A well-balanced meshless tsunami propagation and inundation model

Rüdiger Brecht[† ‡]**, Alexander Bihlo**[†]**, Scott MacLachlan**[†] **and Jörn Behrens**[‡]

[†] *Department of Mathematics and Statistics, Memorial University of Newfoundland, St. John's (NL) A1C 5S7, Canada*

[‡] *Department of Mathematics, Universität Hamburg, Bundesstraße 55, Hamburg 20146, Germany and CEN – Center for Earth Systems Research and Sustainability, Universität Hamburg, Grindelberg 5, Hamburg 20144, Germany*

E-mail: *rbrecht@mun.ca, abihlo@mun.ca, smaclachlan@mun.ca, joern.behrens@uni-hamburg.de*

We present a novel meshless tsunami propagation and inundation model. We discretize the nonlinear shallow-water equations using a well-balanced scheme relying on radial basis function based finite differences. The inundation model relies on radial basis function generated extrapolation from the wet points closest to the wet–dry interface into the dry region. Numerical results against standard one- and two-dimensional benchmarks are presented.

1 Introduction

The shallow-water equations are a simplified, albeit important, model in geophysical fluid mechanics, playing a central role in both atmospheric and ocean sciences. In ocean dynamics, the shallow-water equations govern the evolution of long waves and, thus, are a particularly suitable model for tsunami propagation [21,23,28,37]. An important characterization of a tsunami event is the involvement of vastly different scales, with wave propagation happening over thousands of kilometers in the open ocean and inundation taking place on coastlines with diverse features on the scale of just a few meters. This presence of vastly different scales has justified the extensive use of unstructured meshes and adaptive mesh refinement for the numerical solution of the shallow-water equations [23,28].

Current numerical methodologies for the solution of the shallow-water equations for tsunami modeling include finite-difference methods [21,37,39], finite-volume methods [28], and finite-element and discontinuous Galerkin methods [23,41]. The application of so-called *meshless methods* to the problem of tsunami modeling has received considerably less attention.

One primary appeal of general meshless methods is that they do not rely on a predefined topologically connected mesh but, rather, operate on an (in principle) arbitrary collection of nodes where the numerical solution of the model problem is sought. For this reason, meshless methods are, by design, suitable for problems that benefit from variable spatial resolution. Several meshless methods have been proposed already for the shallow-water equations, such as those based on smoothed particle hydrodynamics [9,43] or radial basis functions [5,10,11,24]. To the best of our knowledge, the shallow-water equations with variable bottom topography have not been considered extensively within the general meshless methodology, as it is quite challenging to preserve the inherent balance between the bottom topography source term and the flux term in the momentum equations. For a review on these difficulties and possible solutions, we refer to [5,43].

The purpose of this paper is to develop a meshless tsunami model based on radial basis functions generated finite differences (RBF-FD). The RBF-FD method was first introduced in [40] and has since seen an extensive development both theoretically and with regards to different fields of applications, in particular in the geosciences [12]. This method has methodology similar to classical finite differences, yet it can be used on both arbitrary nodal layouts and general smooth embedded manifolds [34], such as on the sphere [10]. This makes RBF-FD a natural

method for both far- and near-field simulations of tsunamis. Moreover, we demonstrate in this paper that RBF-based extrapolation is also a suitable method for the simulation of tsunami inundation.

The further organization of the paper is the following. In Section 2, we present the shallow-water equations and discuss the well-balanced meshless RBF-FD methodology employed for discretizing them. Section 3 is devoted to the description of the RBF-based inundation algorithm. In Section 4, we present the results of various standard benchmark tests for the one- and two-dimensional shallow-water equations aiming at assessing the quality of the newly proposed method. The conclusions of the paper as well as a discussion of necessary further developments are found in Section 5.

2 Well-balanced meshless discretization of the shallow-water equations

In this section, we introduce the shallow-water equations and describe a well-balanced meshless method for their discretization.

2.1 The shallow-water equations

We consider the following generalized 2-dimensional conservation law

$$\boldsymbol{\rho}_t + \mathbf{F}_x + \mathbf{G}_y = \mathbf{S}. \tag{1}$$

In order to define shallow-water equations, we specify (following [33]): $\boldsymbol{\rho} = (h, hu, hv)^{\mathrm{T}}$ is the transport vector of total mass and horizontal momentum, $\mathbf{F} = (hu, hu^2 + gh^2/2, huv)^{\mathrm{T}}$ and $\mathbf{G} = (hv, huv, hv^2 + gh^2/2)^{\mathrm{T}}$ are the associated flux vectors, and $\mathbf{S} = (0, -ghb_x, -ghb_y)^{\mathrm{T}}$ is the source term. The height of a constant density water column is denoted by $h = h(t, x, y)$, the two-dimensional, vertically averaged fluid velocity is denoted by $(u, v)^{\mathrm{T}} = (u(t, x, y), v(t, x, y))^{\mathrm{T}}$, the sea bottom topography is $b = b(x, y)$, and g is the gravitational constant. For the sake of brevity, we use subscripts to denote the partial derivatives with respect to t, x and y, i.e. $h_t = \partial h/\partial t$, etc.

We point out that the form of the shallow-water equations used here does not include any bottom friction, although such friction terms can be introduced with the aim of obtaining more accurate inundation results [25]. Here, we present mostly canonical benchmarks, which are typically tested without the use of bottom friction and, so, we also exclude this in our model. The inclusion of bottom friction as well as the application to more elaborate test cases will be the subject of future investigations.

2.2 RBF-FD discretization

We discretize (1) using the RBF-FD method. Radial basis function based discretizations have seen a rapid development over the past 20 years, both for the global formulation (which is akin to the pseudospectral method [16]) and for the local RBF-FD formulation, see e.g. [12, 13] for recent reviews. The RBF methodology has also already been used for solving the shallow-water equations, both in planar geometry and on the surface of the sphere [10, 11, 24, 42, 44].

The discretization of the partial derivative operators in (1) within the framework of the RBF method proceeds as follows. Consider a set of N nodes, $\mathbf{x}_i \in \mathbb{R}^2$, covering the spatial domain $\Omega \subseteq \mathbb{R}^2$ such that no two points coincide. Given the values of the field functions $f \in \{h, u, v\}$ at those nodes, i.e. $f_j = f(\mathbf{x}_j)$, we approximate the action of a linear differential operator $\mathcal{L}$ on f at $\mathbf{x}_i$ by

$$\mathcal{L}f|_{\mathbf{x}=\mathbf{x}_i} \approx \sum_{j=1}^{N} w_{ij}^{\mathcal{L}} f_j. \tag{2}$$

In other words, the action of $\mathcal{L}$ on f at $\mathbf{x}_i$ is approximated as a weighted linear sum of the values of f at the N nodes $\mathbf{x}_j$. Within the context of RBF methods, the weights $w_{ij}^{\mathcal{L}}$ are found by enforcing (2) to be exact when evaluated for radial basis functions $\phi_k(\mathbf{x}) = \phi(||\mathbf{x} - \mathbf{x}_k||)$ centered at $\mathbf{x}_k$, i.e.

$$\mathcal{L}\phi_k(\mathbf{x}_i) = \sum_{j=1}^{N} w_{ij}^{\mathcal{L}} \phi_k(\mathbf{x}_j), \quad k = 1, \ldots, N, \tag{3}$$

which constitutes a linear system for $w_{ij}^{\mathcal{L}}$, $1 \leq j \leq N$, at each node $\mathbf{x}_i$. In the RBF-FD method, one assumes that only points close to the node $\mathbf{x}_i$ contribute to the approximation of the derivative at $\mathbf{x}_i$ and, thus, most of the N weights $w_{ij}^{\mathcal{L}}$ vanish. In practice, this is done by determining the n nearest neighbors of $\mathbf{x}_i$, where typically $n \ll N$. Solving the resulting (small) linear system (3) at each node $\mathbf{x}_i$ allows one to compose the (sparse) differentiation matrix $W^{\mathcal{L}} = (w_{ij}^{\mathcal{L}})$, such that the action of the linear differential operator $\mathcal{L}$ on the field functions f at all nodal points $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)^{\mathrm{T}}$ can be approximated as $\mathcal{L}\mathbf{f} \approx W^{\mathcal{L}}\mathbf{f}$, where $\mathbf{f} = (f(\mathbf{x}_1), \ldots, f(\mathbf{x}_N))^{\mathrm{T}}$. In the following, we have $\mathcal{L} \in \{\partial/\partial x, \partial/\partial y\}$. Note that it is also customary to include certain low-degree polynomials while solving (3), as with a pure RBF basis it is not possible to obtain the correct derivatives of constants, linear functions, etc. For further details, see e.g. [4, 12, 15].

Several RBFs are typically used, with the multiquadric, $\phi(r) = \sqrt{1 + (\epsilon r)^2}$, and the Gaussian RBF, $\phi(r) = \exp(-(\epsilon r)^2)$ being amongst the most popular choices. The parameter ϵ is called the *shape parameter* as it governs the flatness of the RBF. In the following we will work with the Gaussian RBF, since for hyperbolic PDEs the RBF-FD method typically requires the use of hyperdiffusion for stabilization which is most easily accomplished using the Gaussian RBF [14].

When implementing a numerical scheme for the shallow-water equations for ocean modeling, the presence of a nonflat sea bottom topography generally presents a challenge for both meshbased and meshless numerical schemes. More specifically, preserving the so-called *lake at rest solution* is a nontrivial, but crucial endeavor, since the violation of the lake at rest solution typically leads to the stimulation of spurious numerical waves that can render the correct simulation of the actual physical waves extremely challenging. There has been a considerable body of literature devoted to the construction of so-called *well-balanced* numerical schemes for the shallow-water equations, which are schemes that can preserve the lake at rest solution numerically, see [2, 20, 26, 27, 41, 43] for some examples for such well-balanced schemes.

Most recently, in [5], a unifying strategy was proposed for developing general well-balanced meshbased and meshless schemes which is in particular suitable for the RBF-FD methodology. For the sake of completeness of the present exposition, we briefly review the key idea of [5] here for the case of the one-dimensional form of the shallow-water equations,

$$h_t + (hu)_x = 0, \qquad (hu)_t + \left(hu^2 + \frac{1}{2}gh^2\right)_x = -ghb_x.$$

The two-dimensional case is treated analogously by enforcing the well-balanced condition given below for both the x- and y-derivatives.

