A limiter-based well-balanced discontinuous Galerkin method for shallow-water flows with wetting and drying: Triangular grids

Stefan Vater[a,b,*], Nicole Beisiegel[c], Jörn Behrens[a,b]

[a]*Department of Mathematics, Universität Hamburg, Bundesstraße 55, 20146 Hamburg, Germany*
[b]*CEN – Center for Earth System Research and Sustainability, Universität Hamburg, Grindelberg 5, 20144 Hamburg, Germany*
[c]*School of Mathematics & Statistics, University College Dublin, Science Centre – North, G.03, Belfield, Dublin 4, Ireland*

Abstract

A novel wetting and drying treatment for second-order Runge-Kutta discontinuous Galerkin (RKDG2) methods solving the non-linear shallow water equations is proposed. It is developed for two-dimensional triangular meshes and is based on a limiting strategy to accurately represent moving shorelines. In order to maintain uniform time steps subject to a CFL stability constraint, the limiting is not directly applied to the primary variables (fluid depth and momentum) but to total height and velocity. While mass-conservation is inherent to DG methods, well-balancing and non-negativity of the fluid depth is preserved by a small flux modification and a positivity-preserving limiting procedure. The scheme comprises only one free parameter, to which it is not sensitive in terms of stability. A number of test cases, ranging from analytical tests to near-realistic laboratory benchmarks, demonstrate the performance of the numerical method for inundation applications. In particular, super-linear convergence, mass-conservation, well-balancedness, and stability are verified.

Keywords: Shallow water equations, Discontinuous Galerkin Methods, Wetting and Drying, Limiter, Well-balanced Scheme

1. Introduction

In recent years discontinuous Galerkin (DG) methods have gained a lot of scientific interest within the geophysical fluid dynamics (GFD) modeling community, since they have preferable properties with respect to conservation, geometric flexibility and accuracy for coastal ocean modeling applications [see, e.g., 7, 14, 17]. In order to successfully compute near- and onshore propagation of ocean waves, depth-integrated equations such as the shallow water equations are usually employed. Although these are derived under the assumption that vertical velocities are negligible, they efficiently model large scale horizontal flows with high accuracy. Computational problems occur in the coastal area, where the shoreline represents a moving boundary condition, which must be considered in the numerical scheme. Here, it is essential to construct a numerical scheme, which concurrently fulfills the essential requirements for accurate coastal modeling:

- conservation of mass,

- accurate representation of the shoreline (i.e., of wetting and drying), and

- maintenance of steady states (well-balancedness).

While conservation properties are inherent in DG methods, it is not so obvious how to maintain these in the presence of changing wet and dry boundaries. Wetting formerly dry cells and drying wet cells is

---

naturally achieved by fluxes. However, special care needs to be taken in partially dry cells to preserve well-balancedness and mass conservation at the same time.

Further difficulties arise near the shoreline. Since the discretization may not exactly resolve the wet/dry interface, artificial gradients arise in the surface elevation which renders the method unbalanced. Additionally, the fluid depth becomes zero at the shoreline, and one has to ensure its non-negativity in the discretization. Otherwise, the shallow water equations become ill-posed. A third problem is the computation of the velocity, which is needed in the flux computation and obtained by the quotient of momentum and fluid depth, when the equations are discretized in conservation form. Near the shore, both, fluid depth and momentum usually become small, which results in a badly conditioned velocity computation.

Well-balancedness is especially important in GFD applications, since often subtle balances drive the motion. For a "lake-at-rest" situation, where fluid velocities vanish and a constant balanced water level is assumed, only source terms and topography gradients may remain as non-trivial terms in a shallow-water approximation. A well-balanced numerical scheme leaves these terms balanced on a discrete level, too. This discrete property is paramount, since otherwise spurious waves may be triggered in areas where this balance is violated.

Although the moving shoreline can be accurately incorporated into the numerical scheme by adjusting the computational domain, its implementation is difficult and often only simple flow configurations have been successfully considered with this approach [4]. Only recently, an Arbitrary Lagrangian Eulerian (ALE) approach together with a moving r-adaptive mesh was applied to more complex flow situations [1]. Most commonly, an Eulerian approach is considered, where the mesh points do not move and the numerical scheme is applied to all cells, regardless if they are wet or dry. This is often combined with a so-called thin-layer approach, where a point is considered dry, if the fluid depth drops below a given tolerance. Using this tolerance, the velocity can be set directly to zero in such a situation and the problem of possibly dividing by a zero fluid depth is circumvented.

While for finite volume (FV) methods the wetting and drying problem has been thoroughly considered [see, e.g., 2, 27, 33, 41], there are still only a few approaches using DG methods. Bokhove [8] employed a mixed Eulerian-Lagrangian element approach near the moving boundary. A nodal flux modification technique was introduced by Gourgue et al. [19], and further developed in Kärnä et al. [21], who employed a modified bathymetry in order to control the numerical behavior. The most common approach is based on slope-limiting. Kesserwani & Liang [22] directly adapted the idea of hydrostatic reconstruction [2] to a Runge-Kutta discontinuous Galerkin (RKDG) discretization by differentiating between fully wet and dry cells. Other works introduced a scaling around the mean within each cell to obtain positivity and mass conservation at the same time [10, 15, 43].

Most of the mentioned studies concentrated on the positivity of the fluid depth and the well-balancing property, so far. The ill-conditioned computation of the velocity has only been dealt with marginally. Although the solution can be advanced to a new time level by satisfying a suitable CFL condition in an explicit method, spurious velocities near the wet/dry interface can result in a prohibitively small time step. This problem is solved in our approach by incorporating the resulting velocities into a limiting procedure. The idea is borrowed from FV methods, where interface values are often reconstructed from other variables than the primary prognostic variables (see van Leer [25] and references therein). For FV methods this does not pose a problem, since the reconstructed values are only used for the flux computation. In DG methods the situation is more complicated, since the whole in-cell solution is limited and used throughout the computations. Therefore, one is usually bound to use the primary variables for limiting. Here, we develop an approach to modify the primary variables based on other secondary variables. We stress that our scheme maintains common time step restrictions unlike implicit methods, such as Meister & Ortleb [31].

In our approach a previously developed one-dimensional limiter [39] is adopted to the multi-dimensional case using triangular grids. It is based on limiting the total fluid height $H = h + b$ and the velocity in the momentum-based flux computation. This approach is based on the original idea of hydrostatic reconstruction for FV methods [2, 32, 44]. We employ two Barth/Jespersen-type limiters – the original edge-based version [3] and a modified vertex-based development [24], which is particularly suitable for triangular grids. We employ a set of six commonly used test cases to demonstrate stability, accuracy, convergence, well-balancedness and mass-conservation of our scheme in the presence of moving wet-dry interfaces.

This manuscript is organized into four further sections. Following this introduction, we briefly introduce the equations and review the DG discretization scheme. We then detail the wetting and drying algorithm in section 3, before demonstrating rigorously the properties of the limiting approaches with numerical examples (section 4). We conclude with final remarks and an outlook for future applications.

2. The shallow water equations and their RKDG discretization

Two-dimensional water waves in shallow water can be described accurately by the nonlinear shallow water equations. They are derived from the principles of conservation of mass and momentum and can compactly be written in conservative form

$$\boldsymbol{U}_t + \nabla \cdot \boldsymbol{F}(\boldsymbol{U}) = \boldsymbol{S}(\boldsymbol{U}), \tag{1}$$

where the vector of unknowns is given by $\boldsymbol{U} = (h, h\boldsymbol{u})^\mathsf{T}$. Here and below, we have written the partial derivative with respect to time t as $\boldsymbol{U}_t \equiv \frac{\partial \boldsymbol{U}}{\partial t}$ and the divergence with respect to the spatial variable $\boldsymbol{x} = (x, y)^\mathsf{T}$ as $\nabla \cdot \boldsymbol{F}$, which is applied to each component of $\boldsymbol{F}$. The quantity $h = h(\boldsymbol{x}, t)$ denotes the fluid depth of a uniform density water layer and $\boldsymbol{u} = \boldsymbol{u}(\boldsymbol{x}, t)$ is the depth-averaged horizontal velocity. The flux function is defined by $\boldsymbol{F}(\boldsymbol{U}) = \begin{pmatrix} h\boldsymbol{u} \\ h\boldsymbol{u} \otimes \boldsymbol{u} + \frac{g}{2}h^2\boldsymbol{I_2} \end{pmatrix}$, where g is the gravitational constant and $\boldsymbol{I_2}$ the 2×2 identity matrix. Furthermore, the bathymetry (bottom topography) $b = b(\boldsymbol{x})$ is represented by the source term $\boldsymbol{S}(\boldsymbol{U}) = (0, gh\nabla b)^\mathsf{T}$.

Note that realistic simulations might require further source terms such as bottom friction or Coriolis forcing, since these can significantly influence the wetting and drying as well as the propagation behavior. However, as the focus of this paper is the presentation of novel slope limiting techniques that are used for robust inundation modeling, we will only focus on the inviscid case and leave the inclusion of further source terms for future investigation.

For the discretization using the DG method, the governing equations are solved on a polygonal domain $\Omega \subset \mathbb{R}^2$, which is divided into conforming elements (triangles) C_i. On each element, the equations (1) are multiplied by a test function φ and integrated. Integration by parts of the flux term leads to the weak DG formulation

$$\int_{C_i} \boldsymbol{U}_t \varphi \, \mathrm{d}\boldsymbol{x} - \int_{C_i} \boldsymbol{F}(\boldsymbol{U}) \cdot \nabla\varphi \, \mathrm{d}\boldsymbol{x} + \int_{\partial C_i} \boldsymbol{F}^*(\boldsymbol{U}) \cdot \boldsymbol{n} \, \varphi \, \mathrm{d}\sigma = \int_{C_i} \boldsymbol{S}(\boldsymbol{U})\varphi \, \mathrm{d}\boldsymbol{x} \,, \tag{2}$$

where $\boldsymbol{n}$ is the outward pointing normal at the edges of element C_i. The interface flux $\boldsymbol{F}^*$ is not defined in general, as the solution can have different values at the interface in the adjacent elements. This problem is circumvented in the discretization by using the (approximate) solution of the corresponding Riemann problem. For the simulations in this study we used the Rusanov solver [38]. Another integration by parts of the volume integral over the flux yields the strong DG formulation

$$\int_{C_i} \boldsymbol{U}_t \varphi \, \mathrm{d}\boldsymbol{x} + \int_{C_i} \nabla \cdot \boldsymbol{F}(\boldsymbol{U}) \, \varphi \, \mathrm{d}\boldsymbol{x} + \int_{\partial C_i} (\boldsymbol{F}^*(\boldsymbol{U}) - \boldsymbol{F}(\boldsymbol{U})) \cdot \boldsymbol{n} \, \varphi \, \mathrm{d}\sigma = \int_{C_i} \boldsymbol{S}(\boldsymbol{U})\varphi \, \mathrm{d}\boldsymbol{x}, \tag{3}$$

which recovers the original differential equations within a cell, but with an additional term accounting for the jumps at the interfaces. We will deal with both formulations (2) and (3) in this work.

The system of equations is further discretized using a semi-discretization in space with a piecewise polynomial ansatz for the discrete solution components and test functions φ_k. We obtain formally second-order accuracy by using piecewise linear functions, which are represented by nodal Lagrange basis functions [17, 20]. The solution in one element is then given by $\boldsymbol{U}(\boldsymbol{x}, t) = \sum_j \tilde{\boldsymbol{U}}_j(t)\varphi_j(\boldsymbol{x})$, where $(\tilde{\boldsymbol{U}}_j(t))_j$ is the vector of degrees of freedom. The integrals are solved with a 3-point Gauß-Legendre quadrature in order to obtain an exact result for most of the integral terms. With this discretization in space, we obtain a system of ordinary differential equations (ODEs) for $\tilde{\boldsymbol{U}}_j(t)$ in each element C_i of the general form

$$\frac{\partial \tilde{\boldsymbol{U}}_\Delta}{\partial t} = \boldsymbol{H}_\Delta(\tilde{\boldsymbol{U}}_\Delta),$$

3

where now the index Δ indicates the numerical resolution. The resulting system for the degrees of freedom of all cells $\tilde{\boldsymbol{U}}_\Delta$ is then solved using a total-variation diminishing (TVD) s-stage Runge-Kutta scheme [18, 34], which is of the general form

$$
\tilde{\boldsymbol{U}}_\Delta^{(0)} = \tilde{\boldsymbol{U}}_\Delta^n
$$

$$
\tilde{\boldsymbol{U}}_\Delta^{(p)} = \Pi_\Delta \left\{ \sum_{q=0}^{p-1} \alpha_{pq} \tilde{\boldsymbol{U}}_\Delta^{(q)} + \beta_{pq} \Delta t^n \boldsymbol{H}_\Delta \left(\tilde{\boldsymbol{U}}_\Delta^{(q)} \right) \right\} \quad \text{for } p = 1 \ldots s
$$

$$
\tilde{\boldsymbol{U}}_\Delta^{n+1} = \tilde{\boldsymbol{U}}_\Delta^{(s)}
$$

with Runge-Kutta coefficients α_{pq} and β_{pq} and a time step size $\Delta t^n = t^{n+1} - t^n$. $\boldsymbol{H}_\Delta(\tilde{\boldsymbol{U}}_\Delta)$ represents the right hand side of the ODE extended to all cells. The limiter Π_Δ, which is applied in each Runge-Kutta stage p to an intermediate solution $\boldsymbol{U}_\Delta^{(p)}$, is usually employed to stabilize the scheme in case of discontinuities. However, as stated above it can also be used for dealing with the problem of wetting and drying. In the remainder of this article, we employ Heun's method, which is the second-order representative of a standard Runge-Kutta TVD scheme. For the discretization in space we use piecewise linear trial and test functions. This completes the description of the Runge-Kutta discontinuous Galerkin (RKDG) method.

