A fully implicit wetting-drying method for DG-FEM shallow water models, with an application to the Scheldt Estuary

Tuomas Kärnä[a,b,*], Benjamin de Brye[a,b], Olivier Gourgue[a,b], Jonathan Lambrechts[a,b], Richard Comblen[a,b], Vincent Legat[a,b], Eric Deleersnijder[a,c]

[a] *Université catholique de Louvain*
Institute of Mechanics, Materials and Civil Engineering (IMMC)
Avenue George Lemaître 4,
B-1348 Louvain-la-Neuve, Belgium
[b] *Université catholique de Louvain*
G. Lemaître Centre for Earth and Climate Research (TECLIM)
Chemin du Cyclotron 2,
B-1348 Louvain-la-Neuve, Belgium
[c] *Université catholique de Louvain*
Earth and Life Institute (ELI)
G. Lemaître Centre for Earth and Climate Research (TECLIM)
Chemin du Cyclotron 2,
B-1348 Louvain-la-Neuve, Belgium

Abstract

Resolving the shoreline undulation due to tidal excursion is a crucial part of modelling water flow in estuaries and coastal areas. Nevertheless, maintaining positive water column depth and numerical stability has proved out to be a very difficult task that requires special attention. In this paper we propose a novel wetting-drying method in which the position of the sea bed is allowed to fluctuate in drying areas. The method is implemented in a Discontinuous Galerkin Finite Element Model (DG-FEM). Unlike most methods in the literature our method is compatible with fully implicit time-marching schemes, thus reducing the overall computational cost significantly. Moreover, global and local mass conservation is guaranteed which is crucial for long term environmental applications. In addition consistency with tracer equation is also

*Corresponding author
Email address: tuomas.karna@uclouvain.be (Tuomas Kärnä)

ensured. The performance of the proposed method is demonstrated with a set of test cases as well as a real-world application to the Scheldt Estuary. Due to the implicit time integration, the computational cost in the Scheldt application is reduced by two orders of magnitude. Although a DG-FEM implementation is presented here, the wetting-drying method is applicable to a wide variety of shallow water models.

Keywords: shallow water equations; wetting-drying; discontinuous Galerkin; finite element method; implicit time integration

---

1. Introduction

Most coastal areas of continental shelf seas are significantly influenced by tides. When approaching the coast, the tidal signal tends to amplify, especially in funnel-shaped embayments where the tidal range may reach considerable magnitudes. Combined with the fact that many estuaries and embayments also feature gradually sloping bathymetry, the total area submerged under water may vary significantly during the tidal cycle.

Any hydrodynamical model that is being applied to such a tidally influenced domain needs to correctly take into account the sequential exposure and submerging of the seabed.

Ever since the 1970s hydrodynamical models have been equipped with wetting-drying (WD) algorithms. However, the multitude of WD methods found in the literature reveals that numerical modelling of shoreline undulation is far from being trivial. In what follows, we will mostly concentrate on methodology applicable to unstructured grid models, i.e. finite element (FE) and finite volume (FV) formulations.

Perhaps the most natural approach would be to track the WD interface in time and move the boundary nodes, or deform the entire mesh, accordingly. These moving mesh methods probably yield the most appropriate description of the wetting-drying process, but are faced with some difficulties: First of all, one needs to come up with a parametrisation for moving the domain boundary as a function of the flow in boundary elements [1, 2]. Secondly, sophisticated algorithms are needed for maintaining good mesh quality in long simulations. However, re-meshing becomes increasingly difficult if changes in the topography, such as emerging ponds or islands, need to be taken into account. Nevertheless, the greatest drawback is that deforming the mesh is

computationally expensive. For the latter reason, deforming mesh wetting-drying has seldom been applied to real-world problems [3, 4, 5].

Most of the available WD methods have been developed for fixed meshes. The fixed mesh approaches can be further sub-divided into two main categories. In the first category, either nodes or entire elements are deactivated when becoming dry. The dry state is detected with special criteria usually based on total water depth. The first attempts were element reduction methods, where entire elements are tagged as "dry" or "wet" and dry elements are excluded from the computational domain. This implies that the WD interface can only be located at element edges which has an impact on the boundary layer behaviour [6]. Moreover, the sudden inclusion/exclusion of elements may break mass and momentum conservation and also trigger numerical instability [7].

In order to describe a coastline that does not coincide with the element edges, most authors introduce transition (or partially wet) elements that require special treatment. By far the most popular method is to leave a thin layer of water in the dry areas to ensure positive water depth and maintain a continuum across the WD interface [6, 8, 9, 10]. The transition elements are then defined as elements for which some nodes, but not all are dry, i.e. have water level under a prescribed threshold value.

The main difficulty in thin layer methods lies in the way that the transition elements are being treated. The transition elements typically have "hanging nodes" that remain on a higher level than the free surface. In such case the transition elements have spurious water surface slope and thus experience an artificial pressure gradient [11] that tends to drive the water down. Without any treatment, the hanging nodes will eventually dry out. Commonly the transition elements are being explicitly detected and the pressure gradient term is being ignored. Sometimes the elements are further divided into "dam-break" and "flooding" types, where the pressure gradient term needs to be cancelled only in the latter case [6, 8]. Typically several more or less complicated rules are needed for detecting the transitory (or dry) elements and the local physics is changed accordingly. However, such discontinuous switches (such as cancelling the pressure gradient or outward fluxes under a threshold water depth) render these methods highly non-linear and may introduce oscillations and numerical instability. Indeed it is not uncommon to increase bottom friction or dissipation to circumvent such problems. Nevertheless, thin layer methods can produce accurate results even in very difficult applications.

The thin layer methods rely on explicit detection of dry elements, so the WD front can propagate only by one element per time step. Moreover, the time steps are usually constrained by the Courant–Friedrichs–Lewy (CFL) criterion, which is needed to ensure positive water depth or numerical stability [12]. Consequently the computational cost grows significantly, especially in large scale simulations that incorporate both deep and shallow areas [13]. Due to the time step constraints, explicit time integration is commonly used, although semi-implicit schemes also exist. Fully implicit time marching is generally not applicable as it requires that the Jacobian of the system can be computed (exactly or approximately), which is not possible for the discontinuous switches.

Another class of fixed grid WD methods is the artificial porosity approach. In these methods, the bed is assumed to be porous and non-zero water fluxes are allowed for negative depths. These methods include an additional porous layer [14, 11, 15, 7, 3], narrow connecting channels [16, 17] and also bear close resemblance to sub-grid scale bathymetry [18, 6] methods. The main advantage is that the artificial pressure gradient problem can be naturally circumvented. With porous media, free surface will fall under the bed in drying phase and eventually the spurious surface slope will disappear. Naturally the drawback is that "virtual water" is generated as depth attains negative value, although mass-conservative formulations are reported to exist [17, 18]. The key advantage of porosity methods is that the transition between wet and dry areas is smooth and the modifications can be expressed in the primitive equations. Due to the latter reason many methods of this type are compatible with semi-implicit or implicit [14, 11, 7] time integration, reducing computational cost significantly.

Some WD methods are exploiting the fact that the WD process is essentially dominated by the pressure gradient and bottom friction, which can be used to simplify the equations; see for example [14, 19]. Requiring a balance between the two will ultimately lead to the so called diffusive wave approximation of the shallow water equation [20, 21] that can be used to model the WD interface. Such an approach, however, is not applicable to coastal waters in general.

In this paper we present an alternative "negative-depth" implicit WD formulation for FE shallow water models. However, in contrast to the porous media methods, our method is based on the idea that the bed is allowed to move in time as water elevation drops, which leads to a very similar formulation but without the need to introduce the concept of porosity. This

yields simpler implementation and most importantly the proposed method has only one unknown parameter whose value can be estimated fairly reliably.

Although the notion of moving the bathymetry may seem unusual, similar modifications have been applied locally in certain WD methods. An example of an FD model where bathymetry is temporally modified for the computation of elevation gradient is presented in [19]. Similar modifications in FV models can be found in [22, 10]. In all these references the motivation to modify the bathymetry is to avoid spurious pressure forces at the WD interface, precisely as in our case. The major difference is that here the modification is formulated already in the primitive equations.

As our goal is to perform long term environmental simulations, WD treatment should not jeopardise mass conservation nor introduce spurious tracer transport. Therefore special care is taken to guarantee strict mass conservation property. In addition, a consistent tracer equation and its FE implementation is presented.

The proposed WD methods is implemented in the FE Second-generation Louvain-la-Neuve Ice-ocean Model (SLIM)[1] [9, 23] and its validity is demonstrated with a set of standard numerical tests. However, in many cases, the test cases have proved out to be too easy to solve compared to simulations with real-world bathymetry, and thus more challenging test cases are also introduced.

The paper is organised as follows. The shallow water equations with the moving bathymetry are introduced in Section 2 and a fully implicit FE implementation is presented in Section 3. A consistent tracer equation is then presented in Section 4 followed by numerical tests in Section 5 and a real-world application in Section 6.

2. Shallow water equations allowing moving bathymetry

Given Cartesian horizontal coordinates $\boldsymbol{x} = [x, y]^T$, the depth averaged shallow water equations (SWEs) in non-conservative form are:

$$\frac{\partial \eta}{\partial t} + \boldsymbol{\nabla} \cdot (H\ \boldsymbol{u}) = 0\ , \tag{1}$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \boldsymbol{\nabla})\, \boldsymbol{u} + F_c \boldsymbol{e}_z \times \boldsymbol{u} + g \boldsymbol{\nabla} \eta = \frac{\boldsymbol{\tau}_s - \boldsymbol{\tau}_b}{\rho H}\ . \tag{2}$$

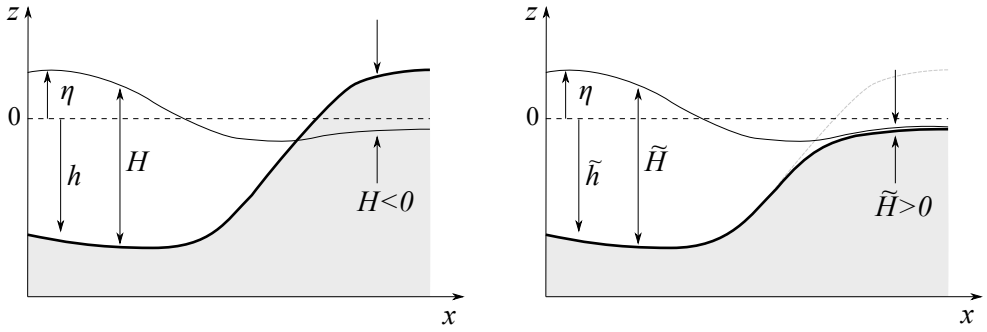
[1] `http://www.climate.be/SLIM`

where the water column depth is $H(\boldsymbol{x},t) = \eta(\boldsymbol{x},t) + h(\boldsymbol{x})$, η being the free surface elevation versus a reference level, h the original static bathymetry, $\boldsymbol{u} = [u,v]^T$ the horizontal (depth averaged) velocity, g the gravity acceleration, F_c the Coriolis factor, ρ density of water, and $\boldsymbol{\tau}_b$ and $\boldsymbol{\tau}_s$ the bottom and surface stress vectors, respectively. Here, the horizontal diffusion term in equation (2) has been omitted because it plays little role in WD processes. Proper DG-FEM treatment of the horizontal diffusion term can be found in [24], for example. In this work the Chézy-Manning formulation for bottom friction is used (n denotes the Manning coefficient):

$$\frac{\boldsymbol{\tau}_b}{\rho} = gn^2 \frac{\|\boldsymbol{u}\|\boldsymbol{u}}{H^{1/3}} \tag{3}$$

To ensure positive water depth, we introduce a smooth (at least once continuously differentiable) function f and redefine bathymetry as $\tilde{h}(\boldsymbol{x},t) = h + f(H)$. The function f is chosen so that the redefined total depth remains positive, i.e. $\tilde{H} = \eta + \tilde{h} > 0$ is always satisfied (see Figures 1 and 2). Thus $\tilde{h}$ is a function of the elevation and static bathymetry:

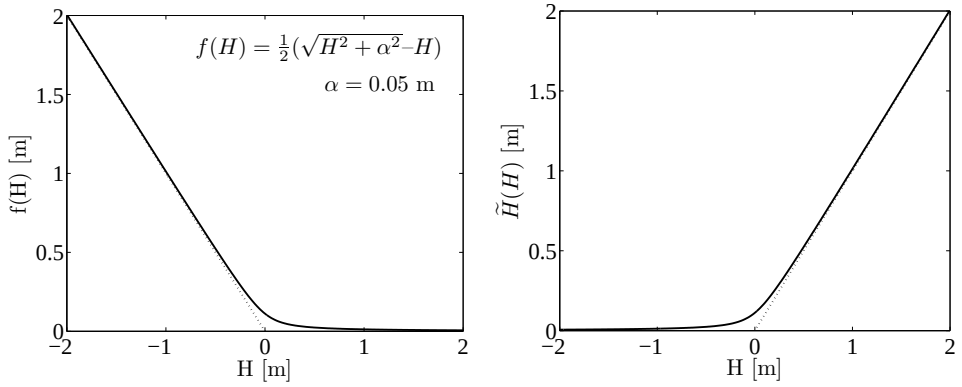
$$\tilde{h}(\eta, h) = h + f(\eta + h). \tag{4}$$



(a) Variables for fixed bathymetry (b) Variables for modified bathymetry

Figure 1: Redefinition of the total water column depth.