As $u = 0$ in the lake at rest solution, any well-balanced discretization has to preserve the identity

$$\frac{1}{2}\partial_x h^2 = -h\partial_x b, \tag{4}$$

numerically in the case that $h + b = c$, for $c = $ const. It is found in [5] that this is in general only possible if one discretizes the balance equation (4), at each point $\mathbf{x}_i$, so that

$$\frac{1}{2}\left(\mathrm{D}_x^{\mathrm{f}} h^2\right)_i = -\overline{h}_i \left(\mathrm{D}_x^{\mathrm{s}}(c - h)\right)_i,$$

for any mesh function, h, and constant, c, where $\mathrm{D}_x^{\mathrm{f}}$ and $\mathrm{D}_x^{\mathrm{s}}$ are the discrete approximations to the spatial first derivative operators in the flux and source terms of the shallow-water equations, respectively, and $\overline{h}_i = \sum_{j=1}^n m_{ij} h_j$ is a specific, consistent average (meaning that $\sum_{j=1}^n m_{ij} = 1$) of the total water height h over the stencil of $\mathbf{x}_i$.

As noted in [5], the above equality implies that

$$(\mathrm{D}_x^{\mathrm{s}} c)_i = 0, \qquad \frac{1}{2}\left(\mathrm{D}_x^{\mathrm{f}} h^2\right)_i = \overline{h}_i \left(\mathrm{D}_x^{\mathrm{s}} h\right)_i, \tag{5}$$

must hold at all nodes $\mathbf{x}_i$. The first condition requires the consistent (exact) derivative of constants by the derivative operator $\mathrm{D}_x^{\mathrm{s}}$ (which in our case of RBF-FD based derivatives requires the inclusion of zero-degree polynomials in the basis), and the second condition can be satisfied provided we interpret $\mathrm{D}_x^{\mathrm{f}} h^2$ as a bilinear form, i.e.

$$\left(\frac{1}{2}\mathrm{D}_x^{\mathrm{f}} h^2\right)_i = \frac{1}{2}\mathbf{h}^{\mathrm{T}} W_i^{\mathrm{f}} \mathbf{h}.$$

This procedure defines a third order differentiation tensor, $\mathbf{W}^{\mathrm{f}}$, with the matrix W_i^{f} being its ith slice.

For the approximation of the source derivative, $\mathrm{D}_x^{\mathrm{s}}$, we write

$$(\mathrm{D}_x^{\mathrm{s}} h)_i = (\mathbf{w}_i^{\mathrm{s}})^{\mathrm{T}} \mathbf{h},$$

where $\mathbf{w}_i^{\mathrm{s}} = (W_{ij}^{\mathrm{s}})_{1 \leqslant j \leqslant n}$ is the ith row of the associated differentiation matrix W^{s}. In a similar manner, we can write $\mathbf{m}_i = (m_{ij})_{1 \leqslant j \leqslant n}$ for the ith row of the averaging matrix M. The second condition of (5) then naturally translates to

$$\frac{1}{2}\mathbf{h}^{\mathrm{T}} W_i^{\mathrm{f}} \mathbf{h} = (\mathbf{m}_i^{\mathrm{T}} \mathbf{h})((\mathbf{w}_i^{\mathrm{s}})^{\mathrm{T}} \mathbf{h}),$$

for all i, which (taking W_i^{f} to be symmetric) requires that

$$W_i^{\mathrm{f}} = \mathbf{m}_i \left(\mathbf{w}_i^{\mathrm{s}}\right)^{\mathrm{T}} + \mathbf{w}_i^{\mathrm{s}} \mathbf{m}_i^{\mathrm{T}}, \tag{6}$$

holds at all nodes $\mathbf{x}_i$.

Practically speaking, one is thus free to choose the averaging matrix M and the derivative matrix W_x^{s} corresponding to the derivative approximation used in the source term and then Eq. (6) prescribes how to choose the weights in the flux derivative $\mathrm{D}_x^{\mathrm{f}} h^2$, given through the derivative tensor $\mathbf{W}^{\mathrm{f}}$, so as to obtain a well-balanced scheme for the shallow-water equations. For a more in-depth discussion and results regarding the consistency of the resulting discrete derivative operators, consult [5].

We use the RBF-FD method for discretizing the two-dimensional shallow-water equations (1), invoking condition (6) to guarantee that the resulting meshless scheme will be well-balanced. For the time-stepping, the second-order explicit midpoint scheme is used. It is pointed out in [14] that the application of the RBF-FD method to purely convective PDEs is prone to numerical instability as the eigenvalues of the associated derivative matrices tend to scatter to the complex right-half plane. As a remedy, the inclusion of hyperdiffusion was proposed, which we have included in our discretization as well. In other words, instead of solving the shallow-water equations (1), we solve $\boldsymbol{\rho}_t + \mathbf{F}_x + \mathbf{G}_y = \tilde{\mathbf{S}}$, where $\tilde{\mathbf{S}}$ is a modified source term of the form $\tilde{\mathbf{S}} = \mathbf{S} + \mathbf{D}$, where

$$\mathbf{D} = (-1)^{\ell+1} \nu \Delta^\ell (0, hu, hv)^{\mathrm{T}}, \qquad \ell \geqslant 1$$

with ν being the diffusion parameter and $\Delta = \partial^2/\partial x^2 + \partial^2/\partial y^2$ being the two-dimensional Laplacian operator. If the underlying RBF is the Gaussian RBF, then $\Delta^\ell \phi(r) = \epsilon^{2\ell} p_\ell(r) \phi(r)$, with $p_\ell(r)$ being computable through the recurrence relation

$$p_0(r) = 1, \quad p_1(r) = 4(\epsilon r)^2 - 4, \quad p_{\ell+1}(r) = 4((\epsilon r)^2 - 2\ell - 1)p_\ell(r) - 16\ell^2 p_{\ell-1}(r), \quad \ell \geqslant 1. \tag{7}$$

4

The polynomials $p_\ell(r)$ are related to the Laguerre orthogonal polynomials, see [14] for further details. Below, we use $\ell = 2$, unless otherwise noted.

To summarize, we present below the algorithm used at every time step. The initialization phase of the algorithm uses the RBF-FD methodology to define the averaging matrix, M, source derivative, W_i^s, as well as the discrete approximation of the hyperdiffusion operator, $\mathbf{D}$. Additionally, we precompute b_x^d and b_y^d, the RBF-FD derivatives of the bottom topography function, $b(x, y)$. In the algorithm below, we use D_x^d and D_y^d to denote the discrete derivative operators of the well-balanced scheme, with the discrete hyperdiffusion operator, $D_{\Delta^2}^d$, computed using (7). At each node, i, we store unknowns h_i, $(hu)_i$, and $(hv)_i$; the products and ratios of vectors h, hu, and hv given in the algorithm below are computed componentwise. We note that the use of the inundation algorithm presented below ensures that no division by zero is ever performed. In what follows, at each time step, we make use of the natural partitioning of the set of all nodes, $X = \{\mathbf{x}_i\}_{i=1}^N$, into sets of wet and dry points,

$$X_{\text{wet}} = \{\mathbf{x}_i \in X : h_i > \delta\}, \quad X_{\text{dry}} = X \backslash X_{\text{wet}},$$

where $\delta \in \mathbb{R}$ is a user-defined (small) positive parameter. We note that this partitioning is updated after every stage of the time-step, but we supress superscripts or subscripts denoting the time-step and stage to simplify notation.

For every timestep:

Compute first stage of explicit midpoint rule:
Compute $hu2 = h \cdot (\frac{hu}{h})^2, hv2 = h \cdot (\frac{hv}{h})^2, hvu = h \cdot \frac{hv}{h} \cdot \frac{hu}{h}$ and $\bar{h} = Mh$, then set $(\mathfrak{h}_x)_i = (\bar{h})_i \cdot (D_x^d h)_i$ and $(\mathfrak{h}_y)_i = (\bar{h})_i \cdot (D_y^d h)_i$ for all nodes $\mathbf{x}_i \in X_{\text{wet}}$.

$$\widetilde{h} = h - \frac{\Delta t}{2}\big(D_x^d(hu) + D_y^d(hv)\big)$$

$$\widetilde{hu} = hu - \frac{\Delta t}{2}\big(D_x^d(hu2) + g\mathfrak{h}_x + D_y^d(hvu) + g\bar{h}b_x^d - \eta D_{\Delta^2}^d(hu)\big)$$

$$\widetilde{hv} = hv - \frac{\Delta t}{2}\big(D_y^d(hv2) + g\mathfrak{h}_y + D_x^d(hvu) + g\bar{h}b_y^d - \eta D_{\Delta^2}^d(hv)\big)$$

Apply boundary conditions.

Apply inundation algorithm.

Compute second stage of explicit midpoint method:
Compute $\widetilde{hu2} = \widetilde{h} \cdot (\frac{\widetilde{hu}}{\widetilde{h}})^2, \widetilde{hv2} = \widetilde{h} \cdot (\frac{\widetilde{hv}}{\widetilde{h}})^2, \widetilde{hvu} = \widetilde{h} \cdot \frac{\widetilde{hv}}{\widetilde{h}} \cdot \frac{\widetilde{hu}}{\widetilde{h}}$ and $\bar{\widetilde{h}} = M\widetilde{h}$, then set $(\widetilde{\mathfrak{h}_x})_i = (\bar{\widetilde{h}})_i \cdot (D_x^d\widetilde{h})_i$ and $(\widetilde{\mathfrak{h}_y})_i = (\bar{\widetilde{h}})_i \cdot (D_y^d\widetilde{h})_i$ for all nodes $\mathbf{x}_i \in X_{\text{wet}}$.

$$h \leftarrow h - \Delta t\big(D_x^d(\widetilde{hu}) + D_y^d(\widetilde{hv})\big)$$

$$hu \leftarrow hu - \Delta t\Big(D_x^d(\widetilde{hu2}) + g\widetilde{\mathfrak{h}_x} + D_y^d(\widetilde{hvu}) + g\bar{\widetilde{h}}b_x^d - \eta D_{\Delta^2}^d(\widetilde{hu})\Big)$$

$$hv \leftarrow hv - \Delta t\Big(D_y^d(\widetilde{hv2}) + g\widetilde{\mathfrak{h}_y} + D_x^d(\widetilde{hvu}) + g\bar{\widetilde{h}}b_y^d - \eta D_{\Delta^2}^d(\widetilde{hv})\Big)$$

Apply boundary conditions.

Apply inundation algorithm.

We note that, since the scheme is explicit, this allows a very efficient implementation of the time-stepping using only sparse matrix multiplications with M and W_i^s, along with componentwise vector operations. Furthermore, the fine-scale parallelism of these operations allows a natural avenue for parallelization, on both classical compute clusters and modern manycore and accelerated architectures.