In order to achieve well-balancedness in equation (2) exact quadrature of the terms involving $\frac{g}{2} h^2$ and the source term are a basic requirement [40]. At the cell interfaces no problems occur, if a continuous representation for the bottom topography is used. Otherwise, one can use the technique introduced in [40, 43], which is based on hydrostatic reconstruction of the interface values and adds a higher order correction to the source term [see 2].

3. Wetting and drying algorithm

The numerical treatment of wetting and drying events in the context of shallow water equations involves some difficulties, which have to be considered in the construction of a numerical method. First, the discretization might not accurately resolve the shoreline. In case of a DG scheme with piecewise polynomial functions within each cell, a shoreline crossing a cell can then introduce artificial gradients in the surface elevation rendering the method unbalanced. A second problem concerns the positivity of the fluid depth, which must be ensured. Otherwise the shallow water equations are undefined. While these problems have been thoroughly discussed in the context of DG methods in the literature [10, 15, 19, 22, 42], a less investigated problem arises in the computation of the velocity which is needed in the flux evaluation. Near the shoreline, the primary variables h and $(h\boldsymbol{u})$ usually become very small, while the velocity, which is computed by the ratio $(h\boldsymbol{u})/h$, is of order 1. Therefore, one either has to pre-condition the primary variables or has to modify the velocity itself in order to get a reasonable velocity approximation in nearly dry regions. In the following we describe how we deal with the described problems. The approach includes a small flux modification applied in semi-dry cells and a specially designed limiter, which ensures positivity of h and a stable velocity computation.

3.1. The wet/dry interface

One of the key properties of a numerical method for its application to geophysical problems is its ability to represent stationary states like the "lake at rest" in case of the shallow water equations without spurious waves. So-called "well-balanced" schemes can reproduce such states with high accuracy, independent of their approximation order. For the lake at rest a DG approximation of the total height $H = h + b$ with piecewise polynomial data introduces some artificial gradients in semi-dry cells where the shoreline does not coincide with cell interfaces. This is due to the non-negativity of the fluid depth and the imposed smoothness of the numerical approximation within a cell. These gradients must be neglected in a numerical scheme. On the other hand, there can be semi-dry cells that are approximated physically correctly (e.g., in a dam-break situation where the water comes from higher elevation) and must be distinguished from the lake-at-rest case.
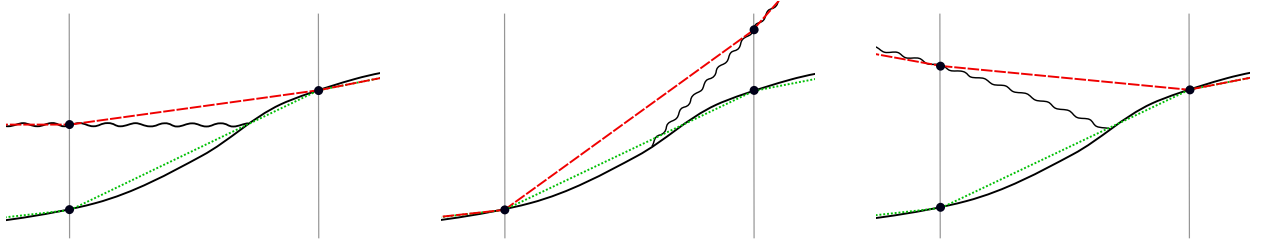
Figure 1: Discretization of different semi-dry cell types using the discontinuous Galerkin scheme with piecewise linear elements (red solid: surface elevation, green dotted: bottom topography). Displayed are a configuration where the gradient in surface elevation should be neglected (left) and two situations which are discretized physically correct (middle and right).

We do this by comparing the maximum total height with the maximum bottom topography within a cell. If the maximum total height is not larger than the maximum bottom topography within cell C_i, i.e.,

$$\max_{x \in C_i} H(x) - \max_{x \in C_i} b(x) < \mathrm{TOL}_{\mathrm{wet}} \ , \tag{4}$$

we are possibly in a steady state and therefore neglect the gradient in H within the cell. Here we introduced the parameter $\mathrm{TOL}_{\mathrm{wet}}$, which is a threshold for the fluid depth under which a point is considered dry. Otherwise, if (4) is not fulfilled, the cell is treated just as a completely wet cell.

Ignoring the gradient in total height H in cells where (4) is fulfilled, is equivalent to setting g to zero there. For the strong DG flux formulation (3) this modification must only be done for the volume integrals over C_i in the flux computation and the evaluation of the source term. In the case of the lake at rest the flux difference $\boldsymbol{F}^*(\boldsymbol{U}) - \boldsymbol{F}(\boldsymbol{U})$, which is computed on the interfaces, automatically cancels. This is different for the weak DG formulation (2). Setting g to zero in cells where the solution satisfies (4), also correctly adjusts the volume integrals, but the flux contributions from the wet interfaces of a semi-dry cell should not be neglected in general, because they are also present in the computation of the update for the adjacent wet cell. For these interfaces an additional flux term is introduced which balances the numerical flux. It includes only the gravitational part and is based on the fluid depth of the semi-dry cell at the wet interface. The momentum equation in a semi-dry cell then reads

$$\int_{C_i} \varphi (h\boldsymbol{u})_t \, \mathrm{d}\boldsymbol{x} - \int_{C_i} \nabla \varphi \cdot \left(h\boldsymbol{u} \otimes \boldsymbol{u} + \frac{g}{2} h^2 \boldsymbol{I_2} \right) \mathrm{d}\boldsymbol{x} +$$
$$\sum_{j=1}^{3} \int_{E_j^i} \left(\boldsymbol{F}_{h\boldsymbol{u}}^* - \frac{g}{2} (h^-)^2 \boldsymbol{I_2} \right) \cdot \boldsymbol{n} \ \varphi \, \mathrm{d}\sigma = - \int_{C_i} \varphi g h \nabla b \, \mathrm{d}\boldsymbol{x} \ ,$$

where h^- is the value of the fluid depth based on cell C_i, and E_j^i, $j \in \{1, 2, 3\}$ are the three edges of C_i. If this discretization is applied to the lake at rest with $\boldsymbol{u} \equiv \boldsymbol{0}$, then all the edge contributions in a semi-dry cell vanish. For dry edges this is because h is zero there, and for wet cells the difference computed under the integral cancels to zero. Note, however, that wellbalancing is easier accomplished by using the strong form, and this form also leads to slightly better results as we will see in section 4.

3.2. Limiting of the fluid depth

To ensure positivity of the fluid depth and a well-conditioned computation of the velocity, we employ a specially designed limiter. This limiter is a direct generalization of the one-dimensional limiter described in Vater et al. [39] to the two-dimensional case of triangular meshes.

Since the limiting modifies the solution itself in a DG method, and not only the values for the computation of the interface fluxes as in finite volume methods, further care must be taken for well-balancing. Therefore, we require that the limiter does not alter the steady state of the lake at rest. A family of limiters which fulfill this requirement are the ones of Barth/Jespersen-type [3], when limiting in total height $H = h + b$. These

5

limiters modify the solution within a cell, such that it does not exceed the maximum and minimum of the cell mean values of adjacent cells. In this work we study the original version by Barth and Jespersen, which incorporates the cells which are connected by a common edge to the cell under consideration. Additionally, we consider a generalization for triangular grids, which was introduced by Kuzmin [24]. This limiter incorporates the cells which are connected by a common vertex for comparison. We refer to these two versions as the "edge-based" and the "vertex-based" limiter, respectively.

Given the cell average or centroid value $H_c = \overline{H} = \overline{h + b}$ of the total height, the piecewise linear in-cell distribution can be described by $H(x) = H_c + (\nabla H)_c \cdot (x - x_c)$. A Barth/Jespersen-type limiter modifies this to

$$\hat{H}(x) = H_c + \alpha_e (\nabla H)_c \cdot (x - x_c), \quad 0 \leq \alpha_e \leq 1,$$

where α_e is chosen, such that the reconstructed solution is bounded by the maximum and minimum centroid values of a given cell neighborhood:

$$H_c^{\min} \leq \hat{H}(x) \leq H_c^{\max}.$$

For the original Barth/Jespersen limiter this cell neighborhood is given by the considered cell and the three cells sharing an edge with this cell. In case of the limiter described by Kuzmin [24], the cell neighborhood is given by the considered cell and the surrounding cells sharing a vertex with this cell. The correction factor is explicitly defined as

$$\alpha_e = \min_i \begin{cases} \min\left\{1, \frac{H_c^{\max} - H_c}{H_i - H_c}\right\}, & \text{if } H_i - H_c > 0 \\ 1, & \text{if } H_i - H_c = 0 \\ \min\left\{1, \frac{H_c^{\min} - H_c}{H_i - H_c}\right\}, & \text{if } H_i - H_c < 0 \end{cases}$$

where H_i are the in-cell values of H at the three vertices of the triangle. The limited fluid depth is then recovered by $\hat{h} = \hat{H} - b$.

For positivity of fluid depth, we follow Bunya et al. [10] and Xing & Zhang [42]. The latter show that under a certain CFL condition positivity in the mean can be guaranteed by using a TVD Runge-Kutta scheme applied to an originally positive solution. The application of a "positivity preserving" limiter then adjusts the solution, such that it is positive everywhere. Let us express the piecewise linear function $\hat{h}$ by its nodal representation with Lagrange basis functions

$$\hat{h}(x,y) = \sum_{i=1}^{3} \hat{h}_i \, \varphi_i(x,y) \quad \text{for } (x,y) \in \Omega_k,$$

where we take as nodes the vertices of the triangle. Then positivity is obtained by enforcing positivity for the nodal values. Denoting the final limited values by $h^{\lim}$ and $h_i^{\lim}$ with $h^{\lim} = \sum h_i^{\lim} \varphi_i$, $h_i^{\lim}$ is determined by the following procedure: If $h_i^{\lim} \geq 0 \; \forall i \in \{1,2,3\}$, then

$$h_i^{\lim} = \hat{h}_i, \; i \in \{1,2,3\}.$$

Otherwise we determine the order of nodal indices $n_i \in \{1,2,3\}$ that satisfy $\hat{h}_{n_1} \leq \hat{h}_{n_2} \leq \hat{h}_{n_3}$ and compute the values in the following sequence:

$$\begin{aligned} h_{n_1}^{\lim} &= 0 \\ h_{n_2}^{\lim} &= \max\{0, \hat{h}_{n_2} - (h_{n_1}^{\lim} - \hat{h}_{n_1})/2\} \\ h_{n_3}^{\lim} &= \hat{h}_{n_3} - (h_{n_1}^{\lim} - \hat{h}_{n_1}) - (h_{n_2}^{\lim} - \hat{h}_{n_2}) \end{aligned}$$

As Bunya et al. [10] show, this algorithm lowers the depths at nodes n_2 and n_3 by equal amounts, and the algorithm is mass conserving.

3.3. Velocity-based limiting of the momentum

In a second step, the momentum distribution is modified by minimizing the in-cell variation of the resulting velocity distribution. This provides a stable computation near the wet/dry interface in situations when both, h and (hu) are small. For this we first compute preliminary limited velocity components $\hat{u}_i$ (and similarly $\hat{v}_i$) at each vertex i of the triangle

$$\hat{u}_i = \max\{\min\{u_i, u_c^{\max}\}, u_c^{\min}\}$$

where $u_i = (hu)_i/h_i$ and $u_c = (hu)_c/h_c$ are the x-velocities computed from the nodal and centroid values of momentum and (the unlimited) fluid depth. Note, that in case $h_i < \mathrm{TOL}_{\mathrm{wet}}$ the velocity is set to 0. The minimum and maximum values $u_c^{\min/\max}$ are determined as in subsection 3.2 for the total height by considering the centroid values of the neighboring cells which share a common edge (edge-based limiter) or a common vertex (vertex-based limiter) with the cell.