Now the SWEs are modified in such a way that the bed fluctuation is properly taken into account, i.e. the redefined total depth $\tilde{H}$ is being con-



(a) Function $f(H)$ (b) Modified total depth $\tilde{H}$

Figure 2: Functions for defining moving bathymetry.

served:

$$\frac{\partial \eta}{\partial t} + \frac{\partial \tilde{h}}{\partial t} + \boldsymbol{\nabla} \cdot \left(\tilde{H} \boldsymbol{u} \right) = 0 \ , \tag{5}$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \boldsymbol{\nabla}) \, \boldsymbol{u} + F_c \boldsymbol{e}_z \times \boldsymbol{u} + g \boldsymbol{\nabla} \eta = \frac{\boldsymbol{\tau}_s - \boldsymbol{\tau}_b}{\rho \tilde{H}} \ , \tag{6}$$

where the second term in the continuity equation (5) is due to the fact that $\tilde{h}$ is not static. Note that the momentum equation is modified simply by replacing H by $\tilde{H}$, which only appears in the forcing and dissipation terms.

The continuity equation can be further developed by noting that $\frac{\partial \eta}{\partial t} + \frac{\partial \tilde{h}}{\partial t} = (1 + f'(H)) \frac{\partial \eta}{\partial t} := A(H) \frac{\partial \eta}{\partial t}$ which leads to

$$A(H) \frac{\partial \eta}{\partial t} + \boldsymbol{\nabla} \cdot \left(\tilde{H} \boldsymbol{u} \right) = 0. \tag{7}$$

The above formulation is closely related to scaling of continuity equation that is presented in porous media methods [11, 14, 17] and also in subgrid scale methods [6]. Indeed, it is easy to see that $0 < A < 1$ so that A is a smooth indicator that ranges from dry ($A \approx 0$) to wet ($A \approx 1$) conditions. In the context of porous media and subgrid scale wetting-drying, A is interpreted as the "active" (or "wet") fraction, i.e. proportion of total element area that is penetrable by water. In contrast to methods where elements are either wet or dry, it is exactly the smoothness of A that prevents spurious oscillations and allows implicit time marching.

Despite the illustrative power of the scaled formulation (7), equation (5) will be implemented in the numerical system thus avoiding the computation of $f'(H)$.

When comparing to other WD methods it is useful to plot the total water column depth on the original static bathymetry. In this case the elevation is given by

$$\tilde{\eta} = \tilde{H} - h = \eta + f. \tag{8}$$

3. Numerical FE implementation

In the previous Section the proposed WD method was presented on the level of the primitive equations. In this section it is shown how the method can be implemented in DG-FEM shallow water models.

3.1. Weak formulation

In order to derive a weak formulation for the latter equations, we define a suitable triangulation $\mathcal{T}$ of the domain Ω and piecewise discontinuous polynomial function spaces $\mathcal{V}^p$ and $\mathcal{W}^p$, such that every function $a : \mathbb{R}^2 \to \mathbb{R}$, $a \in \mathcal{V}^p$, $\boldsymbol{b} : \mathbb{R}^2 \to \mathbb{R}^2$, $\boldsymbol{b} \in \mathcal{W}^p$ is polynomial of order p (at most) inside the triangles $K \in \mathcal{T}$ and discontinuous at the interfaces $e = K \cap K'$. Since the functions are discontinuous at the interfaces we can define a set of polynomial basis functions φ_i and $\boldsymbol{\psi}_i$ that are non-zero only in a single element. Therefore we have representations for $a = \sum_i a_i \varphi_i$ and $\boldsymbol{b} = \sum_i \boldsymbol{b}_i^T \boldsymbol{\psi}_i$.

We now require that the numerical solution to (5) belongs to these spaces, i.e. $\hat{\eta} \in \mathcal{V}^p$ and $\hat{\boldsymbol{u}} \in \mathcal{W}^p$. Multiplying (5) and (6) by test functions $\varphi \in \mathcal{V}^p$ and $\boldsymbol{\psi} \in \mathcal{W}^p$, respectively, integrating by parts and denoting the element-wise surface and contour integrals as

$$\int_{K \in \mathcal{T}} \cdot \, d\boldsymbol{x} = \Big\langle \cdot \Big\rangle_K, \tag{9}$$

$$\int_{e = K \cap K'} \cdot \, dS = \Big\langle\!\Big\langle \cdot \Big\rangle\!\Big\rangle_e, \tag{10}$$

respectively, the weak formulation of the system (5)-(6) becomes

$$\left\langle \frac{\partial \eta}{\partial t}\varphi \right\rangle_K + \left\langle \frac{\partial \tilde{h}}{\partial t}\varphi \right\rangle_K + \sum_{e=K\cap K'} \left\langle\!\left\langle (\tilde{H}\boldsymbol{u})^* \cdot \boldsymbol{n}\varphi \right\rangle\!\right\rangle_e - \left\langle \tilde{H}\boldsymbol{u} \cdot \boldsymbol{\nabla}\varphi \right\rangle_K = 0, \tag{11}$$

$$\begin{aligned} \left\langle \frac{\partial \boldsymbol{u}}{\partial t} \cdot \boldsymbol{\psi} \right\rangle_K + \sum_{e=K\cap K'} \left\langle\!\left\langle \boldsymbol{u}^* \cdot \boldsymbol{n}\ \boldsymbol{u}^* \cdot \boldsymbol{\psi} \right\rangle\!\right\rangle_e - \left\langle \boldsymbol{\nabla} \cdot (\boldsymbol{u}\boldsymbol{\psi}) \cdot \boldsymbol{u} \right\rangle_K \\ + \sum_{e=K\cap K'} \left\langle\!\left\langle g\eta^* \boldsymbol{\psi} \cdot \boldsymbol{n} \right\rangle\!\right\rangle_e - \left\langle g\eta \boldsymbol{\nabla} \cdot \boldsymbol{\psi} \right\rangle_K \\ + \left\langle F_c \boldsymbol{e}_z \times \boldsymbol{u} \cdot \boldsymbol{\psi} \right\rangle_K = \left\langle \left(\frac{\boldsymbol{\tau}_s - \boldsymbol{\tau}_b}{\rho \tilde{H}}\right) \cdot \boldsymbol{\psi} \right\rangle_K. \end{aligned} \tag{12}$$

The major difference between the above weak formulation and standard FE SWE formulation is the novel mass correction term due to the moving bathymetry, i.e. the second term in equation (11).

Due to the discontinuity, the values of η and $\boldsymbol{u}$ are defined twice at element interfaces. Thus the corresponding values in the contour integrals are ambiguous and are marked with an asterisk. These values are solved with an approximate Riemann solver that is derived in the next section.

3.2. Approximate Riemann solver

In this work we are using an approximate Riemann solver that is based on the Roe averages [25], which is commonly used in SWE models, e.g. [9, 10, 26, 27]. Another DG-FEM WD method by Bunya et al. utilises Lax-Friedrichs flux [8]. In [28] Harten-Lax-van Leer-Contact (HLLC) flux, which ensures non-negative water depth, is used. Such flux, however, contains many conditional statements and is thus difficult to implement in an implicit model.

Consider two neighbouring elements, left K_l and right K_r, $e = K_l \cap K_r \neq \varnothing$, such that the x axis is oriented to the normal direction from K_l to K_r and the element interface is along the y axis. The corresponding velocities are denoted u and v, respectively. The Riemann problem is derived with conservative form of the shallow water equations, but since we are only interested in transport in the normal direction, the partial derivatives with respect to y can be omitted. Considering first the non-modified SWEs, one

obtains the following set of equations:

$$\frac{\partial H}{\partial t} + \frac{\partial (Hu)}{\partial x} = 0 \tag{13}$$

$$\frac{\partial (Hu)}{\partial t} + \frac{\partial (Hu^2)}{\partial x} + \frac{\partial (\frac{1}{2} g H^2)}{\partial x} = gH \frac{\partial h}{\partial x} \tag{14}$$

$$\frac{\partial (Hv)}{\partial t} + \frac{\partial (Huv)}{\partial x} = 0. \tag{15}$$

In what follows, we do not take into account the bathymetry gradient $\frac{\partial h}{\partial x}$, because incorporating its influence in approximate Riemann solvers is a still a quite difficult task. There are indeed numerous publications on the topic, mainly with respect to FV methods (see e.g. [29, 10] and references therein). However, such a difficulty is not analysed here for the sake of simplicity.

With this assumption the right hand side of equation (14) disappears and one ends up with a homogeneous set of equations. By defining a state vector $\boldsymbol{q} = [H, Hu, Hv]^T$ the system can be written as (using shorthand notation $\boldsymbol{q}_t := \frac{\partial \boldsymbol{q}}{\partial t}$)

$$\boldsymbol{q}_t + \boldsymbol{F}_x = \boldsymbol{q}_t + \boldsymbol{V}(\boldsymbol{q}) \boldsymbol{q}_x = 0, \tag{16}$$

where

$$\boldsymbol{V}(\boldsymbol{q}) = \begin{bmatrix} 0 & 1 & 0 \\ -u^2 + gH & 2u & 0 \\ -uv & v & u \end{bmatrix}. \tag{17}$$

Approximate Riemann solvers are based on the idea of linearising (16), i.e. replacing $\boldsymbol{V}(\boldsymbol{q})$ with a local constant. Given the state vectors on both sides of the boundary, $\boldsymbol{q}_l$ and $\boldsymbol{q}_r$, the linearisation $\boldsymbol{V}_{\text{lin}}$ can be found by requiring that it satisfies the Rankine-Hugoniot equation: $\boldsymbol{V}_{\text{lin}}(\boldsymbol{q}_l - \boldsymbol{q}_r) = \boldsymbol{F}_l - \boldsymbol{F}_r$. This

leads to the Roe averages [25][2]:

$$H_{\text{Roe}} = (H_l + H_r)/2, \tag{18}$$

$$u_{\text{Roe}} = \frac{\sqrt{H_l}u_l + \sqrt{H_r}u_r}{\sqrt{H_l} + \sqrt{H_r}}, \tag{19}$$

$$v_{\text{Roe}} = \frac{\sqrt{H_l}v_l + \sqrt{H_r}v_r}{\sqrt{H_l} + \sqrt{H_r}}. \tag{20}$$