3 Meshless inundation algorithm

Of particular importance in a tsunami model is the treatment of the wet–dry interface when the incoming wave hits the coastline. Several algorithms have been proposed to deal with this moving boundary condition in numerical models that solve the governing equations in the strong form (for associated results for the governing equations in the weak form, consult, e.g. [3,7,41]). In [38], the authors use a finite-difference model with variable grid spacing near the boundary that ensures existence of a shoreline boundary point on the surface of the beach at all times. Similarly, in [31], based on the earlier work of [35] for the shallow-water equations, a finite-difference model for the nonlinear Boussinesq equations with fixed grid spacing was used that handles the moving boundary by employing (one-dimensional) linear extrapolation of the wave run-up from the last wet points to the first dry points on the beach. This idea was re-considered in [19] where true two-dimensional bilinear extrapolation was used to compute the run-up in a two-dimensional finite-difference model for the Boussinesq equations.

Since our numerical scheme is based on the RBF methodology using the strong form of the shallow-water equations, it is natural to use RBFs also for the inundation model. This is, in particular, justified as (multiquadric) RBFs were originally proposed by Hardy for two-dimensional scattered data interpolation [22] and later found by Franke to be the most accurate of all the 29 methods tested in [18]. Of further interest are the studies carried out in [17] where the authors investigated the Runge phenomenon in the context of RBF interpolation. It has been found that the Runge phenomenon can be controlled by a suitable choice of the shape parameter in the RBF (with spatially varying shape parameters most favorable) and non-equally chosen data points, typically much better than the Runge phenomenon can be controlled in standard polynomial interpolation.

In light of the favorable performance of RBF-based interpolation schemes, we propose to use the extrapolation technique of [19,31,35] but instead of using polynomial-based extrapolation, we use *RBF-based extrapolation*. We again make use of the partitioning of the set of all nodes, X, into wet and dry points, as $X = X_{\mathrm{wet}} \cup X_{\mathrm{dry}}$, with $X_{\mathrm{wet}} = \{\mathbf{x}_i \in X : h_i > \delta\}$. Since the shallow-water equations are not defined at the dry points (where h_i is too small or negative), we need to extrapolate the wet values to the dry points to have a numerical solution defined at all points X. The advantage of this procedure is that then the same derivative approximation can be invoked at all points, even those close to the wet–dry interface, where some of the nearest neighbors on which the RBF derivatives are defined will be dry points.

For each dry point immediately neighboring a wet point we find its n_{e} nearest wet points through which we define an RBF interpolant. Once the RBF interpolant based on the wet points is defined, we use it to extrapolate the field functions u, v and $h + b$ to the dry point under consideration. We found experimentally that the multiquadric RBF basis augmented with first degree polynomials (i.e. constants and the monomimals x and y) is most suitable for the extrapolation procedure. Note that in order to preserve the lake at rest solution in the presence of dry points it is necessary to extrapolate $h + b$, not h alone, to the dry points. The extrapolation is done by the following algorithm:

for each $\mathbf{x}^{\mathrm{dry}} \in X_{\mathrm{dry}}$ immediately neighboring a wet point

> **find $\{\mathbf{x}_{j_1}^{\mathrm{wet}}, \ldots, \mathbf{x}_{j_{nn}}^{\mathrm{wet}}\}$ nearest neighbours of $\mathbf{x}_k^{\mathrm{dry}}$ in X_{wet}**
>
> **extrapolate $h(\mathbf{x}^{\mathrm{dry}}), hu(\mathbf{x}^{\mathrm{dry}}), hv(\mathbf{x}^{\mathrm{dry}})$:**

for $f \in \{h, hu, hv\}$ solve the linear system for w

$$
\begin{pmatrix}
\phi(r_{1,1}) & & \phi(r_{1,nn}) & 1 & (\mathbf{x}_{j_1}^{\text{wet}})^T \\
\vdots & \ddots & & \vdots & \vdots \\
\phi(r_{nn,1}) & & \phi(r_{nn,nn}) & 1 & (\mathbf{x}_{j_{nn}}^{\text{wet}})^T \\
1 & \cdots & 1 & 0 & 0 \\
\mathbf{x}_{j_1}^{\text{wet}} & \cdots & \mathbf{x}_{j_{nn}}^{\text{wet}} & 0 & 0
\end{pmatrix}
\begin{pmatrix}
w_1 \\
\vdots \\
w_{nn} \\
w_{nn+1} \\
\mathbf{w}_{nn+2}
\end{pmatrix}
=
\begin{pmatrix}
f(\mathbf{x}_{j_1}^{\text{wet}}) \\
\vdots \\
f(\mathbf{x}_{j_{nn}}^{\text{wet}}) \\
0 \\
0
\end{pmatrix}
\quad \text{with } r_{i,l} = \|\mathbf{x}_{j_i}^{\text{wet}} - \mathbf{x}_{j_l}^{\text{wet}}\|
$$

$$
\text{Set } f(\mathbf{x}^{\text{dry}}) = \sum_{i=1}^{nn} w_i \phi(r_i) + w_{nn+1} + \mathbf{w}_{nn+2} \cdot \mathbf{x}^{\text{dry}}
$$

Here, we note that the coefficient $\mathbf{w}_{nn+2}$ has the same dimension as $\mathbf{x}^{dry}$.

4 Numerical simulations

Having described the meshless discretization of the shallow-water equations (1) and the associated inundation model, we now proceed to present the results of several classical benchmark tests for both the one-dimensional and two-dimensional form of the shallow-water equations.

4.1 One-dimensional benchmarks

We repeat here some of the one-dimensional tests carried out in [41]. In all experiments we compute the RBF-FD approximation $\mathrm{D}_x^{\mathrm{s}}$, obtained from the procedure outlined in Section 2.2. The RBF used is the Gaussian RBF with shape parameter $\epsilon = 0.1/\Delta x$, where Δx is the (uniform) nodal spacing. Eq. (3) is solved based on the three nearest neighbors of each nodal point (which includes each node itself). The averaging matrix M needed for (6) is obtained from a normalized Gaussian filter over the three nearest neighbors of each node. With this, each time step of the explicit midpoint scheme can be implemented with $\mathcal{O}(1)$ work per spatial mesh point.

4.1.1 Lake at rest solution

It was shown in [5] that the numerical scheme based on a discretization that respects (5)–(6) is well-balanced. Since there the shallow-water equations were considered without an inundation model, we carry out a test case where part of the domain is initially dry, so as to check that the discretization remains well-balanced in the presence of wet–dry interfaces.

Specifically, we consider the domain $\Omega = [0, 1]$, with the smooth bottom topography

$$
b_{\mathrm{s}} = \begin{cases} a \dfrac{\exp(-0.5/(r_m^2 - r^2))}{\exp(-0.5/r_m^2)} & \text{if} \quad r < r_m \\ 0 & \text{otherwise} \end{cases}, \tag{8}
$$

where we set $r = |x - 0.5|$, $a = 1.2$ and $r_m = 0.4$. We use $h = \max(0, 1 - b(x))$ and $u = 0$ as initial conditions. A total of $N = 50$ (regularly spaced) grid points were used, the time step is $\Delta t = 0.002$ and the final integration time is $t = 20$. The extrapolation tolerance is $\delta = 0.0025$ with the three nearest neighbors being used for the extrapolation. The shape parameter for the RBF extrapolation is $\epsilon_{\mathrm{e}} = 1$. Since realistic bathymetry is usually not smooth, in a second experiment, we add some noise on top of the smooth bottom topography b_s. In particular, we consider a noisy bathymetry of the form

$$
b_{\mathrm{n}} = b_{\mathrm{s}} + \sum_{j=1}^{3} a_j \sin(16j\pi x + p_j),
$$

7

where $a = (0.1, 0.2, 0.3)$ and $p = (1.6, 3.2, 0.5)$. Reflecting boundary conditions were used for both experiments. The results of the two numerical experiments are depicted in Figures 1 and 2, which demonstrate that the extrapolating boundary conditions preserve the well-balanced properties of the meshless RBF-FD discretization.
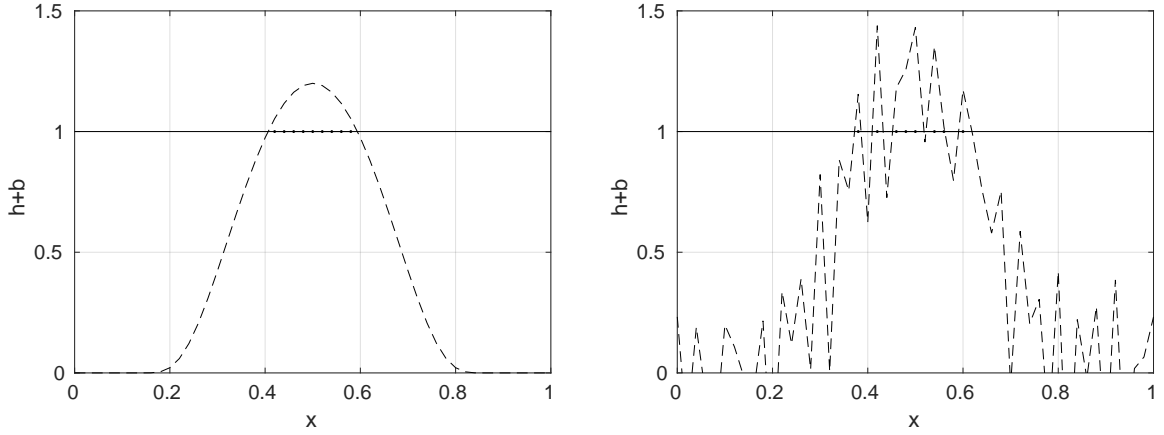

Figure 1. Final surface elevation for the lake at rest solution with smooth (left) and noisy (right) bottom topography; bathymetry (dashed), surface elevation (solid) and interpolated nodes (solid with points).
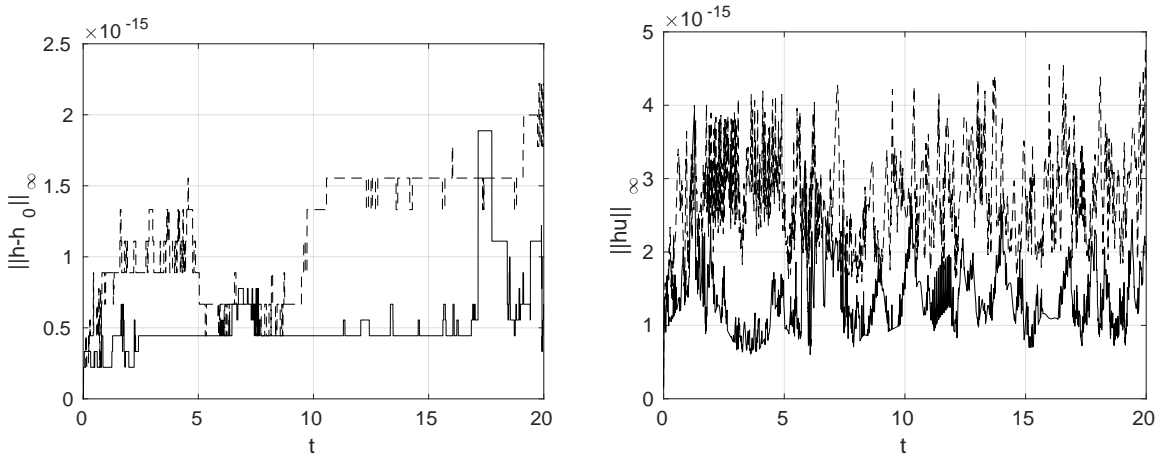
Figure 2. l_∞-error time series for the surface elevation (left) and momentum (right) for the lake at rest solution; smooth bathymetry (solid), noisy bathymetry (dashed).

