



HAL
open science

WETTING AND DRYING TREATMENTS WITH MESH ADAPTATION FOR SHALLOW WATER EQUATIONS USING A RUNGE-KUTTA DISCONTINUOUS GALERKIN METHOD

Camille Poussel, Mehmet Ersoy, Frederic Golay

► **To cite this version:**

Camille Poussel, Mehmet Ersoy, Frederic Golay. WETTING AND DRYING TREATMENTS WITH MESH ADAPTATION FOR SHALLOW WATER EQUATIONS USING A RUNGE-KUTTA DISCONTINUOUS GALERKIN METHOD. 2024. hal-04676186

HAL Id: hal-04676186

<https://hal.science/hal-04676186>

Preprint submitted on 23 Aug 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

WETTING AND DRYING TREATMENTS WITH MESH ADAPTATION FOR SHALLOW WATER EQUATIONS USING A RUNGE-KUTTA DISCONTINUOUS GALERKIN METHOD

CAMILLE POUSSEL, MEHMET ERSOY, AND FRÉDÉRIC GOLAY

ABSTRACT. This work is devoted to the numerical simulation of Shallow Water Equations involving dry areas, a moving shoreline and in the context of mesh adaptation. The space and time discretization using the Runge-Kutta Discontinuous Galerkin approach is applied to nonlinear hyperbolic Shallow Water Equations. Problems with dry areas are challenging for such methods. To counter this issue, special treatment is applied around the shoreline. This work compares three treatments, one based on Slope Modification, one based on p-adaptation and the last one based on eXtended Finite Element methods and mesh adaptation.

1. INTRODUCTION

This work is developed in the framework of interactions between the flow of water in sandy beaches and the free surface flow above the sand. Simulating the flow of groundwater has been done by Clement in 2021 [11] using the adaptive Discontinuous Galerkin method to solve Richards' Equation (SWE). The present work aims to develop the hyperbolic part to couple it with the parabolic one. The free surface flow over sandy beaches is modeled using the Shallow Water Equations (SWE). They are derived by considering the depth-averaged three-dimensional incompressible Navier-Stokes Equations, assuming hydrostatic pressure distribution and neglecting vertical acceleration and viscous effects [16, 37].

Discontinuous Galerkin (DG) methods were introduced in 1989 by Cockburn [12] in the scope of conservative laws. They combine the background of Finite Element (FE) methods and Finite Volume (FV) methods since the solution is sought in a broken Sobolev space and is approximated with discontinuous polynomials. Moreover, in the context of hyperbolic problems, numerical fluxes are approximated, considering the problem's physics. Due to the discontinuous approximation, the DG methods are well adapted to non-conformal meshing. As in FV methods, increasing the DG space approximation order introduces spurious oscillations of the numerical solutions [12, 42]. Slope limiting [12] and moment limiting [34] can be used to counter this problem. More recently, such methods were broadly used to solve SWE [20, 18]. DG methods can not natively treat flooding and drying problems due to the loss of hyperbolicity on the shoreline. Consequently, a post-processing is needed to preserve the positivity of the water height [58]. This work is devoted to analyzing three different drying treatments for one-dimensional problems and shows the ability of a selected one for two-dimensional problems. The first drying method is the most common one used for SWE with dry areas and was introduced by Ern in 2007 [20] and improved after that. It consists in modifying the slope of the water height in order to remain positive. The second presented method is inspired by work on hp-adaptation [52, 22]. It consists in doing FV methods around the shoreline. The last

Key words and phrases. Discontinuous Galerkin, Shallow Water Equations, Wetting and drying, Positivity preserving, mesh adaptation.

Abbreviations: SWE, Shallow Water Equations; DG, Discontinuous Galerkin; FE, Finite Element; X-FE, eXtended Finite Element; FV, Finite Volume; AMR, Adaptive Mesh Refinement; RK, Runge-Kutta; SSP-RK, Strong Stability Preserving Runge-Kutta; RKDG, Runge-Kutta Discontinuous Galerkin; TVD, Total Variations Diminishing; TVB, Total Variations Bounded; CFD, Computational Fluid Dynamics; CFL, Courant-Friedrichs-Lewy

method is greatly inspired by the work on eXtended Finite Element (X-FE) methods[39, 32] and mesh adaptation. It consists in virtually modifying the mesh to fit it on the shoreline. This paper is organized as follows. In Section 2, we recall the expression of SWE in two dimensions with a bathymetry source term. We detail the space discretization in Section 3. In Section 4, the fully discretized DG formulation is given with modification to ensure the well-balanced property and limit spurious oscillations. In Section 5, positivity-preserving procedures are introduced to treat the moving shoreline. In Section 6, three one-dimensional test cases with a moving shoreline are solved, and two two-dimensional problems with flooding and drying are solved.

2. GOVERNING EQUATIONS

The free surface flow over sandy beaches is modeled using SWE. They are derived by considering the depth-averaged three-dimensional incompressible Navier-Stokes Equations, assuming hydrostatic pressure distribution and neglecting vertical acceleration. The reader may refer to the original derivation in 1871 [16] and can also consider reading the derivation of viscous SWE in [25, 37]. These two ways of derivation give the same system by neglecting viscous effects by the end. This section recalls the expression of SWE with two different sets of variables and studies this system's hyperbolicity of non-linear equations.

Let us consider a finite domain Ω , a subset of \mathbb{R}^2 , and consider $T > 0$ as the simulation time. The gravitational acceleration is denoted by g , and $z_b : \Omega \rightarrow \mathbb{R}$ is a smooth function representing the bathymetry. The SWE can be written in its conservative way as follows:

$$(1) \quad \begin{cases} \partial_t h + \operatorname{div}(h\mathbf{u}) = 0, \\ \partial_t(h\mathbf{u}) + \operatorname{div}(h\mathbf{u} \otimes \mathbf{u} + g\frac{h^2}{2}\mathbb{I}) = -gh\nabla z_b, \\ \text{Initial and Boundary conditions,} \end{cases}$$

where the unknown are $h : \Omega \times [0, T[\rightarrow \mathbb{R}$ water height and $\mathbf{u} = (u, v)^T : \Omega \times [0, T[\rightarrow \mathbb{R}^2$ the horizontal velocity. \mathbb{I} denotes the identity matrix.

The three equations in System (1) express mass and momentum conservation laws. They are driven by the fluxes

$$\mathbf{G}_1(\mathbf{U}) = \begin{pmatrix} q_x \\ \frac{q_x^2}{h} + g\frac{h^2}{2} \\ \frac{q_x q_y}{h} \end{pmatrix} \text{ and } \mathbf{G}_2(\mathbf{U}) = \begin{pmatrix} q_y \\ \frac{q_x q_y}{h} \\ \frac{q_y^2}{h} + g\frac{h^2}{2} \end{pmatrix}$$

and forced through the source

$$\mathbf{S}(\mathbf{U}, z_b) = \begin{pmatrix} 0 \\ -gh\partial_x z_b \\ -gh\partial_y z_b \end{pmatrix}$$

where $\mathbf{U} := (h, \mathbf{q})^T : \Omega \times [0, T[\rightarrow \mathbb{R}^3$ are the conservative variables with $\mathbf{q} = h\mathbf{u} = (q_x, q_y)^T$ the horizontal discharge.

The System (1) can be rewritten in its vectorial form :

$$(2) \quad \begin{cases} \partial_t \mathbf{U} + \operatorname{div} \mathbb{G}(\mathbf{U}) = \mathbf{S}(\mathbf{U}, z_b) \text{ in } \Omega \times]0, T[, \\ \text{Initial and Boundary conditions,} \end{cases}$$

where $\mathbb{G}(\mathbf{U}) = (\mathbf{G}_1(\mathbf{U}), \mathbf{G}_2(\mathbf{U}))$. In addition, to have a correct transition between the conservative form System (2) and its non-conservative form

$$\mathbb{J}_1(\mathbf{U}) := \frac{\partial \mathbf{G}_1(\mathbf{U})}{\partial \mathbf{U}} \text{ and } \mathbb{J}_2(\mathbf{U}) := \frac{\partial \mathbf{G}_2(\mathbf{U})}{\partial \mathbf{U}}$$

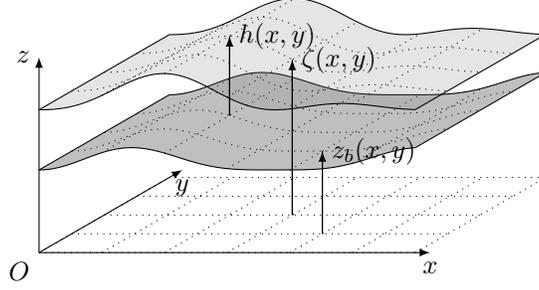


FIGURE 1. Sketch of the water height (h), bathymetry (z_b) and free-surface elevation (ζ) in the context of SWE.

the Jacobian matrix of the fluxes are defined. Using the predefined variables h and z_b , the free-surface elevation ζ can be defined for all $\mathbf{x} := (x, y)$ in Ω and t in $]0, T[$, $\zeta(\mathbf{x}, t) = h(\mathbf{x}, t) + z_b(\mathbf{x})$. Figure 1 shows a sketch of the water height (h), bathymetry (z_b), and free-surface elevation (ζ) in the context of SWE.

Understanding the hyperbolicity of System (2) is crucial as it provides valuable insights into this conservation law system's flow characteristics and numerical solutions. This knowledge is beneficial when employing FV and DG methods. Additionally, the hyperbolicity of the system informs how to handle boundary conditions effectively.

Only a condition on the water height arose by studying the hyperbolicity of SWE. This condition will be necessary when dealing with dry areas if SWE is considered with emerging bathymetry. A thorough study of SWE's hyperbolicity in two dimensions can be found in [56].

We recall here that System (2) is strictly hyperbolic if the eigenvalues of

$$(3) \quad \mathbb{A}(\mathbf{U}) := \alpha \mathbb{J}_1(\mathbf{U}) + \beta \mathbb{J}_2(\mathbf{U})$$

are real and distinct, with $\alpha, \beta \in \mathbb{R}$ such that $\alpha^2 + \beta^2 = 1$, (see [26] for further information about hyperbolicity of systems of conservation laws). Moreover α and β can also be denoted by $\cos(\theta)$ and $\sin(\theta)$ with $\theta \in \mathbb{R}$, so $n_\theta = (\cos(\theta), \sin(\theta))^T$ represents a certain direction in the $x - y$ plane. This view will be used later when approximating the fluxes at the cell interfaces.

3. SPACE DISCRETIZATION

Since the objective is to solve System (2) using a numerical scheme, defining a partition of the computational domain Ω is mandatory. A thorough description of this mesh is described in the following, and the way that the mesh can be adapted and modified during the simulation is also presented.

3.1. Mesh description. The time duration $[0, T]$ is discretized in sub interval $T^n := [t^n, t^{n+1}]$ for all integer n , thus the corresponding time step is $\Delta t^n := t^{n+1} - t^n$. Let us define \mathcal{E}^n a partition of the computational domain Ω valid for all $t \in T^n$. For the sake of simplicity, it is assumed that Ω is a polygonal domain in two space dimensions so that \mathcal{E}^n covers Ω exactly. The mesh \mathcal{E}^n is composed of quadrilateral and triangular elements not necessarily conformal.

For all elements $E \in \mathcal{E}^n$, d_E is its diameter defined as the ratio between its surface (\mathfrak{s}_E) and perimeter (\mathfrak{p}_E) and $d^n := \max_{E \in \mathcal{E}^n} (d_E)$.