The $\boldsymbol{V}_{\text{lin}}$ is obtained by substituting (18)-(19) to (17). The linear system can now be solved by diagonalising $\boldsymbol{V}_{\text{lin}} = \boldsymbol{R}\boldsymbol{D}\boldsymbol{R}^{-1}$. The eigenvalues of $\boldsymbol{V}$ are $\{\lambda_i\}_{i=1}^3 = \{u, u + c_{\text{Roe}}, u - c_{\text{Roe}}\}$, $c_{\text{Roe}} = \sqrt{gH_{\text{Roe}}}$, corresponding to speeds at which waves propagate across the boundary. Introducing new state vectors, $\boldsymbol{Q} = \boldsymbol{R}^{-1}\boldsymbol{q}$, leads to a system of independent equations $\boldsymbol{Q}_t + \boldsymbol{D}\boldsymbol{Q}_x = 0$. These are solved simply by taking the upwind value of $\boldsymbol{Q}_l$ and $\boldsymbol{Q}_r$:

$$Q_i^* = \begin{cases} Q_{i,l}, & \lambda_i > 0 \\ Q_{i,r}, & \lambda_i < 0 \end{cases}, \quad i = 1, \ldots, 3. \tag{21}$$

Defining a jump operator $[[a]] = (a_l - a_r)/2$ and a mean operator $\{a\} = (a_l + a_r)/2$ the solutions can be formulated as

$$\begin{aligned} Q_i^* &= s_i[[Q_i]] + \{Q_i\}, \quad i = 1, \ldots, 3, \\ s_i &= \text{sign}(\lambda_i). \end{aligned} \tag{22}$$

[2]The problem is in fact under-determined, so (18) is chosen rather than deduced. Using an additional equation will lead to fully determined system with the same result, see e.g. [26]

Finally, one can compute the solution $\boldsymbol{q}^* = \boldsymbol{R}\boldsymbol{Q}^*$:

$$H^* = \{H\} + \frac{s_2 + s_3}{2}[[H]] + \frac{s_2 - s_3}{2c_{\mathrm{Roe}}}\left([[Hu]] - u_{\mathrm{Roe}}[[H]]\right) \tag{23}$$

$$(Hu)^* = \{Hu\} + \frac{s_2 + s_3}{2}[[Hu]] + \frac{s_2 - s_3}{2}c_{\mathrm{Roe}}[[H]] + \frac{(s_2 - s_3)u_{\mathrm{Roe}}}{2c_{\mathrm{Roe}}}\left([[Hu]] - u_{\mathrm{Roe}}[[H]]\right) \tag{24}$$

$$(Hv)^* = \{Hv\} + s_1[[Hv]] + \frac{s_2 + s_3 - 2s_1}{2}v_{\mathrm{Roe}}[[H]] + \frac{(s_2 - s_3)v_{\mathrm{Roe}}}{2c_{\mathrm{Roe}}}\left([[Hu]] - u_{\mathrm{Roe}}[[H]]\right) \tag{25}$$

$$\eta^* = H^* - h \tag{26}$$

$$u^* = (Hu)^*/H^* \tag{27}$$

$$v^* = (Hv)^*/H^* \tag{28}$$

Generally, the horizontal coordinate axes do not coincide with the element interface and an appropriate rotation is applied to the coordinate system so that the above formulation remains valid.

In the Riemann solver presented above it is necessary to compute the square root of the total depth. When extending the solver to the moving bathymetry WD method, we therefore use the modified depth $\tilde{H}$ for which positivity is guaranteed. In other words, H is replaced by $\tilde{H}$ in equations (23)-(25). After computing $\tilde{H}^*$, $(\tilde{H}u)^*$ and $(\tilde{H}v)^*$, the final solution is obtained as:

$$\eta^* = \tilde{H}^* - \{\tilde{h}\} \tag{29}$$

$$u^* = (\tilde{H}u)^*/\tilde{H}^* \tag{30}$$

$$v^* = (\tilde{H}v)^*/\tilde{H}^* \tag{31}$$

As stated earlier, the bathymetry source term is ignored in this formulation. Including the bathymetry gradient is not obvious as it leads to an inhomogeneous system that cannot be represented as fluxes [29]. In the case of moving bathymetry this problem is even more complicated as the bathymetry $\tilde{h}$ is discontinuous and also depends non-linearly on the elevation.

The presented procedure (29)-(31) is equivalent to ignoring the possible jumps in the bathymetry and assuming that the two water columns $\tilde{H}_l$ and $\tilde{H}_r$ stand on a common, continuous bed. In practice the procedure appeared to be sufficient as long as the discontinuity in $\tilde{h}$ remains small.

3.3. Implicit time marching

Equations (11-12) are expressed in continuous time and discrete space. It must be noted that these equations are specific compared to common finite element formulations. Indeed, the mass correction term, i.e. the time derivative of the bathymetry $\tilde{h}$, is non-linear. In general there are two ways of solving non-linear PDEs: Either explicit time integration with high temporal resolution or implicit time integration featuring a non-linear solver. In order to solve the equations accurately with reasonable computational cost, implicit time-stepping scheme is adopted here.

Diagonally Implicit Runge-Kutta (DIRK) methods provide high order accuracy and high flexibility, as the time step can be easily varied. Runge-Kutta methods involve multiple stages, and with DIRK schemes a non-linear system of equations is to be solved at each stage.

For brevity, the system (continuous in time and discrete in space) is written as follows:

$$\left\langle \varphi \frac{\partial \left(\eta + \tilde{h}(\eta)\right)}{\partial t} \right\rangle_K = \boldsymbol{S}_\eta(\eta, \boldsymbol{u}, \varphi), \tag{32}$$

$$\left\langle \boldsymbol{\psi} \cdot \frac{\partial \boldsymbol{u}}{\partial t} \right\rangle_K = \boldsymbol{S}_{\boldsymbol{u}}(\eta, \boldsymbol{u}, \boldsymbol{\psi}), \tag{33}$$

where $\boldsymbol{S}_\eta(\eta, \boldsymbol{u})$ and $\boldsymbol{S}_{\boldsymbol{u}}(\eta, \boldsymbol{u})$ are the discrete spatial operators for η and $\boldsymbol{u}$, respectively.

Given the solution $(\eta^{n-1}, \boldsymbol{u}^{n-1})$ at time t^{n-1}, the solution at time t^n is obtained with a fully implicit Runge-Kutta time marching defined below. The superscript index n refers to time steps while superscript i is used to denote the Runge-Kutta stages.

- For each stage $i = 1 \dots s$, solve the non-linear system of equations:

$$\left\langle \varphi \eta^i \right\rangle_K = -\left\langle \varphi \tilde{h}(\eta^i) \right\rangle_K + \left\langle \varphi \left(\eta^{n-1} + \tilde{h}(\eta^{n-1})\right) \right\rangle_K + \Delta t \sum_{j=1}^{i} a_{i,j} \boldsymbol{S}_\eta(\eta^j, \boldsymbol{u}^j), \tag{34}$$

$$\left\langle \boldsymbol{\psi} \cdot \boldsymbol{u}^i \right\rangle_K = \left\langle \boldsymbol{\psi} \cdot \boldsymbol{u}^{n-1} \right\rangle_K + \Delta t \sum_{j=1}^{i} a_{i,j} \boldsymbol{S}_{\boldsymbol{u}}(\eta^j, \boldsymbol{u}^j), \tag{35}$$

- The final stage reads:

$$\left\langle \varphi\eta^n \right\rangle_K = -\left\langle \varphi\tilde{h}(\eta^n) \right\rangle_K + \left\langle \varphi\left(\eta^{n-1} + \tilde{h}(\eta^{n-1})\right) \right\rangle_K + \Delta t \sum_{j=1}^{s} b_j \boldsymbol{S}_\eta(\eta^j, \boldsymbol{u}^j),$$

$$\left\langle \boldsymbol{\psi} \cdot \boldsymbol{u}^n \right\rangle_K = \left\langle \boldsymbol{\psi} \cdot \boldsymbol{u}^{n-1} \right\rangle_K + \Delta t \sum_{j=1}^{s} b_j \boldsymbol{S}_{\boldsymbol{u}}(\eta^j, \boldsymbol{u}^j),$$

In comparison to usual implicit DG-FEM implementation the major difference here is the treatment of the non-linear mass correction term shown in equation (32). As a consequence a new term $\left\langle \varphi\tilde{h}(\eta^i) \right\rangle_K$ appears in equation (34) that depends non-linearly on the state variable η^i.

The weights $a_{i,j}$, b_j and c_i are the Runge-Kutta coefficients. The time corresponding to each stage i is $t^i = t^n + c_i\Delta t$ which is used when computing the external forcings. Here we only use schemes for which $a_{s,j} = b_j$, $c_s = 1$ and thus the s-th RK stage gives the final solution directly, i.e. $\eta^n = \eta^s$, $\boldsymbol{u}^n = \boldsymbol{u}^s$. In this paper we are using second and fourth order accurate schemes, namely DIRK22 [30, section 2.6] and ESDIRK64 [31, Appendix A] that are presented in the Appendix.

3.4. Newton solver

The equations (34)-(35) are fully implicit and non-linear. They are solved with a conventional Newton iteration. The non-linear system is linearised, the required Jacobian matrix of the system being approximated numerically. The Jacobian of equations (34)-(35) is given by

$$\boldsymbol{J}(\eta^i, \boldsymbol{u}^i) = \begin{bmatrix} -\left\langle \varphi\tilde{h}(\eta^i) \right\rangle_K + \Delta t\ a_{i,i} \boldsymbol{S}_\eta(\eta^i, \boldsymbol{u}^i) \\ \Delta t\ a_{i,i} \boldsymbol{S}_{\boldsymbol{u}}(\eta^i, \boldsymbol{u}^i) \end{bmatrix} \begin{bmatrix} \frac{\partial}{\partial \eta^i} & \frac{\partial}{\partial u^i} & \frac{\partial}{\partial v^i} \end{bmatrix}.$$

To compute the Jacobian reliably the presented numerical system has to be continuously differentiable with respect to the state variables η, u, v. In this work the partial derivatives are approximated by means of finite differencing where the perturbation to the state variables was of order 10^{-6}. Such a numerical differentiation is valid for continuously differentiable functions. At discontinuities, however, numerical approximation is not reliable and it often causes oscillations that prevent the Newton solver from converging.

The advantage of the proposed WD method is that all the modifications are indeed smooth without any discontinuities or switches. In what has been

presented, the only non-differentiable operator is the sign function found in the Riemann solver in equation (22). The sign function is replaced with a smooth approximation:

$$\operatorname{sign}(u) \approx \tanh \beta \frac{u}{U}, \quad \beta \approx 60,\ U \approx 1\text{m/s}. \tag{36}$$

For discontinuous Galerkin discretisation, the Jacobian matrix is composed of blocks, where each block corresponds to an element. This block structure enables an efficient Incomplete LU (ILU) factorisation. Using an ILU factorisation with no fill-in as preconditioner for GMRES (Generalized Minimal Residual Method, [32]) iteration appears to be sufficient to ensure convergence. In our implementation, such an efficiency can only be achieved with DG elements.

In practice the Newton solver appeared to be robust and it was observed to converge rapidly. Stopping criterion was the relative error versus initial state, i.e. $\varepsilon_{rel,i} = \varepsilon_i / \varepsilon_0$, where ε_i is the residual of iteration i. Typically only 3 to 6 iterations were required to reach $\varepsilon_{rel} = 10^{-7}$. In the inner loop, the GMRES solver required roughly 50 iterations on average to converge to tolerance $\varepsilon_{rel} = 10^{-3}$ (compared to the beginning of the GMRES iteration).

3.5. Mass conservation

Equation (5) represents mass conservation of the entire water column. The corresponding weak formulation (11) will result in a mass conservative scheme if the term $\left\langle \varphi\, \partial\tilde{h}/\partial t \right\rangle_K$ is computed accurately. The presented Runge-Kutta time marching will conserve mass because at each subiteration $\partial\eta/\partial t$ and $\partial\tilde{h}/\partial t$ are treated similarly. Thus only requirement is that the term $\left\langle \varphi\, \tilde{h}(\eta^i) \right\rangle_K$ is computed accurately in equation (34).

The latter imposes two restrictions: First, due to the non-linearity of $\tilde{h}$ an iterative solver, such as the Newton method proposed here, must be used. Explicit methods can only provide approximate mass conservation. Secondly, the numerical quadrature of $\left\langle \varphi\, \tilde{h}(\eta^i) \right\rangle_K$ must be accurate.