We can now obtain three different linear momentum distributions based on two of the three preliminary velocities, by keeping the cell mean value of the momentum and the distribution of the fluid depth. By this we obtain in each case a velocity for the third node

$$\hat{u}_1^{23} = \frac{3(hu)_c - h_2^{\lim} \cdot \hat{u}_2 - h_3^{\lim} \cdot \hat{u}_3}{h_1^{\lim}}, \quad \hat{u}_2^{13} = \frac{3(hu)_c - h_1^{\lim} \cdot \hat{u}_1 - h_3^{\lim} \cdot \hat{u}_3}{h_2^{\lim}},$$

$$\hat{u}_3^{12} = \frac{3(hu)_c - h_1^{\lim} \cdot \hat{u}_1 - h_2^{\lim} \cdot \hat{u}_3}{h_3^{\lim}}$$

where the lower index denotes the node for which the velocity is computed and the upper index defines on which nodal velocities this is based. The final limited momentum component is then chosen to produce the smallest velocity variation. Set

$$\begin{aligned}
\delta\hat{u}_1 &= \max\{\hat{u}_1^{23}, \hat{u}_2, \hat{u}_3\} - \min\{\hat{u}_1^{23}, \hat{u}_2, \hat{u}_3\}, \\
\delta\hat{u}_2 &= \max\{\hat{u}_1, \hat{u}_2^{13}, \hat{u}_3\} - \min\{\hat{u}_1, \hat{u}_2^{13}, \hat{u}_3\}, \\
\delta\hat{u}_3 &= \max\{\hat{u}_1, \hat{u}_2, \hat{u}_3^{12}\} - \min\{\hat{u}_1, \hat{u}_2, \hat{u}_3^{12}\}.
\end{aligned} \tag{5}$$

If $\delta\hat{u}_1 \leq \delta\hat{u}_i$ for $i \in \{2,3\}$, then

$$(hu)_1^{\lim} = h_1^{\lim} \cdot \hat{u}_1^{23}, \quad (hu)_2^{\lim} = h_2^{\lim} \cdot \hat{u}_2, \quad (hu)_3^{\lim} = h_3^{\lim} \cdot \hat{u}_3. \tag{6}$$

If $\delta\hat{u}_2 \leq \delta\hat{u}_i$ for $i \in \{1,3\}$, then

$$(hu)_1^{\lim} = h_1^{\lim} \cdot \hat{u}_1, \quad (hu)_2^{\lim} = h_2^{\lim} \cdot \hat{u}_2^{13}, \quad (hu)_3^{\lim} = h_3^{\lim} \cdot \hat{u}_3. \tag{7}$$

Otherwise

$$(hu)_1^{\lim} = h_1^{\lim} \cdot \hat{u}_1, \quad (hu)_2^{\lim} = h_2^{\lim} \cdot \hat{u}_2, \quad (hu)_3^{\lim} = h_3^{\lim} \cdot \hat{u}_3^{12}. \tag{8}$$

The final limited momentum is then given by $(hu)_h^{\lim} = \sum_i [(hu)_i^{\lim} \varphi_i(x, y)]$. The same procedure is applied to the y-momentum. In conclusion, the wetting and drying algorithm can be summarised as follows:

<div style="border:1px solid black; padding:10px">

Velocity-Based Wetting and Drying Treatment

1. Flux modification
   (a) Set g to 0 in volume integrals of semi-dry cells, which satisfy (4).
2. Limiting of fluid depth
   (a) Apply edge-based [3] OR vertex-based [24] limiter to total height $H = h + b$.
   (b) Apply positivity-preserving procedure [10] to limited $\hat{h}$ obtained from $\hat{H}$ in step 2a.
3. Limiting of momentum
   (a) Apply edge-based [3] OR vertex-based [24] limiter to velocities at triangle vertices.
   (b) Extrapolate in-cell velocity distribution from two out of three vertex values obtained in step 3a.
   (c) Determine discrete in-cell velocity variation from the three distributions obtained in step 3b.
   (d) Compute limited momentum from velocities with smallest variation and limited fluid depth (cf. (6)–(8)).

</div>

4. Numerical results

This section demonstrates the major functionalities of the limiting procedure described in section 3 allowing for accurate numerical shoreline representation, while preserving well-balancedness and conservation properties. Using a hierarchy of test cases, where we start with configurations where the exact solution to the shallow water equations is known, we show the scheme's mass conservation and well-balancedness, as well as the correct representation of the shoreline. Two test cases, which originate from Thacker [36] are good benchmarks with respect to the shoreline representation. Two further test cases are derived from laboratory experiments, which, together with the runup onto a linearly sloping beach, are standard test cases for the evaluation of operational tsunami models [35]

For the simulations, we use both versions of the limiter, i.e., vertex-based and edge-based limitation of total height and velocity, and compare the results. The presented limiter depends on a free parameter – a wet/dry tolerance $\mathrm{TOL_{wet}}$ that defines below which value of the fluid depth a point is considered dry. This is especially important for the computation of the velocity. We comment on this tolerance for every test case and, overall, conclude that the stability of the limiters is not sensitive to it, but it can influence the location of the wet/dry interface.

Throughout this section, we set the gravitational constant $g = 9.80616$. Here and below we omit dimensions of the physical quantities, which should be thought in standard SI units with m (meters), s (seconds) etc. For the spatial discretization we use regular grids, which are usually derived from one or more rectangles, each divided into two triangles. Such a grid is then repeatedly uniformly refined by bisection to obtain the desired resolution (see [5] for details on grid generation). The discrete initial conditions and the bottom topography are derived from the analytical ones by interpolation at the nodal points (vertices of triangles).

Explicit methods for the solution of partial differential equations are usually subject to a CFL time step restriction [13], which is of the form $\Delta t \leq \mathrm{cfl}\, h_\Delta / c_{\max}$. Here, h_Δ defines an appropriately defined grid parameter and $c_{\max}$ is the maximum propagation speed of information. For one-dimensional problems, Cockburn & Shu [12] proved that $\mathrm{cfl_{1D}} = 1/3$ for the RKDG2 method. However, this cannot be directly transferred to two-dimensional triangular grids. We follow the work of Kubatko et al. [23], who propose as grid parameter the radius of the smallest inner circle of the triangles surrounding a vertex. These nodal values are further aggregated to each triangle by taking the minimum of its three vertices. The 2D CFL number then approximately relates to its 1D counterpart by $\mathrm{cfl_{2D}} \approx 2^{-1/(p-1)}\mathrm{cfl_{1D}}$, where p is the order of discretization. This results in $\mathrm{cfl_{2D}} \approx 0.233$ for our RKDG2 method. For all the test cases, we choose a constant time step size $\Delta t^n = \Delta t$ which guarantees that the CFL condition is essentially satisfied.
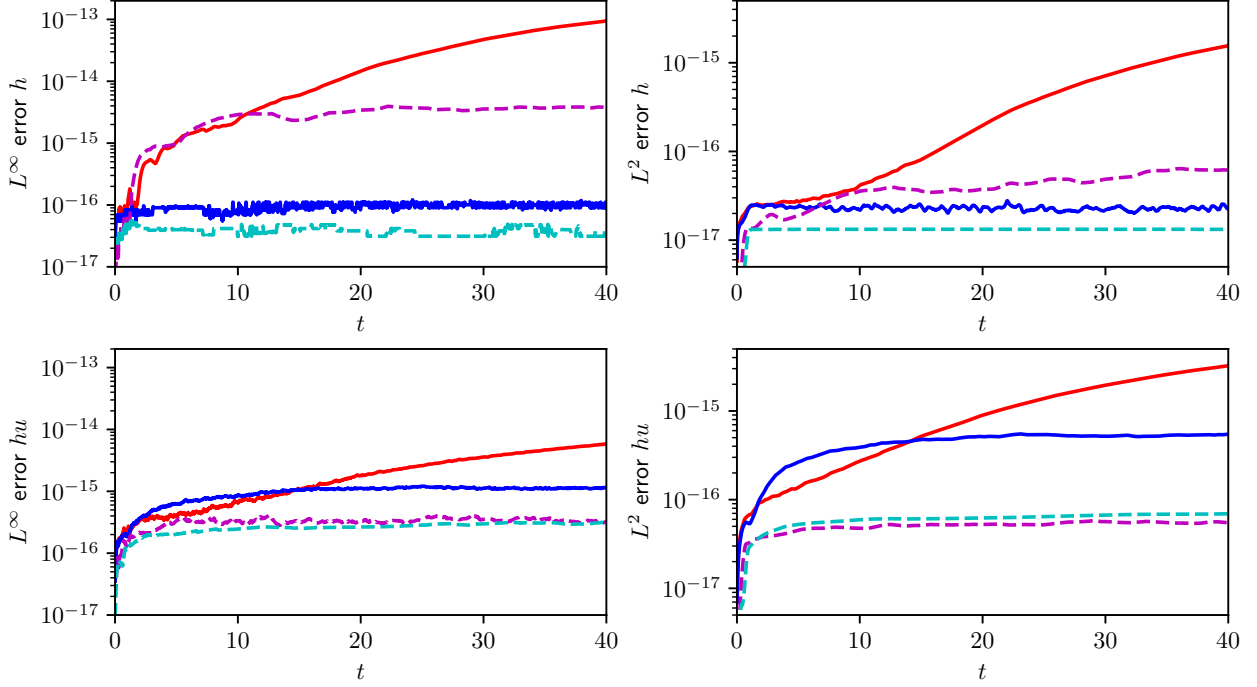
Figure 2: Lake at rest: errors in time for the for fluid depth (top) and momentum (bottom) in the L^∞ (left) and L^2 norms (right). Vertex based limiter with weak DG flux (red), edge based limiter with weak DG flux (blue), vertex based limiter with strong DG flux (magenta dashed), edge based limiter with strong DG flux (cyan dashed).

Besides fluid depth and momentum we often show the velocity, which is derived through post-processing by the quotient of the two other quantities.

4.1. Lake at rest

As a first test we conduct a simple "lake at rest" simulation to examine the well-balancedness of our scheme. In a quadratic and periodic domain $\Omega = [0,1]^2$ a bathymetry is defined by $b(\boldsymbol{x}) = \max\left\{0, 0.25 - 5((x-0.5)^2 + (y-0.5)^2)\right\}$, which features a local not fully submerged parabolic mountain centered around the mid point $(0.5, 0.5)^\top$. The initial fluid depth and velocity are given by

$$h(\boldsymbol{x}, 0) = \max\left\{0, 0.1 - b(\boldsymbol{x})\right\},$$
$$\boldsymbol{u}(\boldsymbol{x}, 0) = \boldsymbol{0}.$$

This is a steady state solution which should be reproduced by the numerical scheme. We run simulations using the strong and weak DG formulations with both limiters and a grid resolution of $\Delta x \approx 0.022$ (leg of right angled triangle) until $T_{\text{end}} = 40$. A time step of $\Delta t = 0.002$ is used, which corresponds to $20\,000$ time steps. The wet tolerance is chosen as $\text{TOL}_{\text{wet}} = 10^{-6}$.

The results are depicted in figure 2. We show the error in the L^2 as well as the maximum (L^∞) norm for the fluid depth (top row) and momentum (bottom row) for all four possible configurations. It can be observed that the vertex-based limiter in combination with the weak DG formulation yields an error which slowly grows in time and does not settle until the end of the simulation. With the edge-based limiter, however, we achieve a bounded error and well-balancedness. The simulations using the strong DG form show well-balancedness regardless of the chosen slope limiter. This behavior can be explained by the superior balancing properties of the strong DG formulation, which has been discussed in section 3 and has already been observed in Beisiegel [6].
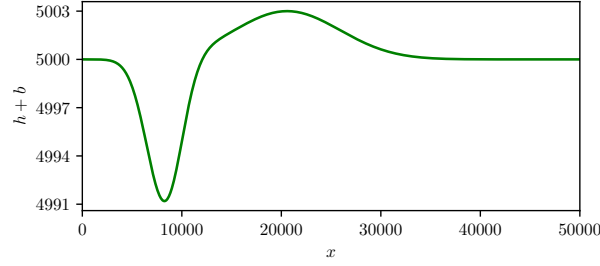
Figure 3: Tsunami runup onto a beach: initial surface elevation at $t = 0$.

4.2. Tsunami runup onto a linearly sloping beach

A standard benchmark problem to evaluate the wetting and drying behavior of a numerical scheme is the wave runup onto a plane beach. Here, we perform this quasi one-dimensional test case to compare the results to the ones already obtained with the one-dimensional version of the scheme [39]. It was originally presented in [37], and admits an exact solution following a technique developed in Carrier et al. [11]. In a rectangular domain $\Omega = [-400, 50\,000] \times [0, 400]$ with linearly sloping bottom topography $b(\boldsymbol{x}) = 5000 - \alpha x$, $\alpha = 0.1$, and initial velocity $\boldsymbol{u}(\boldsymbol{x}, 0) = \boldsymbol{0}$, an initial surface elevation is prescribed in non-dimensional variables by

$$\eta'(x') = a_1 \exp\left\{k_1(x' - x_1)^2\right\} - a_2 \exp\left\{k_2(x' - x_2)^2\right\}.$$

The parameters are given by $a_1 = 0.006$, $a_2 = 0.018$, $k_1 = 0.4444$, $k_2 = 4$, $x_1 = 4.1209$ and $x_2 = 1.6384$. Then, the initial surface profile is recovered by taking $x = Lx'$ and $\eta = \alpha L \eta'$ with the reference length $L = 5000$ (cf. figure 3). As the solution near the left boundary of Ω is always dry and and on the right boundary outgoing waves should not be reflected, we set wall and transparent boundary conditions for the left and right boundary, respectively. The boundaries in y-direction are set periodic, to not introduce any artifacts coming from the definition of the boundary conditions.

The simulations are run with a time step of $\Delta t = 0.04$, and a spatial resolution of $\Delta x = 50$ which corresponds to the length of the leg of a right angled triangle. The resulting Courant number is approximately 0.25, which is attained offshore where the fluid depth is largest. The wet/dry tolerance is set to $\mathrm{TOL_{wet}} = 10^{-2}$. We compare our numerical results with the analytical solution on the interval $x \in [-400, 800]$ at times $t = 160$, 175 and 220. The results are depicted in figure 4.

The left column shows the free surface elevation, where the simulations with both limiters (red and blue lines) match the exact solution (green line). However, it can be observed that due to its smaller stencil the edge-based limiter develops spurious discontinuities at cell edges. The middle and right column show the momentum and the reconstructed velocity at the respective times. As expected the momentum is reproduced well while the velocity shows some spurious over-, and undershoots in the near-dry area, but good results otherwise. In general, the vertex-based limiter yields qualitatively better results for this test case. The results are comparable to those in one space-dimension given in Vater et al. [39] and demonstrate the similarity of our two-dimensional extension of the limiter.