The set of all open faces of all elements $E \in \mathcal{E}^n$ is denoted by \mathcal{F} . Moreover, we can define two subsets of \mathcal{F} , \mathcal{F}^∂ for the boundary faces and \mathcal{F}^{in} for the interior faces:

$$\mathcal{F}^\partial := \bigcup_{F \in \partial\Omega} F \quad \text{and} \quad \mathcal{F}^{\text{in}} := \mathcal{F} \setminus \mathcal{F}^\partial.$$

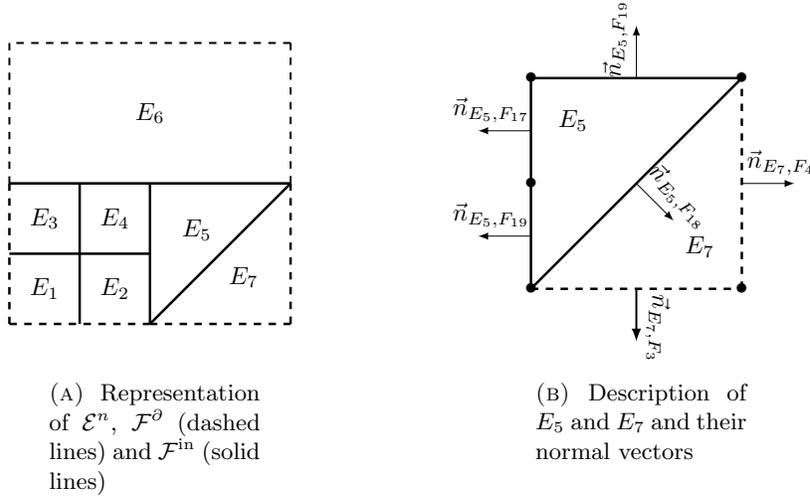


FIGURE 2. Example of a mesh

For a given element $E \in \mathcal{E}^n$, there exists a set of face $\mathcal{F}^E := \{F \in \mathcal{F} | F \in \partial E\}$ which defines boundaries of E . Then for all interior faces of E , i.e. $\forall F \in \mathcal{F}^E \cap \mathcal{F}^{\text{in}}$, there exists a neighboring element E_r such that $E \cap E_r = F$. Consequently the normal unit vector $\vec{n}_{E,F} := (n_x, n_y)^T$ pointing from E to E_r can be defined. Moreover for all boundary faces of E , i.e. $\forall F \in \mathcal{F}^E \cap \mathcal{F}^\partial$, there exists E_∂ a fictitious element such that $E \cap E_\partial = F$. Consequently, the normal unit vector $\vec{n}_{E,F}$ pointing always from E to E_∂ can be defined.

Figure 2a gives a graphical representation for an example mesh composed of triangles and quadrilaterals. In this example the mesh is composed of 7 elements, i.e. $\mathcal{E}^n = \{E_i, i \in 1, \dots, 7\}$. Thus the set of faces $\mathcal{F} = \{F_i, i \in 1, \dots, 19\}$ is defined. It can be split into two subsets, the first one $\mathcal{F}^\partial = \{F_i, i \in 1, \dots, 9\}$ boundary faces of \mathcal{F} , depicted with dashed lines on Figure 2. The second one $\mathcal{F}^{\text{in}} = \{F_i, i \in 10, \dots, 19\}$ interior faces of \mathcal{F} . Figure 2b gives graphical representation for two elements E_5 and E_7 . Faces are also depicted with their normal vectors.

It was necessary to define a mesh \mathcal{E}^n at a specific time level n because the mesh is modified during the simulation. This is called adaptive mesh refinement and is described in the following section.

3.2. Adaptive Mesh Refinement. Adaptive Mesh Refinement (AMR) methods are currently in extensive use and have demonstrated their effectiveness on various types of meshes, whether they are 2D or 3D, structured or unstructured, conforming or non-conforming, and whether domain decomposition is employed or not, see e.g. [15].

This study focuses on applications involving intricate geometries and, potentially, the coupling of models using hybrid meshes. Consequently, a non-conforming unstructured mesh is used, similar to prior research [1]. Despite the potential increase in computational time compared to fully structured Cartesian codes, we have adopted a Block-Based AMR strategy. This chapter exclusively presents the AMR method in two dimensions but can be extended to the three-dimensional by considering hexahedron and tetrahedron [28]. The approach adopted is illustrated in the Figure 3.

The unstructured mesh composed of quadrilaterals and triangular elements, where each one, as in [1], defines a root element also named a block. See Figure 3a where every element is a non-refined block illustrated with a thick contour. Then, a mesh refinement level is defined for each block, initially by the user, and after that, according to the refinement criterion, see Figure 3b.

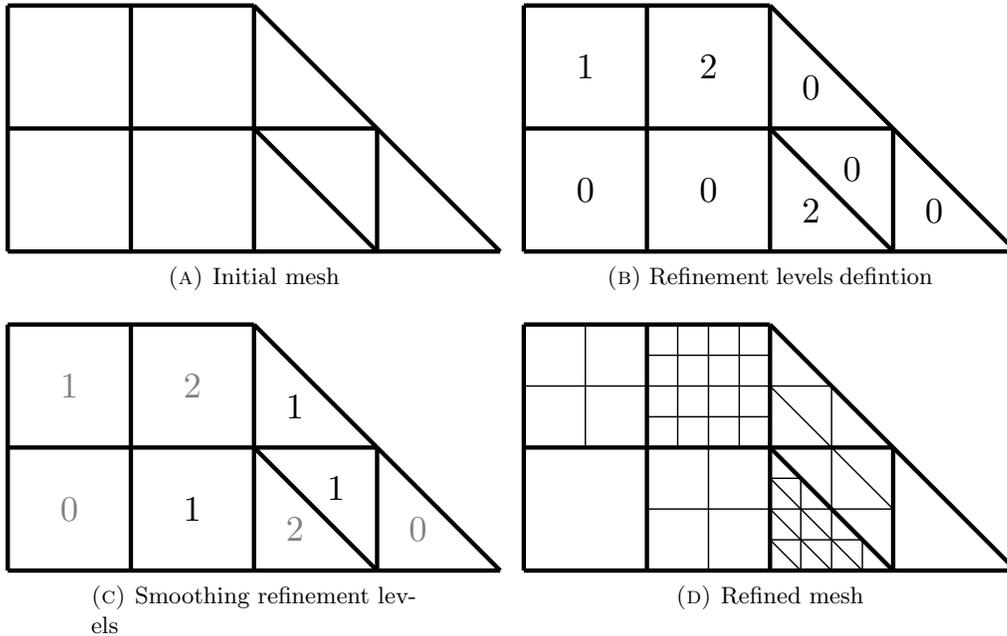


FIGURE 3. Adaptive Mesh Refinement strategy

Coarsening and refinement are decided according to threshold values $0 < \beta_c < \beta_r$. If β denotes the local criterion computed on the block, three situations may occur:

- For $\beta_c < \beta < \beta_r$, the refinement level of the block remains unchanged;
- For $\beta_r < \beta$, the refinement level of the block is increased, and the block is then refined;
- For $\beta < \beta_c$, the refinement level of the block is decreased, and the block is then coarsened.

The limitations associated with this choice are discussed in detail in [45], along with the recommended steps to address these limitations. The mesh refinement levels are also adapted according to the rule that the difference of mesh refinement levels between two adjacent blocks is at most equal to 1. This rule is illustrated in Figure 3c. As shown in [21], this constraint allows smooth transitions between refined and unrefined regions. The mesh is then built using a quad-tree graph, applicable to both quadrangles and triangles. It establishes clear and accurate connections between neighboring elements for each face. This process continues until the desired level of mesh refinement, as depicted in Figure 3d, is achieved. In dynamic or evolving problems, this necessitates frequent re-meshing to track the phenomena under investigation. To minimize re-meshing frequency, the stencil is expanded by considering a coarser root mesh, namely blocks. The local criterion β relies on an error indicator, but determining the error of the numerical solution on each element can be challenging. To address this issue, a posteriori error indicators are used. However, for complex problems, obtaining these indicators can still be challenging. As a result, an intuitive criterion that is independent of the problem is often employed. This criterion involves identifying areas where the numerical approximation may fail, such as the free surface elevation gradient or numerical entropy production in the case of SWE. The method for using this criterion is not discussed in this work, but interested readers can refer to [28] for FV methods and [11] for DG methods.

4. DISCONTINUOUS GALERKIN FORMULATION

In this section, before presenting the weak formulation, we first define the solution space, referred to as the DG space. Following this, we provide the semi-discrete weak formulation and then the

presentation of the weak vectorial formulation well-suited for computation. Finally, ideas on how to handle steady-state solutions and cancel spurious oscillations are given.

DG methods approximate the solution within a FE framework. These methods use trial and test spaces defined by piecewise polynomial functions without explicitly enforcing continuity between neighboring mesh elements. So, the first set to be defined is the set of polynomial functions of degree $p \in \mathbb{N}$ over a mesh element E , $\mathbb{P}^p(E)$. Then, for all $n \in \mathbb{N}$, the set of piecewise polynomials functions on the mesh \mathcal{E}^n is defined as

$$(4) \quad \mathcal{V}^p(\mathcal{E}^n) := \{v : \Omega \rightarrow \mathbb{R} \mid v|_E \in \mathbb{P}^p(E), \forall E \in \mathcal{E}^n\}.$$

\mathcal{V}^p is the set where the solution of System (2) is sought. It is also called the DG space. For more detailed and general definitions of the sets, see [44, chap. 1].

4.1. Semi-discrete Discontinuous Galerkin formulation. By multiplying System (2) by a test function $\varphi \in [\mathcal{V}^p(\mathcal{E}^n)]^3$, then by integration over Ω and finally using the divergence theorem, it gives the semi-discrete weak formulation :

$$(5) \quad \text{For all } n \in \mathbb{N}, \text{ find } \mathbf{U} := (h, q_x, q_y)^T \in [\mathcal{V}^p(\mathcal{E}^n)]^3 \text{ such that } \forall t \in T^n, \forall E \in \mathcal{E}^n \text{ and } \forall \varphi \in [\mathcal{V}^p(\mathcal{E}^n)]^3,$$

$$\left\{ \begin{array}{l} \sum_{E \in \mathcal{E}^n} \int_E \varphi(\mathbf{x}) \partial_t \mathbf{U}(\mathbf{x}, t) dE - \sum_{E \in \mathcal{E}^n} \int_E \partial_{\mathbf{x}_i} \varphi(\mathbf{x}) \mathbf{G}_i(\mathbf{U}(\mathbf{x}, t)) dE + \sum_{E \in \mathcal{E}^n} \sum_{F \in \mathcal{F}^E} \int_F \varphi(\mathbf{x}) \mathbf{G}_F^*(\mathbf{U}(\mathbf{x}, t)) dF \\ \qquad \qquad \qquad = \sum_{E \in \mathcal{E}^n} \int_E \varphi(\mathbf{x}) \mathbf{S}(\mathbf{U}(\mathbf{x}, t), \tilde{z}_b(\mathbf{x})) dE \\ \sum_{E \in \mathcal{E}^n} \int_E \varphi(\mathbf{x}) \mathbf{U}(\mathbf{x}, 0) dE = \sum_{E \in \mathcal{E}^n} \int_E \varphi(\mathbf{x}) U_0(\mathbf{x}) dE \end{array} \right.$$

where \tilde{z}_b is the projection of the bathymetry onto \mathcal{V}^p and \mathbf{G}_F^* is the numerical flux across any face F and U_0 is the initial data. Operations between two vectors in System (5) are the element-wise product known as the Hadamard product. In System (5), if we look for the solution in the set of piecewise constant functions, the volume contribution of the flux is null, and the semi-discrete weak formulation becomes equivalent to the FV formulation. This previous volume contribution is specific to DG methods while p is greater than 1 the polynomial degree.

The numerical flux is defined as follows: $\forall F \in \mathcal{F}, \forall \mathbf{x} \in F$,

$$\mathbf{G}_F^*(\mathbf{U})(\mathbf{x}) = \begin{cases} \tilde{\mathbf{G}} \left(\mathbf{U}|_E(\mathbf{x}), \mathbf{U}|_{E_r}(\mathbf{x}), \vec{n}_{E,F} \right), & \text{if } F \in \mathcal{F}^{\text{in}} \\ \tilde{\mathbf{G}} \left(\mathbf{U}|_E(\mathbf{x}), \mathbf{U}|_{E_\partial}(\mathbf{x}), \vec{n}_{E,F} \right), & \text{if } F \in \mathcal{F}^\partial \end{cases}$$

where $\tilde{\mathbf{G}} : \mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}^2 \rightarrow \mathbb{R}^3$ is the numerical flux approximation. $\mathbf{U}|_{E_\partial}$ is used to enforce boundary conditions weakly through the numerical fluxes. Moreover it is easy to check that $\tilde{\mathbf{G}}$ is *conservative* and *consistent*. Since the solution is not defined on a face F , it becomes necessary to recover its value. In 1959, Godunov introduced the first approach to deal with this problem [27]. The method involves solving the interface's associated Riemann problem and using the computed intermediate state in the flux on the interface. The Riemann problem can be solved using an exact solver, which comes at the cost of an iterative root finder [24]. Moreover, there are numerous methods to approximate the intermediate state of the Riemann problem. The Lax-Friedrichs flux and its variant, the local Lax-Friedrichs flux, also known as the Rusanov flux [24], are simple methods. Harten, Lax, Van Leer introduced HLL approximate Riemann solver in 1983 [31]. For further details about approximate Riemann solvers, refer to [17].