4.1.2 Oscillatory flow in a parabolic basin

We consider the oscillatory flow in a parabolic basin, which is described by the following exact solution first reported in [36]. The domain for this problem is $\Omega = [-5000, 5000]$ with parabolic bottom topography $b = h_0(x/a)^2$, where $a = 3000$ and $h_0 = 10$. The initial conditions are chosen so that the exact solution to the shallow-water equations for this benchmark is

$$h_a(t, x) = h_0 - \frac{B^2}{4g}(1 + \cos 2\omega t) - \frac{Bx}{2a}\sqrt{\frac{8h_0}{g}}\cos(\omega t), \quad u_a(t, x) = \frac{Ba\omega}{\sqrt{2h_0 g}}\sin \omega t$$

where $\omega = \sqrt{2gh_0}/a$ and $B = 5$.

In the first experiment we use $N = 200$ equally spaced nodes and integrate the one-dimensional shallow-water equations up to $t = 3000$ using the time step $\Delta t = 1$. The extrapolation parameter is set to $\delta = 0.01$. Since the water surface never reaches the boundaries of the domain, no

specific boundary conditions have to be imposed. Snapshots of the numerical solution and the exact solution at times $t = 0$, $t = 1000$, $t = 2000$ and $t = 3000$ are depicted in Figure 3. The associated time series of the l_∞ relative height and mass errors are shown in Figure 4.
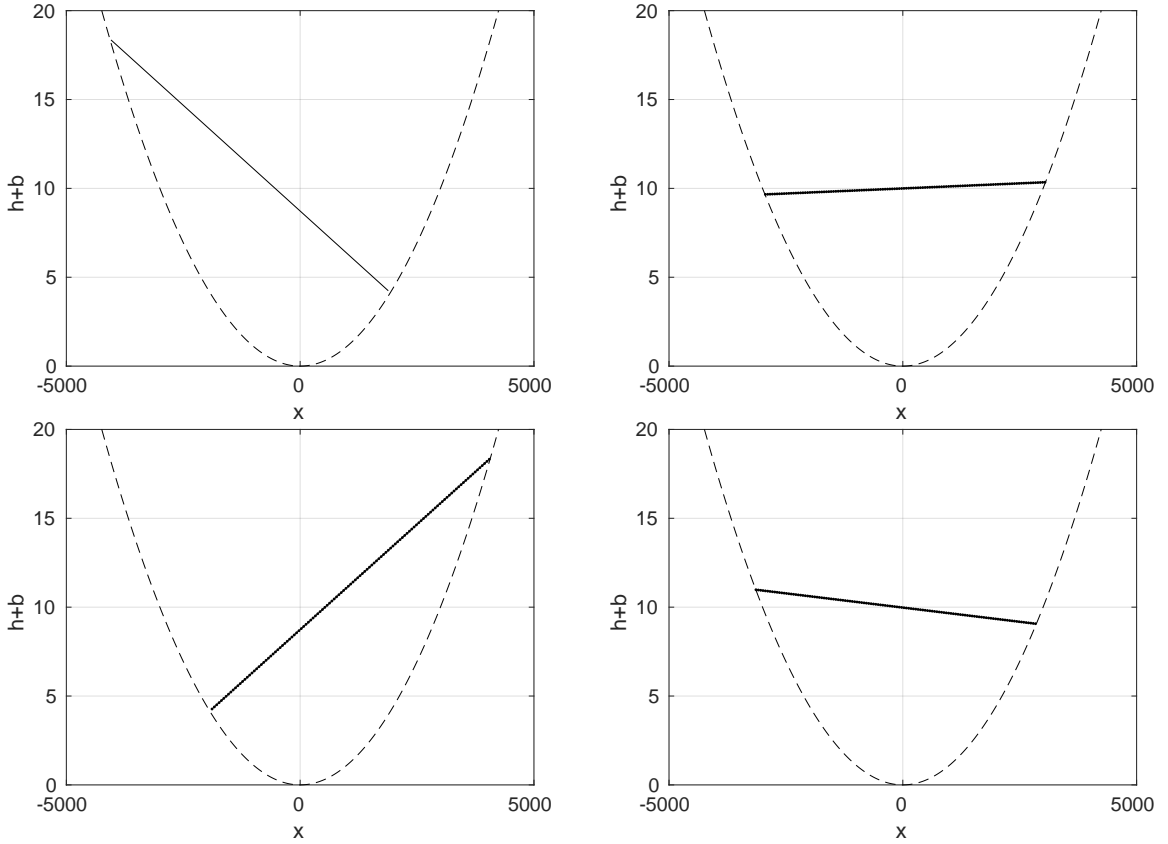

Figure 3. Snapshots of the numerical solution (solid line) and analytical solution (dotted line) of the surface elevation for the parabolic bowl bathymetry (dashed line) at times $t = 0$, $t = 1000$, $t = 2000$ and $t = 3000$.
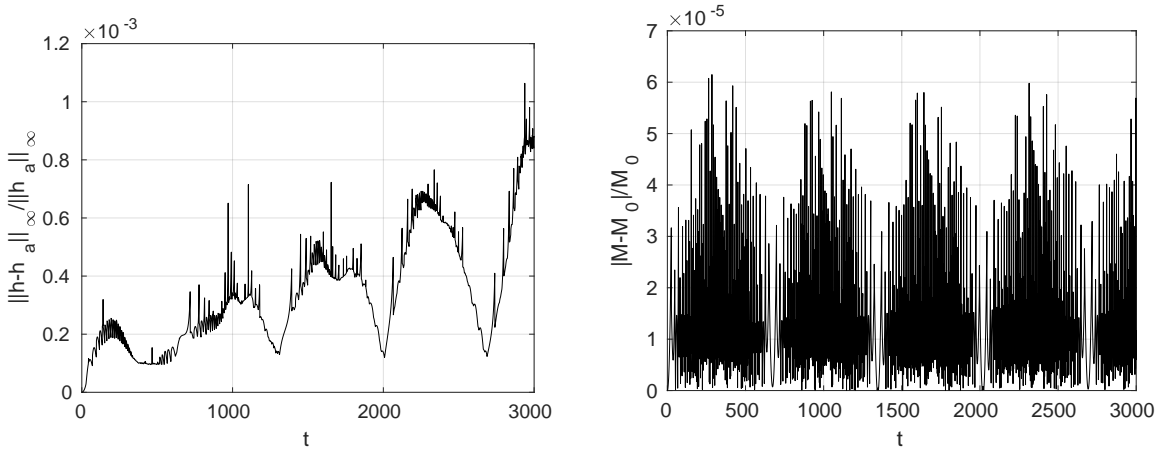
Figure 4. l_∞-error time series for the relative surface elevation error (left) and relative mass error (right).

Note that although our numerical scheme does not preserve mass by construction, the relative change in the total mass over time is small, bounded and oscillatory only. That is, no spurious trend in the mass is introduced by the numerical scheme and the inundation model.

To numerically verify the convergence of the numerical solution, we next carry out a sequence of numerical integrations using $N \in \{200, 400, 800, 1600\}$ equally spaced points. The time steps

are halved each time the number of nodes is doubled. The final integration time is again $t = 3000$. In Table 1, we record the maximum l_∞ errors (relative height error, relative mass error, absolute momentum error), occurred over the integration period. The table shows also the experimental convergence rate $\frac{\|e_c\|}{\|e_f\|}$, where e_c is the error of the coarse and e_f of the fine grid. Similar data is shown in Table 2, for the l_2 norm; we note that both measures of the error show similar overall convergence, but the l_2 norm shows steadier convergence rates than the maximum norm data in Table 1.

Table 1. Convergence study for the oscillatory flow in a parabolic basin.

N	$\|\|h - h_{\mathrm{a}}\|\|_\infty / \|\|h_{\mathrm{a}}\|\|_\infty$	$\frac{\|e_c\|}{\|e_f\|}$	$\|\|M - M_0\|\|_\infty / \|\|M_0\|\|_\infty$	$\frac{\|e_c\|}{\|e_f\|}$	$\|\|hu - (hu)_{\mathrm{a}}\|\|_\infty$	$\frac{\|e_c\|}{\|e_f\|}$
200	$1.1 \cdot 10^{-3}$		$6.1 \cdot 10^{-5}$		$5.9 \cdot 10^{-2}$	
400	$3.3 \cdot 10^{-4}$	3.33	$1.7 \cdot 10^{-5}$	3.59	$3.4 \cdot 10^{-2}$	1.74
800	$7.2 \cdot 10^{-5}$	4.58	$6.9 \cdot 10^{-6}$	2.46	$5.2 \cdot 10^{-3}$	6.54
1600	$3.9 \cdot 10^{-5}$	1.85	$2.6 \cdot 10^{-6}$	2.65	$4.9 \cdot 10^{-3}$	1.06

Table 2. Convergence study for the oscillatory flow in a parabolic basin.

N	$\|\|h - h_{\mathrm{a}}\|\|_2 / \|\|h_{\mathrm{a}}\|\|_2$	$\frac{\|e_c\|}{\|e_f\|}$	$\|\|M - M_0\|\|_2 / \|\|M_0\|\|_2$	$\frac{\|e_c\|}{\|e_f\|}$	$\|\|hu - (hu)_{\mathrm{a}}\|\|_2$	$\frac{\|e_c\|}{\|e_f\|}$
200	$6.1 \cdot 10^{-4}$		$6.1 \cdot 10^{-5}$		$3.9 \cdot 10^{-1}$	
400	$1.5 \cdot 10^{-4}$	4.07	$1.6 \cdot 10^{-5}$	3.81	$1.4 \cdot 10^{-1}$	2.79
800	$3.8 \cdot 10^{-5}$	3.95	$6.8 \cdot 10^{-6}$	2.35	$5.1 \cdot 10^{-2}$	2.75
1600	$1.1 \cdot 10^{-5}$	3.45	$2.6 \cdot 10^{-6}$	2.61	$2.1 \cdot 10^{-2}$	2.43

4.1.3 Tsunami run-up on a sloping beach

The run-up of waves on a sloping beach is a classical test case for numerical schemes for the shallow-water equations within the area of tsunami modeling. As in the case of the parabolic bowl, there exists an analytical solution for this test case, which was first found in the seminal paper [8].