In this work conventional Hammer quadrature rules are used, that are accurate up to polynomials of order $2p + 1$, where p is the degree of the FE discretisation (the extra p is required for the test function). However, no polynomial can meet the desired properties of $\tilde{h}$ outlined in Section 2 and indeed $f(H)$ must be of infinite order. Therefore simply evaluating $\tilde{h}$ at integration points implies that the quadrature is not exact and mass conservation breaks down.

Up to this point the polynomial order of the elevation field has been free but now we restrict ourselves to $p = 1$. Linear elements have the unique property that the extrema are found at the vertices and thus it suffices to ensure that water depth remains positive at the nodes. The moving bathymetry is therefore defined by computing the nodal values using (4) with linear interpolation in between:

$$\hat{\tilde{h}}^n = \sum_{i=1}^{3} \tilde{h}(\eta_i^n, h_i)\varphi_i(\boldsymbol{x}), \tag{37}$$

where φ_i are now linear discontinuous basis functions. Equation (37) results in a mass conservative scheme. For velocity field higher order elements can still be used. In this work first order elements are used for all fields.

3.6. Choosing the function $f(H)$

The function f should meet the following properties:

- $\tilde{H} = H + f(H) > 0$ for all H
- $f \approx 0$ for $H >> 1$
- f must be continuously differentiable

The first property was already mentioned in Section 2. The second property states that the modification is restricted to shallow areas only. And the third property is needed for ensuring convergence of the Newton iteration. In this work the following function, that fulfils the desired properties, is used:

$$f(H) = \frac{1}{2}(\sqrt{H^2 + \alpha^2} - H), \tag{38}$$

Clearly, f is monotonously decreasing, continuously differentiable and satisfy $f(H) > \max\{-H, 0\}\ \forall H \in \mathbb{R}$. The free parameter controls the smoothness of the transition: $f(H) \to \max\{-H, 0\}$, as $\alpha \to 0$ (see Figure 2a). The dimension of both f and α is meters.

In practice the α parameter affects the width of the transition zone between wet ($A(H) \approx 1$, $\tilde{h} \approx h$) and dry ($A(H) \approx 0$) areas. Noting that for $H = 0$, $\tilde{H} = 1/2\ \alpha$, it is easy to see that α also directly controls the water depth in dry areas. As such α is similar to the threshold depth parameter used in thin-layer WD methods, as both are expressed in meters and determine the remaining water layer depth.

For robust operation, the WD interface should be smooth implying that the transition zone should encompass more that one element. Estimating the variation in bathymetry within one element by

$$\epsilon := L_x |\boldsymbol{\nabla} h|, \tag{39}$$

where L_x is the horizontal length scale, one can conclude that $\alpha \approx \epsilon$, which can be used as a rule of thumb for gradually sloping domains.

In terms of numerical stability, other more restrictive constraints may exist. The system should be smooth enough to ensure convergence of the Newton solver. It is also plausible that rapidly varying flows and dealing with shock waves require larger values for α. These restrictions are, however, more difficult to estimate a priori.

4. Tracer consistency

The tracer equation is implemented in conservative form and thus taking into account the moving bathymetry is straightforward. Denoting the tracer concentration by C, the depth-averaged tracer equation in conservative form is given by

$$\frac{\partial HC}{\partial t} + \boldsymbol{\nabla} \cdot (HC\boldsymbol{u}) = S + \boldsymbol{\nabla} \cdot (\kappa H \boldsymbol{\nabla} C) \tag{40}$$

where κ is the tracer horizontal diffusivity and S contains the sources and sinks.

We now show that it is possible to formulate the tracer equation in such a way that it is consistent with the continuity equation (5). Consistency in this context means that setting $C = 1$ in the tracer equation should lead to exactly the same numerical procedure that is used to solve the continuity equation [33].

It is clear that replacing H by $\tilde{H}$ in equation (40) leads to equation that is consistent with (5). The weak form of the modified tracer equation becomes

$$\begin{aligned} \left\langle \frac{\partial \tilde{H}C}{\partial t}\varphi \right\rangle_K + \sum_{e=K\cap K'} \left\langle\!\left\langle (\tilde{H}C\boldsymbol{u})^* \cdot \boldsymbol{n}\varphi \right\rangle\!\right\rangle_e - \left\langle \tilde{H}C\boldsymbol{u} \cdot \boldsymbol{\nabla}\varphi \right\rangle_K = \\ \sum_{e=K\cap K'} \left\langle\!\left\langle \kappa \tilde{H}^* \boldsymbol{\nabla} C^* \cdot \boldsymbol{n}\varphi \right\rangle\!\right\rangle_e - \left\langle \kappa \tilde{H} \boldsymbol{\nabla} C \cdot \boldsymbol{\nabla}\varphi \right\rangle_K . \end{aligned} \tag{41}$$

Clearly, setting $C = 1$ in above leads to (11). Again the unknown values $(\tilde{H}C)^*$ in the element interfaces are computed with an approximate Riemann solver. A solution for $(\tilde{H}C)^*$ can be derived by adding the tracer equation (with zero diffusivity) to the system (16). Because in this context the tangential velocity v is essentially treated as a passive tracer, one obtains a similar solution:

$$(HC)^* = \{HC\} + s_1[[HC]] + \frac{s_2 + s_3 - 2s_1}{2} C_{\mathrm{Roe}}[[H]] + \frac{(s_2 - s_3)C_{\mathrm{Roe}}}{2c_{\mathrm{Roe}}} \left([[Hu]] - u_{\mathrm{Roe}}[[H]]\right) \tag{42}$$

In the above equation C_{Roe} denotes the tracer concentration value used in the linearised equations. Similarly to v_{Roe}, solving the Rankine-Hugoniot relation results in the Roe average:

$$C_{\mathrm{Roe}} = \frac{\sqrt{H_l}C_l + \sqrt{H_r}C_r}{\sqrt{H_l} + \sqrt{H_r}}, \tag{43}$$

Clearly, for an arbitrary constant tracer concentration $C = C_0$, one gets $C_{\mathrm{Roe}} = C_0$ and $(HC)^* = C_0 H^*$, which confirms consistency.

If a first order time discretisation is used the first term of equation (41) becomes

$$\left\langle \frac{\partial \tilde{H}c}{\partial t} \right\rangle_K = \left\langle \frac{(\tilde{h}^{n+1} + \eta^{n+1})c^{n+1} - (\tilde{h}^n + \eta^n)c^n}{\Delta t} \right\rangle_K, \tag{44}$$

which is linear in c^{n+1} and no iterative solver is needed. Equation (41) can therefore be solved in a conventional manner when $\tilde{H}^{n+1}$ is known. However, for consistency the same fully implicit Runge-Kutta time integration presented in Section 3.3 is used for solving the tracer equation.

5. Numerical tests

The presented DG-FEM WD method was tested with several test cases commonly used in the literature and also a couple of novel ones. All the tests were solved with DIRK22 time integration with 600 second time step unless otherwise noted.

5.1. Balzano test cases

The first test cases considered are those by Balzano [34]. They feature a rectangular 13800 m long basin, with mildly sloping bathymetry ranging from zero to 5 meters. As forcing, water level perturbation is prescribed at the deep end. The Manning bottom friction coefficient is set to 0.02 $\mathrm{s/m^{1/3}}$ while Coriolis force, viscosity and free surface stress are being ignored. The test cases are originally one dimensional but here they are solved in a 2D basin whose width is 7200 meters. For exact description of the bathymetry the reader should refer to [34] or [9].

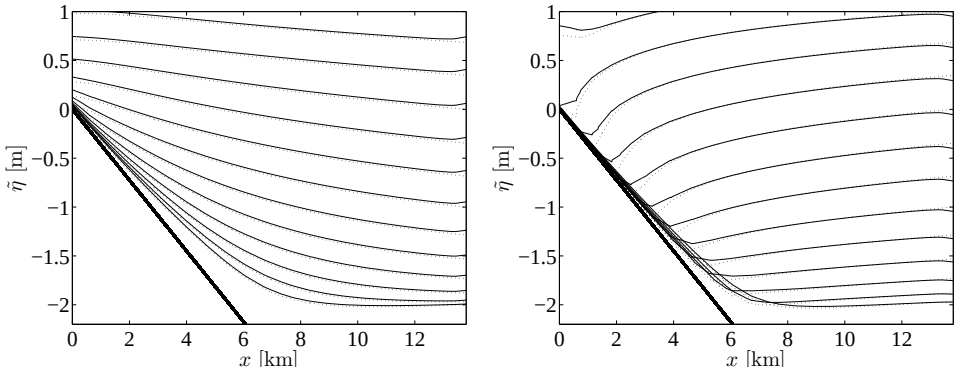
The first test case considers a wave run up on a uniformly sloping bed. A sinusoidal water level perturbation with an amplitude of 2 m and period of 12 hours is imposed. Here we are using a uniform mesh of 1200 m horizontal resolution, similar to the original one dimensional tests by Balzano. The vertical length scale $\epsilon = 0.43$ m and a slightly smaller value is used for the α smoothness parameter $\alpha = 0.3$ m. Figure 3 shows the modified elevation field $\tilde{\eta}$ at 20 minutes intervals for the drying and flooding phase. It is noteworthy that the $\tilde{\eta}$ elevation curves correspond to the total water column depth and thus these plots are directly comparable to other results in literature (e.g. [34, 15]).

For comparison the results obtained with a fully explicit flux-limiting WD method (see [9]) are also shown in 3. Clearly the two methods are indeed in good agreement. Moreover, the retention volume (water trapped in the dry areas) is not very large, in fact smaller than with some methods presented in [34] and no oscillations or wiggles are present. The only noticeable difference is that the shocks at the flooding front appear to be milder than with the explicit method, which is due to the smooth WD transition.

The second test is similar, except that the bed now features a flat shelf. The corresponding elevation field is plotted in Figure 4. The results are again in good agreement with the fully explicit WD method and similar to those presented in literature.

The third test case is different because the bed now features a small "pond" that retains water in the dry phase. In this simulation the water level at the open boundary is dropped sinusoidally to the minimal level, holding it there for an indefinitely long time in order to test whether water is leaking through the dry area. The elevation field after 100 hours, which essentially corresponds to the static solution, is shown in Figure 5. Because water fluxes do not vanish as long as the pressure gradient term operates, the pond eventually dries up. The flux depends on the parameter α (the smaller

the value of α, the smaller the flux) and the bottom friction parametrisation, but neither can prevent the pond from drying as time goes to infinity. Figure 5b presents the η field that is used in the numerical implementation. Indeed it is seen that the static case corresponds to situation where the elevation field is horizontal and thus the pressure gradient term is zero. This test reveals that water is leaking through dry areas which can be seen as a drawback of the presented method. However, it must be stressed that such drawback is common also to all porous media methods [15].

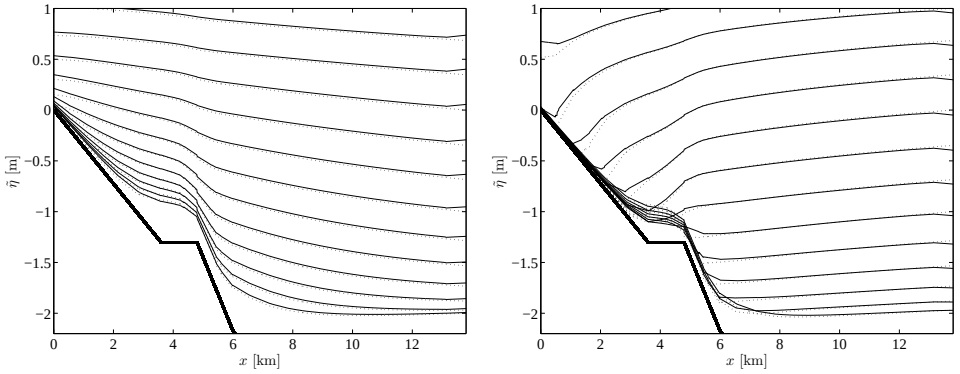


(a) drying phase (b) flooding phase

Figure 3: Balzano 1 test case. Elevation field at 20 minutes intervals for (a) the drying phase and (b) flooding phase. The thick line corresponds to the static bed. For comparison results obtained with an explicit WD method are show with a dotted line.

