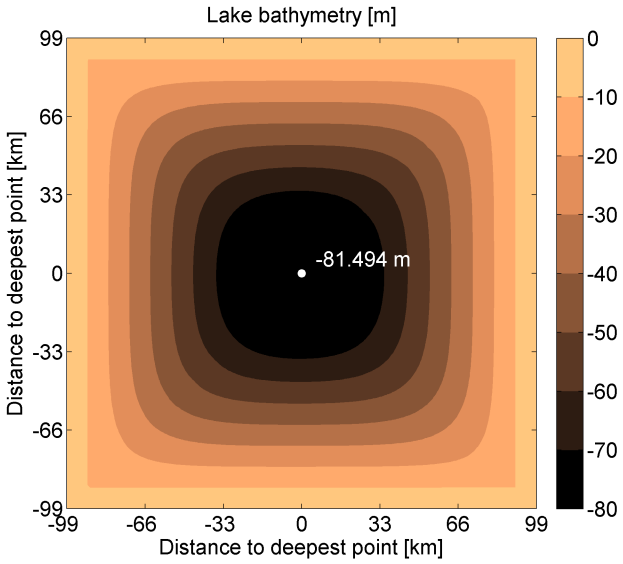


Figure 8.11: *Geometry and bathymetry of the schematized lake.*

For comparison, the outcomes of a Delft3D computation are available. This Delft3D computation has been run on a grid with cells of varying size: the left part of the schematised lake has a constant grid size of 2000 m and the right part of 1000 m, yielding a grid refinement of a factor of two. Results from Delft3D are only available for the first 2.5 days (60 hours) of the simulated time.

Model description

The settings for this wind test case are:

- ◇ domain sizes are $L = B = 198$ km,
- ◇ the computational time step varies in time, using a CFL timestep limit equal to 0.7,
- ◇ a Manning type roughness with a uniform friction value equal to 2.6 s $m^{-1/3}$,
- ◇ a uniform horizontal eddy viscosity equal to 1.0 m^2/s,
- ◇ a linear dependency of the wind drag coefficient C_d on the wind speed, varying from $C_d = 6.3 \cdot 10^{-4}$ for zero wind speed to $C_d = 7.23 \cdot 10^{-3}$ for a wind speed equal to 100 m/s,
- ◇ the simulated time is 12 days.

Five grids have been established in D-Flow FM for this particular test case: three Cartesian grids with strictly square cells and two triangular grids:

- ◇ a coarse Cartesian grid with 22×22 square cells of size 9×9 km^2,
- ◇ a medium Cartesian grid with 66×66 square cells of size 3×3 km^2,
- ◇ a fine Cartesian grid with 198×198 square cells of size 1×1 km^2,
- ◇ a coarse triangular grid with cells of typical edge length of 9 km,
- ◇ a medium triangular grid with cells of typical edge length of 3 km.

Recall that the bathymetry resolution is 1 km. The two coarse grids are shown in Figure 8.12.

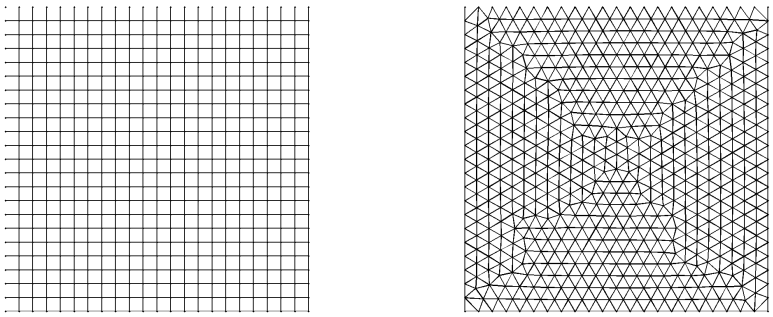


Figure 8.12: *Cartesian and triangular grids for the schematized lake case. For both grids, a refined (by a factor of 3 in each direction) equivalent is available in addition. For the Cartesian, a refined grid (by a factor of 9 in each direction) is also available.*

Results

As soon as the computation has started, oscillations come into existence that damp out towards a steady state. This steady state is reached after about 12 days. The velocity field in this steady state is shown in Figure 8.13. The flow in the shallow parts of the domain is in the direction of the wind, whereas in the deepest part of the domain the flow is in opposite direction. The results shown in Figure 8.13 are obtained on the coarse triangular grid.

Time series are shown in Figure 8.14 for all the five grids. The development towards a steady state is clearly seen. The associated time scale is independent of the grid size; the eventual equilibrium water level, however, is.

The equilibrium values for the maximum and minimum water levels are tabulated in Table 8.1.

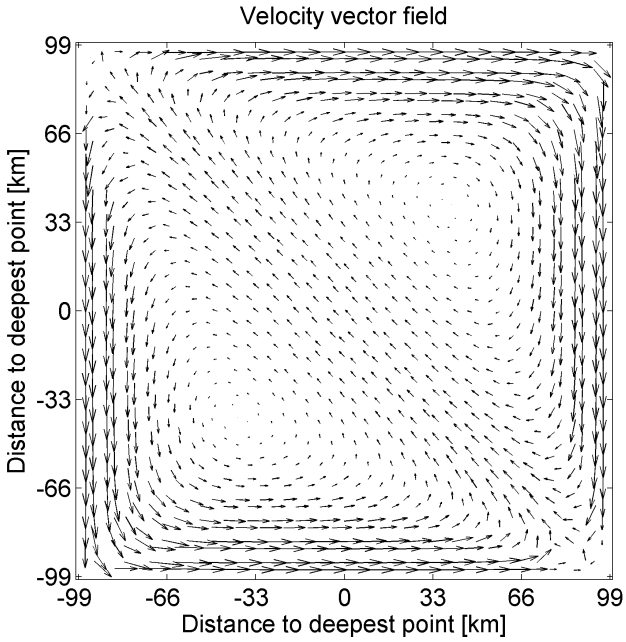


Figure 8.13: *Velocity vectors for the coarse triangular grid (steady state).*

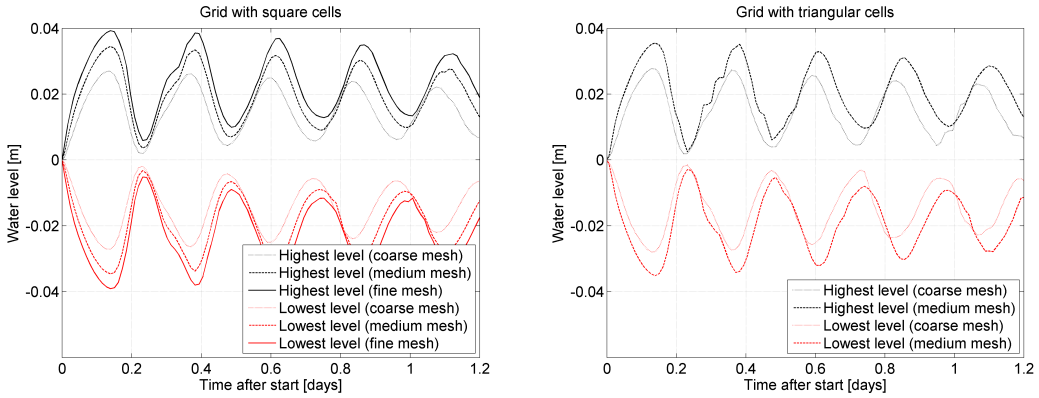


Figure 8.14: *Minimum water levels (in red) and maximum water levels (in black) for the two grids: evolution towards steady state.*

This table reflects the steepening of the water level gradient with decreasing grid size which is probably due to the more accurate representation of the bathymetry on finer grids. The shape of the cells appears to hardly influence the final result.

From the equivalent Delft3D computations, the maximum and minimum water levels are only available for the first 60 hours, with intervals of 6 hours. These results are shown in Figure 8.15 for these first 60 hours. The associated D-Flow FM results are only highlighted for the fine mesh with square cells. No particular conclusions are drawn from this concise comparison.

Table 8.1: *Maximum and minimum water levels in meters for several grids, as well as the associated slope.*

Grid type	Refinement	h_{max} [m]	h_{min} [m]	slope [-]
square cells	coarse	0.01394841	-0.01338925	$1.02278545 \cdot 10^{-7}$
	medium	0.01888039	-0.01769081	$1.32614055 \cdot 10^{-7}$
	fine	0.02323332	-0.02120412	$1.59502644 \cdot 10^{-7}$
triangular cells	coarse	0.01398769	-0.01339067	$1.01621116 \cdot 10^{-7}$
	medium	0.01949882	-0.01776750	$1.36178155 \cdot 10^{-7}$

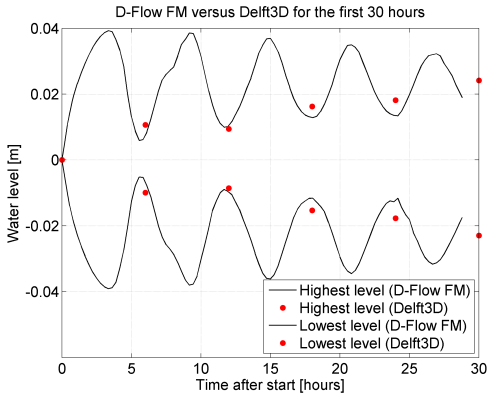


Figure 8.15: *Comparison of the D-Flow FM results (for the fine Cartesian grid) with the Delft3D results, for the first 30 hours of the simulated time.*

Conclusion

A square lake with a non-uniform topography has been subjected to an external force induced by a uniform wind field. The solutions reached a steady state, showing an increasing water level slope with increasing cell density of the grid. This is probably due to the more accurate representation of the bathymetry. Differences in cell shape (square/triangle) appeared to yield minor mutual differences.

Version

This test has been carried out with version dflow-fm-x64-1.1.116.36629.

9 Weirs

9.1 Critical weir flow

Purpose

Weirs are applied to regulate river flow. In particular, in Dutch rivers many so-called groynes are located in the summer bed of rivers. Furthermore, in the winter bed of rivers many small dikes and other small obstacles exist. In D-Flow FM, the effect of these weirs (either groynes or dikes) on the water levels can be modelled. Weirs are fixed, non-movable obstacles causing energy losses due to constriction of flow.
In D-Flow FM there are two approaches to represent these weirs: *subgrid* or *supergrid*. In case of subgrid modelling these weirs are not represented in the bathymetry. Instead, in the input file a so-called *Fixedweir* should be defined at the location of an obstacle. Then, locally a special numerical scheme for horizontal advection is applied in order to compute the energy loss. Alternatively, in case of supergrid modelling a weir is modelled by elevating the local bed level at several grid cells. The standard advection scheme is applied, without any adaptation for weirs. This supergrid approach requires a relatively high grid resolution near obstacles and thus leads to more computation time compared to the subgrid approach.
The flow condition at the crest of a weir may be critical or subcritical. The present test case investigates both subgrid and supergrid modelling for computing critical flow over a weir in case of quasi-1D flow.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.6.1: D-Flow FM can be used to investigate the hydrodynamic impact of hydraulic structures, such as gates, weirs and barrier
- ◇ claim 2.3.6.2: D-Flow FM can accurately simulate subcritical and supercritical flows and the transition region when the flow changes from subcritical to supercritical or vice versa

Approach

For critical flow the discharge over a weir is completely determined by the energy head upstream. Then, the discharge is limited by:

$$Q_{critical} = c_{critical} W \frac{2}{3} E_1 \sqrt{\frac{2}{3} g E_1} \tag{9.1}$$

with $c_{critical}$ an empirical coefficient for representing the geometry that is specified by the user in the input file, W the width of the channel and E_1 the energy height upstream. For more details we refer to the D-Flow FM technical manual. In order to examine the accuracy of both approaches for weirs in D-Flow FM, five test models are constructed. Each of the five models is based on a simple Cartesian grid consisting of 8×1 grid cells. Variation is achieved by choosing a certain refinement strategy for both approaches (either subgrid or supergrid).

Table 9.1: *Computed discharges for the five different model grids.*

supergrid, 30 m grid	137 m^3/s
supergrid, locally 10 m near weir	47.7 m^3/s
supergrid, 10 m grid	58.0 m^3/s
subgrid, 30 m grid	54.4 m^3/s
subgrid, 10 m grid	54.4 m^3/s

Model description

This validation case is referred to as 'Beyer004' (Beyer, 2001). For both approaches we study the effect of local grid refinement. The following grids are generated (see Figure 9.10):

- ◇ grid 1: the basis grid consisting of 8×1 grid cells of 30 m $\times 30$ m each, for which the weir is represented in the bed level;
- ◇ grid 2: as grid 1, but with a local grid refinement of the cell directly downstream of the weir; the crest of the weir is represented over the full three cells of 10 m each, which is the result of the refinement;
- ◇ grid 3: the basis grid consisting of 24×1 grid cells of 10 m $\times 30$ m each, for which the weir is represented in the bed level;
- ◇ grid 4: as grid 1, but with the subgrid approach instead of supergrid approach;
- ◇ grid 5: as grid 3, but with the subgrid approach instead of supergrid approach.

Upstream of the model a water level boundary is specified at a constant level of 13 m. Downstream also a water level boundary condition is specified. Initially the level is 13 m and goes to 11.5 m at the end of the simulation.The initial water level is 13 m. The bathymetry is constant at a level of 11 m, yielding a constant initial water depth of 2 m. The simulation period is 2 hrs. The weir is located in the centre of the model domain and has a crest height of 12 m. So, the energy height upstream E_1 of the crest is about 1 m. In this testcase the bottom friction is switched off. For this testcase the theoretical discharge for the critical flow Q_{critical} over the weir yields 54.4 m^3/s.

Results

In Table 9.1 the computed discharges are shown for the five different model grids, in which both the subgrid and supergrid approaches have been validated.

It is evident that on the coarse grid (30 m) the supergrid approach yields inaccurate results. For the supergrid approach (locally) a higher resolution is required. For the subgrid approach the numerical method is designed such that for critical flow the theoretical critical discharge is computed exactly as given in Equation (9.1). The accuracy is independent of the grid resolution. This explains why in this validation case the subgrid model results are identical on 10 m and 30 m grids.

Conclusions

This validation case shows that D-Flow FM is able to compute the discharge flow accurately over a weir for critical flow.. The subgrid approach (fixed-weir) for computing discharges for critical flow gives exact results and is to be preferred. In case of supergrid modelling the error is dependent on the grid size. The order is 10 $\%$ provided that a locally refined grid resolution is applied around the weir crest.

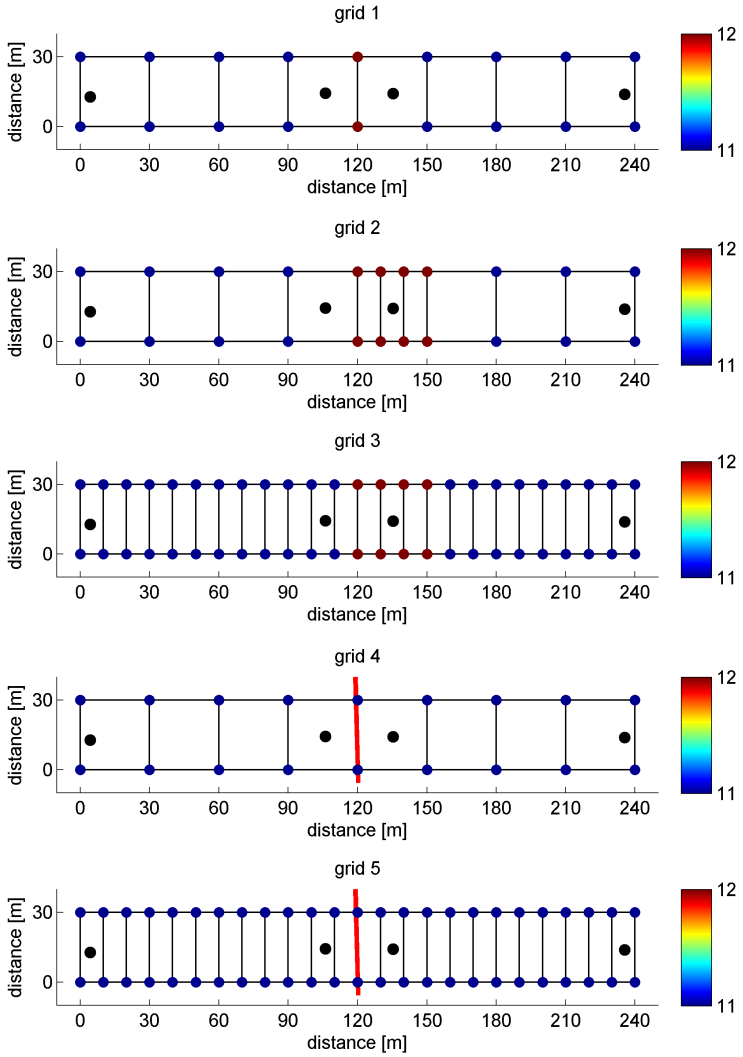


Figure 9.1: *Computational grid for the weir case. The weir, with crest level 12 m (w.r.t. reference), is represented by the red line. The observation points are represented by the black dots. The coloring of the grid cell corner points indicate the bed level.*

Version

This test has been carried out with version dflow-fm-x64-1.1.142.40690.

9.2 Transition between subcritical and critical weir flow

Purpose

The flow condition at the crest of a weir is either critical or subcritical. The present test case investigates the transition of subcritical to critical flow and vice versa. This is only validated for the subgrid approach in D-Flow FM. For subcritical flow the energy loss over a weir is completely determined by discretization of the advective terms in the momentum equations. D-Flow FM has available a so-called contraction coefficient in order to calibrate the energy loss due to weirs.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.6.1: D-Flow FM can be used to investigate the hydrodynamic impact of hydraulic structures, such as gates, weirs and barrier
- ◇ claim 2.3.6.2: D-Flow FM can accurately simulate subcritical and supercritical flows and the transition region when the flow changes from subcritical to supercritical or vice versa

Approach

In D-Flow FM the energy loss over a weir is completely determined by discretization of the advective terms in the momentum equations. The numerical scheme introduced by Kramer and Stelling (2008) is locally applied at the crest of the weir. Upstream of a weir the discretization of the advective terms is such that 'energy conservation' is guaranteed, while at the downstream part 'conservation of momentum' is used. In this way, the resulting energy loss over a weir depends on the local mesh size and the local bathymetry.
D- Flow FM offers the possibility to fine tune this energy loss via adapting the contraction coefficient. In case of critical flow the contraction coefficient is 1.0 and the discharge coefficient coincides with $c_{critical}$ in Eq. (9.1). The critical flow scales linearly with the value of this contraction coefficient. For subcritical flow the contraction coefficient influences the discharge over a weir as well. Thus, a contraction coefficient influences the discharge at critical flow and the energy loss at subcritical flow.
The default value for this contraction coefficient reads 1.0. When decreasing the contraction coefficient (e.g. to 0.9), then the wetted cross section of a cell face at which a weir is located, is reduced by 10%. As a result, the velocity at the weir crest increases and thus the energy loss. Similarly, by increasing the contraction coeffcient the energy loss reduces. In summary, the contraction coefficient can be seen as a calibration parameter for the energy loss.

In this validation case the model results are compared with the model results of software system WAQUA, which adds in the momentum equation an empirical energy loss, derived from measured datasets. Numerical modelling of weirs in WAQUA has been extensively validated with measurements and is applied for many decades to compute water levels in the main Dutch rivers.

Model description

The following models are generated (see Figure 9.10):

- ◇ model 1: with a grid consisting of 8×1 grid cells of 30 m $\times 30$ m each in combination with the subgrid approach for the weir. Noted that this case is identical to the 'critical weir flow' test case with grid 4. The contraction coeffcient equals 1.0. This test case is referred to as 'Beyer004' (Beyer, 2001).
- ◇ model 2: as model 1, but with a contraction coefficient of 0.9;
- ◇ model 3: as model 1, but with a contraction coefficient of 1.1;
- ◇ model 4: as model 1, but with a different bathymetry. In models 1 to 3 a flat bathymetry of +11 m is applied. In model 4 the bathymetry after the weir is +1 m; thus, 10 m lower than in the other three models. This test case is referred to as 'Beyer007' (Beyer, 2001).

To simulate the transition between subcritical and critical weir flow, the discharge at the inflow boundary is increased from 0 to 200 m^3/s in 24 hours.The next 24 hours the discharge is decreased from 200 m^3/s to 0 m^3/s. The downstream boundary is constant at a level of 12.5 m. The initial water elevation is 12.5 m. The depth is constant at +11 m, yielding an initial water depth of 1.5 m. The weir is located in the centre of the model domain and has a crest level of 12 m. So, the initial energy height upstream E_1 is 0.5 m. The global mesh size is 30 m. The simulation period is 2 days. This tescase is run without bottom friction. In this testcase the change from critical to subcritical flow and vice versa is examed. The condition changes around 55 m^3/s.

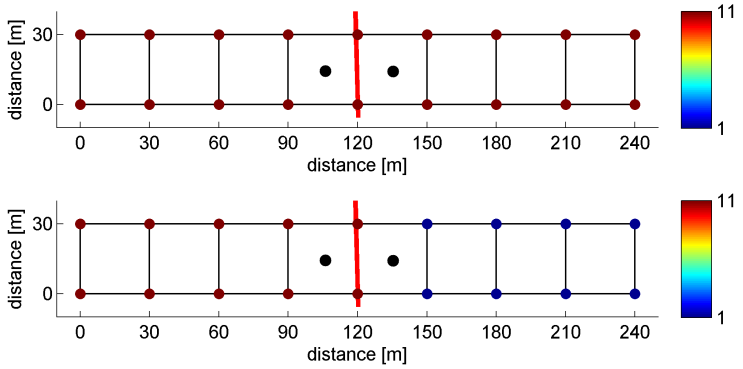


Figure 9.2: *Computational grid for the weir case. The weir, with crest level 12 m (w.r.t. reference), is represented by the red line. The observation points are represented by the black dots. The coloring of the grid cell corner points indicate the bed level.*

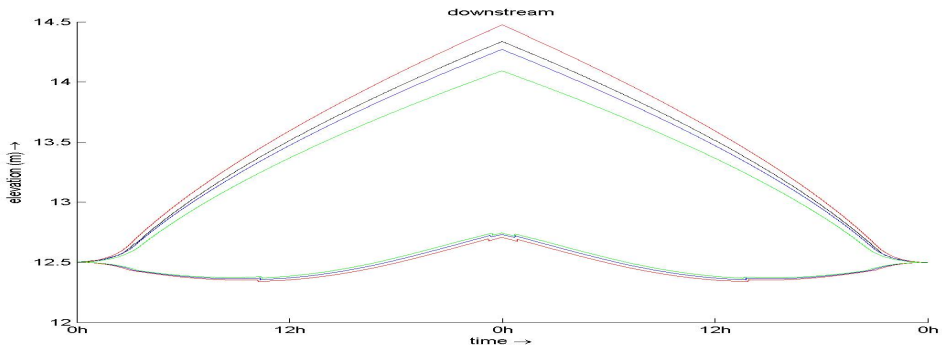


Figure 9.3: *Transition between subcritical and critical weir flow*

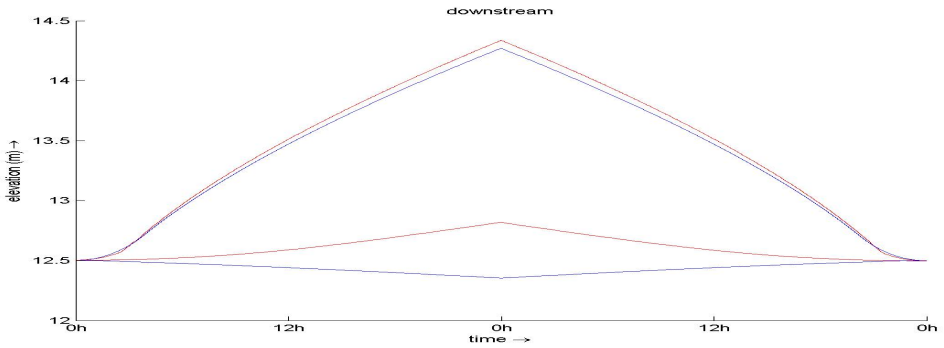


Figure 9.4: *Transition between subcritical and critical weir flow (2)*

Results

Figure 9.3 contains the water levels both upstream and downstream of the weir. The difference in the water level over the weir is an indication for the computed energy loss. It shows that the energy losses for D-Flow FM with the default contraction coefficient of 1.0 (see blue lines) and WAQUA (seem black lines) are in reasonable agreement with each other.
By decreasing the contraction coefficient to 0.9, the energy loss increases (see red line). Similarly, when increasing the contraction coeffcient to 1.1, the energy loss decreases.
For model 4 the bathymetry downstream of the weir is decreased by 10 m. Also for this model, the D-Flow FM and WAQUA model results are in acceptable agreement with each other; see Figure 9.4, with in red and blue the water levels for models 4 and 1, respectively.

Conclusions

This validation case shows that D-Flow FM is able to smoothly simulate the transition from subcritical to critical flow and vice versa. The agreement with the validated WAQUA software is reasonable. Furthermore, by fine tuning the contraction coefficient the user has the possibility to further improve the model results.

Version

This test has been carried out with version dflow-fm-x64-1.1.142.40690.

9.3 Empirical Tabellenboek and Villemonte approach for fixed weirs

Purpose

The present validation case investigates the empirical Tabellenboek and Villemonte approach for the computation of the weir energy losses. Next to a numerical approach for weirs, which is applied in the other weir validation cases in this document, D-Flow FM also allows for the use of the empirical Tabellenboek and Villemonte approach.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.6.1: D-Flow FM can be used to investigate the hydrodynamic impact of hydraulic structures, such as gates, weirs and barrier
- ◇ claim 2.3.6.2: D-Flow FM can accurately simulate subcritical and supercritical flows and the transition region when the flow changes from subcritical to supercritical or vice versa

Approach

Two different subgrid approaches are available in D-Flow FM to compute the energy losses by fixed weirs. First of all, a numerical approach is available. Then, a special discretization of the advective terms before and after a fixed weir is applied. This option corresponds with keyword `fixedweirtype=6`. Next to the numerical approach, there is an empirical approach to compute the energy loss by weirs, for which two options are availabe in D-Flow FM, namely the so-called 'Tabellenboek' and 'Villemonte'. The Tabellenboek option corresponds with keyword `fixedweirtype=8` and the Villemonte approach with keyword `fixedweirtype=9`. The two corresponding empirical formulas have been taken from the Simona software, see the website `http://www.helpdeskwater.nl/onderwerpen/applicaties-model/applicaties-per/watermanagement/watermanagement/simona`. These two empirical formulas have been derived from flume measurements. In the empirical approach an additional term is added to the momentum equation. We remark that the discretization of the advective terms is not changed. Modelling of weirs via the Tabellenboek with the Simona software suite has been extensively validated with measurements and is applied for many decades to compute water levels in the main Dutch rivers.

In this validation case the D-Flow FM model results for the numerical approach and the two empirical approaches (Tabellenboek and Villemonte) are compared with each other. The same model as in validation case 9.2 is applied, in which the transition of subcritical to critical flow and vice versa is examined.

Model description

The following models are generated (see Figure 9.6):

- ◇ model 1: with a grid consisting of 8×1 grid cells of 30 m $\times 30$ m each in combination with the subgrid approach for a weir. Noted that this case is identical to the 'critical weir flow' test case of validation case 9.2. The numerical approach (`fixedweirtype=6`) is applied. This validation case is referred to as 'Beyer004' (Beyer, 2001).
- ◇ model 2: as model 1, but with the Tabellenboek approach (`fixedweirtype=8`)
- ◇ model 3: as model 1, but with the Villemonte approach (`fixedweirtype=9`)

To simulate the transition between subcritical and critical weir flow, the discharge at the inflow boundary is increased from 0 to 200 m^3/s in 24 hours. The next 24 hours the discharge is decreased from 200 m^3/s to 0 m^3/s. The downstream boundary is constant at a level of

12.5 m. The initial water elevation is 12.5 m. The depth is constant at +11 m, yielding an initial water depth of 1.5 m. The weir is located in the centre of the model domain and has a crest level of 12 m. So, the initial energy height upstream E_1 is 0.5 m. The simulation period is two days. This validation case is run without bottom friction. In this validation case the change from critical to subcritical flow and vice versa is examined. The flow condition changes around a discharge of 55 m^3/s.
For Villemonte the default settings are applied. This e.g. means that the downward and upward slope are at the scale of 1-to-4.

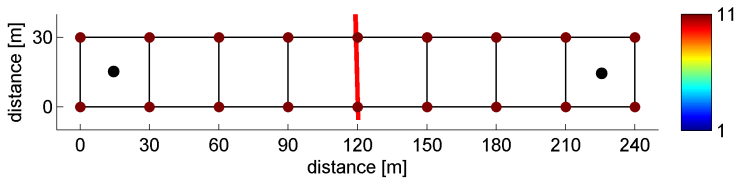


Figure 9.5: *Computational grid for the weir case. The weir, with crest level 12 m, is represented by the red line. The observation points are represented by the black dots. The coloring of the grid cell corner points indicate the bed level.*

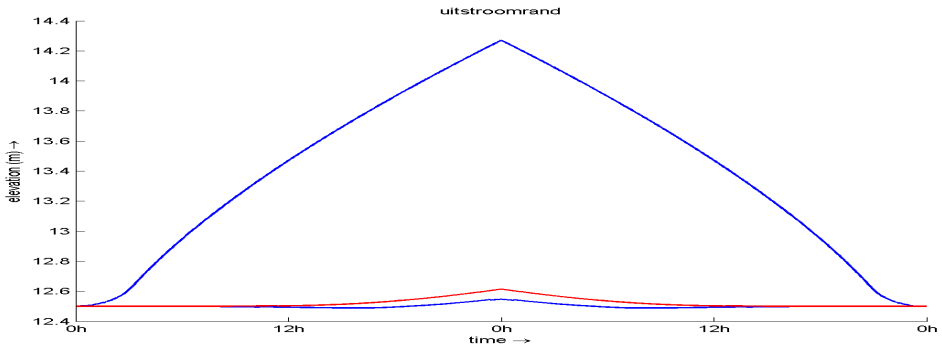


Figure 9.6: *Water levels near inflow and outflow boundary for the numerical approach (in blue), the Tabellenboek (in green) and Villemonte (in red)*

Results

Figure 9.6 contains the water levels near the inflow and outflow boundary for the numerical approach (in blue), the Tabellenboek (in green) and Villemonte (in red). Near the infow boundary all three simulations give almost identical results. Therefore, only the results for the numerical approach are visible. Near the outflow boundary the Tabellenboek (in green) and Villemonte (in red) yield almost identical results. The numerical approach (in blue) leads to slightly different water levels. The difference in the water level over the weir is an indication for the computed energy loss. Figure 9.6 shows that the computed energy losses for all three approaches in D-Flow FM are comparable.

Conclusions

This validation case shows that the three approaches in D-Flow FM for the computation of energy losses due to fixed weirs yield similar results. The two empirical approaches yield almost identical results. The numerical approach gives almost similar results. In validation case 9.2 it was already shown that the numerical approach yields model results that are in reasonable agreement with the validated Simona software. Now, it is also shown that this is the case for the Tabellenboek and the Villemonte approaches.

In practice, these two empirical approaches are to be preferred, because a disadvantage of the numerical approach is that it does not allow that fixed weirs are located in neighbouring grid cells. In real-life models this is often the case.

Version

This test has been carried out with version dflow-fm-x64-1.1.169.44046

9.4 Two-dimensional critical flow over an oblique weir

Purpose

In earlier test cases, the focus has been on testing weirs in D-Flow FM for quasi one-dimensional flows. In the present test case, two-dimensional critical flow over an oblique weir is considered. For modelling of fixed weirs different approaches are available. In these tescases the numerical approach (`fixedweirscheme=6`) is applied.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.6.1: D-Flow FM can be used to investigate the hydrodynamic impact of hydraulic structures, such as gates, weirs and barrier
- ◇ claim 2.3.6.2: D-Flow FM can accurately simulate subcritical and supercritical flows and the transition region when the flow changes from subcritical to supercritical or vice versa

Approach

For critical flow the discharge over a weir is completely determined by the energy head upstream. Two model domains with a mesh size of 30 m and 10 m are considered, respectively. The oblique weir in these models makes a corner of either 45 degrees or 30 degrees with the grid.

Model description

The following grids are generated (see Figure 9.10, Figure 9.8 and Figure 9.9):

- ◇ grid 1: grid cells of 30 m with an oblique weir; see left panel in Figure 1,
- ◇ grid 2: grid cells of 10 m with an oblique weir; see right panel in Figure 1,
- ◇ grid 3: uniform skew channel flow with grid cells of 30 m under 45 degrees; see left panel in Figure 2,
- ◇ grid 4: as grid 3, but with straight (1D) flow near the weir; see middle panel in Figure 2,
- ◇ grid 5: as grid 3, but with a supergrid approach for the weir; see right panel in Figure 2,
- ◇ grid 6: uniform skew channel flow with grid cells of 30 m under 30 degrees; see top panel in Figure 3,
- ◇ grid 7: as grid 6, but with a supergrid approach for the weir; see middle panel in Figure 3,
- ◇ grid 8: as grid 6, but with a mesh size of 10 m; see bottom panel in Figure 3.

For weirs that make a corner of 45 degrees with the grid (see grids 1 to 5), the critical flow should be a factor of $\sqrt{2}$ larger compared to the 1D weir validation case for critical flow. This is due to the fact that the length of the weir increases with this factor. This yields $\sqrt{2} \times$ 54.92 m^3/s =71.6 m^3/s. For weirs that make a corner of 30 degrees with the grid the discharge should be $\sqrt{5} \times$ 54.92 m^3/s =97.61 m^3/s.

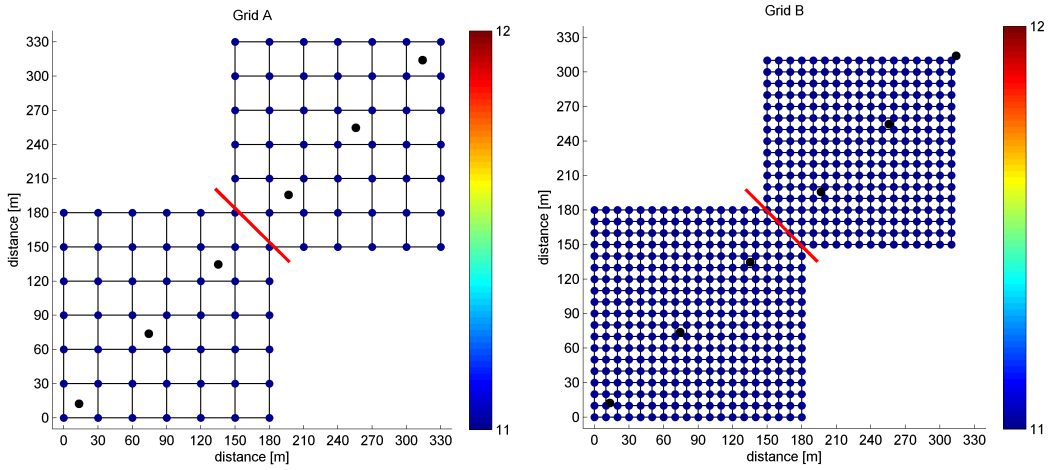


Figure 9.7: *weir grid in the form of a sandglas.*

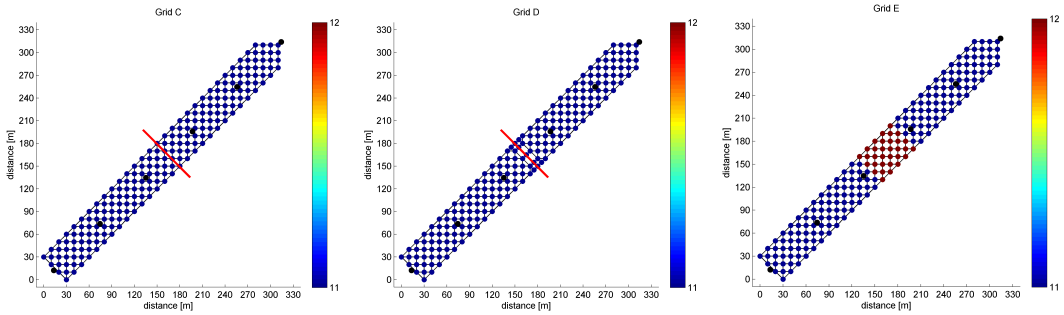


Figure 9.8: *uniform weir flow on a skew grid under 45 degrees.*

Results

In Table 9.4 the computed discharges are shown for the five different model grids, in which an oblique weir that makes a corner of 45 degrees with the grid. The theoretical value reads 71.6 m^3/s. This shows that the subgrid approach for weirs in D-Flow FM is able to compute discharges over weirs in a reasonable accurate way in such cases. The errors are in the order of 5 to 10 $\%$. In case of a supergrid approach the errors are somewhat larger.

In Table 9.5 the computed discharges are shown for three different model grids, in which an oblique weir that makes a corner of 30 degrees with the grid. Now, the theoretical value equals 97.6 m^3/s. Again, the subgrid approach computes more accurate discharges compared to the supergrid approach.

Table 9.2: *Discharges across the weir for the 45 $^\circ$ variant.*

skew grid, 30 m grid	70.1 m^3/s
skew grid, 10 m	70.7 m^3/s
uniform skew grid, 30 m grid	74.5 m^3/s
uniform skew grid with straight flow near weir, 30 m grid	76.9 m^3/s
uniform skew grid, supergrid approach, 30 m grid	80.4 m^3/s

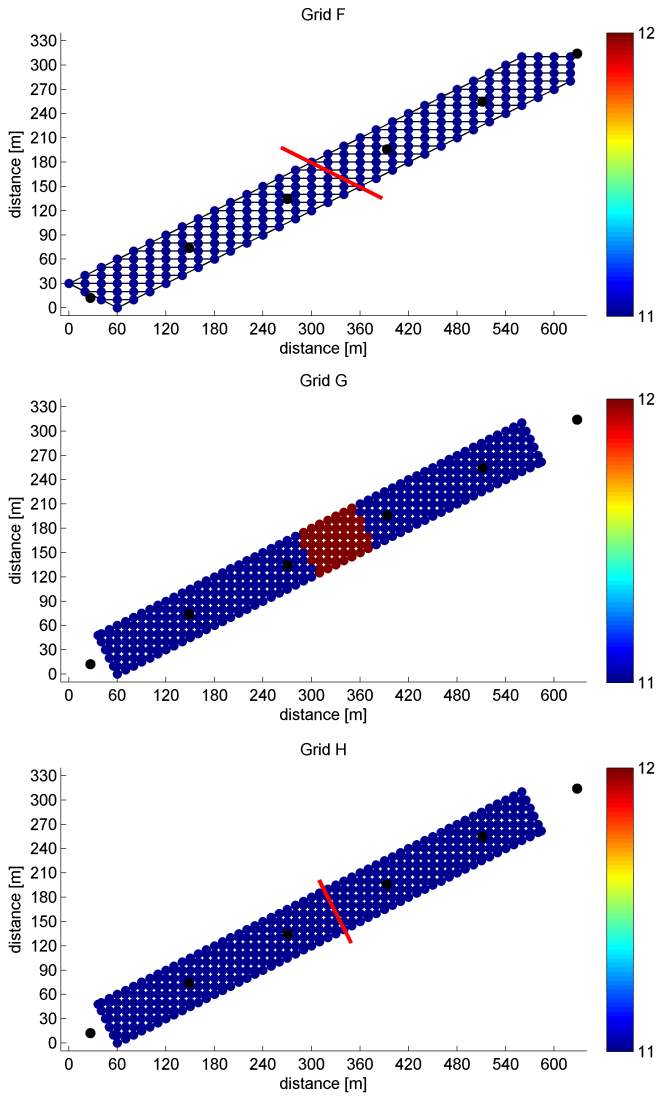


Figure 9.9: *uniform weir flow on a skew grid under 30 degrees.*

Conclusions

This validation case shows that the numerical approach (`fixedweirscheme=6`) in D-Flow FM can compute critical flow over an oblique weir in a reasonable accurate way. The subgrid approach for computing discharges for critical flow is to be preferred over the supergrid approach, both in terms of accuracy and computation time.

Version

This test has been carried out with version dflow-fm-x64-1.2.94.66139.

Table 9.3: *Discharges across the weir for the 30 ° variant.*

uniform skew grid, 30 m grid	103.0 m^3/s
uniform skew grid, supergrid approach, 10 m grid	94.7 m^3/s
uniform skew grid, 10 m grid	91.8 m^3/s

9.5 Flooding over an obliquely oriented weir

Purpose

In Dutch rivers dikes have been constructed in order to protect flooding and to create a foreland (in Dutch "uiterwaarden") that act as a buffer to store floodwaters. The so-called summer dikes separate the main river channel from the forelands. These dikes are quite often not aligned with the computational grid. In this validate case we therefore examine the flooding of a schematized river weir that represents a summer dike that is not aligned with the grid.

Linked claims

- ◇ claim 2.3.6.1: D-Flow FM can be used to investigate the hydrodynamic impact of hydraulic structures, such as gates, weirs and barrier
- ◇ claim 2.3.6.2: D-Flow FM can accurately simulate subcritical and supercritical flows and the transition region when the flow changes from subcritical to supercritical or vice versa

Approach

From the WAQUA software system it is known that flooding over an obliquely oriented weir can cause abrupt changes in model results. Even oscillations in the time behaviour of discharges over such a weir can occur. Therefore, the main criterion in this validation case is whether the flow over this weir smooth a smooth time evolution.

Model description

This validation case is referred to as 'Beyer044', see (Beyer, 2001). This validation case is referred to as 'Beyer044' (Beyer, 2001). The test model consists of a square grid of 10 by 10 grid cells with a diagonal channel. The dikes along this channel are at a height of 14 m, exact for two weirs that are at +12 m and thus 2 m lower; see the red lines in the left panel of Figure 1. This validation case consists of many 'stair cases' with respect to the location of the weirs.

As an alternative, D-Flow FM allows the possibility to nicely aligned weirs in a diagonal channel. This is illustrated in the right panel of Figure 1. Only behind the location of the weirs a few triangulars are required to connect the grid cells in the main channel (summer bed) and the winter bed.

Results

Figure 9.11 contains the water levels at both sides of the weir, representing the water level in the river (summer bed) and in the winter bed. After about 8 hours the weir starts to overflow, yielding an increase in water level at the winter bed (see red line). After about 11 hrs the water levels in the summer bed (green and blue line) and winter bed are more or less equal. The green and blue line represent water levels at two locations very near to the weir. This figure also shows that flooding over an obliquely oriented weir occurs smoothly. The transition from critical to subcritical flow causes a very small disturbance at around 7hrs. Figure 9.12 shows a similar results for the model grid in which the weirs are nicely aligned with the grid. The water levels at both sides of the weir are also smooth. In both cases the transition from critical to subcritical flow is almost perfectly smooth.

Conclusions

This validation case shows that D-Flow FM is able to smoothly simulate flooding over an obliquely oriented weir, both over aligned and non-aligned grids.

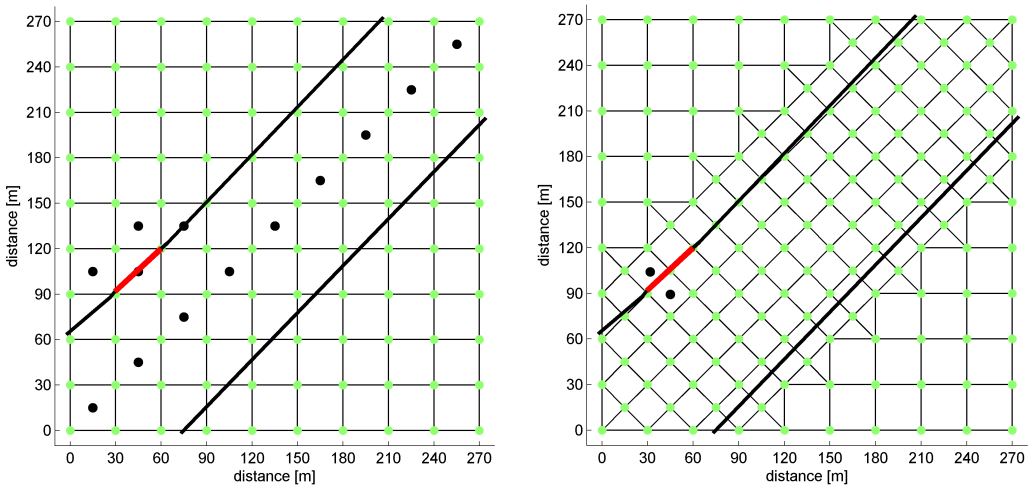


Figure 9.10: *Computational grid for the weir case. The weir, with crest level 12 m (w.r.t. reference), is represented by the red line. The black lines indicate closed walls. The observation points are represented by the black dots. The green coloring of the grid cell corner points indicate a bed level of 10 m (w.r.t. reference).*

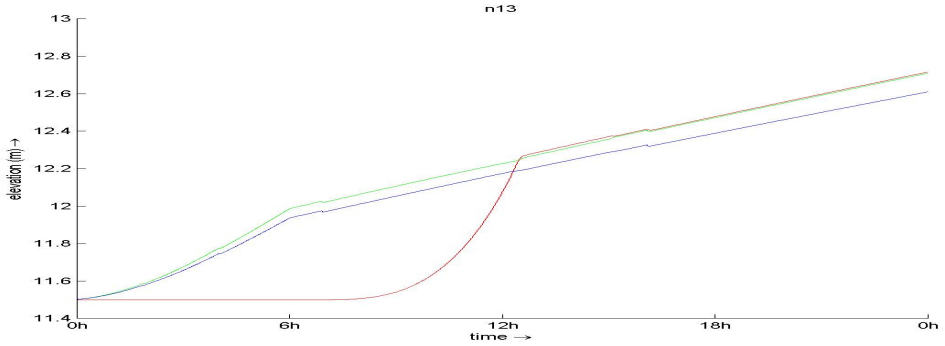


Figure 9.11: *Water levels for flow over a skew weir*

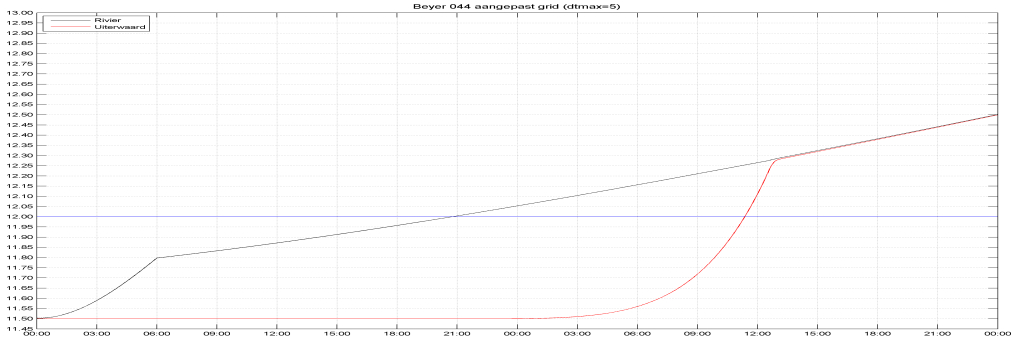


Figure 9.12: *Water levels for flow over a skew weir for an aligned grid*

Version

This test has been carried out with version dflow-fm-x64-1.1.142.40690.

9.6 Two-dimensional critical flow over an oblique weir

Purpose

In earlier test cases, the focus has been on testing weirs in D-Flow FM for quasi one-dimensional flows. In the present test case, two-dimensional critical flow over an oblique weir is considered. For modelling of fixed weirs different approaches are available. In these tescases the Villemonte approach (`fixedweirscheme=9`) is applied.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.6.1: D-Flow FM can be used to investigate the hydrodynamic impact of hydraulic structures, such as gates, weirs and barrier
- ◇ claim 2.3.6.2: D-Flow FM can accurately simulate subcritical and supercritical flows and the transition region when the flow changes from subcritical to supercritical or vice versa

Approach

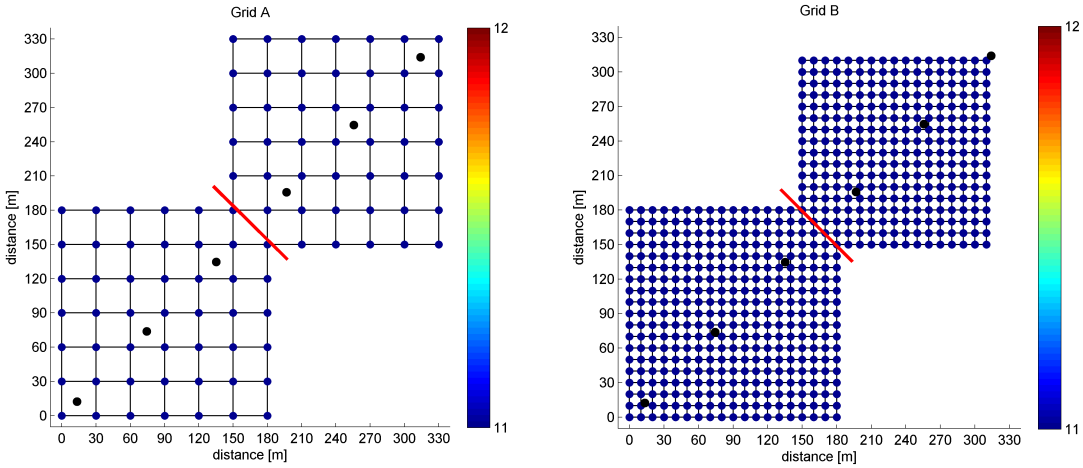
For critical flow the discharge over a weir is completely determined by the energy head upstream. Two model domains with a mesh size of 30 m and 10 m are considered, respectively. The oblique weir in these models makes a corner of either 45 $^\circ$ or 30 $^\circ$ with the grid.