Here, we follow [41] and consider the computational domain $\Omega = [-500, 50\ 000]$, with the bottom topography being defined as $b = 5000 - 0.1x$. Since we will consider the solution at times $t = 160$, $t = 175$ and $t = 220$, simple reflecting boundary conditions can be employed at the right boundary since the reflected waves cannot travel to the sloping beach in that time. We choose $\Delta x = 20$ with a time step of $\Delta t = 0.025$. The extrapolation parameter was set to $\delta = 0.1$. The initial condition, numerical solutions and exact solutions at the sampling times are displayed in Figure 5, which demonstrates that the inundation process is correctly approximated by the numerical model.

4.2 Two-dimensional benchmarks

We consider three two-dimensional benchmarks, again the lake at rest, but on a non-uniform mesh, flow around a conical island, and the Monai Valley experiment. Again, the well-balanced RBF-FD discretization described in Section 2.2 is used. The RBF-FD discretization again uses the Gaussian RBF with shape parameter $\epsilon = 0.1/\sqrt{\Delta x^2 + \Delta y^2}$, where Δx and Δy are the spacings in x- and y-direction. A total of nine nearest neighbors are used for the derivative approximation (which again includes each node as the center of the stencil itself), unless otherwise
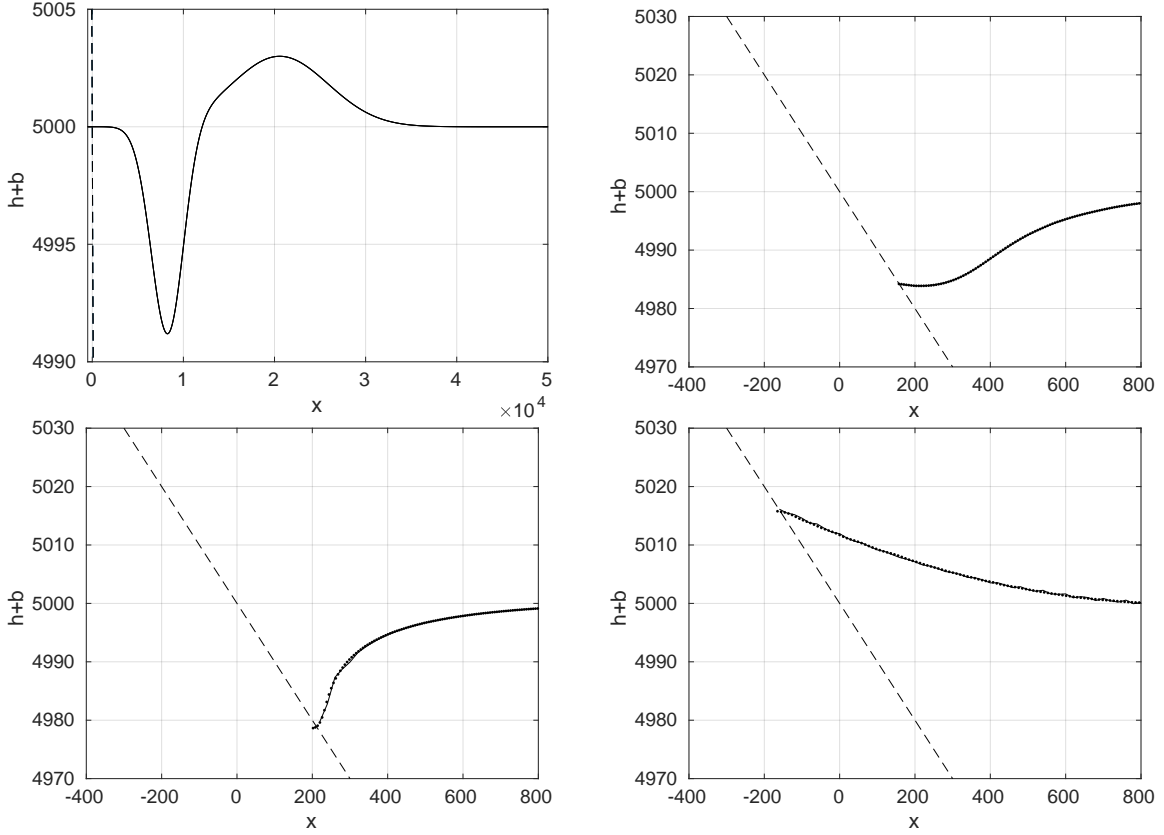
Figure 5. Snapshots of the numerical solution (solid line) and analytical solution (dotted line) of the surface elevation for the sloping beach bathymetry (dashed line). Top left shows the entire computational domain at time $t = 0$, while the inundation area is shown for $t = 160$ (top right), $t = 175$ (lower left) and $t = 220$ (lower right).

indicated. A two-dimensional normalized Gaussian filter over these nine nearest neighbors is used to construct the averaging matrix, M.

4.2.1 Two-dimensional lake at rest

We repeat the smooth lake at rest test case in two dimensions but, in addition, we use a non-uniform mesh. We consider the domain $\Omega = [0, 1]^2$ covered by $n = 2500$ nodes. To demonstrate that our scheme is meshless, we start from a uniform 50×50 mesh of the unit square and add $(0.1\Delta x \mathcal{N}(0, 1), 0.1\Delta y \mathcal{N}(0, 1))$ as a disturbance to the coordinates of each grid point, where $\Delta x = \Delta y = 0.02$ and $\mathcal{N}(0, 1)$ is a normally distributed random variable with zero mean and variance one. We use reflecting boundary conditions. The bottom topography is given by

$$b(x, y) = b_s(\|(x - 0.5, y - 0.5)\|_2)$$

where b_s is defined in Equation (8). Again we use $h = \max(0, 1 - b(x, y))$ and $u = 0$ as initial conditions. The time step is $\Delta t = 0.0015$ and the final integration time is $t = 20$. The extrapolation parameter is set to $\delta = 0.05$. In this case, we use 25 neighbours for the derivative approximation, to ensure sufficiently accurate approximations on the non-uniform mesh.

The initial surface elevation and node distribution can be seen in Figure 6. Figure 7 shows the errors in the computed solution over time.
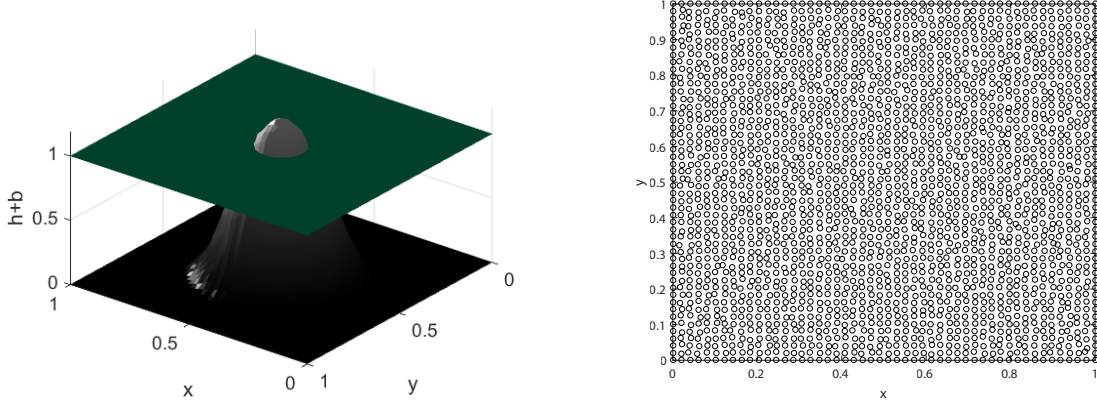
11

Figure 6. Initial surface elevation for the lake at rest solution with smooth bottom topography (left); node distribution for the two-dimensional *lake at rest* test case.
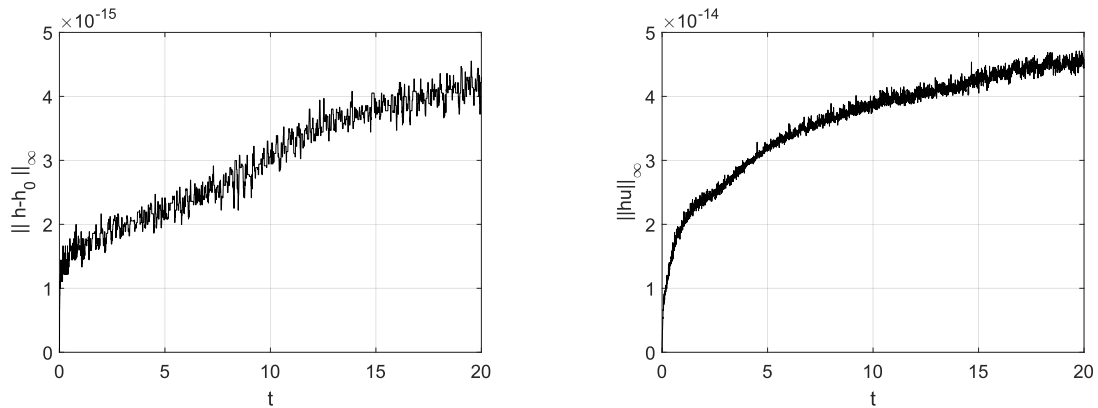

Figure 7. l_∞-error time series for the surface elevation (left) and momentum (right) for the lake at rest solution.

4.2.2 Flow around a conical island

The flow around a conical island is another classical benchmark test for tsunami models, following the experimental study described in [6]; see also [29]. The experiment is an idealization of the 1992 Flores Island tsunami run-up on Babi Island. The setup of the experiment is a 25m long and 30m wide basin with a flat ground. The border of the basin absorbs the incoming wave, therefore there are no reflections.

The origin of this conical island is located at $x = 12.96$m and $y = 13.8$m. In Figure 8, the setup, shape and properties of the island can be seen. The initial water depth is $h_0 = 0.32$m. The y-axis is parallel to the wavemaker, which generates solitary waves. We will compare the data of 4 of the 27 gauges originally used in the experiment, to measure the surface wave elevations. The position of the four gauges are indicated in Figure 8.