4.3. Long wave resonance in a paraboloid basin

The following two test cases particularly address the correct representation of a moving shoreline. They were originally defined in Thacker [36] and have an analytical solution. The first problem is a purely radial symmetric flow, where we also discuss the impact of the wet/dry tolerance on our method. We apply it in a scaled version as given in Lynett et al. [29]. In a quadratic domain $\Omega = [-4000, 4000]^2$ with a parabolic bottom topography given by $b(\boldsymbol{x}) = \tilde{b}(r) = H_0 \frac{r^2}{a^2}$ where $r = |\boldsymbol{x}| = \sqrt{x^2 + y^2}$, the initial fluid depth and velocity are prescribed by

$$h(\boldsymbol{x}, 0) = \max\left\{0, H_0\left(\frac{\sqrt{1 - A^2}}{1 - A} - \frac{|\boldsymbol{x}|^2(1 - A^2)}{a^2(1 - A)^2}\right)\right\}$$

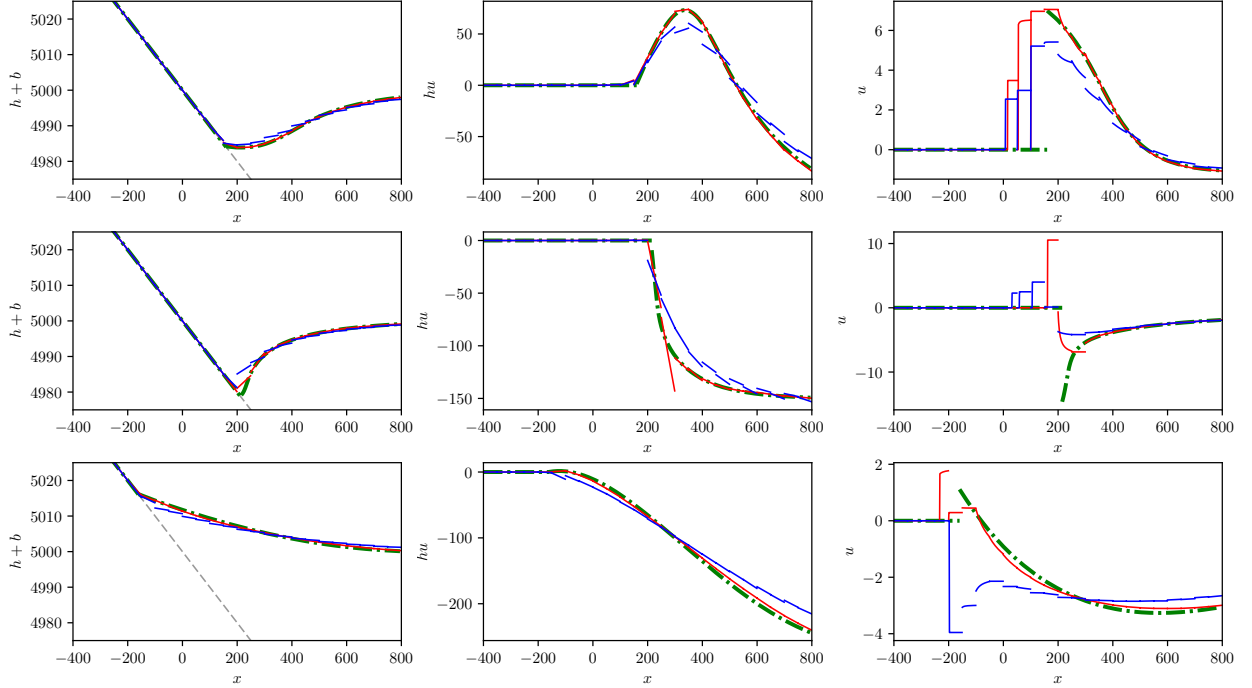$$\boldsymbol{u}(\boldsymbol{x}, 0) = \boldsymbol{0}$$

10

Figure 4: Tsunami runup onto a beach: surface elevation, x-momentum and x-velocity (derived by $u = (hu)/h$) along line $y = 200$ at times $t = 160$ (top), $t = 175$ (middle) and $t = 220$ (bottom). Exact solution (green dash-dotted), vertex based limiter (red), edge based limiter (blue).

where

$$A = \frac{a^4 - r_0^4}{a^4 + r_0^4},$$

$H_0 = 1$, $r_0 = 2000$, $a = 2500$. The exact radially symmetric solution is then given by

$$h(\boldsymbol{x}, t) = \max\left\{ 0, H_0 \left(\frac{\sqrt{1 - A^2}}{1 - A\cos(\omega t)} - \frac{|\boldsymbol{x}|^2 (1 - A^2)}{a^2 (1 - A\cos(\omega t))^2} \right) \right\}$$

$$(u, v)(\boldsymbol{x}, t) = \begin{cases} \dfrac{\omega A \sin(\omega t)}{2(1 - A\cos(\omega t))} \boldsymbol{x} & \text{if } h(\boldsymbol{x}, t) > 0 \\ \boldsymbol{0} & \text{otherwise,} \end{cases}$$

where ω is the frequency defined as $\omega = \sqrt{8gH_0}/a$.

The simulations are run for two periods (P) of the oscillation, which is until $T_{\text{end}} = 2P = 2 \cdot (2\pi/\omega)$, with a time step of $\Delta t = P/700 \approx 2.534$, and a spatial resolution of $\Delta x = 125/\sqrt{2} \approx 88.39$ (leg of right angled triangle). The initial Courant number is approximately 0.16. This is lower than the theoretically maximal Courant number because of possibly occurring spurious velocities in nearly dry regions affecting the Courant number at later times of the simulation. The maximum Courant number that is obtained for $\text{TOL}_{\text{wet}} = 10^{-14}$ is 0.22, whereas it is 0.16 for $\text{TOL}_{\text{wet}} = 10^{-2}$.

The results for fluid depth, x-momentum and x-velocity at times $t = 1.5P, 1.75P$ and $2P$ over a cross section $y = 0$ are shown in figure 5. Qualitatively, the vertex-based limiter (red line) shows slightly better results than the edge-based limiter (blue line). Note the small scale for the momentum at $t = 1.5P$ and $t = 2P$ in comparison with $t = 1.75P$. The momentum plots in comparison with the velocity plots also show the action of the limiter: While the momentum, especially for the edge-based limiter, is non-monotone in some regions, the velocity is mostly monotone. Only near the wet/dry interface some spurious velocities are visible when the overall velocity is close to zero.
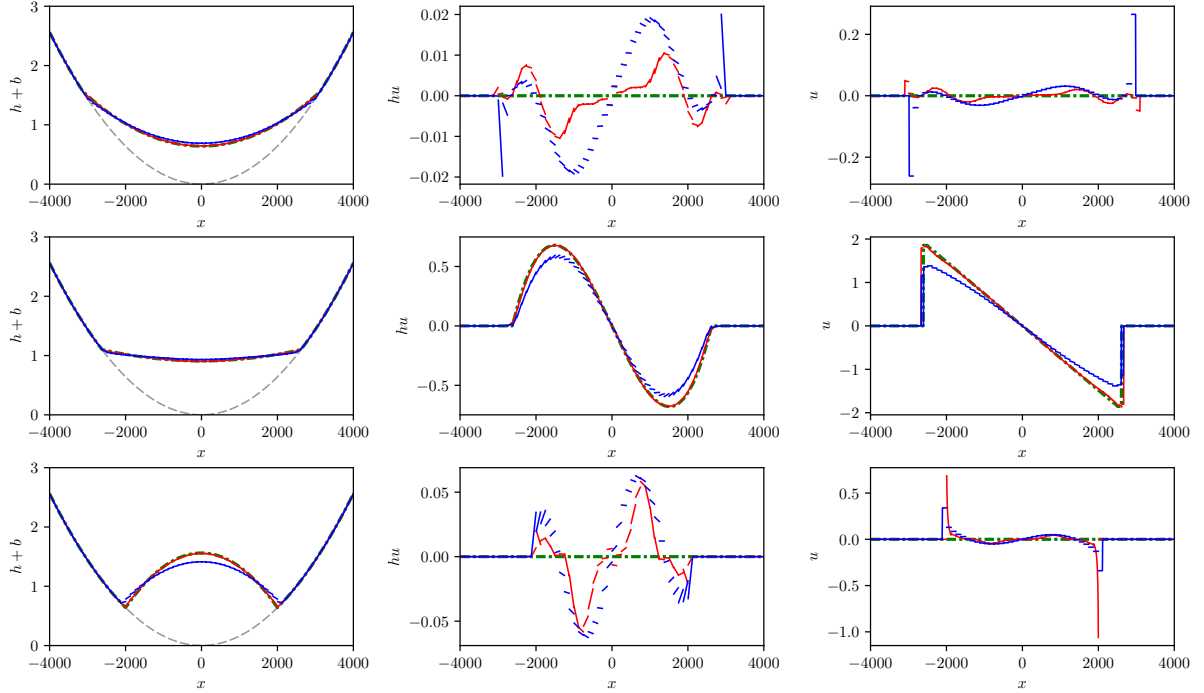
11

Figure 5: Long wave resonance in a paraboloid basin: cross section of fluid depth, x-momentum and x-velocity (left to right) at times $t = 1.5P$, $t = 1.75P$ and $t = 2P$ (top to bottom) at $y = 0$. Exact solution (green dash-dotted), vertex based limiter (red), edge based limiter (blue), $\text{TOL}_{\text{wet}} = 10^{-2}$.

A comparison of simulation results at final time $t = 2P$ with different wet/dry tolerances TOL_{wet} reveals that the results for the prognostic variables as well as the reconstructed velocities are largely insensitive to the chosen tolerance (see figure 6). However, as the wet/dry tolerance gets small a larger area is considered wet by the scheme. This is visible in the velocity plot, where spurious velocities start to appear in nearly dry regions. We further illustrate the effect of the parameter TOL_{wet} by comparison of fully two-dimensional fields obtained with the vertex-based limiter (figure 7). The top and bottom rows show the fluid depth and velocity with $\text{TOL}_{\text{wet}} = 10^{-2}$ and $\text{TOL}_{\text{wet}} = 10^{-8}$, respectively. The results for the fluid depth are largely identical with the exception that the area that the model recognizes as "wet" is much larger with a smaller tolerance. An interesting insight can be gained from the velocity plots. In the additional "wet" area obtained with a smaller tolerance, small values of momentum and fluid depth lead to spurious velocities. However, we note that in spite of the observed existence of spurious velocities, these are still bounded and their magnitudes are within the range of the exact solution to the problem.

4.4. Oscillatory flow in a parabolic bowl

The second test case which goes back to Thacker [36] is also defined in a parabolic bowl, but describes a circular flow with a linear surface elevation in the wet part of the domain. It is the 2D analogue of the 1D test case described in Vater et al. [39]. Here we follow the particular setup of Gallardo et al. [16] [see also 31]. In a square domain $\Omega = [-2, 2]^2$ with bottom topography $b = b(\boldsymbol{x}) = 0.1 \left(x^2 + y^2 \right)$, an analytical solution of the shallow water equations is given by

$$h(\boldsymbol{x}, t) = \max \left\{ 0, 0.1 \left(x \cos(\omega t) + y \sin(\omega t) + \tfrac{3}{4} \right) - b(\boldsymbol{x}) \right\}$$

$$(u, v)(\boldsymbol{x}, t) = \begin{cases} \frac{\omega}{2} \left(-\sin(\omega t), \cos(\omega t) \right) & \text{if } h(\boldsymbol{x}, t) > 0 \\ \boldsymbol{0} & \text{otherwise}, \end{cases}$$
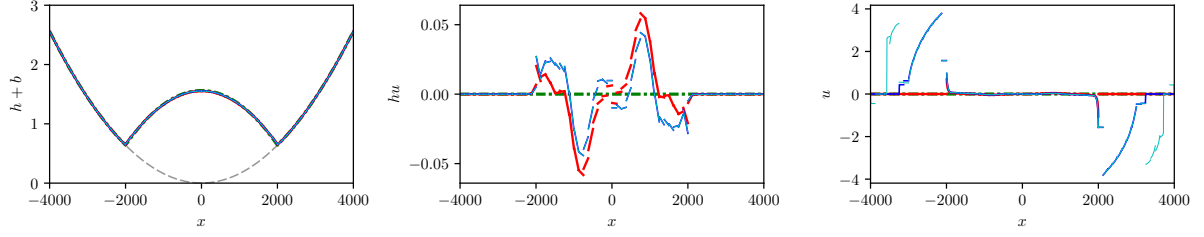
with $\omega = \sqrt{0.2g}$.