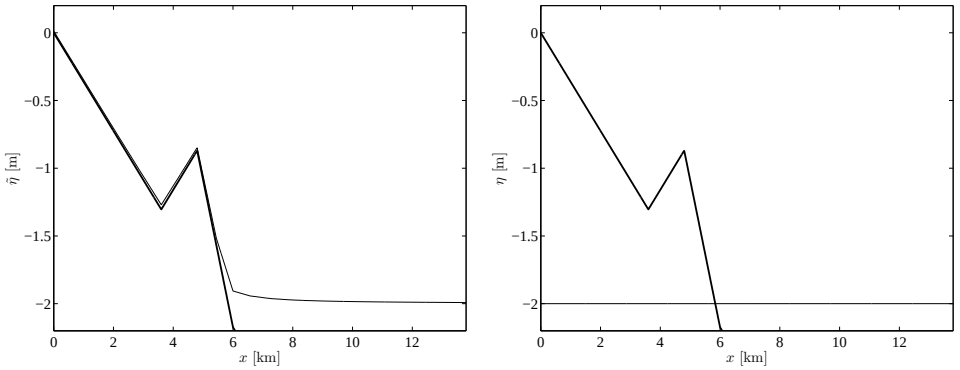
5.2. Steeper slopes

During the model development is was noted that the standard test cases proposed in the literature e.g. in [34] and [35] tend to be too easy compared to real-world simulations with complex topography. This is especially true with a steep bathymetry [36]. In this section we therefore present similar tests as the first test by Balzano but with increasing difficulty by steepening the bed slope. The proposed method was tested with several different mesh resolutions and the goal was to determine the smallest α values for which the method remains stable. To ensure similar conditions the tests were designed such that in each case at least one element would be completely dry at low water. The results are presented in Table 1.



(a) drying phase (b) flooding phase

Figure 4: Balzano 2 test case. Elevation field at 20 minutes intervals for (a) the drying phase and (b) flooding phase. The thick line corresponds to the static bed. For comparison results obtained with an explicit WD method are show with a dotted line.



(a) $\tilde{\eta}$ (b) η

Figure 5: Balzano 3 test case. The static solution after 100 h of simulation time. a) Total water column depth superimposed on the original bathymetry. b) Elevation field as seen by the pressure gradient term. The thick line corresponds to the static bed.

It is seen in Table 1 that the smallest possible α value remains relatively constant in all the cases despite the fact that the nature of the WD process varies dramatically. Indeed, the vertical length scale ϵ varies from 50 m (up-

	L_x [m]				
$\lvert\nabla h\rvert$ [m/m]	25	50	100	250	500
0.10	0.3	0.3	0.3	0.3	0.5
0.05	0.3	0.3	0.3	0.3	0.5
0.01	0.2	0.2	0.3	0.3	0.3
0.001	0.2*	0.2*	0.3*	0.5*	0.5*

Table 1: Smallest stable α parameter values (in meters) for various bed slopes and mesh resolutions. *) A mild shock wave appears at flooding phase, which requires larger α for coarse mesh.

per right corner) to 0.025 m (lower left corner in Table 1). In the case of very steep slopes the surface remains nearly horizontal and no wave propagation effects are visible at the WD front. For more gradual bathymetry wave effects become apparent and mild shocks appear at the flooding phase. To deal with such shocks a larger retention depth (water column depth in the dry area) may be needed to maintain stability. Larger retention depth implies faster wave propagation at the WD front which naturally smooths the shocks.

5.3. Thacker test case

Thacker [37] presented an analytical solution for water oscillation in a paraboloid bowl which has been used as a test case in many references [38, 34, 39, 28, 12]. The test case consists of a regular basin, a paraboloid of revolution, in which the free surface oscillates without any external forcing. The basin is large, 430.620 km in diameter, but very shallow as the depth is not more than 50 m at the centre. Initially the free surface is also a paraboloid of revolution. The dimensions of the bowl are chosen such that the free surface oscillation has a 12 hour period. For exact description of the test case see e.g. [9, 34].

This is a challenging test case as no bottom friction (or other dissipation) is present. Moreover the horizontal length scales are larger than in a typical application. Due to the large diameter of the basin, a common grid resolution ranges from 4 km to 10 km. Consequently, although the bed slope is very gradual, the vertical WD length scale is large: $\epsilon = 2.4$ m for a 10 km mesh. A slightly smaller value $\alpha = 2.2$ m was used for the smoothness parameter. Using a smaller α resulted in oscillations at the boundary, that in the absence

of any dissipation eventually deteriorated the solution in the whole domain. In order to simulate the WD process accurately, the simulations were carried out in a slightly larger domain, 495.2 km in diameter.

A cross section of the elevation at three time instances is shown in Figure 6a. In the centre of the domain, the numerical model is very close to the analytical solution while the difference increases towards the WD front. The smooth transition between the wet and dry regimes is clearly visible. As stated earlier the smoothness of the solution is controlled by the parameter α and thus smaller α yields more accurate solution. However, the smallest feasible α value appeared to depend on the horizontal resolution, and α was observed to be proportional to ϵ. Thus more accurate solutions can be obtained by decreasing α but the mesh has to be refined accordingly.

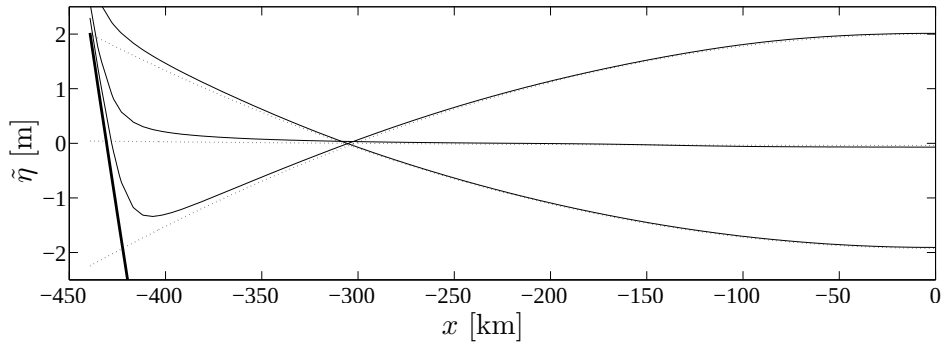
Figure 6b shows the water elevation at the centre of the domain versus time. It is seen that the proposed method is stable without noticeable additional dissipation. Moreover, as the signal does not attenuate significantly in time, the numerical dissipation also remains moderate. This test also justifies the higher order Runge-Kutta time integration presented in Section 3.3: Clearly first order implicit Euler method is too dissipative to be used in practical applications. Here the same time step was used for the implicit Euler method, but similar excessive dissipation was observed also for shorter time steps.

5.4. Mass conservation

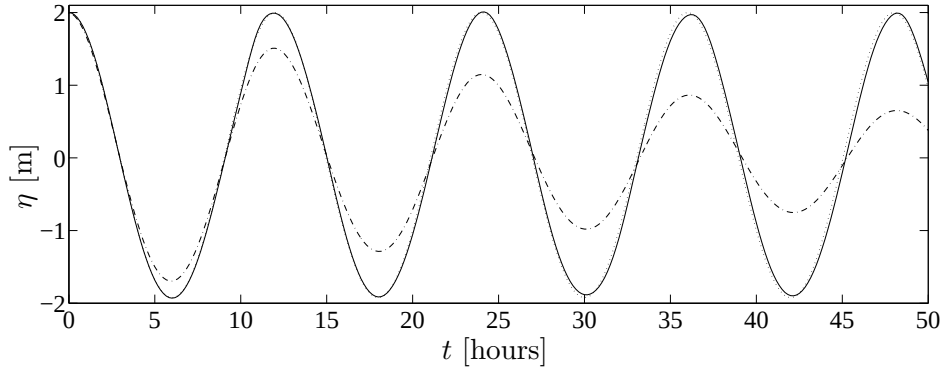
Because the Thacker test case features a closed basin it is well suited for testing the mass conservation property presented in Section 3.5. Denoting the total mass at time t by $M(t)$ the relative error $\mathcal{E} = (M(0) - M(t))/M(0)$ was of order 10^{-11} throughout the 25 hour simulation period, demonstrating that mass is conserved with sufficient accuracy.

5.5. Rate of convergence

The Thacker test case was also used to test how fast the proposed method converges toward the analytical solution when the mesh is refined. The test case was run with several meshes with increasing resolution $L_x = \{10,\ 15,\ 20,\ 30,\ 40\}$ km. As the smoothness parameter α affects the solution, it was tuned to match the length scale: the corresponding variable values were set to $\alpha = 0.9\epsilon$ i.e. $\alpha = \{2.16,\ 3.24,\ 4.32,\ 6.48,\ 8.64\}$ m. These values are close to the smallest stable values in each case because decreasing α significantly



(a)



(b)

Figure 6: Thacker test case. a) Solid thin line: elevation after 6, 9, and 12 hours. Dotted line: analytical solution. Thick line: bathymetry. b) Water level at the centre of domain versus time. Solid line: Numerical solution. Dotted line: analytical solution. Dash-dotted line: numerical solution with implicit Euler time integration.

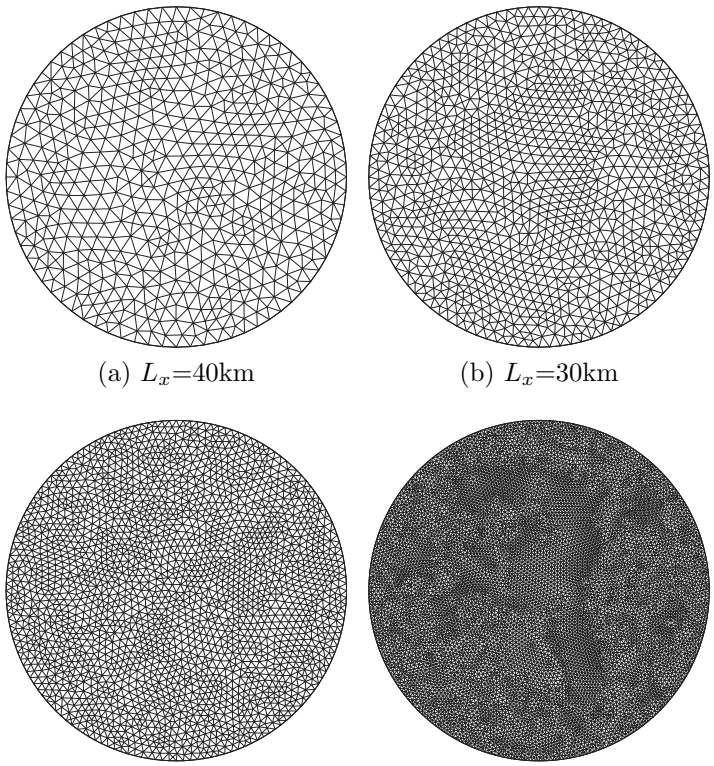
caused oscillations. Examples of the different meshes are illustrated in Figure 7.

The L_2 error of the elevation field $\tilde{\eta}$ was used as an error measure:

$$\mathcal{E}_{L_2} = \sqrt{\left\langle (\tilde{\eta} - \eta_a)^2 \right\rangle_\Omega / \left\langle 1 \right\rangle_\Omega}, \tag{45}$$

where η_a denotes the analytical solution that takes into account the dry bed, i.e. $\eta_a = \max\{\eta_{exact}, -h\}$. It is noteworthy that the error measure encompasses the entire domain, both wet and dry areas.

The observed L_2 error is shown on Figure 8, showing roughly 1.5 rate of



(a) L_x=40km (b) L_x=30km

(c) L_x=20km (d) L_x=10km

Figure 7: Meshes for the Thacker convergence test case.

convergence. It has to be stressed, however, that the convergence test is not entirely fair as we are comparing two different solutions. Indeed the analytical solution presented in [37] was developed for the original SWEs while in here the modified equations (5)-(6) are being solved. Moreover, the error is largest at the WD transition, where the numerical solution is smoother than the analytical (see Figure 6a). Therefore the error is mainly dominated by the parameter α instead of the spatial discretisation. Nevertheless, based on this test it can be stated that refining the mesh allows smaller α values leading to smaller discrepancies, and thus the solution does converge towards the solution of the original SWEs.