We choose $\Delta x = 0.125, \Delta y = 0.14$ and $\Delta t = 0.02$. Then we simulate case A, with height-to-depth ratio $H = 0.04$, and case C, with $H = 0.18$. The recorded water heights at gauges 6, 9, 16 and 22 can be seen in Figure 9. Figure 10 shows snapshots of the propagating wave for both cases. The maximal horizontal run-up agrees well with the measured data of the experiment, which can be seen in Figure 11.
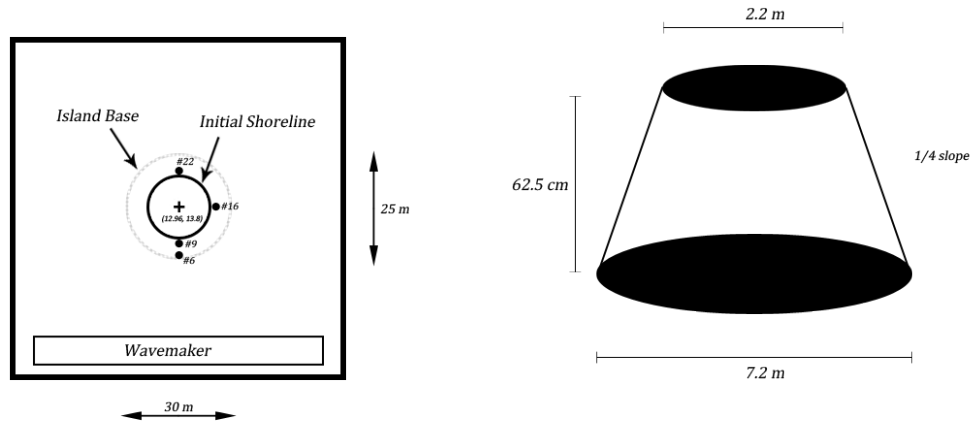
12

Figure 8. Setup of the experiment and properties of the conical island. Positions of the gauges: (9.36, 13.8)#6; (10.72, 13.8)#9; (12.96, 11.53)#16; (15.2, 13.8)#22
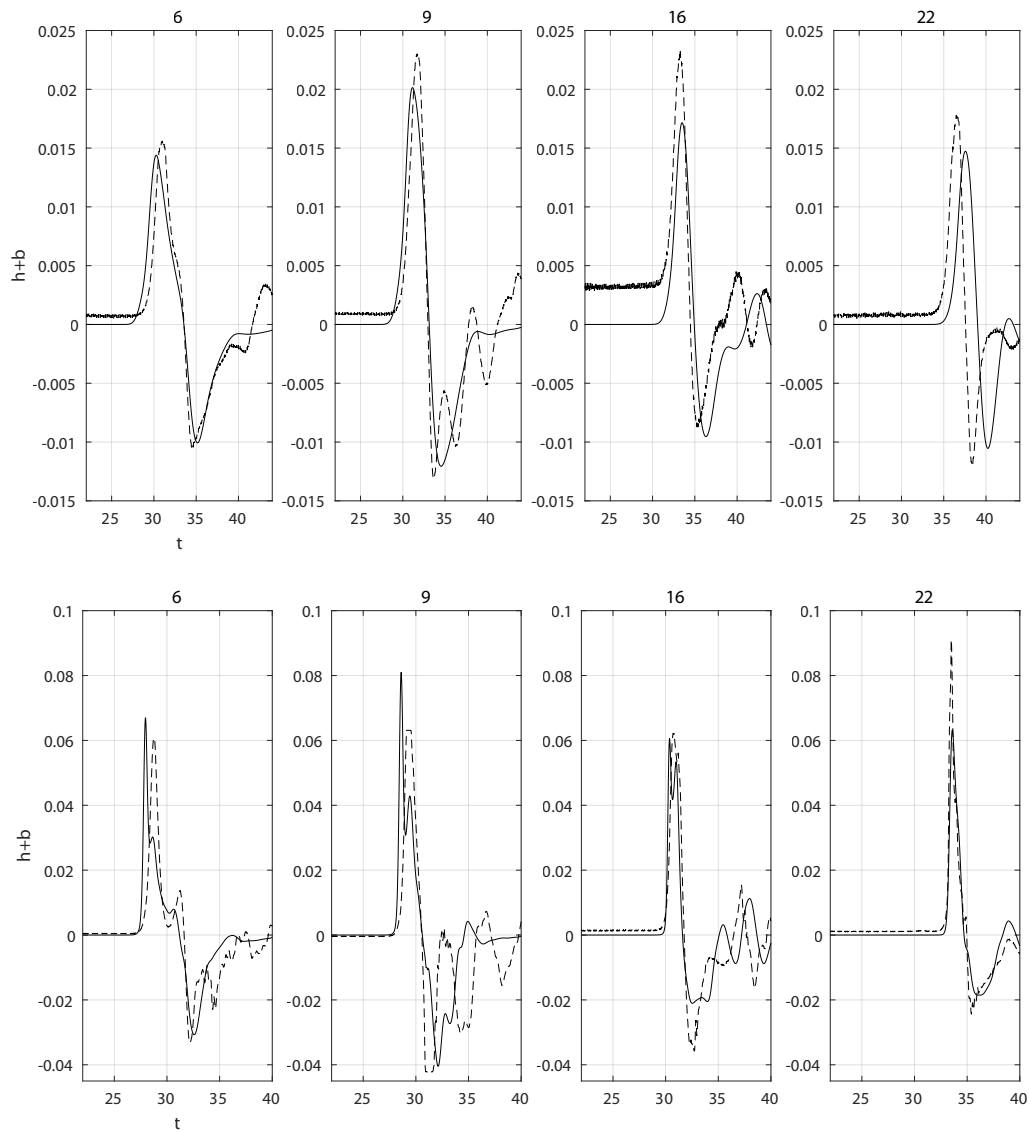

Figure 9. Comparison of measured (dashed) and simulated (solid) water elevation at the gauges for case A (above) and case C (below) for the conical island.
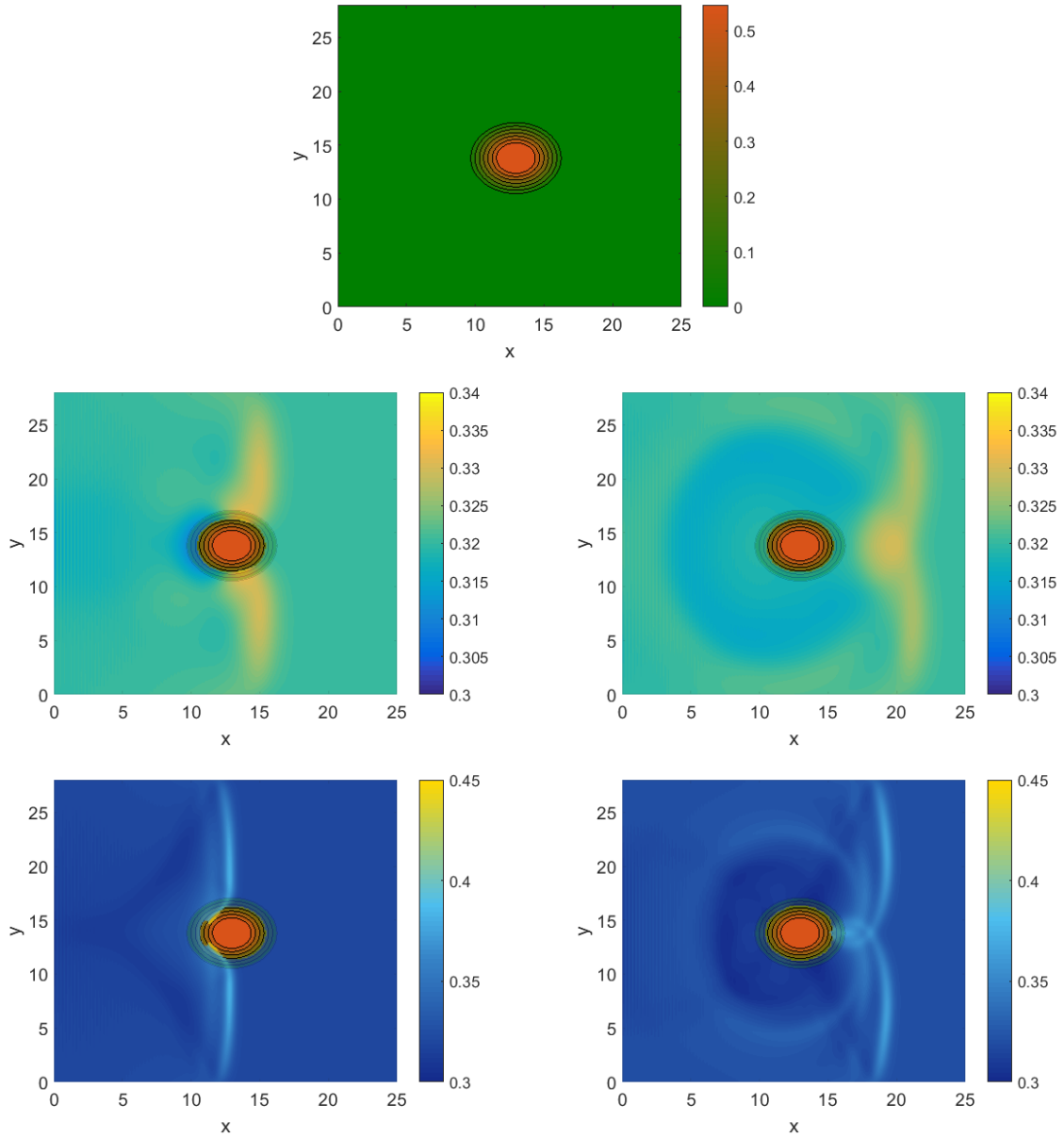
Figure 10. Image at the top shows the initial water elevation and the height of the island. Snapshots of the computed solution $(h + b)$ for the conical island test case at times $t = 12$ and $t = 16$ for case A (above) and at times $t = 10$ and $t = 14$ for case C (below) for the conical island.
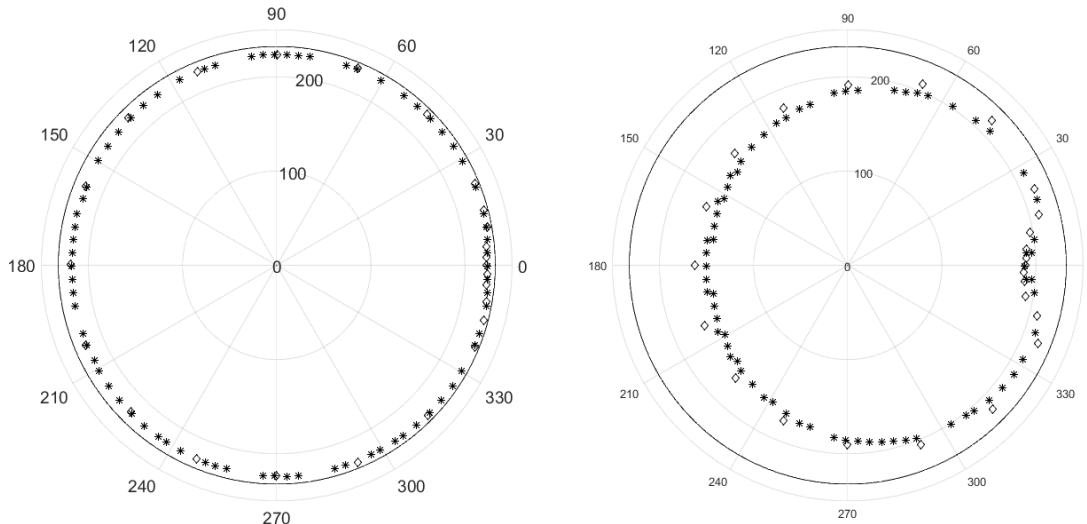
Figure 11. Run-up for case A (left) and case C (right), measured data ($\Diamond$) and computed solution ($*$) for the conical island example.