Figure 6: Long wave resonance in a paraboloid basin: cross section of fluid depth, x-momentum and x-velocity (left to right) at time $t = 2P$ at $y = 0$ for the vertex based limiter. Exact solution (green dash-dotted), solution with $\mathrm{TOL_{wet}} = 10^{-2}$ (red), with $\mathrm{TOL_{wet}} = 10^{-8}$ (blue) and with $\mathrm{TOL_{wet}} = 10^{-14}$ (cyan).
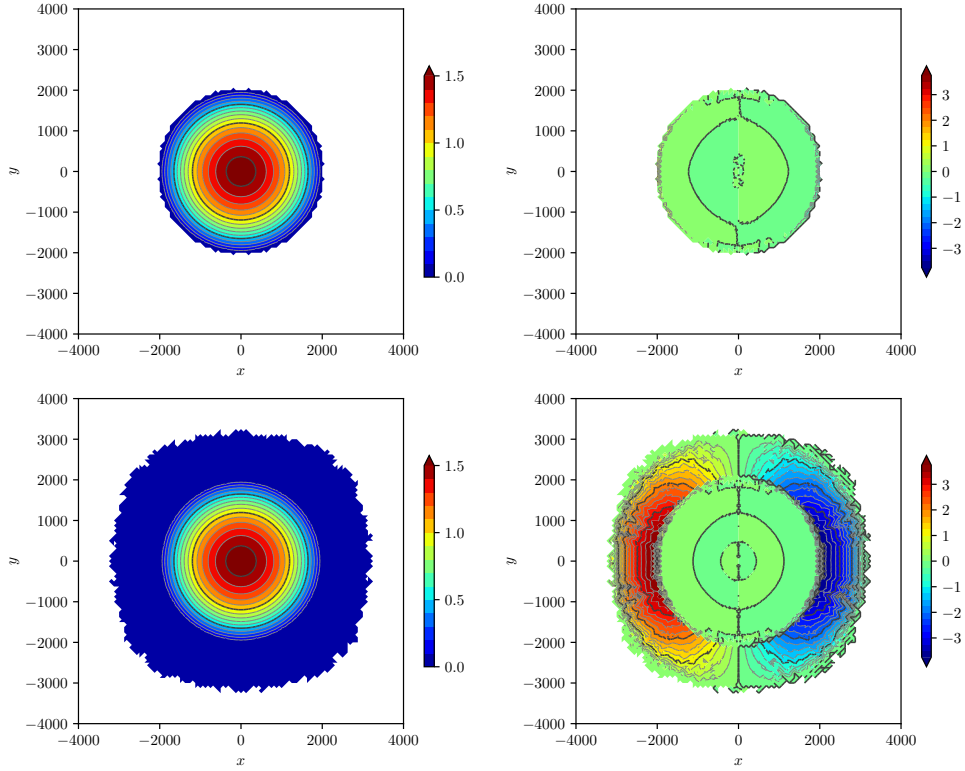
Figure 7: Long wave resonance in a paraboloid basin: 2d view of fluid depth (left) and x-velocity (right) at time $t = 2P$ for the vertex based limiter. Note, that only the area is colored, where the fluid depth is above the wet/dry tolerance. $\mathrm{TOL_{wet}} = 10^{-2}$ (top) and $\mathrm{TOL_{wet}} = 10^{-8}$ (bottom).

13
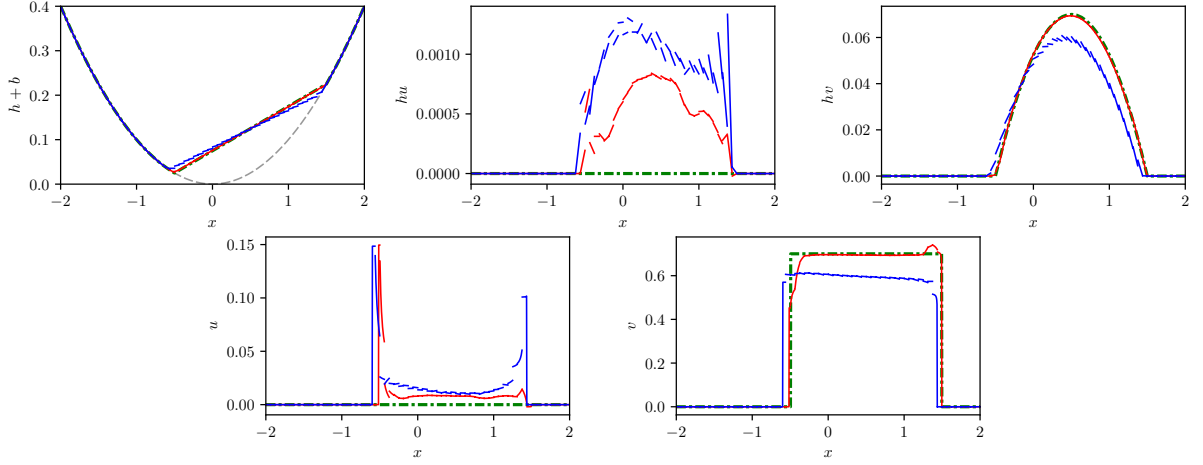
Figure 8: Oscillatory flow: cross section at $y = 0$ for time $t = 2P$. fluid depth, x-momentum, y-momentum, x-velocity, y-velocity (left to right, top to bottom). Exact solution (green dash-dotted), vertex based limiter (red), edge based limiter (blue), $\mathrm{TOL_{wet}} = 10^{-3}$.

Starting with $t = 0$ we ran simulations for two periods until $T_{\mathrm{end}} = 2P = 2 \cdot (2\pi/\omega)$ of the oscillation with a time step of $\Delta t = P/1000 \approx 0.004487$. The spatial resolution is set to $8\,192$ elements, which is a Cartesian grid with 64^2 squares divided into two triangles of an edge length of 0.0625 (leg of right angled triangle). The numerical results are shown in figures 8–12. Figure 8 shows cross sections over the line $y = 0$ at time $t = 2P$. The exact solution is plotted in green and the numerical approximation in red and blue for the vertex-based and edge-based limiter, respectively. The tolerance is chosen to $\mathrm{TOL_{wet}} = 10^{-3}$. We observe good agreement of the numerical results with the analytical solution for fluid depth, momentum and velocity. Note, however, that the edge-based limiter tends to produce artificial discontinuities in the solution and tends to slightly under-predict the momentum due to a higher inherent diffusion, which results from the edge-based stencil. This also results in a too small velocity and is visible in the 2D plots in figure 9. Moreover, the contour plot of the x-momentum (middle column) shows that the introduced discontinuities are clearly visible. The results with the vertex-based limiter (top) are smoother and show less diffusion.

Besides the accuracy on fixed grids, also the convergence of the wetting and drying scheme is of interest. While we cannot expect second order convergence due to the non-smooth transition between wet and dry regions in the flow variables, the convergence rate should be at least approximately linear. For the convergence calculation we have computed the solution up to $t = 2P$ on several grids with the number of cells ranging from $2\,048$ to $524\,288$ and fixed ratio $\Delta t/\Delta x$ and a wet/dry tolerance $\mathrm{TOL_{wet}} = 10^{-8}$. The experimental convergence rate is then calculated by the formula

$$\gamma_c^f := \frac{\log(\|e_c\|/\|e_f\|)}{\log(\Delta x_c/\Delta x_f)}.$$

In this definition, e_c and e_f are the computed error functions of the solution on a coarse and a fine grid (denoted by the number of cells) and Δx_c and Δx_f are the corresponding grid resolutions. In figure 10 and table 1 we show the results of this convergence analysis. The DG method converges with both limiters, however, the convergence rate that is achieved in the L^2 norm is higher with the vertex-based limiter (≈ 1.6) than it is with the edge-based limiter (≈ 1).

The test case of an oscillatory flow in a parabolic bowl is also suitable to evaluate the conservation of mass and of total energy $E = \int_\Omega h(\boldsymbol{u} \cdot \boldsymbol{u})/2 + gh(h/2 + b)\,\mathrm{d}x$ for the numerical method, since there is no flow across the boundary of the domain. While mass conservation should hold up to machine accuracy, total energy can only hold within the approximation error. We can see that mass conservation is not affected by the slope limiters (left plot of figure 11), while only the vertex-based limiter (right plot of same figure) nearly conserves energy. This indicates that the edge-based limiter exposes some numerical dissipation.
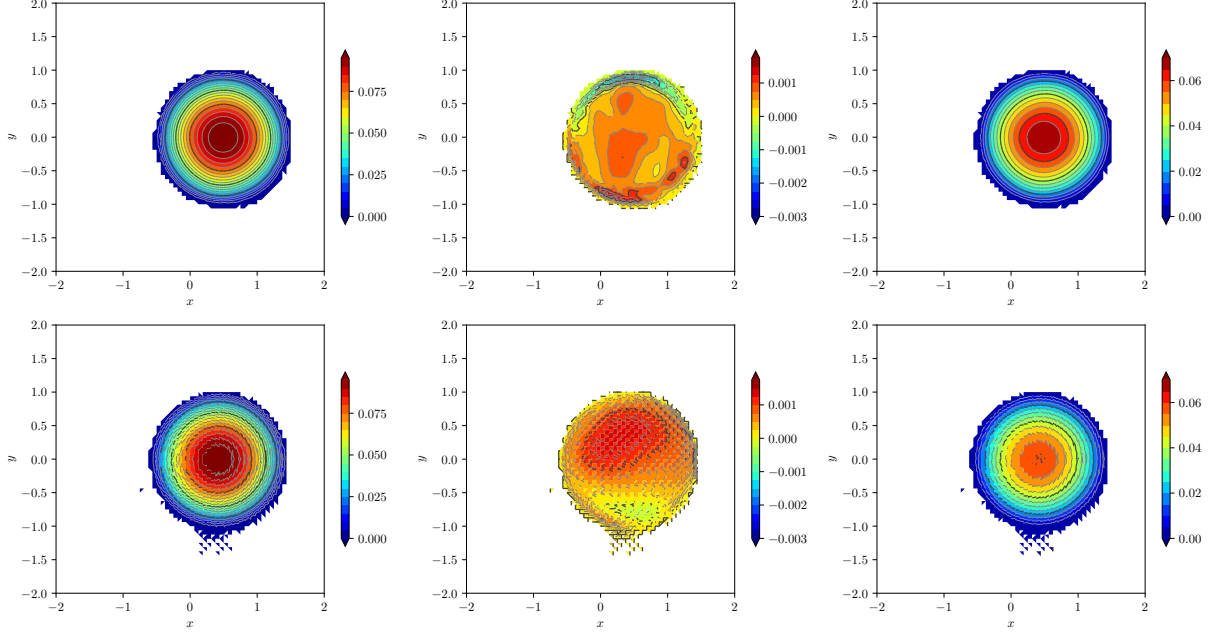
14

Figure 9: Oscillatory flow: 2d view for vertex based limiter (top) and edge based limiter (bottom). Fluid depth, x-momentum, y-momentum (left to right) at time $t = 2P$, $\text{TOL}_{\text{wet}} = 10^{-3}$.
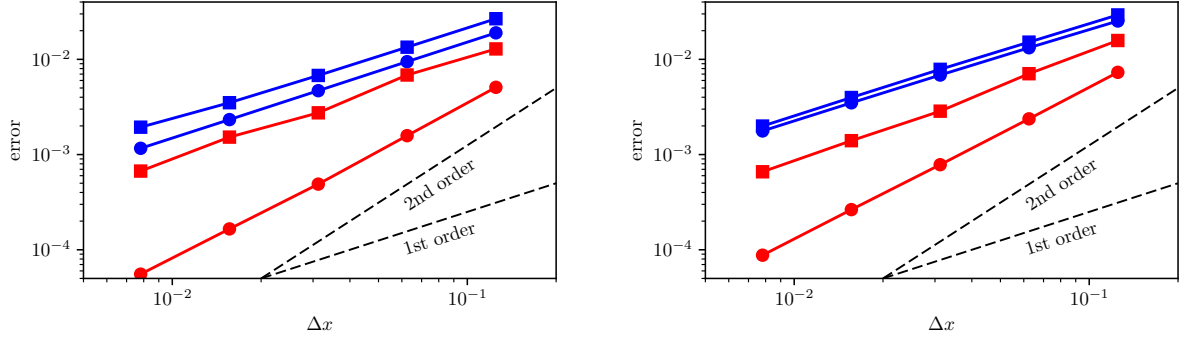

Figure 10: Oscillatory flow: errors in fluid depth (left) and momentum (right) measured in the L^2 norm (circles) and the L^∞ norm (squares). Vertex based limiter (red), edge based limiter (blue).

	$L^2(h)$	$L^2(m)$	$L^\infty(h)$	$L^\infty(m)$		$L^2(h)$	$L^2(m)$	$L^\infty(h)$	$L^\infty(m)$
γ_{2048}^{8192}	1.6873	1.6230	0.9104	1.1587	γ_{2048}^{8192}	1.0048	0.9332	0.9926	0.9494
γ_{8192}^{32768}	1.6903	1.5996	1.3190	1.3072	γ_{8192}^{32768}	1.0125	0.9527	0.9860	0.9491
γ_{32768}^{131072}	1.5626	1.5671	0.8477	1.0294	γ_{32768}^{131072}	1.0090	0.9694	0.9513	0.9834
γ_{131072}^{524288}	1.5779	1.5901	1.1845	1.0847	γ_{131072}^{524288}	1.0012	0.9802	0.8538	0.9957
γ_{fitted}	1.6289	1.5926	1.0690	1.1496	γ_{fitted}	1.0077	0.9593	0.9505	0.9688