Model description

The following grids are generated (see Figure 9.13, Figure 9.14 and Figure 9.15):

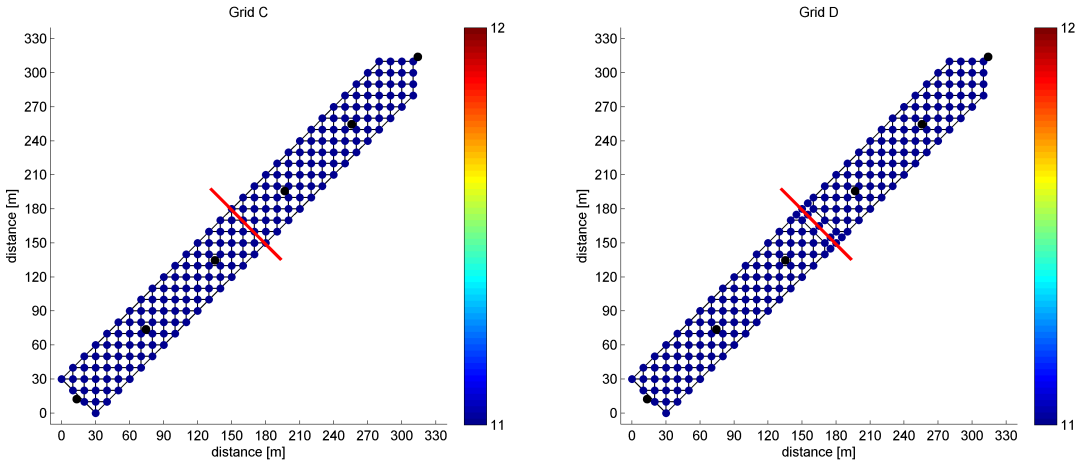
- ◇ grid A: grid cells of 30 m with an oblique weir; see Figure 9.13a,
- ◇ grid B: grid cells of 10 m with an oblique weir; see Figure 9.13b,
- ◇ grid C: uniform skew channel flow with grid cells of 30 m under 45 $^\circ$; see Figure 9.14a,
- ◇ grid D: as grid C, but with straight (1D) flow near the weir; see Figure 9.14b,
- ◇ grid F: uniform skew channel flow with grid cells of 30 m under 30 $^\circ$; see Figure 9.15a,
- ◇ grid H: as grid F, but with a mesh size of 10 m; see Figure 9.15b.

For weirs that make a corner of 45 $^\circ$ with the grid (see grids A to E), the critical flow should be a factor of $\sqrt{2}$ larger compared to the 1D weir validation case for critical flow, see Equation (9.1). This is due to the fact that the length of the weir increases with this factor. This yields $\sqrt{2} \times$ 54.92 $m^3 s^{-1}$ = 71.6 $m^3 s^{-1}$. For weirs that make a corner of 30 $^\circ$ with the grid the discharge should be $\sqrt{5} \times$ 54.92 $m^3 s^{-1}$ = 97.61 $m^3 s^{-1}$.



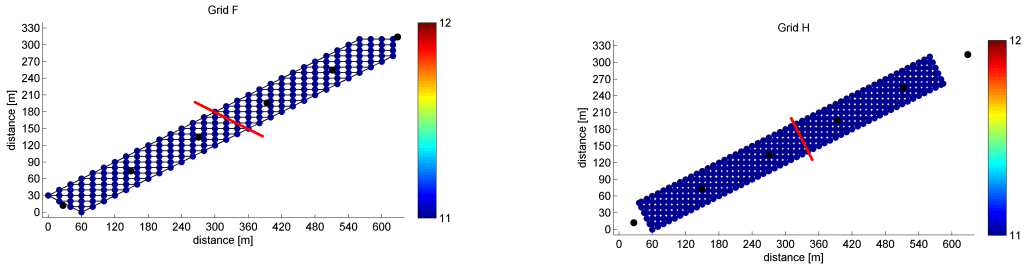
(a) *Grid A* ***(b)*** *Grid B*

Figure 9.13: *Weir grid in the form of a sandglas.*



(a) *Grid C* ***(b)*** *Grid D*

Figure 9.14: *Uniform weir flow on a skew grid under 45°.*



(a) *Grid F* ***(b)*** *Grid H*

Figure 9.15: *Uniform weir flow on a skew grid under 30°.*

Results

In Table 9.4 the computed discharges are shown for the five different model grids, in which an oblique weir that makes a corner of 45 ° with the grid. The theoretical value reads 71.6 $m^3 s^{-1}$. In Table 9.5 the computed discharges are shown for two different model grids, in which an oblique weir that makes a corner of 30 degrees with the grid. Now, the theoretical value equals 97.6 $m^3 s^{-1}$. The results in both tables show that the Villemonte subgrid approach for weirs in D-Flow FM is able to compute discharges over weirs in a reasonable accurate way in such cases. The errors are in the order of 5 to 10 %.

Table 9.4: *Discharges across the weir for the 45° variant.*

skew grid, 30 m grid	71.8 $m^3 s^{-1}$
skew grid, 10 m	72.5 $m^3 s^{-1}$
uniform skew grid, 30 m grid	74.5 $m^3 s^{-1}$
uniform skew grid with straight flow near weir, 30 m grid	76.1 $m^3 s^{-1}$

Table 9.5: *Discharges across the weir for the 30° variant.*

uniform skew grid, 30 m grid	105.7 $m^3 s^{-1}$
uniform skew grid, 10 m grid	91.7 $m^3 s^{-1}$

Conclusions

This validation case shows that the Villemonte approach (`fixedweirscheme=9`) in D-Flow FM can compute critical flow over an oblique weir in a reasonable accurate way.

Version

This test has been carried out with version dflow-fm-x64-1.2.94.66139.

10 Barriers

10.1 Gate flow

Purpose

In shallow water flow, different flow regimes occur near a gate, which depends on the downstream water level. Both subcritical and critical flow is possible. In D-Flow FM moveable structures are called 'gates'. This section focusses on gates. A gate consists of a sill and/or a gate door. Gate flow occurs between the sill and the gate, but overtopping of a gate door is possible as well. The present test cases investigate both critical and subcritical flow through a gate in case of quasi-1D flow.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.6.1: D-Flow FM can be used to investigate the hydrodynamic impact of hydraulic structures, such as gates, weirs and barrier
- ◇ claim 2.3.6.2: D-Flow FM can accurately simulate subcritical and supercritical flows and the transition region when the flow changes from subcritical to supercritical or vice versa

Approach

Four possible conditions can occur at gates. The flow is either influenced (or in other words 'restricted') by a gate door or not. In case of gate-restricted flow either critical or subcritical flow through a gate can occur. If the flow is not restricted by a gate door, we have 'free surface flow' through a gate. Then, the flow is influenced by the sill and we may have critical and subcritical flow as well. In total, this yields only four different cases, which are illustrated in Figure 10.1.

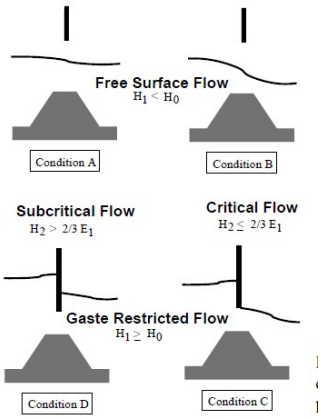


Figure 10.1: *gate conditions*

Thus, four different flow conditions cases are investigated. These cases are based on the Rajaratnam flume experiment from 1967. The D-Flow FM model results are compared with the WAQUA model results. Numerical modelling of gates in WAQUA has been extensively validated with measurements and is applied for many decades to compute water levels in the main Dutch rivers. Furthermore, the same numerical approach of so-called 'discharge

relations' from SOBEK are applied in D-Flow FM. SOBEK is the Dutch' modelling systems for 1D networks. This means that based on the water level before and after the gate, a discharge is computed from theoretical formula for the discharge. D-Flow FM uses the formulas of SOBEK, which means that there are slightly different compared to WAQUA.

Model description

The model domain has a length of 300 m and is subdivided into 30 equidistant grid cells of 10 m. At both ends there is a water level boundary. In the mid a gate is situated. By varying the water levels and by varying the lower edge of the gate door, four different cases are represented:

- ◇ model 1: subcritical flow for gate restricted flow;
- ◇ model 2: critical flow for gate restricted flow;
- ◇ model 3: subcritical free surface flow (with sill, but without gate door);
- ◇ model 4: critical free surface flow (with sill, but without gate door)

In all four models 35 cases of different combinations of upstream and downstream water levels are verified. A simulation period of 35 days is applied, so that each case lasts for one day.

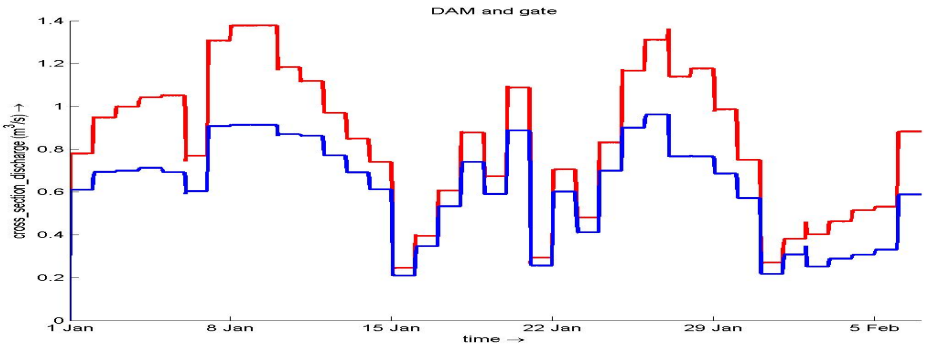


Figure 10.2: *Subcritical flow for gate restricted flow*

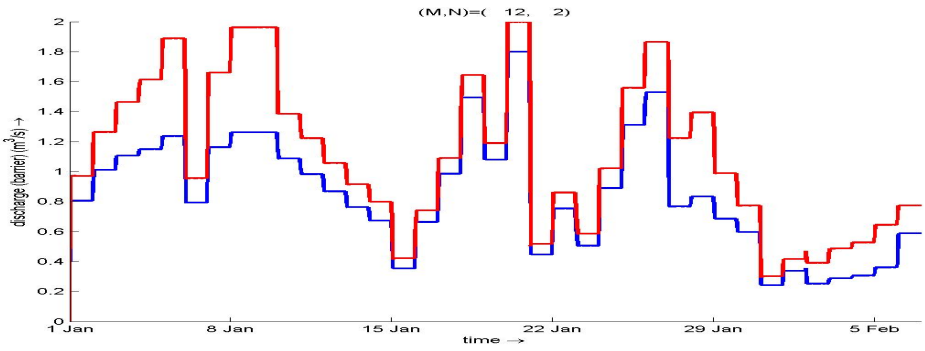


Figure 10.3: *Critical flow for gate restricted flow*

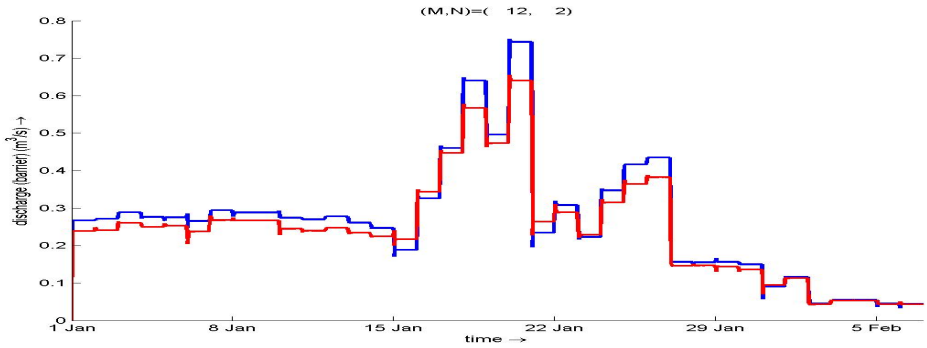


Figure 10.4: *Subcritical flow for sill*

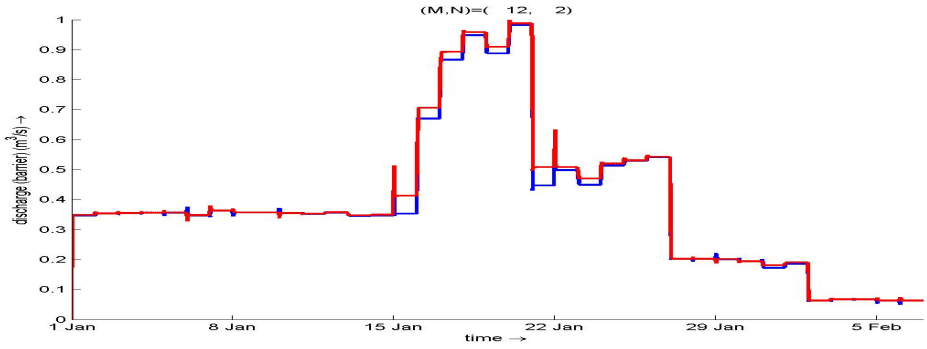


Figure 10.5: *Critical flow for sill*

Results

Figure 10.2 contains the model results for the gate restricted flow with subcritical conditions. In all four figures the discharge through the gate computed by WAQUA is in red, while the discharge computed by D-Flow FM is in blue. In Figure 10.3 the results for gate restricted flow under critical conditions are shown. Figure 10.4 contains the results for free surface subcritical gate flow due to a sill. In Figure 10.5 the results are shown for free surface critical gate flow due to a sill.

In all four figures the agreement between computed gate discharges between D-Flow FM and WAQUA is reasonable. One should take in mind that the computed water levels at both sides of the gate somewhat differ between D-Flow FM and WAQUA because of different underlying numerical schemes. Consequently, there are differences in discharge as well. Differences also occur due to the slightly different theoretical discharge relations that are applied to compute the discharge through a gate. D-Flow FM uses the discharge formulations of SOBEK, while WAQUA uses slightly different formulations.

Conclusions

This validation case shows that D-Flow FM is able to compute the discharge flow through a gate in a reasonably accurate way. Qualitatively there is a good agreement with the WAQUA model results. Quantitatively differences occur due to the different numerical schemes and the slightly different theoretical formulas for the discharge through a gate. This holds for all gate conditions, namely for gate restricted flow and free surface flow, both under subcritical and critical conditions.

Version

This test has been carried out with version dflow-fm-x64-1.1.149.41664.

11 Boundary conditions

11.1 Time dependent boundary condition

Purpose

In many hydraulic computations, the boundary conditions are time dependent. This test case examines whether the D-Flow FM correctly facilitates the coupling with time dependent boundary conditions.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.8.1: D-Flow FM can deal with boundary conditions that are prescribed as time-series, as harmonic components and as astronomic components
- ◇ claim 2.3.8.2: D-Flow FM can deal with boundary conditions for the water level, the velocity and the discharge

Approach

The case is kept as simple as possible. A rectangular domain is chosen with one single open boundary condition which is time dependent. A Delft3D computation is run as a backdrop to check if the resulting time series collapse.

Model description

A domain of sizes 130 m $\times$ 1 m is covered by a grid containing 130 $\times$ 1 cells. At one of the two short edges of the domain, a time dependent water level boundary is imposed. At the other three edges, the boundary is closed and frictionless. The water level signal imposed at the open boundary, is shown in Figure 11.1.

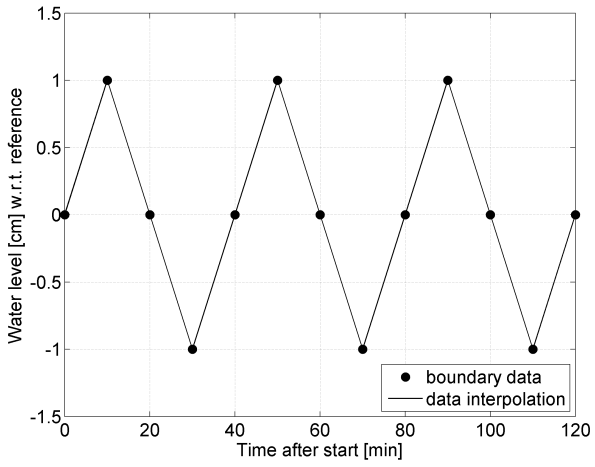


Figure 11.1: *Imposed data on the open boundary. The dots indicate the actual data; the intermediate lines indicate the interpolation in time.*

The following settings are inserted:

- ◇ the bottom level of the domain is set equal to -0.2 m w.r.t. the reference level,
- ◇ the initial water level is 0 m w.r.t. the reference level,

- ◇ the bottom friction is set equal to 60 $m^{1/2}/s$ (as Chézy coefficient),
- ◇ the horizontal eddy viscosity is set equal to 0 m^2/s.

A Delft3D computation is used to compare the D-Flow FM output with. The settings of the Delft3D computation is chosen exactly the same as the D-Flow FM computation to ensure sound comparison.

Results

In each grid cell an observation point is present. The observation point at the center of the domain is chosen for comparison of the D-Flow FM output with the Delft3D output. The water level timeseries and streamwise velocity timeseries from both packages are shown in Figure 11.2.

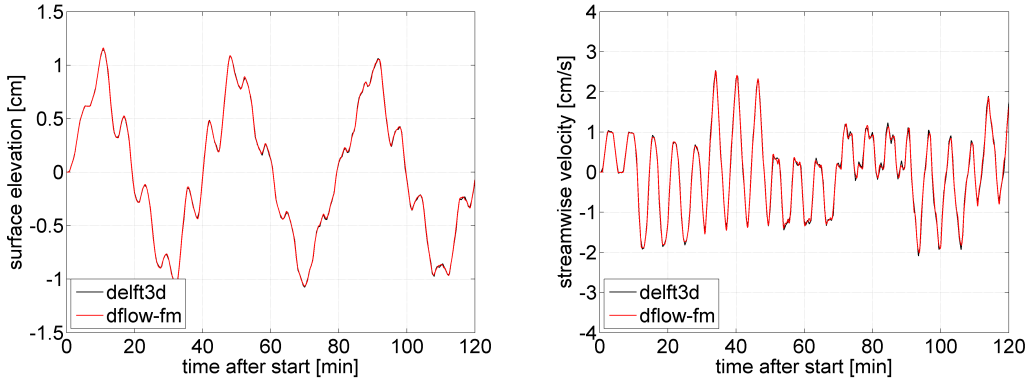


Figure 11.2: *Water level evolution (left panel) and streamwise velocity evolution (right panel) at the observation point in the very center of the channel.*

Figure 11.2 shows good agreement between the D-Flow FM results and the Delft3D results indicating the proper working of the time dependent water level boundary condition.

One important remark has to be made. Suppose that the boundary conditions are specified in time from t_{b1} to t_{b2} and that the computational time runs from start time t_{c1} to stop time t_{c2}, then care should be taken in case:

- ◇ $t_{c1} < t_{b1}$, then the boundary condition is set to zero for $t_{c1} \leq t < t_{b1}$,
- ◇ $t_{c2} > t_{b2}$, then the boundary condition value is extrapolated for $t_{b2} \leq t < t_{c2}$.

Conclusion

The time dependent boundary conditions functionality appears to work properly. In case the computational start/stop times do not match with the boundary condition times, more safety should be implemented in the code.

Version

This test has been carried out with version dflow-fm-x64-1.1.90.31666.

11.2 Harmonic boundary condition

Purpose

In many hydraulic computations, the boundary conditions are prescribed through harmonic components. This test case examines whether the D-Flow FM correctly facilitates this functionality.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.8.1: D-Flow FM can deal with boundary conditions that are prescribed as time-series, as harmonic components and as astronomic components
- ◇ claim 2.3.8.2: D-Flow FM can deal with boundary conditions for the water level, the velocity and the discharge

Approach

The case is kept as simple as possible. A rectangular domain is chosen with one single open boundary condition which is time dependent. A Delft3D computation is run as a backdrop to check if the resulting time series collapse.

Model description

A domain of sizes 130 m $\times$ 1 m is covered by a grid containing 130 $\times$ 1 cells. At one of the two short edges of the domain, a time dependent water level boundary is imposed. At the other three edges, the boundary is closed and frictionless. In Delft3D, harmonic boundaries are prescribed by means of `.bch`-files; in D-Flow FM, harmonic boundaries are prescribed by means of `.cmp`-files.

The following settings are inserted:

- ◇ the bottom level of the domain is set equal to -0.2 m w.r.t. the reference level,
- ◇ the initial water level is 0 m w.r.t. the reference level,
- ◇ the bottom friction is set equal to 60 $m^{1/2}$/s (as Chézy coefficient),
- ◇ the horizontal eddy viscosity is set equal to 0 m^2/s.

The Delft3D settings for the boundary condition, as `.bch`-file, is as follows:

```
        0.0000    2160.0000    4320.0000    6480.0000

         0.001        0.005        0.004        0.001
         0.001        0.005        0.004        0.001

                       90.0         24.0        -10.0
                       90.0         24.0        -10.0
```

Remark that in the `.bch`-file, the periodicity is prescribed through a frequency with unit [degrees/hour]. In D-Flow FM, the periodicity is prescribed through the period time span in minutes. Hence, the frequency 2160 degrees/hour turns to 60 min/hour $\times$ 360 degrees/period $/$ 2160 degrees/hour $=$ 10 minutes. The D-Flow-FM settings for the boundary condition, as `.cmp`-file, is as follows (period [min] – amplitude [m] – phase shift [degrees]):

```
        0.0000000        0.0010000        0.0000000
       10.0000000        0.0050000       90.0000000
        5.0000000        0.0040000       24.0000000
        3.3333333        0.0010000      -10.0000000
```

A Delft3D computation is used to compare the D-Flow FM output with. The settings of the Delft3D computation is chosen exactly the same as the D-Flow FM computation to ensure sound comparison. Output results at each observation point are writte each 6 seconds.

Results

In each grid cell an observation point is present. The observation point at the center of the domain is chosen for comparison of the D-Flow FM output with the Delft3D output. The water level timeseries and streamwise velocity timeseries from both packages are shown in Figure 11.3.

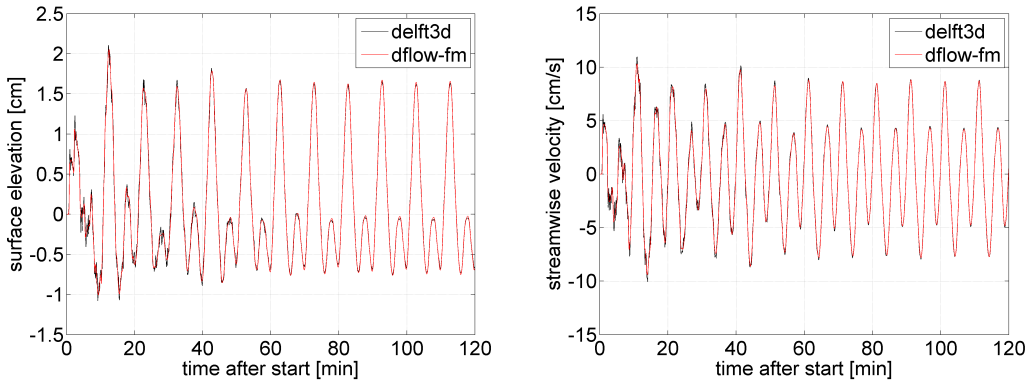


Figure 11.3: *Water level evolution (left panel) and streamwise velocity evolution (right panel) at the observation point in the very center of the channel.*

Figure 11.3 shows good agreement between the D-Flow FM results and the Delft3D results regarding the reproduction of the four elementary components. This indicates the proper working of the time dependent water level boundary condition. Rather small deviations only comprise some spin-up wiggle-like fluctuations in the first 30 minutes.

Conclusion

The harmonic boundary conditions functionality appears to work properly.

Version

This test has been carried out with version dflow-fm-x64-1.1.90.31666.

11.3 Astronomic boundary condition

Purpose

In many geophysical computations, the boundary conditions are prescribed through astronomic components. This test case examines whether the D-Flow FM correctly facilitates this functionality.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.8.1: D-Flow FM can deal with boundary conditions that are prescribed as time-series, as harmonic components and as astronomic components
- ◇ claim 2.3.8.2: D-Flow FM can deal with boundary conditions for the water level, the velocity and the discharge

Approach

The case is kept as simple as possible. A rectangular domain is chosen with one single open boundary condition which is time dependent. A Delft3D computation is run as a backdrop to check if the resulting time series collapse.

Model description

A domain of sizes about 445 km $\times$ 4.45 km is covered by a grid containing 41 $\times$ 4 cells. At the two short edges of the domain, a time dependent water level boundary is imposed. At the other two edges, the boundary is closed and frictionless. In Delft3D, astronomic boundaries are prescribed by means of `.bca`-files; in D-Flow FM, astronomic boundaries are prescribed by means of `.cmp`-files. The computation is run in spherical coordinates.

The following settings are inserted:

- ◇ the bottom level of the domain is set equal to -10.0 m w.r.t. the reference level,
- ◇ the initial water level is 0 m w.r.t. the reference level,
- ◇ the bottom friction is set equal to 0.024 (as Manning coefficient),
- ◇ the horizontal eddy viscosity is set equal to 25.0 m^2/s.

At the left boundary (water level boundary), two signals are prescribed: a M2-signal with an amplitude equal to 0.5 m and phase equal to 0° and a S2-signal with an amplitude equal to 1.0 m and a phase equal to 0°.

At the right boundary (water level boundary), two signals are prescribed: a M2-signal with an amplitude equal to 0.5 m and phase equal to 0° and a S2-signal with an amplitude equal to 1.0 m and a phase equal to 11.2047°.

A Delft3D computation is used to compare the D-Flow FM output with. The settings of the Delft3D computation is chosen exactly the same as the D-Flow FM computation to ensure sound comparison. Output results at each observation point are writte each 10 minutes.

Results

Along the centerline of the grid, five observation points are present. The observation point at the center of the domain is chosen for comparison of the D-Flow FM output with the Delft3D output. The water level time series and streamwise velocity timeseries from both packages are shown in Figure 11.4.

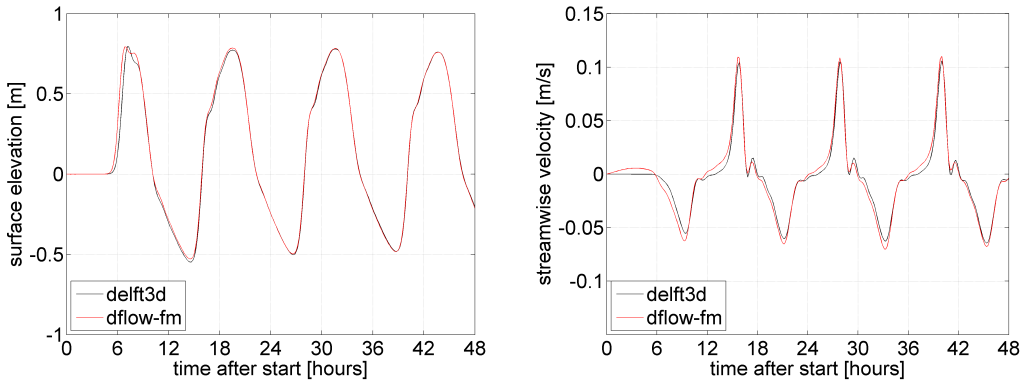


Figure 11.4: *Water level evolution (left panel) and streamwise velocity evolution (right panel) at the observation point in the very center of the channel.*

Figure 11.4 shows fair agreement between the D-Flow FM results and the Delft3D results regarding the reproduction of the two elementary astronomic components. This indicates the fair working of the time dependent water level boundary condition. A difference is seen when the water at the center is set in motion: the water level increase is slightly differently predicted by the two models.

For the streamwise velocities, the deviations between the two packages are relatively large, regarding the simplicity of the testcase. Particularly, it is remarkable that in the first 6 hours of the simulated time, the D-Flow FM velocities are non-zero, whereas the Delft3D velocities are exactly zero and the waterlevel has not been affected by the external forcing yet.

Conclusion

The astronomic boundary conditions functionality appears to work properly. However, the deviations in the streamwise velocities are relative large regarding the simplicity of the testcase.

Version

This test has been carried out with version dflow-fm-x64-1.1.90.31666.

11.4 Salinity boundary condition

Purpose

If salinity plays a role in a hydraulic computation, boundary conditions for the salinity have to be prescribed as well, obviously. This prescription can be done either through time-series files or component files. This test case examines whether the D-Flow FM correctly facilitates this functionality.

Linked claims

Claims that are related to the current test case are:

◇ claim 2.3.7.2: D-Flow FM can be used for an accurate prediction of the density driven flow

Approach

The case is kept as simple as possible. A rectangular domain is chosen with two open boundary conditions (water level as a sea and discharge as a river) which are both time dependent. A Delft3D computation is run as a backdrop to check if the resulting time series collapse.

Model description

A domain of sizes 130 m $\times$ 1 m is covered by a grid containing 130 $\times$ 1 cells. At the left short edges of the domain, a time dependent water level boundary is imposed by means of a `.cmp`-file containing one harmonic component. At the right short edge, a discharge boundary is imposed by means of a `.cmp`-file containing five harmonic components. At the other two edges, the boundary is closed and frictionless. At the water level boundary, a constant salinity of 12.5 ppt is prescribed; at the discharge boundary, a constant salinity of 0.0 ppt is prescribed.

The following settings are inserted:

◇ the bottom level of the domain is set equal to -0.2 m w.r.t. the reference level,
◇ the initial water level is 0 m w.r.t. the reference level,
◇ the bottom friction is set equal to 60 $m^{1/2}$/s (as Chézy coefficient),
◇ the horizontal eddy viscosity is set equal to 0 m^2/s.

A Delft3D computation is used to compare the D-Flow FM computation with. The settings of the Delft3D computation are chosen exactly the same as the D-Flow FM computation to ensure sound comparison. Output results at each observation point are written each 6 seconds.

Results

In each grid cell an observation point is present. The observation point 20 km from the water level boundary (and hence 90 km from the discharge boundary) is chosen for comparison of the D-Flow FM output with the Delft3D output. First, two computations are run as a backdrop: one computation with constant zero salinity and one computation with a constant salinity of 12.5 ppt at both the two boundaries as well as initially. The results for the water level in these two cases is shown in Figure 11.5: in the left panel with 0 ppt at the boundaries (and as initial condition) and in the right panel with 12.5 ppt at the boundaries (and as initial condition).

Figure 11.5 shows perfect agreement of the water levels from the D-Flow FM results with the Delft3D results, both with 0 ppt and 12.5 ppt. This indicates the proper processing of the time dependent water level and discharge boundary signals.

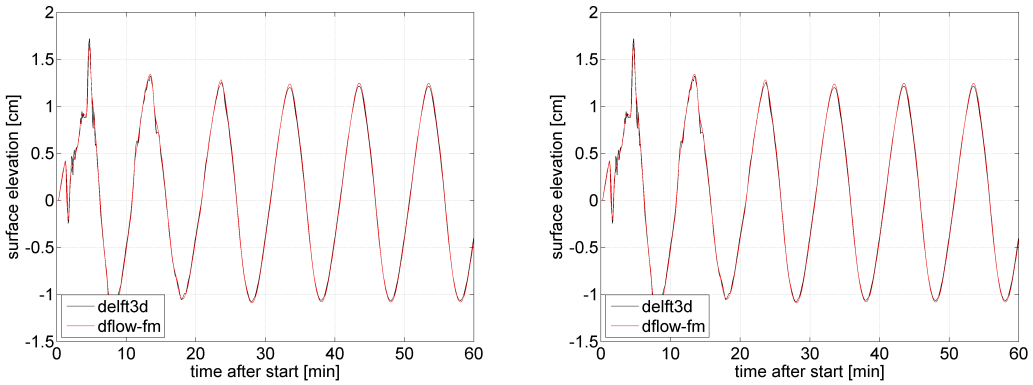


Figure 11.5: *Water level evolution at the observation point 20 km near the water level boundary. The results stem from a computation without salinity (left panel) and a computation with constant salinity (right panel); hence, 0 ppt (left) and 12.5 ppt (right), respectively, at boundaries and as initial value.*

Figure 11.6 shows the results of the computation if variable salinity actually is included, i.e. 12.5 ppt at the water level boundary and 0 ppt at the discharge boundary. The initial salinity field yields 0 ppt. At the considered observation point, the salinity evolution shows a block-like development. The results from D-Flow FM show smooth blocks, whereas the Delft3D results show some wiggles.

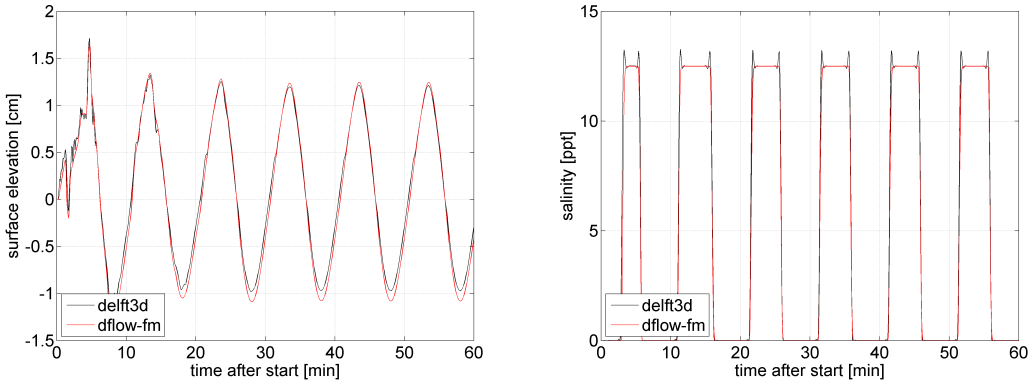


Figure 11.6: *Water level evolution (left panel) and salinity evolution (right panel) at the observation point 20 km near the water level boundary.*

It is remarkable that the water levels do not coincide anymore: the D-Flow FM results and the Delft3D results deviate to typical values of 10% in amplitude of the water level. This deviation is absent in Figure 11.5. The comparison of Figure 11.6 with Figure 11.5 shows unaffected D-Flow FM results and affected Delft3D results with changing boundary condition configuration. The specific boundary condition functionality for the salinity itself appears to work properly.

Conclusion

The salinity boundary conditions functionality appears to work properly.

Version

This test has been carried out with version dflow-fm-x64-1.1.90.31666.

11.5 Qh-boundary conditions

Purpose

The purpose of this validation case is to examine the performance of D-Flow FM for a schematized homogeneous channel flow simulation where the downstream boundary condition has been prescribed using a Qh-boundary type, in which a water level boundary condition is prescribed based on the discharge passing through the outflow boundary section. In this sense, this type of boundary condition could also be referred to as a $Q\zeta$-boundary condition, since ζ represents the water level with respect to the reference level.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.8.1: D-Flow FM can deal with boundary conditions that are prescribed as time-series, as harmonic components and as astronomic components
- ◇ claim 2.3.8.3: D-Flow FM can deal with boundary conditions for derived quantities by means of a Qh-table, a Neumann-type condition for the water level and a Riemann invariant

Approach

A flow setup is chosen for which an analytical solution is known. The flow setup selected is a quasi one-dimensional channel flow which homogeneous friction and an equilibrium between the pressure gradient the bed friction, provided a discharge prescribed upstream and a water level prescribed downstream. The water level described downstream is computed from a Qh-table.

Model description

Three grids are developed to be used to compute the quasi one-dimensional channel flow on. These grids are visualized in Figure 11.7. The three testcases concern a straight channel flow with a constant bed slope and constant roughness factor under equilibrium flow conditions.

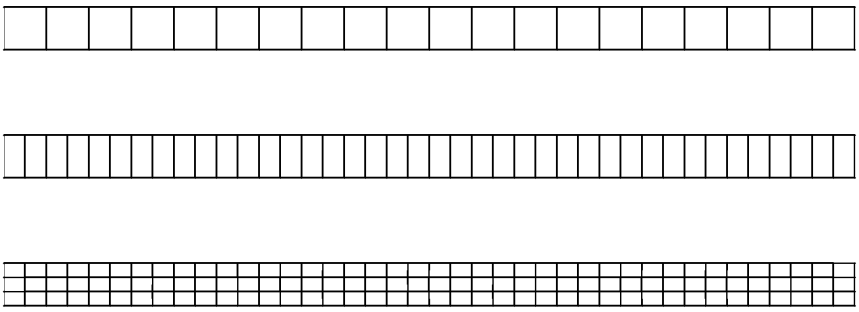


Figure 11.7: *Setup of the grids for the testing of the QH-boundary, flow is from left to right.*

A discharge Q equal to 2500 m^3/s is prescribed. The bottom slope i_b is prescribed to be 10^{-4}. The bottom level at the inflow boundary is 0 m (w.r.t. reference) and at the outflow boundary -1 m (w.r.t. reference). The Chézy friction coefficient C is set to 65 m$^{1/2}$/s. At the downstream boundary the equilibrium flow depth is prescribed which is computed by

$$h_e = \left(\frac{Q}{BC\sqrt{i_b}} \right)^{2/3} = 3.89677. \tag{11.1}$$

The outflow boundary is provided as a Qh-table:

```
* Downstream Q-h water level boundary
* Discharge (m3/s)
* Water level (m)
*
   0    1.846766937769912
2500    2.846766937769912
5000    3.846766937769912
7500    4.846766937769912
```

The prescription of the outflow water level values requires some explanation. Since, the equilibrium depth is 3.89677 m, the relation $Q = 2500$ m^3/s versus $h = 3.89677$ should be part of the Qh-table. However, since D-Flow FM uses ghost-cells at the boundary (mirrored cell-centers), the artifically added length of $1 \cdot \Delta x$ should be taken into account when considering the total length of the channel.

With respect to the mirroring of cell-center locations at the boundaries, one additional remark should be made. The center of the fictitous boundary cells (ghost cells) is computed on the basis of two parameters, namely d_j and b_A, being the distance from the inside cell-center location to the grid rim and the area of the cell, respectively. The distance from the rim to the fictitious cell-center is set as $\max(d_j, \frac{1}{2}\sqrt{b_A})$. For the grids shown in Figure 11.7, this mirrored cell-center is hence on d_j (exact mirror), $\frac{1}{2}\sqrt{b_A}$ and d_j (exact mirror), respectively. The values for the outflow Qh-tables are modified in line with this location.

Results

The result for one of the grids in Figure 11.7 - the upper one - is visualized in Figure 11.8 against the backdrop of the analytical solution. The root-mean-square error with respect to the analytical solution is presented as well.

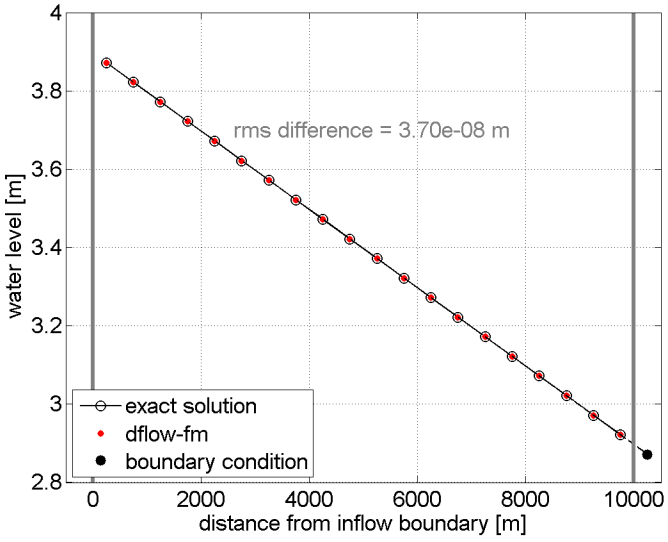


Figure 11.8: *Analytical and computed waterlevel versus the distance from the inflow boundary. The results are shown for the coarsest grid.*

For all the three grids, the root-mean-square errors are:

- ◇ the coarsest grid: $3.70 \cdot 10^{-8}$ m,
- ◇ the refined (one direction) grid: $3.61 \cdot 10^{-3}$ m,
- ◇ the refined (two directions) grid: $3.69 \cdot 10^{-8}$ m.

On the basis of these numbers, two remarks can be made. Firstly, it is seen that the Qh-table approach appears to work properly. Secondly, it is seen that the rms-error is the largest for the second grid. Perhaps not coincidentally, this is the only grid that does *not* use the exact mirrored cell-center at the boundary.

Conclusion

The Qh-table boundary condition type returns the expected results for one-dimensional channel flow, which corroborates the proper working of this type of boundary condition prescription.

Version

This test has been carried out with version dflow-fm-x64-1.1.137.40169.

11.6 Stationary flow with Neumann type outflow boundary

Purpose

The purpose of this validation case is to examine the working of Neumann type boundary conditions. This type of boundary conditions has been implemented for the water level, i.e. $\partial\zeta/\partial x_n = 0$ at the boundary.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.8.1: D-Flow FM can deal with boundary conditions that are prescribed as time-series, as harmonic components and as astronomic components
- ◇ claim 2.3.8.3: D-Flow FM can deal with boundary conditions for derived quantities by means of a Qh-table, a Neumann-type condition for the water level and a Riemann invariant

Approach

For this Neumann type boundary case, a straight channel is chosen derived from the Chézy friction type test case. For this particular case, an analytical solution is known. The slope of the bottom varies linearly with a slope i_b. Given a discharge Q and a friction coefficient (in this case, Chézy's factor C is used), the water depth h_e, given as:

$$h_e = \left(\frac{Q}{BC\sqrt{i_b}} \right)^{2/3} \tag{11.2}$$

can be computed as the equilibrium water depth.

Model description

For this test case, one particular computational grid is generated. The grid is of Cartesian type. The longitudinal size L of the domain is 10 km, whereas the lateral size B of the domain is 500 m. In longitudinal direction 20 cells are deployed, whereas in lateral direction only one grid cell is present.

A discharge Q equal to 2500 m^3/s is prescribed at the inflow boundary. The bottom slope i_b is prescribed to be 10^{-4}. The bottom level at the inflow boundary is 0 m (w.r.t. reference) and at the outflow boundary -1 m (w.r.t. reference). The Chézy friction coefficient C is set to 65 m$^{1/2}$/s. Given the above values, the equilibrium water depth h_e is computed as:

$$h_e = \left(\frac{2500}{500 \cdot 65 \cdot \sqrt{1.0 \cdot 10^{-4}}} \right)^{2/3} = 3.8967669 \text{ m}. \tag{11.3}$$

The computational time step is fixed at 10 seconds. At the outflow boundary, a Neumann condition is prescribed: $\partial\zeta/\partial x_n = -i_b$. At the inflow, `jbasqbnddownwindhs = 1` is prescribed.

Results

The actual water level differences (numerical versus analytical) at cell face points along the channel are shown in Figure 11.9. Figure 11.9 reveals that the slopes of the free surface agree well and that the absolute water levels themselves coincide as well. From this, it can be concluded that the Neumann outflow boundary condition correctly does what it is supposed to do.

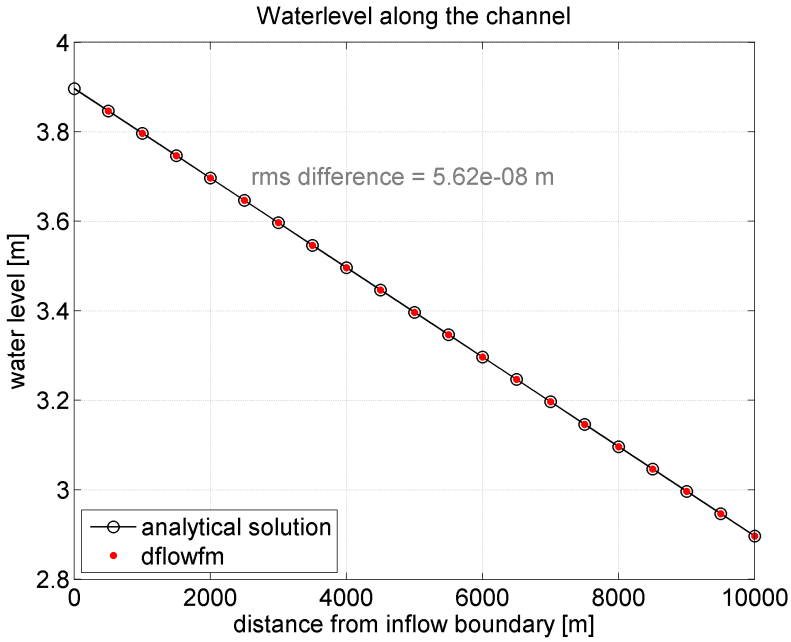


Figure 11.9: *Spatial development of the water level along the center axis of the channel. The water levels are visualized at the cell faces.*

The way of visualizing the water level, as done in Figure 11.9, requires some explanation. Thereto, consider an arbitrary computational cell, as shown in Figure 11.10. Recall that the principle variables are the water level (located in the cell center) and the face-normal velocities (located at the center of the faces). The level of the bed is given in the cell corners.

To prescribe the bed friction (defined at flow links), D-Flow FM needs information on the hydraulic radius. This information can be provided by the user through the `mdu`-file key `Conveyance2D`. By default, D-Flow FM uses an analytic 2D conveyance approach. As an alternative, the user could specify simply the waterdepth as hydraulic radius. In order to resolve the face-normal velocity, D-Flow FM utilizes the (locally) upwind water level for the prescription of the hydraulic radius for the bed friction.

Consider, for instance, velocity u_C. Suppose, the flow is unidirectional with an orientation

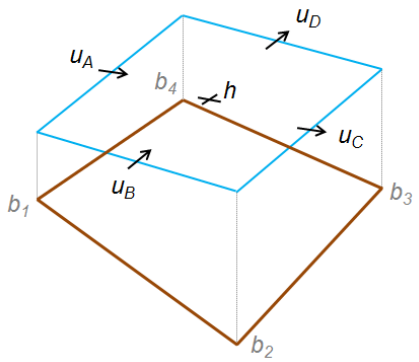


Figure 11.10: *Computational cell with surface level h, face velocities u_A, u_B, u_C and u_D and bed levels b_1, b_2, b_3 and b_4.*

exactly aligned with the orientation of u_C. In that case, the surface level h, as given in Figure 11.10, is used as surface elevation at the cell face center. The bed level applied to this location comprises the mean, the minimum or the maximum value of the quantity-set (b_2, b_3), depending on the choice for the keyword `izbndpos` in the associate mdu-file: `3`, `4` or `5`, respectively.

For the testcase under consideration, in which:

- $b_1 = b_4$,
- $b_2 = b_3$,
- $u_B = u_D = 0$ m/s and
- u_A should equal u_C,

the waterdepth is equal to $d = h - b_2 = h - b_3$. This value for the waterdepth is compared with the analytical value (provided by Equation (11.3)) and redrawn as water level at the cell faces in Figure 11.9. As shown by the root-mean-square value in Figure 11.9, the differences between the numerical outcomes and the analytical value can be considered small.

Conclusion

For a stationary hydraulic computation of flow over an inclined bottom (with friction, a discharge inflow boundary and a Neumann outflow boundary), D-Flow FM is able to accurately reproduce the water surface slope, as well as the absolute water levels.

Version

This test has been carried out with version dflow-fm-x64-1.1.90.31666.

11.7 Non-stationary flow with Neumann type boundary

Purpose

By means of a Neumann boundary condition, the desired value of the gradient of the water level can be prescribed at a boundary. The purpose of this test case is to investigate if D-Flow FM correctly deals with Neumann type boundary conditions in case of the simulation of a non-stationary flow in a rectangular box.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.8.1: D-Flow FM can deal with boundary conditions that are prescribed as time-series, as harmonic components and as astronomic components
- ◇ claim 2.3.8.3: D-Flow FM can deal with boundary conditions for derived quantities by means of a Qh-table, a Neumann-type condition for the water level and a Riemann invariant

Approach

Since a stationary flow case has already been considered, a non-stationary case is considered now. To that end, a two-dimensional rectangular box with one open boundary is defined. The boundary condition at the open boundary is intended to fill the domain with water. At one of the boundaries perpendicular to the open boundary, a Neumann boundary is prescribed, whereas at the other boundary perpendicular to the open boundary, a free slip boundary condition is imposed.

Model description

A domain of sizes 100 m $\times$ 40 m is considered. The boundary conditions and initial conditions are chosen as shown in Figure 11.11. The domain is covered by 10 $\times$ 4 grid cells.

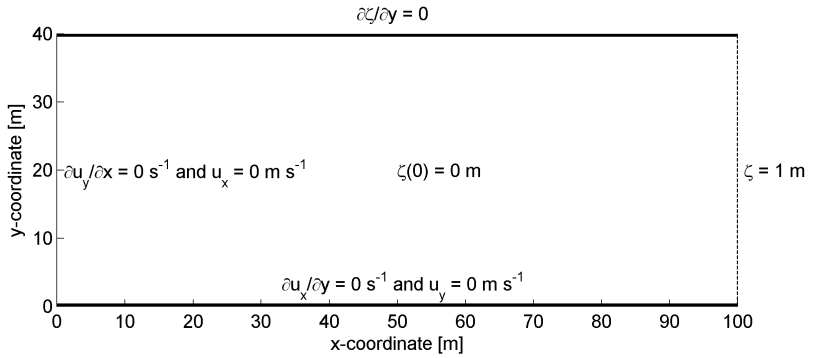


Figure 11.11: *Flow geometry and imposed boundary boundary conditions and initial condition.*

The upper boundary is of Neumann-type, the left and lower boundaries are of free slip type (`irov = 0`). At the open boundary (the right boundary), a constant water level equal to 1 m (w.r.t. the reference level) is prescribed. The initial condition yields a water level equal to 0 m. Thus, after start a non-stationary flow comes into existence representing the filling of the box with water.

The following settings are inserted:

- ◇ the bottom level of the domain is set equal to -5.0 m w.r.t. the reference level,
- ◇ the bottom friction is set equal to 0.023 (as Manning coefficient),
- ◇ the horizontal eddy viscosity is set equal to 1.0 m^2/s.

Results

After 10 seconds, the water level and the velocity vectors look like shown in Figure 11.12. The basic check whether the test case succeeds is to check the water level in the ghost cell, just outside the domain. Since this these data are not available in the standard map-files, these values can only be checked in the development environment of D-Flow FM. This check has been performed yielding successful outcomes, though not visualized in this validation document.

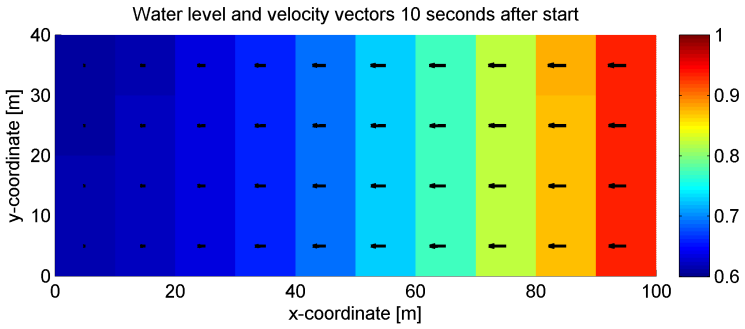


Figure 11.12: *Water level and velocity vectors 10 seconds after the start of the computation.*

In order to provide another way of proving the validity of the case, the water levels and velocity vectors are shown in Figure 11.12. Since $\partial\zeta/\partial x_n = 0$ is applied at the upper boundary, a free-slip boundary is present at the lower boundary and a constant bed level is present in the entire domain, no lateral flow is induced. Hence, a (nearly) symmetrical flow field is expected. The flow behavior exposed in Figure 11.12 is hence expected.

Conclusion

The Neumann boundary condition test for non-stationary flow yields results that coincide with the expectations: the imposed water level gradient at the Neumann boundary is met along the full duration of the computation.

Version

This test has been carried out with version dflow-fm-x64-1.1.90.31666.