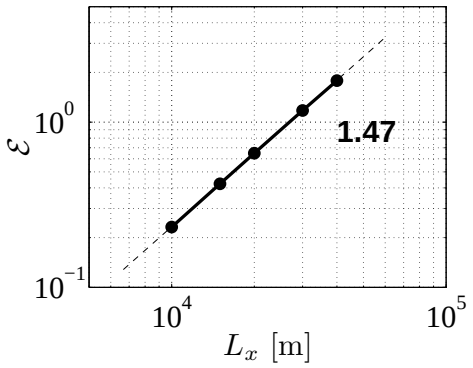


Figure 8: Convergence of the L_2 error versus spatial resolution. The observed rate of convergence is mainly due to the α parameter values.

5.6. Test on tracer consistency and conservation

Tracer consistency was tested with the Thacker test case by adding a passive tracer in the simulation. The tracer concentration was initially set to unity throughout the domain. As demonstrated in Section 4, the concentration should remain equal to unity at all times. In fact, the solution for such a tracer is trivial, and therefore the numerical method should converge instantly without any iterations. Such behaviour was indeed observed. During the simulation the error in tracer concentration was of order 10^{-8} which is the same magnitude as the residual tolerance used in the Newton iteration.

Tracer mass conservation was also tested. Figure 9 illustrates the relative error in total tracer mass versus time. The relative error is defined by $\mathcal{E}_C = (M_C(0) - M_C(t))/M_C(0)$, where $M_C(t)$ denotes the total tracer mass at time t. The tracer mass is conserved up to precision 10^{-15} which is the same order of magnitude as the numerical precision of the model.

6. Application to the Scheldt Estuary

The Scheldt Estuary is situated between Belgium and the Netherlands (Figure 10). The entire Scheldt River catchment area in northern France, Belgium and the Netherlands hosts approximately 7 million people and also features heavy industrial activity. Due to substantial and partly untreated discharges, the water quality in the river is generally poor with elevated levels of heavy metals, fecal bacteria and nutrients [40].

The Scheldt is a macrotidal estuary driven by the semidiurnal tides on the North Sea. Indeed, the residual water flux due to river discharge is roughly

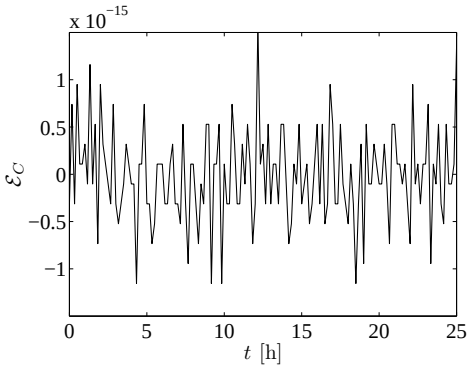


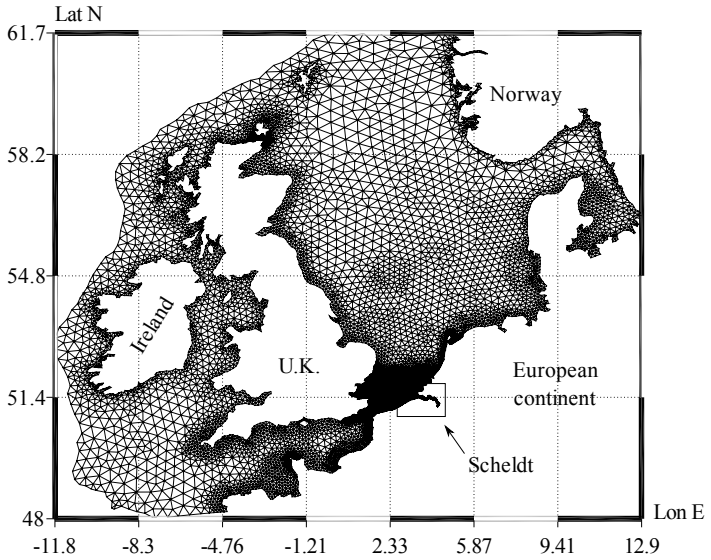
Figure 9: Error in total tracer mass versus time. The error is the same magnitude as the numerical precision.

two orders of magnitude smaller than that of the tidal flow. The width reduces from 6 km near the mouth to 500 m near Antwerp, roughly 80 km upstream. The tidal signal is amplified as it travels up the estuary, the tidal range near the mouth in Vlissingen being 3.8 meters and 5.2 meters further upstream in Antwerp. The estuary is generally very shallow (mean depth is roughly 10 m) but features deep flood and ebb channels (see Figure 11) that can reach the depth of 60 meters [41]. The main estuary also features large tidal flats (mainly Saeftinge and Ballatsplaat) and sand banks between the two main channels, both of which are submerged during high water.

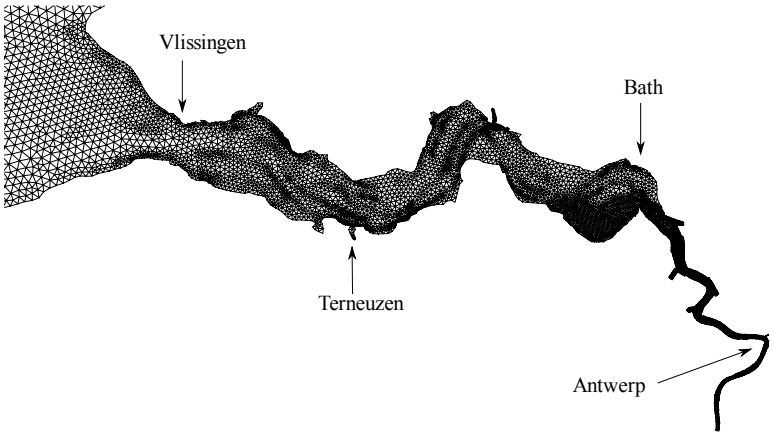
It is clear that taking wetting-drying phenomenon into account in such a domain is essential. The model domain is discretised with a triangular mesh that not only encompasses the Scheldt Estuary but also most of the North-western European Continental Shelf Sea (NWECSS) extending all the way to the shelf break (see Figure 10). Although the computational domain is extended drastically, the increase in computational cost remains moderate as roughly half of the elements are located inside the area of interest, i.e. the Scheldt. The mesh was generated with the GMSH software [42, 43].

The advantage of such domain extension is that the tide can be prescribed at the shelf break using data from global tidal models. Moreover, the meteorological events in the North Sea can easily be incorporated in the model. In this work ETOPO1 bathymetry data[3] [44] is used for the NWECSS while the

[3] `http://www.ngdc.noaa.gov/mgg/global/`



(a) Triangular mesh of the entire 2D domain



(b) Enlarged view of the Scheldt Estuary

Figure 10: Computational domain of the Scheldt application. The mesh contains 29130 elements. The upstream river network is modelled with 1D version of the SLIM (not shown here).

Scheldt bathymetry is obtained from KustZuid model[4]. The tidal signal at

[4]Courtesy of M. Zijlema of the National Institute for Coastal and Marine Management (RIKZ)

the open boundary is defined using TPXO7.1 model[5] [45]. The meteorological forcings (wind stress and atmospheric pressure) are from global NCEP reanalysis data[6] [46]. At the upstream boundary the 2D model is coupled with a 1D river network model that covers all the connected tidal rivers and tributaries. The tidal rivers are forced with observed river discharge at the upstream boundaries. In addition the freshwater discharges of Thames, Seine, Rhine and Meuse rivers are prescribed using daily average discharge data. More detailed information on the model setup can be found in [9] and [23].

The hydrodynamics were solved using the DIRK22 time marching with 20 minutes time-step. The WD the smoothness parameter was set to $\alpha = 0.5$ m.

Although model validation is not the main purpose of this paper, the bottom friction coefficient was tuned to obtain a good match of the tidal signal propagation by comparing water elevation to measurements in several stations in Scheldt and North Sea. The Manning coefficient ranges from 0.0235 $\mathrm{s}^{-1}\mathrm{m}^{1/3}$ in the shelf sea to 0.026 $\mathrm{s}^{-1}\mathrm{m}^{1/3}$ near Antwerp, increasing linearly along the estuary. SLIM has been validated for tidal elevation previously [23] and similar performance is obtained with the proposed WD method. Examples of two time series versus observations are presented in Figure 14. The model is clearly in good agreement with the data and there is no evidence that the moving bathymetry WD process would deteriorate the solution by affecting wave propagation, for example.

Snapshots of free surface elevation at high and low water can be seen in Figure 11 where the drying sand banks are clearly visible. Moreover, in Figure 12 it can be seen that the modification of the bathymetry is indeed restricted to the dry areas which also support the notion that the physical processes in the wet domain remain essentially unmodified. The depth averaged velocity field is shown in Figure 13. The flow circulates smoothly around the sand banks and no spurious noise is visible. Asymmetric flow patterns between ebb and flood channels is clearly observed [41].

One can conclude that qualitatively the model performance is very similar to what has been reported earlier [9]. The major difference is in the computational cost. As most of the WD methods rely on explicit time integration, the longest permissible time step is heavily restricted by the CFL condition.

[5] `http://www.oce.orst.edu/research/po/research/tide/index.html`

[6] `http://www.cdc.noaa.gov/cdc/data.ncep.reanalysis.surfaceflux.html`

Simulation	Number of elements	Method	Δt	T	CPUtime	Gain
Balzano 1	561	expl.	≈ 5 s	12 h	116.9 s	
	561	impl.	600 s	12 h	20.46 s	5.7
Scheldt	29130	expl.	< 0.2 s	1 h	34592 s	
	29130	impl.	1200 s	1 h	129.64 s	267

Table 2: Comparison of CPU times for flux-limiting and moving bathymetry wetting-drying. The presented method runs significantly faster due to the implicit time stepping.

In the Scheldt Estuary the time step is less that one second due to the deep channels and relatively small elements.

With the proposed fully implicit time marching, however, there is no intrinsic restriction on the time step, it is only required to resolve the tide and the forcing signals. In this simulation therefore a much longer time step of 20 minutes is used.

In Table 2 the overall CPU time is compared to the explicit flux-limiting WD method [9]. The comparison was carried out for Balzano test 1 and the Scheldt application running the simulations for a short period of time (T). These tests were run on an Intel Xeon processor using four nodes. It is seen that already in the Balzano benchmark implicit code runs much faster, while in the Scheldt application the explicit time stepping is becoming far too expensive, running more than 200 times slower. It must be noted, however, that such a comparison is not entirely fair as neither the code nor the model setup is optimised for explicit computation. For example, one should eliminate too small elements in the mesh, which has not been carried out. Nevertheless, in order to be able to capture the tidal dynamics in the estuary, sufficiently fine mesh is required [23] and such a setup is very demanding for explicit methods.