Table 1: Oscillatory flow: convergence rates between different grid levels for fluid depth (h) and momentum (m) in the L^2 and L^∞ norms. Vertex based limiter (left) and edge based limiter (right). Also displayed is the mean convergence rate γ_{fitted}, which is obtained by a least squared fit.
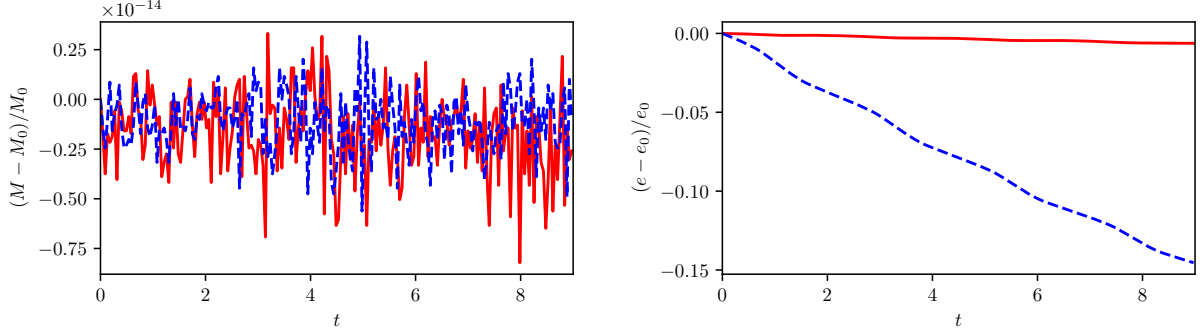
Figure 11: Oscillatory flow: time series of relative mass (left) and energy (right) changes for the vertex based limiter (red) and edge based limiter (blue).
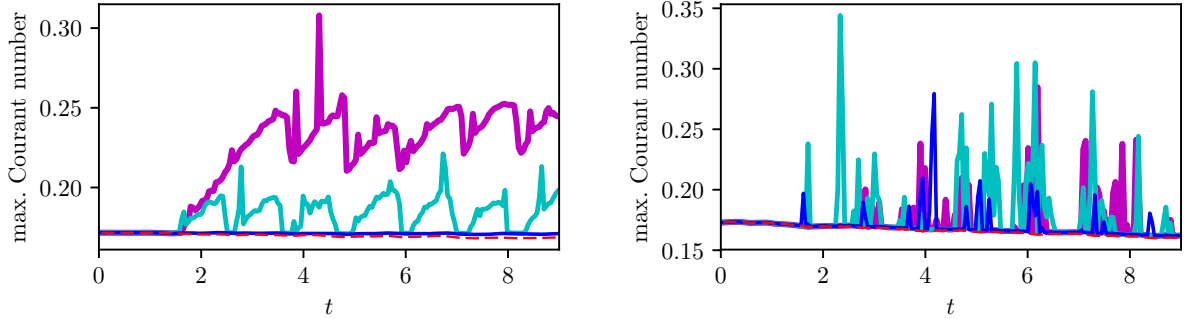

Figure 12: Oscillatory flow: time series of maximum Courant number with wet/dry tolerance $\text{TOL}_{\text{wet}} = 10^{-2}$ (red dashed), $\text{TOL}_{\text{wet}} = 10^{-4}$ (blue), $\text{TOL}_{\text{wet}} = 10^{-8}$ (cyan), $\text{TOL}_{\text{wet}} = 10^{-14}$ (magenta). Vertex based limiter (left), edge based limiter (right).

Finally, we record the Courant number for this test case over time for different wet/dry tolerances. In figure 12 we plot the Courant number for the vertex-based (left) and edge-based limiter (right) with wet/dry tolerances $\text{TOL}_{\text{wet}} \in \{10^{-2}, 10^{-4}, 10^{-8}, 10^{-14}\}$. It can be observed that for all tolerances the Courant number stays bounded and mostly below the theoretical limit. However, when the wet/dry tolerance becomes smaller, spurious velocities start to arise and affect the Courant number. If the tolerance is set large enough ($\approx 10^{-4}$), we obtain a nearly constant Courant number over time, which is similar to the Courant number of the exact problem.

4.5. Runup onto a complex three-dimensional beach

The 1993 Okushiri tsunami caused many unexpected phenomena, such as an extreme runup height of 32m which was observed near the village of Monai on Okushiri island. The event was reconstructed in an 1/400 scale laboratory experiment, using a large-scale tank (205m long, 6m deep, 3.4m wide) at Central Research Institute for Electric Power Industry (CRIEPI) in Abiko, Japan [30]. For the test case the coastal topography in a domain of 5.448m × 3.402m and the incident wave from offshore is provided. Beside the temporal and spatial variations of the shoreline location, the temporal evolution of the surface elevation at three specified gauge stations are of interest (figure 13).

At the offshore boundary we set the incident wave as right-going simple wave. This means, given the fluid depth h of the incident wave the x-velocity is computed by

$$u = 2(\sqrt{gh} - \sqrt{gh_0}), \tag{9}$$

where $h_0 = 0.13535$m denotes the water depth at rest. At the other three boundaries there were reported to be walls. So we set reflective wall boundary conditions at these locations.
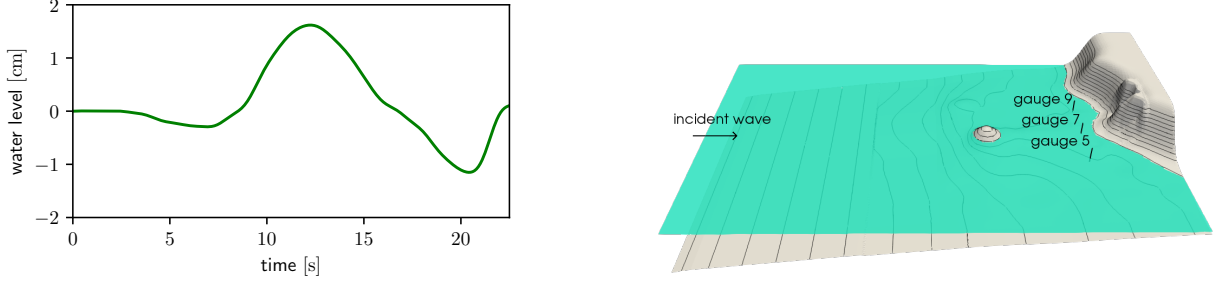
16

Figure 13: Okushiri: time series of incident wave which is used as boundary condition (left) and experimental setup (right).
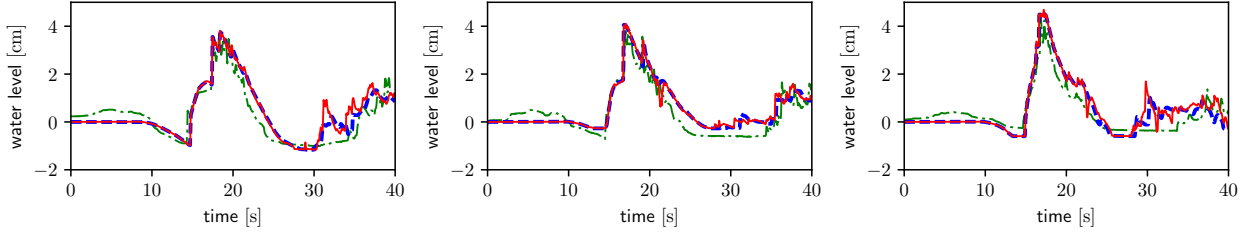

Figure 14: Okushiri: time series of gauge data: gauge 5 (left), gauge 7 (middle), gauge 9 (right), experimental data (green dash-dotted), vertex based limiter (red), edge based limiter (blue dashed). $\text{TOL}_\text{wet} = 10^{-4}$.

We ran simulations with a time step of $\Delta t = 0.001$ until $40\,000$ steps ($T_\text{end} = 40$) on a grid with $393\,216$ elements (384×256 rectangles divided into four triangles). The wet/dry tolerance was set to $\text{TOL}_\text{wet} = 10^{-4}$. The results are depicted in figures 14 and 15. Figure 14 shows the comparison of the numerical results with experimental data at gauges 5, 7, and 9. Overall, we observe good agreement with both limiters (red and blue lines). Detailed contour plots of the coastal area together with the experimentally derived shoreline are shown in figure 15 for times $t = 15.0, 15.5, 16.0, 16.5, 17.0$. This shoreline was taken from [26] and adjusted to the figures, which means it can only be used as a rough estimate. However, the flood line is represented well and we also demonstrate a good match of the maximum run-up (red dot) at $t = 16.5$.

4.6. Flow around a conical island

This test is part of a series of experiments carried out at the US Army Engineer Waterways Experiment Station in a 25×28.2m basin with a conical island situated at its center and is reported in Briggs et al. [9] [see also 28]. The experiment was motivated by the 1992 Flores Island tsunami run-up on Babi Island. The conical island has its center at $\boldsymbol{x}_I = (12.96, 13.8)^\top$ and is defined by

$$b(\boldsymbol{x}) = \tilde{b}(r) = \begin{cases} 0.625 & r \leq 1.1 \\ (3.6 - r)/4 & 1.1 \leq r \leq 3.6 \\ 0 & \text{otherwise}, \end{cases}$$

with $r = \|\boldsymbol{x} - \boldsymbol{x}_I\|_2$ being the distance from the center. The initial fluid depth and velocity are given by $h(\boldsymbol{x}, 0) = \max\{0, h_0 - b(\boldsymbol{x})\}$ and $\boldsymbol{u}(\boldsymbol{x}, 0) = \boldsymbol{0}$, where $h_0 = 0.32$. Three different solitary waves (denoted by case A, B and C) were generated by a wavemaker in the experiments at the left side of the domain, from which we only consider case A and C. Besides the trajectories of the wave paddle, time series of the surface elevation at 27 different gauge stations, 8 of which are freely available, were measured. The first four gauge stations were situated in a $L/2$ distance in x-direction from the toe of the beach, where $L/2$ is the distance at which the solitary wave height drops to 5% of its maximum height, and L defines the wave length. In the numerical simulations we describe the wave by an incoming analytical solitary wave through the boundary condition on the left side of the domain. In order to make the analytical wave compatible to measurements,
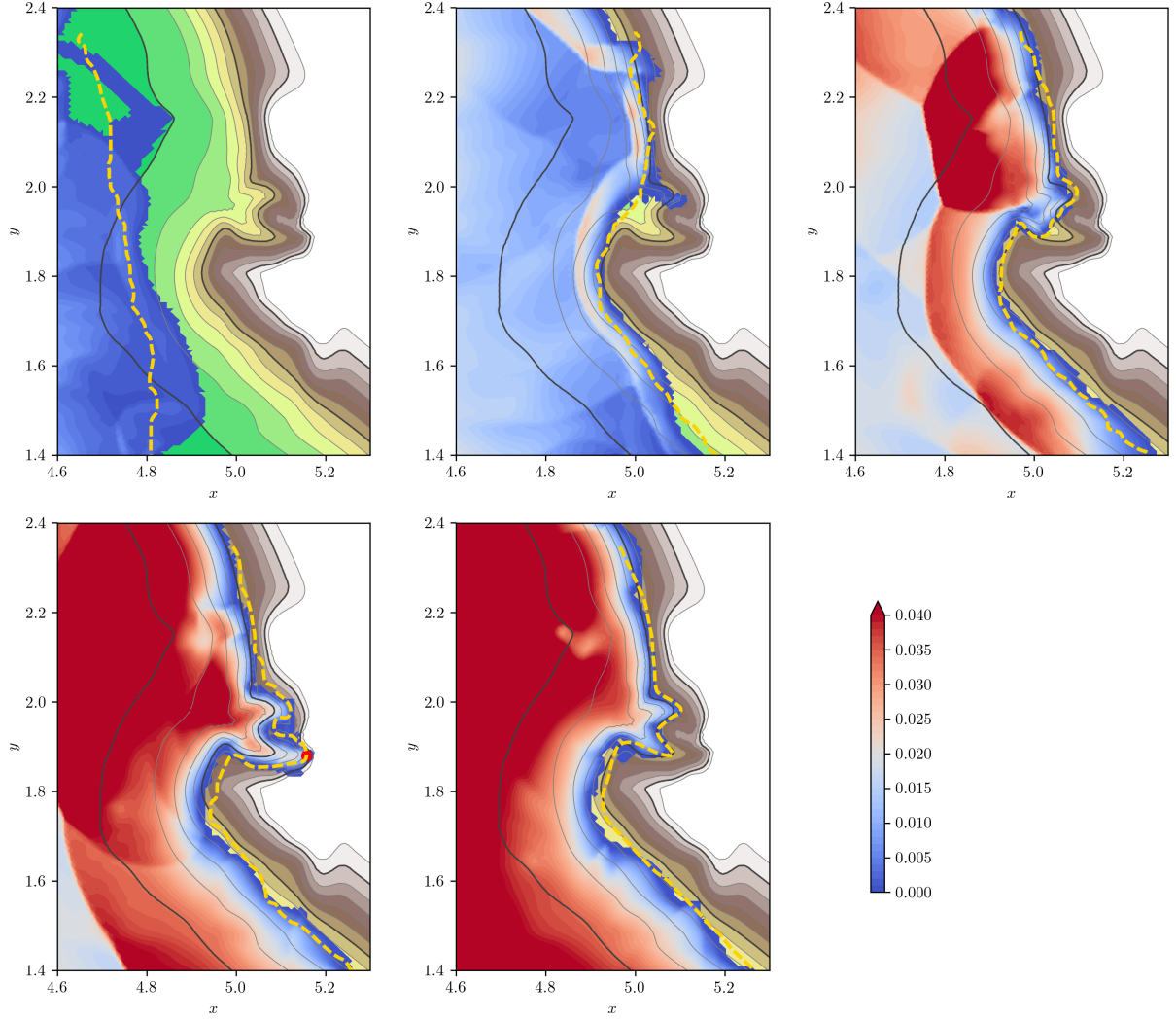
17

Figure 15: Okushiri: detailed contour plot of the coastal area at $t = 15.0, 15.5, 16.0, 16.5, 17.0$, (top left to right bottom), vertex based limiter with $\text{TOL}_{\text{wet}} = 10^{-4}$. Contour colors are given for the fluid depth. Contour lines represent the topography at $0.0, 0.01, \ldots, 0.11$. Also given is the approximate location of the shoreline from the experiment (yellow dashed) and the maximum observed runup (red circle), which happened around $t = 16.5$.

it needs to be adjusted with the parameters given below. The wave is defined by

$$h_b(t) = h_0 + a \left(\frac{1}{\cosh(K(cT - ct - x_0))} \right)^2$$

where $K = \sqrt{\frac{3a}{4h_0^3}}$ and $c = \sqrt{gh_0} \left(1 + \frac{a}{2h_0} \right)$. To obtain the other parameters, we fitted the solitary wave to the experimental data at the first four gauge stations. This resulted in an amplitude and time shift of $a = 0.014$ and $T = 8.85$ for case A. The parameter $x_0 = 5.76$ is the x-coordinate of the first four gauges. For case C the parameters are $a = 0.057$, $T = 7.77$ and $x_0 = 7.56$. Compared to the experiments this also includes a time shift of 20. As in the Okushiri test case the velocity at the boundary is defined to obtain a right running simple wave (cf. (9)).