11.8 Weakly reflecting boundary conditions in one dimension

Purpose

At a Riemann boundary we do not allow any outgoing perturbation with respect to some reference boundary state to reflect back from the boundary. This can be achieved by prescribing the incoming Riemann invariant. In D-Flow FM, a Riemann invariant at the boundary can be imposed by means of prescribing the reference *boundary water level*, using linear theory. Reflections can hence be reflected for *small* perturbations. In the present test, it is checked whether or not these reflections actually do not occur. In this test case, a one-dimensional case is considered.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.8.1: D-Flow FM can deal with boundary conditions that are prescribed as time-series, as harmonic components and as astronomic components
- ◇ claim 2.3.8.3: D-Flow FM can deal with boundary conditions for derived quantities by means of a Qh-table, a Neumann-type condition for the water level and a Riemann invariant

Approach

A quasi-1D channel is developed with a flat bed and a flat initial water level. At both the ends of the channel, Riemann boundary conditions are imposed with a reference water level associated with the flat initial water level field. Perturbations are induced by lifting the initial water level in the center of the channel. The flow, thus induced, consists of two propagating waves: one to the left and one to the right. The focus of this test is to measure the extent to which the wave is reflected at the two boundaries.

Model description

For this test, two domains are developed: a short domain and an elongated variant of it. The short domain has a length of 10 km and a width of 500 m. This grid is covered by 40×3 square cells. The elongated domain has a length of 3 $\times$ 10 km and a width of 500 m. The elongated grid is covered by 120×3 square cells.

The bed is flat at a level equal to 0 m (with respect to the reference level). The initial water level is 10 m (with respect to the reference level) everywhere. The perturbation is given a Gaussian shape, to prevent sharp moving fronts. Hence, the initial water level reads:

$$h(x, t = 0) = \exp\left(-\left(\frac{x - L/2}{c}\right)^2\right) + h_0$$

with x from 0 to L, with $L = 10$ km, $h_0 = 10$ m (w.r.t. reference) and $c = 1$ km. Along the domain, 8 observation points are placed to monitor the development of the water level in time. The initial water level and the observation points are shown in Figure 11.13.

Processes associated with diffusion and bed friction are turned off. The simulated time is 3600 seconds. The wave propagation speed $\sqrt{gh}$ equals about 10 m/s. Travelling along half the domain takes about 5000 m$/10$ m/s $= 500$ s for the wave in the short/small domain, whereas 15000 m$/10$ m/s $= 1500$ s for the large/elongated domain. Notice that the perturbation yields 10% of the water depth.

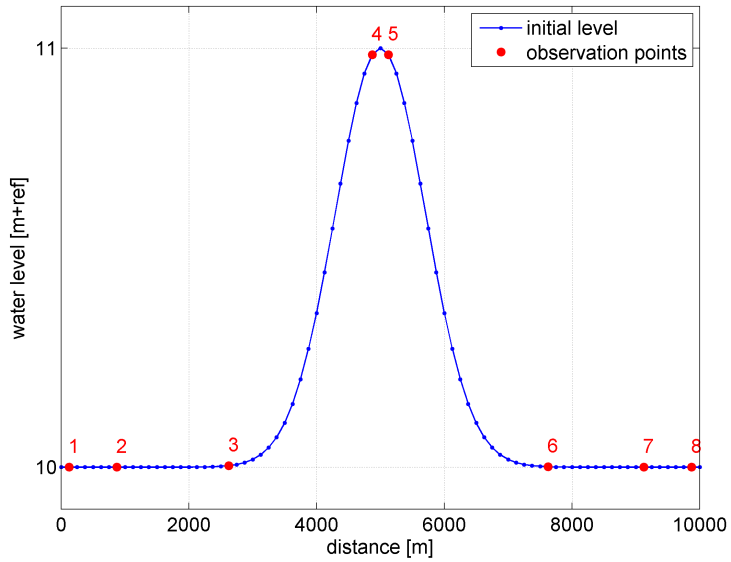


Figure 11.13: *Initial water level along the domain. The observation point locations are given in red.*

Results

In Figure 11.14, time series are provided of the computed water level at several stations in the domain. It is seen that the initially induced wave can leave the domain without major reflections; the reflection at the boundary is minor with an amplitude of about 2 cm. The reflections due to the boundary (which is only *weakly* reflective, because of the linearization) is directly visible through the small-amplitude ripples.

If the center station (ID = 5) is considered, the presence of the reflected wave can easily be recomputed. For the small domain, the peak occurs at 1000 s after start (equal to twice 500 s needed to travel forth and back); for the large domain, the peak occurs at 3000 s after start (equal to twice 1500 s).

For the other stations, a similar behavior is observed. In these cases, small amplitude waves are represented in the time series signal. It is left to the reader to recompute the occurrence of the peak times by hand.

Conclusion

Riemann-type boundary conditions are successfully facilitated by D-Flow FM. The Riemann-type boundary condition is weakly reflective; hence, some reflection is present if the perturbation is relatively large.

Version

This test has been carried out with version dflow-fm-x64-1.1.191.47129.

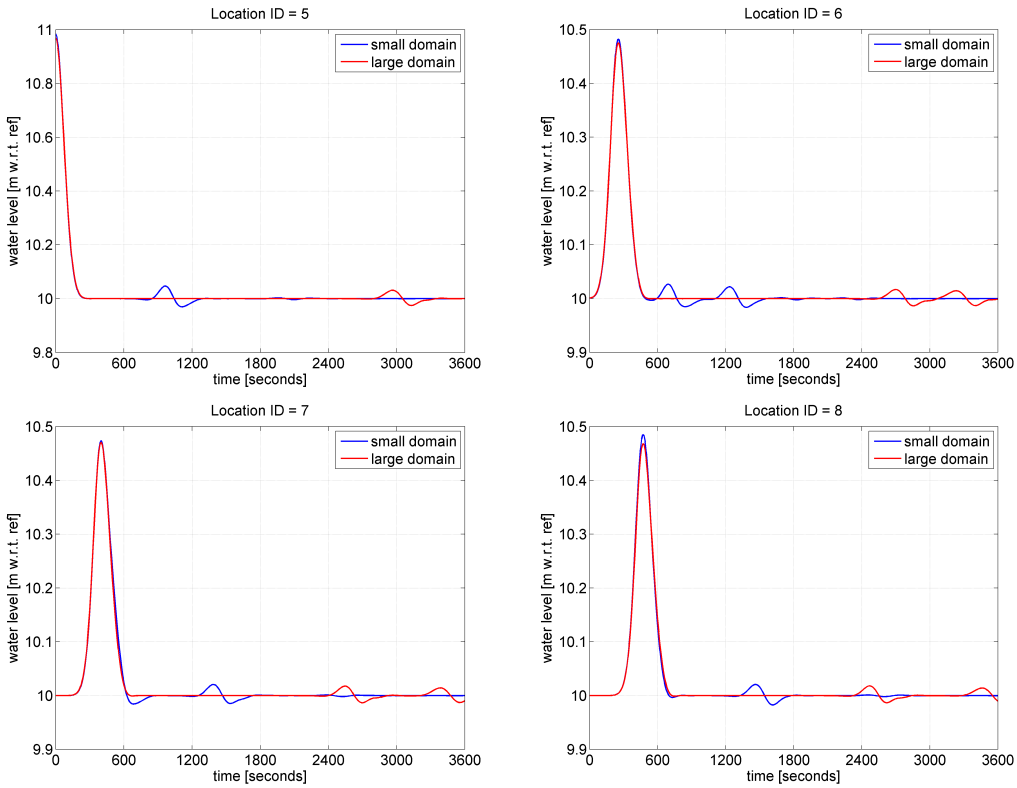


Figure 11.14: *Time series of the computed water level at the observation point locations. The results for the small/short domain are given in blue; the results for the large/elongated domain are given in red.*

11.9 Weakly reflecting boundary conditions in two dimensions

Purpose

At a Riemann boundary we do not allow any outgoing perturbation with respect to some reference boundary state to reflect back from the boundary. This can be achieved by prescribing the incoming Riemann invariant. In D-Flow FM, a Riemann invariant at the boundary can be imposed by means of prescribing the reference *boundary water level*, using linear theory. Reflections can hence be reflected for *small* perturbations. In the present test, it is checked whether or not these reflections actually do not occur. In this test case, a two-dimensional case is considered.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.8.1: D-Flow FM can deal with boundary conditions that are prescribed as time-series, as harmonic components and as astronomic components
- ◇ claim 2.3.8.3: D-Flow FM can deal with boundary conditions for derived quantities by means of a Qh-table, a Neumann-type condition for the water level and a Riemann invariant

Approach

A circular domain is developed with a flat bed and a flat initial water level. Along the boundary of the domain (the edge of the circle), a Riemann boundary condition is imposed with a reference water level associated with the flat initial water level field. Perturbations are induced by lifting the initial water level in the center of the channel. This thus constructed initial field induces a flow field that fundamentally differs from its one-dimensional equivalent: the water level at the center (where the perturbation is the largest) is subceeded in due course, whereas this is not the case in the one-dimensional case.

Model description

For this test, two domains are developed: a short domain and an elongated variant of it. The short domain has a radius of 90 km, whereas the elongated domain has a radius of 120 km. This grid is covered by triangular cells.

The bed is flat at a level equal to 0 m (with respect to the reference level). The initial water level is 10 m (with respect to the reference level) everywhere. The perturbation is given a Gaussian shape, to prevent sharp moving fronts. Hence, the initial water level reads:

$$h(x, t = 0) = \exp\left(-\frac{x^2 + y^2}{c^2}\right) + h_0$$

with $h_0 = 10$ m (w.r.t. reference) and $c = 20$ km. Along the domain, 9 observation points are placed to monitor the development of the water level in time. The initial water level and the observation points are shown in Figure 11.15.

Processes associated with diffusion and bed friction are turned off. The simulated time is 36000 seconds (10 hours). The wave propagation speed $\sqrt{gh}$ equals about 10 m/s. Travelling from the center of the domain to the edge of the domain takes about 90000 m$/10$ m/s $= 9000$ s for the wave in the short/small domain, whereas 120000 m$/10$ m/s $= 12000$ s for the large/elongated domain. Notice that the perturbation yields 10% of the water depth.

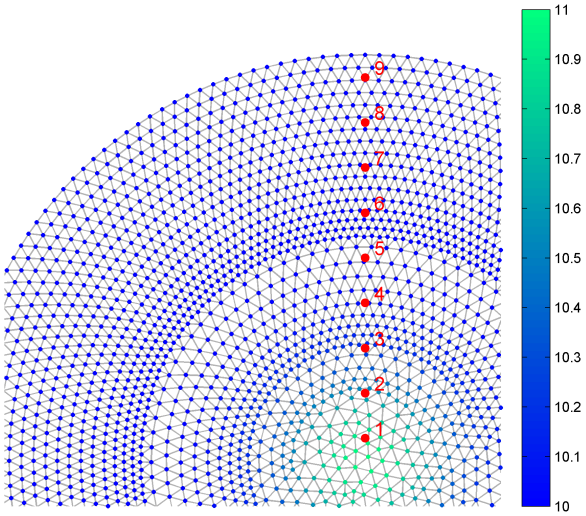


Figure 11.15: *Initial water level in the domain in meters respect to the reference level. The observation point locations are given in red. The grid is circular and consists of triangular cells. Only a part of the grid is shown. The radius of the visualized grid is 90 km.*

Results

If station number 1 is considered (5 km out of the center), the presence of the reflected wave can easily be approximated. For the small domain, the peak occurs at approximately 18000 s after start (equal to twice 9000 s needed to travel forth and back); for the large domain, the peak occurs at 24000 s after start (equal to twice 12000 s).

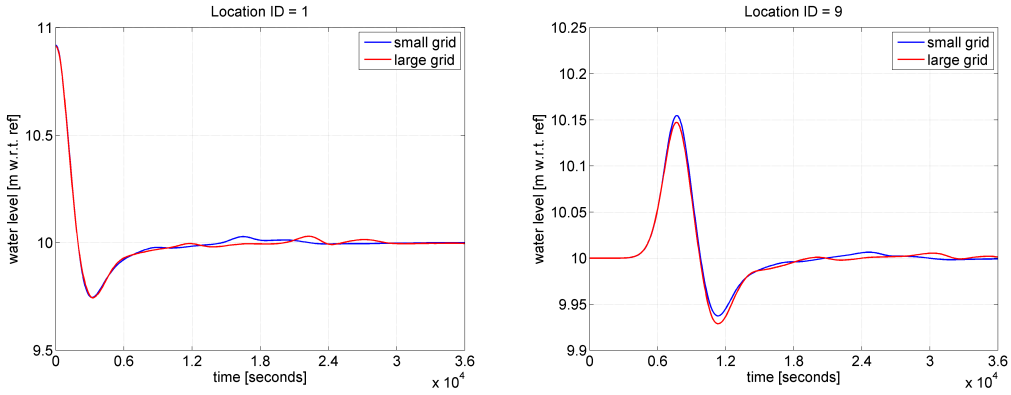


Figure 11.16: *Time series of the computed water level at the observation point locations. The results for the small/short domain are given in blue; the results for the large/elongated domain are given in red.*

For the other stations, a similar behavior is observed. In these cases, small amplitude waves are represented in the time series signal. It is left to the reader to recompute the occurrence of the peak times by hand.

Conclusion

Riemann-type boundary conditions are successfully facilitated by D-Flow FM. The Riemann-type boundary condition is weakly reflective; hence, some reflection is present if the perturbation is relatively large.

Version

This test has been carried out with version dflow-fm-x64-1.1.191.47129.

12 Input and output functionality

12.1 Output test for a realistic model run

Purpose

The `[output]` field of the `[mdu]`-file facilitates the export of flow data to the output directory at specified times. This test investigates the result of specifying non-zero-value intervals at all entries. The `MapOutputTimeVector`-entry is kept out of consideration.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.1.3: D-Flow FM is able to produce his-files, map-files and restart files with output timeseries and field output data

It is claimed that the following output is created:

- ◇ one `_his.nc`-file containing multiple flow variables at the times prescribed by the time interval, including the start time and end time,
- ◇ one `_map.nc`-file containing multiple flow variables at the times prescribed by the time interval and the input file, including the start time and end time,
- ◇ several `_rst.nc`-files at the times prescribed by the time interval, including the start time and end time.

Approach

In the `[output]` field in the associated `[mdu]`-file, zero values are inserted for the time intervals. Only for the `HisInterval`-entry, the `MapInterval`-entry and the `RstInterval`-entry, input is provided.

Model description

A Westerscheldt model is used to generate output data. Two boundaries are present: at the one boundary, a periodic water level signal is imposed, whereas at the other boundary a discharge is prescribed. The physical parameters as such are not of particular interest for this test case. Start time is 10 minutes and stop time is 120 minutes w.r.t. the reference time. The `_his.nc`-file, the times `_map.nc`-file times and the `_rst.nc`-file times are:

- ◇ input as the 'interval', the 'start period' and 'end period' (s): `1200.0`
- ◇ input as the 'interval', the 'start period' and 'end period' (s): `960.0 3600.0`
- ◇ input as the 'interval', the 'start period' and 'end period' (s): `480.0 6000.0 6800.0`

as claimed result in seconds after the reference date. The user time step `DtUser` is 20 seconds and the maximum time step `DtMax` are set to 60.0 seconds. The initial time step is set to 1.0 s. The time step is CFL limited.

Results

In the directory `dflowfmoutput`, a `simplebox_his.nc` file and a `simplebox_map.nc` file, respectively, have appeared containing flow data at the times:

- ◇ 600.00 1800.00 3000.00 4200.00 5400.00 6600.00 7200.00 in seconds after the reference date for the `his`-file,
- ◇ 600.00 3600.00 4560.00 5520.00 6480.00 7200.00 in seconds after the reference date for the `map`-file.

The following `_rst.nc`-files are produced:

```
westerscheldt_19920831_000000_rst.nc
westerscheldt_19920831_001000_rst.nc
westerscheldt_19920831_014000_rst.nc
westerscheldt_19920831_014800_rst.nc
westerscheldt_19920831_020000_rst.nc
```

Conclusion

The obtained result comprises the result that is claimed to be obtained.

Version

This test has been carried out with version dflow-fm-x64-1.1.116.35661.

12.2 Restart from a map-file, with stationary boundary conditions (2D)

Purpose

D-Flow FM facilitates the restart of a computation from flow data resulting from a previously performed computation. For this purpose a _map.nc-file and an _rst.nc-file can be used. Both types can be specified in the .mdu-file.

Linked claims

Claims that are related to the current test case are:

◇ claim 2.3.1.3: D-Flow FM is able to produce his-files, map-files and restart files with output timeseries and field output data

It is claimed that:

◇ a computation restarted from a _map.nc-file yields *exactly* the same output data as the previously performed computation from which the original _map.nc-file was a result.

Approach

A computation is run for 240 seconds. Along the time path, several _map.nc-files and _rst.nc-files are written. From the output _map.nc file that has been generated, the flow field at 60 seconds, after the original start, is used as input for the computation subjected to the test. The results after 240 seconds, both from the cold start and from the restart file, should *exactly* be the same. The boundary conditions specified are constant in time, in this case.

Model description

A simple 10 $\times$ 4 grid is generated with one open boundary. At this boundary, a constant water level signal and salinity signal are imposed. The physical parameters as such are not of particular interest for this test case. The start time and stop time of the original computation are 0 seconds and 240 seconds w.r.t. the reference time, respectively. Output, either as a _map.nc-file and as an _rst.nc-file, is generated at arbitrary times. The restart information is specified in the mdu-file as follows:

◇
```
[restart]
RestartFile = original/simplebox_map.nc
RestartDateTime = 20010101000100
```

Results

The difference between the results from the cold start and from the restart at $t = 240$ seconds, is examined for the water levels, the face-normal velocities and the salinity by means of the root-mean-square difference measure:

◇ water levels h, measured difference $\Delta h_{rms} =$ 0.0000671 m (w.r.t. reference),
◇ velocities u, measured difference $\Delta u_{rms} =$ 0.0000114 m/s,
◇ salinity s, measured difference $\Delta s_{rms} =$ 0.0001740 ppt.

Since all the three values are non-zero, it can be concluded that the restart is not fully *exact*. The restart functionality itself appears to be properly.

Conclusion

The functional working of the possibility to use _map.nc-files for restart purpose does operate properly. However, the restart is not fully exact.

Version

This test has been carried out with version dflow-fm-x64-1.1.116.35661.

12.3 Restart from a map-file, with instationary boundary conditions (2D)

Purpose

D-Flow FM facilitates the restart of a computation from flow data resulting from a previously performed computation. For this purpose a _map.nc-file and an _rst.nc-file can be used. Both types can be specified in the .mdu-file.

Linked claims

Claims that are related to the current test case are:

◇ claim 2.3.1.3: D-Flow FM is able to produce his-files, map-files and restart files with output timeseries and field output data

It is claimed that:

◇ a computation restarted from a _map.nc-file yields *exactly* the same output data as the previously performed computation from which the original _map.nc-file was a result.

Approach

A computation is run for 240 seconds. Along the time path, several _map.nc-files and _rst.nc-files are written. From the output _map.nc file that has been generated, the flow field at 60 seconds, after the original start, is used as input for the computation subjected to the test. The results after 240 seconds, both from the cold start and from the restart file, should *exactly* be the same. The boundary conditions specified are varying in time, in this case.

Model description

A simple 10 $\times$ 4 grid is generated with one open boundary. At this boundary, a time-varying water level signal and salinity signal are imposed. The physical parameters as such are not of particular interest for this test case. The start time and stop time of the original computation are 0 seconds and 240 seconds w.r.t. the reference time, respectively. Output, either as a _map.nc-file and as an _rst.nc-file, is generated at arbitrary times. The restart information is specified in the mdu-file as follows:

◇
```
[restart]
RestartFile = original/simplebox_map.nc
RestartDateTime = 20010101000100
```

Results

The difference between the results from the cold start and from the restart at $t = 240$ seconds, is examined for the water levels, the face-normal velocities and the salinity by means of the root-mean-square difference measure:

◇ water levels h, measured difference $\Delta h_{rms} =$ 0.0000976 m (w.r.t. reference),
◇ velocities u, measured difference $\Delta u_{rms} =$ 0.0000303 m/s,
◇ salinity s, measured difference $\Delta s_{rms} =$ 0.0002557 ppt.

Since all the three values are non-zero, it can be concluded that the restart is not fully *exact*. The restart functionality itself appears to be properly.

Conclusion

The functional working of the possibility to use _map.nc-files for restart purpose does operate properly. However, the restart is not fully exact.

Version

This test has been carried out with version dflow-fm-x64-1.1.116.35661.

12.4 Restart from an rst-file, with stationary boundary conditions (2D)

Purpose

D-Flow FM facilitates the restart of a computation from flow data resulting from a previously performed computation. For this purpose a _map.nc-file and an _rst.nc-file can be used. Both types can be specified in the .mdu-file.

Linked claims

Claims that are related to the current test case are:

◇ claim 2.3.1.3: D-Flow FM is able to produce his-files, map-files and restart files with output timeseries and field output data

It is claimed that:

◇ a computation restarted from a _rst.nc-file yields *exactly* the same output data as the previously performed computation from which the original _rst.nc-file was a result.

Approach

A computation is run for 240 seconds. Along the time path, several _map.nc-files and _rst.nc-files are written. From the output _rst.nc files that have been generated, the file containing the flow field after 60 seconds (w.r.t. the original start), is used as input for the computation subjected to the test. The results after 240 seconds, both from the cold start and from the restart file, should *exactly* be the same. The boundary conditions specified are constant in time, in this case.

Model description

A simple 10 $\times$ 4 grid is generated with one open boundary. At this boundary, a constant water level signal and salinity signal are imposed. The physical parameters as such are not of particular interest for this test case. The start time and stop time of the original computation are 0 seconds and 240 seconds w.r.t. the reference time, respectively. Output, either as a _map.nc-file and as an _rst.nc-file, is generated at arbitrary times. The restart information is specified in the mdu-file as follows:

◇
```
[restart]
RestartFile = original/simplebox_20010101_000100_rst.nc
RestartDateTime =
```

Results

The difference between the results from the cold start and from the restart at $t = 240$ seconds, is examined for the water levels, the face-normal velocities and the salinity by means of the root-mean-square difference measure:

◇ water levels h, measured difference $\Delta h_{rms} =$ 0.0000671 m (w.r.t. reference),
◇ velocities u, measured difference $\Delta u_{rms} =$ 0.0000114 m/s,
◇ salinity s, measured difference $\Delta s_{rms} =$ 0.0001740 ppt.

Since all the three values are non-zero, it can be concluded that the restart is not fully *exact*. The restart functionality itself appears to be properly.

Conclusion

The functional working of the possibility to use `_rst.nc`-files for restart purpose does operate properly. However, the restart is not fully exact.

Version

This test has been carried out with version dflow-fm-x64-1.1.116.35661.

12.5 Restart from an rst-file, with instationary boundary conditions (2D)

Purpose

D-Flow FM facilitates the restart of a computation from flow data resulting from a previously performed computation. For this purpose a `_map.nc`-file and an `_rst.nc`-file can be used. Both types can be specified in the `.mdu`-file.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.1.3: D-Flow FM is able to produce his-files, map-files and restart files with output timeseries and field output data

It is claimed that:

- ◇ a computation restarted from a `_rst.nc`-file yields *exactly* the same output data as the previously performed computation from which the original `_rst.nc`-file was a result.

Approach

A computation is run for 240 seconds. Along the time path, several `_map.nc`-files and `_rst.nc`-files are written. From the output `_rst.nc` files that have been generated, the file containing the flow field after 60 seconds (w.r.t. the original start), is used as input for the computation subjected to the test. The results after 240 seconds, both from the cold start and from the restart file, should *exactly* be the same. The boundary conditions specified are varying in time, in this case.

Model description

A simple 10 $\times$ 4 grid is generated with one open boundary. At this boundary, a time-varying water level signal and salinity signal are imposed. The physical parameters as such are not of particular interest for this test case. The start time and stop time of the original computation are 0 seconds and 240 seconds w.r.t. the reference time, respectively. Output, either as a `_map.nc`-file and as an `_rst.nc`-file, is generated at arbitrary times. The restart information is specified in the `mdu`-file as follows:

- ◇
```
[restart]
RestartFile = original/simplebox_20010101_000100_rst.nc
RestartDateTime =
```

Results

The difference between the results from the cold start and from the restart at $t = 240$ seconds, is examined for the water levels, the face-normal velocities and the salinity by means of the root-mean-square difference measure:

- ◇ water levels h, measured difference $\Delta h_{rms} =$ 0.0000976 m (w.r.t. reference),
- ◇ velocities u, measured difference $\Delta u_{rms} =$ 0.0000303 m/s,
- ◇ salinity s, measured difference $\Delta s_{rms} =$ 0.0002557 ppt.

Since all the three values are non-zero, it can be concluded that the restart is not fully *exact*. The restart functionality itself appears to be properly.

Conclusion

The functional working of the possibility to use `_rst.nc`-files for restart purpose does operate properly. However, the restart is not fully exact.

Version

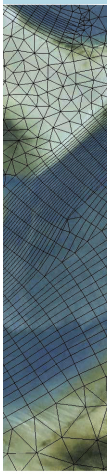
This test has been carried out with version dflow-fm-x64-1.1.116.35661.

13 Miscellaneous

13.1 Dry points through a sample set

Purpose

Under certain circumstances, it may become necessary to exclude user-specified grid cells from the flow computations. These cells are marked by 'dry points' that are supplied by the user by means of samples or polygons. This test-case examines whether D-Flow FM can correctly detect and disable the dry cells, based on a sample set with dry points.



Linked claims

Claims that are related to the current test case are:

◇ claim 2.3.8.5: D-Flow FM can take into account the explicit specification of dry points

Approach

A Cartesian mesh is deployed in a rectangular domain. Appropriate boundary conditions are applied to create a stationary flow field. Dry points are specified by a sample set and effectively divide the mesh into three parts: one part contains both the inflow and outflow boundaries and is separated from the other two parts by means of the dry points. The two other parts are divided such that on of them contains only the inflow boundary, while the other only contains the outflow boundary.

The three parts are connected by 1D flowlinks to enable a stationary flow. The flow-field through the one part that contains both boundaries should be identical to the flow-field through the two other parts that connected by the 1D flowlinks.

Model description

The computational domain is 50×40 m^2. A velocity boundary condition of 1 m/s is supplied at the left-hand side and a uniform waterlevel of 0 m (w.r.t. reference) is specified at the right-hand side of the domain.

The computational grid contains cells of 10×10 m^2 and is depicted in Figure 13.3. The upper part of the middle column of cells, and a complete row of cells are disable by the samples as shown in the figure. The upper-left and upper-right part of the grid are connected through the two 1D-flowlinks shown in pink. The width and bedlevel of the 1D flowlinks equals the width and bedlevel of the 2D gridcells.

The bottom part of the domain contains (a part of) both the inflow and outflow boundaries and is completely separated from the top parts by the dry points. The top-left part contains only (a part of) the inflow-boundary, while the top-right part contains only (a part of) the outflow boundary. The top parts are separated by dry points, but connected trough the 1D flowlinks.

The simulation period is set to 150 sec. and the time-step is 1 sec. Since horizontal momentum diffusion and higher-order advection are discretized differently at the 1D flowlinks, we have set the horizontal viscosity coefficient to zero and employed a first-order advection scheme, respectively.

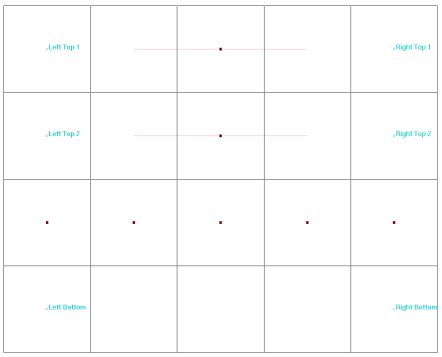


Figure 13.1: *Computational mesh; dry points are specified by the samples as shown in red; 1D links are shown in pink; observation stations are shown as blue crosses.*

Results

The water levels at the observations points are presented in Figure 13.4. The water levels in the top part of the domain are identical to the water levels in the bottom part.

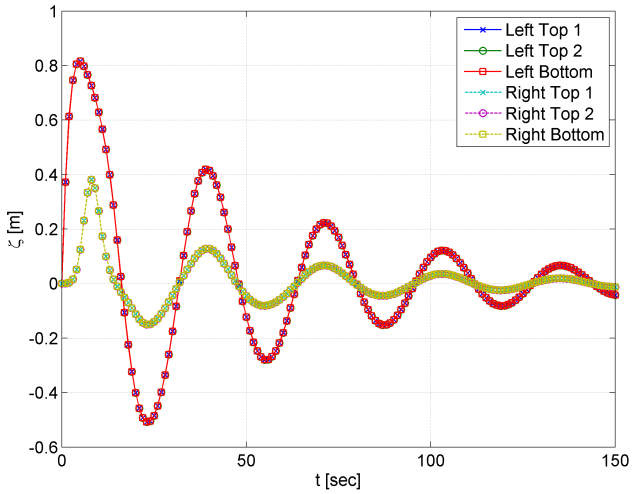


Figure 13.2: *Water levels at the observation points*

Conclusion

Dry points can be specified by means of a sample set and used to exclude computational cells from the computations.

Version

This test has been carried out with version dflow-fm-x64-1.1.116.36629.

13.2 Dry points through a polygon

Purpose

Under certain circumstances, it may become necessary to exclude user-specified grid cells from the flow computations. These cells are marked by 'dry points' that are supplied by the user by means of samples or polygons. This test-case examines whether D-Flow FM can correctly detect and disable the dry cells defined by a polygon.

Linked claims

Claims that are related to the current test case are:

◇ claim 2.3.8.5: D-Flow FM can take into account the explicit specification of dry points

Approach

A Cartesian mesh is deployed in a rectangular domain. Appropriate boundary conditions are applied to create a stationary flow field. Dry points are defined by a polygon and effectively divide the mesh into three parts: one part contains both the inflow and outflow boundaries and is separated from the other two parts by means of the dry points. The two other parts are divided such that on of them contains only the inflow boundary, while the other only contains the outflow boundary.

The three parts are connected by 1D flowlinks to enable a stationary flow. The flow-field through the one part that contains both boundaries should be identical to the flow-field through the two other parts that connected by the 1D flowlinks.

Model description

The computational domain is 50×40 m^2. A velocity boundary condition of 1 m/s is supplied at the left-hand side and a uniform waterlevel of 0 m (w.r.t. reference) is specified at the right-hand side of the domain.

The computational grid contains cells of 10×10 m^2 and is depicted in Figure 13.3. The upper part of the middle column of cells, and a complete row of cells are disable by the polygon as shown in the figure. The upper-left and upper-right part of the grid are connected through the two 1D-flowlinks shown in pink. The width and bedlevel of the 1D flowlinks equals the width and bedlevel of the 2D gridcells.

The bottom part of the domain contains (a part of) both the inflow and outflow boundaries and is completely separated from the top parts by the dry points. The top-left part contains only (a part of) the inflow-boundary, while the top-right part contains only (a part of) the outflow boundary. The top parts are separated by dry points, but connected trough the 1D flowlinks.

The simulation period is set to 150 sec. and the time-step is 1 sec. Since horizontal momentum diffusion and higher-order advection are discretized differently at the 1D flowlinks, we have set the horizontal viscosity coefficient to zero and employed a first-order advection scheme, respectively.

Results

The water levels at the observations points are presented in Figure 13.4. The water levels in the top part of the domain are identical to the water levels in the bottom part.

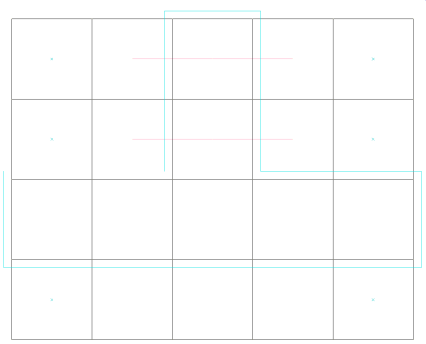


Figure 13.3: *Computational mesh; dry points are specified by the polygon as shown in blue; 1D links are shown in pink; observation stations are shown as blue crosses.*

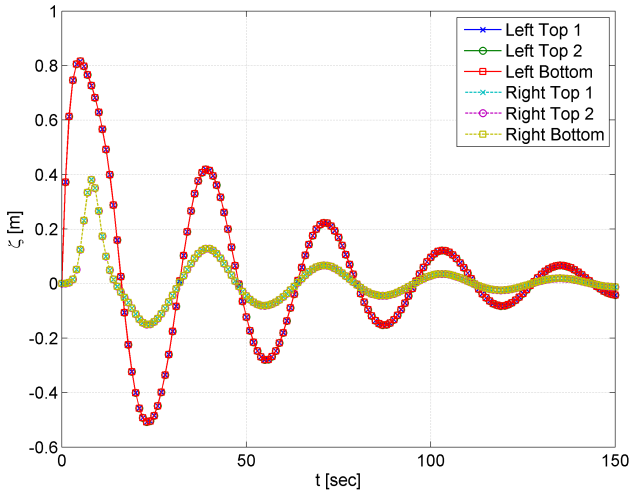


Figure 13.4: *Water levels at the observation points*

Conclusion

Dry points can be specified by means of a polygon and used to exclude computational cells from the computations.

Version

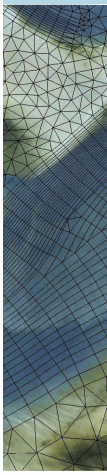
This test has been carried out with version dflow-fm-x64-1.1.116.36629.

14 Practical cases

14.1 Schematized Frisian inlet

Purpose

For D-Flow FM, a coupling with an external wave module has been established. The quality of the joint functionality is assessed in a distinct test engine (engine `e26` for D-Flow FM, also see engine `e23` for the coupling with Delft3D) by means of a set of five separate test cases. For all these five test cases, insight in the flow behavior itself, i.e. *without* waves, is required in particular to see if the fundament of the wave test case is sound. For this purpose, the results of these five test cases (without waves) are analyzed with the associated Delft3D results as a backdrop. This particular test case examines one of these five test cases. In this test case, a strongly schematized version of the Frisian Inlet is considered.



Linked claims

Claims that are related to the current test case are:

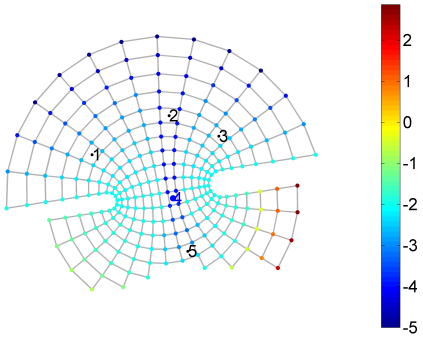
- claim 2.3.1.2: D-Flow FM is suitable for the prediction of the tidal dynamics in estuaries or coastal seas

Approach

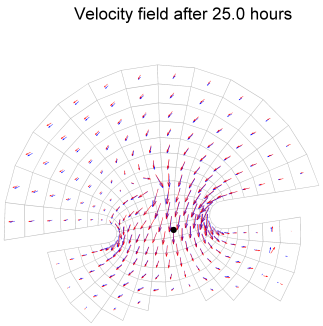
The following actions have been deployed in order to come to a sound test case validation:

1. Working copies of the original Delft3D models have been created for the five test cases.
2. The Delft3D models are modified in order to:
   - switch off the wave functionality,
   - switch off any reference to sed/mor-type of functionality,
   - switch off any reference to source/sink-functionality,
   - switch off any reference to 3D-functionality (scope is restricted to 2D).
3. The five Delft3D test models are converted to D-Flow FM equivalents by means of the Matlab tool `dflowfmConverter`, available within OpenEarth.
4. The way of dealing with the bathymetry is different in D-Flow FM compared to Delft3D. In order to come closest to a methodology, the settings for the bathymetry are set to:
   - Delft3D: `Dryflp=NO`, `Dpsopt=MAX` and `Dpuopt=MEAN`,
   - D-Flow FM: `BedLevType = 3` and `Conveyance2D = -1`.
5. In case astronomic or harmonic boundary conditions are imposed, one additional modification is applied regarding the number of support points. Consider a boundary condition prescribed through 2 support points, with different amplitude and phase for each support point, then the approaches are as following:
   - Delft3D: linearly interpolate the amplitudes and phases between the 2 support points, and then construct the timeseries for the quantity under consideration,
   - D-Flow FM: construct the timeseries at the 2 support points, and then linearly interpolate between the support points.

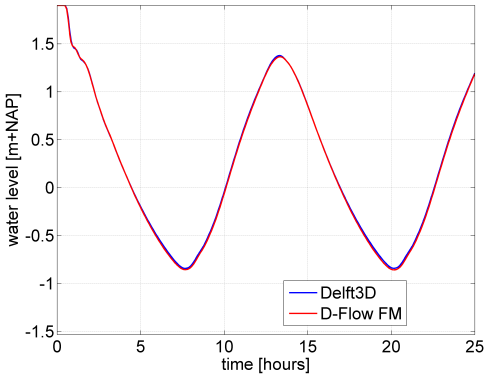
   These two approaches can lead to differences if the distance between 2 support points is relatively large. In order to circumvent differences due to this cause, additional support points are prescribed in the Delft3D model, yielding additional support points in the converted D-Flow FM equivalents.
6. A dedicated Matlab script is written to visualize the entire flow field (velocity vectors) as well as the water level and velocity magnitude at an arbitrary observation point of interest.



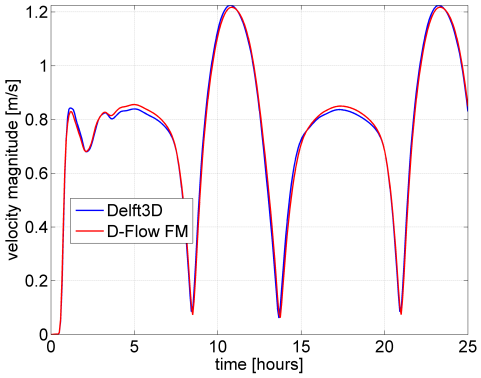
(a) *The grid, the bathymetry and the locations of the observation points.*



(b) *Velocity vectors after 25.0 hours from the start of the simulation (Delft3D in blue, D-Flow FM in red).*



(c) *Timeseries of the water level at observation location 4 (see Figure 14.1a).*



(d) *Timeseries of the velocity magnitude at observation location 4 (see Figure 14.1a).*

Figure 14.1: *Analysis of the schematized Frisian Inlet case.*

Model description

The model consists of a curvilinear grid. This grid as well as the imposed bathymetry are visualized Figure 14.1a. The model has directly been derived from the original Delft3D test case found in this repository:

```
https://repos.deltares.nl/repos/DSCTestbench/cases/trunk/e23_d3dflow-wave/
```

as test case:

```
e23_f01_c08-f34
```

In addition to the original case, an extra boundary support point has been added at the north side.

Results

Some selected results are highlighted in Figure 14.1. The flow fields from Delft3D and D-Flow FM after 25 hours of simulated time have appeared to be very close to each other. To keep the analysis brief and concise, only the water level and velocity magnitude are plotted for only one location: Figure 14.1 shows that the water levels and the velocity magnitudes fairly coincide at the observation point in the center of the domain, which is a representative result for the other observation locations as well.

Conclusion

The test case for the schematized Frisian Inlet is suitable to serve as a test case for the joint functionality of D-Flow FM with the external wave module.

Version

This test has been carried out with version dflow-fm-x64-1.1.116.36086.

14.2 Realistic Frisian inlet

Purpose

For D-Flow FM, a coupling with an external wave module has been established. The quality of the joint functionality is assessed in a distinct test engine (engine `e26` for D-Flow FM, also see engine `e23` for the coupling with Delft3D) by means of a set of five separate test cases. For all these five test cases, insight in the flow behavior itself, i.e. *without* waves, is required in particular to see if the fundament of the wave test case is sound. For this purpose, the results of these five test cases (without waves) are analyzed with the associated Delft3D results as a backdrop. This particular test case examines one of these five test cases. In this test case, a schematized though realistic version of the Frisian Inlet is considered.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.1.2: D-Flow FM is suitable for the prediction of the tidal dynamics in estuaries or coastal seas

Approach

The following actions have been deployed in order to come to a sound test case validation:

1. Working copies of the original Delft3D models have been created for the five test cases.
2. The Delft3D models are modified in order to:
   - ◇ switch off the wave functionality,
   - ◇ switch off any reference to sed/mor-type of functionality,
   - ◇ switch off any reference to source/sink-functionality,
   - ◇ switch off any reference to 3D-functionality (scope is restricted to 2D).
3. The five Delft3D test models are converted to D-Flow FM equivalents by means of the Matlab tool `dflowfmConverter`, available within OpenEarth.
4. The way of dealing with the bathymetry is different in D-Flow FM compared to Delft3D. In order to come closest to a methodology, the settings for the bathymetry are set to:
   - ◇ Delft3D: `Dryflp=NO`, `Dpsopt=MAX` and `Dpuopt=MEAN`,
   - ◇ D-Flow FM: `BedLevType = 3` and `Conveyance2D = -1`.
5. In case astronomic or harmonic boundary conditions are imposed, one additional modification is applied regarding the number of support points. Consider a boundary condition prescribed through 2 support points, with different amplitude and phase for each support point, then the approaches are as following:
   - ◇ Delft3D: linearly interpolate the amplitudes and phases between the 2 support points, and then construct the timeseries for the quantity under consideration,
   - ◇ D-Flow FM: construct the timeseries at the 2 support points, and then linearly interpolate between the support points.

   These two approaches can lead to differences if the distance between 2 support points is relatively large. In order to circumvent differences due to this cause, additional support points are prescribed in the Delft3D model, yielding additional support points in the converted D-Flow FM equivalents.
6. A dedicated Matlab script is written to visualize the entire flow field (velocity vectors) as well as the water level and velocity magnitude at an arbitrary observation point of interest.

Model description

The model consists of a curvilinear grid. This grid as well as the imposed bathymetry are visualized Figure 14.2a. The model has directly been derived from the original Delft3D test case found in this repository:

```
https://repos.deltares.nl/repos/DSCTestbench/cases/trunk/e23_d3dflow-wave,
```

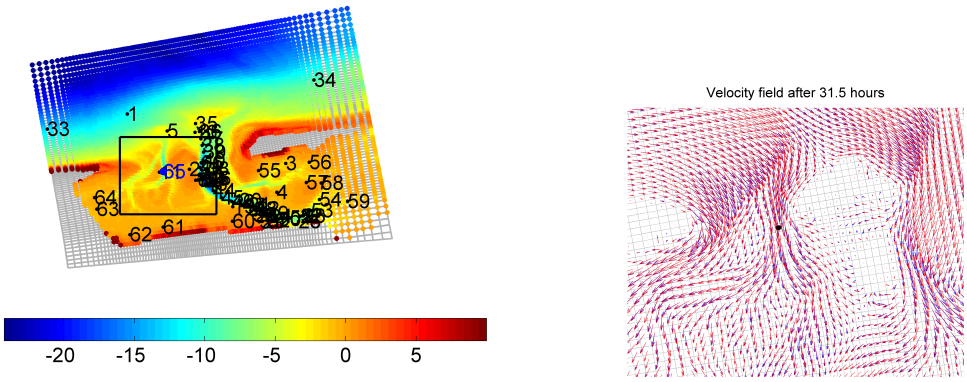
as test case:

```
e23_f01_c04-botnow_curnow
```

which comprises a realistic Frisian Inlet model with tidal flats.

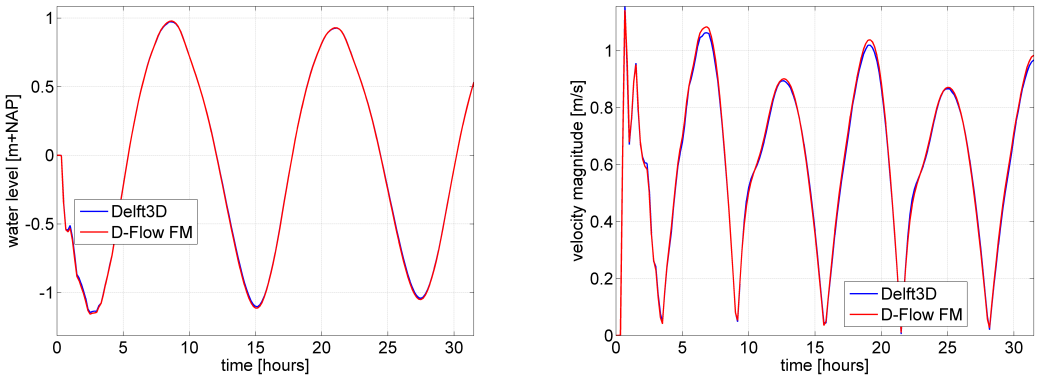
Results

Some selected results are highlighted in Figure 14.2. The flow fields from Delft3D and D-Flow



(a) *The grid, the bathymetry and the locations of the observation points (box highlighted in Figure 14.2b).*

(b) *Velocity vectors after 31.5 hours from the start of the simulation (Delft3D in blue, D-Flow FM in red).*



(c) *Timeseries of the water level at observation location 65 (see Figure 14.2a).*

(d) *Timeseries of the velocity magnitude at observation location 65 (see Figure 14.2a).*

Figure 14.2: *Analysis of the realistic Frisian Inlet case.*

FM after 31.5 hours of simulated time have appeared to be very close to each other. To keep the analysis brief and concise, only the water level and velocity magnitude are plotted for

only one location: Figure 14.2 shows that the water levels and the velocity magnitudes fairly coincide at the observation point in the center of the domain, which is a representative result for the other observation locations as well.

Conclusion

The test case for the realistic Frisian Inlet is suitable to serve as a test case for the joint functionality of D-Flow FM with the external wave module.

Version

This test has been carried out with version dflow-fm-x64-1.1.116.36086.

14.3 Californian coast

Purpose

For D-Flow FM, a coupling with an external wave module has been established. The quality of the joint functionality is assessed in a distinct test engine (engine `e26` for D-Flow FM, also see engine `e23` for the coupling with Delft3D) by means of a set of five separate test cases. For all these five test cases, insight in the flow behavior itself, i.e. *without* waves, is required in particular to see if the fundament of the wave test case is sound. For this purpose, the results of these five test cases (without waves) are analyzed with the associated Delft3D results as a backdrop. This particular test case examines one of these five test cases. In this test case, a part of a coastal area along California is considered.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.1.2: D-Flow FM is suitable for the prediction of the tidal dynamics in estuaries or coastal seas

Approach

The following actions have been deployed in order to come to a sound test case validation:

1. Working copies of the original Delft3D models have been created for the five test cases.
2. The Delft3D models are modified in order to:
   - ◇ switch off the wave functionality,
   - ◇ switch off any reference to sed/mor-type of functionality,
   - ◇ switch off any reference to source/sink-functionality,
   - ◇ switch off any reference to 3D-functionality (scope is restricted to 2D).
3. The five Delft3D test models are converted to D-Flow FM equivalents by means of the Matlab tool `dflowfmConverter`, available within OpenEarth.
4. The way of dealing with the bathymetry is different in D-Flow FM compared to Delft3D. In order to come closest to a methodology, the settings for the bathymetry are set to:
   - ◇ Delft3D: `Dryflp=NO`, `Dpsopt=MAX` and `Dpuopt=MEAN`,
   - ◇ D-Flow FM: `BedLevType = 3` and `Conveyance2D = -1`.
5. In case astronomic or harmonic boundary conditions are imposed, one additional modification is applied regarding the number of support points. Consider a boundary condition prescribed through 2 support points, with different amplitude and phase for each support point, then the approaches are as following:
   - ◇ Delft3D: linearly interpolate the amplitudes and phases between the 2 support points, and then construct the timeseries for the quantity under consideration,
   - ◇ D-Flow FM: construct the timeseries at the 2 support points, and then linearly interpolate between the support points.

   These two approaches can lead to differences if the distance between 2 support points is relatively large. In order to circumvent differences due to this cause, additional support points are prescribed in the Delft3D model, yielding additional support points in the converted D-Flow FM equivalents.
6. A dedicated Matlab script is written to visualize the entire flow field (velocity vectors) as well as the water level and velocity magnitude at an arbitrary observation point of interest.

Model description

The model consists of a curvilinear grid. This grid as well as the imposed bathymetry are visualized Figure 14.3a. The model has directly been derived from the original Delft3D test case found in this repository:

```
https://repos.deltares.nl/repos/DSCTestbench/cases/trunk/e23_d3dflow-wave/
```

as test case:

```
e23_f01_c07-wavewatch_file
```

For the central California test case, some additional modifications are applied to the original Delft3D-model, namely:

- ◇ The `.bnd`-file contains reflection coefficients: initially, these values have been set 10^4. However, D-Flow FM does not support reflection coefficients. Hence, these coefficients are set to zero.
- ◇ The `.bnd`-file contains a piecewise setup of the boundary conditions locations; initially, at the connection of the various pieces, the same boundary conditions values are specified for the two adjacent cells, spoiling the strict linear interpolation of the boundary conditions. In the original data outcomes, spurious flow patterns were observed at the connections of the several pieces spanning the boundary conditions. As a remedy, the two adjacent points, connecting two pieces of the boundary locations specification have been merged into one cell.
- ◇ The option `tlfsmo` (Fourier smoothing) is set to zero, for the implementation of this smoothing option is different in D-Flow FM compared to Delft3D. In Delft3D, the smoothing operation is only applied to the equilibrium position of the imposed signal, whereas D-Flow FM applies the smoothing to the entire signal: to either the equilibrium position as well to the periodic components.

Results

Some selected results are highlighted in Figure 14.3. A remarkable observation from the right panel of Figure 14.3 is that oscillations occur in the output signal of Delft3D. This kind of oscillations is observed at every observation point. Some investigations have, however, shown that these oscillations can circumvented through returning to the original setting: either through setting the reflection coefficient to a high value, or through prescribing the original boundary locations specifications.

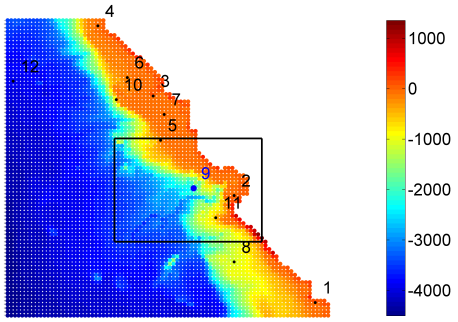
However, the first option spoils the agreement of both data (Delft3D results do not show oscillations anymore, by the output signals are flawed); the second option introduces spurious flow patterns (it is remarkable though that the piecewise prescription makes the oscillations disappear).