7. Conclusions

In this paper a novel fully implicit WD method has been proposed. Although the method relies on artificially moving the bathymetry in dry areas, both the numerical test and the real-world application in the Scheldt Estuary

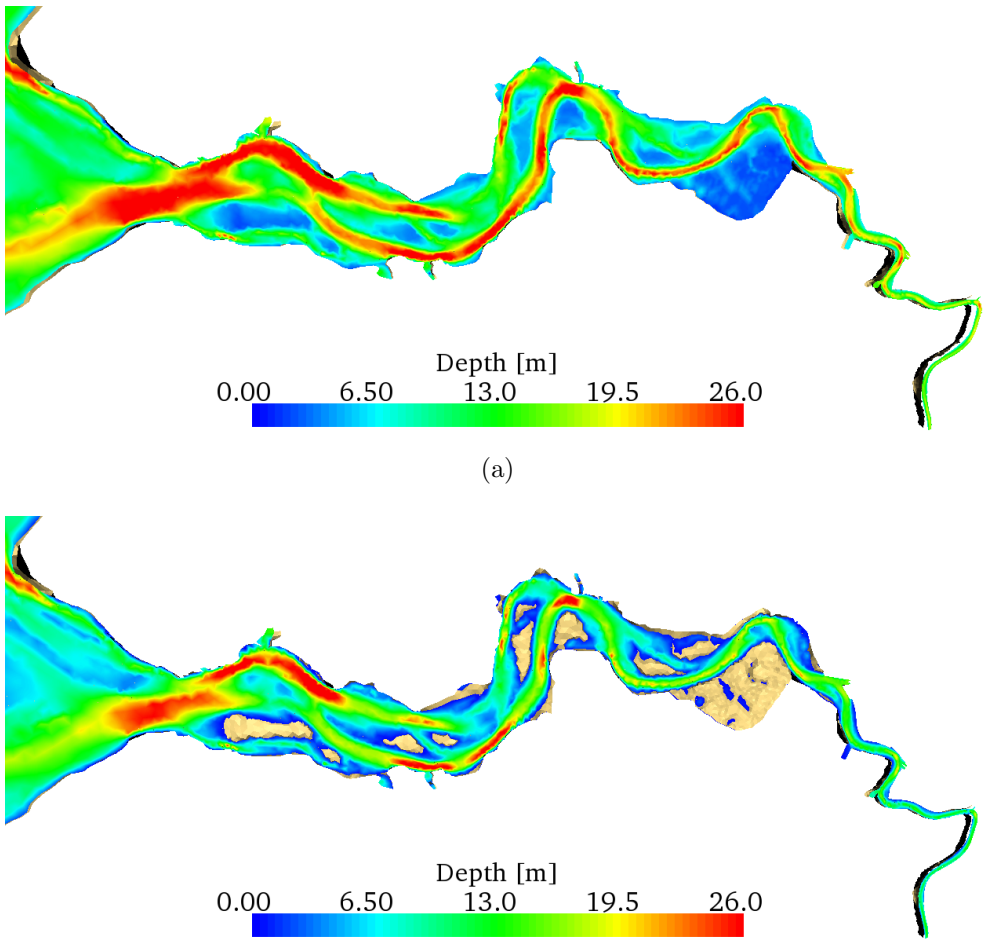


Figure 11: Wetting-drying in the Scheldt Estuary during spring tide. a) Depth at high water b) Depth at low water. Exposed sand banks are clearly visible. Maximum depth is roughly 60 m.

confirms that the WD processes are modelled with good accuracy. No spurious behaviour was noticed in the dry areas and wave propagation properties seem intact.

Although the moving bathymetry WD method resembles porous media approach, there are several differences. Since our model relies on artificial displacement of bathymetry, there is no need to introduce the concept of porosity nor properties of the porous layer. Generally this leads to a simpler

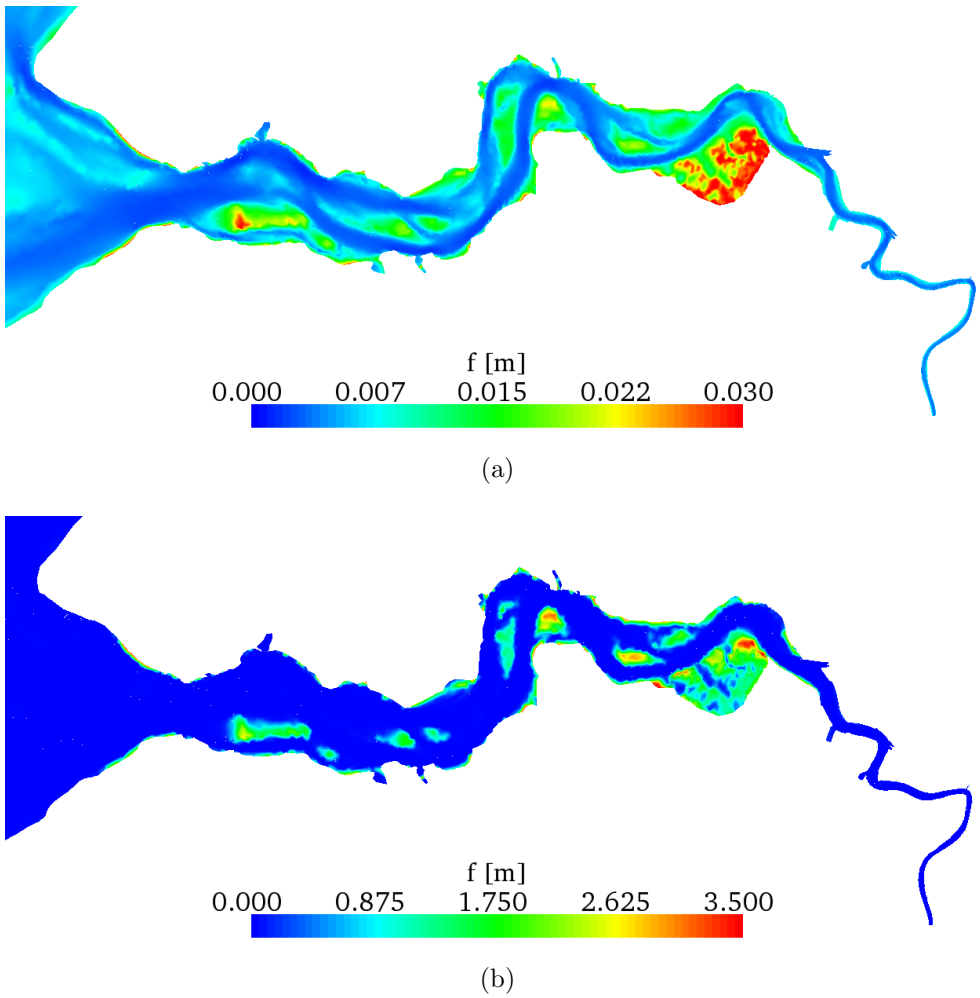
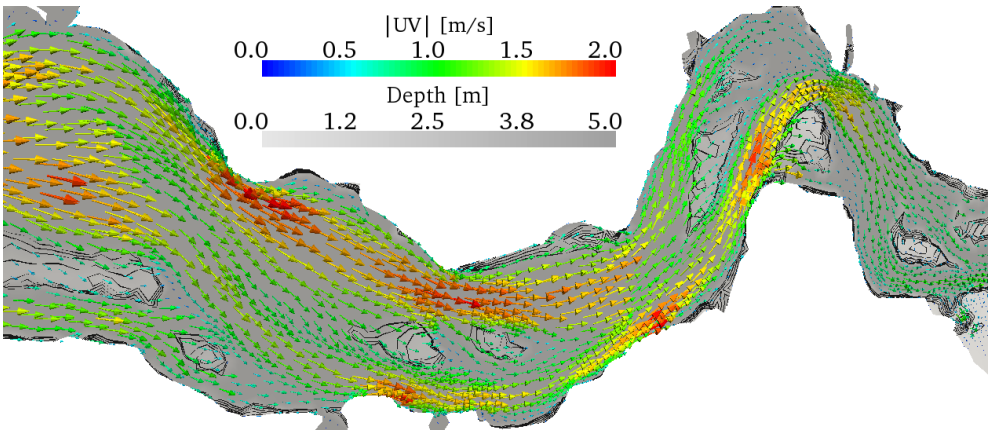


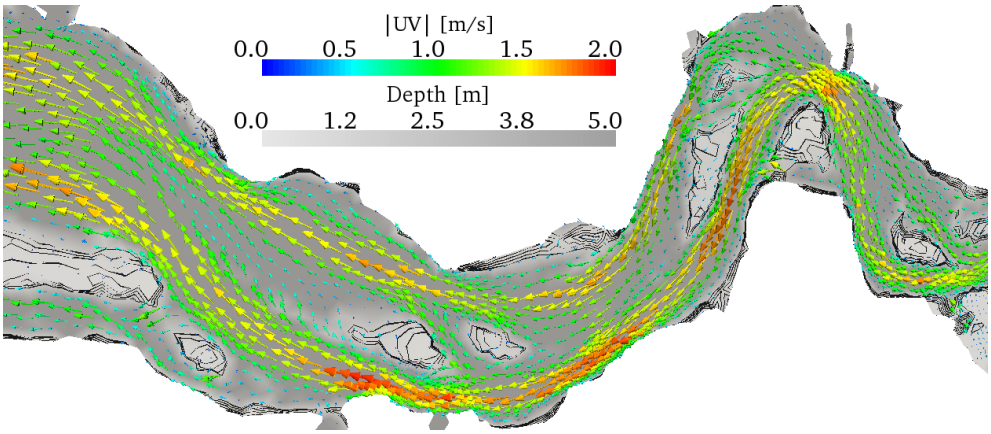
Figure 12: Bathymetry displacement $|\tilde{h} - h|$ during a) high water b) low water. The modification remains very small in the deep areas.

numerical formulation and fewer number of unknown parameters than in porous media WD methods [16, 17, 15]. Indeed the moving bed was defined by the means of a single-parameter function, which has proved out to be sufficient in all the various test cases.

Since the main field of application of the SLIM is long-term transport simulations simulations, we have paid special attention to ensure strict mass conservation and tracer consistency properties, which (especially for the latter) is not common in WD literature. The numerical tests confirm that



(a)

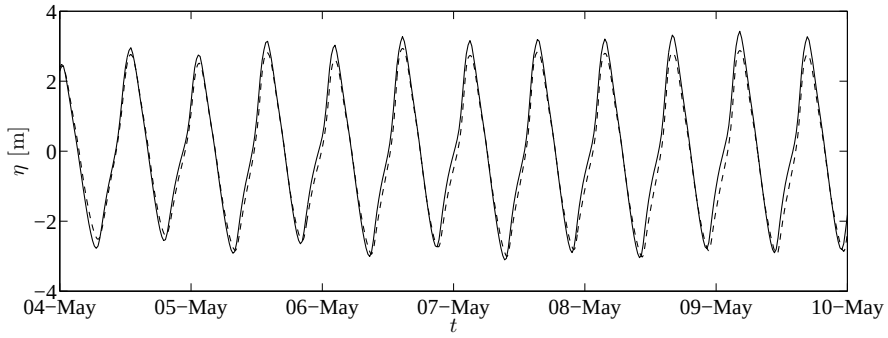


(b)

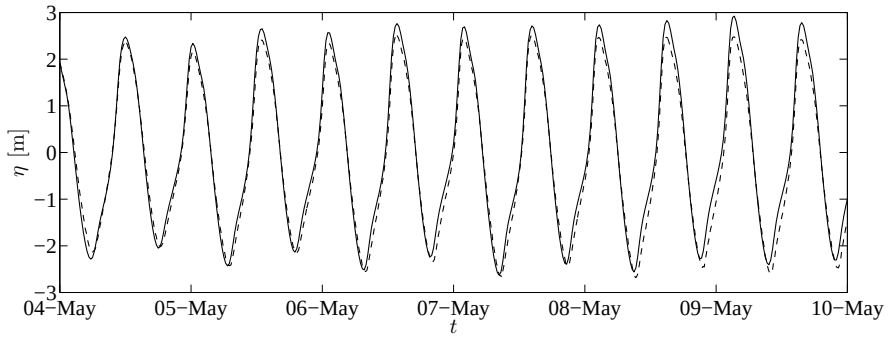
Figure 13: Velocity fields at a) flood and b) ebb tide.

the scheme is mass conservative, tracer mass is conserved up to numerical precision and no spurious transport due to the WD method appears.

The main advantage of the presented method, however, lies in the computational efficiency due to the implicit time integration. Indeed, it is not obvious to develop a fully implicit WD method that is strictly mass conservative and consistent with tracer equation, especially in FE framework. To be able to really exploit the advantages of implicit time integration, higher order DIRK schemes are suggested here. When using long time steps of order ten minutes, simple implicit Euler time marching scheme proved out to be



(a) Bath



(b) Terneuzen

Figure 14: Simulated water elevation (solid line) versus measurements (dashed line) in two stations.

far too dissipative.

In the Scheldt application the total CPU time was reduced by a factor of 200 in comparison to explicit time stepping, that is often required in WD simulations. As the unstructured grid FE models tend to be slower than established FD models, this speed-up is essential in practical applications.

Lastly it should be stressed that although this paper deals with FE implementation, other formulations are not excluded, because the method can be described already at the level of the primitive equations.

Future work will include the development of a better representation of the bathymetry in the approximate Riemann solver and also an extension to higher order elements, which is not possible for most WD methods.