For the numerical simulations the domain is slightly adjusted to have dimensions 25.92×27.60, and the conical island is exactly centered. The domain is discretized into 1024×1024 uniform squares which
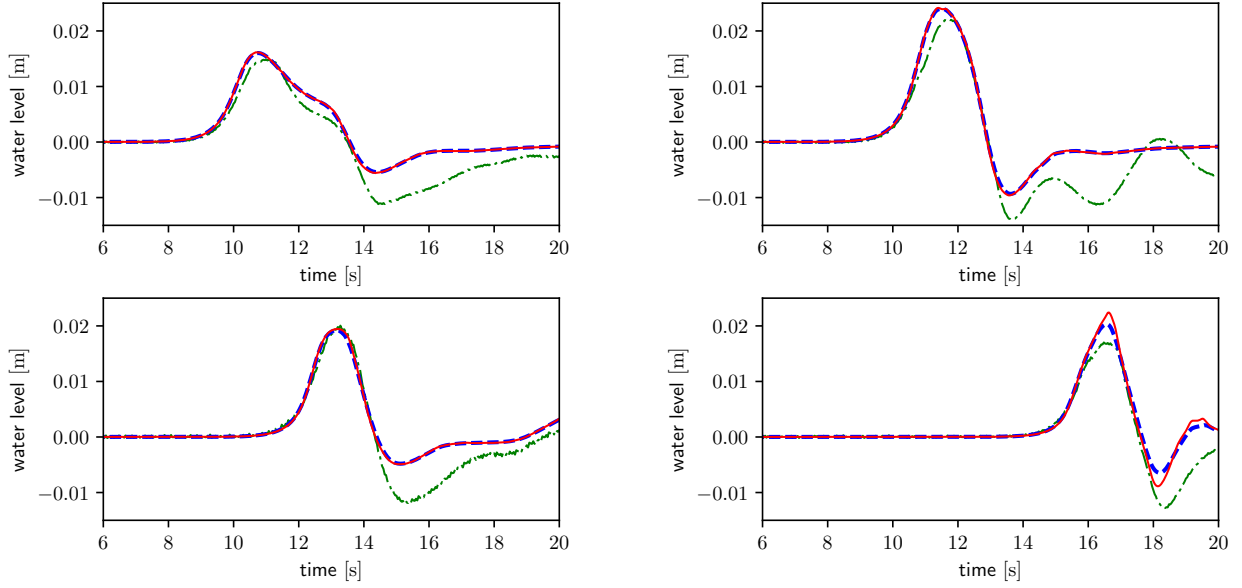
18

Figure 16: Conical Island: time series for case A of surface elevation at wave gauges 6, 9, 16, 22 (top left to bottom right) experimental data (green dash-dotted, shifted by $\Delta t = -20$), vertex based limiter (red), edge based limiter (blue dashed).

are divided into two triangles ($2\,097\,152$ elements). The time step is $\Delta t = 0.0025$, and the computations are run until $T_{\mathrm{end}} = 20$. Simulations were computed using both limiters with a wet/dry tolerance of $\mathrm{TOL}_{\mathrm{wet}} = 10^{-3}$. On the left side of the domain, we impose an inflow boundary condition to prescribe the solitary wave. Furthermore, a transparent boundary condition is set on the right side and wall boundary conditions are set at the top and the bottom of the domain.

In figures 16 and 17 we compare the resulting time series of the surface elevation at gauge stations 6, 9, 16, and 22 for case A and C, respectively, with the experimental data. Note that some time series from the experiments were slightly shifted to have an initial zero water level. While gauge 6 and 9 are right in front of the island, gauge 16 is on the side and gauge 22 in the rear of it. It can be seen that for the smaller wave amplitude (case A) the experimental data can be reproduced well. On the other hand, for the higher amplitude in case C non-linear effects become dominant and are not balanced because of the lack of wave dispersion. The result is a steepening of the wave at the front and a flattening at the rear.

In figure 18 some snapshots of the simulations using the vertex based limiter are displayed. They demonstrate that the initial wave correctly splits into two wave fronts after hitting the island. These wave fronts collide behind the island at a later time in the simulation. Finally, we compare the computed maximum run-up on the island with the measurements from the experiments in figure 19. For both configurations of wave amplitude the simulations resemble measurements well and only slightly overestimate the runup. These deviations are larger in case C at the front of the island, where the wave first arrives. This behavior is probably due to the lack of wave dispersion and the distorted shape of the wave, when the inundation happens. We attribute the better fit of the runup resulting from the simulation with the edge based limiter to the additional diffusion introduced by this limiter, and not to a better physical modelling of the runup.

5. Conclusions

In this work a new wetting and drying treatment for RKDG2 methods applied to the shallow water equations is presented. It is based on a simple flux modification and a limiter, which renders the scheme mass-conservative, well-balanced and stable for a wide range of flow regimes. It is a natural extension of a previously developed 1D scheme [39] to the case of two-dimensional triangular grids. The key idea is a velocity-based limiting of the momentum, which preconditions the velocity computation near the shoreline,
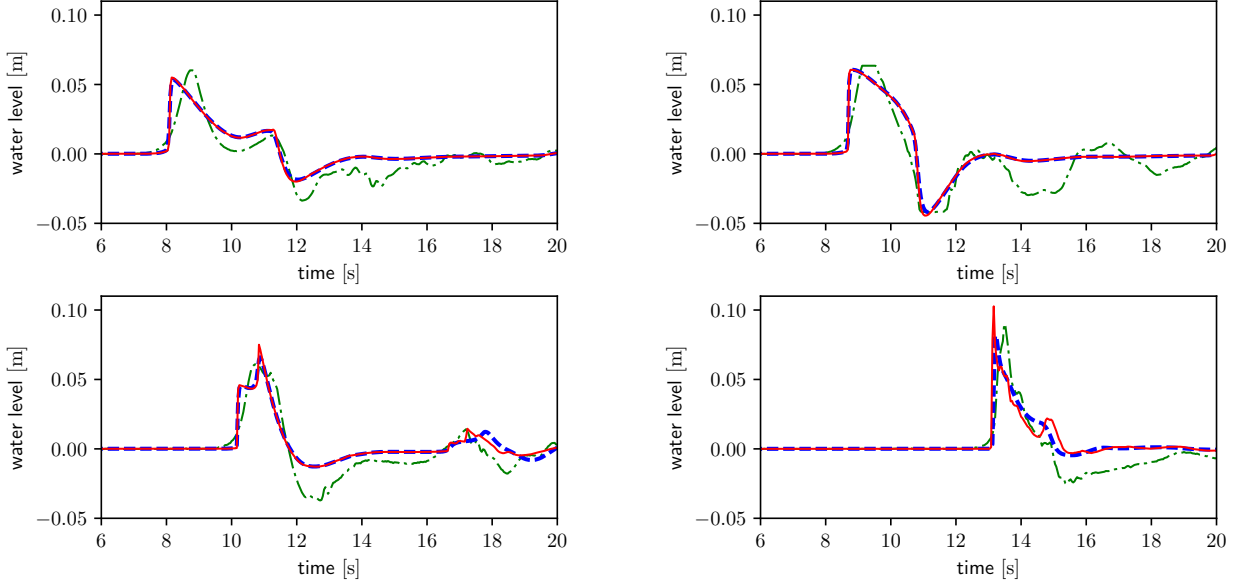
19

Figure 17: Conical Island: time series for case C of surface elevation at wave gauges 6, 9, 16, 22 (top left to bottom right) experimental data (green dash-dotted, shifted by $\Delta t = -20$), vertex based limiter (red), edge based limiter (blue dashed).

where both, fluid depth and momentum become small. This in turn guarantees a uniform time step with respect to the CFL stability constraint for explicit methods.

While mass-conservation is inherent for DG methods, well-balancing is achieved by limiting in total height $H = h + b$ and using a non-destructive limiter for the lake at rest steady state. The incorporation of the velocity into the limiting procedure prevents large spurious velocities near the wet/dry interface. Only one single parameter TOL_wet enters the scheme, which determines below which fluid depth a point is considered dry. We show that the stability of the method is unaffected by this parameter. It rather determines the effective area which is considered wet by the discretization. A carefully chosen wet/dry tolerance thus leads to an accurate shoreline computation.

Two versions of the limiter were presented that differed in the selection of cells to be included into the limiting procedure. The "edge-based" version is based on the original Barth/Jespersen [3] limiter. Due to its small stencil, it even modifies states with constant gradients and therefore introduces additional diffusion into the method. The "vertex-based" version is an extension of the Barth/Jespersen limiter [24] especially designed for triangular grids and is non-destructive to linear states. It results in more accurate computations in most situations, but has a larger stencil.

The presented test cases range from simple configurations where the analytical solution is known to the reproduction of laboratory experiments. They illustrate the method's applicability to a variety of flow regimes and verify its numerical properties: well-balancing in the case of a lake at rest, accurate representation of the shoreline, even in case of fast transitions, and convergence to the exact solution. Comparison with laboratory experiments show good agreement. Some of the test cases are also benchmark problems for the evaluation of operational tsunami models [35]. With the successful simulation of these problems, we could show that the presented model also satisfies the requirements for its application to such geophysical problems.

Future research will concentrate on the extension of the current scheme to adaptive grids and its application to tsunami and storm surge simulations. In this respect, additional source terms, such as the parametrization of sub-grid roughness by bottom friction and wind drag must be incorporated into the model. Furthermore, possibilities to extend the proposed concept to higher than second-order RKDG methods are investigated.
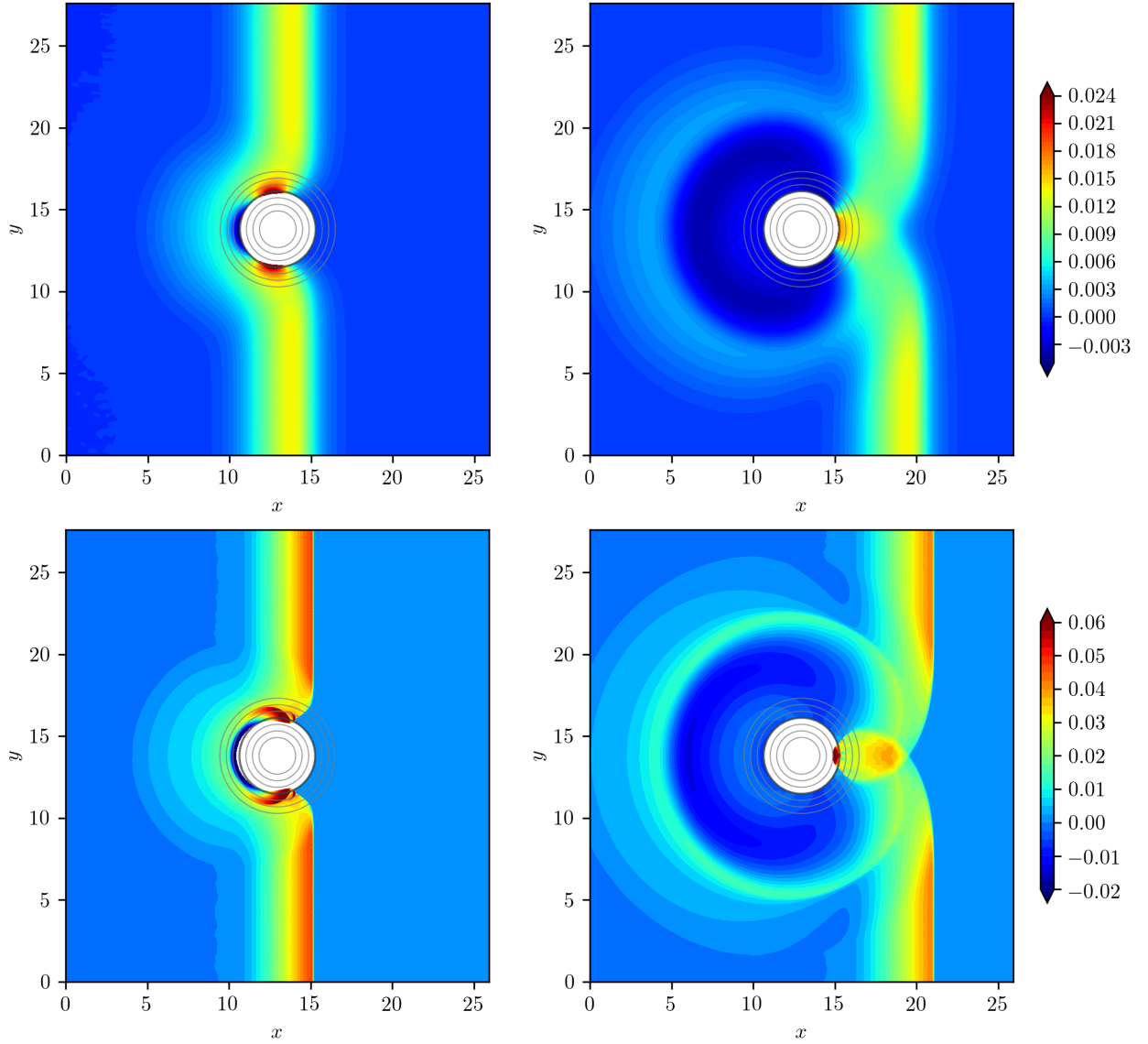
Figure 18: Conical Island: contour plot of surface elevation for case A (top) at $t = 13$ (left) and $t = 16$ (right) using the vertex based limiter. Same for case C (bottom) at $t = 11$ (left) and $t = 14$ (right).