This work uses the local Lax-Friedrich approximate solver to evaluate $\tilde{\mathbf{G}}$ at every $F \in \mathcal{F}$. Its selection is based on its suitability for multi-dimensional hyperbolic problems, where it is easy to implement and yields good results for DG methods. The local Lax-Friedrichs flux, then $\tilde{\mathbf{G}}$, is defined as follows:

$$\begin{aligned} \forall a, b \in \mathbb{R}^3 \text{ and } \forall \vec{n} = (n_x, n_y)^T \in \mathbb{R}^2, \\ \tilde{\mathbf{G}}(a, b, \vec{n}) = \frac{1}{2} \left(\mathbb{G}(a) + \mathbb{G}(b) \right) \cdot \vec{n} - \frac{c}{2} (b - a). \end{aligned}$$

where c is the maximum of the characteristic speeds through the face F :

$$c = \max_{w \in \{a, b\}} \{ |\text{spec}(\mathbb{J}_1(w)n_x + \mathbb{J}_2(w)n_y)| \}.$$

Here spec denotes the spectrum, i.e. the set of eigenvalues. It is easy to check that $\tilde{\mathbf{G}}$ is conservative and consistent.

4.2. Fully discrete Discontinuous Galerkin formulation. Using two-dimensional polynomial basis functions, such as tensor product Legendre basis [34] for quadrangle or the Dubiner basis [19] for triangles, the semi-discrete weak formulation can be written in a vectorial form. The solution on each element can be written as a dot product between two vectors. One holds degrees of freedom of the solution on E (\mathbf{U}_E), and the other holds the basis functions (Φ). Thus for any element $E \in \mathcal{E}$ and for all $t \in [0, T]$, it can be written as follows:

$$\forall \mathbf{x} \in E, \mathbf{U}(\mathbf{x}, t) = \Phi(\mathbf{x}) \cdot \mathbf{U}_E(t).$$

Since the solution is sought in the DG space, it gives a matrix-vector system for each element. Consequently, to get the solution on Ω , one must solve System (5) for all $E \in \mathcal{E}$, for all $t \in [0, T]$ it can be written by inverting the mass matrix :

$$(6) \quad \text{Solve } \forall E \in \mathcal{E}, \frac{d\mathbf{U}_E(t)}{dt} = \mathbf{H}_E(\mathbf{U}_E(t))$$

where $\mathbf{H}_E : \mathbb{R}^{3N_{dof}^E} \rightarrow \mathbb{R}^{3N_{dof}^E}$. The choice of the basis is essential since it drastically affects the shape of the system; for instance, the tensor product Legendre basis makes it diagonal.

4.3. Time discretization. Since there is no analytical solution to the ordinary differential equation derived in Equation (6), it is necessary to discretize and solve the time derivative numerically. This work will only consider fully explicit methods, which naturally introduce stability conditions. It is essential to carefully select the time step to avoid producing numerical noise. Choosing the time step is a critical concern because these instabilities can worsen over time and greatly diminish the accuracy of the numerical approximation. The explicit Euler method can be firstly considered while dealing with time discretization. Although using a basic first-order explicit time scheme may seem like the quickest and easiest approach, it is essential to acknowledge its limitations in practice. To avoid spoiling the potential benefits of high-order methods in space, it is recommended to implement more precise methods for advancing in time. As a result, our numerical examples use high-order explicit Runge-Kutta (RK) algorithms. It is worth noting that these time-marching methods can be considered as a combination of Euler schemes.

The time derivative in Equation (6) is discretized using the explicit RK method of order $q = p+1$ with p the polynomial degree in \mathcal{V}^p . Let use the discretization of $[0, T]$ introduce in Section 3 and as a reminder the time duration $[0, T]$ is discretized in sub interval $T^n = [t^n, t^{n+1}]$ and the corresponding time step is denoted by $\Delta t^n = t^{n+1} - t^n$. Moreover, $n = 0$ is used to define the beginning of the computation time and the initial condition:

$$\forall \mathbf{x} \in \Omega, \mathbf{U}(\mathbf{x}, t^0) = \mathbf{U}(\mathbf{x}, 0).$$

Using the initial condition the RK method computes the solution at $\mathbf{U}^{n+1}(\mathbf{x}) := \mathbf{U}(\mathbf{x}, t^{n+1})$ with the solution $\mathbf{U}^n(\mathbf{x}) := \mathbf{U}(\mathbf{x}, t^n)$ for all $\mathbf{x} \in \Omega$ and some sub-iterate solutions between t^n and t^{n+1} . The method reads :

- (1) Set $\mathbf{U}_E^{(0)} = \mathbf{U}_E^n$,
- (2) For $i = 1, \dots, q$ compute the intermediate states:

$$(7) \quad \mathbf{U}_E^{(i)} = \sum_{j=0}^{i-1} \alpha_{ij} \mathbf{W}_{ij}, \quad \mathbf{W}_{ij} = \mathbf{U}_E^{(j)} + \frac{\beta_{ij}}{\alpha_{ij}} \Delta t^n \mathbf{H}_E(\mathbf{U}_E^{(j)}),$$

- (3) Set $\mathbf{U}_E^{n+1} = \mathbf{U}_E^{(q)}$.

Coefficients β_{ij} and α_{ij} of the RK method are chosen according to the work of Shu [50] and Tadmor [49]. They give the class of Strong Stability Preserving Runge-Kutta (SSP-RK) schemes. They maintain the Total Variations Diminishing (TVD) property while striving for more precise time discretizations. Their work on this topic also involved collaborations with Gottlieb, see [30]. Essentially, these schemes can be viewed as combinations of first-order explicit Euler schemes, which makes it possible to determine stability conditions straightforwardly based on the coefficients involved.

Of course, because the time discretization is explicit, the time step Δt^n is limited by the Courant-Friedrichs-Lewy (CFL) condition. Classically, in similar work solving the SWE using DG methods with SSP-RK time discretization, the time step is given by the work of [14]:

$$\max_{E \in \mathcal{E}^n} \left(\frac{\lambda_E^n}{d_E} \right) \Delta t^n \leq \frac{1}{2p+1},$$

with

$$(8) \quad \lambda_E^n := \max_{F \in \partial E} \left(\max_{(x,y) \in F} \left(\left(\mathbf{u} \cdot \vec{n}_{E,F} \pm \sqrt{gh} \right) (x, y, t^n) \right) \right).$$

This CFL condition is obtained by a von Neumann stability analysis of the Runge-Kutta Discontinuous Galerkin (RKDG) methods explained before but on a linear advection equation. It has been proven for $p = 1$ in [13]. This results can be extended to non-linear problem as long as it can be diagonalized, see [10]. However, in the case $p \geq 2$, the number $\frac{1}{2p+1}$ is insufficient and numerical estimations must be used. See [14] for more details on these numerical coefficients.

4.4. Well balanced property and limiting procedure. Maintaining equilibrium states is desirable for methods handling the SWE. Specifically, we will focus on preserving *steady-states at rest*. These states are defined by the conditions $h + z_b \equiv C$ (a constant) and $\mathbf{q} \equiv 0$ over the whole domain Ω . If these conditions are not maintained, numerical waves can occur, as seen in [20]. Schemes that prevent this situation are referred to as well-balanced schemes. Unfortunately, the RKDG scheme defined in Section 4 does not exhibit well-balanced properties. This is because achieving well-balancing necessitates compatibility between the numerical flux and the source term discretization. Modification of System (5) can be found in [20] to achieve well-balanced properties.

System (5) can be rewritten as a matrix-vector system for each element:

$$(9) \quad \text{Solve } \forall E \in \mathcal{E}, \quad \frac{d\mathbf{U}_E(t)}{dt} = \mathbf{H}_E^{\text{wb}}(\mathbf{U}_E(t)),$$

with \mathbf{H}_E^{wb} the well balanced modified vector of the right hand side of Equation (6). \mathbf{H}_E^{wb} has been modified according to the work of Ern [20].

When dealing with hyperbolic conservation laws, it is crucial to consider the treatment of discontinuities. High-order numerical methods may generate artificial oscillations close to discontinuities, resulting in non-physical solutions, numerical instabilities, and unbounded computational outputs.

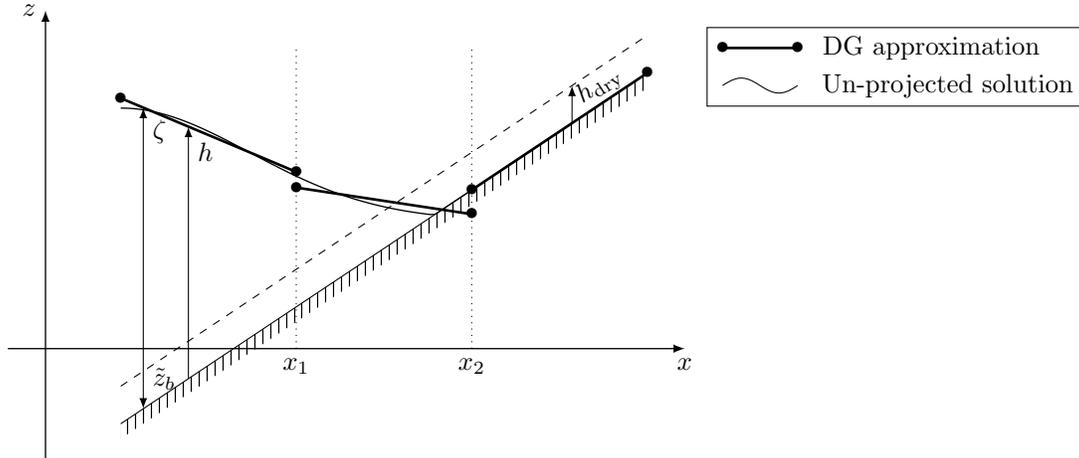


FIGURE 4. Example in one dimension of a wetting-drying situations

In this work, we focus on applying the Krivodonova's [34] method on a non-conformal mesh made up of rectangular elements with Δx (Δy) denoting the length of the rectangle along the x (y) axis. However, it is worth noting that a moment limiter [19] can be applied to triangular elements, representing a generalization for non-rectangular elements. A full description of the limiting procedure can be found in [34].

As long as the water height remains far from zero, those modifications ensure that the method is well-balanced and oscillations are limited. However, when considering water height is close to zero, the DG scheme shows some difficulties.

5. FLOODING AND DRYING TREATMENT

Simulating shallow water equations can pose a significant challenge due to dry areas with no water. This is especially true for applications involving fast-moving interfaces between wet and dry regions, such as dam breaks, flood waves, and run-up phenomena over shores and sea defense structures. Conventional numerical approaches may fail near the dry/wet front without proper consideration and yield undesirable negative water height. Since the computation domain is discretized with geometrical elements, the shoreline will likely cross an element, resulting in a partially wet element. The wetting and drying treatment identifies the wet and dry elements and modifies the numerical solution to avoid negative water height.

Figure 4 depicts a common problem when dealing with the shoreline using the DG space \mathcal{V}^1 . When the solution and the bathymetry are in this space, negative water height occurs unless the edge of the element is precisely at the shoreline.

Several wetting and drying treatments are available for continuous Galerkin-based methods, as referenced in [41]. One technique is the mesh adaption approach, which accurately tracks the dry front by modifying the meshes; an example can be found in [29]. However, this method can be computationally expensive. Another technique involves mesh reduction, where the dry elements are removed and later restored once they become wet; in his work Balzano in 1998 [2] presents a comparison of different methods. This approach may result in oscillation and loss of mass and momentum, leading to conservation failure. Moreover there exists other methods based on a kinetic approach, see [43, 51, 7].

Finally, the thin layer technique involves keeping a very thin layer of dry elements and incorporating them into the computation. The thin layer method has been commonly used in the context

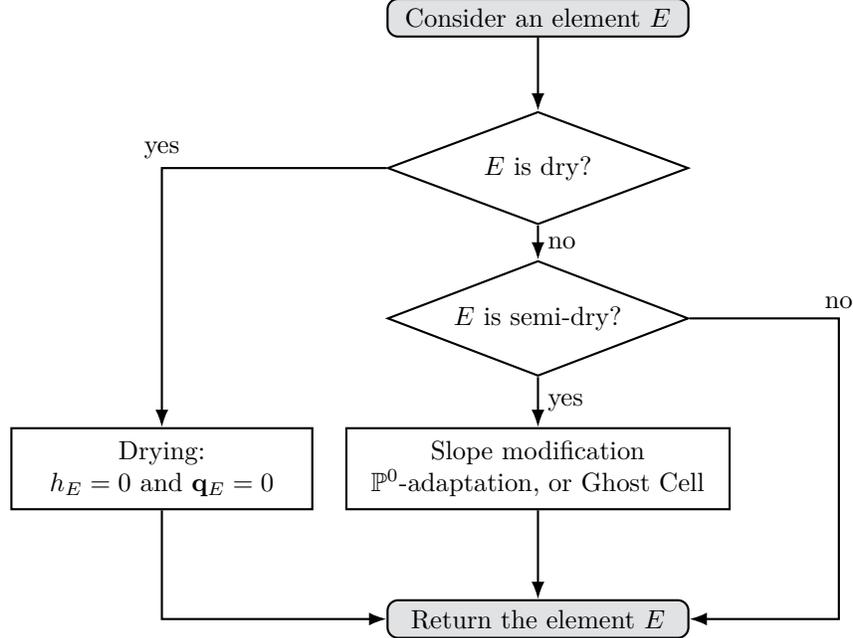


FIGURE 5. Flowchart of the thin layer approach

of FE methods [3, 41], FV methods [6] and DG methods [20, 8, 17] for solving shallow water equations.