Conclusion

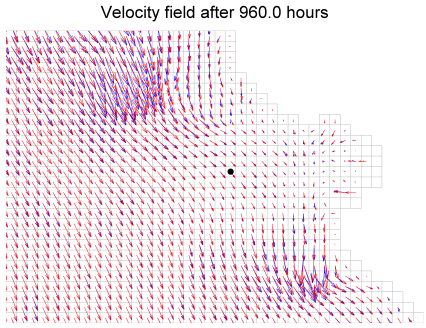
The test case for the California coast case is suitable to serve as a test case for the joint functionality of D-Flow FM with the external wave module.

Version

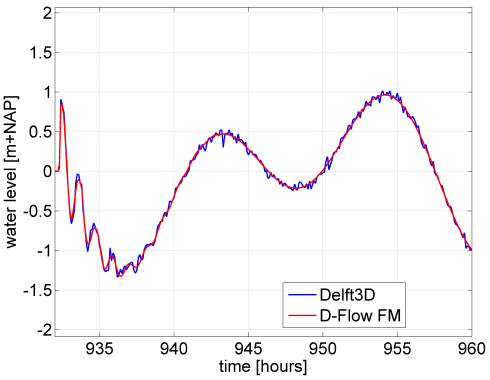
This test has been carried out with version dflow-fm-x64-1.1.116.36086.



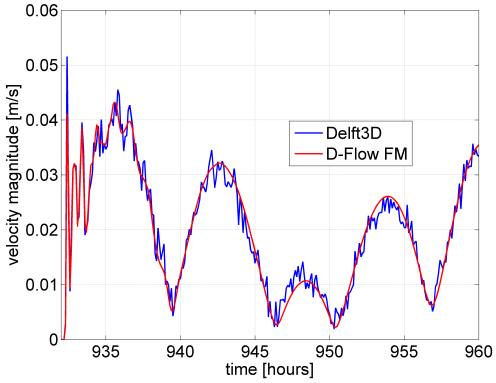
(a) *The grid, the bathymetry and the locations of the observation points (box highlighted in Figure 14.3b).*



(b) *Velocity vectors after 960 hours from the start of the simulation (Delft3D in blue, D-Flow FM in red).*



(c) *Timeseries of the water level at observation location 9 (see Figure 14.3a).*



(d) *Timeseries of the velocity magnitude at observation location 9 (see Figure 14.3a).*

Figure 14.3: *Analysis of the schematized Californian coast case.*

14.4 Wind over Lake Loosdrecht

Purpose

For D-Flow FM, a coupling with an external wave module has been established. The quality of the joint functionality is assessed in a distinct test engine (engine `e26` for D-Flow FM, also see engine `e23` for the coupling with Delft3D) by means of a set of five separate test cases. For all these five test cases, insight in the flow behavior itself, i.e. *without* waves, is required in particular to see if the fundament of the wave test case is sound. For this purpose, the results of these five test cases (without waves) are analyzed with the associated Delft3D results as a backdrop. This particular test case examines one of these five test cases. In this test case, wind driven flow in Lake Loosdrecht is considered.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.1.2: D-Flow FM is suitable for the prediction of the tidal dynamics in estuaries or coastal seas

Approach

The following actions have been deployed in order to come to a sound test case validation:

1. Working copies of the original Delft3D models have been created for the five test cases.
2. The Delft3D models are modified in order to:
   - ◇ switch off the wave functionality,
   - ◇ switch off any reference to sed/mor-type of functionality,
   - ◇ switch off any reference to source/sink-functionality,
   - ◇ switch off any reference to 3D-functionality (scope is restricted to 2D).
3. The five Delft3D test models are converted to D-Flow FM equivalents by means of the Matlab tool `dflowfmConverter`, available within OpenEarth.
4. The way of dealing with the bathymetry is different in D-Flow FM compared to Delft3D. In order to come closest to a methodology, the settings for the bathymetry are set to:
   - ◇ Delft3D: `Dryflp=NO`, `Dpsopt=MAX` and `Dpuopt=MEAN`,
   - ◇ D-Flow FM: `BedLevType = 3` and `Conveyance2D = -1`.
5. In case astronomic or harmonic boundary conditions are imposed, one additional modification is applied regarding the number of support points. Consider a boundary condition prescribed through 2 support points, with different amplitude and phase for each support point, then the approaches are as following:
   - ◇ Delft3D: linearly interpolate the amplitudes and phases between the 2 support points, and then construct the timeseries for the quantity under consideration,
   - ◇ D-Flow FM: construct the timeseries at the 2 support points, and then linearly interpolate between the support points.

   These two approaches can lead to differences if the distance between 2 support points is relatively large. In order to circumvent differences due to this cause, additional support points are prescribed in the Delft3D model, yielding additional support points in the converted D-Flow FM equivalents.
6. A dedicated Matlab script is written to visualize the entire flow field (velocity vectors) as well as the water level and velocity magnitude at an arbitrary observation point of interest.

Model description

The model consists of a curvilinear grid. This grid as well as the imposed bathymetry are visualized Figure 14.4a. The model has directly been derived from the original Delft3D test case found in this repository:

```
https://repos.deltares.nl/repos/DSCTestbench/cases/trunk/e23_d3dflow-wave/
```

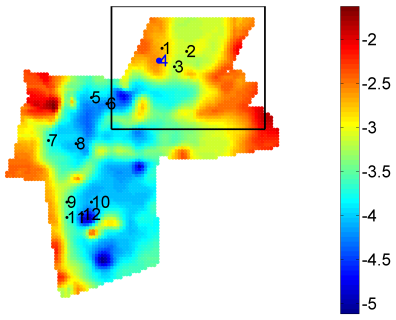
as test case:

```
e23_f99_c99-of_loosdrecht_mean
```

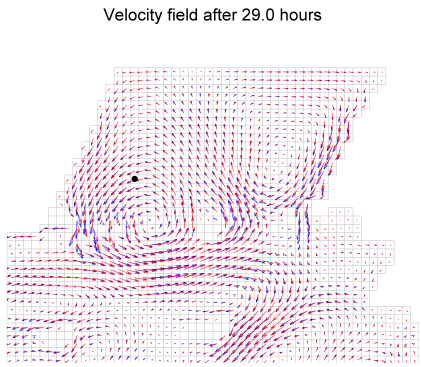
The test case with wind driven flow in Lake Loosdrecht is a test case without specific boundary conditions. Only a time-varying wind field is present, prescibed by a `unimagdir`-type ascii-file. The computational grid is filled by a significant number of thin dams. Thus, eddy formation due to topographical forcing is expected. The surface elevations and velocity magnitudes are expected to be relatively small.

Results

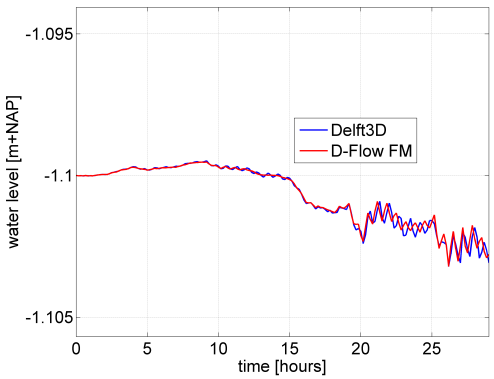
The flow field after the specified simulated time is shown in Figure 14.4 for the northern part of the geometry as well as the velocity magnitude time series at the observation point marked by a black dot. Having considered the surface elevations and velocity magnitudes at several



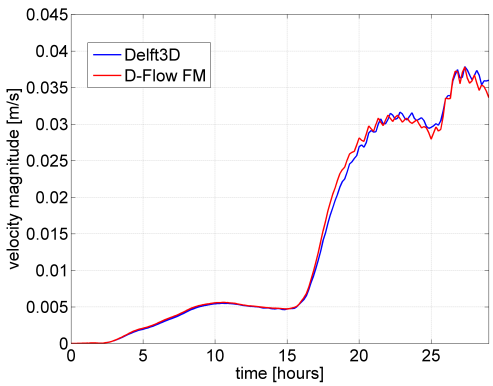
(a) *The grid, the bathymetry and the locations of the observation points (box highlighted in Figure 14.4b).*



(b) *Velocity vectors after 29.0 hours from the start of the simulation (Delft3D in blue, D-Flow FM in red).*



(c) *Timeseries of the water level at observation location 4 (see Figure 14.4a).*



(d) *Timeseries of the velocity magnitude at observation location 4 (see Figure 14.4a).*

Figure 14.4: *Analysis of the Lake Loosdrecht case.*

locations as well as having inspected the eddy pattern at several points in time, the differences between Delft3D and D-Flow FM can be considered small, taking into account the subtlety of the flow situation.

Conclusion

The test case for wind driven flow over Lake Loosdrecht is suitable to serve as a test case for the joint functionality of D-Flow FM with the external wave module.

Version

This test has been carried out with version dflow-fm-x64-1.1.116.36086.

14.5 Flow around an obstacle placed on an inclined bed

Purpose

For D-Flow FM, a coupling with an external wave module has been established. The quality of the joint functionality is assessed in a distinct test engine (engine `e26` for D-Flow FM, also see engine `e23` for the coupling with Delft3D) by means of a set of five separate test cases. For all these five test cases, insight in the flow behavior itself, i.e. *without* waves, is required in particular to see if the fundament of the wave test case is sound. For this purpose, the results of these five test cases (without waves) are analyzed with the associated Delft3D results as a backdrop. This particular test case examines one of these five test cases. In this test case, the flow over an inclined bed and around an obstacle as well is considered.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.1.2: D-Flow FM is suitable for the prediction of the tidal dynamics in estuaries or coastal seas

Approach

The following actions have been deployed in order to come to a sound test case validation:

1. Working copies of the original Delft3D models have been created for the five test cases.
2. The Delft3D models are modified in order to:
   - ◇ switch off the wave functionality,
   - ◇ switch off any reference to sed/mor-type of functionality,
   - ◇ switch off any reference to source/sink-functionality,
   - ◇ switch off any reference to 3D-functionality (scope is restricted to 2D).
3. The five Delft3D test models are converted to D-Flow FM equivalents by means of the Matlab tool `dflowfmConverter`, available within OpenEarth.
4. The way of dealing with the bathymetry is different in D-Flow FM compared to Delft3D. In order to come closest to a methodology, the settings for the bathymetry are set to:
   - ◇ Delft3D: `Dryflp=NO`, `Dpsopt=MAX` and `Dpuopt=MEAN`,
   - ◇ D-Flow FM: `BedLevType = 3` and `Conveyance2D = -1`.
5. In case astronomic or harmonic boundary conditions are imposed, one additional modification is applied regarding the number of support points. Consider a boundary condition prescribed through 2 support points, with different amplitude and phase for each support point, then the approaches are as following:
   - ◇ Delft3D: linearly interpolate the amplitudes and phases between the 2 support points, and then construct the timeseries for the quantity under consideration,
   - ◇ D-Flow FM: construct the timeseries at the 2 support points, and then linearly interpolate between the support points.

   These two approaches can lead to differences if the distance between 2 support points is relatively large. In order to circumvent differences due to this cause, additional support points are prescribed in the Delft3D model, yielding additional support points in the converted D-Flow FM equivalents.
6. A dedicated Matlab script is written to visualize the entire flow field (velocity vectors) as well as the water level and velocity magnitude at an arbitrary observation point of interest.

Model description

The model consists of a curvilinear grid. This grid as well as the imposed bathymetry are visualized Figure 14.5a. The model has directly been derived from the original Delft3D test case found in this repository:

`https://repos.deltares.nl/repos/DSCTestbench/cases/trunk/e23_d3dflow-wave/`

as test case:

`e23_f01_c06-wavecon`

The test case comprises the flow around an obstacle in a schematized coastal zone. The coastal zone is represented as an inclined bed. Originally, the option `tlfsmo` (Fourier smoothing) was switched on. Since the implementation of this Fourier smoothing procedure for the boundary conditions is different in D-Flow FM compared to Delft3D (in Delft3D, the smoothing operation is only applied to the equilibrium position of the imposed signal, whereas D-Flow FM applies the smoothing to the entire signal), the key `tlfsmo` is set to zero.

In order to prevent shockwaves due to a initial mismatch between boundary conditions and initial water level, the initial water level is set to 1.506 m+NAP, which is the initial value of the (homogeneous) boundary conditions signal.

Results

Illustrative results for the water level and the velocity magnitude for an observation point in the wake of the obstacle are shown in Figure 14.5 as an example.

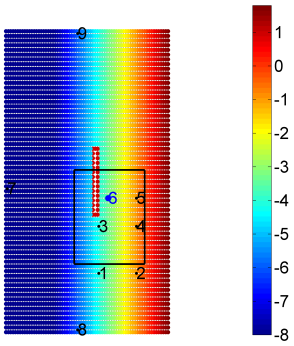
For this testcase, it is seen that the water levels computed by Delft3D and D-Flow FM coincide. The velocities, however, show different behavior as these develop in time: the trend fairly coincides, but the fluctuations significantly differ. An assessment on the basis of more analytical considerations is desired in order to examine whether of the two packages represent the soundest physical behavior.

Conclusion

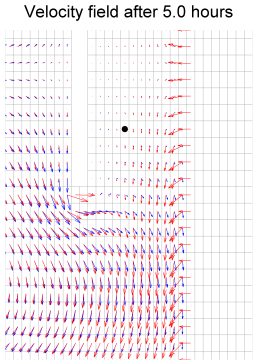
The test case for flow around an obstacle placed on an inclined bed could be used as a test case for the joint functionality of D-Flow FM with the external wave module. However, one should be aware of the above described descrepancies.

Version

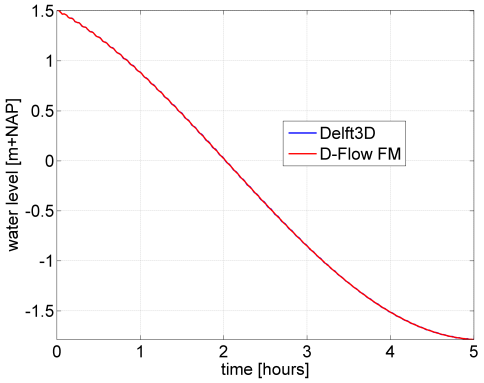
This test has been carried out with version dflow-fm-x64-1.1.116.36086.



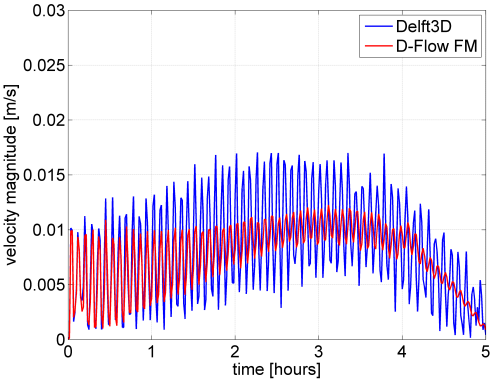
(a) *The grid, the bathymetry and the locations of the observation points (box highlighted in Figure 14.5b).*



(b) *Velocity vectors after 31.5 hours from the start of the simulation (Delft3D in blue, D-Flow FM in red).*



(c) *Timeseries of the water level at observation location 65 (see Figure 14.5a).*



(d) *Timeseries of the velocity magnitude at observation location 65 (see Figure 14.5a).*

Figure 14.5: *Analysis of the realistic Frisian Inlet case.*

14.6 Spiderweb winds

Purpose

Hurricanes or cyclones can be represented in a hydraulic computation as an external force by imposing a spiderweb wind field. This testcase considers the results of a D-Flow FM computation with a spiderweb wind field.

Linked claims

Claims that are related to the current test case are:

- ◇ claim 2.3.1.2: D-Flow FM is suitable for the prediction of the tidal dynamics in estuaries or coastal seas
- ◇ claim 2.3.5.3: D-Flow FM can be used for the prediction of storm surges due to cyclone winds
- ◇ claim 2.3.5.4: D-Flow FM can accurately simulate the effects of space and time varying wind stresses at the free water surface

Approach

A test model from the Delft3D testbench is converted to a D-Flow FM equivalent. This model comprises a schematisation of the Mexican Gulf area, with the hurricane Katrina represented through a spiderweb wind field file.

Model description

The Delft3D-model containing the model schematization contains 170×127 grid cells. This grid is converted to a D-Flow FM grid. This grid is shown in Figure 14.6. In addition, a second grid is generated containing the domain discretization through triangles. The boundary conditions are specified as water levels, by means of astronomic components. To benefit the transparency of the computational results, the computations is started from an initial flow field yielding zero velocities and a plane water level equal to 0 m with respect to the reference level. For the same purpose, no salinity boundary conditions are imposed.

Relevant input settings are:

- ◇ the averaged pressure is 101325 N/m^2, as is the average air pressure on the open boundaries,
- ◇ the density of the water and the density of the air are 1025.0 kg/m^3 and 1.2 kg/m^3 respectively,
- ◇ the bottom friction is set equal to 0.024 (as Manning coefficient),
- ◇ the horizontal eddy viscosity is set equal to 0 m^2/s,
- ◇ the wind drag coefficient is specified as a function of the wind speed through 2 breakpoints, as is the default.

The cyclone is specified by means of the file `katrina.spw`. This file contains spatial information of the wind speed magnitude (m/s), its direction (in degrees) and the pressure drop as a function of time. Each 180 minutes, a spatial distribution is provided in a polar coordinates reference system. The cyclone eye trajectory is visualized by means of red dots in Figure 14.6.

A Delft3D computation is used to compare the D-Flow FM output with. The settings of the Delft3D computation is chosen exactly the same as the D-Flow FM computation to ensure sound comparison. Output data are evaluated at the observation points as shown in Fig-

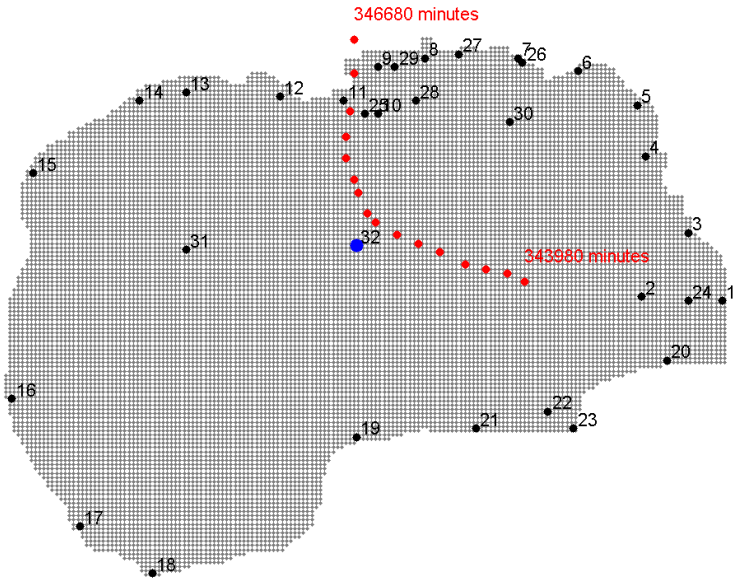


Figure 14.6: *Spherical grid covering the Mexican Gulf area south of New Orleans. The black dots denoted observation points; the red dots denote the cyclone eye trajectory in time.*

ure 14.6.

Results

In each grid cell an observation point is present. Observation point numbers 32 (shown in blue in Figure 14.6), 25 and 11 are chosen for comparison of the D-Flow FM output with the Delft3D output. The water level time series from both packages are shown in Figure 14.7. The three observations are the most interesting, since these are located close to the path of the eye of the cyclone.

Figure 14.7 shows the results of a computation *without* the cyclone wind imposed (left panels) and the results of a computation *with* cyclone wind imposed (right panels). The first computation is used to be able to make distinction between the differences due to wind and differences due to other causes.

The left panels of Figure 14.7 show that D-Flow FM slightly deviates from the Delft3D results: the maximum water levels, computed by D-Flow, at the three locations considered are slightly lower compared to Delft3D. A similar conclusion holds for the computation with the spiderweb wind field, be it slightly more pronounced.

However, the more one tends to the coast, the differences between the Delft3D-results and the D-Flow FM results increase (for instance at point 12, not shown here). This might be due to the inactivation of advection terms in this area, in Delft3D. Otherwise, these differences might have a common cause compared to the simple testcase of wind over a schematized lake.

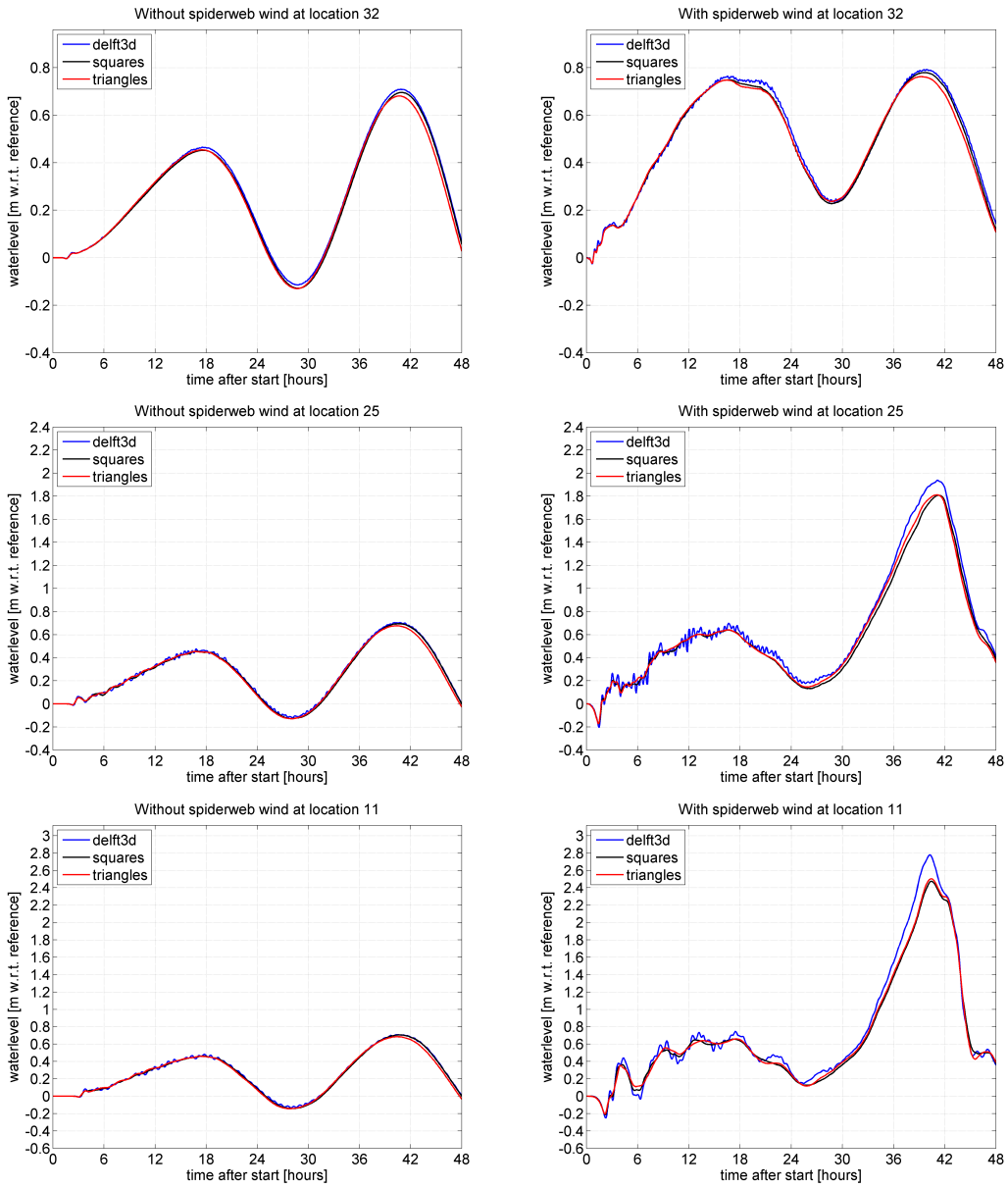


Figure 14.7: *Computed water level signals at three locations across the domain. Left panels: results of the computation* without *the cyclone wind; right panels: results of the computation* with *cyclone wind.*

Conclusion

The spiderweb wind functionality appears to work properly. However, near the coast of the domain, differences between D-Flow FM and Delft3D tend to increase.

Version

This test has been carried out with version dflow-fm-x64-1.1.116.36629.

15 In depth considerations

15.1 Dam break over a wet bed: quantitative analysis on numerical accuracy

Purpose

In other sections, the accuracy of D-Flow FM has been verified for a one-dimensional model of flooding over a wet bed resulting from a dam break. The verification of the accuracy was done there in a highly *qualitative* form, through graphical comparison of the computed state variables (water depth h and flow velocity u) and the analytical solution of the flow equations. In this section this one-dimensional dam-break case study is revisited, but now the accuracy with which the analytical solution is (re)produced by D-Flow FM is analysed *quantitatively*. In particular it is verified how the accuracy of the model predictions responds to refinements of the computational grid (Δx) and the time step (Δt), and thus obtain insight in the order of accuracy of D-Flow FM's numerical scheme.

Linked claims

Claims that are investigated in the current test case are:

- ◇ The accuracy of the time integration scheme in D-Flow FM appears to be of first order in this application.
- ◇ The accuracy of the spatial discretisation scheme is of first order.

Approach

In the present validation tests the same model has been used as earlier in this document, with in particular the same geometry, computational settings, and model parameters (initial and boundary conditions, length L of the domain, location x_0 of the dambreak, etc.). These and other settings that have been used in the various D-Flow FM computations will be mentioned below in a separate section.

For the present one-dimensional dambreak over a wet bed an analytical solution of the flow equations is available. This analytical solution (also recapitulated below in a separate section) forms the basis for our quantitative analysis of the numerical accuracy of D-Flow FM. In this analysis the following approach is adopted.

D-Flow FM simulations have been carried out for various systematic variations of both the grid size in the spatial discretisation, and the time step in the numerical time integration of the flow equations.

In the variations of the spatial discretisations four uniform rectangular grids are used, i.e. grids with cells of a constant size Δx along the longitudinal (x) direction of the computational domain. The cells of these grids are respectively of size $\Delta x = 500, 100, 50$ and 25 m. With a longitudinal length $L = 60$ km of the domain this means that in the x-direction these grids consist of 120, 600, 1200, and 2400 cells. The four grids that are here considered therefore provide stepwise (and mutually 'comparable') refinements of the spatial resolution.

Compared to other, conciser tests a main extension within the validation study is that also the effect of the time step Δt on the accuracy is investigated by varying Δt in the numerical integration of the model equations. These variations of Δt are within a range of 0.01 to 5 seconds.

The effect of grid-size and time-step refinements on the accuracy of D-Flow FM simulations

will be quantified through statistical error measures derived from the errors $\Delta h(x,t)$ and $\Delta u(x,t)$ in the model's prediction of the water depth $h(x,t)$ and flow velocity $u(x,t)$ at the cell centres[1]. The errors are defined by $\Delta h(x,t) := h_A(x,t) - h_C(x,t)$ and $\Delta u(x,t) := u_A(x,t) - u_C(x,t)$, with subscript C indicating a variable computed by D-Flow FM and with subscript A indicating the analytical solution. Apart from the water depth and the velocity, the errors in the computed flow discharge $\Delta(h(x,t) \cdot u(x,t))$ will be analysed as well.

Three error measures L_1, L_2, and L_∞ will be considered for the quantification and comparison of the accuracy of the D-Flow FM solutions for the various variations in grid size and time step. These L_k ($k = 1, 2, \infty$) are actually vector norms applied to the set of model errors $\{\Delta h(x,t)\}_{(x,t)\in A}$ and $\{\Delta u(x,t)\}_{(x,t)\in A}$ on some sub-area A of the computational domain. They are defined by the following generic formula[2]:

$$L_k(\Delta\xi) = \sqrt[k]{\frac{1}{N_A}\sum\nolimits_{(x,t)\in A} |\Delta\xi(x,t)|^k} \tag{15.1}$$

In these equations, N_A denotes the number of cell centres $(x,t) \in A$. The ξ represents one of the three considered flow quantities, i.e. water depth h, flow velocity u, or the discharge $h \cdot u$.

The error measures will be computed for several sub-areas A of the (spatial and temporal) computational domain. The selection of the sub-areas is guided by the regions in the (x,t) domain where the analytical solution of the flow variables $h(x,t)$, $u(x,t)$ and $h(x,t)\cdot u(x,t)$ has specific properties. In one of the sections below this selection of adequate sub-areas is adressed in detail. Next, the dependency of the error norms on the grid size Δx and time step Δt will be evaluated for the various sub-areas, from which the order of accuracy of the numerical solutions can be obtained.

Analytical solution for the one-dimensional dambreak over a wet bed

The spatial domain is assumed to range from from $x = 0$ to $x = L$, and the location of the dambreak is at $x = x_0$. The initial system state (at $t = 0$, denoting the begin of the dambreak) is a stepwise water depth with $h(x) = h_l$ for $0 \le x \le x_0$ and $h(x) = h_r$ for $x_0 < x < L$. This longitudinal water depth profile thus includes a discontinuity at $x = x_0$.

For the case of no bed friction, and no viscosity effects, an analytical solutiuon is available for the flow equations (see Delestre et al., 2011) . For a convenient formulation of this analytical solution three linear separators $x_A(t)$, $x_B(t)$, and $x_C(t)$ of the (x,t)-model domain are

[1] Ideally, the error across the entire domain should be considered. It is, however, not clear how the discrete solution at cell centres should be extended in between the cell centres. Water depth h is assumed piecewise constant per grid cell in the time derivative of the continuity equation (a central approximation), but its value at the cell faces is obtained by (first-order) upwind. Velocity u at cell centers is obtained from averaging the values at the faces of the cells, thereby introducing averaging errors. Although in 1D u at cell faces could be considered when the grid is aligned with the flow (as is the case here), this is not possible in general, which is why we decided not to exploit this possibility.

By only considering the value of h and u at cell centers, at most additional second-order errors are introduced due to the use of the 1-point Gauss integration rule (h) and 2-point central averaging (u). If necessary, this should be taken into account when interpreting the results.

Note that the discharges computed at the cell centers do *not* correspond with the discharges at the cell faces used in the D-Flow FM simulations. Because of the use of a first-order upwind h in the latter, the deviation will in general be first order. This is a discretization error, *not* an error in the interpretation of the results.

[2] Note that $L_1(\cdot)$ is the mean absolute error over A, while $L_2(\cdot)$ is the RMS (root mean square) of the model error. It is readily verified that $L_\infty(\cdot)$ corresponds to the maximum of the pointwise absolute errors $|\Delta\xi(x,t)|$ over A.

defined according to:

$$\begin{cases} x_A(t) = x_0 \ - \ t \cdot \sqrt{g \cdot h_l} \\ x_B(t) = x_0 \ + \ 2t \cdot \sqrt{g \cdot h_l} - 3tc_m \\ x_C(t) = x_0 \ + \ 2t \cdot \dfrac{c_m^2 \left(\sqrt{g \cdot h_l} - c_m \right)}{c_m^2 - g \cdot h_r} \end{cases} \tag{15.2}$$

These separators induce four partitions of the model domain and on these sub-domains the analytical solution for the water depth reads:

$$h(x,t) = \begin{cases} h_l, & \text{if} \quad x < x_A(t) \\ \dfrac{4}{9g} \left(\sqrt{g \cdot h_l} \ - \ \dfrac{x \ - \ x_0}{2t} \right)^2 & \text{if} \quad x_A(t) \leq x \leq x_B(t) \\ \dfrac{c_m^2}{g} & \text{if} \quad x_B(t) \leq x \leq x_C(t) \\ h_r & \text{if} \quad x > x_C(t). \end{cases} \tag{15.3}$$

The analytical solution for the velocity along the four sub-domains is provided by:

$$u(x,t) = \begin{cases} 0 & \text{if} \quad x < x_A(t), \\ \dfrac{2}{3} \cdot \left(\dfrac{x \ - \ x_0}{t} + \sqrt{g \cdot h_l} \right) & \text{if} \quad x_A(t) \leq x \leq x_B(t), \\ 2 \cdot \left(\sqrt{g \cdot h_l} - c_m \right) & \text{if} \quad x_B(t) \leq x \leq x_C(t), \\ 0 & \text{if} \quad x > x_C(t). \end{cases} \tag{15.4}$$

These analytical solutions involve a velocity c_m that depends on the water depths h_l and h_r at the left and right hand boundaries of the domain. This dependency of c_m and h_l and h_r is implicitly defined by the equation:

$$-8gh_r c_m^2 \left(\sqrt{gh_l} - c_m \right)^2 + \left(c_m^2 - gh_r \right)^2 \left(c_m^2 + gh_r \right) = 0. \tag{15.5}$$

See Section 4.1.1 in Delestre et al. (2011). For given h_l and h_r a numerical root finding method must be used to derive c_m.

Later in this chapter illustrations of these analytical solutions for $h(x,t)$, $u(x,t)$, and the discharge $h(x,t) \cdot u(x,t)$ will be shown, together with corresponding numerical solutions according to D-Flow FM.

Model description

In all computations the following values were adopted for the flow and geometric parameters present in the analytical 1D dambreak model of Equation (15.9) and Equation (15.10).

- ◇ domain length: $L = 60,000$ m (and thus $0 \leq x \leq 60,000$ m).
- ◇ Water depth boundary conditions: $h_l = 2$ m and $h_r = 0.1$ m.
- ◇ Location of the dam break: $x_0 = 30,000$ m.
- ◇ Gravity acceleration: $g = 9.81$ m/s^2 .

For these parameters the following value is found for the velocity c_m implicitly defined by Equation (15.7): $c_m = 2.4665507$ m/s. This velocity corresponds to a water depth $h_m := {c_m}^2/g = 0.6201705$ m.

The same computational grids have been used as earlier in this document, although only the uniform rectangular grids 5, 1, 2, 3 are now taken into account[3]. These four grids are all built up of equally sized rectangular cells with $\Delta x = 500, 100, 50$ and 25 m respectively. They thus provide stepwise (and mutually 'comparable') refinements of the spatial resolution.

In the assessment of the sensitivity and accuracy for the time step D-Flow FM simulations have been carried out with eight variations of Δt: $\Delta t \in \{0.01, 0.05, 0.125, 0.25, 0.5, 1.0, 2.0, 5.0\}$ (all in seconds). In each case the simulation period $T = 3600$ sec.

The computational settings in the D-Flow FM simulations are as follows:

- ◇ No bed friction, no surface drag, no horizontal viscosity.
- ◇ Schematisation of advection: AdvecType= 3 (Perot q(uio-u)).
- ◇ Limiter type for cell center advection velocity: Limtypmom= 4 (Monotone Central).
- ◇ Time integration: TimeStepType= 2 (full implicit step reduce). $\theta = 0.55$

Selection of the sub-areas in the model error analysis

The spatial and temporal evolution of the flow variables in the present case of a dam break over a wet bed are depicted in Figure 15.11, Figure 15.12 and Figure 15.13. The blue curves

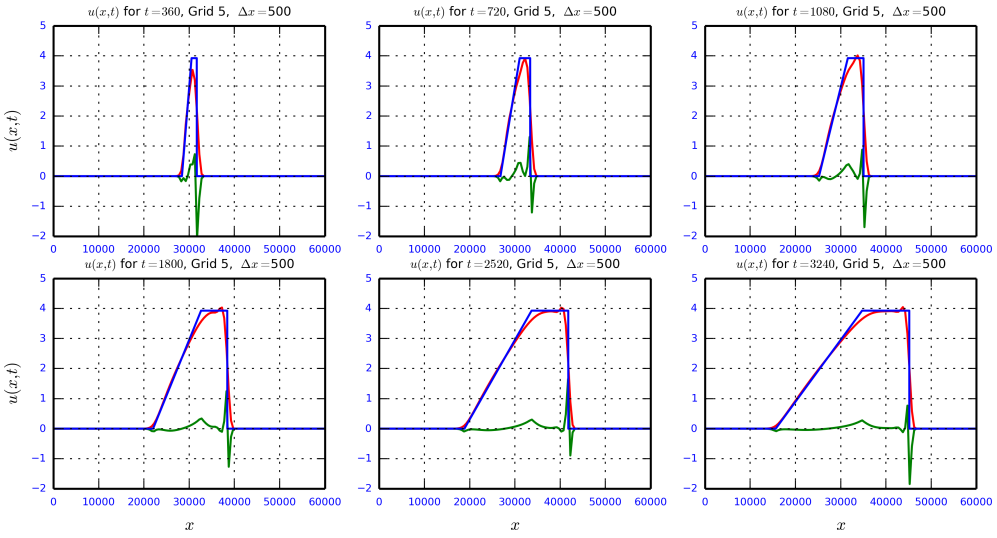


Figure 15.1: *Velocity $u(x,t)$ as function of the longitudinal coordinate x for 6 successive time points t. The velocity $u_A(x,t)$ according to the analytical solution is plotted in blue, while the solution $u_C(x,t)$ according to D-Flow FM using Grid 5 is shown in red. The error in the computed velocity, $\Delta u(x,t)$, is depicted in green.*

in these figures represent the $u(x,t)$, $h(x,t)$ and $h(x,t) \cdot u(x,t)$ according to the analytical solution of the flow equations (see Equation (15.9) and Equation (15.10)).

The red curves in the panels of the three figures show these flow variables as computed by

[3]The uniform triangular grid 4, the non-uniform rectangular grid 6, and the rectangular grid with local refinements and irregular triangular connections 7 have been discarded. Results obtained on these grids are not suitable for consideration in the present systematic study on accuracy as a function of grid resolution and time step.

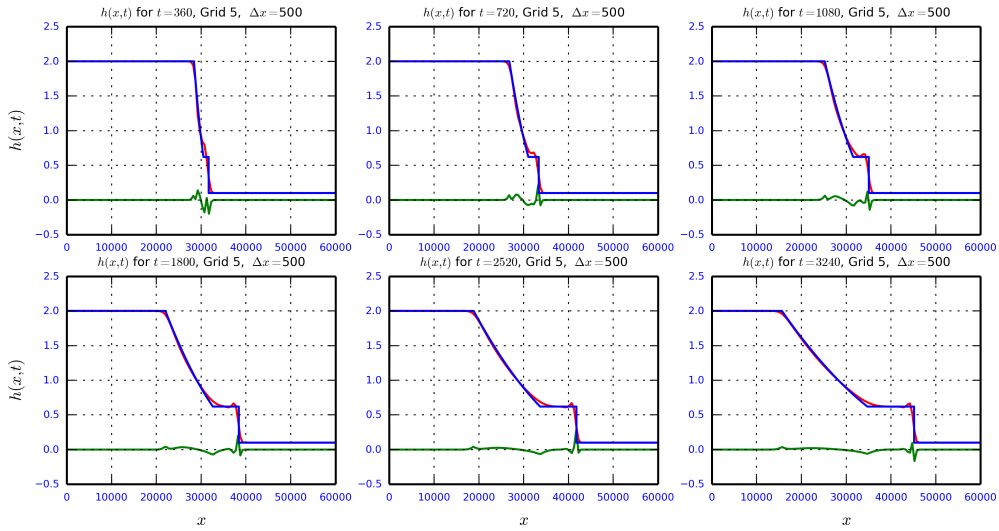


Figure 15.2: *Water depth $h(x,t)$ as function of the longitudinal coordinate x for 6 successive time points t. The water depth $h_A(x,t)$ according to the analytical solution is plotted in blue, while the solution $h_C(x,t)$ according to D-Flow FM using Grid 5 is shown in red. The error in the computed water depth, $\Delta h(x,t)$, is presented in green.*

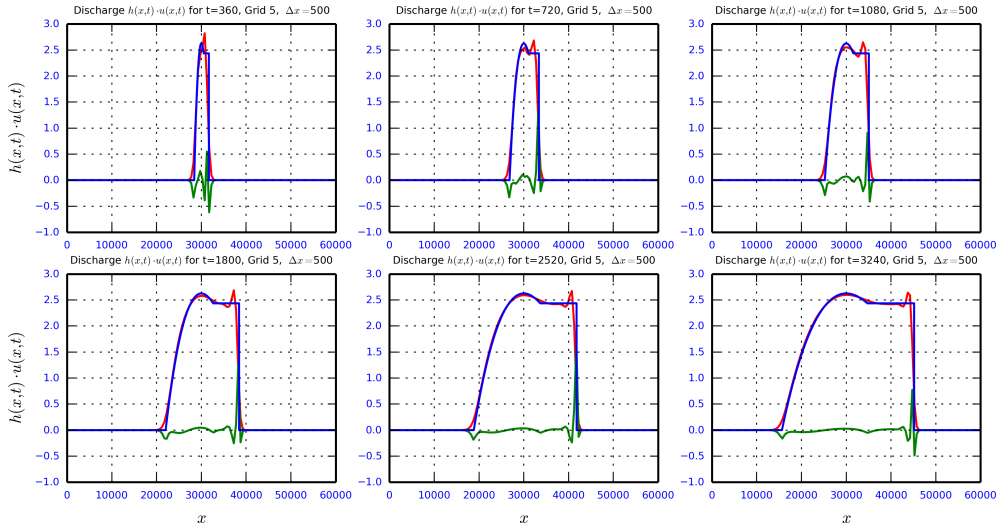


Figure 15.3: *Discharge $h(x,t)u(x,t)$ as function of the longitudinal coordinate x for 6 successive time points t. The discharge according to the analytical solution is plotted in blue, while the corresponding solution of D-Flow FM using Grid 5 is shown in red. The error in the computed discharge is presented in green.*

D-Flow FM, using the coarsest grid 5 with $\Delta x = 500\ m$ in combination with a very small time step of $\Delta t = 0.01$ sec to ensure negligibly small time integration errors. The difference between the red curves and the blue curves in the figures are therefore uniquely due to the spatial discretisation of D-Flow FM.

In the plotting procedure the computed solutions (red) were plotted before the analytical solutions (blue). As a result, the red curves are masked by the blue curves where and when the computed solution agrees well with the analytical solution. Loosely stated, it can thus be noted that the *less* is seen of the red curves, the *smaller* the errors, the *better* the quality of the numerical simulation. For a more comprehensive inspection of the quality of the nu-

merically computed solution, the errors $\Delta u(x,t)$, $\Delta h(x,t)$ and $\Delta(h(x,t) \cdot u(x,t))$ have also been plotted separately in Figure 15.11, Figure 15.12 and Figure 15.13. See the curves presented in green.

To verify the effect of grid refinement on the error in the model predictions, the same plots, but now for the finest Grid 3 with $\Delta x = 25$ m, are presented in Figure 15.14, Figure 15.15 and Figure 15.16. The time step is again $\Delta t = 0.01$ sec and small enough for the time integration errors to be negligible.

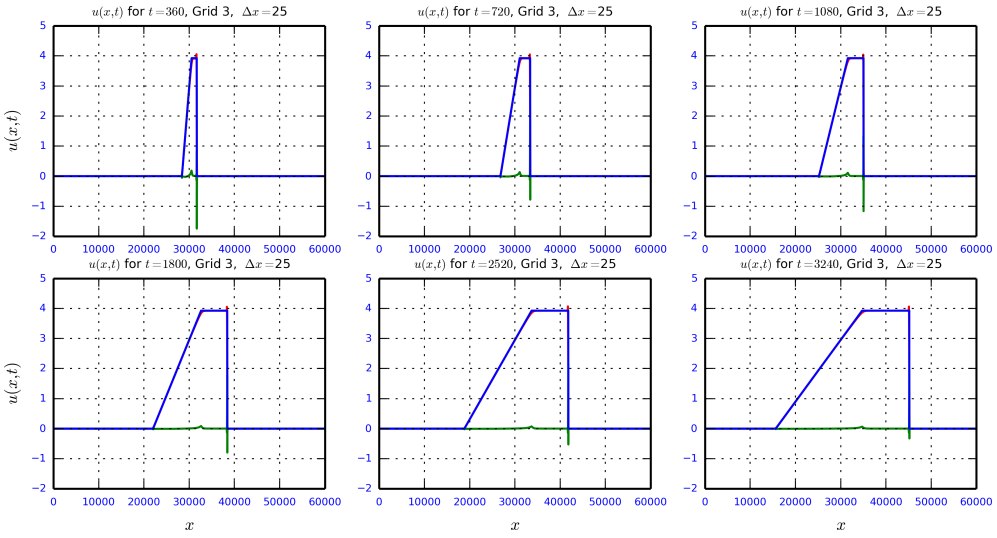


Figure 15.4: *Velocity $u(x,t)$ as function of the longitudinal coordinate x for 6 successive time points t. The velocity $u_A(x,t)$ according to the analytical solution is plotted in blue, while the solution $u_C(x,t)$ according to D-Flow FM using Grid 3 is shown in red. The error in the computed velocity, $\Delta u(x,t)$, is depicted in green.*

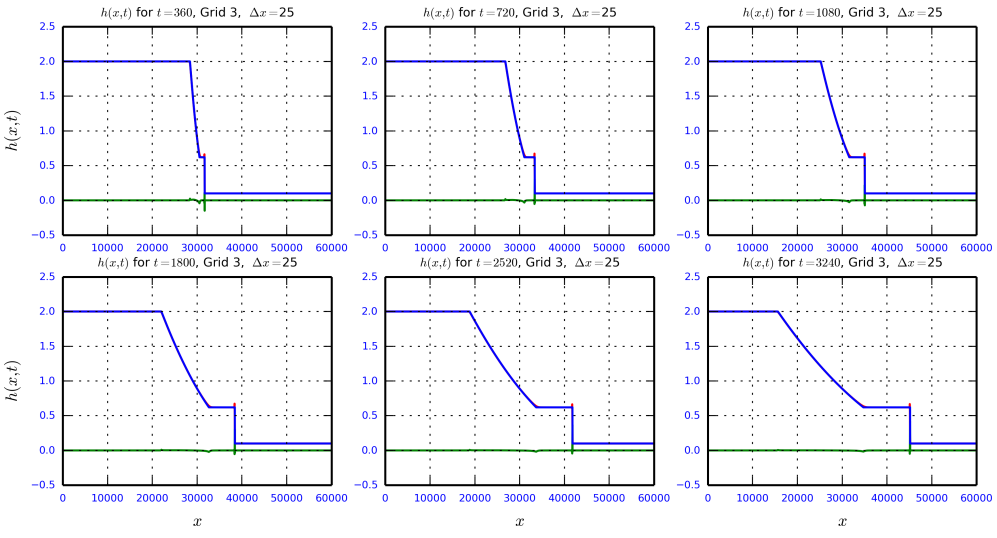


Figure 15.5: *Water depth $h(x,t)$ as function of the longitudinal coordinate x for 6 successive time points t. The water depth $h_A(x,t)$ according to the analytical solution is plotted in blue, while the solution $h_C(x,t)$ according to D-Flow FM using Grid 3 is shown in red. The error in the computed water depth, $\Delta h(x,t)$, is presented in green.*

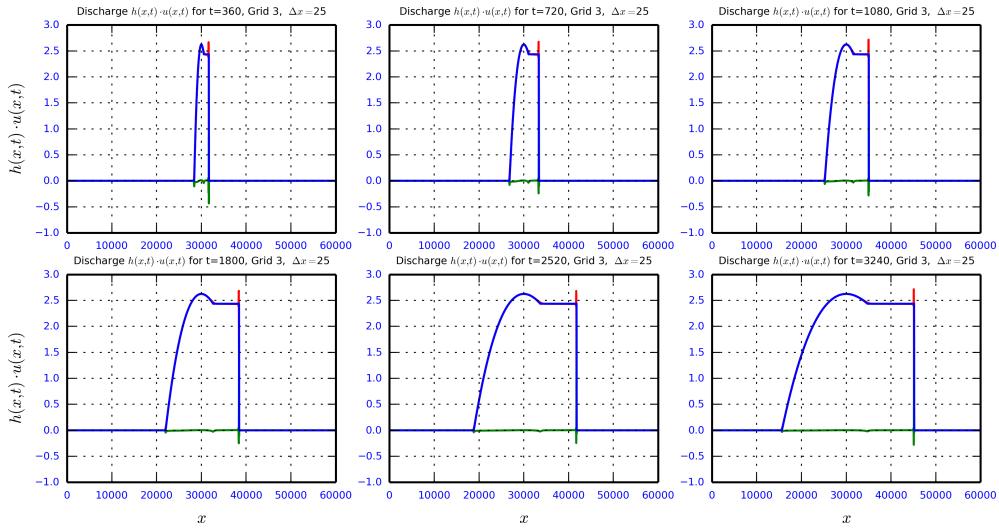


Figure 15.6: *Discharge $h(x,t)u(x,t)$ as function of the longitudinal coordinate x for 6 successive time points t. The discharge according to the analytical solution is plotted in blue, while the corresponding solution of D-Flow FM using Grid 3 is shown in red. The error in the computed discharge is presented in green.*

From the formulas and the figures of the analytical solution four sub-areas of different flow regimes can be recognised in the spatio-temporal domain $\Omega := \{(x,t)|t \geq 0\}$. These four sub-areas are bounded by the three concurrent straight lines $\{(x_A(t),t)\}$, $\{(x_B(t),t)\}$ and $\{(x_C(t),t)\}$. See Equation (15.8) for the expressions of these $x_A(t)$, $x_B(t)$ and $x_C(t)$. In the *lower* panel of Figure 15.17 the location of these lines within Ω is shown (plotted in red), together with the four sub-areas $\{\Omega_1,\Omega_2,\Omega_3,\Omega_4\}$ that they determine. Through the graphs of the water depth (red), velocity (blue) and discharge (green) depicted in the *upper* panel of Figure 15.17 (with h, u, and $h\cdot u$ as function of x, for fixed time $t = 3240$ sec) the relation of the sub-areas Ω_i with the different flow regimes can be conveniently verified.

In the sub-areas Ω_1 and Ω_4 the solution of the flow equations is constant and equal to the initially imposed conditions. In a validation of the model's quality these sub-areas are thus of minor or no interest.

In sub-area Ω_3 the water depth and velocity are also constant but with values depending on the (initial) water depth in the left (Ω_1) and right (Ω_4) part of the domain. The accuracy with which these constants are reproduced by the numerical model is included in the present validation study.

In sub-area Ω_2 the water depth and velocity are not constant. Along the x-direction the water depth follows a monotonically decreasing parobolic profile, while the velocity increases linearly from 0 at $x = x_A(t)$ to its maximum at $x = x_B(t)$. The various panels in the figures Figure 15.11, Figure 15.12 and Figure 15.13 show how these profiles of the flow variables evolve in time. This temporal evolution consists of a gradual spatial stretching of the profiles in both the downstream and upstream direction. Within this stretching the shape of the profiles remains the same (self-similar solution). Because of the non-trivial variation, the solution in this sub-area Ω_2 is highly relevant in the present D-Flow FM validation.