Acknowledgements

Tuomas Kärnä and Jonathan Lambrechts are Research fellows with the Belgian Fund for Research in Industry and Agriculture (FRIA). Richard Comblen is a Research fellow with the Belgian National Fund for Scientific Research (FNRS). Eric Deleersnijder is a Research associate with the Belgian National Fund for Scientific Research (FNRS). The research was conducted within the framework of the Interuniversity Attraction Pole TIMOTHY (IAP VI.13), funded by the Belgian Science Policy (BELSPO), and the programme ARC 04/09-316, funded by the Communauté Française de Belgique. Eric Deleersnijder is indebted to Guus Stelling and Julie Pietrzak for useful discussions on wetting-drying algorithms.

Appendix

Runge-Kutta schemes are defined by the coefficients $a_{i,j}$, b_j and c_i which are usually gathered in Butcher's tableau:

$$\begin{array}{c|cccc} c_1 & a_{1,1} & 0 & \dots & 0 \\ c_2 & a_{2,1} & a_{2,2} & \dots & 0 \\ c_3 & a_{3,1} & a_{3,2} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s,1} & a_{s,2} & \dots & a_{s,s} \\ \hline & b_1 & b_2 & \dots & b_s \end{array} .$$

Following [30, section 2.6] and [31, Appendix A], respectively, the implicit Runge-Kutta schemes are defined as follows.

DIRK22

$$\begin{array}{c|cc} \gamma & \gamma & 0 \\ 1 & 1-\gamma & \gamma \\ \hline & 1-\gamma & \gamma \end{array}$$

$$\gamma = (2-\sqrt{2})/2$$

ESDIRK64

$$\begin{array}{c|ccccccc} \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 \\ \frac{83}{250} & \frac{8611}{62500} & -\frac{1743}{31250} & \frac{1}{4} & 0 & 0 & 0 \\ \frac{31}{50} & \frac{5012029}{34652500} & -\frac{654441}{2922500} & \frac{174375}{388108} & \frac{1}{4} & 0 & 0 \\ \frac{17}{20} & \frac{15267082809}{155376265600} & -\frac{71443401}{120774400} & \frac{730878875}{902184768} & \frac{2285395}{8070912} & \frac{1}{4} & 0 \\ 1 & \frac{82889}{524892} & 0 & \frac{15625}{83664} & \frac{69875}{102672} & -\frac{2260}{8211} & \frac{1}{4} \\ \hline & \frac{82889}{524892} & 0 & \frac{15625}{83664} & \frac{69875}{102672} & -\frac{2260}{8211} & \frac{1}{4} \end{array}$$

[1] E. Marchandise, J.-F. Remacle, A stabilized finite element method using a discontinuous level set approach for solving two phase incompressible flows, Journal of Computational Physics 219 (2006) 780–800.

[2] R. J. Sobey, Wetting and drying in coastal flows, Coastal Engineering 56 (2009) 565–576.

[3] D. Yuan, B. Lin, R. Falconer, Simulating moving boundary using a linked groundwater and surface water flow model, Journal of Hydrology 349 (2008) 524–535.

[4] L. Zheng, C. Chen, H. Liu, A modeling study of the satilla river estuary, georgia. i: Flooding-drying process and water exchange over the salt marsh-estuary-shelf complex, Estuaries and Coasts 26 (2003) 651–669.

[5] M. Zijlema, G. S. Stelling, Efficient computation of surf zone waves using the nonlinear shallow water equations with non-hydrostatic pressure, Coastal Engineering 55 (2008) 780–790.

[6] P. D. Bates, J.-M. Hervouet, A new method for moving–boundary hydrodynamic problems in shallow water, Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences 455 (1999) 3107–3128.

[7] B. van't Hof, E. A. H. Vollebregt, Modelling of wetting and drying of shallow water using artificial porosity, International Journal for Numerical Methods in Fluids 48 (2005) 1199–1217.

[8] S. Bunya, E. J. Kubatko, J. J. Westerink, C. Dawson, A wetting and drying treatment for the runge-kutta discontinuous galerkin solution to the shallow water equations, Computer Methods in Applied Mechanics and Engineering 198 (2009) 1548–1562.

[9] O. Gourgue, R. Comblen, J. Lambrechts, T. Kärnä, V. Legat, E. Deleersnijder, A flux-limiting wetting-drying method for finite-element shallow-water models, with application to the scheldt estuary, Advances in Water Resources 32 (2009) 1726–1739.

[10] I. Nikolos, A. Delis, An unstructured node-centered finite volume scheme for shallow water flows with wet/dry fronts over complex topography,

Computer Methods in Applied Mechanics and Engineering 198 (2009) 3723–3750.

[11] M. Heniche, Y. Secretan, P. Boudreau, M. Leclerc, A two-dimensional finite element drying-wetting shallow water model for rivers and estuaries, Advances in Water Resources 23 (2000) 359–372.

[12] V. Casulli, A high-resolution wetting and drying algorithm for free-surface hydrodynamics, International Journal for Numerical Methods in Fluids 60 (2009) 391–408.

[13] G. S. Stelling, S. P. A. Duinmeijer, A staggered conservative scheme for every froude number in rapidly varied shallow water flows, International Journal for Numerical Methods in Fluids 43 (2003) 1329–1354.

[14] J. T. C. Ip, D. R. Lynch, C. T. Friedrichs, Simulation of estuarine flooding and dewatering with application to Great Bay, New Hampshire, Estuarine, Coastal and Shelf Science 47 (1998) 119–141.

[15] C. Nielsen, C. Apelt, Parameters affecting the performance of wetting and drying in a two-dimensional finite element long wave hydrodynamic model, Journal of Hydraulic Engineering 129 (2003) 628–636.

[16] A. B. Kennedy, Q. Chen, J. T. Kirby, R. A. Dalrymple, Boussinesq modeling of wave transformation, breaking, and runup. i: 1d, Journal of Waterway, Port, Coastal, and Ocean Engineering 126 (2000) 39–47.

[17] Y. Jiang, O. W. Wai, Drying-wetting approach for 3d finite element sigma coordinate model for estuaries with large tidal flats, Advances in Water Resources 28 (2005) 779–792.

[18] A. Defina, Two-dimensional shallow flow equations for partially dry areas, Water Resources Research 36 (2000) 3251–3264.

[19] H. Burchard, K. Bolding, M. R. Villarreal, Three-dimensional modelling of estuarine turbidity maxima in a tidal estuary, Ocean Dynamics 54 (2004) 250–265.

[20] R. Alonso, M. Santillana, C. Dawson, On the diffusive wave approximation of the shallow water equations, European Journal of Applied Mathematics 19 (2008) 575–606.

[21] M. Santillana, C. Dawson, A local discontinuous galerkin method for a doubly nonlinear diffusion equation arising in shallow water modeling, Computer Methods in Applied Mechanics and Engineering 199 (2010) 1424–1436.

[22] M. Castro, A. Ferreiro, J. García-Rodriguez, J. González-Vida, J. Macías, C. Parés, M. Elena Vázquez-Cendón, The numerical treatment of wet/dry fronts in shallow flows: application to one-layer and two-layer systems, Mathematical and Computer Modelling 42 (2005) 419–439.

[23] B. de Brye, A. de Brauwere, O. Gourgue, T. Kärnä, J. Lambrechts, R. Comblen, E. Deleersnijder, A finite-element, multi-scale model of the scheldt tributaries, river, estuary and rofi, Coastal Engineering 57 (2010) 850–863.

[24] B. Rivière, Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations: Theory and Implementation, number 35 in Frontiers in Applied Mathematics, SIAM, 2008.

[25] P. L. Roe, Approximate riemann solvers, parameter vectors, and difference schemes, Journal of Computational Physics 135 (1997) 250–258.

[26] D. H. Zhao, H. W. Shen, J. S. Lai, G. Q. T. III, Approximate riemann solvers in fvm for 2d hydraulic shock wave modeling, Journal of Hydraulic Engineering 122 (1996) 692–702.

[27] R. Comblen, J. Lambrechts, J.-F. Remacle, V. Legat, Practical evaluation of five partly discontinuous finite element pairs for the non-conservative shallow water equations, International Journal for Numerical Methods in Fluids (2009).

[28] A. Ern, S. Piperno, K. Djadel, A well-balanced runge-kutta discontinuous galerkin method for the shallow-water equations with flooding and drying, International Journal for Numerical Methods in Fluids 58 (2008) 1–25.

[29] D. L. George, Augmented Riemann solvers for the shallow water equations over variable topography with steady states and inundation, Journal of Computational Physics 227 (2008) 3089–3113.

[30] U. M. Ascher, S. J. Ruuth, R. J. Spiteri, Implicit–explicit Runge–Kutta methods for time-dependent partial differential equations, Applied Numerical Mathematics: Transactions of IMACS 25 (1997) 151–167.

[31] G. Jothiprasad, D. J. Mavriplis, D. A. Caughey, Higher-order time integration schemes for the unsteady navier-stokes equations on unstructured meshes, Journal of Computational Physics 191 (2003) 542–566.

[32] Y. Saad, M. H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM Journal on Scientific and Statistical Computing 7 (1986) 856–869.

[33] L. White, E. Deleersnijder, V. Legat, Tracer conservation for three-dimensional, finite-element, free-surface, ocean modeling on moving prismatic meshes, Monthly Weather Review 136 (2008) 420–442.

[34] A. Balzano, Evaluation of methods for numerical simulation of wetting and drying in shallow water flow models, Coastal Engineering 34 (1998) 83–107.

[35] M. Leclerc, J.-F. Bellemare, G. Dumas, G. Dhatt, A finite element model of estuarian and river flows with moving boundaries, Advances in Water Resources 13 (1990) 158–168.

[36] P. Brufau, M. E. Vázquez-Cendón, P. García-Navarro, A numerical model for the flooding and drying of irregular domains, International Journal for Numerical Methods in Fluids 39 (2002) 247–275.

[37] W. C. Thacker, Some exact solutions to the nonlinear shallow-water wave equations, Journal of Fluid Mechanics Digital Archive 107 (1981) 499–508.

[38] P. J. Lynett, T.-R. Wu, P. L. F. Liu, Modeling wave runup with depth-integrated equations, Coastal Engineering 46 (2002) 89–107.

[39] P. D. Bates, M. S. Horritt, Computational Fluid Dynamics, John Wiley & Sons, Ltd, pp. 121–146.

[40] W. Baeyens, B. van Eck, C. Lambert, R. Wollast, L. Goeyens, General description of the Scheldt estuary, Hydrobiologia 366 (1997) 1–14.

[41] C. M. Swinkels, C. M. C. J. L. Jeuken, Z. B. Wang, R. J. Nicholls, Presence of connecting channels in the western scheldt estuary, Journal of Coastal Research 25 (2009) 627–640.

[42] C. Geuzaine, J.-F. Remacle, Gmsh: A 3-d finite element mesh generator with built-in pre- and post-processing facilities, International Journal for Numerical Methods in Engineering 79 (2009) 1309–1331.

[43] J. Lambrechts, R. Comblen, V. Legat, C. Geuzaine, J.-F. Remacle, Multiscale mesh generation on the sphere, Ocean Dynamics 58 (2008) 461–473.

[44] C. Amante, B. W. Eakins, Etopo1 1 arc-minute global relief model: Procedures, data sources and analysis, NOAA Technical Memorandum NESDIS NGDC-24 (2009) 19.

[45] G. Egbert, A. Bennett, M. Foreman, Topex/poseidon tides estimated using a global inverse model, Journal of Geophysical Research 99 (1994) 24821–24852.

[46] E. Kalnay, M. Kanamitsua, R. Kistlera, W. Collinsa, D. Deavena, L. Gandina, M. Iredella, S. Sahaa, G. Whitea, J. Woollena, Y. Zhua, A. Leetmaaa, R. Reynoldsa, M. Chelliahb, W., H. W. Ebisuzakib, J. Janowiakb, K. C. Mob, C. Ropelewskib, J. Wangb, R. Jennec, D. Joseph, The NCEP/NCAR 40-Year Reanalysis Project, Bull. Amer. Meteor. Soc. 77 (1996) 437–431.