The three methods presented in the following will be based on the thin-layer approach where the last one is part of the originality of this paper. Nevertheless, the post-treatment applied to semi-dry elements differs for each method. The thin layer approach relies on fixing a water height threshold, h_{dry} , and using it to detect partially wet elements by looking at water height on the mesh's edges, nodes or integration points and comparing it to h_{dry} . The thin layer method at a time level $n \in \mathbb{N}$ can be summarized as follows:

- First, for each elements $E \in \mathcal{E}^n$ the quantity m_E^n is computed. This is the minimum value of the water height on the element E at a given time t^n .

$$m_E^n := \min_{\mathbf{x} \in E} h_E^n(\mathbf{x}).$$

- Second, if $m_E^n \leq h_{\text{dry}}$ and the average water height (\bar{h}_E^n) is lower than the threshold, $\bar{h}_E^n \leq h_{\text{dry}}$ then the element is considered dry. That is to say that the water height and the discharge are set to 0 on the element E , i.e. $\tilde{\mathbf{U}}^n(\mathbf{x}) = 0, \forall \mathbf{x} \in E$. This is done to control dry elements and avoid nonphysical phenomena.
- Third, if $m_E^n \leq h_{\text{dry}}$ and the average water height (\bar{h}_E^n) is greater than the threshold, $\bar{h}_E^n > h_{\text{dry}}$ then the element is considered semi-dry. That is to say that the water height and the discharge are modified on the element E to enforce the positivity of the water height. This is done using positivity-preserving methods. In this work, the positivity-preserving methods are only applied to piecewise linear solutions. Consequently, the solution on semi-dry elements must be projected onto the piecewise linear space. This is done by a L^2 -projection of the solution onto \mathcal{V}^1 . Far from the semi-dry element, one can increase the order of approximation by using p -adaptation.

Figure 5 depicts a flowchart for the thin layer approach.

The first method, the most commonly used for DG methods, is Slope Modification. Essentially, this involves adjusting the slope of the water height while keeping the mean water height unchanged. The second method, named \mathbb{P}^0 -adaptation, draws inspiration from FV methods and assumes that the solution and bathymetry are piecewise constant on semi-dry elements. The final method, inspired by X-FE methods [39, 38, 23], maintains the concept of mesh adaptation and reduction without altering the mesh itself. Named the Ghost Cell method, this technique involves modifying the support of polynomial basis functions on semi-dry elements in order to calculate the solution only where the water height is positive.

5.1. Slope Modification. The notion of Slope Modification in the context of DG methods was initially introduced by Ern in 2007 [20]. This approach relies on using the thin layer technique to identify partially wet elements and subsequently modify the slope. This method is summarized as follows:

- On semi-dry elements, the slope of water height is modified to ensure that $h = 0$ at the node below the threshold. Furthermore, the discharge is adjusted in these modified elements such that $\mathbf{q} = 0$ at nodes where $h = 0$. For further information regarding this approach, the interested reader is referred to [20].

While enforcing the positivity of the water height, this method preserves mass conservation but does not preserve momentum-conservative.

Building on the method mentioned above, Bunya introduced a new technique in 2009 [8] that does not violate the positivity of mass and is momentum-conservative. However, in this method, a special treatment for the flux computation is necessary to prevent instability. Furthermore, Xing in 2010 and 2013 [58, 57], Zhang in 2012 [59], and Duran in 2014 [17] use a positivity-preserving method similar to Bunya's approach, but it is more straightforward to implement. This method exhibits a CFL-like condition and Slope Modification technique of the water height and discharge. It is designed to be both mass and momentum-conservative. For simplicity's sake, we will only consider piecewise linear solutions ($p = 1$) in this context.

Working with the positivity limiter of [58] implies using the β -point Gauss-Lobatto quadrature, where β is the smallest integer such that $2\beta - 3 \geq p$ with p the polynomial degree of the DG space. This quadrature is used because it is designed to place quadrature points on the edge of the element. In the following, let us denote S_E , the set of points of this quadrature rule, and \hat{w}_1 , the weight associated with the first quadrature point. The CFL-like condition has been derived so that the mean water height on each element is always positive. In other words, for all $n \in \mathbb{N}$ and for all elements E if $\bar{h}_E(t^n) \geq 0$ then $\bar{h}_E(t^{n+1}) \geq 0$. This condition is given by

$$(10) \quad \max_{E \in \mathcal{E}^n} \left(\frac{\lambda_E^n}{d_E} \right) \Delta t^n \leq \hat{w}_1,$$

with λ_E^n having the same definition as in Equation (8). In the case of $p = 1$ the Gauss-Lobatto quadrature points are exactly nodes of the element with a weight of $\hat{w}_1 = 1/3$. The positivity limiter is the enforced in two steps:

- First, for each elements E , the quantity \tilde{m}_E^n is computed. This quantity is the minimum value of the water height on the element E at the Gauss-Lobatto quadrature points at a given time t^n . It is given by :

$$\tilde{m}_E^n := \min_{\mathbf{x} \in S_E} h_E^n(\mathbf{x}).$$

- Second, $h_E^n(\mathbf{x}) \geq 0, \forall \mathbf{x} \in S_E$ needs to be enforced because it is a sufficient condition to ensure the positivity of the water height average under the CFL-like condition of Equation (10). This can be done using the positivity preserving limiter of [58]. We denote

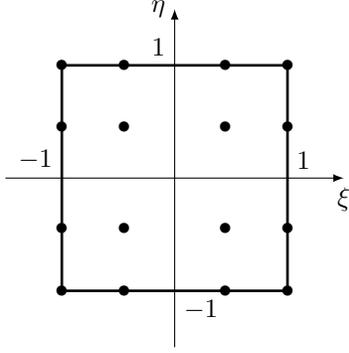


FIGURE 6. Gauss-Lobatto quadrature points on a two-dimensional element with 8 quadrature points

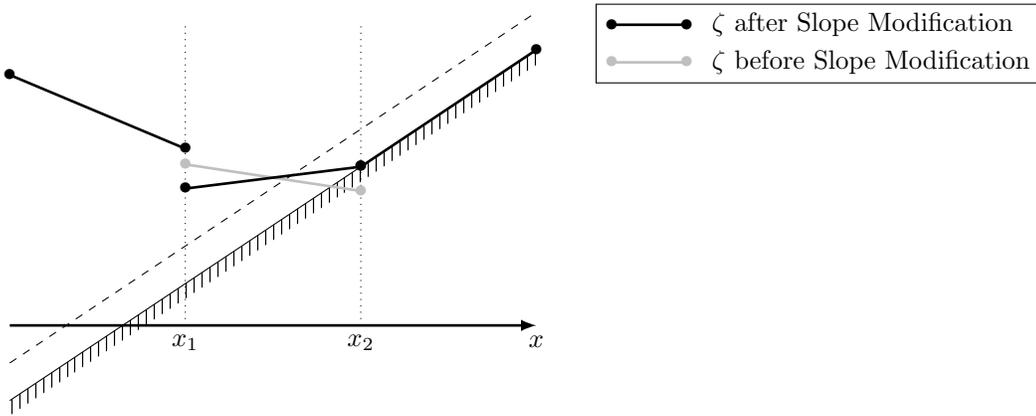


FIGURE 7. Example in one dimension of the Slope Modification method

$\bar{\mathbf{U}}_E^n = (\bar{h}_E^n, \bar{\mathbf{q}}_E^n)^T$ the mean value of \mathbf{U} on the element E at a time level n . Assuming that $\bar{h}_E^n \geq 0$, the modified solution $\tilde{\mathbf{U}}^n$ is given by

$$\tilde{\mathbf{U}}^n(\mathbf{x}) = \theta_E^n (\mathbf{U}^n(\mathbf{x}) - \bar{\mathbf{U}}_E^n) + \bar{\mathbf{U}}_E^n,$$

where

$$\theta_E^n = \min \left\{ 1, \frac{\bar{h}_E^n}{\bar{h}_E^n - \tilde{m}_E^n} \right\}.$$

It is easy to observe that $\tilde{h}_E^n(\mathbf{x}) \geq 0$, $\forall \mathbf{x} \in S_E$ and that this method is mass-conservative and momentum-conservative. Once the modified solution $\tilde{\mathbf{U}}^n$ is computed, its polynomial decomposition can be quickly recovered by a L^2 -projection of $\tilde{\mathbf{U}}^n$ onto \mathcal{V}^p . By construction this method is also well suited for two-dimensional problems, see Figure 6.

Figure 7 shows how, in the context of Figure 4, the Slope Modification procedure modifies the solution to enforce the positivity of the water height. The solution before the positivity-preserving method is depicted in gray.

However, in the latter two methods, it is possible to observe very high fluid velocity values ($\mathbf{u} = \mathbf{q}/h$) in nearly dry regions. Unfortunately, the rate at which h and \mathbf{q} converge towards zero is unknown. Those overshoots of the fluid velocity may produce nonphysical perturbations in the shoreline area and degrade the solution. Moreover, they also lead to tiny time steps. To address this issue, Ern has set a maximum value for the fluid speed and adjusted the discharge accordingly.

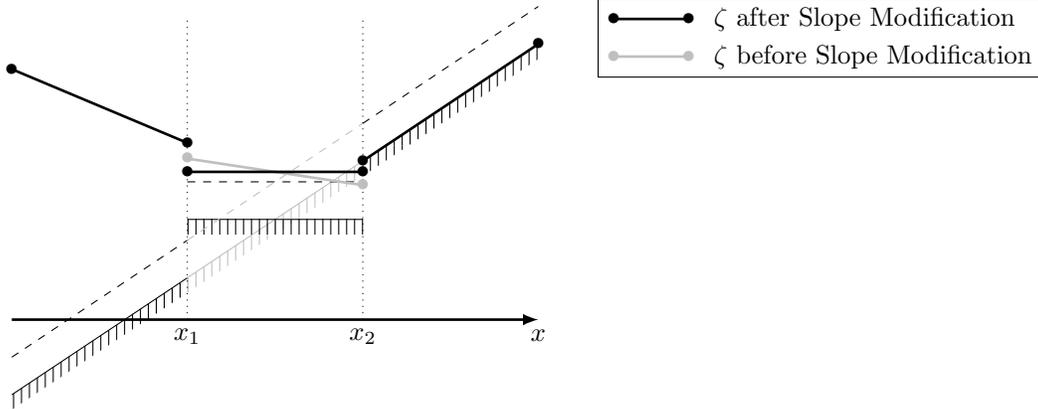


FIGURE 8. Example in one dimension of the \mathbb{P}^0 -adaptation method

Similarly, Xing suggests that $\mathbf{u} = 0$ should be set if $h \leq 10^{-6}$. The fundamental idea behind both strategies is that the fluid velocity in nearly dry areas should be equivalent to that in wet regions. More recently, Vater [53, 54] has proposed a new approach that involves determining the maximum velocity value based on the maximum velocity around semi-dry elements. It's worth noting that this method can alter the solution for the given problem since there is no way to predict whether velocity is bounded.

Even though these methods are well-balanced in fully wet areas, they are not well-balanced in partially wet areas. The free-surface elevation is also modified by modifying the water height slope to eliminate negative water height. It degrades the lake at rest state and produces nonphysical perturbations in the shoreline area.

5.2. \mathbb{P}^0 -adaptation. In order to solve the issue of very large values of speed in the shoreline area, the idea is to have a strategy close to what is done in FV methods. The idea is to consider that the solution is piecewise constant on semi-dry elements. This idea is greatly inspired by works where hp-adaptation is done in DG methods solving Euler Equations [40, 55] and solving SWE [52, 35, 22]. In this work, only p-adaptation is applied to the shoreline treatment. It is observed that in FV methods, the thin layer approach does not produce an overshoot in the fluid velocity. So the idea is to consider the solution piecewise constant ($p = 0$) on semi-dry elements and not modify the polynomial degree of the DG method on other elements. This is implemented into a positivity preserving procedure called \mathbb{P}^0 -adaptation. This method is quite short to describe:

- The polynomial degree on the semi-dry element E is set to 0, preserving the average water height and discharge. Do not forget to modify the polynomial order of the bathymetry as well.