From the figures of the analytical solutions presented above, it is recognised that the interface between the sub-areas Ω_i (formed by the lines $\{(x_A(t),t)\}$, $\{(x_B(t),t)\}$ and $\{(x_C(t),t)\}$) are of special interest. At these interfaces the flow variables are non-smooth (at $\{(x_A(t),t)\}$ and $\{(x_B(t),t)\}$) or even discontinuous (at $\{(x_C(t),t)\}$). In an investigation of the accuracy

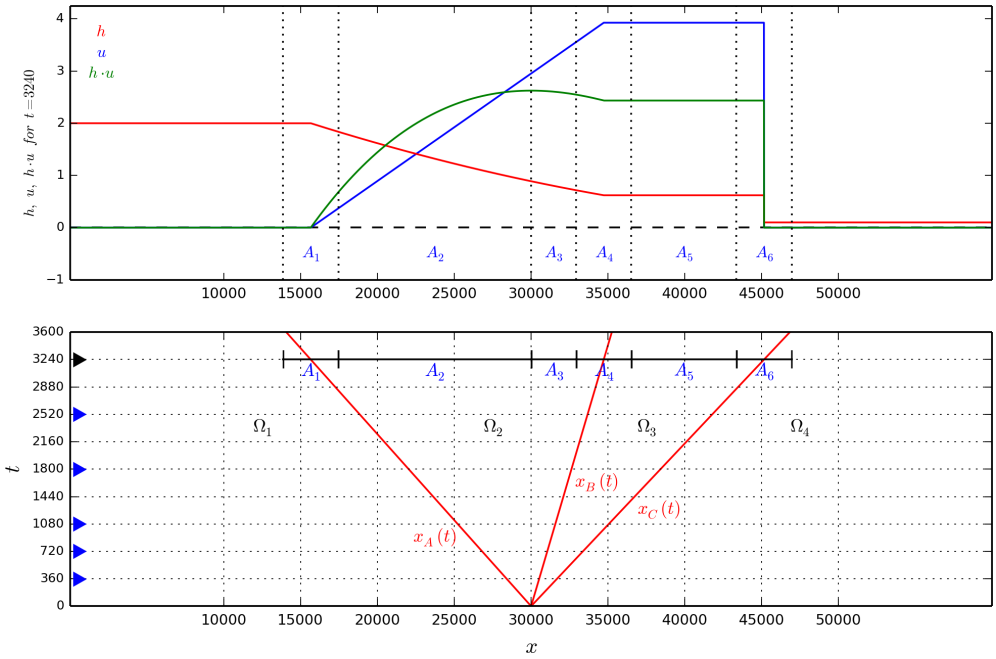


Figure 15.7: *Lower panel: Partition $\{\Omega_1, \Omega_2, \Omega_3, \Omega_4\}$ of the model's spatio temporal domain by the three lines $\{(x_A(t), t)\}$, $\{(x_B(t), t)\}$ and $\{(x_C(t), t)\}$, and the sub-areas A_i where the error norms are evaluated. The symbols ▶ mark the times t for which in preceding figures the flow variables u, h, and $h \cdot u$ have been shown as function of the spatial variable x. In the upper panel these flow variables are once more illustrated for $t = 3240$ sec. In this way the different characteristics of the flow for the different sub-areas can be conveniently recognised. For this $t = 3240$ sec the various norms for the D-Flow FM errors have been evaluated.*

of numerical flow models such as D-Flow FM, the capability to deal with such non-smoothness is an important issue. For the present case this importance is revealed by the errors plotted in Figure 15.11 and Figure 15.12. It can be observed that the errors reach local maxima at the interfaces between the sub-areas and that they are spread over quite a few grid cells $\Delta x = 500$m. The same behaviour is present in Figure 15.14 and Figure 15.15, but not clearly visible because of the use of the much smaller grid size $\Delta x = 25$m.

On the basis of these considerations the assessment of the accuracy of the D-Flow FM simulations is carried out for the two sub-areas Ω_2 and Ω_3, and for small neighbourhoods of their boundaries x_A, x_B and x_C. For the sub-area Ω_2 another partition is made, consisting of the part to the left and the part to the right of the original dam-break position x_0. The reason for this is the observation (cf. Figure 15.11 and Figure 15.12) that in sub-area Ω_2 the errors in the computed water depth and the velocity tend to be somewhat larger for $x > x_0$ than for $x < x_0$.

A good impression of the simulation accuracy of D-Flow FM is obtained by computing and presenting the error norms of $\Delta u(x, t)$ and $\Delta h(x, t)$ for a time level toward the end of the simulation. This will exclude spurious effects due to any excessive errors in the start-up phase of the computation that may have been caused by the discontinuous initial condition. We have chosen the time level $t = 3240$sec, which in the previous figures corresponds with the solutions shown in the lower-right panels.

Altogether this gives the following sub-areas A_i at $t = 3240$ sec for which results of the accuracy assessment will be presented in the next section:

1 Area A_1 containing the grid points x_n for which $13849\text{m} < x_n < 17449\text{m}$;
2 Area A_2 containing the grid points x_n for which $17449\text{m} < x_n < 30000\text{m}$;
3 Area A_3 containing the grid points x_n for which $30000\text{m} < x_n < 32928\text{m}$;
4 Area A_4 containing the grid points x_n for which $32928\text{m} < x_n < 36528\text{m}$;
5 Area A_5 containing the grid points x_n for which $36528\text{m} < x_n < 43365\text{m}$;
6 Area A_6 containing the grid points x_n for which $43365\text{m} < x_n < 46965\text{m}$.

The location of these sub-areas within the computational domain Ω, and the characteristics of the flow variables on these sub-areas is shown in Figure 15.17.

Finally it is mentioned that the analytical solution has been used to construct plots of the relevant components of the momentum equation: $\frac{\partial(h\cdot u)}{\partial t}$ (storage term), $\frac{\partial(h\cdot u^2)}{\partial x}$ (convection term), and $gh \cdot \frac{\partial h}{\partial x}$ (pressure term). All other components, related to bed friction, horizontal viscosity, Coriolis, etc. are zero. The plots of the three relevant components (not presented here) were used to verify the relative weight that these components contribute to the flow propagation. With the present setting of the model parameters (h_l and h_r) is was oberved that in magnitude these components are reasonably in balance. As a result they contribute significantly, and more or less equally, to the results and conclusions of this validation study. These results are summarised in the next section.

Results

For each of the four uniform grids (with the grid sizes $\Delta x = 500$, 100, 50 and 25 m) D-Flow FM simulations were carried out for the eight different time steps $\Delta t = 0.01$, 0.05, 0.125, 0.25, 0.5, 1.0, 2.0 and 5.0 sec. For all these simulations, the three error norms $L_1(\cdot)$, $L_2(\cdot)$ and $L_\infty(\cdot)$ (see Equation (15.6)) were evaluated. This was done for all the six sub-areas A_i that were selected in the preceding paragraph.

To begin with, the effect of the time-step size has been verified using the spatial grid with the finest resolution ($\Delta x = 25$ m). For this finest grid (and any other grid if larger time steps would have been considered) we could have that with increasing Δt the errors in the D-Flow FM simulations gradually become mainly, or even fully, determined by the errors in the applied time integration method. From the dependency of the error norms on Δt the time accuracy of D-Flow FM can be evaluated and quantified.

The result is shown in Figure 15.8. To limit the total number of figures, and also because the behaviour of the other two error norms is similar, only the error behaviour in the L_1 norm is shown in Figure 15.8. For all variables, and in all sub-areas, we see roughly the same error behaviour. For large Δt an error behaviour close to first order (error proportional to Δt) is observed, which is in agreement with the applied time integration scheme (first-order explicit in the convection term $u \cdot \frac{\partial u}{\partial x}$, and first-order implicit in the pressure term $g \cdot \frac{\partial h}{\partial x}$ of the momentum equation).

We recall that the error behaviour is sensitive to the way that the error has been measured, cf. Footnote 11. This may be the reason why the error in sub-area A_6, where the hydraulic jump is, seems to be higher order in Δt (see again Figure 15.8). Unless special shock-fitting techniques are used, steady discontinuities like hydraulic jumps by definition limit the accuracy of *any* numerical scheme to $O(\Delta x)$ in the L_1 norm, to $O(\sqrt{\Delta x})$ in the L_2 norm, and to $O(1)$ in the L_∞ norm. The same applies in time for moving discontinuities, which is the case that we are dealing with here. In other words, the seemingly higher-order behaviour of the D-Flow FM time integration scheme in sub-area A_6 must be a spurious effect. It is probably due to the fact that the numerical solution and the exact solution are not compared

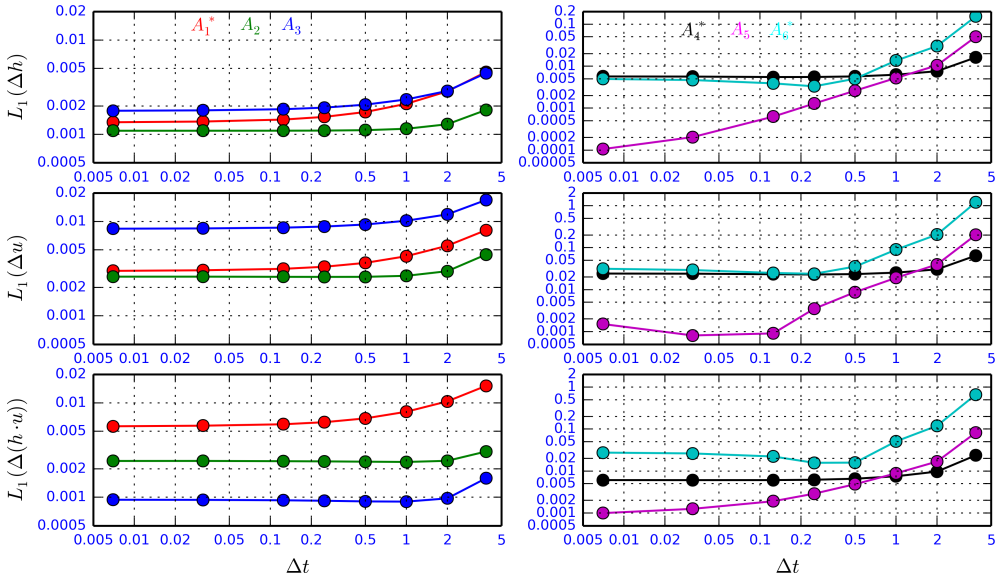


Figure 15.8: *Dependency of the error norm $L_1(\cdot)$ on Δt, for the water depth, velocity and discharge on one hand, and on the other hand on the six sub-areas A_i that were selected in the validation of the accuracy of the numerical scheme. These $L_1(\cdot)$ are as obtained with the presently finest grid size $\Delta x = 25$ m.*

everywhere in the domain[4], but only at a limited number of points.

The error reaches a constant value for $\Delta t < 0.5$sec, indicating that for time steps that are that small the time integration error becomes (negligibly) small compared to the space discretisation error.

In practical applications a suitable time step would be the one that ensures a balance between time integration errors and space discretisation errors. The small increase in accuracy of using a smaller time step would not outweigh the larger additional computational costs of a smaller time step, while with a larger time step (and hence a dominating time integration error) one would not fully exploit the accuracy provided by the spatial grid resolution. For the present case, Figure 15.8 shows that the optimal time step is around 1 to 2 sec. Since the grid size in this computation is 25m and the maximum flow velocity is 4m/sec, this means that the optimal Courant number $|u|\Delta t/\Delta x$ is about 0.2 to 0.3. It has been verified that this also applies to the computation with a grid size of 100m, so this conclusion is grid-resolution invariant, at least for this application.

A Courant number of 0.2 to 0.3 is significantly lower than the stability limit $|u|\Delta t/\Delta x \leq 1$. The conclusion is therefore that for this application (where we have used a uniform rectangular grid) the quality of the D-Flow FM space discretisation is better than that of its time integration. A consequency of this is that in highly unsteady applications (such as the one considered here), the time step may have to be taken smaller than the one that follows from the stability condition in order to ensure optimal performance of D-Flow FM.

The error behaviour for sub-area A_5 shown in Figure 15.8 is remarkable. It continues to decrease for time steps smaller than 0.5sec. This applies in particular to the error in the water depth. Sub-area A_5 is the only part of the domain where the solution is both non-trivial and constant, and where the space discretisation therefore becomes exact. As this

[4] That would require the numerical solution to be defined as a (discrete) *function* in space, and not as values in only a finite number of (grid) points as we do here.

sub-area is free of space discretisation errors, any error for very small time step must be due to the solution predicted in the sub-areas adjacent to A_5. An explanation for the remarkable error behaviour in A_5 may lie in the conservation properties of the space discretisation. It seems that this ensures that, regardless of any other errors, the water depth and flow velocity in sub-area A_5 are exact. However, the applied time integration scheme is *not* momentum conserving. This would explain why in sub-area A_5 the error is mainly determined by the applied time integration.

The accuracy of D-Flow FM with respect to the spatial discretisation has been verified in a similar way. To ensure that errors due to the time integration can safely be ignored, only results from the simulations with the extremely small time step ($\Delta t = 0.01$sec) have been considered. As before, we only show the error behaviour in the L_1 norm. For all variables and in all sub-areas we see in Figure 15.9 roughly the same error behaviour as a function of the spatial grid resolution. It turns out to be close to first order. This was expected for sub-area A_6 because of the presence of the hydraulic jump, but not for the other sub-areas where the solution is smoother. We expected in particular higher-order behaviour in the zones A_2 and A_3 where the solution is infinitely smooth, not to mention zone A_5 where the solution is even constant. The conclusion reads that somewhere in the D-Flow FM scheme a first-order approximation is applied (first-order upwind of the water depth in the determination of the mass flux through cell faces), which *for this application* has a dominant effect on the spatial accuracy[5]. Since a full description of the numerical scheme applied in D-Flow FM is not available, it is not possible to interprete the obtained validation results any further.

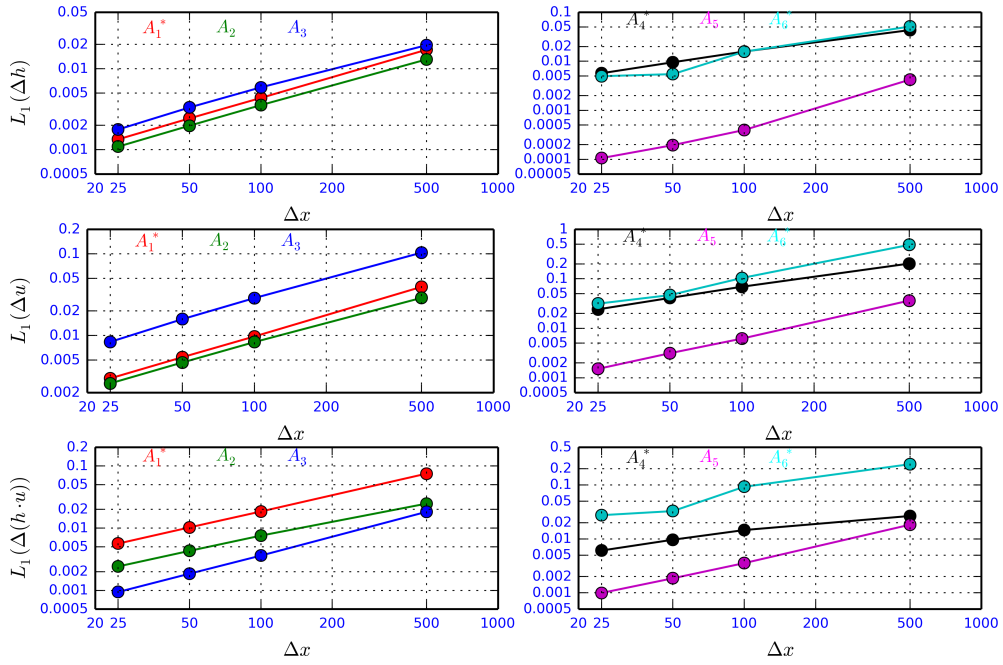


Figure 15.9: *Dependency of the error norm $L_1(\cdot)$ on Δx, for the water depth, velocity and discharge on one hand, and on the other hand on the six sub-areas A_i that were selected in the validation of the accuracy of the numerical scheme. These $L_1(\cdot)$ are as obtained with the presently smallest time step $\Delta t = 0.01$ sec.*

[5]Note that the way the (total) water depth is approximated at cell faces may have less impact on the overall accuracy in applications with little variation in the water level, such as tidal-flow and river-flow simulations. With the bathymetry defined at grid points, the depth with respect to the model datum is approximated second-order accurate at cell faces. With little variation in the water level, a first-order approximation error of taking the water level at cell faces upwind may have only a minor effect on the computation of the (total) water depth at cell faces. Additional validation studies are required to investigate this.

For a proper quantitative assessment of the order of accuracy, the effective order of accuracy as observed in the computations has been determined by means of linear regression analyses (of the form $\log(L_k(\Delta x)) \sim a \cdot \log(\Delta x) + b$) of the logarithm of the errors as a function of the logarithm of the grid size (i.e. the functions as shown in Figure 15.9). The error on the coarsest grid has been discarded in this analysis. On this coarsest grid with $\Delta x = 500$ m each sub-area only contains half a dozen or so grid cells, which is not enough for a reliable error analysis. This has been confirmed by looking at the correlation coefficients. They are close to 1 with the error on the coarsest grid discarded, but when this error is included the regression analyses give rather poor correlations.

The estimate for the slope parameter a in the regression $\log(L_k(\Delta x)) \sim a \cdot \log(\Delta x) + b$ provides the order of accuracy of the numerical scheme. The so obtained order-of-accuracy results are graphically shown in Figure 15.10. This is done for all L_k error measures and also all the sub-areas A_i that have been considered. We see indeed an error behaviour that is close to order 1, although in sub-area A_4 (where the solution is non-smooth but continuous) the error behaviour tends to be closer to order $1/2$. Rather surprising is the fact that the order of accuracy of the L_1 error of discharge hu in sub-area A_4 (where the solution is continuous) is noticeably *lower* than the one in sub-area A_6 (where the solution is *dis*continuous!). We have no explanation for this unexpected result.

The order of accuracy in sub-area A_5 tends to be somewhat higher than one. This has undoubtedly to do with the fact that the solution in this sub-area is constant and trivial to compute.

The error behaviour in sub-area A_6 with the hydraulic jump turns out to be (close to) first order in the L_1 norm (see again Figure 15.10), half order in the L_2 norm, and zero order in the L_∞ norm. These orders of accuracy agree, as already explained above, with the theory how discontinuities limit the accuracy of virtually *any* numerical scheme.

Conclusion

The main conclusions of the present D-Flow FM validation study can be summarised as follows.

- ◇ In agreement to what would be expected for the applied time integration scheme, the error behaviour with regard to the time step Δt is close to first order. This proportionality of the errors to the time step was found in all three error norms, and in all sub-areas with different flow characteristics.
- ◇ The accuracy with regard to the spatial discretisation is of first order in all norms, under the condition of a smooth (or at least continuous) dependency of the flow variables on the spatial and temporal coordinates. In sub-areas with discontinuities the order of accuracy depends on the applied error norm. For the maximum norm $L\infty$ this order is zero.
- ◇ For the present flow conditions (as briefly recapitulated below) a relatively small time step Δt is required to obtain a proper balance between the errors from the spatial discretisation on one hand and those of the time integration on other. In fact, for optimal performance the Courant number should be about 0.2 to 0.3, and is thus significantly lower than the stability limit of 1.
- ◇ Practically speaking D-Flow FM is well capable to reproduce the analytical solution: in absolute value the model errors are quite small. This is at the cost, however, of a small time step, in combination with a sufficiently fine grid.

It must be mentioned that these findings can partly (or even to a large extent) be induced by the present flow conditions, consisting of a non-steady, highly dynamic evolution of the flow, with non-smooth and discontinuous spatio/temporal variations (shock waves, jumps in

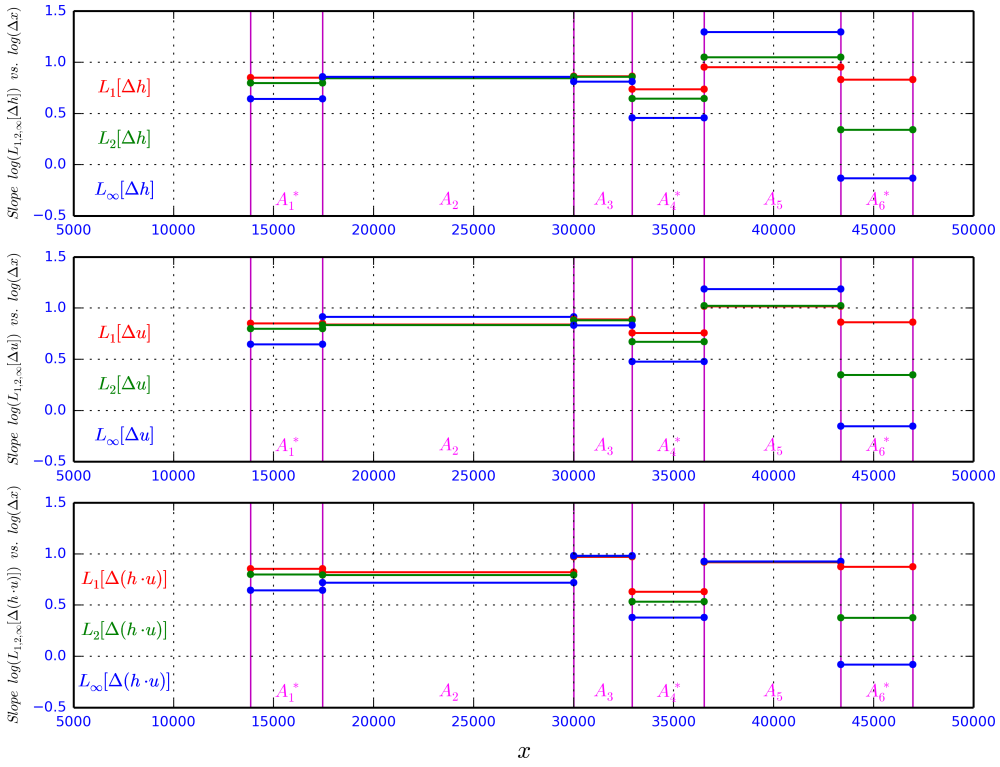


Figure 15.10: *Order of accuracies, $O(\Delta x)$, found for the various error norms $L_k(\Delta x)$. Through the three sub-panels the dependency of $O(\Delta x)$ on the flow quantities h, u, and $h \cdot u$ is presented. Within the sub-panels the dependency of $O(\Delta x)$ on sub-areas A_i in the computational domain is shown. These orders of accuracy are as obtained with the presently smallest time step $\Delta t = 0.01$ sec.*

the water depth and velocity profiles), and absence of friction and/or viscosity effects. At the same time it must be emphasized that the present conclusions are derived for, and may be limited to, uniform rectangular grids only.

Version

This test has been carried out with version dflow-fm-x64-1.1.111.34732M.

15.2 Dam break over a dry bed: quantitative analysis on numerical accuracy

Purpose

In other test cases, the accuracy of D-Flow FM has been verified for a one-dimensional model of flooding over a dry bed resulting from a dam break. The verification of the accuracy was done there in a highly *qualitative* form, through graphical comparison of the computed state variables (water depth h and flow velocity u) and the analytical solution of the flow equations. In this section this one-dimensional dam-break case study is revisited, but now the accuracy with which the analytical solution is (re)produced by D-Flow FM is analysed *quantitatively*. In particular it is verified how the accuracy of the model predictions responds to refinements of the computational grid (Δx) and the time step (Δt), and thus obtain insight in the order of accuracy of D-Flow FM's numerical scheme.

Linked claims

Claims that are investigated in the current test case are:

- ◇ The accuracy of the time integration scheme in D-Flow FM appears to be somehow less than first order in this application.
- ◇ The accuracy of the spatial discretisation scheme is of almost of first order.

Approach

In the present validation tests the same model has been used as earlier in this document, with in particular the same geometry, computational settings, and model parameters (initial and boundary conditions, length L of the domain, location x_0 of the dambreak, etc.). These and other settings that have been used in the various D-Flow FM computations will be mentioned below in a separate section.

It must be mentioned that in this chapter the D-Flow FM computations, and the associated accuracy analysis, will actually deal with the modelling of a *dambreak over an "almost" dry bed*, rather than the modelling of the flow over a *dry bed in strict sense*. An *"almost" dry bed* means that the water depth at the downstrean water level boundary (here and in the remainder of this chapter denoted by h_R) has been set to a small positive value instead of exactly zero as should be done for a dry bed in strict sense.

The reason for choosing an almost dry bed originates from D-Flow FM's numerical treatment of drying and flooding. In this treatment a threshold procedure is adopted involving a minimal positive water depth in areas which are, practically speaking, dry.

In a separate section at the end of this chapter this issue is revisited, and particularly the effect of $h_R \downarrow 0$ on the flow in the entire domain will be adressed.

For both the wet and the dry bed case analytical solutions of the dambreak 1D flow equations are available. These analytical solutions (recapitulated below in a separate section) form the basis for our quantitative analysis of the numerical accuracy of D-Flow FM. The approach followed in the present accuracy analysis is described below.

D-Flow FM simulations have been carried out for various systematic variations of both the grid size in the spatial discretisation, and the time step in the numerical time integration of the flow equations.

In the variations of the spatial discretisations four uniform rectangular grids are used, i.e. grids with cells of a constant size Δx along the longitudinal (x) direction of the computational

domain. The cells of these grids are respectively of size $\Delta x = 500, 100, 50$ and 25 m. With a longitudinal length $L = 60$ km of the domain this means that in the x-direction these grids consist of 120, 600, 1200, and 2400 computational cells. The four grids that are here considered therefore provide stepwise (and mutually 'comparable') refinements of the spatial resolution.

Compared to the conciser testcase on the dambreak over a dry bed, a main extension within the validation study is that also the effect of the time step Δt on the accuracy is investigated through variations of Δt in the numerical integration of the model equations. These variations of Δt are within a range of 0.01 to 3 seconds.

The effect of grid-size and time-step refinements on the accuracy of D-Flow FM simulations will be quantified through statistical error measures derived from the errors $\Delta h(x,t)$ and $\Delta u(x,t)$ in the model's prediction of the water depth $h(x,t)$ and flow velocity $u(x,t)$ at the cell centres[6]. The errors are defined by $\Delta h(x,t) := h_A(x,t) - h_C(x,t)$ and $\Delta u(x,t) := u_A(x,t) - u_C(x,t)$, with subscript C indicating a variable computed by D-Flow FM and with subscript A indicating the analytical solution. Apart from the water depth and the velocity, the errors in the computed flow discharge $\Delta(h(x,t) \cdot u(x,t))$ (i.e. a discharge per unit length in the transverse direction) will be analysed as well.

Three error measures L_1, L_2, and L_∞ will be considered for the quantification and comparison of the accuracy of the D-Flow FM solutions for the various variations in the grid size and the time step. These L_k ($k = 1, 2, \infty$) are actually vector norms applied to the set of model errors $\{\Delta h(x,t)\}_{(x,t)\in A}$ and $\{\Delta u(x,t)\}_{(x,t)\in A}$ on some sub-area A of the computational domain. They are defined by the following generic formula[7]:

$$L_k(\Delta\xi) = \sqrt[k]{\frac{1}{N_A}\sum\nolimits_{(x,t)\in A} |\Delta\xi(x,t)|^k} \tag{15.6}$$

In these equations, N_A denotes the number of cell centres $(x,t) \in A$. The ξ represents one of the three considered flow quantities, i.e. water depth h, flow velocity u, or the discharge $h \cdot u$.

The error measures will be computed for several sub-areas A of the (spatial and temporal) computational domain. The selection of the sub-areas is guided by the regions in the (x,t) domain where the analytical solution of the flow variables $h(x,t)$, $u(x,t)$ and $h(x,t)\cdot u(x,t)$ has specific properties. In one of the sections below this selection of adequate sub-areas is adressed in detail. Next, the dependency of the error norms on the grid size Δx and time step Δt will be evaluated for the various sub-areas, from which the order of accuracy of the numerical solutions can be obtained.

[6] Ideally, the error across the entire domain should be considered. It is, however, not clear how the discrete solution at cell centres should be extended in between the cell centres. Water depth h is assumed piecewise constant per grid cell in the time derivative of the continuity equation (a central approximation), but its value at the cell faces is obtained by (first-order) upwind. Velocity u at cell centers is obtained from averaging the values at the faces of the cells, thereby introducing averaging errors. Although in 1D u at cell faces could be considered when the grid is aligned with the flow (as is the case here), this is not possible in general. For this reason we decided not to exploit this possibility.

By only considering the value of h and u at cell centers, at most additional second-order errors are introduced due to the use of the 1-point Gauss integration rule (h) and 2-point central averaging (u). If necessary, this should be taken into account when interpreting the results.

Note that the discharges computed at the cell centers do *not* correspond with the discharges at the cell faces used in the D-Flow FM simulations. Because of the use of a first-order upwind h in the latter, the deviation will in general be first order. This is a discretization error, *not* an error in the interpretation of the results.

[7] Note that $L_1(\cdot)$ is the mean absolute error over A, while $L_2(\cdot)$ is the RMS (root mean square) of the model error. It is readily verified that $L_\infty(\cdot)$ corresponds to the maximum of the pointwise absolute errors $|\Delta\xi(x,t)|$ over A.

Analytical solutions for the one-dimensional dambreak models

The spatial domain is assumed to range from from $x = 0$ to $x = L$, and the location of the dambreak is at $x = x_0$. The initial system state (at $t = 0$, denoting the begin of the dambreak) is a stepwise constant water depth with $h(x) = h_L$ for $0 \leq x \leq x_0$ and $h(x) = h_R$ for $x_0 < x < L$. This longitudinal water depth profile is thus discontinuous at $x = x_0$.

In case of no bed friction, and no viscosity effects, an analytical solutiuon is available for the 1D dambreak flow equations (see Delestre et al., 2011). This holds for both a wet and a dry bed. First we present the general solution for the wet bed case. This solution will be used in the accuracy analysis described in the remaining chapters. Though wet in strict sense, in the computations a rather low water depth h_R will be adopted for the downstream bounday to obtain an "almost" dry bed. For comparison and later use the analytical solution for a 'true' dry case ($h_R = 0$) is given as well.

Analytical solution for wet bed, $h_R > 0$

From the (prescribed) water depths h_L and h_R, at respectively the upstream and downstream model boundaries, velocities c_L and c_R are defined by $c_L := \sqrt{g \cdot h_L}$ and $c_R := \sqrt{g \cdot h_R}$ (g is, as usual, the acceleration of gravity). In the formula's that follow below for the analytical solutions of the water depth $h(x,t)$ and $u(x,t)$ also a velocity c_M is present. This c_M depends on h_L and h_R, or equivalently on their corresponding velocities c_L and c_R. The depencency of c_M on c_L and c_R is governed by the following equation (Delestre and Lucas, 2011, but here with a correction of a typographical error):

$$c_M^6 \; - \; 9c_R^2 \cdot {c_M}^4 \; + \; 16c_L c_R^2 \cdot c_M^3 \; - \; \left(c_R^2 + 8c_L^2\right) c_R^2 \cdot c_M^2 \; + \; c_R^6 = 0 \tag{15.7}$$

As a result c_M is the root of a polynomial of degree six. It can be demonstrated that Min $(c_L,\ c_R) \leq c_M \leq$ Max $(c_L,\ c_R)$, but for general c_L and c_R the exact value of c_M cannot be found analytically. For this reason some numerical root finding method will have to be applied to derive c_M from c_L and c_R.

For a convenient formulation of the analytical solution three linear separators $x_A(t)$, $x_B(t)$, and $x_C(t)$ of the (x,t)-model domain are now defined according to:

$$\begin{cases} x_A(t) = x_0 \; - \; c_L \cdot t \\ x_B(t) = x_0 \; + \; (2 \cdot c_L \; - \; 3 \cdot c_M) \cdot t \\ x_C(t) = x_0 \; + \; 2 \cdot \dfrac{c_M^2 \left(c_L - c_M\right)}{c_M^2 - c_R^2} \cdot t \end{cases} \tag{15.8}$$

These separators induce four partitions of the model domain and on these sub-domains the analytical solution for the water depth reads:

$$h(x,t) = \begin{cases} h_L, & \text{if} \quad x < x_A(t) \\ \dfrac{4}{9g}\left(c_L \; - \; \dfrac{x \; - \; x_0}{2t}\right)^2 & \text{if} \quad x_A(t) \leq x \leq x_B(t) \\ h_M := \; \dfrac{c_M^2}{g} & \text{if} \quad x_B(t) \leq x \leq x_C(t) \\ h_R & \text{if} \quad x > x_C(t). \end{cases} \tag{15.9}$$

The analytical solution for the velocity along the four sub-domains is provided by:

$$u(x,t) = \begin{cases} 0 & \text{if} \quad x < x_A(t), \\ \frac{2}{3} \cdot \left(\frac{x \ - \ x_0}{t} \ + \ c_L \right) & \text{if} \quad x_A(t) \leq x \leq x_B(t), \\ 2 \cdot (c_L \ - \ c_M) & \text{if} \quad x_B(t) \leq x \leq x_C(t), \\ 0 & \text{if} \quad x > x_C(t). \end{cases} \tag{15.10}$$

For the derivation of these equations one is referred to Section 4.1.1 in Delestre et al. (2013).

Later in this chapter illustrations of these analytical solutions for $h(x,t)$, $u(x,t)$, and the discharge $h(x,t) \cdot u(x,t)$ will be shown, together with corresponding numerical solutions of D-Flow FM.

Analytical solution for a dry bed, $h_R = 0$

The solutions for the dry case can be obtained from those of the wet bed (as presented above) by letting $h_R \downarrow 0$ or equivalently $c_R \downarrow 0$. This can be done conveniently by using that

$$c_M = \sqrt[4]{8} \cdot \sqrt{c_L \cdot c_R} \tag{15.11}$$

for c_R close to zero. This limit behaviour of c_M can be derived from Equation (15.7). It is then also readily verified that for every time t it holds that

$$\lim_{c_M \downarrow 0} x_C(t) = \lim_{c_M \downarrow 0} x_B(t) \ = \ x_0 \ + \ 2c_L \cdot t \tag{15.12}$$

. The 'remaining' $x_A(t)$ and $x_B(t)$ then satisfy:

$$\begin{cases} x_A(t) = x_0 \ - \ c_L \cdot t \\ x_B(t) = x_0 \ + \ 2c_L \cdot t \end{cases} \tag{15.13}$$

These two separators induce three partitions of the model domain and now the analytical solution for the water depth reads:

$$h(x,t) = \begin{cases} h_L, & \text{if} \quad x < x_A(t) \\ \frac{4}{9g} \left(c_L \ - \ \frac{x \ - \ x_0}{2t} \right)^2 & \text{if} \quad x_A(t) \leq x \leq x_B(t) \\ 0 & \text{if} \quad x_B(t) \leq x \end{cases} \tag{15.14}$$

For the velocity the following analytical solution is now found along the three partitions of the domain:

$$u(x,t) = \begin{cases} 0 & \text{if} \quad x < x_A(t), \\ \frac{2}{3} \cdot \left(\frac{x \ - \ x_0}{t} \ + \ c_L \right) & \text{if} \quad x_A(t) \leq x \leq x_B(t), \\ 0 & \text{if} \quad x_B(t) \leq x \end{cases} \tag{15.15}$$

Model description

In all computations the following values were adopted for the flow and geometric parameters present in the analytical 1D dambreak model of Equation (15.9) and Equation (15.10).

- ◇ domain length: $L = 60,000$ m (and thus $0 \leq x \leq 60,000$ m).
- ◇ Water depth boundary conditions: $h_L = 2$ m (upstream) and $h_R = 0.0001$ m (downstream).
- ◇ Location of the dam break: $x_0 = 30,000$ m.
- ◇ Gravity acceleration: $g = 9.81$ m/s^2 .

For these parameters the following value is found for the velocity c_M implicitly defined by Equation (15.7): $c_M = 0.5840798$ m/s. This velocity corresponds to a water depth $h_M := {c_M}^2/g = 0.03477566$ m.

The same computational grids have been used as earlier in this document, although only the uniform rectangular grids 5, 1, 2, 3 are now taken into account[8]. These four grids are all built up of equally sized rectangular cells with $\Delta x = 500,\ 100,\ 50$ and 25 m respectively. They thus provide stepwise (and mutually 'comparable') refinements of the spatial resolution.

In the assessment of the sensitivity and accuracy for the time step D-Flow FM simulations have been carried out with eight variations of Δt: $\Delta t \in \{0.01, 0.05, 0.125, 0.25, 0.5, 1.0, 2.0, 3.0\}$ (all in seconds). In each case the simulation period $T = 3600$ sec.

The computational settings in the D-Flow FM simulations are as follows:

- ◇ No bed friction, no surface drag, no horizontal viscosity.
- ◇ Schematisation of advection: AdvecType= 3 (Perot q(uio-u)).
- ◇ Limiter type for cell center advection velocity: Limtypmom= 4 (Monotone Central).
- ◇ Time integration: TimeStepType= 2 (full implicit step reduce). $\theta = 0.55$
- ◇ Drying/flooding: chkadvd= 0.0 (parameter in handling advection over almost dry bed)
- ◇ Drying/flooding: epshu= 0.001 (threshold determining wet or dry bed)

In D-Flow FM the numerical setting parameter $chkadvd$ ('check advection depth') is relevant in simulations with frictionless flows over a fully or almost fully dry bed (as in the present case). For (almost) dry cells it may then happen that within a time step Δt the predicted water level is below the cell's depth. In that case the time step should be repeated with a smaller Δt. To prevent a substantial increase of computation time a correction procedure is applied, however, rather than a recomputation with a smaller time step. In this correction procedure explicitly discretised terms of the flow equations, and particularly the ones that are extracting water from nearby dry cells, are reduced by a factor. This factor is governed by (the ratio of) the current water depth and parameter $chkadvd$. In compuations with extremely low water depths $chkadvd$ should be set to zero. In all other situations its value is not or less critical and the default value of 0.1 (m) can well be used.

The parameter $epshu$ is a threshold in the D-Flow FM's modelling of drying and flooding. Cells where the water depth is greater than $epshu$ are treated as wet, while for cells with water depth less $epshu$ a flooding/drying procedure is activated.

In D-Flow FM a default $epshu$= 0.0001 is set. This default is adopted in the present simulations. For reasons of consistency the water level at downstream boundary was set equal to

[8]The uniform triangular grid 4, the non-uniform rectangular grid 6, and the rectangular grid with local refinements and irregular triangular connections 7 have been discarded. Results obtained on these grids are not suitable for consideration in the present systematic study on accuracy as a function of grid resolution and time step.

this drying/flooding threshold: $h_R = epshu = 0.0001$.

Selection of the sub-areas in the model error analysis

The spatial and temporal evolution of the flow variables in the present case of a dam break over an "almost" dry bed ($h_R = 0.0001$) are depicted in Figure 15.11, Figure 15.12 and Figure 15.13. The blue curves in these figures represent the $u(x,t)$, $h(x,t)$ and $h(x,t)$ ·

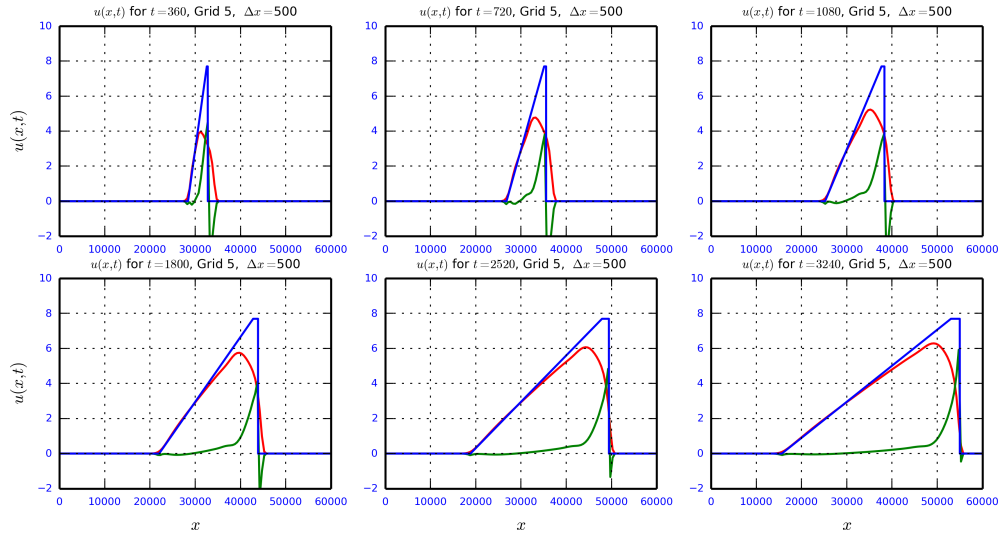


Figure 15.11: *Velocity $u(x,t)$ as function of the longitudinal coordinate x for 6 success ve time points t. The velocity $u_A(x,t)$ according to the analytical solution is plotted in blue, while the solution $u_C(x,t)$ according to D-Flow FM using Grid 5 (Δx = 500 m, Δt = 0.01 sec) is shown in red. The error in the computed velocity, $\Delta u(x,t)$, is depicted in green.*

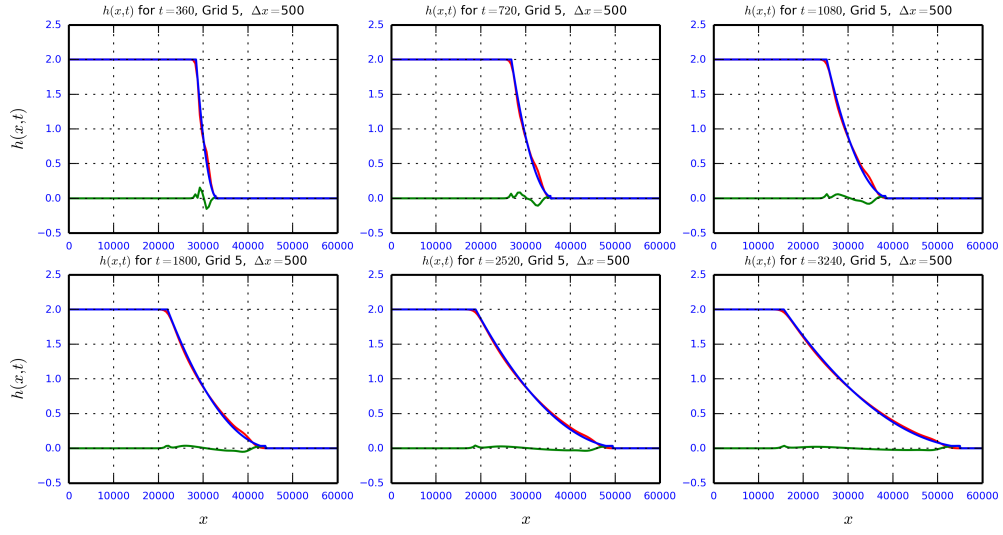


Figure 15.12: *Water depth $h(x,t)$ as function of the longitudinal coordinate x for 6 successive time points t. The water depth $h_A(x,t)$ according to the analytical solution is plotted in blue, while the solution $h_C(x,t)$ according to D-Flow FM using Grid 5 (Δx = 500 m, Δt = 0.01 sec) is shown in red. The error in the computed water depth, $\Delta h(x,t)$, is presented in green.*

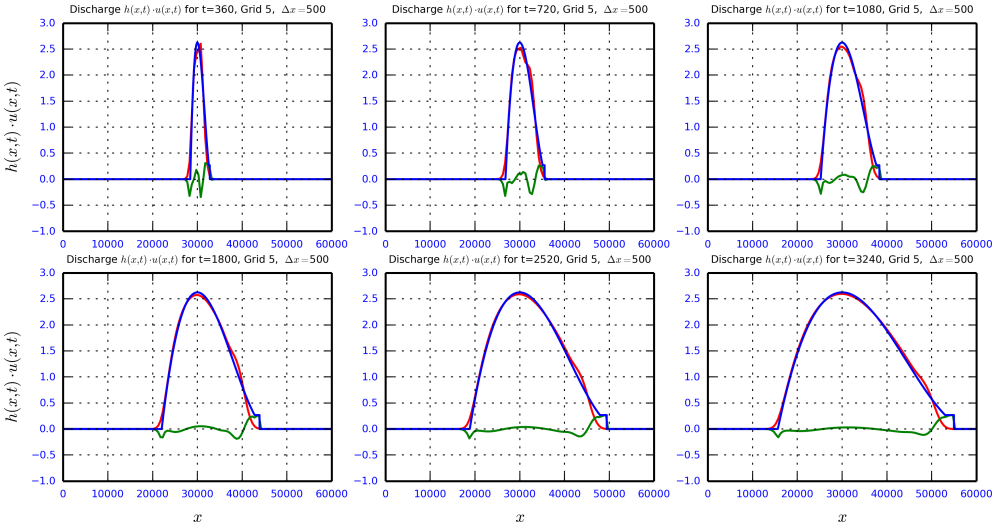


Figure 15.13: *Discharge $h(x,t)u(x,t)$ as function of the longitudinal coordinate x for 6 successive time points t. The discharge according to the analytical solution is plotted in blue, while the corresponding solution of D-Flow FM using Grid 5 (Δx= 500 m, Δt= 0.01 sec) is shown in red. The error in the computed discharge is presented in green.*

$u(x,t)$ according to the analytical solution of the flow equations (see Equation (15.9) and Equation (15.10)).

The red curves in the panels of the three figures show these flow variables as computed by D-Flow FM, using the coarsest grid 5 with $\Delta x = 500$ m in combination with a very small time step of $\Delta t = 0.01$ sec to ensure negligibly small time integration errors. The difference between the red curves and the blue curves in the figures are therefore uniquely due to the spatial discretisation of D-Flow FM.

In the plotting procedure the computed solutions (red) were plotted before the analytical solutions (blue). As a result, the red curves are masked by the blue curves where and when the computed solution agrees well with the analytical solution. Loosely stated, it can thus be noted that the *less* is seen of the red curves, the *smaller* the errors, the *better* the quality of the numerical simulation. For a more comprehensive inspection of the quality of the numerically computed solution, the errors $\Delta u(x,t)$, $\Delta h(x,t)$ and $\Delta(h(x,t)\cdot u(x,t))$ have also been plotted separately in Figure 15.11, Figure 15.12 and Figure 15.13. See the curves presented in green.

From these figures it is readily recognised that with the present coarse grid ($\Delta x = 500$ m) the analytical water depth is reasonably reproduced by D-Flow FM. For the discharges the error is more substantial, particular in the vicinity of the separation lines $x_A(t), x_B(t)$ and $x_C(t)$.

For the velocities, however, the error is even quite large. The shape of the analytical velocity profile is rather poorly reproduced. In particular the peak values are severely under estimated. At the same time a remarkable phenomenon is also observed within the time propagation of the downstream velocity front (i.e. near $x_B(t)$ and $x_C(t)$). In the first half hour of the simulation the (position of the) computed front is clearly beyond the one of the analytical solution. However, as time evolves the downstream propagation of the computed front gradually delays and in the end, at $t = 3240$ sec, the location of both fronts more or less coincide. We do not have a pausible explanation for this behaviour.

To verify the effect of grid refinement on the error in the model predictions, the same plots, but now for the finest Grid 3 with $\Delta x = 25$ m, are presented in Figure 15.14, Figure 15.15 and Figure 15.16. The time step is again $\Delta t = 0.01$ sec and small enough for the time integration errors to be negligible.

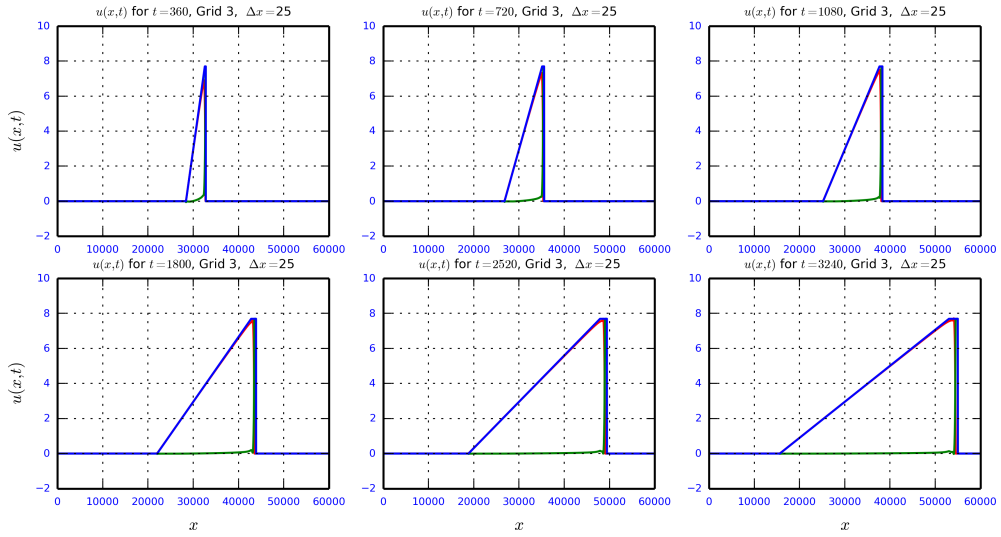


Figure 15.14: *Velocity $u(x,t)$ as function of the longitudinal coordinate x for 6 successive time points t. The velocity $u_A(x,t)$ according to the analytical solution is plotted in blue, while the solution $u_C(x,t)$ according to D-Flow FM using Grid 3 (Δx= 25 m, Δt= 0.01 sec) is shown in red. The error in the computed velocity, $\Delta u(x,t)$, is depicted in green.*

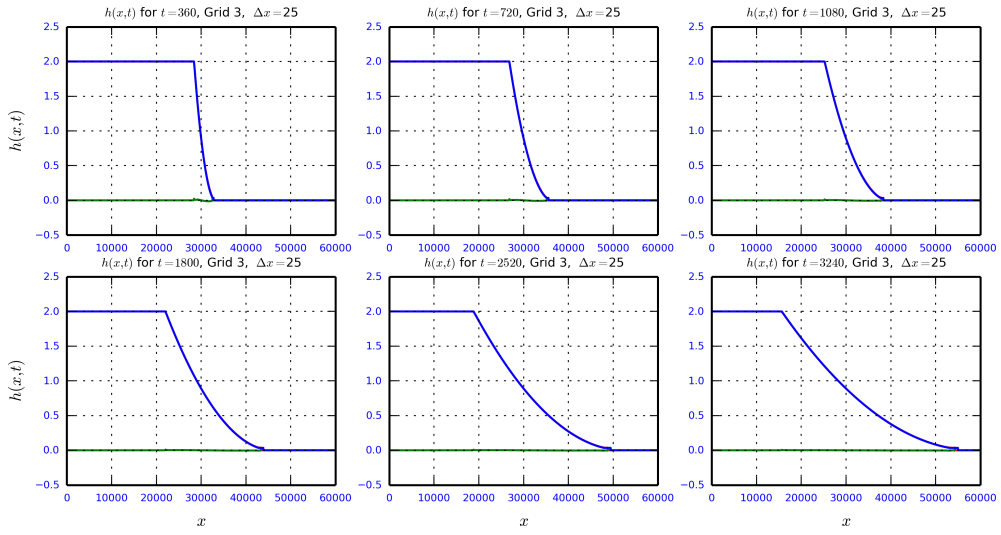


Figure 15.15: *Water depth $h(x,t)$ as function of the longitudinal coordinate x for 6 successive time points t. The water depth $h_A(x,t)$ according to the analytical solution is plotted in blue, while the solution $h_C(x,t)$ according to D-Flow FM using Grid 3 (Δx= 25 m, Δt= 0.01 sec) is shown in red. The error in the computed water depth, $\Delta h(x,t)$, is presented in green.*

Figure 15.14, Figure 15.15 and Figure 15.16 clearly reveal that the grid refinement from Δx= 500 m to Δx= 25 m has significantly improved the quality of the simulation. Most eminently this is observed for the velocity profiles. Nevertheless a slight phase error in the prediction of

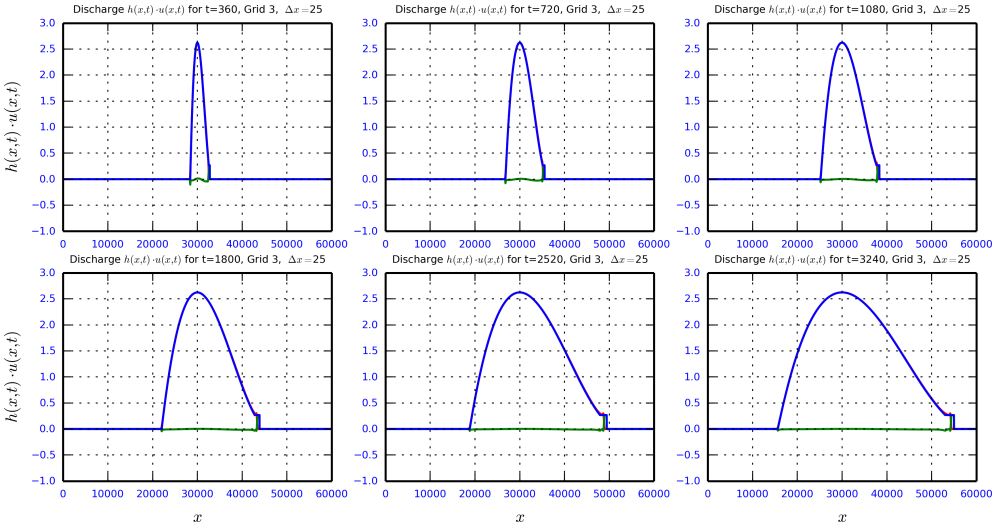


Figure 15.16: *Discharge $h(x,t)u(x,t)$ as function of the longitudinal coordinate x for 6 successive time points t. The discharge according to the analytical solution is plotted in blue, while the corresponding solution of D-Flow FM using Grid 3 (Δx= 25 m, Δt= 0.01 sec) is shown in red. The error in the computed discharge is presented in green.*

the downstream front (in the vicinity of $x = x_C(t)$) tends to persist. This is actually even better noticable in the figure with the temporal evolution of the discharge profiles, see Figure 15.16.