[1] Arpaia, L., & Ricchiuto, M. (2016). *r-adaptation for Shallow Water flows: conservation, well balancedness, efficiency.* Research Report RR-8956 Inria Bordeaux Sud-Ouest. URL: `https://hal.inria.fr/hal-01372496`.

[2] Audusse, E., Bouchut, F., Bristeau, M.-O., Klein, R., & Perthame, B. (2004). A fast and stable well-balanced scheme with hydrostatic reconstruction for shallow water flows. *SIAM Journal on Scientific Computing*, *25*, 2050–2065. doi:`10.1137/S1064827503431090`.

[3] Barth, T. J., & Jespersen, D. C. (1989). The design and application of upwind schemes on unstructured meshes. *AIAA Paper*, *89-0366*.

[4] Bates, P. D., & Hervouet, J.-M. (1999). A new method for moving-boundary hydrodynamic problems in shallow water.
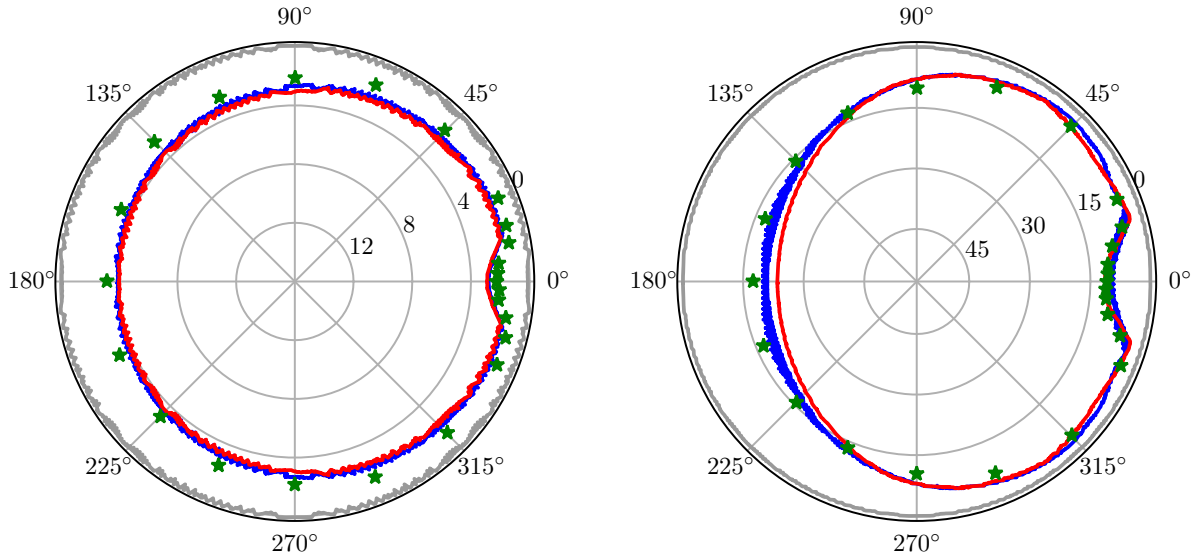
Figure 19: Conical Island: maximum vertical runup in cm for case A (left) and case C (right). Results using vertex based limiter (red), edge based limiter (blue), and from the experiments (green asterisks). Initial numerical shoreline is displayed in gray. Note the enlarged runup scale for case A.

      Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences, *455*, 3107–3128. doi:`10.1098/rspa.1999.0442`.

[5] Behrens, J., Rakowsky, N., Hiller, W., Handorf, D., Läuter, M., Päpke, J., & Dethloff, K. (2005). amatos: Parallel adaptive mesh generator for atmospheric and oceanic simulation. *Ocean Modelling*, *10*, 171–183. doi:`10.1016/j.ocemod.2004.06.003`.

[6] Beisiegel, N. (2014). *High-order Adaptive Discontinuous Galerkin Inundation Modeling*. Ph.D. thesis Universität Hamburg. URL: `http://ediss.sub.uni-hamburg.de/volltexte/2014/7036`.

[7] Bernard, P. E., Chevaugeon, N., Legat, V., Deleersnijder, E., & Remacle, J.-F. (2007). High-order h-adaptive discontinuous Galerkin methods for ocean modelling. *Ocean Dynamics*, *57*, 109–121. doi:`10.1007/s10236-006-0093-y`.

[8] Bokhove, O. (2005). Flooding and drying in discontinuous Galerkin finite-element discretizations of shallow-water equations. part 1: One dimension. *Journal of Scientific Computing*, *22–23*, 47–82. doi:`10.1007/s10915-004-4136-6`.

[9] Briggs, M., Synolakis, C., Harkins, G., & Green, D. (1995). Laboratory experiments of tsunami run-up on circular island. *Pure Appl. Geophys.*, *144*, 569–-593.

[10] Bunya, S., Kubatko, E. J., Westerink, J. J., & Dawson, C. (2009). A wetting and drying treatment for the Runge-Kutta discontinuous Galerkin solution to the shallow water equations. *Comput. Methods Appl. Mech. Engrg.*, *198*, 1548–1562. doi:`10.1016/j.cma.2009.01.008`.

[11] Carrier, G. F., Wu, T. T., & Yeh, H. (2003). Tsunami run-up and draw-down on a plane beach. *Journal of Fluid Mechanics*, *475*, 79–99. doi:`10.1017/S0022112002002653`.

[12] Cockburn, B., & Shu, C.-W. (1991). The Runge-Kutta local projection p^1-discontinuous-Galerkin finite element method for scalar conservation laws. *RAIRO Modélisation Mathématique et Analyse Numérique*, *25*, 337–361.

[13] Courant, R., Friedrichs, K. O., & Lewy, H. (1928). Über die partiellen Differenzengleichungen der mathematischen Physik. *Mathematische Annalen*, *100*, 32–74. doi:`10.1007/BF01448839`.

[14] Dawson, C., Trahan, C. J., Kubatko, E. J., & Westerink, J. J. (2013). A parallel local timestepping Runge–Kutta discontinuous Galerkin method with applications to coastal ocean modeling. *Computer Methods in Applied Mechanics and Engineering*, *259*, 154–165. doi:`10.1016/j.cma.2013.03.015`.

[15] Ern, A., Piperno, S., & Djadel, K. (2008). 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*, 1–25. doi:`10.1002/fld.1674`.

[16] Gallardo, J. M., Parés, C., & Castro, M. (2007). On a well-balanced high-order finite volume scheme for shallow water equations with topography and dry areas. *Journal of Computational Physics*, *227*, 574–601. doi:`10.1016/j.jcp.2007.08.007`.

[17] Giraldo, F. X., & Warburton, T. (2008). A high-order triangular discontinuous Galerkin oceanic shallow water model. *International Journal for Numerical Methods in Fluids*, *56*, 899–925. doi:`10.1002/fld.1562`.

[18] Gottlieb, S., Shu, C.-W., & Tadmor, E. (2001). Strong stability-preserving high-order time discretization methods. *SIAM*

Review, *43*, 89–112. doi:10.1137/S003614450036757X.

[19] Gourgue, O., Comblen, R., Lambrechts, J., Kärnä, T., Legat, V., & Deleersnijder, E. (2009). A flux-limiting wetting–drying method for finite-element shallow-water models, with application to the Scheldt Estuary. *Advances in Water Resources*, *32*, 1726–1739. doi:10.1016/j.advwatres.2009.09.005.

[20] Hesthaven, J. S., & Warburton, T. (2008). *Nodal discontinuous Galerkin methods: algorithms, analysis, and applications*. Springer.

[21] Kärnä, T., de Brye, B., Gourgue, O., Lambrechts, J., Comblen, R., Legat, V., & Deleersnijder, E. (2011). A fully implicit wetting-drying method for DG-FEM shallow water models, with an application to the Scheldt Estuary. *Computer Methods in Applied Mechanics and Engineering*, *200*, 509–524. doi:10.1016/j.cma.2010.07.001.

[22] Kesserwani, G., & Liang, Q. (2010). A discontinuous Galerkin algorithm for the two-dimensional shallow water equations. *Computer Methods in Applied Mechanics and Engineering*, *199*, 3356–3368. doi:10.1016/j.cma.2010.07.007.

[23] Kubatko, E. J., Dawson, C., & Westerink, J. J. (2008). Time step restrictions for Runge–Kutta discontinuous Galerkin methods on triangular grids. *Journal of Computational Physics*, *227*, 9697–9710. doi:10.1016/j.jcp.2008.07.026.

[24] Kuzmin, D. (2010). A vertex-based hierarchical slope limiter for p-adaptive discontinuous Galerkin methods. *J. Comput. Appl. Math.*, *233*, 3077–3085.

[25] van Leer, B. (2006). Upwind and high-resolution methods for compressible flow: From donor cell to residual-distribution schemes. *Communications in Computational Physics*, *1*, 192–206.

[26] LeVeque, R. J. (2011). NTHMP tsunami benchmark problems. https://github.com/rjleveque/nthmp-benchmark-problems.

[27] Liang, Q., & Marche, F. (2009). Numerical resolution of well-balanced shallow water equations with complex source terms. *Advances in Water Resources*, *32*, 873–884. doi:10.1016/j.advwatres.2009.02.010.

[28] Liu, P. L.-F., Cho, Y.-S., Briggs, M. J., Kanoglu, U., & Synolakis, C. E. (1995). Runup of solitary waves on a circular island. *Journal of Fluid Mechanics*, *302*, 259–285. doi:10.1017/S0022112095004095.

[29] Lynett, P. J., Wu, T.-R., & Liu, P. L.-F. (2002). Modeling wave runup with depth-integrated equations. *Coastal Engineering*, *46*, 89–107. doi:10.1016/S0378-3839(02)00043-1.

[30] Matsuyama, M., & Tanaka, H. (2001). An experimental study of the highest run-up height in the 1993 hokkaido nansei-oki earthquake tsunami. In *National Tsunami Hazard Mitigation Program Review and International Tsunami Symposium (ITS)* (pp. 879–889).

[31] Meister, A., & Ortleb, S. (2014). On unconditionally positive implicit time integration for the DG scheme applied to shallow water flows. *International Journal for Numerical Methods in Fluids*, *76*, 69–94. doi:10.1002/fld.3921.

[32] Noelle, S., Pankratz, N., Puppo, G., & Natvig, J. R. (2006). Well-balanced finite volume schemes of arbitrary order of accuracy for shallow water flows. *Journal of Computational Physics*, *213*, 474–499. doi:10.1016/j.jcp.2005.08.019.

[33] Popinet, S. (2012). Adaptive modelling of long-distance wave propagation and fine-scale flooding during the Tohoku tsunami. *Natural Hazards and Earth System Science*, *12*, 1213–1227. doi:10.5194/nhess-12-1213-2012.

[34] Shu, C.-W., & Osher, S. (1988). Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of Computational Physics*, *77*, 439–471. doi:10.1016/0021-9991(88)90177-5.

[35] Synolakis, C. E., Bernard, E. N., Titov, V. V., Kânoğlu, U., & González, F. I. (2007). *Standards, Criteria, and Procedures for NOAA Evaluation of Tsunami Numerical Models*. Technical Report NOAA Technical Memorandum OAR PMEL-135 NOAA/OAR/PMEL.

[36] Thacker, W. C. (1981). Some exact solutions to the nonlinear shallow-water wave equations. *Journal of Fluid Mechanics*, *107*, 499–508. doi:10.1017/S0022112081001882.

[37] The Third International Workshop on Long-Wave Runup Models (2004). Benchmark problem #1: Tsunami runup onto a plane beach. http://isec.nacse.org/workshop/2004_cornell/bmark1.html.

[38] Toro, E. F. (2009). *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*. (3rd ed.). Springer.

[39] Vater, S., Beisiegel, N., & Behrens, J. (2015). A limiter-based well-balanced discontinuous Galerkin method for shallow-water flows with wetting and drying: One-dimensional case. *Advances in Water Resources*, *85*, 1–13. doi:10.1016/j.advwatres.2015.08.008.

[40] Xing, Y., & Shu, C.-W. (2006). A new approach of high order well-balanced finite volume weno schemes and discontinuous Galerkin methods for a class of hyperbolic systems with source terms. *Communications in Computational Physics*, *1*, 100–134.

[41] Xing, Y., & Shu, C.-W. (2011). High-order finite volume WENO schemes for the shallow water equations with dry states. *Advances in Water Resources*, *34*, 1026–1038. doi:10.1016/j.advwatres.2011.05.008.

[42] Xing, Y., & Zhang, X. (2013). Positivity-preserving well-balanced discontinuous Galerkin methods for the shallow water equations on unstructured triangular meshes. *Journal of Scientific Computing*, *57*, 19–41. doi:10.1007/s10915-013-9695-y.

[43] Xing, Y., Zhang, X., & Shu, C.-W. (2010). Positivity-preserving high order well-balanced discontinuous Galerkin methods for the shallow water equations. *Advances in Water Resources*, *33*, 1476–1493. doi:10.1016/j.advwatres.2010.08.005.

[44] Zhou, J., Causon, D., Mingham, C., & Ingram, D. (2001). The surface gradient method for the treatment of source terms in the shallow-water equations. *Journal of Computational Physics*, *168*, 1–25. doi:10.1006/jcph.2000.6670.