It is easy to check that this method is mass-conservative and momentum-conservative as long as the average water height remains positive. To have this condition, the CFL-like condition in Equation (10) needs to be used. Figure 8 shows how, in the context of Figure 4 the \mathbb{P}^0 -adaptation procedure modifies the solution to enforce the positivity of the water height. The solution before the positivity-preserving method is depicted in gray.

Adjusting the polynomial order is a simple process thanks to the hierarchism property of the Legendre or Dubiner basis. We can easily introduce modal coefficients to expand the modal space while setting them to zero. Conversely, we can reduce the modal space by removing the corresponding higher-order modal coefficients. It is noticeable that the difference in polynomial order between adjacent elements does not exceed a value determined by the user, which is set to 1 in the current investigation. Moreover, to be sure that potential issues due to the shoreline are avoided,

one can decide to not only set semi-dry elements to \mathbb{P}^0 but also the elements adjacent to them in a user-defined stencil.

This approach is designed to conserve mass and momentum while limiting high fluid velocities near the shoreline through numerical diffusion via FV methods. It is a simple technique that delivers reliable outcomes, although it can be unbalanced in the shoreline region. The positivity-preserving technique is well-balanced when applied in fully wet regions since it is inactive in these areas. Nevertheless, the positivity enforcing method can change free surface elevation and produce nonphysical perturbations in the solution at the lake at rest state. This method may locally change the bathymetry and may lead to spurious oscillations. A careful analysis has to be done in order to capture the lake at rest state. The following method is designed to overcome these issues.

5.3. Ghost Cell. The present study highlights the inadequacy of Slope Modification and \mathbb{P}^0 -adaptation techniques in the shoreline region, as they can cause perturbations that significantly alter the solution. These perturbations are particularly pronounced in areas of high fluid velocity, leading to uncontrolled solutions and a potential computation shutdown. To overcome this issue, the concept of Ghost Cell is introduced, which provides a well-balanced positivity-preserving method. This approach is heavily inspired by the X-FE methods, which were first introduced by Belytschko in 1999 [4] and have since undergone extensive development [39, 38, 32, 23]. The X-FE method is an extension of FE methods that are designed to model smooth solutions. However, one can observe that quantities may vary rapidly over a small scale, resulting in solutions exhibiting discontinuities, singularities, high gradients, or other non-smooth properties. Such phenomena are common in solid mechanics, where cracks, dislocations, and inclusions are prevalent, and in fluid mechanics, where shocks and near interfaces in multi-phase flows are common. Two fundamental approaches have been proposed to address this issue: modifying the mesh to conform with such singularities, modifying the polynomial approximation space by adding special shape functions to the bases, or replacing some shape functions. The latter approach is closely related to the concept of Ghost Cell. The challenge with a semi-dry element is that one part is wet, and the other is dry. Since the solution is considered linear on the element, having such a property without negative water height is impossible. To overcome this challenge, two ideas have been proposed: splitting the element (into two elements in 1D), which involves a meshing step and computation cost, or modifying the support of the shape function. In the latter approach, the shape function is no longer defined on the entire element but only on a sub-part, the wet part. The shape functions are considered null on the dry part since the solution is null. The following provides a comprehensive explanation of the Ghost Cell method. Figure 9 shows how, in the context of Figure 4, the Ghost Cell procedure modifies the solution to enforce the positivity of the water height. The solution before the positivity-preserving method is depicted in gray.

This approach involves dividing semi-dry elements into two distinct parts: the wet portion and the dry portion. Let us consider a semi-dry element $E \in \mathcal{E}^n$. From this element, we can create two sub-elements: the wet sub-element $E_{\text{wet}} = \{\mathbf{x} \in E \mid h_E^n(\mathbf{x}) \geq h_{\text{dry}}\}$ and the dry sub-element $E_{\text{dry}} = \{\mathbf{x} \in E \mid h_E^n(\mathbf{x}) \leq h_{\text{dry}}\}$.

The positivity procedure only considers the DG space \mathcal{V}^1 , so determining where to split E is straightforward. For a one-dimensional problem, the shoreline is represented by a point, denoted as x^* . This point can be easily accessed by locating the crossing point between two lines: the linear projection of the water height and the constant h_{dry} . In a two-dimensional problem, the shoreline is represented by a line, denoted as $\mathcal{S}_E^n = \{\mathbf{x} \in E \mid h_E^n(\mathbf{x}) = h_{\text{dry}}\}$. This line can be easily accessed by locating the crossing line between two planes: the linear projection of the water height and the horizontal plane h_{dry} . In one dimension, the shapes of E_{wet} and E_{dry} are simple: two segments. In two dimensions, it is less obvious, therefore, more work needs to be done on the splitting. This

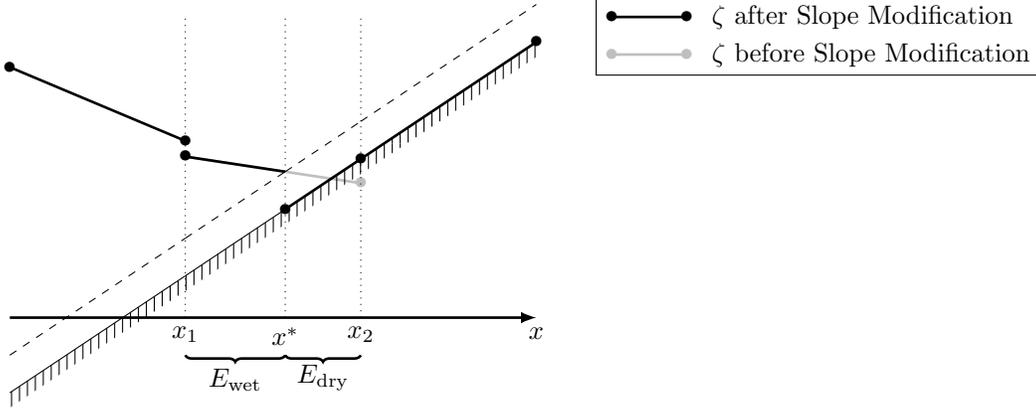


FIGURE 9. Exemple in one dimension of the Ghost Cell method

work focuses on establishing the Ghost Cell procedure in one dimension, and extending it to two dimensions is not trivial or addressed in this work.

Using the splitting of semi-dry elements and considering mappings are linear. Integration on the physical element E and the reference element can be reworked. For a given function $f : E \rightarrow \mathbb{R}$ substitution theorem leads to a transformation of integral:

$$\int_E f(x, y) dE = \int_{E_{\text{wet}}} f(x, y) dE_{\text{wet}} + \int_{E_{\text{dry}}} f(x, y) dE_{\text{dry}}$$

Moreover, if f is differentiable on E with the chain rule, it gives the transformation of integral:

$$\int_E \nabla f(x, y) dE = \int_{E_{\text{wet}}} \nabla f(x, y) dE_{\text{wet}} + \int_{E_{\text{dry}}} \nabla f(x, y) dE_{\text{dry}}$$

Moreover, another modification to previous definitions made before can be done. Based on the splitting of semi-dry elements and with the thin layer approach, the definition of \mathbf{U} on the element E can be rewritten as follows:

$$(11) \quad \forall (\mathbf{x}, t) \in E \times T^n, \quad \mathbf{U}_E(\mathbf{x}, t) := \begin{cases} \Phi(\mathbf{x}) \cdot \mathbf{U}_E(t) & \text{if } \mathbf{x} \in E_{\text{wet}} \setminus F^* \\ 0 & \text{otherwise} \end{cases}$$

where F^* is the common boundary face of E_{wet} and E_{dry} , it is defined as $F^* := \mathcal{F}^{E_{\text{wet}}} \cap \mathcal{F}^{E_{\text{dry}}}$.

One can notice that for all elements $E \in \mathcal{E}^n$, if $E_{\text{wet}} = E$, previous modifications are inactive. Consequently, these modifications applied to the well-balanced weak formulation are effective only if a semi-dry element is considered. Therefore, the weak formulation on a semi-dry element E can be rewritten as follows, noting that $\mathbf{U}|_{E_{\text{dry}}} = 0$ and $\mathbf{U}|_{F^*}$ is continuous:

$$(12) \quad \int_E \varphi(\mathbf{x}) \partial_t \mathbf{U}(\mathbf{x}, t) dE = \int_{E_{\text{wet}}} \partial_{x_i} \varphi(\mathbf{x}) \mathbf{G}_i(\mathbf{U}(\mathbf{x}, t)) dE_{\text{wet}} + \int_{E_{\text{wet}}} \varphi(\mathbf{x}) \mathbf{S}(\mathbf{U}(\mathbf{x}, t), \tilde{z}_b(\mathbf{x})) dE_{\text{wet}} \\ - \sum_{F \in \mathcal{F}^{E_{\text{wet}}} \setminus F^*} \int_F \varphi(\mathbf{x}) \mathbf{G}_F^{\text{wb}}(\mathbf{U}(\mathbf{x}, t), \tilde{z}_b(\mathbf{x})) dF - \sum_{F \in \mathcal{F}^{E_{\text{dry}}} \setminus F^*} \int_F \varphi(\mathbf{x}) \mathbf{G}_F^{\text{wb}}(\mathbf{U}(\mathbf{x}, t), \tilde{z}_b(\mathbf{x})) dF.$$

This new expression in Equation (12) shows that variations of $\mathbf{U}|_E$ rely only on the wet part of the element and the boundary flux. Consequently, any integration is done on the dry part, so the mapping between E_{dry} and \hat{E}_{dry} is useless. However, the faces of the dry part need to be tracked since they are needed to compute boundary fluxes. In addition, the Ghost Cell method modifies

discharge to enforce that $\mathbf{q}(\mathbf{x}) = 0$ for all $\mathbf{x} \in F^*$. It is done such that the average discharge is kept unchanged.

The Ghost Cell method in one dimension on a semi-dry element can be summarized as follows: Consider an element $E = [x_i, x_{i+1}]$ and its corresponding length $x_{i+1} - x_i$ and middle $x_b = (x_i + x_{i+1})/2$. Then denote x^* the point such that $h_E(x^*) = h_{\text{dry}}$.

- First, compute x^* , such that $h_E(x^*) = h_{\text{dry}}$;
- Second,
 - If $h(x_i) \leq h_{\text{dry}}$ the left node is dry, thus define the wet part as $E_{\text{wet}} = [x^*, x_{i+1}]$;
 - If $h(x_{i+1}) \leq h_{\text{dry}}$ the right node is dry, thus define the wet part as $E_{\text{wet}} = [x_i, x^*]$;
- Third, modify \mathbf{q} such that $\mathbf{q}(x^*) = 0$ and the average keeps unchanged.

One can see that this approach is well-balanced, mass conservative, momentum conservative, and positivity preserving. Only considering the wet portion of the elements in the weak formulation effectively prevents overshoots in fluid velocity, a common issue with Slope Modification and \mathbb{P}^0 -adaptation. However, it does not completely eliminate the possibility of large, nonphysical fluid velocity that can alter the expected outcome. The Ghost Cell method's overshoots of fluid velocity on the shoreline depend on the convergence rates of h and \mathbf{q} towards zero. To manage this issue, it is necessary to control these rates.

6. VALIDATION AND NUMERICAL RESULTS

The first four test cases were focused on validating *RIVAGE*, our in house code, in its one-dimensional and two-dimensional form. In order to assess the accuracy of the numerical solutions generated by the DG solver, we compared them with known analytical solutions, while for experimental test cases, we compared the numerical solutions with data from the literature. Finally, the last test case involves using *RIVAGE* to solve a real experimental problem, using the block adaptation method.

6.1. Lake at rest. The first test case is aiming to verifying whether the DG formulation with drying treatments preserve the still water steady state with a non-flat bathymetry containing a wet dry interface. The computational domain is considered one-dimensional $\Omega = [0, 1]$ and the simulation time $T = 1.5s$. The bathymetry is given for all $\mathbf{x} \in \Omega$:

$$z_b(x) = \begin{cases} 1.25(x - 0.25) & \text{if } 0.25 \leq x < 0.45, \\ 0.25 & \text{if } 0.45 \leq x < 0.55, \\ 1.25(x - 0.75) & \text{if } 0.55 \leq x < 0.75, \\ 0 & \text{otherwise.} \end{cases}$$

Two different initial datas are considered:

- Case 1 : the initial stationary solution is for all $x \in \Omega$:

$$h(x, 0) = \max(0.2 - z_b(x), 0) \text{ and } \mathbf{q}(x, 0) = 0.$$

- Case 2 : the initial stationary solution is for all $x \in \Omega$:

$$h(x, 0) = \max(0.16 - z_b(x), 0) \text{ and } \mathbf{q}(x, 0) = 0.$$

Moreover solid wall boundary conditions are considered.