From the formulas and the figures of the analytical solution four sub-areas of different flow regimes can be recognised in the spatio-temporal domain $\Omega := \{(x,t)|t \geq 0\}$. These four sub-areas are bounded by the three concurrent straight lines $\{(x_A(t),t)\}$, $\{(x_B(t),t)\}$ and $\{(x_C(t),t)\}$. See Equation (15.8) for the expressions of these $x_A(t)$, $x_B(t)$ and $x_C(t)$. In the *lower* panel of Figure 15.17 the location of these lines within Ω is shown (plotted in red), together with the four sub-areas $\{\Omega_1, \Omega_2, \Omega_3, \Omega_4\}$ that they determine. Through the graphs of the water depth (red), velocity (blue) and discharge (green) depicted in the *upper* panel of Figure 15.17 (with h, u, and $h \cdot u$ as function of x, for fixed time $t = 3240$ sec) the relation of the sub-areas Ω_i with the different flow regimes can be conveniently verified.

In the sub-areas Ω_1 and Ω_4 the solution of the flow equations is constant and equal to the initially imposed conditions. In a validation of the model's quality these sub-areas are thus of minor or no interest.

In sub-area Ω_3 the water depth and velocity are also constant but with values depending on the (initial) water depth in the left (Ω_1) and right (Ω_4) part of the domain. The accuracy with which these constants are reproduced by the numerical model is included in the present validation study. It must yet be taken into account that the (x,t) area covered by Ω_3 is rather small since $x_B(t)$ is close to $x_C(t)$ for each time t. In its turn this is due to the 'very low' water depth prescribed at the downsteam boundary. Particularly for the coarse spatial and temporal discretisations the number of gridpoints within sub-area Ω_3 will then be too small for a solid estimate of the error norms. In the evalution of these norms Ω_3 will therefore be extended with parts of the neighbouring other sub-areas.

In sub-area Ω_2 the water depth and velocity are not constant. Along the x-direction the water depth follows a monotonically decreasing parobolic profile, while the velocity increases linearly from 0 at $x = x_A(t)$ to its maximum at $x = x_B(t)$. The various panels in the figures Figure 15.11, Figure 15.12 and Figure 15.13 show how these profiles of the flow variables

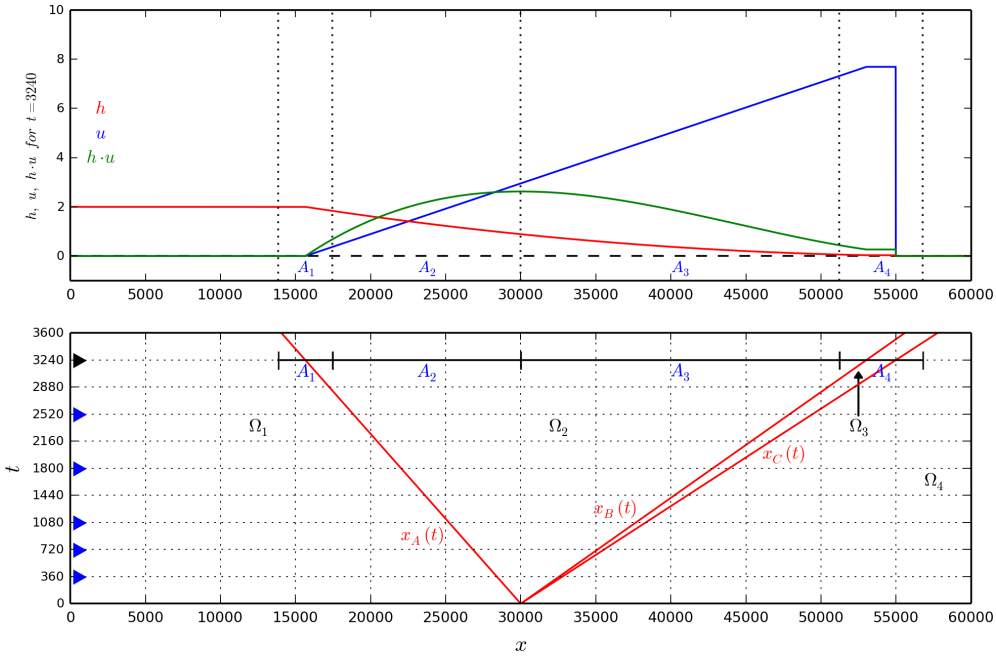


Figure 15.17: *Lower panel: Partition $\{\Omega_1, \Omega_2, \Omega_3, \Omega_4\}$ of the model's spatio temporal domain by the three lines $\{(x_A(t), t)\}$, $\{(x_B(t), t)\}$ and $\{(x_C(t), t)\}$, and the sub-areas A_i where the error norms are evaluated. The symbols ▶ mark the times t for which in preceding figures the flow variables u, h, and $h \cdot u$ have been shown as function of the spatial variable x. In the upper panel these flow variables are once more illustrated for $t = 3240$ sec. In this way the different characteristics of the flow for the different sub-areas can be conveniently recognised. For this $t = 3240$ sec the various norms for the D-Flow FM errors have been evaluated.*

evolve in time. This temporal evolution consists of a gradual spatial stretching of the profiles in both the downstream and upstream direction. Within this stretching the shape of the profiles remains the same (self-similar solution). Because of the non-trivial variation, the solution in this sub-area Ω_2 is highly relevant in the present D-Flow FM validation.

From the figures of the analytical solutions presented above, it is recognised that the interface between the sub-areas Ω_i (formed by the lines $\{(x_A(t), t)\}$, $\{(x_B(t), t)\}$ and $\{(x_C(t), t)\}$) are of special interest. At these interfaces the flow variables are non-smooth (at $\{(x_A(t), t)\}$ and $\{(x_B(t), t)\}$) or even discontinuous (at $\{(x_C(t), t)\}$). In an investigation of the accuracy of numerical flow models such as D-Flow FM, the capability to deal with such non-smoothness is an important issue. For the present case this importance is revealed by the errors plotted in Figure 15.11 and Figure 15.12. It can be observed that the errors reach local maxima at the interfaces between the sub-areas and that they are spread over quite a few grid cells when $\Delta x = 500$m. The same behaviour is present in Figure 15.14 and Figure 15.15, but not so clearly visible because of the use of the much smaller grid size $\Delta x = 25$m.

On the basis of these considerations the assessment of the accuracy of the D-Flow FM simulations is carried out for the two sub-areas Ω_2 and Ω_3, and for small neighbourhoods of their boundaries x_A, x_B and x_C. As argued before the size of Ω_3 is too small to cover a sufficiently number of gridpoints in the error analysis for the coarsest grids. Therefore Ω_3 is extended with a part of the its boundary with Ω_2 on the left, and with Ω_4 on the right. On the other hand, for the sub-area Ω_2 a *partition* is made, and this area is split up in two sub-areas consisting of its part to the left and its part to the right of the original dam-break position x_0. The reason for this splitting of Ω_4 is the observation (cf. Figure 15.11 and Figure 15.12) that in sub-area

Ω_2 the errors in the computed water depth and the velocity turn out to be significant larger for $x > x_0$ than for $x < x_0$.

A good impression of the simulation accuracy of D-Flow FM is obtained by computing and presenting the error norms of $\Delta u(x,t)$ and $\Delta h(x,t)$ for a time level toward the end of the simulation. This will exclude spurious effects due to any excessive errors in the start-up phase of the computation that may have been caused by the discontinuous initial condition. We have chosen the time level $t = 3240$ sec, which in the previous figures corresponds with the solutions shown in the lower-right panels.

Altogether this gives the following four sub-areas A_i at $t = 3240$ sec for which results of the accuracy assessment will be presented in the next section:

1 Area A_1 containing the grid points x_n for which $13849\text{m} < x_n < 17449\text{m}$;
2 Area A_2 containing the grid points x_n for which $17449\text{m} < x_n < 30000\text{m}$;
3 Area A_3 containing the grid points x_n for which $30000\text{m} < x_n < 51226\text{m}$;
4 Area A_4 containing the grid points x_n for which $51226\text{m} < x_n < 56790\text{m}$;

The location of these four sub-areas A_i within the computational domain Ω, and the characteristics of the flow variables on these sub-areas is shown in Figure 15.17.

Finally it is mentioned that the analytical solution has been used to construct plots of the relevant components of the momentum equation: $\frac{\partial(h\cdot u)}{\partial t}$ (storage term), $\frac{\partial(h\cdot u^2)}{\partial x}$ (convection term), and $gh \cdot \frac{\partial h}{\partial x}$ (pressure term). All other components, related to bed friction, horizontal viscosity, Coriolis, etc. are zero. The plots of the three relevant components (not presented here) were used to verify the relative weight that these components contribute to the flow propagation. With the present setting of the model parameters (i.e. the water depth boundary conditions h_L and h_R) it was oberved that in magnitude these components are reasonably in balance. As a result they contribute significantly, and more or less equally, to the results and conclusions of this validation study. These results are summarised in the next section.

Results

For each of the four uniform grids (with the grid sizes $\Delta x = 500$, 100, 50 and 25 m) D-Flow FM simulations were carried out for the eight different time steps $\Delta t = 0.01$, 0.05, 0.125, 0.25, 0.5, 1.0, 2.0 and 3.0 sec. It must be noted that larger time steps than the present maximum $\Delta t = 3.0$ sec did not provide meaningful results. For the finest grids instabilities were found in the D-Flow FM solutions in the form of (non-physical) large oscillating variations in both the velocity and the water level. Actually, such a behaviour was also to some extend already observed in the simulations with $\Delta t = 3.0$ sec.

For all these simulations, the three error norms $L_1(\cdot)$, $L_2(\cdot)$ and $L_\infty(\cdot)$ (see Equation (15.6)) were evaluated. This was done for all the four sub-areas A_i that were selected in the preceding paragraph.

Results of time step variations

To begin with, the effect of the time-step size has been verified using the spatial grid with the finest resolution ($\Delta x = 25$ m). For this finest grid (and any other grid if larger time steps would have been considered) we may expect that with increasing Δt the errors in the D-Flow FM simulations gradually become mainly, or even fully, determined by the errors in the applied time integration method. From the dependency of the error norms on Δt the time accuracy of D-Flow FM can be evaluated and quantified.

In the (graphical) inspection of the effect of the time step on the error norms a non-regular behaviour was observed for the largest Δt values (larger than 1 sec). Likely this is due to (small) numerical instabilities that may occur when these time steps are combined with the present smallest grid size of 25 m.

It must yet be noted that within the error norms such irregularities were observed 'only' for the down strean situated sub-areas A_3 and A_4. For A_3 this would not have been expected beforehand given the smooth dependency in this area of the (analytical) flow variables on the spatial and temporal coordinates. Probably the solutions in this sub-area are affected through by those in sub-area A_4 through back propagation.

To maintain the full range of Δt variations the time step analysis was repeated for the simulations on the finest grid but one (i.e. Grid 2 with $\Delta x = 50$ m). The result is graphically shown in Figure 15.18. To limit the total number of figures, and also because of simular characteristics of the other two error norms, only the error behaviour in the L_1 norm is shown in Figure 15.18. For all variables we see roughly the same error behaviour. Such a similar behaviour is not found for the different sub-areas, however. For area A_1 (around the separator $x_A(t)$, in the upstream part of the domain) the expected behaviour is produced in the sense that the L_1-errors monotonically decrease when reducing the time step Δt. This also tends to hold for $L_1(\Delta h)$ on sub-area A_3. For all other $L_1(\cdot; A_i)$, however, a response opposite to as expected is found: an increase of the errors for decreasing Δt. The authors have no plausible explanation at hand for this quite remarkable behaviour.

It must yet be noted that despite this 'discrepancy' the errors do not grow significantly when $\Delta t \downarrow 0$. In fact, (except for the velocity on area A_4) they then converge to a reasonably small value. This limiting value tends to be found for all $\Delta t \leq 0.25$ sec and this suggests that the limiting value fully represents the error of the spatial discretisation in D-Flow FM's numerical scheme. Due to a limited sensivity for a wide range of the applied time steps, we may also conclude that (in this case) the errors originating for the spatial discretisation by far dominate those of the temporal discretisation.

As a matter of the here found non-uniform behaviour of the error norms no unequivocal conclusion can be formulated about the numerical scheme's order of accuracy with respect to the time integration. Formally a first order would be expected (error proportional to Δt) because of first-order explicit discretisation of the convection term $u \cdot \frac{\partial u}{\partial x}$, and first-order implicit in the pressure term $g \cdot \frac{\partial h}{\partial x}$ of the momentum equation. Figure 15.18 suggests that even for sub-area A_1 this first order accuracy is not achieved.

We recall that the error behaviour is sensitive to the way that the error has been measured, cf. Footnote 11. This may be the reason why the error in sub-area A_6, where the hydraulic jump is, seems to be higher order in Δt (see again Figure 15.18). Unless special shock-fitting techniques are used, steady discontinuities like hydraulic jumps by definition limit the accuracy of *any* numerical scheme to $O(\Delta x)$ in the L_1 norm, to $O(\sqrt{\Delta x})$ in the L_2 norm, and to $O(1)$ in the L_∞ norm. The same applies in time for moving discontinuities, which is the case that we are dealing with here. In other words, the seemingly higher-order behaviour of the D-Flow FM time integration scheme in sub-area A_6 must be a spurious effect. It is probably due to the fact that the numerical solution and the exact solution are not compared everywhere in the domain[9], but only at a limited number of points.

The error reaches a constant value for $\Delta t < 0.5$sec, indicating that for time steps that are that small the time integration error becomes (negligibly) small compared to the space

[9] That would require the numerical solution to be defined as a (discrete) *function* in space, and not as values in only a finite number of (grid) points as we do here.

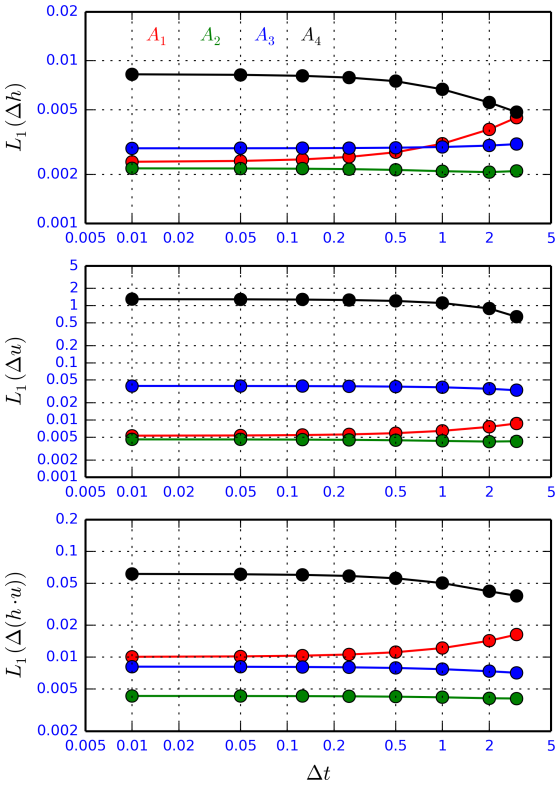


Figure 15.18: *Dependency of the error norm $L_1(\cdot)$ on Δt, for the water depth, velocity and discharge on one hand, and on the other hand on the six sub-areas A_i that were selected in the validation of the accuracy of the numerical scheme. These $L_1(\cdot)$ are as obtained with the presently finest grid size but one, with $\Delta x = 50$ m.*

discretisation error.

In practical applications a suitable time step would be the one that ensures a balance between time integration errors and space discretisation errors. The small increase in accuracy of using a smaller time step would not outweigh the larger additional computational costs of a smaller time step, while with a larger time step (and hence a dominating time integration error) one would not fully exploit the accuracy provided by the spatial grid resolution. For the present case, Figure 15.18 shows that the optimal time step is around 1 to 2 sec. Since the grid size in this computation is 25m and the maximum flow velocity is 4m/sec, this means that the optimal Courant number $|u|\Delta t/\Delta x$ is about 0.2 to 0.3. It has been verified that this also applies to the computation with a grid size of 100m, so this conclusion is grid-resolution invariant, at least for this application.

A Courant number of 0.2 to 0.3 is significantly lower than the stability limit $|u|\Delta t/\Delta x \leq 1$. The conclusion is therefore that for this application (where we have used a uniform rectangular grid) the quality of the D-Flow FM space discretisation is better than that of its time integration. A consequency of this is that in highly unsteady applications (such as the one considered here), the time step may have to be taken smaller than the one that follows from the stability condition in order to ensure optimal performance of D-Flow FM.

The error behaviour for sub-area A_5 shown in Figure 15.18 is remarkable. It continues to decrease for time steps smaller than 0.5sec. This applies in particular to the error in the water depth. Sub-area A_5 is the only part of the domain where the solution is both non-

trivial and constant, and where the space discretisation therefore becomes exact. As this sub-area is free of space discretisation errors, any error for very small time step must be due to the solution predicted in the sub-areas adjacent to A_5. An explanation for the remarkable error behaviour in A_5 may lie in the conservation properties of the space discretisation. It seems that this ensures that, regardless of any other errors, the water depth and flow velocity in sub-area A_5 are exact. However, the applied time integration scheme is *not* momentum conserving. This would explain why in sub-area A_5 the error is mainly determined by the applied time integration.

Results of grid size variations The accuracy of D-Flow FM with respect to the spatial discretisation has been verified in a similar way as described above for the time step. To ensure that errors due to the time integration can safely be ignored, only results from the simulations with the extremely small time step ($\Delta t = 0.01$sec) have been considered. The error behaviour in the L_1 and the L_∞ norm is shown in Figure 15.19. The L_2 norm showed a similar behaviour as L_1 and is therefore not further considered.

For all variables (h, u, and $h \cdot u$) and in all sub-areas A_i we see in Figure 15.19 roughly the same error behaviour of L_1 as a function of the spatial grid resolution. This also holds for the L_∞ norm. A main difference that is found when comparing these two norms is that for sub-area A_4 the L_∞ norm is virtually constant for all three flow variables, while the corresponding L_1 norm monotonically decreases with increasing spatial resolution.

From the slopes in the $\log L(\Delta x)$ vs. $\log(\Delta x)$ plots of Figure 15.19 is can be seen that in some cases the order of accuracy of the numerical scheme is close to one, but in most case noticeably less. On sub-area A_4 the order is even zero, when using the L_∞ norm.

An accuracy order close to, but not larger than one could be expected for sub-area A_4 because of discontinuous and non-smooth dependencies of the flow variables on the spatial coordinate (near $x_B(t)$ and $x_C(t)$). For the other sub-areas, with particularly A_2 and A_3 where the (analytical) solution is infinitely smooth, a higher-order of accuracy would have been expected.

The conclusion reads that somewhere in the D-Flow FM scheme a first-order approximation is applied (first-order upwind of the water depth in the determination of the mass flux through cell faces), which *for this application* has a dominant effect on the spatial accuracy[10]. Since a full description of the numerical scheme applied in D-Flow FM is not available, it is not possible to interprete the obtained validation results any further.

For a proper quantitative assessment of the order of accuracy, the effective order of accuracy as observed in the computations has been determined by means of linear regression analyses (of the form $\log(L_k(\Delta x)) \sim a \cdot \log(\Delta x) + b$) of the logarithm of the errors as a function of the logarithm of the grid size (i.e. the functions as shown in Figure 15.19). The error on the coarsest grid has been discarded in this analysis. On this coarsest grid with $\Delta x = 500$ m each sub-area only contains half a dozen or so grid cells, which is not enough for a reliable error analysis. This has been confirmed by looking at the correlation coefficients. They are close to 1 with the error on the coarsest grid discarded, but when the error on this grid is included the regression analyses give less strong correlations.

[10]Note that the way the (total) water depth is approximated at cell faces may have less impact on the overall accuracy in applications with little variation in the water level, such as tidal-flow and river-flow simulations. With the bathymetry defined at grid points, the depth with respect to the model datum is approximated second-order accurate at cell faces. With little variation in the water level, a first-order approximation error of taking the water level at cell faces upwind may have only a minor effect on the computation of the (total) water depth at cell faces. Additional validation studies are required to investigate this.

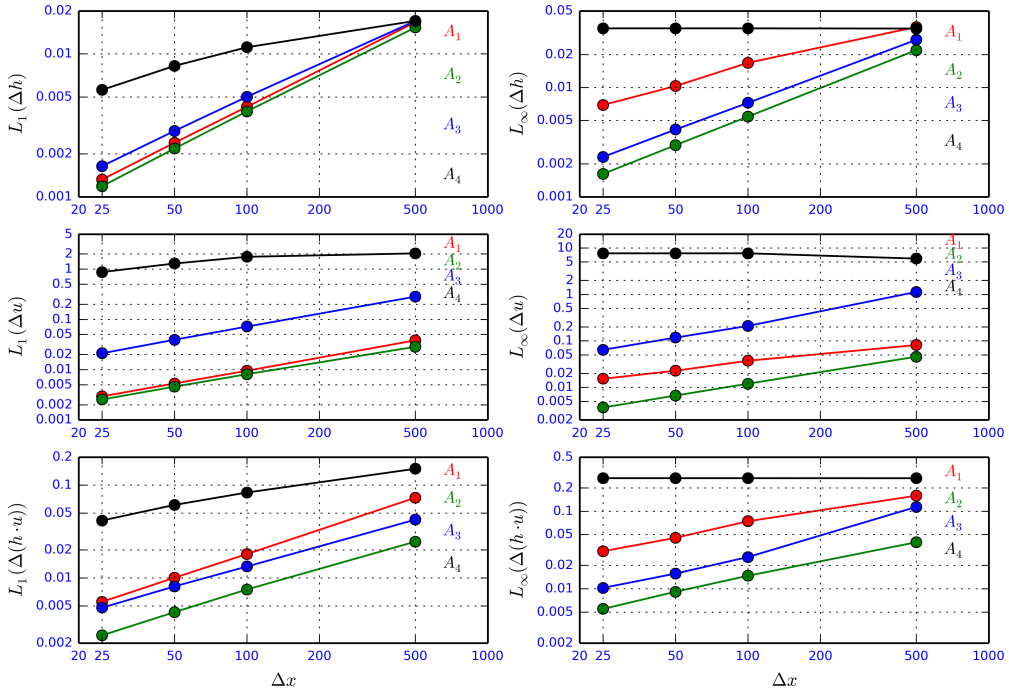


Figure 15.19: *Dependency of the error norms $L_1(\cdot)$ and $L_\infty(\cdot)$ on Δx, for the water depth, velocity and discharge on one hand, and on the other hand on the four sub-areas A_i that were selected in the validation of the accuracy of the numerical scheme. The $L_1(\cdot)$ and $L_\infty(\cdot)$ are as obtained with the presently smallest time step $\Delta t = 0.01$ sec.*

The estimate for the slope parameter a in the regression $\log(L_k(\Delta x)) \sim a \cdot \log(\Delta x) + b$ provides the order of accuracy of the numerical scheme. The so obtained order-of-accuracy results are graphically shown in Figure 15.20. This is done for all L_k error measures and also all the four sub-areas A_i that have been considered.

In the sub-areas A_2 and A_3, where the solution of the flow equations is smooth, Figure 15.20 shows an error behaviour that is less than, but still reasonably close to, order 1. In sub-area A_1, where the solution is not smooth but still continuous, the order of accuracy measured in the L_∞ norm decreases to about 0.6.

In sub-area A_4, where due to the hydraulic jump the solution is neither smooth nor continuous, the error behaviour is of order $1/2$ for the L_1 norm, $1/4$ for the L_2 norm, and even zero for the L_∞ norm. This does not agree with the theory how discontinuities limit the accuracy of virtually *any* numerical scheme. In fact, as already explained above, orders 1, $1/2$, and $1/4$ would have been expected. The reason for this discrepancy is not known.

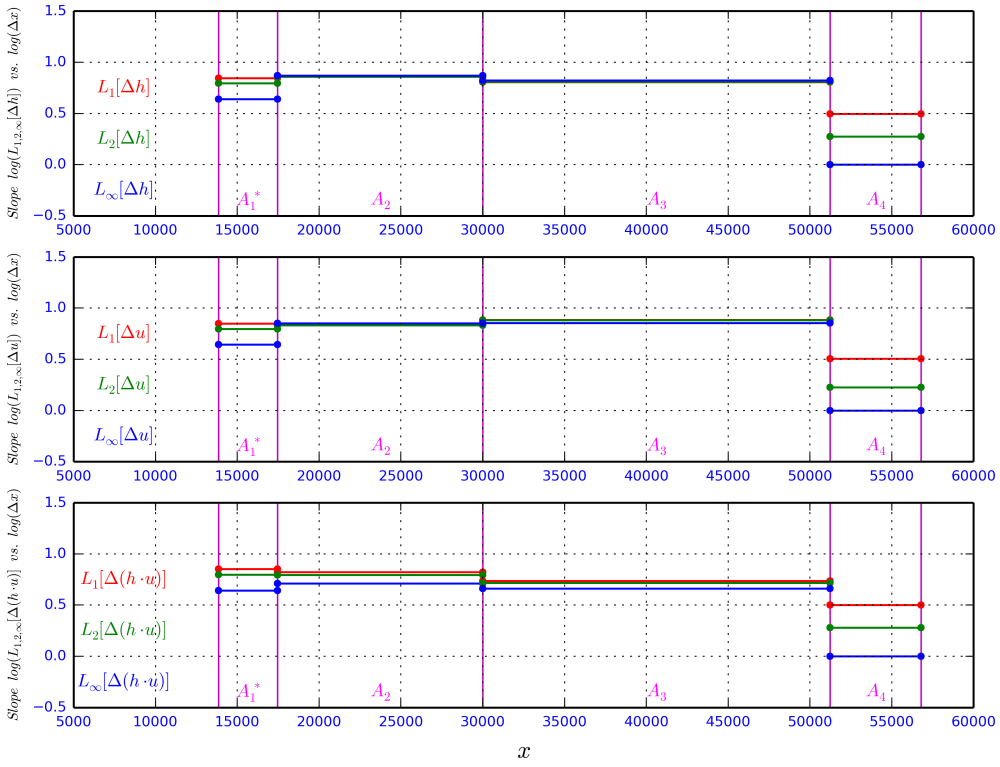


Figure 15.20: *Order of accuracies, $O(\Delta x)$, found for the various error norms $L_k(\Delta x)$. Through the three sub-panels the dependency of $O(\Delta x)$ on the flow quantities h, u, and $h \cdot u$ is presented. Within the sub-panels the dependency of $O(\Delta x)$ on sub-areas A_i in the computational domain is shown. These orders of accuracy are as obtained with the presently smallest time step $\Delta t = 0.01$ sec.*

Discussion: almost dry bed compared to fully dry bed

In all preceding simulations of D-Flow FM a 1D dambreak flow over an "almost" dry bed was computed and compared to the corresponding analytical solutions. This is more or less under the assumption that the applied downstream water depth boundary condition of $h_R = 0.0001$ (in combination with the upstream boundary condition of $h_L = 2$) provides a close approximation of a dry bed in strict sense. The latter would require that $h_R = 0$ (and h_L as before). In this paragraph the "assumption" that $h_R = 0.0001$ indeed represents an almost dry bed is verified in some more detail. In this analysis we restict to the velocities since this quantity was found to depend most sensitively on the setting of h_R. Moreover, and similar as before, the longitudinal velocity profile for $t = 3240$ sec is considered in particular.

The analytical solutions presented in Equation (15.10) and Equation (15.15) can conveniently be used to show the dependency of the velocity $u(x, t = 3240)$ sec on the water depth h_R at the downstream ("dry") boundary. In Figure 15.21 these velocity profiles are shown for the following six variations of h_R: 0.1, 0.01, 0.001, 10^{-4}, 10^{-6}, and 10^{-9}. The curve plotted in cyan represents the profile for the in this validation study used h_R= 0.0001. From the figure an (unexpected) high sensitivity can be observed for h_R. This sensivity is in twofold: both with respect to the maximum velocity along the domain, but also with respect to the propagation speed of the upstream front (and thus also its longitudinal position) along the channel. While practically speaking an h_R of 0.1 mm could be considered as virtually dry, the figure shows that in "theoretical" sense this is not really the case. In fact, for $h_R = 10^{-9}$ (i.e. 1 μ·mm) the peak velocity is still considerably larger than for $h_R = 10^{-4}$, and so is the front's propagation speed.

This raises the question whether for such considerably smaller values of h_R the analytical solutions can still reasonably be reproduced by D-Flow FM. In analogy to Figure 15.14 analytical and D-Flow FM computed velocity profiles are mutually compared in Figure 15.22 for the case that h_R was set to 10^{-9} m (and with $\Delta t = 0.01$ sec , $\Delta x = 25$ m, and $epshu$ set equal to h_R). From Figure 15.22 it is observed that now the velocity profile is not reproduced with very high accuarcy. Deficiencies concentrate exactly on the two aspects for which on the basis of the analytical solutions the highest sensitivity was found for $h_R \downarrow 0$: the peak value (under estimated by the simulation) and the propagation speed of the downstream front (also under estimated). Such errors were also found in a D-Flow FM computation with $h_R = 10^{-6}$.

Probably this reduction of the computation's quality for very small h_R originates from the drying and flooding mechanism in D-Flow FM. In absence of documentation this could not be verified in sufficient detail, and remains an important issue for clarification and further validation.

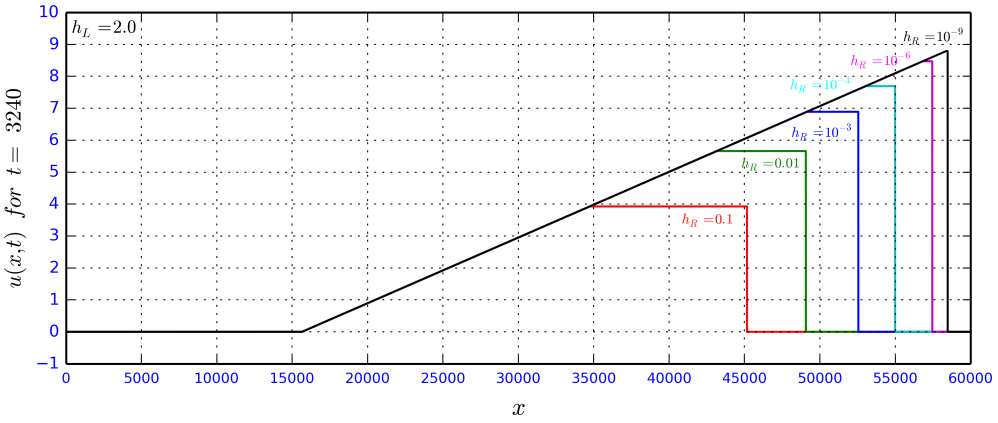


Figure 15.21: *Dependency of the velocity $u(x, t = 3240)$ sec on the downstream water depth boundary condition h_R*

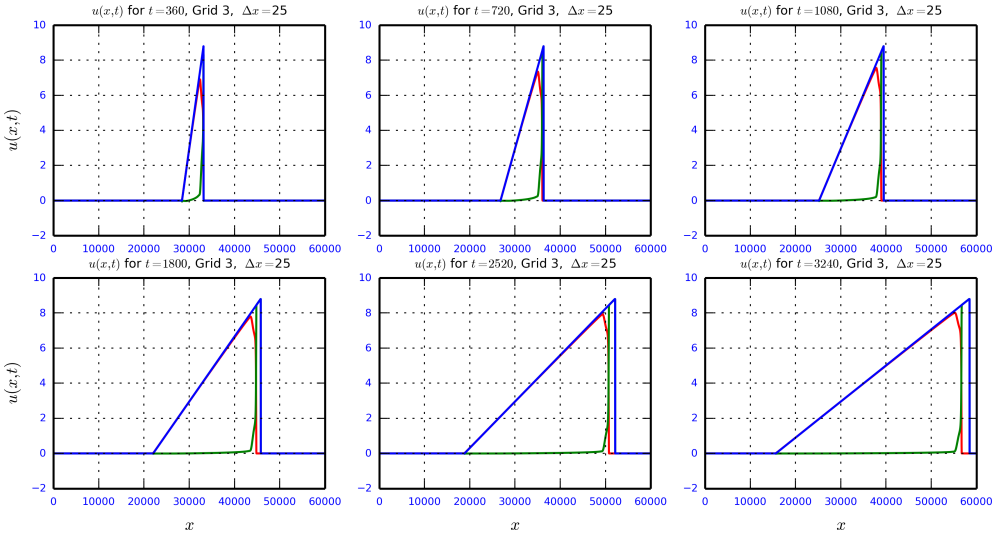


Figure 15.22: *Velocity $u(x, t)$ as function of the longitudinal coordinate x for 6 successive time points t. The velocity $u_A(x, t)$ according to the analytical solution is plotted in blue, while the solution $u_C(x, t)$ according to D-Flow FM using Grid 3 (Δx= 25 m, Δt= 0.01 sec, $h_R = 10^{-9}$ m) is shown in red. The error in the computed velocity, $\Delta u(x, t)$, is depicted in green.*

Conclusion

The main conclusions of the present D-Flow FM validation study can be summarised as follows.

- ◇ The aim of the present study was to validate D-Flow FM for the modelling of a 1D dambreak flow over a dry bed. In the simulations, however, this situation was approximated by an "almost dry bed". In this approximation a small water depth h_R was prescribed at the model's downstream boundary: $h_R = 0.0001$ m in combination with an $h_L = 2$ m at the upstream boundary. A detailed analysis of the analytical solutions revealed that the water depth and velocities for this (practically speaking) small value of h_R are still significantly different from those for a dry bed in strict sense, i.e. when h_R is set to exactly 0 m. D-Flow FM simulations with an h_R significantly smaller than 0.0001 (together with corresponding reductions of the D-Flow FM's drying/flooding control parameter epshu) resulted in considerably less accurate predictions of the analytical solutions. This degradation of the quality is presumably related to the drying and flooding procedure in D-Flow FM. In this study it was then decided to limit the downstream boundary h_R to D-Flow FM's default value of the flooding/drying threshold parameter epshu.
- ◇ From the applied time integration scheme, a first order error behaviour would be expected with regard to the time step Δt. For the present case, simulating a 1D dambreak flow over an almost dry bed, this behaviour is not well (re)produced. In fact, in downstream sub-areas of the computational domain (and in particular the sub-area with the heavy discontinuity in the velocity) errors were found to *increase* (slightly, but monotonically) when *reducing* the time step. In the most upstream sub-area the expected response to time step reduction was found, but with an accuracy order noticeably less than 1.
- ◇ In the present flow simulations the errors in the modelling are virtually completely due to the spatial discretisation.
- ◇ In all norms the accuracy with regard to the spatial discretisation is less than, but reasonably close to, first order. This is under the condition of a smooth (or at least continuous) dependency of the flow variables on the spatial and temporal coordinates. In sub-areas with discontinuities the order of accuracy depends on the applied error norm. For the L_1 norm this order is $1/2$ where an higher order 1 would be expected.
- ◇ Practically speaking D-Flow FM is well capable to reproduce the analytical solution of the *almost* dry bed dambreak flow equations: in absolute value the model errors are quite small. This is at the cost, however, of a considerably fine grid, and through the Courant number, combined with a suitably small time step.

It must be mentioned that these findings can partly (or even to a large extent) be induced by the present flow conditions, consisting of a non-steady, highly dynamic evolution of the flow, with non-smooth and discontinuous spatio/temporal variations (shock waves, jumps in the water depth and velocity profiles), and absence of friction and/or viscosity effects. At the same time it must be emphasized that the present conclusions are derived for, and may be limited to, uniform rectangular grids only.

Version

This test has been carried out with version dflow-fm-x64-1.1.111.34732M.

15.3 Wave run-up and run-down on a beach with constant slope

Purpose

Many practical applications of hydrodynamic models have to deal with the simulation of flows in seas or estuaries with shallow shore boundaries. Within the propagation of these flows the run-up and run-down of (long) waves and associated wetting and drying of shallow regions must then carefully be taken into account. This drying and flooding, involving moving boundaries (dry/wet interfaces), is difficult to resolve with a high spatial and temporal resolution, and with high accuracy as well. (Quasi) analytical solutions of the non-linear shallow water equations are available for special cases only. These can then be used for testing numerical model solutions. A particular example of an analytical solution is provided by Carrier and Greenspan [1958] for wave run-up and run-down on a beach with constant slope. In this section this quasi-analytical solution is used for a validation of D-Flow FM. For this purpose various D-Flow FM simulations have been carried out and its predictions are quantitatively compared to the corresponding analytical solutions. In particular it is verified how the accuracy of D-Flow FM responds to refinements of the spatial grid (Δx) and the time step (Δt), to obtain insight in the order of accuracy of D-Flow FM's numerical scheme for this kind of applications.

Linked claims

Claims that are investigated in the current validation case are:

- ◇ The accuracy of the time integration scheme in D-Flow FM is of first order.
- ◇ The accuracy of the spatial discretisation scheme is of order 0.4.
- ◇ The accuracy for predicting the time-varying shoreline position is of order 0.6 to 0.8 with respect to the spatial discretisation.

Approach

As already mentioned, analytical solutions of the non-linear shallow water equations for wave run-up and run-down on a beach are available for special cases only. The original approach of Greenspan and Carrier (which we will closely follow) is restricted to flows in one direction (i.e. dependent on one spatial coordinate, denoted by x). It is assumed that the flow evolves without friction and that viscosity effects are absent. Another essential assumption to allow an analytical solution is a linear bottom profile, i.e. a depth that depends linearly on the spatial coordinate x. Carrier and Greenspan's approach to construct the general analytical solution and the specific analytical solution used in the validation (including the parameter setting) are described below in separate sections. These analytical solutions form the basis for our quantitative analysis of the numerical accuracy of D-Flow FM. The approach in this accuracy analysis is as follows.

D-Flow FM simulations have been carried out for various systematic variations of both the grid size and the time step in the discretisation of the flow equations. In the variations of the spatial discretisation four one-dimensional uniform rectangular grids have been used, i.e. grids with cells of constant size Δx in the longitudinal direction of the computational domain. These grids will be described later, but here we already mention that they are constructed such that they provide stepwise (and mutually comparable) refinements of the spatial discretisation of the D-Flow FM model. The effect of time step Δt on the accuracy of the numerical simulation is investigated through variations of Δt.

The effect of grid-size and time-step refinements on the accuracy of D-Flow FM simulations will be quantified through statistical error measures derived from the errors $\Delta\zeta(x,t)$ and $\Delta u(x,t)$ in the model's prediction of water level $\zeta(x,t)$ and flow velocity $u(x,t)$ at the cell

centres[11].

The errors are defined by $\Delta\zeta(x,t) := \zeta_A(x,t) - \zeta_C(x,t)$ and $\Delta u(x,t) := u_A(x,t) - u_C(x,t)$, with subscript C indicating a variable computed by D-Flow FM and with subscript A indicating the corresponding analytical solution.

Three error measures L_1, L_2, and L_∞ have been considered for the quantification and comparison of the accuracy of the D-Flow FM solutions for the various variations in the grid size and the time step. These L_k ($k = 1, 2, \infty$) are actually vector norms applied to the set of model errors $\{\Delta\zeta(x,t)\}_{(x,t)\in A}$ and $\{\Delta u(x,t)\}_{(x,t)\in A}$ on some subarea A of the computational domain. They are defined by the following generic formula[12]:

$$L_k(\Delta\xi) = \sqrt[k]{\frac{1}{N_A}\sum\nolimits_{(x,t)\in A} |\Delta\xi(x,t)|^k} \tag{15.16}$$

In these equations, N_A denotes the number of cell centres $(x,t) \in A$. The ξ represents one of the two considered flow quantities, i.e. water depth ζ or flow velocity u.

The error measures will be computed for several sub-areas A of the full (spatial and temporal) computational domain. In one of the sections below these sub-areas are defined in further detail. Next, the dependency of the error norms on the grid size Δx and time step Δt will be evaluated for the various sub-areas, from which the order of accuracy of the numerical solutions can be obtained.

Carrier and Greenspan's analytical solutions

Flow equations in dimensional form

In this section the main steps of Carrier and Greenspan's approach are described that lead to a quasi-analytical solution of the one-dimensional (non-linear) shallow water equations. This is for the special case of no friction and no viscosity effects, and also with a bottom of constant slope. The one-dimensional flow equations are then as follows:

$$\begin{cases} \dfrac{\partial\zeta}{\partial t} + \dfrac{\partial}{\partial x}\big(u\cdot(\zeta+D)\big) = 0 \\ \dfrac{\partial u}{\partial t} + u\cdot\dfrac{\partial u}{\partial x} + g\cdot\dfrac{\partial\zeta}{\partial x} = 0 \end{cases} \tag{15.17}$$

In this equation x and t are respectively the spatial and time coordinate, ζ is the surface elevation with respect to a reference level, u is the horizontal flow velocity, D is the water depth with respect to the reference level, and g is the gravitational acceleration. The quantity $\zeta + D$ is thus the total water depth.

[11] Ideally, the error across the entire domain should be considered. It is, however, not clear how the discrete solution at cell centres should be extended in between the cell centres. Water depth h is assumed piecewise constant per grid cell in the time derivative of the continuity equation (a central approximation), but its value at the cell faces is obtained by (first-order) upwind. Velocity u at cell centers is obtained from averaging the values at the faces of the cells, thereby introducing averaging errors. Although in 1D u at cell faces could be considered when the grid is aligned with the flow (as is the case here), this is not possible in general. For this reason we decided not to exploit this possibility.

By only considering the value of ζ and u at cell centers, at most additional second-order errors are introduced due to the use of the 1-point Gauss integration rule (ζ) and 2-point central averaging (u). If necessary, this should be taken into account when interpreting the results.

[12] Note that $L_1(\cdot)$ is the mean absolute error over A, while $L_2(\cdot)$ is the RMS (root mean square) of the model error. It is readily verified that $L_\infty(\cdot)$ corresponds to the maximum of the pointwise absolute errors $|\Delta\xi(x,t)|$ over A.

The uniform slope of the bottom is represented by $D(x) := -\alpha \cdot x$ for some $\alpha > 0$. As a result the bottom's vertical position increases with the spatial coordinate x, and at $x = 0$ the bottom crosses the reference level. For a regular water-level profile $\zeta(x)$ there will be some x_{DW} for which $\zeta(x) + D(x) = 0$ for all $x > x_{DW}$. This x_{DW} can vary in time, however. In the model, the beach where the wave run-up and run-down takes place is thus at the right hand side.

Flow equations in non-dimensional form

The first step in Carrier and Greenspan's approach is to convert the flow equations into a representation with dimensionless variables. This is done through the following scaling of the involved quantities:

$$\begin{cases} x' = x/L_0 \\ t' = t/\sqrt{\dfrac{L_0}{\alpha \cdot g}} \\ \zeta' = \zeta/(\alpha \cdot L_0) \\ u' = u/\sqrt{g \cdot L_0 \cdot \alpha} \end{cases} \tag{15.18}$$

This scaling is governed by the parameter L_0 which represents some characteristic length. A proper choice for L_0 must be derived from the physical flow conditions in the study where the model is applied.

After substitution of the transformed variables of Equation (15.18) into Equation (15.17), the flow equations become:

$$\begin{cases} \dfrac{\partial \zeta}{\partial t} + \dfrac{\partial}{\partial x}\big(u \cdot (\zeta - x)\big) = 0 \\ \dfrac{\partial u}{\partial t} + u \cdot \dfrac{\partial u}{\partial x} + \dfrac{\partial \zeta}{\partial x} = 0 \end{cases} \tag{15.19}$$

Formally speaking the variables x, t, ζ and u in this equation should have been written as x', t', ζ' and u' but for ease of of notation the primes have been omitted.

Hodograph transformation of the non-dimensional flow equations

In a second step the hyperbolic equations of Equation (15.19) are rewritten in a form where the characteristic variables are the independent variables. Next another (so called *hodograph*) coordinate transformation is applied such that the instantaneous shoreline position in the physical space (which varies in time) is mapped onto a fixed point σ in the transformed time-space coordinate system (σ, λ). The transformation is such that the non-linear problem can be reformulated into a linear partial differential equation for the potential function $\phi(\sigma, \lambda)$:

$$\frac{\partial}{\partial \sigma}\left(\sigma \cdot \frac{\partial \phi(\sigma,\lambda)}{\partial \sigma}\right) - \sigma \cdot \frac{\partial^2 \phi(\sigma,\lambda)}{\partial \lambda^2} = 0 \tag{15.20}$$

with coordinate $\sigma \leq 0$. The fixed line $\sigma = 0$ corresponds to the free boundary at the beach. Since this free boundary moves up and down the bottom slope as a wave runs up and down the beach, σ, but also λ, depend in a complex way on both the physical coordinates (x, t) (and vice versa) and the flow variables (ζ, u).

From some solution $\phi(\sigma, \lambda)$ of Equation (15.20), the velocity u can be obtained through:

$$u = \frac{1}{\sigma} \cdot \frac{\partial \phi(\sigma, \lambda)}{\partial \sigma} \tag{15.21}$$

Next, from $\phi(\sigma, \lambda)$ and u, the water level ζ can be computed according to:

$$\zeta = \frac{1}{4} \cdot \frac{\partial \phi(\sigma, \lambda)}{\partial \lambda} - \frac{1}{2} \cdot u^2 \tag{15.22}$$

As a result the water level and velocity are available for the independent, but transformed, coordinates (σ, λ). The corresponding spatial and temporal coordinates in the physical space (*dimensionless* x and t as defined by scaling Equation (15.18)) are provided by:

$$x = \frac{1}{4} \cdot \frac{\partial \phi(\sigma, \lambda)}{\partial \lambda} - \frac{1}{16} \cdot \sigma^2 - \frac{1}{2} \cdot u^2 \tag{15.23}$$

and

$$t = \frac{1}{2} \cdot \lambda - u \tag{15.24}$$

Inverse of the hodograph transformation

The choice of the function $\phi(\sigma, \lambda)$ satisfying Equation (15.20) thus determines the coordinates (x, t) and the flow variables (ζ, u) at those coordinates. These are still expressed in terms of the transformed independent variables (σ, λ), however. The issue is therefore how to determine reversely for a given pair (x, t) the corresponding (σ, λ), so that they can be substituted into Equation (15.21) and Equation (15.22) to get the flow variables ζ and u as a function of (x, t). The mutual dependency of (x, t) and (σ, λ) is rather complex. No analytical recipe is available to get (σ, λ) directly for some arbitrary (x, t) combination. Therefore a numerical approach was followed for this inversion. A quasi-Newton method was used to minimise the following cost function with respect to σ and λ given the physical coordinates (x^*, t^*):

$$J(\sigma, \lambda) = \left(x^* - \left(\frac{1}{4} \cdot \frac{\partial \phi(\sigma, \lambda)}{\partial \lambda} - \frac{1}{16} \cdot \sigma^2 - \frac{1}{2} \cdot u^2 \right) \right)^2 + \left(t^* - \left(\frac{1}{2} \cdot \lambda - u \right) \right)^2 \tag{15.25}$$

The u in the right-hand side of this equation is as defined in Equation (15.21). In case the Jacobian matrix $\partial(x, t)/\partial(\sigma, \lambda)$ (with the x and t of Equation (15.23) and Equation (15.24)) never vanishes in the half plane $\sigma < 0$ (and waves do not break as pointed out by Carrier and Greenspan [1958]), there is a unique minimum of this cost function. When the minimisation yields a pair (σ, λ) for which $|J(\sigma, \lambda)| \leq \varepsilon$, with ε a sufficiently small convergence tolerance, the inversion procedure has provided the desired transformed coordinates (σ, λ) corresponding to the given (x^*, t^*). By means of Equation (15.22) and Equation (15.21) the values of the water level and velocity can then be computed.