4.2.3 Monai Valley experiment

The Monai Valley experiment [32] is a model of the 1993 Okushiri tsunami, which caused an extreme 32 meter run-up in the Monai Valley on Okushiri island. The purpose of this experiment is to recreate the run-up in a 1/400 model of the relevant part of Okushiri island, see also [30, Chapter 11] for further details.

We discretize the domain $\Omega = [0, 5.488] \times [0, 3.402]$ with a regular mesh with step sizes $\Delta x = \Delta y = 0.014$. Reflective boundary conditions are employed everywhere except at $x = 0$, where the incident wave is prescribed up to $t = 22.5$. For $t > 22.5$ we use open boundary conditions at $x = 0$. Water levels are recorded at the three points $(4.521, 1.196)$, $(4.521, 1.696)$ and $(4.521, 2.196)$, which correspond to gauges 1, 2 and 3 of the experimental setup at which points measurements of the water height are provided. We integrate the shallow-water equations until $t = 25$, which is long enough to record the maximum run-up which occurs at approximately $t = 20$ at the reference locations. The time step in the simulation was set to $\Delta t = 0.01$. All experimental data as well as the incident wave profile were obtained from [1].

The evolution is illustrated by four snapshots in Figure 12. The recorded water heights at the gauges is in good accordance with the experimentally recorded values, see Figure 13.

5 Conclusions

In this paper, we have proposed a novel numerical procedure for solving the shallow-water equations, which is suitable for tsunami modeling. In particular, both the well-balanced numerical scheme and the inundation algorithm are based on radial basis functions. This makes the model truly meshless and, hence, capable of employing variable resolution as well as operating on arbitrary coastal regions, without the need to use an underlying orthogonal mesh. First benchmark tests demonstrate the competitiveness of the proposed methodology both in the one- and two-dimensional setting.

A main defining characteristic of tsunamis is that they occur on a multitude of spatial and temporal scales, which can pose considerable challenges to numerical solvers for the shallow-water equations. An advantage of the RBF methodology is that is can be easily adopted to both arbitrary geometries (e.g. the plane and the sphere) and arbitrary nodal layouts, both features
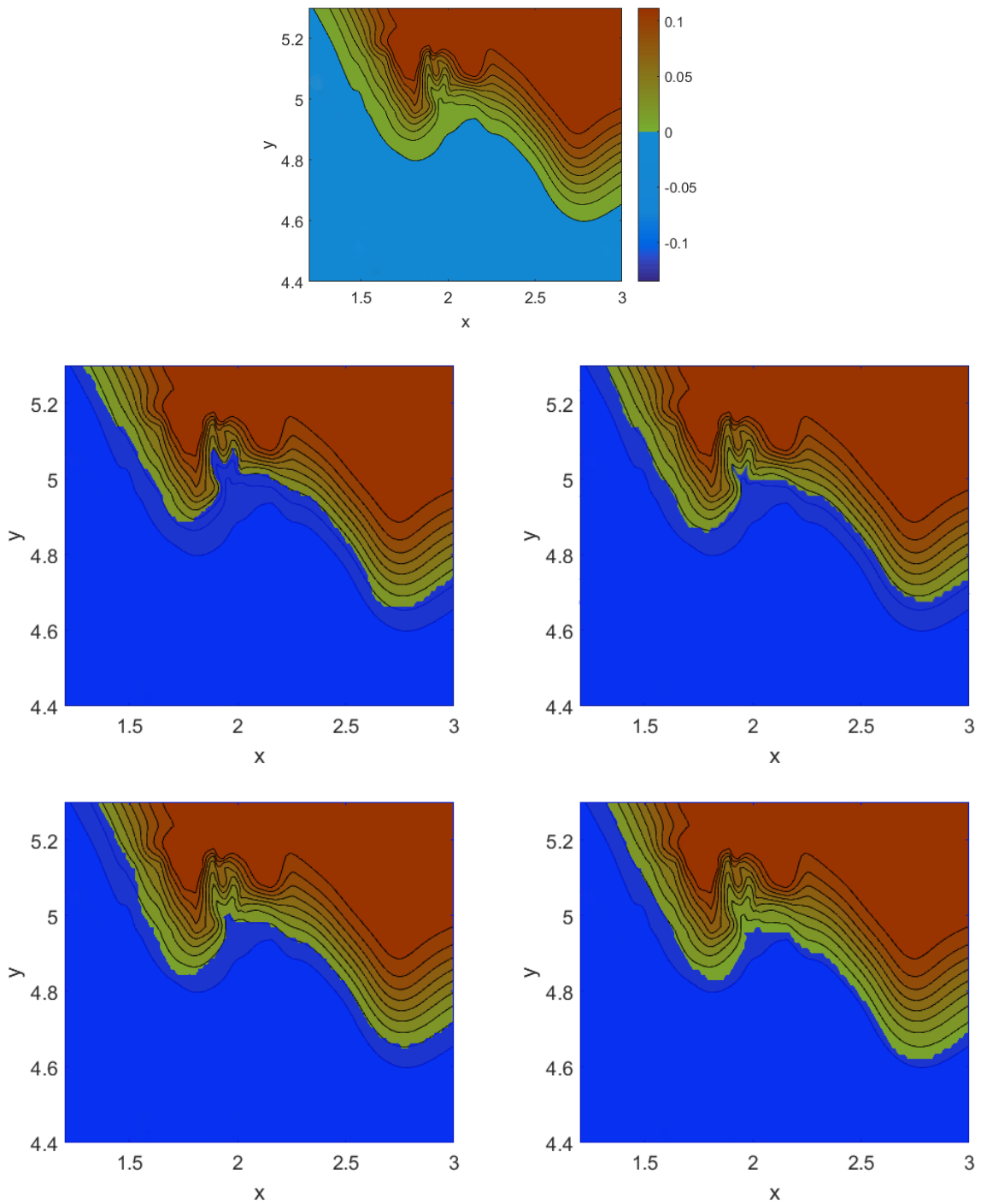
Figure 12. Image at the top shows the initial water elevation and the height of the valley. Snapshots of the computed solution $(h + b)$ for the Monai Valley test case at times $t = 16.2$ (top left) (time of maximal run-up), $t = 17$ (top right), $t = 18$ (lower left) and $t = 19$ (lower right).
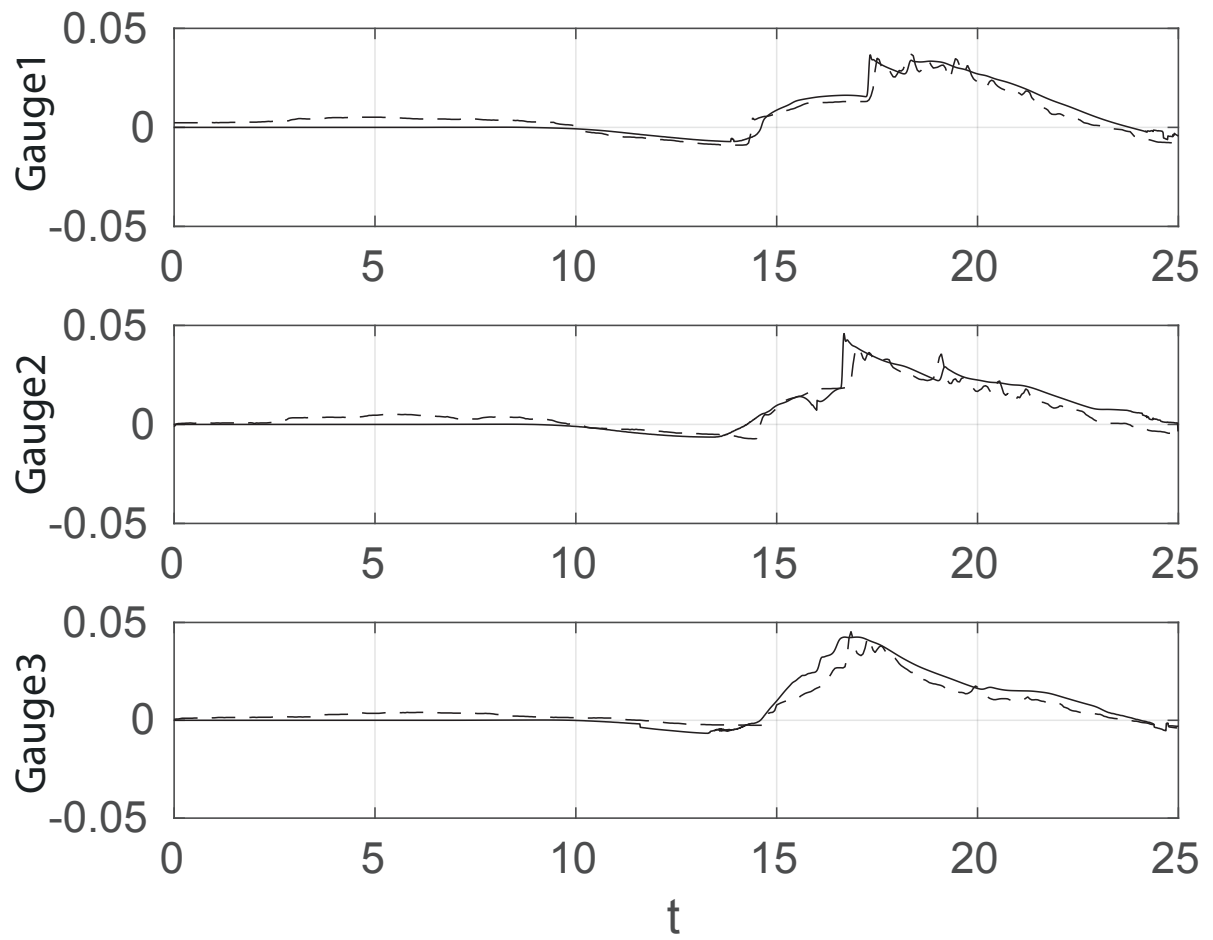
16

Figure 13. Gauges 1,2,3 for the Monai Valley test case.

important for the far- and near-field modeling of tsunamis. While, in the present work, we have concentrated on the near coast propagation of tsunamis as well as coastal inundation, in future work we will combine the methodology proposed here with a global scale tsunami propagation model. Suitable candidates for numerical models have already been proposed in [10, 11], which we will adopt to be able to handle arbitrary sea bottom topographies.