Solutions of these two cases are computed with Ω discretized with 200 elements. It gives regularly spaced elements of size $\Delta x = 0.05m$. With this mesh for the Case 1, the shoreline fits the mesh whereas for the Case 2 the shoreline lies between two nodes. In addition for these cases the solution is considered piecewise linear ($p = 1$), SSP-RK of order 2 is employed, $\alpha_{\text{CFL}} = 0.5$ and $h_{\text{dry}} = 10^{-6}$.

For these two test cases the initial steady state should be preserved for the whole computation time. Table 1 gives L^2 -error and L^∞ -error of the numerical solution compared with the initial data

TABLE 1. Errors for the lake at rest test case with different flooding and drying treatments.

Case	Flooding and drying method	L ² -error			L [∞] -error		
		h	q _x	u	h	q _x	u
1	Slope Modification	6.90 10 ⁻⁹	0.00	0.00	1.19 10 ⁻⁸	0.00	0.00
–	ℙ ⁰ -adaptation	5.30 10 ⁻¹⁶	0.00	0.00	8.33 10 ⁻¹⁷	0.00	0.00
–	Ghost Cell	5.29 10 ⁻¹⁶	0.00	0.00	8.33 10 ⁻¹⁶	0.00	0.00
2	Slope Modification	7.75 10 ⁻⁵	2.74 10 ⁻⁵	1.35 10 ⁻¹	1.43 10 ⁻³	2.18 10 ⁻⁰⁴	3.83 10 ¹
–	ℙ ⁰ -adaptation	4.07 10 ⁻⁴	2.01 10 ⁻⁴	2.88 10 ⁻²	3.45 10 ⁻³	8.22 10 ⁻⁴	4.27
–	Ghost Cell	2.18 10 ⁻¹⁶	0.00	0.00	5.55 10 ⁻¹⁷	0.00	0.00

at $t = 1.5s$. It can be seen that if the mesh coincide with the shoreline, the Case 1, the numerical solution computed with any drying treatments preserve the lake at rest state. Nevertheless if the shoreline does not fit the mesh only the Ghost Cell method is well balanced, the initial state is preserved. Slope Modification and ℙ⁰-adaptation methods are not well balanced if shoreline falls between two nodes.

This test case enlighten that the the DG formulation is well balanced without dry areas. Nevertheless if a problem with dry areas is considered the DG formulation is fully well balanced only using Ghost Cell methods unless the mesh fits the shoreline position.

6.2. Carrier and Greenspan benchmark. The Carrier-Greenspan benchmark, created by Carrier and Greenspan in 1958 [9], features a periodic wave moving up and down on a sloping beach, involving both wetting and drying in each period. This test case has been used in the context of FV methods by Marche in 2007 [37], as well as in DG methods by Bokhove in 2005 [5] and Duran in 20014 [18].

The solution of the SWE for this test case is the motion of a periodic wave of dimensionless amplitude A^* and frequency ω^* traveling shoreward and being reflected out to sea generating standing wave on a beach plane. This problem is one-dimensional, hence the computational domain is $\Omega = [-l, l + 4]$ with l the typical length scale of this specific problem. Moreover α the beach slope is introduced. Dimensionless variables are defined as follows:

$$x^* = x/l, \quad \zeta^* = \zeta/(\alpha l), \quad u^* = u/\sqrt{g\alpha l} \quad \text{and} \quad t^* = t\sqrt{l/(\alpha g)}.$$

The analytical solution is given by the following problem:

$$(13) \quad \begin{cases} \lambda^* = 2(t^* + u^*) \\ \sigma^* = 4\sqrt{\zeta^* - x^*} \\ \zeta^* = \frac{\omega^* A^*}{4} J_0(\omega^* \sigma^*) \cos(\omega^* \lambda^*) - \frac{(u^*)^2}{2} \\ u^* = \frac{-\omega^* A^* J_1(\omega^* \sigma^*) \sin(\omega^* \lambda^*)}{\sigma^*} \end{cases}$$

where J_0 and J_1 are the Bessel functions of zero and first order respectively. This analytical solution is commonly used to assess the model's ability to handle run-up and run-down phenomena, as well as to examine wave dynamics near a continental shelf. The equation is applicable when $0 \leq A^* \leq 1$, and $A^*/4$ signifies the maximum vertical distance the shoreline can move. The initial condition for $t = 0$ is provided, and the analytical variations of the surface elevation at the left boundary

TABLE 2. L^2 -error, L^∞ -error and convergence rates for the Carrier and Greenspan benchmark at $t^* = 4\tau^*$.

Flooding and drying method	Δx	h		q_x		u	
		L^2 -error	r	L^2 -error	r	L^2 -error	r
Slope Modification	0.24	$1.38 \cdot 10^{-3}$		$1.52 \cdot 10^{-3}$		$5.31 \cdot 10^{-1}$	
—	0.12	$3.44 \cdot 10^{-4}$	2.01	$4.77 \cdot 10^{-4}$	1.67	$1.61 \cdot 10^{-1}$	1.72
—	0.06	$7.28 \cdot 10^{-5}$	2.24	$2.25 \cdot 10^{-4}$	1.08	$3.64 \cdot 10^{-2}$	2.14
—	0.03	$4.18 \cdot 10^{-5}$	0.80	$1.12 \cdot 10^{-4}$	1.00	$1.95 \cdot 10^{-2}$	0.90
—	Fitted	—	1.75	—	1.24	—	1.64
\mathbb{P}^0 -adaptation	0.24	$4.17 \cdot 10^{-3}$		$1.42 \cdot 10^{-2}$		$1.35 \cdot 10^{-1}$	
—	0.12	$1.54 \cdot 10^{-3}$	1.43	$5.42 \cdot 10^{-3}$	1.39	$8.23 \cdot 10^{-2}$	0.71
—	0.06	$6.84 \cdot 10^{-4}$	1.17	$2.26 \cdot 10^{-3}$	1.26	$8.09 \cdot 10^{-3}$	3.357
—	0.03	$4.64 \cdot 10^{-4}$	0.56	$8.09 \cdot 10^{-4}$	1.48	$3.88 \cdot 10^{-3}$	1.06
—	Fitted	—	1.07	—	1.37	—	1.87
Ghost Cell	0.24	$1.72 \cdot 10^{-3}$		$2.56 \cdot 10^{-3}$		$1.52 \cdot 10^{-1}$	
—	0.12	$4.92 \cdot 10^{-4}$	1.81	$6.90 \cdot 10^{-4}$	1.89	$5.85 \cdot 10^{-2}$	1.38
—	0.06	$2.05 \cdot 10^{-4}$	1.26	$2.68 \cdot 10^{-4}$	1.37	$2.99 \cdot 10^{-3}$	4.296
—	0.03	$5.51 \cdot 10^{-5}$	1.89	$1.35 \cdot 10^{-4}$	0.99	$1.69 \cdot 10^{-1}$	-5.82
—	Fitted	—	1.62	—	1.41	—	0.38

serve as an offshore inlet boundary condition, allowing us to generate motion. For a comprehensive explanation of the solution, please refer to Carrier and Greenspan’s original research [9].

In this test case, we compute the numerical solution using a dimensionless amplitude of $A^* = 0.6$, $\omega^* = 1$, a length scale of $l = 20m$, and a beach slope of $\alpha = 1/30$. The solution is computed on the domain $\Omega = [-20, 4]$ and until $t^* = 4\tau^*$, where $\tau^* = 2\pi/\omega^*$ is the period. We perform four computations using different mesh sizes, $\Delta x = \{0.24, 0.12, 0.06, 0.03\}$, each composed of regularly spaced one-dimensional elements. The solution is considered piecewise linear ($p = 1$), and we use SSP-RK of order 2 with $\alpha_{\text{CFL}} = 0.5$ and $h_{\text{dry}} = 10^{-6}$. We also employ a stencil of 10 elements for the \mathbb{P}^0 -adaptation method. At $t = 4\tau^*$, we present the L^2 -error, L^∞ -error, and convergence rates in Table 2. For the three drying treatments, we find that the computed orders of accuracy are between 1 and 2 for h and q_x . However, convergence rates for u are not constant and are calculated when u is null at the end of four periods.

By analyzing how well the numerical solution recovers the shoreline position, further insights can be gained. The time series of the shoreline position for the three drying treatments in Figure 11 shows that the computational results are in phase with the analytical solution. During the run-up phase, the drying treatments perform well, while the \mathbb{P}^0 -adaptation method performs the worst and the Ghost Cell method performs the best during the run-down phase. To compare the numerical solutions to the analytical solution, Figure 10 shows a snapshot during the run-down phase, as it has been observed that drying treatment does not perform equally during the run-up and run-down phases. Non-physical results are obtained around the shoreline from the fluid speed. Additionally, the Ghost Cell method produces few overshoots in the fluid speed, while the \mathbb{P}^0 -adaptation method produces the most.

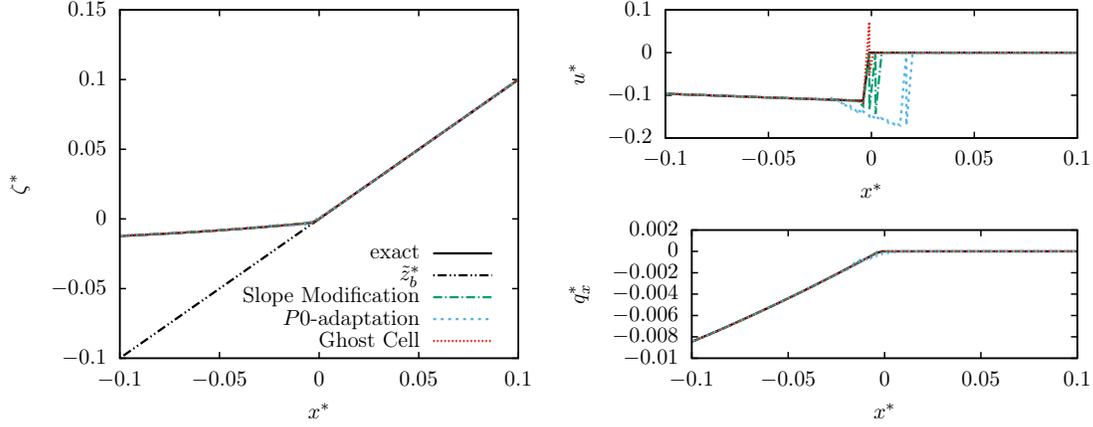


FIGURE 10. Carrier and Greenspan, Zoom on the shoreline for the numerical solutions at $t^* = 3.3T^*$ and for $\Delta x = 0.06m$ with $p = 1$ and SSP-RK method of order 2

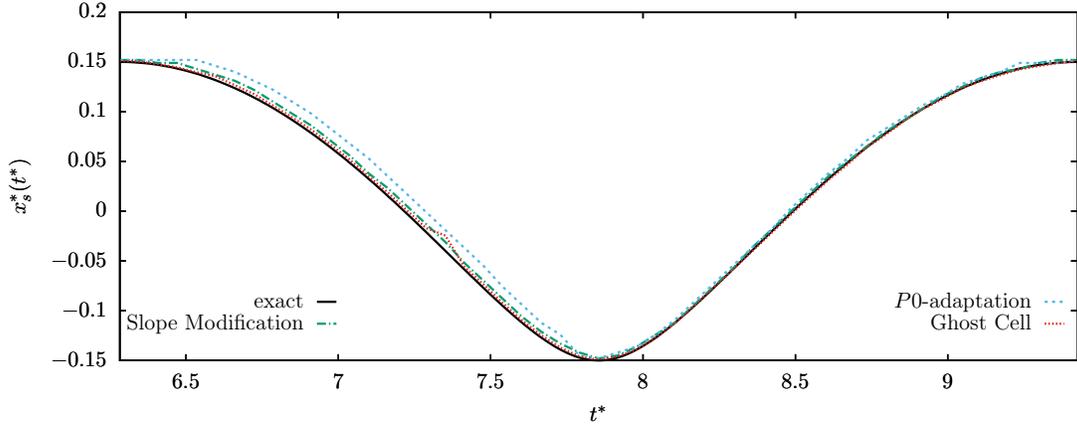


FIGURE 11. Carrier and Greenspan, time series of the shoreline position for $\Delta x = 0.06m$ and $2T^* \leq t^* \leq 3T^*$ with $p = 1$ and SSP-RK method of order 2

6.3. Roeber test case. The last benchmark for the one-dimensional variant of our DG solver involves an experimental test case. This standard test case for Boussinesq-type equations [47, 46] can also be applied to SWE [33]. The test examines wave transformation over an idealized fringing reef and was conducted through laboratory experiments spanning from 2007 to 2009 at the O.H. Hinsdale Wave Research Laboratory of Oregon State University.