If Jacobian matrix $\partial(x, t)/\partial(\sigma, \lambda)$ vanishes, or when the σ that is found is larger than zero, no proper solution (σ, λ) is available for (x^*, t^*). This will typically be for locations x where the beach is dry at time t and no well-defined water level and velocity exist. In this way the inversion procedure also provides a diagnostic for the dry/wet status of a location at a given time.

Choice of the potential $\phi(\sigma, \lambda)$

In the present D-Flow FM validation study the following potential $\phi(\sigma, \lambda)$ satisfying differential equation (15.20) has been adopted:

$$\phi(\sigma, \lambda) = A \cdot J_0(\omega\sigma) \cdot \cos(\omega\lambda - \psi) \tag{15.26}$$

with J_0 the 0th-order Bessel function of the first kind.

Carrier and Greenspan [1958] explain that for this choice of $\phi(\sigma, \lambda)$ we have the physical situation of a wave travelling shoreward from the deep region. It is reflected in the shallow region with the reflected wave travelling back out to sea. The reflection coefficient is unity and no breaking will occur as long as (dimensionless) amplitude $A \leq 1$.

The solution is periodic in time at every location x. While the period is $2\pi/\omega$ with respect to λ, the period is π/ω in time t of the non-dimensional flow model (15.19) because of the factor $1/2$ in Equation (15.24). The periodicity in time is not fully harmonic; particularly at locations near the shore large overtones are present. From Equation (15.23) we obtain the approximation $x \approx 1/16 \cdot \sigma^2$ for $\sigma \ll -1$, so far at sea ($x \ll -1$) the *spatial* shape of the wave is of the form $J_0(4 \cdot \sqrt{x})$. The wave is significantly distorted near the shore, however. This can be recognised from the water level shown in Figure 15.23. This figure is further introduced and discussed in another paragraph below.

Extremes of the water level, velocity, and the wave run-up/run-down excursions

Substitution of Equation (15.26) in Equation (15.21), using $dJ_0(x)/dx = -J_1(x)$, yields the expression for the velocity:

$$u(\sigma, \lambda) = -A \cdot \omega \cdot \frac{J_1(\omega\sigma)}{\sigma} \cdot \cos(\omega\lambda - \psi) \tag{15.27}$$

From this equation it can be derived that the extremes of u are assumed for $\sigma = 0$ and $\lambda = (n\pi + \psi)/\omega$, $n = 0, 1, \ldots$. These velocity extremes, of value $\pm$½$\cdot A \cdot \omega^2$, thus occur at the dry/wet interface on the beach. Similarly it can readily be verified that the extremes of ζ also occur at the dry/wet interface and amount to a value of $\pm$¼$\cdot A \cdot \omega$. On the basis of Equation (15.23) the extremes of the wave run-up and run-down excursions on the beach can be derived. These extremes are identical to those found for the water level, i.e. $\pm$¼$\cdot A \cdot \omega$.

Note that all these values are still for the scaled dimensionless model. Extremes of the flow variables of the original dimensional model are considered below in another paragraph.

Remaining free parameters in Carrier and Greenspan's analytical solution

With the $\phi(\sigma, \lambda)$ of Equation (15.26) we are left with three free parameters in the solution of the dimensionless model: amplitude A, radial frequency ω, and phase lag ψ. For a given choice of these parameters the manipulations covered by Equation (15.21) to Equation (15.26) provide a (quasi-)analytical solution for the flow equations of Equation (15.19). Here the spatial and temporal coordinates and flow variables are in a physical representation, but still in a scaled and dimensionless form. To obtain a solution for the flow equations in the original physical, and fully dimensional, representation the inverse of scaling (15.18) must be applied. In this rescaling the characteristic length L_0 and the slope α of the bottom provide two additional free parameters. Formally the gravitational acceleration g is also adjustable in the modelling but this will not be considered; g is set equal to $9.81 m/s^2$.

In summary the list of free parameters is:

$$\Theta := (A, \omega, \psi; L_0, \alpha) \tag{15.28}$$

The settings of the free parameters that have been used in the D-Flow FM simulations of the present validation study are presented in the next paragraph.

D-Flow FM model description

Initial condition

For running D-Flow FM an initial condition for the water level must be prescribed. For the velocity an initial condition is adopted that is uniformly zero all over the spatial domain. From Equation (15.20) and Equation (15.24) it can readily be verified that for $t = 0$ the velocity $u(x,t) = 0$ for all x if the phase lag is set to $\psi = -½\pi$. With this choice we thus obtain an initial condition for D-Flow FM that for both ζ and u is fully consistent with the analytical solution of the model equations. Because of this consistency, initial perturbations due to the mismatch between the analytical solution and the numerical solution should be small. In fact, it should be possible to make the initial perturbations arbitrarily small by taking a sufficiently high resolution in the spatial and temporal discretisation.

Besides the phase lag ψ, the generation of the initial condition requires a proper choice of the other two parameters: amplitude A and angular frequency ω in the potential function $\phi(\sigma,\lambda)$ of Equation (15.26). With the phase ψ as specified above the following values were assigned to the parameters in the potential:

$$A = ½\ ,\ \omega = 1\ ,\ \psi = -½\pi \tag{15.29}$$

With these values of the parameters in $\phi(\sigma,\lambda)$ the initial condition of ζ for the still *dimensionless* flow equations (15.19) is computed; it is shown in Figure 15.23. The velocity is uniformly set equal to zero at $t = 0$, as already mentioned. For D-Flow FM computations this initial condition must still be rescaled to the one for the dimensional model (15.17), i.e. with the coordinates (x,t) and flow variables (ζ,u) in their original physical representation. This rescaling is considered in a later paragraph.

Boundary condition

Another issue is to obtain proper boundary conditions. At the right side of the model the boundary is positioned at a location on the beach that is always dry. No water level or velocity needs to be prescribed there. At the left side a boundary condition is imposed at a location x_L at sea that is always wet and sufficiently far away from the part of the beach that is subjected to drying and flooding. At this location x_L a water level is prescribed[13]. The velocity at that location is generated by the numerical model within the time integration of the flow equations.

For optimal consistency with the analytical solution it is favorable to prescribe the water-level boundary condition at an x_L where for all times t the velocity is constantly zero. In strict sense no such location exists for Carrier and Greenspan's analytical solution. It is possible, however, to find locations x where the amplitude of u has a local minimum. From Equation (15.27) we obtain that $u(\sigma,\lambda) = 0$ for all $\sigma = \sigma_k := z_k/\omega$, with z_k the k-th root of the first-order ordinary Bessel function $J_1(z)$. The physical coordinate x_k corresponding with σ_k is obtained from Equation (15.23). This value is not constant in time but varies around a mean value of $-\sigma_k^2/16$. They correspond with the x–locations in Figure 15.23 where the red curve assumes an (upper or lower) extreme (nodes in the u–signal correspond with anti-nodes in the ζ–signal).

[13] In the present computations the seaward boundary is thus dealt with in standard form, i.e. treated as an open boundary with a prescribed water level. Alternative boundary conditions (e.g. a weakly reflecting condition that require a combination of water level and velocity) have not been considered in the present study.

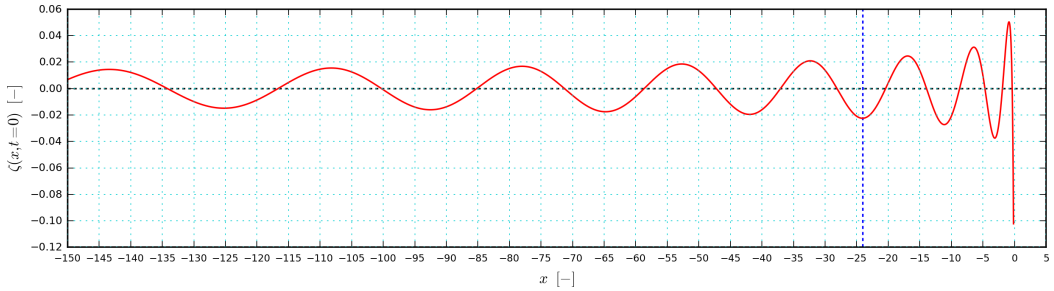


Figure 15.23: *Initial water-level profile $\zeta(x, t = 0)$ in Carrier and Greenspan's analytical solution of the scaled and dimensionless flow model (15.19). In this solution the potential function $\phi(\sigma, \lambda)$ of Equation (15.26) has been adopted with the set of parameters $A =$½, $\omega = 1$, and $\psi = -$½π. For these parameters, and still $t = 0$, the velocity profile is uniformly equal to zero. The vertical dashed blue line denotes the x location where in the D-Flow FM simulations (*after re-transformation *to the dimensional flow model of Equation (15.17)) the seaward water-level boundary condition is imposed.*

The 6th root of $J_1(z)$ was selected to define the seaside boundary location x_L of the D-Flow FM model. This 6th root of $J_1(z)$ reads $z_6 = 19.61586$ leading to $x_L = -\sigma_6^2/16 = -24.04887$. In Figure 15.23 this x_L is marked by the vertical dashed blue line. The analytically computed water-level time series prescribed at x_L is shown in Figure 15.24. Evaluation of the velocity time series at $x_L = -24.04887$ showed that the extremes of this series turn out to be $2.256 \cdot 10^{-5}$, which is indeed very small compared to the velocity's overall extremes of 0.25. We recall that all these numbers are still for the *scaled non-dimensional* model. Below it will be specified which value X_L in the physical domain (where the actual D-Flow FM computations are carried out) is assigned to the $x_L = -24.04887$ in the *non-dimensional* domain.

With regard to this X_L, and the way a water-level boundary condition is specified in D-Flow FM, the following must be mentioned. In the construction of the spatial grids (for the various choices of grid size Δx), the two faces of the cell at the seaward boundary have each time been set exactly at $x = X_L$ and at $x = X_L + \Delta x$. However, in D-Flow FM a prescribed water-level time series is imposed at the 'virtual' cell centre left of this cell, i.e. at $x = X_L -$½Δx. To obtain optimal consistency between numerical and analytical solutions, the water-level time series to be imposed have therefore be computed at the locations $x = X_L -$½Δx, i.e. a different time series has been used for each different Δx.

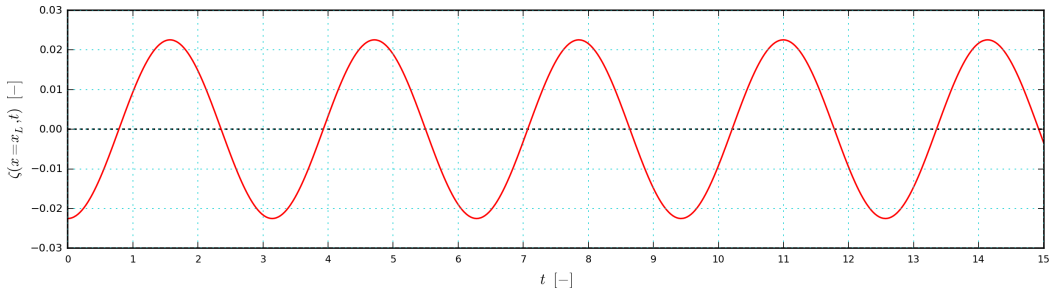


Figure 15.24: *Time series of the water-level boundary condition $\zeta(x = x_L, t)$ in Carrier and Greenspan's analytical solution of the scaled and dimensionless flow model (15.19). In this solution the potential function $\phi(\sigma, \lambda)$ of Equation (15.26) has been adopted with the set of parameters $A =$ ½, $\omega = 1$, and $\psi = -$½π. The boundary is located at $x_L = -24.04887$, as marked by the vertical blue dashed line in Figure 15.23.*

Setting of the physical domain parameters

So far the setting of the free parameters and the resulting initial and boundary condition have been presented for the scaled, non-dimensional flow model (15.19). For the determination of the solution of the model in the dimensional/physical domain the remaining parameters α and L_0 must yet be specified. In the physical domain the x-coordinate where the linear bottom profile $D(x)$ crosses the reference level is, just as for the non-dimensional model, set to $x = 0$ [m]. The wave run-up and run-down induced drying and flooding of the beach is thus centered around this location. A choice must be made for the dimensional coordinate that corresponds to the dimensionless $x_L = -24.04887$ where for the non-dimensional model the boundary is positioned. In the D-Flow FM computations this boundary has been set to $X_L = -30,000$ [m]. At this boundary depth D is set to 300 [m]. This then leads to a bed slope parameter $\alpha = 0.01$.

Rescaling of the analytical solution of the non-dimensional model

All parameters in the non-dimensional Carrier and Greenspan model have now been initialised (see Equation (15.28)). From the correspondence

$$x_L = -24.04887 \text{ (non-dimensional)} \leftrightarrow X_L = -30000\text{m (dimensional)} \tag{15.30}$$

the characteristic length (i.e. the L_0 in Equation (15.18)) is obtained: $L_0 = 1247.45995$m. With this L_0, and with the bed slope $\alpha = 0.01$, all scaling parameters listed in Equation (15.18) can be evaluated. On the basis of (the reciprocal of) this scaling a Carrier and Greenspan's analytical solution of the *non-dimensional* model (15.19) can be translated into an analytical solution of the corresponding *dimensional* model (15.17) solved in D-Flow FM simulations. This rescaling from non-dimensional variables (marked with a prime) to the dimensional variables is as follows:

$$\begin{cases} L_0 = 1247.45995\text{m}\ ,\ \alpha = 0.01\ ,\ g = 9.81\text{m/s}^2 \rightarrow \\ \quad x = L_0 \cdot x' = 1247.45995 \cdot x' \\ \quad t = \sqrt{\dfrac{L_0}{\alpha g}} \cdot t' = 112.766163 \cdot t' \\ \quad \zeta = \alpha L_0 \cdot \zeta' = 12.4745995 \cdot \zeta' \\ \quad u = \sqrt{g L_0 \alpha} \cdot u' = 11.0623606 \cdot u' \end{cases} \tag{15.31}$$

From this scaling we obtain the extremes of the flow variables in the analytical solution of the *dimensional* model:

$$\begin{cases} \text{extremes of water level } \zeta : \pm\tfrac{1}{4} A\omega\alpha L_0 = \pm 1.55932494\text{m} \\ \text{extremes of velocity } u : \pm\tfrac{1}{2} A\omega^2 \sqrt{g L_0 \alpha} = \pm 2.76559014\text{m/s} \\ \text{extremes of dry/wet interface position} : \pm\tfrac{1}{4} A\omega L_0 = \pm 155.932494\text{m} \end{cases} \tag{15.32}$$

In the non-dimensional model the water level and velocity are periodic in time with period π/ω. In the dimensional model the period is then (with substitution of the numerical values of the various quantities)

$$\text{Period } P : \sqrt{\frac{L_0}{\alpha g}} \cdot \frac{\pi}{\omega} = 354.265348153\text{s} = 5.904422469\text{minutes} \tag{15.33}$$

Illustrations of the analytical solution of the dimensional model

With the parameter settings described above analytical solutions of the flow equations of the *dimensional* model were computed. The spatial and temporal characteristics of water level and velocity are shown in Figure 15.25, Figure 15.26, Figure 15.27, and Figure 15.28. The dimensions and magnitudes of these ζ and u are now as in the actual D-Flow FM simulations.

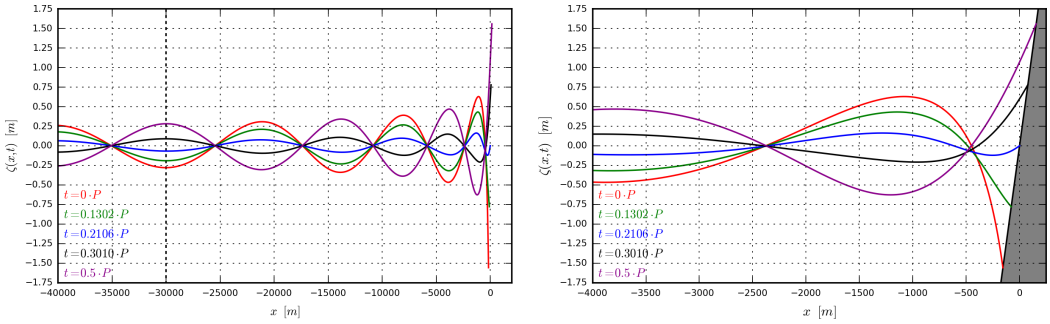


Figure 15.25: *Analytical water-level solution of dimensional flow model (15.17). The spatial variation of ζ is shown here for a set of selected times t expressed as a fraction of period P (5.9044 minutes). Further explanations in the main text.*

The spatial variation of the water level at a set of selected time points t is illustrated in the two panels of Figure 15.25. In the left panel for the whole computational domain, in the right panel for the part of the beach where the wave run-up and run-down takes place. The times for which the spatial profiles have been plotted are expressed in the period P of the periodic solution ($P = 5.9044$ minutes). The red curve is for $t = 0$ and represents the initial condition for the water level in the D-Flow FM computations (see also the corresponding non-dimensional form shown in Figure 15.23). This initial condition is at a time where the wave run-up is minimal. The curve plotted in magenta, for $t = \frac{1}{2}P$, represents the situation where the wave run-up is at its maximum excursion. The black and the green curves are for times in between, where the wave run-up is halfway its extremes. The blue curve is for a time that at the beach the wave run-up is at its equilibrium elevation $\zeta = 0$. Note that this equilibrium is not (as possibly expected) for time $t = \frac{1}{4}P$, and neither is ζ uniformly equal to zero at this time. From the $\zeta = 0$ level crossings near $x = -500$m it can also be observed that the x–coordinate of these crossings varies slightly in time. To a lesser extent this also holds for the crossings further seaward; the 'spread' in the x-coordinates becomes gradually negligible. This confirms an earlier statement that there is no spatial location where in time the surface elevation is constantly zero.

In the same way, but for *different* times t, the spatial variation of the velocity is shown in the two panels of Figure 15.26. The curves in red and magenta are for times where the velocity at the dry/wet interface on the beach is at its lower and upper extreme. Like with the water level there is no spatial location where the velocity is constantly zero. It can be observed that the amplitude of the velocity variations decreases much more rapidly in seaward direction than for the water level. The dashed black closed trajectory in the right panel provides a kind of phase space plot of the system's state at the dry/wet interface on the beach. In fact, the trajectory consists of the points $(x_{WD}(t), u(x_{WD}(t)))$ with $x_{WD}(t)$ the front of the wave such that the beach is wet for $x < x_{WD}(t)$ and dry for $x > x_{WD}(t)$. The $u(x_{WD}(t))$ is the velocity of the wave's front at time t. In time the dashed trajectory is circulated in clockwise direction.

An alternative view on the analytical solution of the *dimensional* flow model is given in Figure 15.27 and Figure 15.28. It consists of plots of time series of water level and velocity at different locations, mainly in the vicinity of the beach and in the drying-and-flooding area (whence the interruptions in the curves, indicating dry-bed conditions). The time series are presented from $t = 0$ to $t = 2P$, i.e. over two cycles of the temporal periodicity.

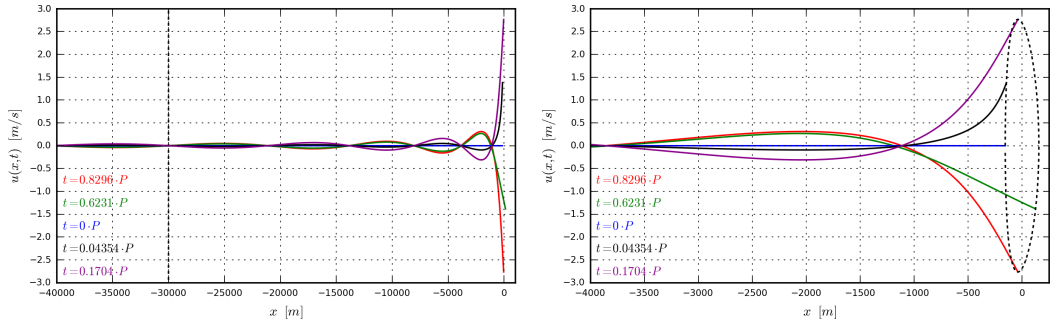


Figure 15.26: *Analytical velocity solution of the dimensional flow model (15.17). The spatial variation of u is shown here for a set of selected times t expressed as a fraction of period P (5.9044 minutes). Further explanations in the main text.*

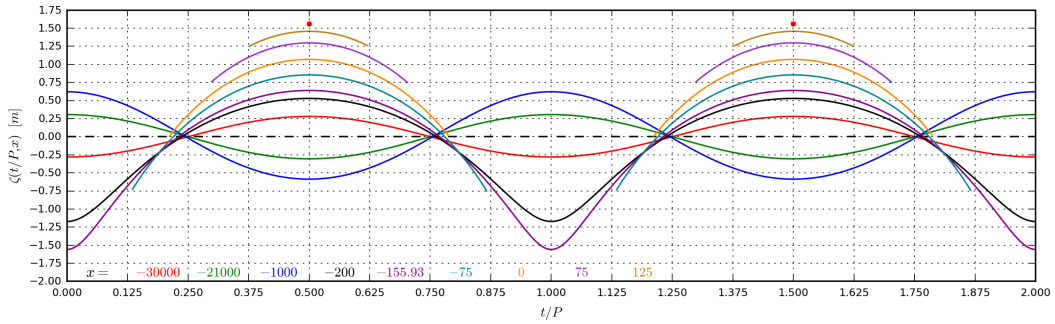


Figure 15.27: *Analytical water-level solution of dimensional flow model (15.17). Time series of ζ are shown for a selected set of locations x. Further explanations in the main text.*

The water-level time series are shown in Figure 15.27. The red curve is the water level at the left boundary of the D-Flow FM model and thus represents the left boundary condition (cf. the corresponding non-dimensional form shown in Figure 15.24). The gaps in the curves for $x > -155.93$m reflect time intervals where the beach is dry. For $x \leq -155.93$m the beach is always wet. The time series at these locations are thus 'complete' but clearly not harmonic, especially in the vicinity of the beach where the non-linear components in the flow equations produce significant overtones.

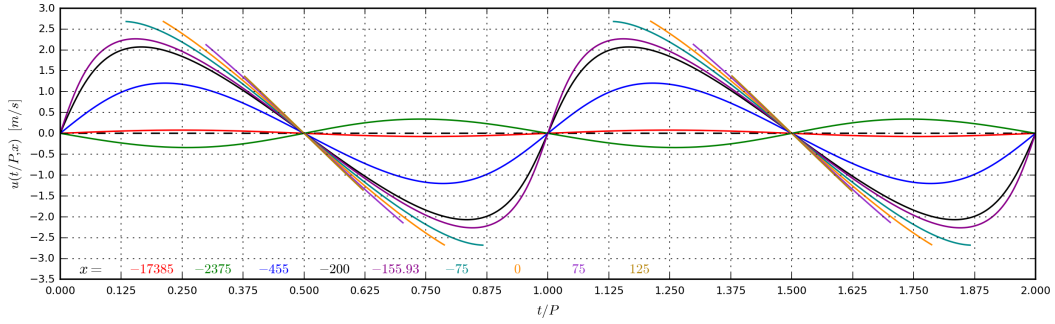


Figure 15.28: *Analytical velocity solution of dimensional flow model (15.17). Time series of u are shown for a selected set of locations x. Further explanations in the main text.*

The velocity time series are shown in Figure 15.28, in and near the drying-and-flooding area

at the same x-locations as the water-level time series in Figure 15.27. At sea, however, other locations have been chosen, in particular those where the velocity amplitude is not too small. Notice that the velocity is always exactly zero for times t that are an integer multiple of ½P. Such behaviour is not found for the water levels: the times at which the equilibrium level $\zeta = 0$ is crossed depends on the location x, which implies a spatially varying phase.

Computational settings in the D-Flow FM simulations

The computational settings in the D-Flow FM simulations are as follows:

- ◇ No bed friction, no surface drag, no horizontal viscosity.
- ◇ Schematisation of advection: `AdvecType = 33` (Perot q(uio-u)).
- ◇ Limiter type for cell-center advection velocity: `Limtypmom = 4` (Monotone Central).
- ◇ Time integration: `TimeStepType = 2` (full implicit step reduce); `teta0 = 0.55` ($\theta = 0.55$).
- ◇ Bed level handling: `BedlevType = 3` (bottom levels at velocity points, using the mean cell depth).
- ◇ Drying/flooding: `chkadvd = 0.0` (parameter in handling advection over almost dry bed).
- ◇ Drying/flooding: `epshu = 0.0001` (threshold determining wet or dry bed).

In D-Flow FM the numerical setting parameter `chkadvd` ('check advection depth') is relevant in simulations with frictionless flows over a fully or almost fully dry bed (as in the present case). For (almost) dry cells it may then happen that within a time step Δt the predicted water level gets below the cell depth. In that case the time step should be repeated with a smaller Δt. To prevent a substantial increase of computation time, an ad-hoc correction procedure is applied instead. In this correction procedure explicitly discretised terms of the flow equations, and particularly those that are extracting water from nearby dry cells, are reduced by a factor. This factor is determined by the ratio of the current water depth and parameter `chkadvd`, which represents a sort of threshold depth in *meters*. To avoid an excessive effect on the results, `chkadvd = 0.0` should be set in computations with extremely low water depths. In other situations its value is less critical and the default value of 0.1 (m) can be used.

The parameter `epshu` is a threshold in the D-FLOW FM's modelling of drying and flooding. Cells with a water depth larger than `epshu` are treated as wet, while for cells with a water depth less that `epshu` a flooding/drying procedure is activated. In D-Flow FM the default value `epshu = 0.0001` (m) is set. This default is adopted in the present simulations.

Grid and time step variations in the D-Flow FM simulations

In the present D-Flow FM computations four different grid sizes have been considered to discretize the domain that runs from the left boundary at $x_L = -30,000$m to the right boundary that was set at x_R=300m[14]. x_R was chosen about twice as large as the maximum excursion of the wave run-up on the beach (cf. (15.32)) to ensure a sufficiently large stretch of dry beach. It guarantees a number of cells near the right boundary to remain dry in all simulations,

[14]The boundaries x_L and x_R were slightly modified in the actual computations to x_L=-30000.001 m and x_R=299.999 m. The reason for this tiny shift of 1mm to the left is a problem that was encountered with the uniform grid size $\Delta x = 1$m. With the boundaries set at x_L=-30000 and x_R=300, a grid size of $\Delta x = 1$m means that there is a cell face exactly located at $x = -999$m. In D-Flow FM, however, external inputs with a value of -999 are treated as missing values. Reading the depth $D(x)$ from the input file, the sample $(x, D) = (-999, -9.99)$ was thus not accepted. The problem could only be circumvented by a small shift of the boundary locations and hence of the grid points.

In the computation of the analytical solution (initial condition, boundary condition, flow solution inside the domain) this shift has been taken into account. In other words, the results of the accuracy analyses are not affected by the applied shift.

making specification of a condition at that boundary irrelevant.

All four grids are equidistant with a constant cell size Δx. The coarsest grid that has been considered has a size of $\Delta x = 10$m. With a constant bed slope of $\alpha = 0.01$ a bed level variation of 10cm is then present across each cell. The other three grids are 'gradual' refinements of the coarse grid and have cell sizes of $\Delta x = 5, 2,$ and 1m respectively. These four variations of Δx thus provide stepwise (and mutually 'comparable') refinements of the spatial resolution. With the boundaries at the x_L and x_R listed above, the number of grid cells varies from to 3030 to 30300.

To assess the effect of the time step on the accuracy, D-Flow FM simulations have been carried out for Δt in the range of 0.05 to 1.5s. The upper bound of 1.5s is the largest time step possible on the coarsest grid within the D-Flow FM stability restriction. On the finer grids the stability limit sets a smaller maximum Δt.

Besides the time step also the simulation time has been varied. With these variations spin-up effects were analysed, to assess after which simulation time transient effects have disappeared and a steady periodic regime is reached. A periodic behaviour in time is expected because of the application of a periodic boundary condition and the periodic analytical solution. The periodicity of the D-Flow FM solutions, however, may be affected by the discretisation of the flow equations. For the simulation period time intervals from $30P$ to $3000P$ have been considered. We recall that $P = 5.9044$[minutes] is the length of the temporal periodicity of the analytical solution and the boundary signal.

Selection of the sub-areas in the model error analysis

In the model error analysis the three error norms of Equation (15.16) have been evaluated for 7 sub-domains A_i of the (spatial and temporal) computational domain.

In space the sub-domains A_i consist of 7 consecutive intervals that together cover the entire spatial domain. For their boundaries we have used the zero crossings $\zeta(x, t = 0) = 0$ of the initial distribution of the water level. The 7 sub-domains are shown in Figure 15.29. We recall that the zero crossings of the water level vary in time, but with a magnitude that is very small compared to the width of the A_i, i.e. the boundaries of the A_i are roughly formed by the nodes of the standing wave pattern of the water level. By considering several intervals, the quality of the D-Flow FM's solutions can be established all over the computational domain, and also for different flow conditions. Notice that, because of the increasing surface-elevation-to-depth ratio and the strongly increasing velocity-to-wave-celerity ratio (the Froude number), non-linear effects increase significantly in the direction from A_1 to A_7. Non-linear effects will in particular be significant in domain A_7 where the beach is repeatedly subjected to drying and flooding.

In time each sub-domain A_i uses the same interval. It consists of a number of periods P taken at the end of a simulation time.

As for the numerical solution considered per sub-domain A_i: the solution at all cell centres x_n within a sub-domain A_i is considered, but in time only the solution at the time levels $t_m = t_0 + m \cdot \delta t$ that lie inside the time interval of the A_i are considered, with m a non-negative integer and $\delta t := P/8$. With these 8 "snapshots" of the flow variables per time cycle P, the variability of the flow variables within a temporal cycle is reasonably covered. t_0 was set such that only the spatial solution of the last 30 periods of a D-Flow FM computation was written to output, giving $M = 240$ time levels $\{t_m\}_{m=1}^{M}$ with flow maps for each of the 4 spatial grids that have been considered. This gives a total of $240 \times 3030 = 727200$ to $240 \times 30300 = 7272000$ values of computed water level and velocity.

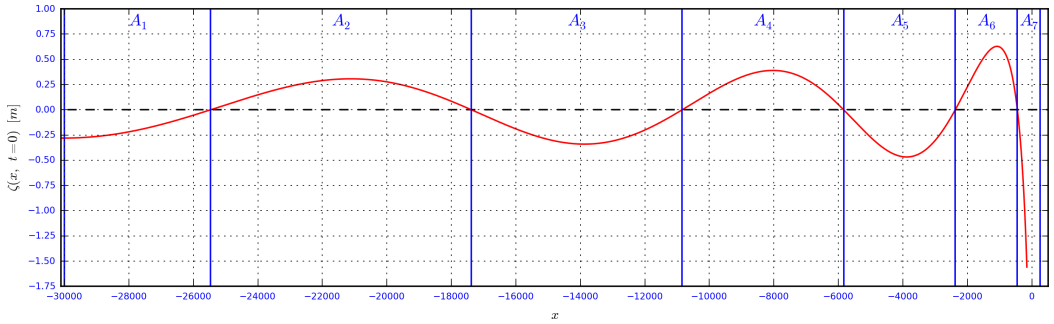


Figure 15.29: *Spatial boundaries of the 7 sub-domains A_i where the error norms of Equation (15.16) have been evaluated. The boundaries (vertical blue lines) are defined by the zero crossings of the initial water level (red line, cf. the part of the dimensionless initial water-level profile after the vertical dashed blue line shown in Figure 15.23).*

Notice that storing and processing the numerical solution at all numerical time levels within the A_i is not feasible. Because of the small time steps Δt that are used (as a result of the stability restriction), that would have led to huge output files and very large numbers of solution values to be processed, even for this 'small' 1D validation test. Recall that, because an iterative solution procedure is required to determine the exact value of ζ and u at some coordinates (x, t) (which variables are all a function of the transformed coordinates (σ, λ), cf. Equation (15.21) to Equation (15.24)), it is especially the computation of the analytical solution at the output locations that is time consuming.

The space-time solution values computed by D-Flow FM on the grids of (x_n, t_m) points of the sub-domains A_i have been saved in (mapped to) output files using the functionality "`MapInterval`". In such a map the water level and velocity is available for every cell of the model's spatial discretisation at every output time level.

In the evaluation of the error norms the analytical solution is computed in all (x_n, t_m) points of the 7 sub-domains A_i, with the (x_n, t_m) strictly as read from the map files produced by D-Flow FM. For A_1 to A_6 the analytical and computed water level and velocities can straightforwardly be obtained and so can the error norms for these regions. For sub-domain A_7 the drying and flooding complicates the comparison of the flow variables and the evaluation of the error norms. The reason of this complication is twofold:

- ◇ For the analytical solution it can unambiguously be determined whether or not a (x_n, t_m) is dry or wet. For an output location of D-Flow FM this status is less straightforward to establish. In the present application the following criterion was applied to decide on the dry/wet status of a location in sub-region A_7: the cell with cell centre x_n is dry at time t_m if the total water depth $\zeta(x_n, t_m) + D(x_n)$ returned by D-Flow FM (i.e. the elevation of the free surface with respect to the bed) is less than 0.001m (i.e. less than 1mm).
- ◇ We also have to deal with the situation that a $(x_n, t_m) \in A_7$ can be dry according to the analytical solution, while it is wet according to D-Flow FM (and vice versa). Here the approach was followed that in dry cells, whether according to the analytical solution or to the numerical D-Flow FM solution, the analytical or numerical water depth and velocity were set to zero. Cells that are dry in *both* the analytical and the D-Flow FM solution are discarded in the evaluation of the errors norms, meaning that these cells are *not* included[15] in the error sum of Equation (15.16). On the other hand, all cells that are wet, also those that are only wet in either the analytical or the numerical solution, are

[15] Including these cells, where by definition the error is zero, over the (in principle) arbitrarily large dry part of the beach would otherwise have blurred the error measurement in sub-domain A_7.

included in the error sum of Equation (15.16). In this way the effect of mismatches in the numerical prediction of the position of the dry/wet front are taken into account in the error measurement for sub-domain A_7.

Results

For each of the four uniform grids (with grid sizes $\Delta x = 10,\ 5,\ 2$ and 1m) D-Flow FM computations were carried out for various time steps Δt and simulation times T. We will frequently express T as $K \cdot P$, with P the period of the solution in time (5.9044 minutes for the analytical solution) and K the computed number of 'P-cycles'. Depending on the actual spatial grid and associated computation time, K has been varied from 30 to even 3000. In the next paragraphs the results of the computations are reported and discussed.

Length of the simulation time

In first instance D-Flow FM computations with the coarsest spatial discretisation ($\Delta x = 10$m) were carried out to assess the dependency of the errors ($\Delta\zeta$, Δu) on the time step Δt. At the same time, however, the effects of spin up were verified, by means of visual inspection of time series of the 'residuals' $\Delta\zeta$ and Δu. This was done to determine the simulation time T required to attain a state where the D-Flow FM prediction of the flow variables shows a reasonably periodic behaviour.

The first experiments with T of length $30P$ revealed that this simulation time was by far not long enough to reach a satisfactorily periodic regime of the numerical solution. This was despite the fact that the initial condition used in D-Flow FM was set equal to the analytical solution, indicating that the numerical solution is 'significantly' different from the analytical solution. This was also found in later experiments with finer grid.

With Δx still equal to 10m the simulation time was increased to $T = 1000P$. As is shown in Figure 15.30 and Figure 15.31 this appears to be sufficient to achieve a stationary regime. In the analysis of the model's spin up time, possible effects of errors due to the D-Flow FM temporal discretisation were minimised by setting the time step to a very small value: $\Delta t = 0.05$s. The adequacy of this value of Δt will be clarified later when the results of time-step variations are presented.

Figure 15.30 shows the temporal evolution of the water-level error $\Delta\zeta$ at the location $x = -175$m. This location is always wet, but very close to the region of the beach that is repeatedly subjected to wave run-up and run-down, cf. the right panel of Figure 15.25 and Figure 15.27. The time series of the velocity error Δu shown in Figure 15.31 are for the location $x = -300$m, which is also in the vicinity of the drying and flooding region, cf. the right panel of Figure 15.26.

From the three panels in both figures the following can be remarked:

- ◇ From the time series in the top panels it is readily observed that transient effects are present up to, say, $t \approx 500P$.
- ◇ The middle panels show that, apart from the periodicity P induced by the boundary condition, in the beginning of the simulation a period of about $12P$ to $13P$ is present as well. A spectral analysis of the first 30 periods has revealed that this long-period perturbation seems to be mainly a beating in the time signal due to the interaction of several components with periods close to P, rather than a persistent long-term periodic component. The periodograms of $\Delta\zeta$ and Δu show that the energy is concentrated in a rather small band around P, while hardly any energy is present for periods larger than P. This would imply that non-linear numerical errors in D-Flow FM apparently generate additional components

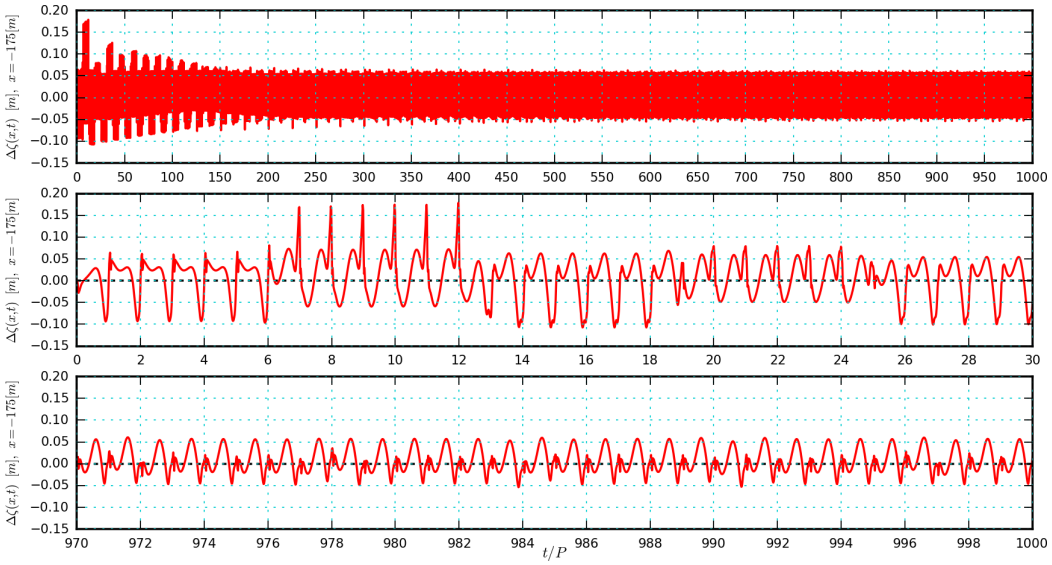


Figure 15.30: *Time series of the water-level residual $\Delta\zeta(x,t)$ at $x = -175$m for D-Flow FM results computed on the coarsest spatial grid $\Delta x = 10$m with time step $\Delta t = 0.05$s. In the upper panel the time series of $\Delta\zeta(x,t)$ is shown for the entire simulation time $T = 1000P$. In the middle and lower panel the time series is shown for the first 30 periods P and the last 30 periods P.*

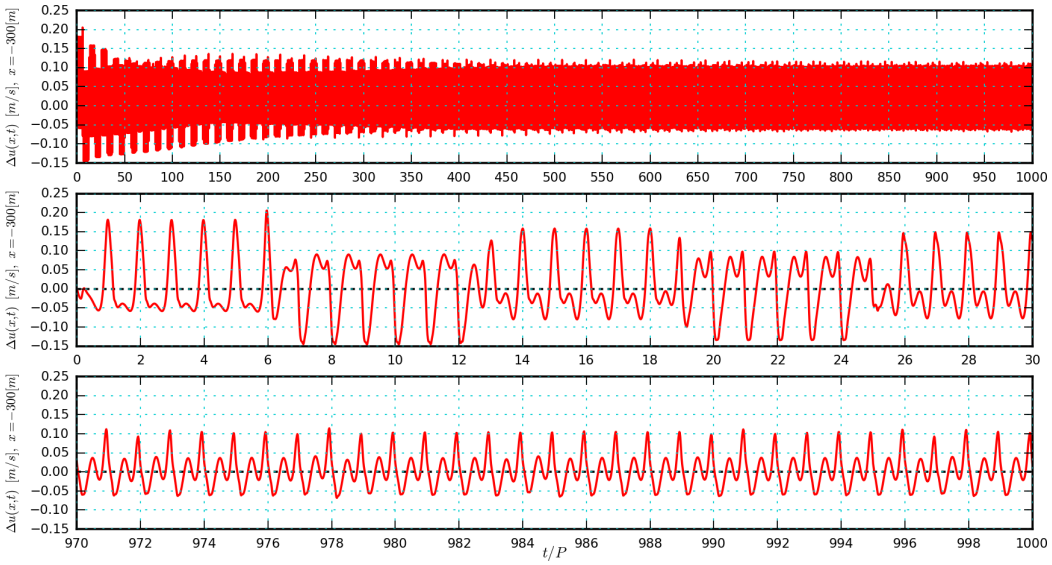


Figure 15.31: *Time series of the velocity residual $\Delta u(x,t)$ at $x = -300$m for D-Flow FM results computed on the coarsest spatial grid $\Delta x = 10$m with time step $\Delta t = 0.05$s. In the upper panel the time series of $\Delta v(x,t)$ is shown for the entire simulation time $T = 1000P$. In the middle and lower panel the time series is shown for the first 30 periods P and the last 30 periods P.*

with periods slightly different from P that damp out very slowly.
However, the slow variation of $\Delta\zeta$ and Δu averaged over P, which variation is superimposed on the variation within the periods P, indicate the presence of a long-period component. A simple linear analysis shows that the longest wave that can oscillate between the boundaries has a period of $12.487843P$. The next possible standing waves have periods that are 3, 5, 7, ... times smaller. This agrees very well with what is observed

in the figures.

◇ The last 30 periods depicted in the lower panels reflect a steady and periodic temporal behaviour, indicating that indeed a stationary regime has been reached.

Important: since the boundary conditions of the model (water level imposed at the left open boundary; velocity equal to zero 'imposed' at the right closed boundary) are fully reflecting and since no physical dissipation is present in the model (bottom friction and horizontal viscosity have both been set to zero), initial perturbations are damped out by numerical dissipation *only*, which, because of the fine to very fine grids that have been used, is very small. This is why it takes many periods to attain a steady periodic regime. Results presented below will show that the number of periods required is roughly inversely proportional to the grid size: the smaller the grid size, the more periods are required to attain a steady periodic regime. This is in agreement with the dissipation introduced by the applied first-order upwind discretisation, which scales with the grid size.

Notice that the amount of dissipation inside the domain could have been increased by using a larger value of the time discretisation parameter θ. This would have made the time integration more dissipative, but also less accurate, so this was not an option. A viable option would have been to use a dissipative Riemann-type condition at the open left boundary. Since the implementation of that boundary condition in D-Flow FM is less accurate than the water-level boundary condition, this was not an option either. We expect, however, a significantly faster attenuation of initial perturbations with a Riemann-type condition. This remains to be investigated.

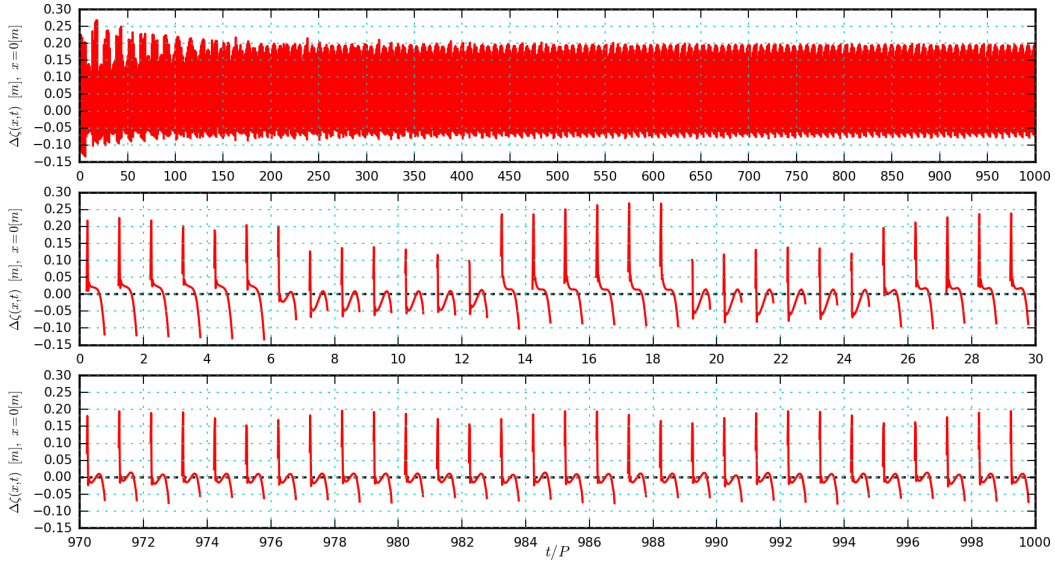


Figure 15.32: *Time series of the water-level residual $\Delta\zeta(x,t)$ at $x = 0$m for D-Flow FM results computed on the coarsest spatial grid $\Delta x = 10$m with time step $\Delta t = 0.05$s. In the upper panel the time series of $\Delta\zeta(x,t)$ is shown for the entire simulation time $T = 1000P$. In the middle and lower panel the time series is shown for the first 30 periods P and the last 30 periods P.*

Also in the region with drying and flooding the residuals become reasonably stationary after a simulation time of 500 periods, although at $t = 1000P$ a small long-period perturbation of about $8P$ is still present. This is illustrated in Figure 15.32 where the water level residuals at $x = 0$m are shown. This location is at the drying and flooding's equilibrium position at the beach, cf. the right panel of Figure 15.25 and Figure 15.27.

Comparing the results for the water-level residual at the end of the simulation time $1000P$ at $x = -175$m (lower panel of Figure 15.30) and at $x = 0$m (lower panel of Figure 15.32) it becomes clear that the largest errors in the D-Flow FM results occur on the beach in the run-up/run-down area. This is as was to be expected.

So far spin up effects have been presented for the coarsest computational grid. The spin up has also been examined for the other three grids. For an impression of the results one is referred to Figure 15.33. Like in Figure 15.30, this figure shows the water-level residuals at $x = -175$m, but now for the D-Flow FM computation on the finest grid $\Delta x = 1$m. In the top panel a very slow oscillation of hundreds of periods can be observed in (in particular) the upper extremes of the $\Delta\zeta$ series that at $T = 1000P$ has still not vanished, i.e. a stationary regime has not yet been reached. This can be observed in more detail from the last 30 periods depicted in the lower panel of Figure 15.33. Similar results were found for the time series of the water-level and velocity residual at other locations. The very slow oscillation is likely to be the result of a beating between initial perturbations with periods very close to P. This has not been investigated any further.

A comparison of $\Delta\zeta$ shown in Figure 15.33 with the time series of Figure 15.30 reveals that the decrease of the cell size from 10 to 1m leads to a reduction of the errors in the water level, as expected. The reduction is, however, considerably less than a factor 10, which is the reduction that one would expect from the use of an (at least) first-order accurate discretisation scheme. For this application the order of accuracy of D-Flow FM is apparently considerably less than first order.

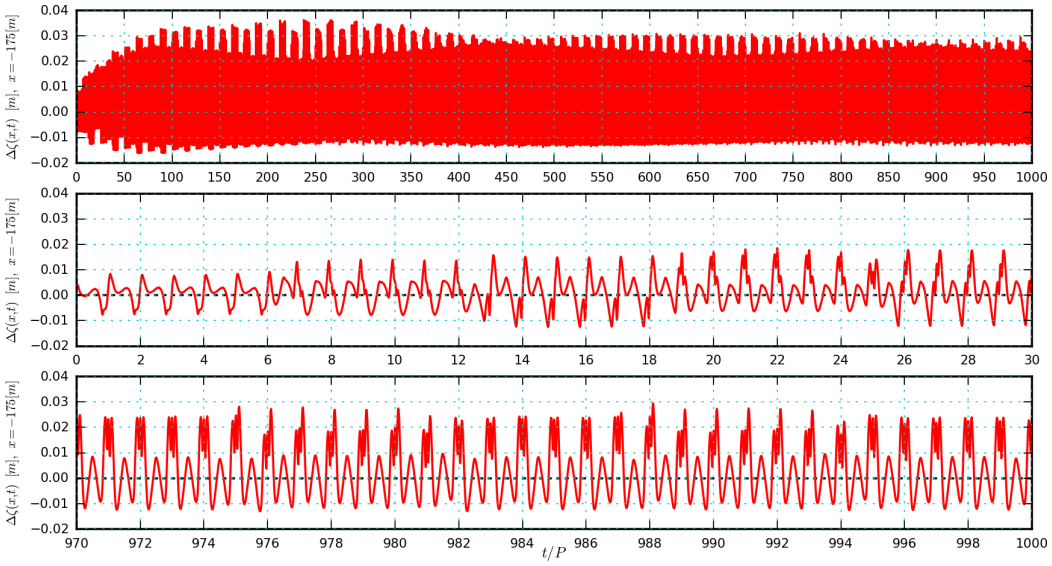


Figure 15.33: *Time series of the water-level residual $\Delta\zeta(x,t)$ at $x = -175$m for D-Flow FM results computed on the finest spatial grid $\Delta x = 1$m with time step $\Delta t = 0.05$s. In the upper panel the time series of $\Delta\zeta(x,t)$ is shown for the entire simulation period $T = 1000P$. In the middle and lower panel the time series is shown for the first 30 periods P and the last 30 periods P.*

Just as for the grid with $\Delta x = 1$m a stationary regime was not reached either at $T = 1000P$ for the two grids with cell size $\Delta x = 5$m and $\Delta x = 2$m. Further experiments with these two grids revealed that a considerably longer simulation time than $1000P$ is required to achieve that transient effects sufficiently disappear. For the present one but finest grid ($\Delta x = 2$m) this is illustrated by means of Figure 15.34. In the same way as before, the time evolution of the error $\Delta\zeta$ is shown, but now for a considerably longer simulation length of $T = 3000P$. The $\Delta\zeta$ is here for the location $x = -8035$m and provides a typical example of the temporal

behaviour of $\Delta\zeta$ and Δu in this simulation.