With the exception of the two-dimensional lake at rest solutions, all other benchmark tests were carried out on a regular, orthogonal mesh. This was done in order to facilitate comparison with other numerical models for the same benchmark problems that are usually done on an orthogonal mesh as well. A full demonstration of the meshless capabilities of the proposed shallow-water discretization, as well as more complicated real-world tsunami simulations, is a subject for future work.

One potential problem of the RBF-FD methodology as used in the present paper is that the resulting schemes are not exactly mass conserving. While we have numerically verified that the error in mass conservation is purely oscillatory (and showing no spurious growth or decay), in order for the meshless methodology to be competitive with standard finite volume or discontinuous Galerkin methods (which both typically preserve mass) it will be essential to develop a meshless methodology that is conservative when applied to hyperbolic conservation laws. We will consider this issue in future investigations.

Acknowledgments

References

[1] Tsunami Runup onto a Complex Three-dimensional Beach; Monai Valley, http://nctr.pmel.noaa.gov/benchmark/Laboratory/Laboratory_MonaiValley/, accessed: 2017-02-18.

[2] Audusse E., Bouchut F., Bristeau M.O., Klein R. and Perthame B., A fast and stable well-balanced scheme with hydrostatic reconstruction for shallow water flows, *SIAM J. Sci. Comput.* **25** (2004), 2050–2065.

[3] Bates P.D. and Hervouet J.M., A new method for moving–boundary hydrodynamic problems in shallow water, *Proc. Royal Soc. London Ser. A: Math. Phys. Eng. Sci.* **455** (1999), 3107–3128.

[4] Bayona V., Moscoso M., Carretero M. and Kindelan M., RBF-FD formulas and convergence properties, *J. Comput. Phys.* **229** (2010), 8281–8295.

[5] Bihlo A. and MacLachlan S., Well-balanced mesh-based and meshless schemes for the shallow-water equations, arXiv:1702.07749, 2017.

[6] Briggs M.J., Synolakis C.E., Harkins G.S. and Green D.R., Laboratory experiments of tsunami runup on a circular island, in *Tsunamis: 1992–1994*, Springer, 1995, pp. 569–593.

[7] Bunya S., Kubatko E.J., Westerink J.J. and Dawson C., A wetting and drying treatment for the Runge–Kutta discontinuous Galerkin solution to the shallow water equations, *Comput. Methods Appl. Mech. Eng.* **198** (2009), 1548–1562.

[8] Carrier G.F. and Greenspan H.P., Water waves of finite amplitude on a sloping beach, *J. Fluid Mech.* **4** (1958), 97–109.

[9] De Leffe M., Le Touzé D. and Alessandrini B., SPH modeling of shallow-water coastal flows, *J. Hydraul. Res.* **48** (2010), 118–125.

[10] Flyer N., Lehto E., Blaise S., Wright G.B. and St-Cyr A., A guide to RBF-generated finite differences for nonlinear transport: Shallow water simulations on a sphere, *J. Comput. Phys.* **231** (2012), 4078–4095.

[11] Flyer N. and Wright G.B., A radial basis function method for the shallow water equations on a sphere, *Proc. R. Soc. A* **465** (2009), 1949–1976.

[12] Fornberg B. and Flyer N., *A Primer on Radial Basis Functions with Applications to the Geosciences*, vol. 3529, SIAM Press, Philadelphia, PA, 2015.

[13] Fornberg B. and Flyer N., Solving PDEs with radial basis functions, *Acta Numer.* **24** (2015), 215–258.

[14] Fornberg B. and Lehto E., Stabilization of RBF-generated finite difference methods for convective PDEs, *J. Comput. Phys.* **230** (2011), 2270–2285.

[15] Fornberg B., Lehto E. and Powell C., Stable calculation of Gaussian-based RBF-FD stencils, *Comput. Math. Appl.* **65** (2013), 627–637.

[16] Fornberg B., Wright G. and Larsson E., Some observations regarding interpolants in the limit of flat radial basis functions, *Comput. Math. Appl.* **47** (2004), 37–55.

[17] Fornberg B. and Zuev J., The Runge phenomenon and spatially variable shape parameters in RBF interpolation, *Comput. Math. Appl.* **54** (2007), 379–398.

[18] Franke R., Scattered data interpolation: tests of some methods, *Math. Comput.* **38** (1982), 181–200.

[19] Fuhrman D.R. and Madsen P.A., Simulation of nonlinear wave run-up with a high-order Boussinesq model, *Coast. Eng.* **55** (2008), 139–154.

[20] Gallouët T., Hérard J.M. and Seguin N., Some approximate Godunov schemes to compute shallow-water equations with topography, *Comput. & Fluids* **32** (2003), 479–513.

[21] Goto C., Ogawa Y., Shuto N. and Imamura F., *IUGG/IOC time project: Numerical method of tsunami simulation with the leap-frog scheme*, *Tech. rep.*, IOC Manuals and Guides, UNESCO, Paris, 1997.

[22] Hardy R.L., Multiquadric equations of topography and other irregular surfaces, *J. Geophys. Res.* **76** (1971), 1905–1915.

[23] Harig S., Pranowo W.S. and Behrens J., Tsunami simulations on several scales, *Ocean Dyn.* **58** (2008), 429–440.

[24] Hon Y.C., Cheung K.F., Mao X.Z. and Kansa E.J., Multiquadric solution for shallow water equations, *J. Hydraul. Eng.* **125** (1999), 524–533.

[25] Kaiser G., Scheele L., Kortenhaus A., Løvholt F., Römer H. and Leschka S., The influence of land cover roughness on the results of high resolution tsunami inundation modeling, *Nat. Hazards Earth Syst. Sci.* **11** (2011), 2521–2540.

[26] Kurganov A. and Petrova G., A second-order well-balanced positivity preserving central-upwind scheme for the Saint-Venant system, *Commun. Math. Sci.* **5** (2007), 133–160.

[27] LeVeque R.J., Balancing source terms and flux gradients in high-resolution Godunov methods: the quasi-steady wave-propagation algorithm, *J. Comput. Phys.* **146** (1998), 346–365.

[28] LeVeque R.J., George D.L. and Berger M.J., Tsunami modelling with adaptively refined finite volume methods, *Acta Numer.* **20** (2011), 211–289.

[29] Liu P.L.F., Cho Y.S., Briggs M.J., Kanoglu U. and Synolakis C.E., Runup of solitary waves on a circular island, *J. Fluid Mech.* **302** (1995), 259–285.

[30] Liu P.L.F., Yeh H. and Synolakis C., *Advanced Numerical Models for Simulating Tsunami Waves and Runup*, vol. 10 of *Advances in Coastal and Ocean Engineering*, World Scientific, Singapore, 2008.

[31] Lynett P.J., Wu T.R. and Liu P.L.F., Modeling wave runup with depth-integrated equations, *Coast. Eng.* **46** (2002), 89–107.

[32] Matsuyama M. and Tanaka H., An experimental study of the highest run-up height in the 1993 Hokkaido Nansei-oki earthquake tsunami, in *Proceedings of the International Tsunami Symposium (7–10 August 2001, Seattle, WA)*, 2001, pp. 879–889.

[33] Pedlosky J., *Geophysical fluid dynamics*, Springer, New York, 1987.

[34] Shankar V., Wright G.B., Kirby R.M. and Fogelson A.L., A radial basis function (RBF)-finite difference (FD) method for diffusion and reaction–diffusion equations on surfaces, *J. Sci. Comput.* **63** (2015), 745–768.

[35] Sielecki A. and Wurtele M.G., The numerical integration of the nonlinear shallow-water equations with sloping boundaries, *J. Comput. Phys.* **6** (1970), 219–236.

[36] Thacker W.C., Some exact solutions to the nonlinear shallow-water wave equations, *J. Fluid Mech.* **107** (1981), 499–508.

[37] Titov V.V. and Gonzalez F.I., *Implementation and testing of the method of splitting tsunami (MOST) model*, *Tech. rep.*, NOAA Technical Memorandum ERL PMEL-112, 1997.

[38] Titov V.V. and Synolakis C.E., Modeling of breaking and nonbreaking long-wave evolution and runup using VTCS-2, *J. Waterway, Port, Coastal, Ocean Eng.* **121** (1995), 308–316.

[39] Titov V.V. and Synolakis C.E., Numerical modeling of tidal wave runup, *J. Waterway, Port, Coastal, Ocean Eng.* **124** (1998), 157–171.

[40] Tolstykh A.I., On using RBF-based differencing formulas for unstructured and mixed structured-unstructured grid calculations, in *Proc. 16th IMACS World Congress*, vol. 228, vol. 228, 2000, pp. 4606–4624.

[41] Vater S., Beisiegel N. and Behrens J., A limiter-based well-balanced discontinuous Galerkin method for shallow-water flows with wetting and drying: One-dimensional case, *Adv. Water Resour.* **85** (2015), 1–13.

[42] Wong S.M., Hon Y.C. and Golberg M.A., Compactly supported radial basis functions for shallow water equations, *Appl. Math. Comput.* **127** (2002), 79–101.

[43] Xia X., Liang Q., Pastor M., Zou W. and Zhuang Y.F., Balancing the source terms in a SPH model for solving the shallow water equations, *Adv. Water Resour.* **59** (2013), 25–38.

[44] Zhou X., Hon Y.C. and Cheung K.F., A grid-free, nonlinear shallow-water model with moving boundary, *Eng. Anal. Bound. Elem.* **28** (2004), 967–973.