The problem at hand involves a channel that measures $83.7m$ in length and $2.5m$ in depth. The shallow water assumption is upheld, with the vertical length scale being smaller than the horizontal length scale. At one end of the flume, a solid wall is present. The flume's bathymetry involves a reef with an exposed crest and a flat section after the crest. The crest is $0.065m$ above the free surface, with both sides of the crest having a slope of $1/12$. The flat section is $0.14m$ deep. The problem can be viewed as a one-dimensional problem dealing with SWE. The initial condition is a simulated solitary wave, which is a first-order solution of the Boussinesq equation centered on $x_0 = 17.6m$. The wave has a dimensionless amplitude of $A/h_0 = 0.3$, with $h_0 = 2.5m$. It is given

by :

$$\zeta(x) = \frac{A}{\cosh\left(\sqrt{\frac{3A}{4h_0^3}}(x - x_0)\right)^2} \quad \text{and} \quad q(x) = \sqrt{g(h_0 + A)}\zeta(x).$$

For this problem, variables denoted with a star are dimensionless: $\zeta^* = \zeta/h_0$ and $t^* = t\sqrt{g/h_0} + 55.3$. Experimental results are captured by 14 wave gauges placed in the middle of the flume. 8 gauges are placed before the crest and 6 after the crest.

For our numerical simulation, the one-dimensional computational domain $\Omega = [0, 83.7]$ is used, and the simulation is carried until $T = 93.3s$. Solid wall boundary conditions are applied on both sides of the flume. Two different meshes are used: one with 200 regularly spaced elements of size $\Delta x = 0.4185m$, and the other with 2000 regularly spaced elements of size $\Delta x = 0.04185m$. With the first mesh, the solution is considered piecewise linear ($p = 1$) and use a SSP-RK of order 2. For the Slope Modification drying treatment, numerical parameters are set $\alpha_{\text{CFL}} = 0.8$ and $h_{\text{dry}} = 10^{-3}$. With Ghost Cell and \mathbb{P}^0 -adaptation methods, numerical parameters are set to $\alpha_{\text{CFL}} = 0.9$ and $h_{\text{dry}} = 10^{-6}$. The stencil for the \mathbb{P}^0 -adaptation method is 10. With the second mesh, the solution is considered piecewise constant using the FV method. The high-resolution mesh solution of FV methods is labeled *P0-HR* in the figures.

In Figure 12, we can see the numerical solution at various times before and after the wave breaks over the crest. This solution was computed using the Ghost Cell method. While the computed solution aligns well with experimental data, there are still some discrepancies, particularly at $t^* = 67.05$. At this time, the wave starts to break and develop a plunging breaker. While SWE replicates this phenomenon by creating a vertical-faced propagating bore, the wave height is underestimated, though the total water mass remains preserved. As the wave breaks over the crest, it travels down the back slope of the reef crest, creating a flow that displaces the previously stagnant water and a hydraulic jump off the back reef. Laboratory observations have confirmed the generation of the hydraulic jump. Numerically solved SWE correctly predict the phase and amplitude of the discontinuities of the hydraulic jump after the crest.

Displayed in Figure 13 are numerical solutions and experimental data for two selected wave gauges. The first one is before the crest and the second one is after the it. For the first gauge, all three drying treatments perform comparably and align with the high-resolution FV solution. Additionally, as previously mentioned, the initial wave pass is accurately captured despite a slight underestimation of its height. Furthermore, waves reflected by the wall are also well-predicted albeit with a slight delay. However, the SWE model does not account for dispersive effects, thus free surface oscillations are not accurately represented. For the second gauge, the three drying treatments have also equivalent performance and align with the high-resolution FV solution. Secondly, it reinforces previous findings regarding the accurate prediction of the hydraulic jump's height and phase. Lastly, it is worth noting that while the reflected waves between the wall and the crest are accurately predicted in height, there is a notable phase delay. To address this delay, a different model than SWE is required. Kazolea and Roeber used a Boussinesq model that considers dispersive effects, which yielded better results than SWE.

The final one-dimensional test case has been completed and the results indicate that the SWE solved using the DG formulation performs exceptionally well across all benchmarks. While the convergence rates align with expectations, they fall slightly below the optimal order of accuracy of 2 for a piecewise linear solution. Interestingly, all three drying treatments produced similar outcomes, but the Ghost Cell method proved to be the most resilient, particularly when dealing with non-physical fluid speeds.

6.4. Two-dimensional parabolic bowl. This test case is the second one using a parabolic bowl as bathymetry. However, this time, the problem is two-dimensional. It allows to assess the capacity

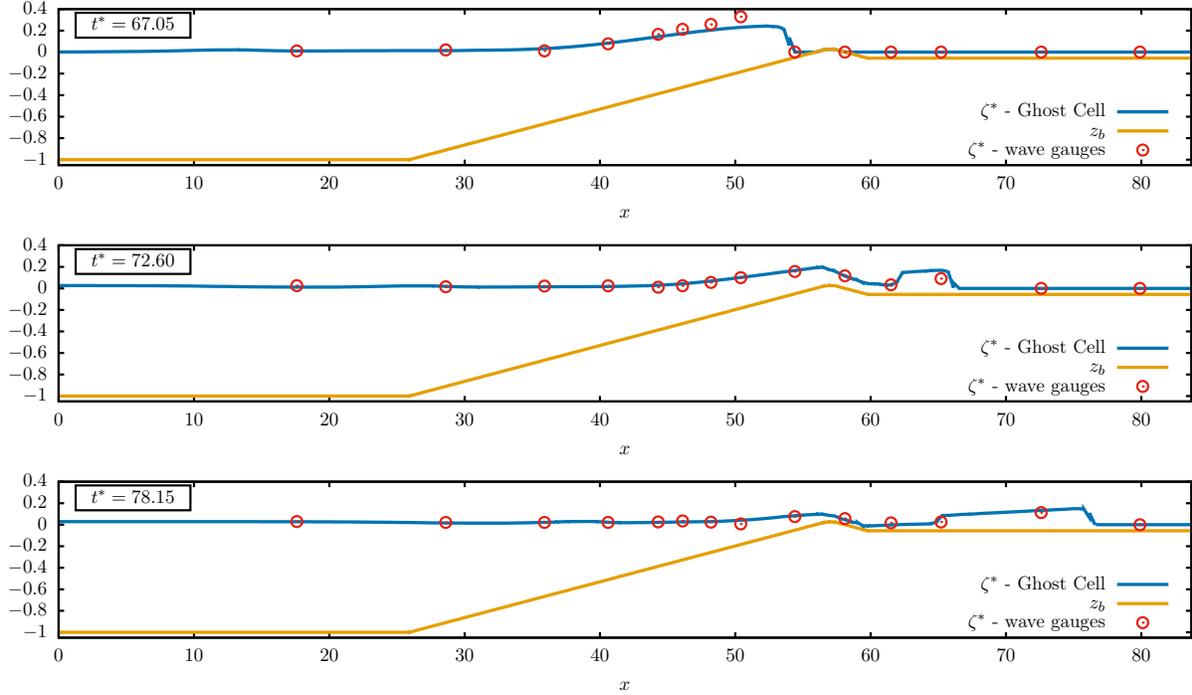


FIGURE 12. Roeber test case, numerical solution at different times compared to wave gauges data with $\Delta x = 0.4185m$, $p = 1$, SSP-RK method of order 2 and with the Ghost Cell method

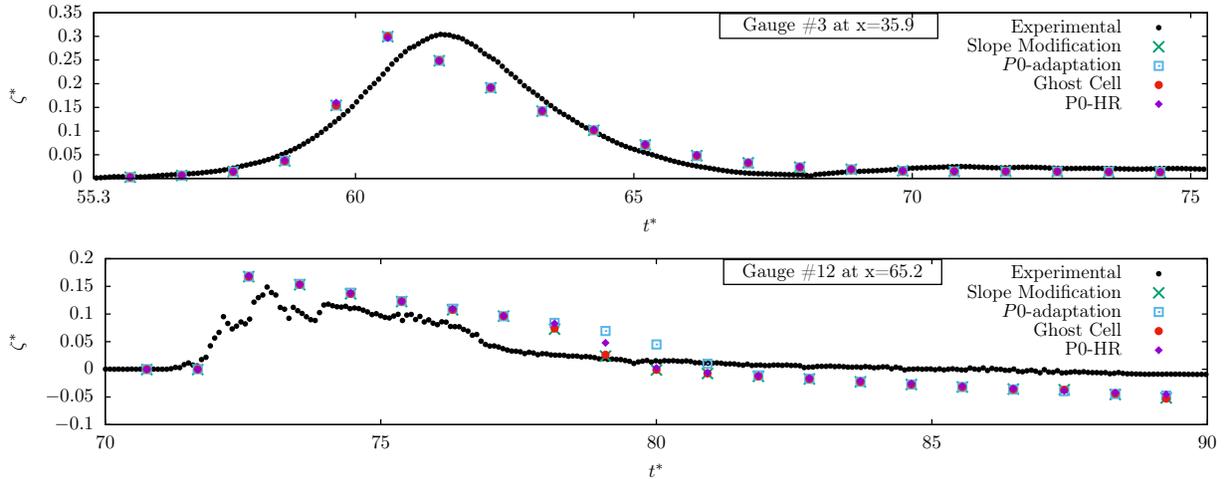


FIGURE 13. Roeber test case, time series of the normalized free surface at the wave gauges number 3 and 11 with $\Delta x = 0.4185m$, $p = 1$, SSP-RK method of order 2 and with the Ghost Cell method

of the two-dimensional DG solver to handle wetting and drying. The bathymetry is given by:

$$z_b(x, y) = \alpha r^2$$

with $r = \sqrt{x^2 + y^2}$ and α a given positive constant. At the initial time the water is also is a parabolic shape with a zero velocity. The exact solution is given by:

$$\zeta(r, t) = \frac{1}{A + B \cos(\omega t)} + \alpha(B^2 - A^2) \frac{r^2}{(A + B \cos(\omega t))^2},$$

and

$$\begin{pmatrix} u \\ v \end{pmatrix} (x, y, t) = -\frac{B\omega \sin(\omega t)}{A + B \cos(\omega t)} \begin{pmatrix} A/2 \\ B/2 \end{pmatrix}$$

where $\omega^2 = 8g\alpha$ and A and B are given constant such that $A > 0$ and $|B| < A$.

For this test case, we are using the same parameters set by Ern in 2007 [20]. Our constants are set to $\alpha = 1.6 \cdot 10^{-7} m^{-1}$, $A = 1 m^{-1}$, and $B = -0.41884 m^{-1}$. The computational domain is a square with a side length of $8000m$, which is denoted as $\Omega = [-4000, 4000] \times [-4000, 4000]$. The solution is considered piecewise linear ($p = 1$) and a SSP-RK of order 2 is used. In this method, numerical parameters are set to $\alpha_{\text{CFL}} = 0.9$ and $h_{\text{dry}} = 10^{-6}$. Four different meshes are considered, each composed of quadrilaterals of the same size. The size of the elements are $\Delta x = \Delta y \{100, 200, 400, 800\}$. Finally, the Slope Modification is used for the drying treatment.

Figure 14 shows the free surface elevation at different times. The L^2 -error is also displayed, with the color of the surface elevation indicating the error. These results are extracted from the computation with a grid size of $\Delta x = 100m$. It can be noticed that the error on ζ is always below 10^{-3} across the the selected times.

In Table 3, we can observe the L^2 -error values for h , q_x , and q_y at various times. These times have been carefully selected to showcase the variations in the drying treatment behavior during both the run-up and run-down phases. The time $t = 6/4\tau$ is the moment when the water is at the bottom of its course, just before it moves up, while $t = 2\tau$ refers to when the water is at the top of its course, just before it moves down. Additionally, $t = 5/4\tau$ is during the run-up phase while $t = 7/4\tau$ is during the run-down phase. Notably, during the run-up phase, there is no difference in convergence rate for h , but the DG solver performs better for the discharge. The most significant difference is between $t = 6/4\tau$ and $t = 2\tau$. As expected, the convergence rates are good at $t = 6/4\tau$ just after the run-up phase but are significantly degraded after the run-down phase. Indeed, this is due to the fact that the solution is not regular at the wet boundary, as displayed on Figure 15.

Remark 1. *Far from the dry and flood areas, in the case of smooth solution, classical space and time convergence rates are recovered, namely p if p is even and $p+1$ otherwise, using the appropriate time integration, as in [20].*

This test case confirms observations done on one-dimensional test cases. The run-down phase is more challenging to solve than the run-up phase. The DG solver converges better during the run-up phase than during the run-down phase. Moreover, the DG solver cannot predict the fluid speed u during the run-down phase.