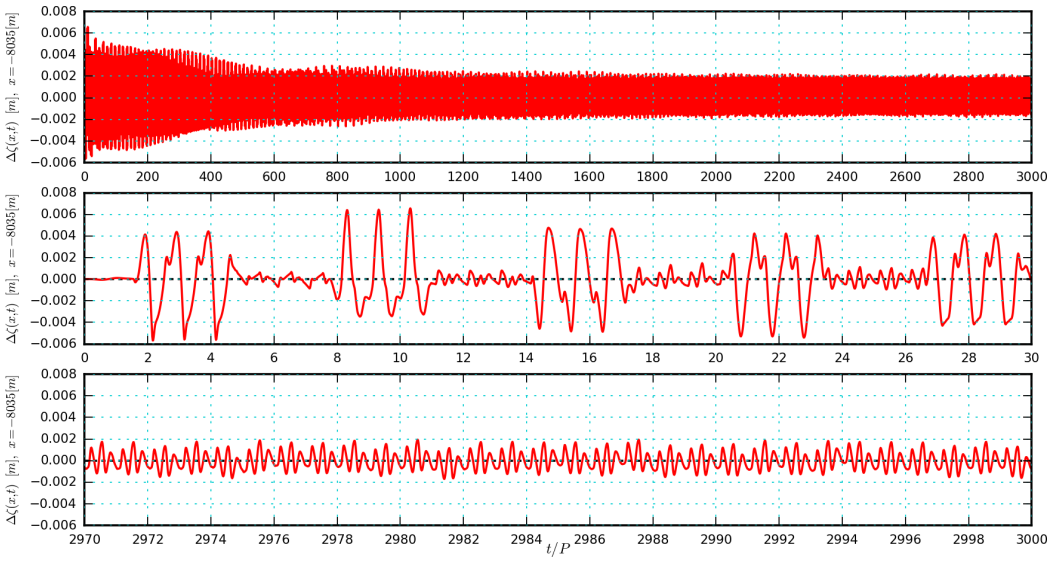


Figure 15.34: *Time series of the water-level residual $\Delta\zeta(x,t)$ at $x = -8035$m for D-Flow FM results computed on the one but finest spatial grid $\Delta x = 2$m with time step $\Delta t = 0.05$s. In the upper panel the time series of $\Delta\zeta(x,t)$ is shown for the entire simulation period $T = 3000P$. In the middle and lower panel the time series is shown for the first 30 periods P and the last 30 periods P.*

The top panel of Figure 15.34 shows a mix of variations at different time scales. At $t = 1000P$ the time series for the grid size $\Delta x = 2$m has clearly not yet convergenced to a stationary behaviour. In fact, only for times t larger than about $2000P$ the largest fluctuations seem to have disappeared. From a similar inspection of the time series computed with $\Delta x = 5$m it was deduced that the simulation time should at least be about $1400P$. Extrapolating these estimates we find that a simulation time of at least $4000P$ will be required to reach a stationary regime on the grid $\Delta x = 1$m. Such a long simulation time would require an excessive amount of computational time[16]. It is for this reason that simulations on the fine grid with $\Delta x = 1$m have been limited to a time of $T = 1000P$.

As mentioned before, the only dissipation present in the computational D-Flow FM model is numerical: a small dissipation in time by setting time-integration parameter θ to a value slightly larger than $1/2$ ($\theta = 0.55$ has been used), and a small dissipation in space due to the use of upwind on fine to very fine grids. Because of the use of a very small time step, the temporal numerical dissipation is negligible compared to the spatial numerical dissipation, which scales with the grid size. This explains why on finer grids a larger simulation time is necessary for the solution to reach a steady periodic behaviour. The time that is required turns out to be roughly inversely proportional to the applied Δx. This is consistent with the use of first-order upwind.

Altogether the following simulation times T have been chosen in the analysis of the accuracy of D-Flow FM simulations for the Carrier-Greenspan validation case as a function of time step Δt and grid-cell size Δx:

- ◇ The *analysis of the effect of the time step* is carried out for the coarsest grid with cell size $\Delta x = 10$m with the simulation time T set to $1000P$. For comparison, the time-

[16] This computational time is estimated to be about 4 days on a HP personal computer with an Intel Core(TM)i7 processor, and CPU@2.40 GHz.

step analysis has also been carried out for the finest grid with $\Delta x = 1$m, also with $T = 1000P$. Formally the simulation time should have been much longer for this grid but this was not done for the reason mentioned above (prohibitively long computational times).

◇ The *analysis of the effect of the grid size* is carried out with time step $\Delta t = 0.05$s. The D-Flow FM models with cell size $\Delta x \in \{2, 5, 10\}$[m] have been run for a simulation time of $T = 3000P$. The model with the smallest grid size $\Delta x = 1$m is also included in the analysis, but this model has only run for a simulation time of $1000P$.

As mentioned before, each time the $\Delta\zeta$ and Δu of the last 30 periods are used in the evaluation of the error norms.

Results of time-step variations

To begin with, the effect of the time-step size on the accuracy of the computed resultst has been verified using the coarsest spatial grid with $\Delta x = 10$m. On the basis of the Courant stability condition and an expected maximum velocity of about 3m/s (according to the analytical model, cf. Equation (15.32)), the time step must be less than about 3s on this grid. For this time step the D-Flow FM simulations turned out to be unstable. Due to the large overprediction of the velocity at and near the dry-wet interface, Δt had to be 1.5s or less. Hence D-Flow FM computations were carried out for $\Delta t \in \{0.05, 0.1, 0.2, 0.5, 1.0, 1.5\}$s. As explained before a simulation time of $T = 1000P$ was adopted in all simulations.

Next the error norms were computed for the 7 sub-domains A_i of the computational domain. The result is graphically shown in Figure 15.35. From this figure it is seen that for all norms and for all sub-domains (and for both water level and velocity) the errors hardly, if at all, decrease with the time step as soon as Δt is less than 1s. Clearly, the maximum time step for which stable solutions are still obtained is already small enough for the time integration error to be (close to) negligibly small compared to the space discretisation error. The conclusion reads that for this application, the stability constraint on the time step is larger than the accuracy constraint.

The variation of the error over the sub-domains, and thus over the spatial region, is quite large. This variation is monotone; the errors increase towards the beach. At the beach (sub-domain A_7) the errors are about 10 times larger than near the seaward boundary (sub-domain A_1) for the water level, and even much larger for the velocity. This points at considerable errors in the computed velocities at and near the dry/wet interface. This is likely to be caused by the very low water depth at these locations, as a result of which the velocity is not well defined (notice that the depth-integrated velocity or mass flux $q := (\zeta + D) \cdot u$ must converge to zero at the shoreline position). In other words, errors in a small water depth tend to lead to large errors in the velocity. Through back propagation of these errors the solution in the other sub-domains is affected. Because of the larger depth, and the prescribed exact condition at the left boundary, the magnitude of these errors decreases in seaward direction. This explains the observed A_i–dependency of the error norms.

The same error behavior has been observed in a similar time-step study for the finest grid with $\Delta x = 1$m. Again D-Flow FM simulations over a time of $T = 1000P$ were carried out. As shown before, formally this length should have been longer since the numerical solution is not yet fully stationary after $1000P$ on such a fine grid. The required much longer computations are practically not feasible, however.

For the grid with $\Delta x = 1$m no stable predictions were obtained for time steps larger than 0.2s. D-Flow FM simulations were therefore performed for $\Delta t \in \{0.05, 0.1, 0.15, 0.2\}$s.

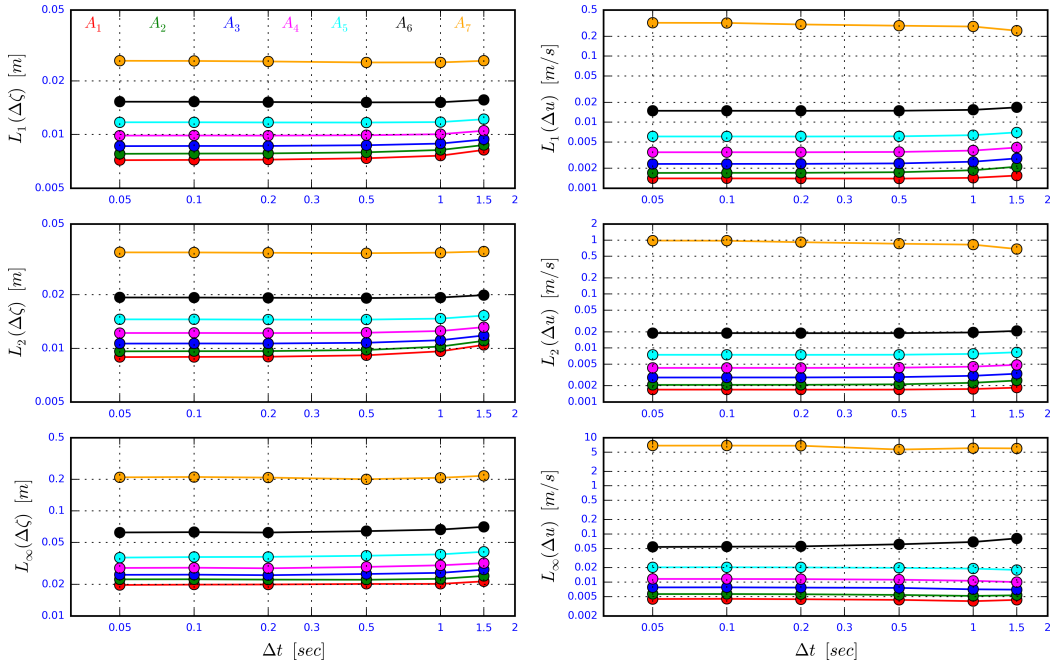


Figure 15.35: *Dependency of the error norms $L_1(\cdot)$, $L_2(\cdot)$ and $L_\infty(\cdot)$ on Δt for water depth and velocity in the 7 sub-domains A_i that were selected to validate the accuracy of the numerical scheme. These norms are as obtained for the presently coarsest grid with $\Delta x = 10$m. The simulation time T consisted of 1000 temporal periods P of 5.9044minutes. The $\Delta\zeta$ and Δu of the last 30 periods were used in the computation of the various error norms.*

The error norms that were obtained for these time steps are shown in Figure 15.36 for the 7 sub-domains A_i considered.

This time a very weak sensitivity of the norms on the time step was found, confirming the conclusion that for this application the stability constraint on the time step is larger than the accuracy constraint. The velocity error norms are virtually constant over the time steps, and different for the different sub-domains of course. As for the water depth we find that the error norms tend to *increase* somewhat for *decreasing* time steps. It seems that in this application and with this grid size, time integration errors and space discretisation errors partially cancel against each other, whence a larger overall error for smaller time steps when the time integration error and hence the cancelling effect becomes smaller.

Like for the case with $\Delta x = 10$m a monotone increase of the error norms over the sub-domains A_i is found. A comparison between the results for $\Delta x = 1$m with those for $\Delta x = 10$m shows that the magnitude of the error norms have decreased uniformly. Comparing Figure 15.36 with Figure 15.35 it is readily seen that this decrease is much less than the factor 10 of the grid refinement.

Because the choice of the time step is dominated by the stability constraint, no conclusion can be drawn as to the accuracy in time of the numerical scheme. Formally a first-order error behaviour would have been expected (error proportional to Δt), because of the first-order explicit discretisation of the convection terms and the use of a θ–value larger than $1/2$ ($\rightarrow$ first-order accurate implicit time integration of the pressure term of the momentum equation).

To conclude we mention that the results of this accuracy analysis justify the use of the time step $\Delta t = 0.05$s in the simulations of the previous paragraph where we investigated the

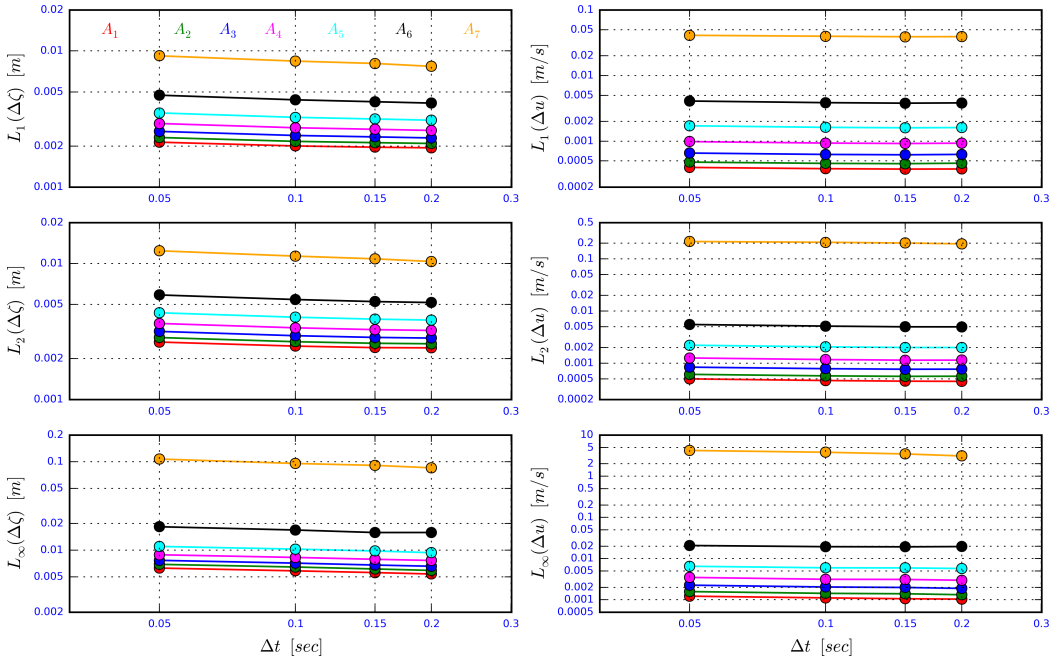


Figure 15.36: *Dependency of the error norms $L_1(\cdot)$, $L_2(\cdot)$ and $L_\infty(\cdot)$ on Δt for water depth and velocity in the 7 sub-domains A_i that were selected to validate the accuracy of the numerical scheme. These norms are as obtained for the presently finest grid with $\Delta x = 1$m. The simulation time T consisted of 1000 temporal periods P of 5.9044minutes. The $\Delta\zeta$ and Δu of the last 30 periods were used in the computation of the various error norms.*

effect of the length of the simulation time.

Results of grid size variations

The accuracy of D-Flow FM with respect to the spatial discretisation has been verified in a similar way as described above for the time step. To ensure that errors due to the time integration can safely be ignored, only results from simulations with the smallest time step $\Delta t = 0.05$s have been considered. For the grids with $\Delta x \in \{2, 5, 10\}$m the length of the simulation time has been taken equal to $T = 3000P$. To avoid exceptionally long computational times (as explained above) this length was limited to $T = 1000P$ for the grid with a cell size of 1m.

The error behaviour in the various norms is shown in Figure 15.37. For the sub-domains A_1 to A_6 we see roughly the same behaviour of the error norms as a function of the spatial grid resolution: a gradually decrease when the cell size Δx becomes smaller. The largest error decrease is in the step where Δx is reduced from 10 to 5m. The relative error decrease is less when the cell size is reduced further. The decrease is larger for the L_1 and L_2 norms than for the L_∞ norm (in fact, the L_1– and L_2–norm errors have very comparable magnitude). For sub-domain A_7 the dependency on the cell size Δx is less clear. For example, for the water depth the steepness in the reduction of the error when Δx decreases is less on A_7 than on the other sub-domains. For the velocity, however, this reduction tends to be more in line or is even slightly higher on A_7 than on the other 6 sub-domains.

Just as in the sensitivity analysis for the time step, it is found that (in all norms, and for both water depth and velocity) the magnitude of the errors increases monotonously in the direction of the beach. In A_7, i.e. the sub-domain containing the region with drying and flooding, the magnitude of the errors is about 5 (water depth) to over a 100 (velocity) times larger than in

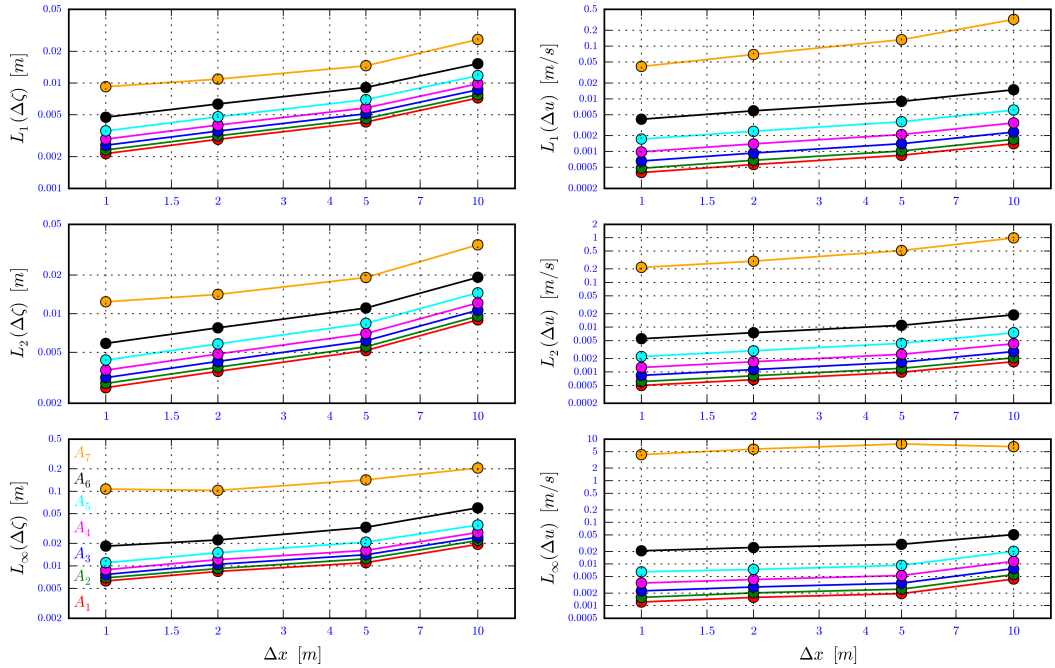


Figure 15.37: *Dependency of the error norms $L_1(\cdot)$, $L_2(\cdot)$ and $L_\infty(\cdot)$ on Δx for water depth and velocity in the 7 sub-areas A_i that were selected to validate the accuracy of the numerical scheme. These norms are as obtained for the presently smallest time step $\Delta t = 0.05$s. For the grid sizes $\Delta x \in \{2, 5, 10\}$m the D-Flow FM simulation time T consisted of 3000 temporal periods P of 5.9044minutes. For the grid with $\Delta x = 1$m the D-Flow FM computation was limited to $T = 1000P$. The $\Delta\zeta$ and Δu of the last 30 periods were used in the computation of the various error norms.*

the other sub-domains. Due to the very small water depths in A_7, the computation of the velocity is very sensitive to errors in that sub-domain. As a result, velocity errors are always quite large in A_7; they reach values up to 5m/s (which is more than the analytical maximum velocity, cf. Equation (15.32)) while they are less than 5cm/s in the other areas.

For a proper quantitative assessment of the order of accuracy, the effective order of accuracy of the computational results has been determined by means of linear regression analyses, i.e. by fitting functions linear in the logarithm of the grid size to the logarithms of the errors. Those linear functions, of the form $a_k \cdot \log(\Delta x) + b_k$ with a_k the order of accuracy (and b_k the error level), approximate the error plots on log-log scale shown in Figure 15.37. The errors on the coarsest grid have been discarded in this analysis because of the systematically different slope between the errors on the grids $\Delta x = 5$m and $\Delta x = 5$m (cf. Figure 15.37), for which we have no explanation. Apparently, a grid size of $\Delta x = 10$m (which in this application gives a high spatial resolution) is not fine enough to have a consistent order-of-accuracy behaviour. NB, the results on the finest grid $\Delta x = 1$m have been included, despite the fact that these are based on a simulation time $T = 1000P$ that is actually too short.

The slopes a_k determined in the regression analyses provide an estimate of the effective order of accuracy of D-Flow FM's spatial discretisation for this application. For both water depth and velocity they have been computed for all 3 times 7 combinations of error norm (L_1, L_2 and L_∞) and sub-domain (A_1 to A_7). The obtained order-of-accuracy results are graphically shown in Figure 15.38.

From this figure it is seen that the order of accuracy is virtually constant over the sub-domains

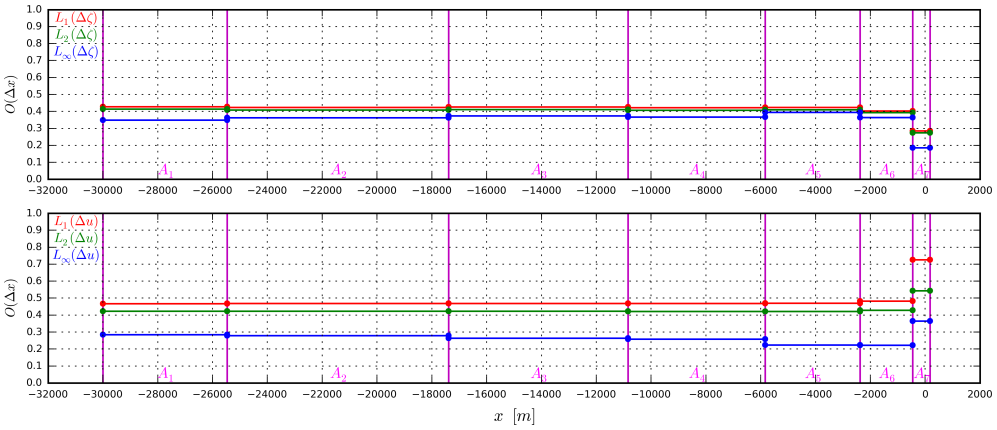


Figure 15.38: *The spatial order of accuracy (coefficient a_k in the error behavior $O(\Delta x^{a_k})$) of water level (upper panel) and velocity (lower panel) for the various error norms $L_k(\cdot)$ in the 7 sub-domains A_i. The orders of accuracy have been derived from D-Flow FM computational results using the presently smallest time step $\Delta t = 0.05$s and $\Delta x \in \{1, 2, 5\}$m. In the computation with $\Delta x = 1$m the simulation time was $T = 1000P$; in the other computations $T = 3000P$ has been used.*

A_1 to A_6. In view of the continuity and smoothness of the solution in these areas this was to be expected. In these areas both the water-depth error and the velocity error show a 0.4 to 0.5 order-of-accuracy behaviour in the average norms L_1 and L_2. This is much less than the order of (at least) 1 that would have been expected from D-Flow FM's spatial discretisation. The reason for this large discrepancy is not known. It is not known either why, given the smoothness of the solution in the sub-domains A_1 to A_6, the order of accuracy in the maximum norm L_∞ is lower than in the average norms, especially for the velocity where the order of accuracy in the L_∞–norm is only 0.3 to 0.2.

In sub-domain A_7 the water-level order of accuracy is lower than in the other sub-domains. In the L_∞ norm it is even less than 0.2. This very low order of accuracy may be due to relatively large grid-independent errors in the prediction of the position of the dry-wet interface moving back and forth, also leading to significant error *levels* in water level and velocity. In contrast to the water level, for the velocity in A_7 a higher order of accuracy is found than in the other sub-domains. In the L_1 norm this order is even more than 0.7. It must be realized however, that this gain in order of accuracy comes with much larger error *levels* in A_7, cf. the orange curves in the right panels of Figure 15.37.

We have not investigated to what extent the errors in drying-and-flooding zone A_7 depend on the parameters `chkadvd` and `epshu` of the drying-and-flooding procedure. We have not investigated either if these errors would diminish in the presence of a physical dissipation mechanism, in particular bottom friction (in which case an analytical solution for comparison is no longer available).

Accuracy of the prediction of the shoreline position

In this paragraph the results are presented of an analysis on the accuracy of D-Flow FM's prediction of the dry/wet interface. The reported error behaviour provides an indication of the quality of D-Flow FM's drying and flooding procedure. The approach that has been followed is virtually the same as the one used above in the assessment of the effect of the grid size on the errors in the flow solution (water level and velocity). The error $\Delta x_{DW}(\cdot)$ in the prediction of the shoreline position has been computed in the same error norms L_1, L_2 and L_∞ as used

before.

The shoreline prediction error is defined by $\Delta x_{DW}(t_m) := x_{DW,A}(t_m) - x_{DW,C}(t_m)$, with $x_{DW}(t_m)$ the horizontal position of the dry/wet interface at time t_m. Subscript A refers to the analytically derived x_{DW}, while subscript C indicates the shoreline location as predicted by D-Flow FM. In the computation of the error norms the results of the same 4 simulations have been used as in the preceding error analysis where the effect of the grid size was studied. We recapitulate that these are the simulations with time step $\Delta t = 0.05$s and the spatial grids with $\Delta x \in \{1, 2, 5, 10\}$m. The simulation time is $1000P$ for the finest grid, and $3000P$ for the other coarser grids. Again the 240 time points $t_m := m \cdot P/8$ in the last 30 periods P of 5.9044 minutes of a simulation were used in the evaluation of the error norms $L_k(\Delta x_{DW})$.

The analytical solution $x_{DW,A}(t)$ for a given t can be derived from Equation (15.21) to Equation (15.24) with a similar inversion of the hodograph transformation as described before. Again this inversion has been carried out by minimisation of a cost function (cf. Equation (15.25), but now with the constraint that $\sigma = 0$). For the determination of the shoreline position $x_{DW,C}(t)$ computed by D-Flow FM the following procedure was used. Given the flow variables ζ and u in all grid cells at some time t, the dry-wet interface at time t, $x_{DW,C}(t)$, is considered to be the right face of the first wet cell encountered when moving from the right model boundary on the beach (which is always dry) to the left boundary at sea (which is always wet). In other words, $x_{DW,C}(t)$ is considered to be the interface between a wet cell and a dry cell that is highest up the slope.

Since this procedure determines the shoreline position up to a deviation of Δx, the accuracy of this estimate for $x_{DW,C}$ is first order in the grid size. In consequence the order of accuracy of Δx_{DW} cannot be higher than first order in Δx. On the other hand, the criterion that is applied in D-Flow FM to decide on the dry/wet status of a cell is of similar accuracy: a cell face x_n is considered to be dry at time t_m if the total water depth $\zeta(x_n, t_m) + D(x_n)$ at that face is less than `epshu`, in which case it is set to zero, thereby closing the face and the flow access to an entire grid cell. Because of the latter, the accuracy of the D-Flow FM drying and flooding procedure cannot be higher than first order in Δx.

The effect of grid size Δx on the quality of the D-FLOW FM prediction of the shoreline will be illustrated by means of a few figures. In Figure 15.39 time series of $x_{DW}(t)$ are presented, both for the analytical solution and as determined from the D-Flow FM computational results. For a proper view on the quality of the D-Flow FM results only the last three periods P of the simulations are shown. For the finest grid with $\Delta x = 1$m this is thus the time interval $t = 997P$ to $t = 1000P$. For the other, coarser grids this is the time interval $2997P$ to $3000P$. To enable a comparison of the results, the data points of the finest grid have been plotted with a time shift of $2000P$ to the right.

From Figure 15.39 it is readily recognised that, as expected, the largest mismatch between the computed and the analytical shoreline occurs on the coarsest grid with $\Delta x = 10$m. There is a clear temporal structure in this mismatch, in the sense that the largest errors are found immediately after the point of time where $x_{DW,A}$ is at its maximum. This is thus at the moment that, after its maximum excursion on the beach, the wave retreats back to sea. For the computation with $\Delta x = 10$m the drying of the beach is then highly retarded compared to the analytical solution. To a lesser extent this also occurs for $\Delta x = 5$m. Further reduction of the grid size gives correspondingly better predictions of x_{DW} in the drying phase. In all cases the flooding of the beach is reasonably well reproduced, even on the coarsest grid.

The quality of the D-Flow FM simulation of the run-up/run-down process can visually be summarized by presenting the results shown in Figure 15.39 in a different way, namely by plotting, for the 8 moments in time of the last period considered, the computed location of the shoreline

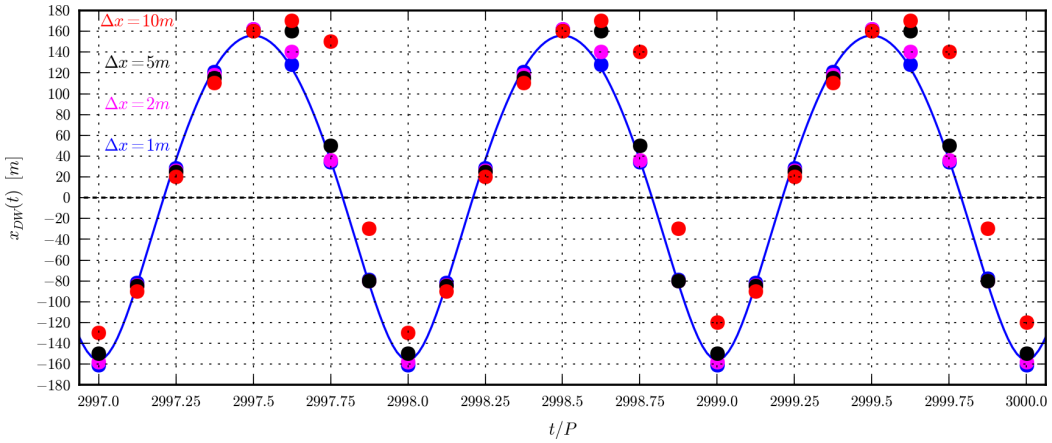


Figure 15.39: *Time series of the location x_{DW} of the dry/wet interface according to the analytical solution (solid curve in blue), and as computed by D-Flow FM (coloured dots) for various spatial grid sizes Δx. They are shown here for the last three periods P of the simulation time. This simulation time is $T = 3000P$, except for the finest grid with $\Delta x = 1m$ where $T = 1000P$.*

$x_{DW,C}$ against the analytical value $x_{DW,A}$. The result is shown in Figure 15.40 for the four grids considered.

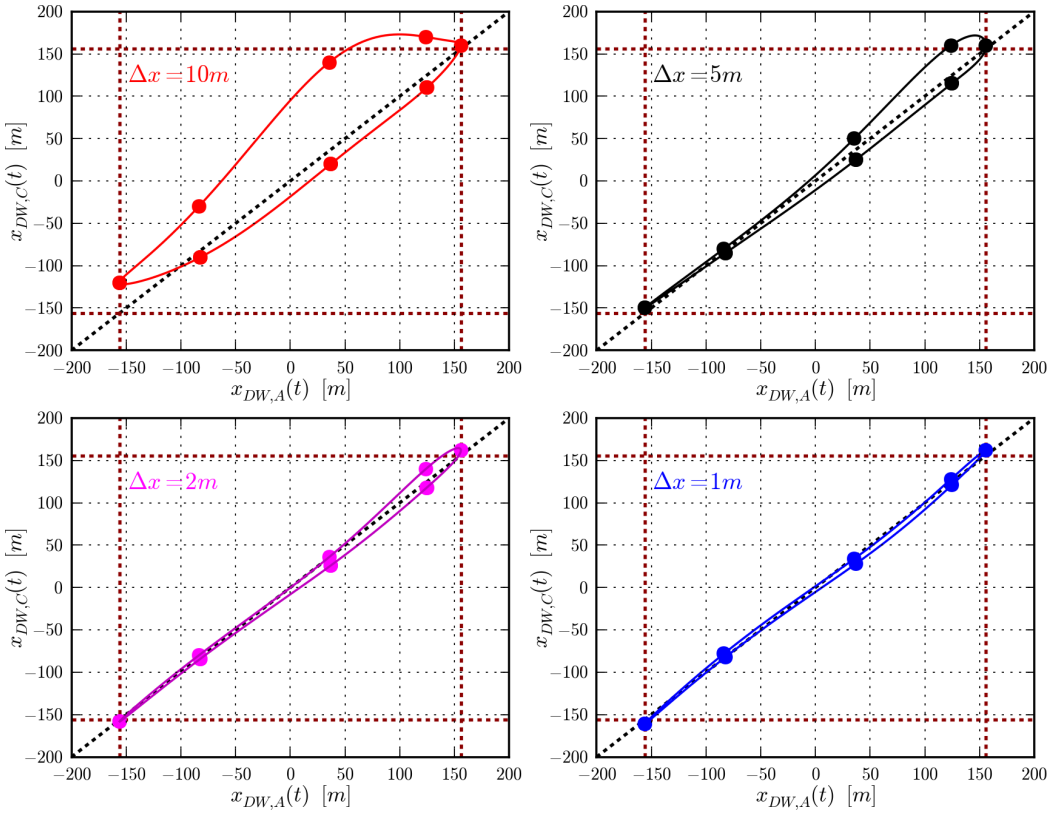


Figure 15.40: *Computed locations $x_{DW,C}$ of the dry/wet interface plotted against the corresponding analytical values $x_{DW,A}$, for various spatial grid sizes Δx and for the last period P of the simulation time. This simulation time is $T = 3000P$, except for the finest grid with $\Delta x = 1m$ where $T = 1000P$. The run-up/run-down process is counterclockwise along the smooth closed fit through the data points.*

To facilitate the interpretation of the results, a smooth closed fit through the data points has been added. The drying-and-flooding process as predicted by D-Flow FM is counterclockwise along these curves. From the fact that the curves in upward direction from the lower-left corner to the upper-right corner closely follow the diagonal, it becomes immediately clear that the run-up is accurately predicted. This confirms the conclusion that the flooding of the beach is reasonably well reproduced by D-FLOW-FM. In the other direction there is, for the coarser grids, a strong mismatch between the computational results and the analytical solution that takes the form of a delay of the run-down. The mismatch vanishes on the finer grids. Notice that with a grid size of $\Delta x = 1$ or 2m, the drying-and-flooding area of 300m is very well resolved. In practice such high resolutions will seldomly be possible. In fact, one would normally consider the drying-and-flooding area to be well resolved with a few dozen grid cells in that area. The present validation case indicates that that may not be sufficient.

The results show that in particular the run-down speed tends to be underestimated by D-Flow FM. This may be a consequence of the drying procedure applied in D-Flow FM, where water is taken out of the computation when the total water depth gets below threshold value `epshu`. With less water running down a slope, the run-down acceleration is likely to be underpredicted. We see this also from the velocity error in the lower panel of Figure 15.31. In the first half of each period (the run-up phase), the error is mainly negative, indicating an overprediction of the velocity by D-Flow FM. In the second half of each period, the twice larger positive error indicates a stronger underprediction of the run-down velocity. The maximum errors of about -0.06m/s and 0.10m/s are at this location $x = -300$m respectively 3.5% and 5.9% of the analytical maximum velocity $u = \mp 1.7$m/s, cf. Figure 15.26. Another, more likely candidate for the cause of the underprediction of the run-down acceleration is the D-Flow FM upwind discretisation of the convection terms. The upstream velocity values that it uses are in the case of run-down located in the dry area and hence not well defined. It is not clear which approximation procedure/extrapolation is followed here in D-Flow FM. It could be that it leads to a rather inaccurate D-Flow FM convection discretisation in the case of drying/run-down.

Notice that because of the absence of a physical dissipation mechanism, the simulation of run-up and run-down in the present test is very sensitive to discretisation errors. It is to be expected that the presence of some bottom friction (in practice always the case), which due to the very small water depths has a large effect in run-up/run-down areas, will considerably reduce the effect of errors in the convection terms. We have not investigated if an exact run-up/run-down solution may be constructed for the shallow-water equations with friction term to study this in more detail. It may be possible to do so if the bottom friction term is kept linear in the velocity.

Figure 15.41 gives a close view on the magnitude and temporal structure of the errors Δx_{DW}. In particular it is seen that on the finest grids the errors are considerably smaller than on the coarsest grids, and also tend to evolve more randomly in time rather than systematic.

The error norms that were derived from the errors depicted in Figure 15.41 are shown in Figure 15.42. In addition to the three norms L_1, L_2 and L_∞ (in the upper three panels), a fourth quality indicator is shown in the lower panel. This fourth indicator is the (absolute value of the) standard mean of the error Δx_{DW}, i.e. $|\frac{1}{M}\sum_{m=1}^{M}\Delta x_{DW}(t_m)| = |\frac{1}{M}\sum_{m=1}^{M}\{x_{DW,A}(t_m) - x_{DW,C}(t_m)\}|$. This quantity provides a measure for the systematic error (or bias) in D-Flow FM's prediction of the time-varying shoreline.

The plots of all three L-norms show a similar behaviour: a somewhat stronger decrease of the error in the first refinement step where the grid size is reduced from $\Delta x = 10$m to $\Delta x = 5$m than in the next steps where the spatial grid is further refined. In that first step the errors seem

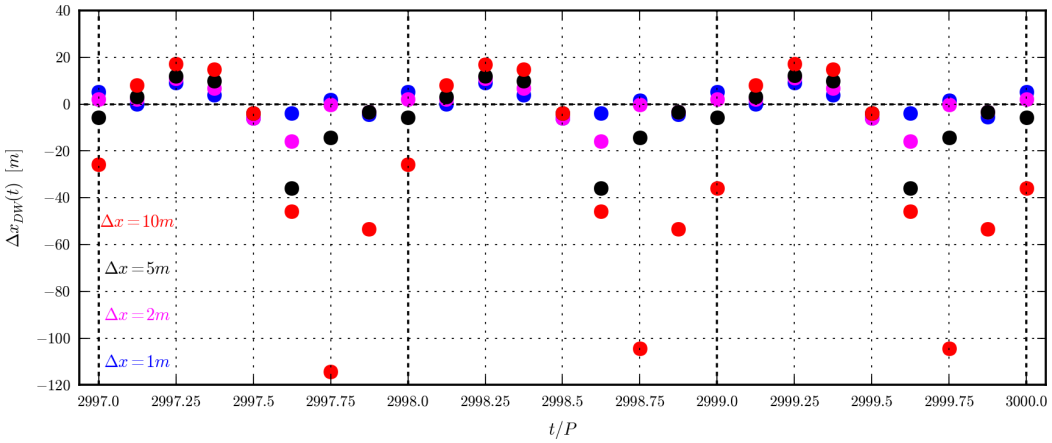


Figure 15.41: *Time series of the error $\Delta x_{DW}(t_m) := x_{DW,A}(t_m) - x_{DW,C}(t_m)$ in D-Flow FM's prediction of the location x_{DW} of the dry/wet interface for various spatial grid sizes Δx. They are shown here for the last three periods P of the simulation time. This simulation time is $T = 3000P$, except for the finest grid with $\Delta x = 1$m where $T = 1000P$.*

to have been reduced with a factor of at least 2, suggesting a first-order dependency on Δx. In the other steps the factor of error reduction is notably smaller than the factor of grid-size reduction, and the order of accuracy with respect to Δx is less than one.

To verify this more quantitatively a linear regression $\log(L_k(\Delta x_{DW})) \sim a_k \cdot \log(\Delta x) + b_k$ has been carried out. The slopes a_k represent the order of accuracy in the prediction of the shoreline position with respect to the grid size. The data points for the coarsest grid were again excluded in this regression analysis because of the difference in Δx-sensitivity between the errors around $\Delta x = 10$m and around the Δx of the finer grids.

The estimates that were found for the orders of accuracy are listed for each norm in the upper-left corner of the corresponding panel in Figure 15.42. They vary from 0.572 for the average norm L_1 to 0.808 for the maximum norm L_∞, and are thus considerably less than one. While for L_∞ the largest order of accuracy is found, it must be noted that even on the finest grids the magnitude of that error is quite large. As can be seen from the blue curve in the bottom panel, the error in the prediction of the shoreline position does not show a clear sensitivity to grid refinements.

The bottom panel of Figure 15.42 shows that on the two finest grids the position of the dry/wet interface is well predicted by D-Flow FM; the mean of Δx_{DW} turns out to be less than 1m on these grids. On the coarsest grid the systematic error is considerable: about twice the grid size. A large error on the coarsest grid in the prediction of the shoreline position was already observed in Figure 15.39, Figure 15.40, and Figure 15.41.

Finally it is noted that for the coarsest grids the presently identified errors in the shoreline prediction are also quite large in *relative* sense. That is, when compared to the length of the beach section that is repeatedly subjected to drying and flooding. This length is about 300m and corresponds to 30 grid cells of the coarsest grid. With the present constant and mild slope of the bed (1/100), and not excessively high water levels and velocities in the coastal region, the drying and flooding region is then reasonably densely covered by the grid. An absolute mean error of about 50m (according to L_1 and L_2), and a maximum error of about 100m (according to L_∞) are then as large as 20 to 30% of the drying and flooding section. To increase the accuracy, the grid density on the beach has to be increased considerably.

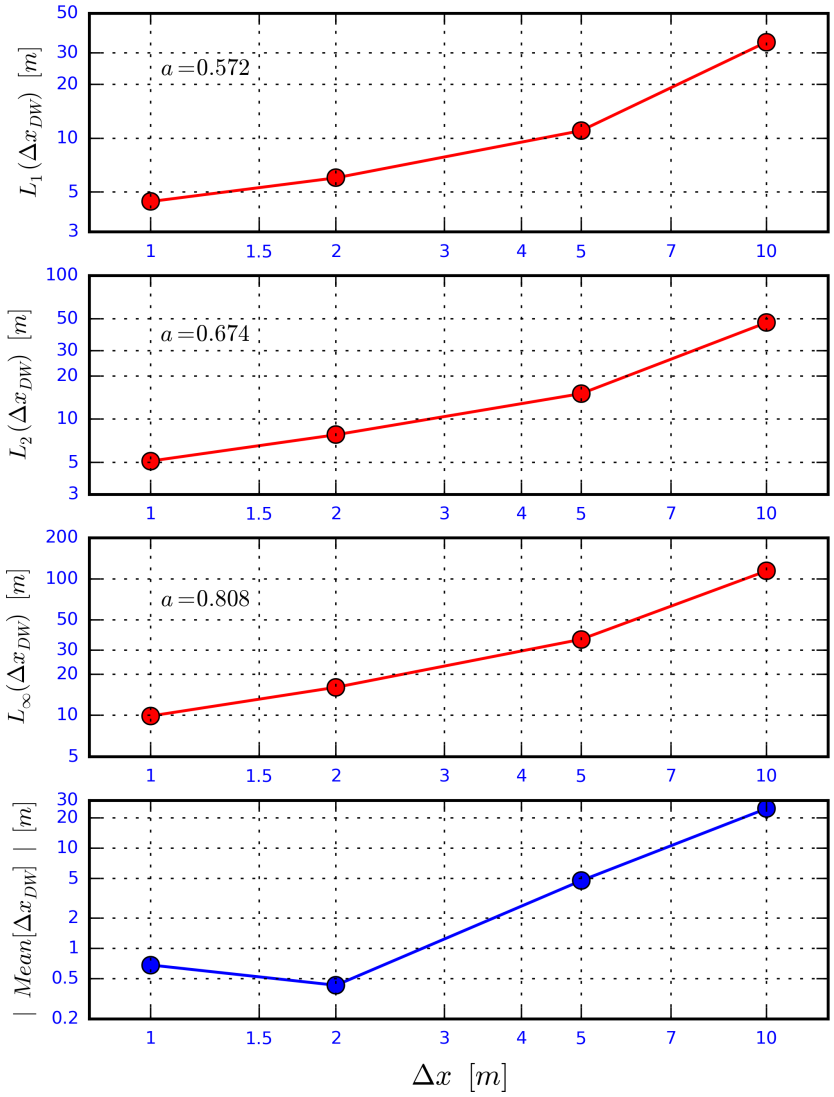


Figure 15.42: *Dependency of the error norms $L_1(\cdot)$, $L_2(\cdot)$ and $L_\infty(\cdot)$ on Δx for D-Flow FM's prediction of the temporally varying position $x_{DW}(t)$ of the dry/wet interface at the beach. These norms are as obtained for the presently smallest time step $\Delta t = 0.05$s. For the grid sizes $\Delta x \in \{2, 5, 10\}$m the D-Flow FM simulation time T consisted of 3000 temporal periods P of 5.9044minutes. For the grid with $\Delta x = 1$m the D-Flow FM computation was limited to $T = 1000P$. The Δx_{DW} of the last 30 periods were used in the computation of the various error norms.*

Conclusion

The main conclusions of the present D-Flow FM validation study can be summarized as follows.

- ◇ In agreement with Carrier and Greenspan's analytical solution the D-Flow FM computations were carried out without friction and horizontal viscosity, and with a periodic boundary condition for the water level. The simulation time required to reach a periodic-stationary regime of the numerical solution turned out to be extremely long. This allows the conclusion that the numerical dissipation in D-Flow FM is limited. From the computations with the different grids it was found that the smaller the cell size Δx the longer a simulation is required to reach stationary conditions. This is consistent with the fact that the numerical dissipation is proportional to the grid size and hence must be due to the numerical

viscosity introduced by the upwind discretisation of the convection terms.

- ◇ In view of the applied time integration scheme, a first-order error behaviour with regard to the time step Δt would be expected. For the present case this behaviour could not be verified. Only a very small sensitivity of the accuracy of the computed solution on the time step was found. The explanation for this observation must be that, because of the rather large time scale of the problem, the time-step restriction due to the stability constraint is much larger than the one due to the accuracy constraint.
  On the finest grid, and for all sub-domains of the computational domain, the errors were found to *increase* (slightly, and monotonously) when *reducing* the time step. This could be due to the fact that space discretisation errors and time discretisation cancel to some extent, as a result of which the overall error increases as the time step (and hence the time discretisation error) decreases. Why this would only happen on the finest grid is not clear.
- ◇ In the present flow simulations the D-Flow FM modelling errors are virtually completely due to the spatial discretisation.
- ◇ In all norms the accuracy with regard to the spatial discretisation was found to be merely of order 0.4, even on sub-domains of the computational domain where the (analytical) solution depend very smoothly on the temporal and spatial coordinates. For the present application D-Flow FM's spatial accuracy is thus considerably less than first order. In the sub-domain containing the drying and flooding (as induced by the repeated wave run-up and run-down) the order of accuracy is somewhat lower for the water level, and somewhat higher for the velocity. In absolute sense, however, the water-level errors and velocity errors in the run-up/run-down sub-domain are considerably higher than elsewhere.
- ◇ The accuracy of D-FLOW FM's implementation of drying and flooding has been verified on the basis of the error in D-Flow FM's prediction of the (time-dependent) spatial position of the shoreline, i.e. the dry/wet interface at the beach. Depending on the applied error norm, the accuracy with respect to the spatial discretisation turned out to be of order 0.6 to 0.8. In the L_1 norm and for the grids considered (*not* in general), the error is roughly three times the cell size Δx, while in the L_∞ norm it is about 10 times Δx. The errors in D-Flow FM's prediction of the position of the dry/wet interface are largest at the beginning of the drying phase, i.e. immediately after the moment of maximum wave run-up, when the water starts retreating from the beach again. The error in this phase is systematic, in the sense that D-FLOW FM's drying of the beach is retarded compared to the analytical solution. A very high spatial resolution is needed to reduce this systematic error to a fairly small level.
- ◇ Practically speaking (and apart from the flow in the direct vicinity of the dry/wet interface) D-Flow FM is well capable to reproduce the Carrier and Greenspan analytical solution for the wave run-up and run-down on a beach with constant slope: in absolute value the model errors are reasonably small, at the condition (and at the cost), however, of a very fine spatial grid.

It is important to notice that, due to the absence of in particular bottom friction, the Carrier and Greenspan test case is not fully representative of the simulation of drying and flooding in practice. It is to be expected that the presence of bottom friction may facilitate the simulation of drying and flooding in practical applications. In the present frictionless application we have to deal in the very shallow drying-and-flooding area with repeated run-up and run-down, with the undamped simulation of significant non-linear dynamics. On the other hand, although the presence of bottom friction may reduce the errors in the run-up/run-down sub-domain, it remains to be investigated if this will improve the rather poor (order of) accuracy in the other sub-domains.

It must be emphasized that the present results and conclusions have been obtained with uniform rectangular spatial grids. It is fair to assume that the use of less uniform grids, and

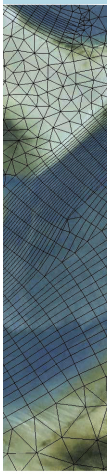
especially the use of irregular unstructured grids, will lead to larger errors.

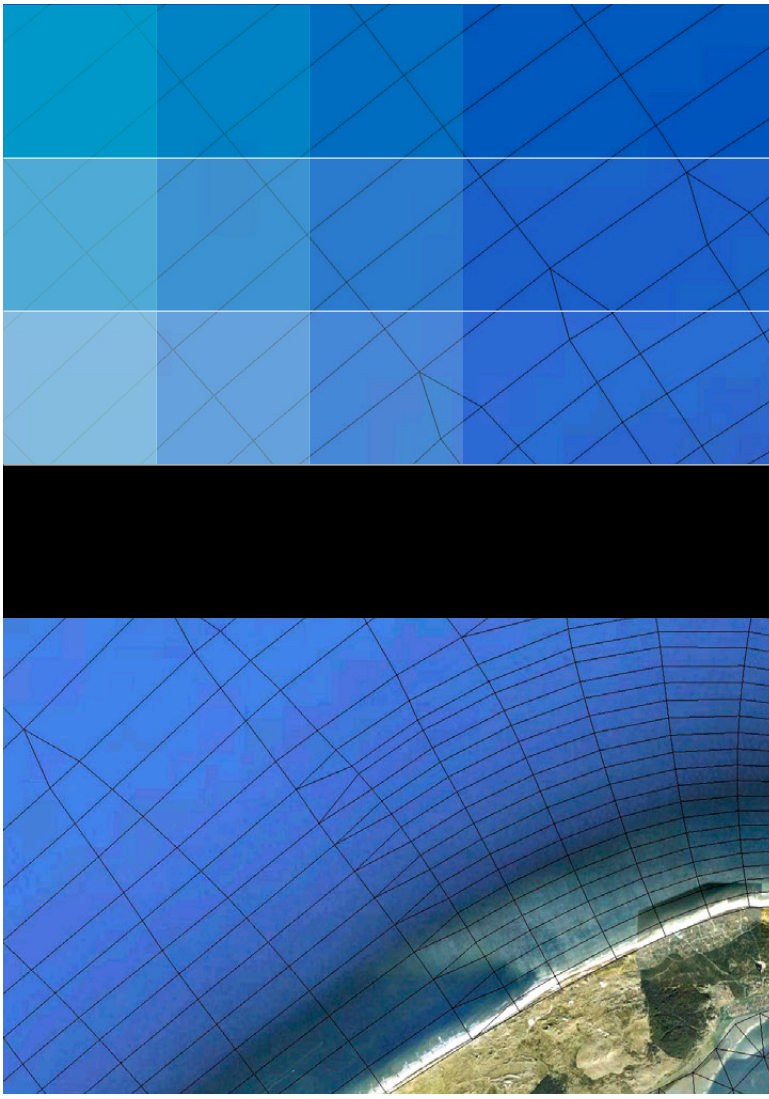
Version

This test has been carried out with D-Flow FM version 1.1.137.39598MS.

References

Dee, D., International Association of Hydrological Research, and Association Internationale de Recherche Hydraulique (1994). *Guidelines for Documenting the Validity of Computational Modelling Software*. IAHR/AIRH. URL: https://books.google.nl/books?id=1NLGPAAACAAJ.





Deltares systems

PO Box 177
2600 MH Delft
Boussinesqweg 1
2629 HV Delft
The Netherlands

+31 (0)88 335 81 88
software@deltares.nl
www.deltares.nl/software