6.5. Swingler and Lynett test case. Swingler and Lynett [36] performed laboratory experiments in the O.H. Hinsdale Wave Research Laboratory of Oregon State University, to study specific phenomena, that are known to occur when solitary waves approach a shoreline, such as shoaling, refraction and breaking for model validation [46, 48]. The flume is $48.8m$ long and $26.5m$ wide with a water depth of $0.78m$. A three dimensional bathymetry is built in this flume. It is composed of a triangular shelf with the apex at $x = 12.6m$ and centered on the middle of the flume. The shelf has a slight positive slope until $x = 25.5m$. On the shoreward on the shelf the bathymetry contains a beach of a $1 : 30$ slope until $x = 31m$. After the beach slope, the beach is flat until the end of the flume. The beach slope intersects the free surface level at $x = 25.75m$. On the offshore side of the triangular shelf, the bathymetry bluids up at $x = 10.2m$ to reach the shelf altitude. It gives a varying slope from the side of the flume to the apex af the shelf. The slope becomes steeper as

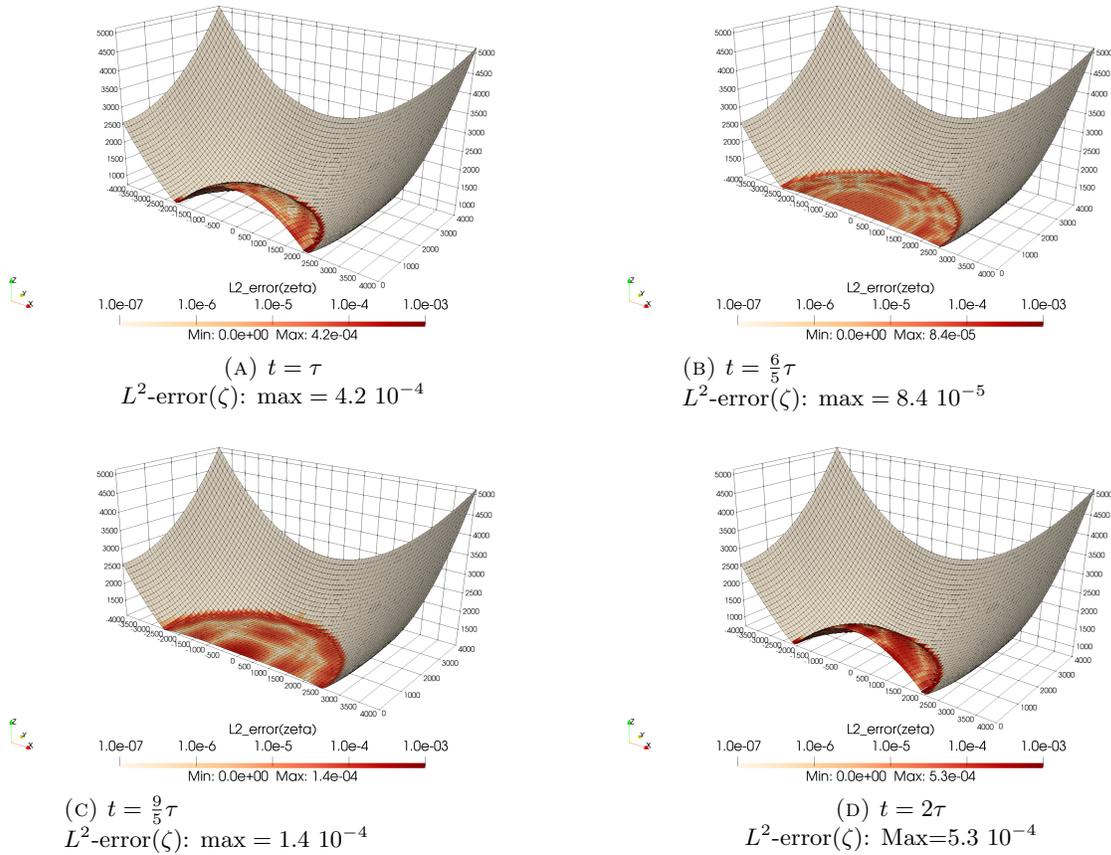


FIGURE 14. Two-dimensional parabolic bowl, free surface elevation with $L^2\text{-error}(\zeta)$ coloring with $\Delta x = 100m$, $p = 1$, SSP-RK method of order 2 and with Slope Modification

the distance from the apex decreases. Finally, on the triangular shelf, a conical island is centered at $x = 17m$ and $y = 0m$. The island apex is $0.45m$ above the free surface level and the basis radius is $3m$ long.

Moreover, for the experiments wave and advection gauges were placed in the flume. Wave gauges WG1, WG2, WG3 and WG7 are placed on the centerline of the flume and respectively at $x = 7.5m$, $13m$, $21m$ and $25.5m$. WG4, WG5 and WG6 are placed at $y = 5m$ and respectively at $x = 7.5m$, $13m$ and $21m$. WG8 and WG9 are placed at $x = 25m$ and respectively at $y = 5m$ and $10m$. Advection gauges ADV1 and ADV2 are placed at $y = 0m$ and respectively at $x = 13m$ and $21m$. The last advection gauge ADV3 is placed at $x = 21m$ and $y = -5m$.

The problem is a two-dimensional problem when solved with SWE. Hence, the computational domain is $\Omega = [-5, 48.8] \times [-13.25, 13.25]$. The computational time is $T = 50s$. The initial condition is a simulated solitary wave, which is a first-order solution of the Boussinesq equation centered on $x_0 = 5m$. The wave has a dimensionless amplitude of $A/h_0 = 0.5$, with $h_0 = 0.78m$. It is given by :

$$\zeta(x) = \frac{A}{\cosh\left(\sqrt{\frac{3A}{4h_0^3}}(x - x_0)\right)^2} \quad \text{and} \quad q(x) = \sqrt{g(h_0 + A)}\zeta(x).$$

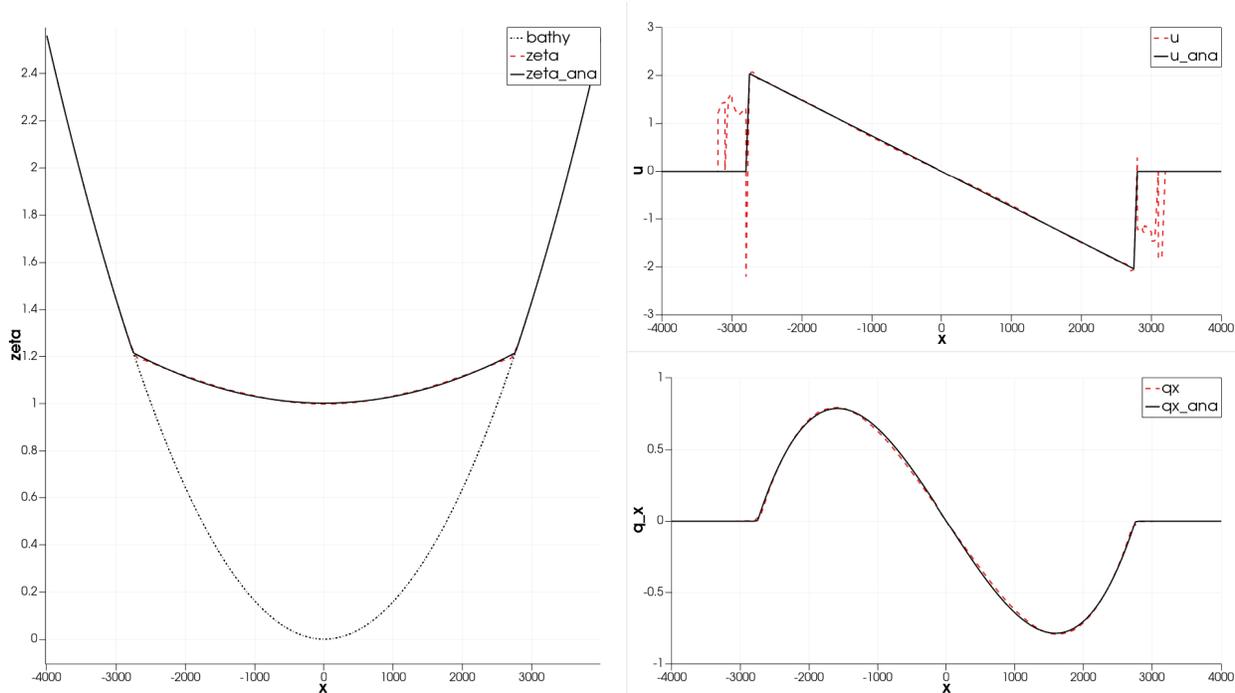


FIGURE 15. Two-dimensional parabolic bowl, cross section of the free surface elevation (left), the fluid speed (top right) and discharge (bottom right) at $t = \frac{7}{4}\tau$ with $\Delta x = 100m$, $p = 1$, SSP-RK method of order 2 and with Slope Modification

For the numerical simulation, solid wall boundary conditions are applied at each side of the domain. The mesh is composed of 60×40 quadrilaterals blocs along respectively the x and y axis. The mesh can be refined up to a level 3. The refinement criteria is based of the gradient of the free surface and thresholds are fixed to $\beta_c = \beta_r = 0.01$. The solution is considered piecewise linear ($p = 1$) and a SSP-RK of order 2 is used. The DG method is used to solve the SWE and the Slope Modification drying treatment is used with $\alpha_{CFL} = 0.9$ and $h_{dry} = 10^{-4}$. Moreover the fluid speed on the shoreline is bounded between $-100m/s$ and $100m/s$.

The water surfaces at various times are displayed in Figure 16. At the beginning, the wavefront steepens as the solitary wave approaches. By $t = 6.5s$, the surge resulting from this completely overtops the cone, while the wave along the basin's sides continues to shoal. By $t = 8.5s$, the refracted and diffracted waves collide on the shelf. Around $t = 11s$, the water withdraws from the cone, and the bore-front from the diffracted wave moves onshore and reinforces the refracted waves from the reef edge. As time passes, the water advances up the slope and reaches the flat area. At around $t = 14.5s$, a new bore is created from the draw-down of the water on the slope and collides with the refracted waves, while water propagates on the flat area.

In Figure 17, we observe the recorded surface elevation measurements at WG1, 3 and 8 over time. WG1 indicates an accurate prediction of the wave's arrival and height on the centerline of the domain and before the crest. However, the model successfully computes the cone overtop with a slight delay and without over-predicting the wave height, leading to a collision of the refracted and diffracted waves, as seen in the results at WG3 around time $t = 8s$. On the north side of the shelf WG8 shows that the numerical model reasonably predicts wave shoaling, refraction, and breaking. After $t = 40s$ the numerical results begin to deviate from the measurements due to the late arrival

TABLE 3. L^2 -error and convergence rates for the two-dimensional parabolic bowl at different times.

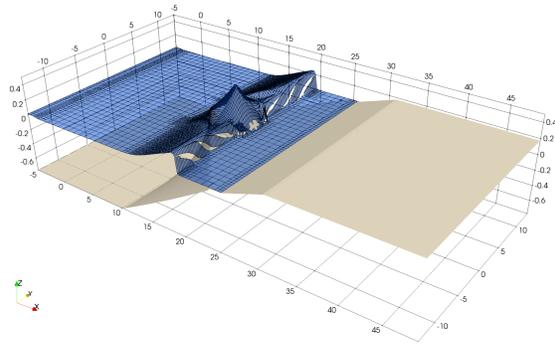
t	Δx	h		q_x		u	
		L^2 -error	r	L^2 -error	r	L^2 -error	r
5/4 τ	800	3.43 10^{-1}		1.78		1.79	
–	400	2.68 10^{-1}	0.36	9.33 10^{-1}	0.93	9.31 10^{-1}	0.94
–	200	2.01 10^{-1}	0.41	5.10 10^{-1}	0.87	5.09 10^{-1}	0.87
–	100	1.47 10^{-1}	0.46	3.23 10^{-1}	0.66	3.23 10^{-1}	0.66
–	Fitted	—	0.41	—	0.83	—	0.83
6/4 τ	800	7.65 10^{-1}		8.48 10^{-1}		8.23 10^{-1}	
–	400	3.73 10^{-1}	1.04	3.91 10^{-1}	1.12	3.26 10^{-1}	1.34
–	200	1.94 10^{-1}	0.94	1.47 10^{-1}	1.41	1.46 10^{-1}	1.16
–	100	1.53 10^{-1}	0.34	1.36 10^{-1}	0.11	1.36 10^{-1}	0.11
–	Fitted	—	0.79	—	0.93	—	0.90
7/4 τ	800	4.24 10^{-1}		1.63		1.66	
–	400	5.15 10^{-1}	–.28	8.38 10^{-1}	0.96	8.60 10^{-1}	0.95
–	200	2.68 10^{-1}	0.94	4.49 10^{-1}	0.90	4.60 10^{-1}	0.90
–	100	1.90 10^{-1}	0.49	4.19 10^{-1}	0.10	4.19 10^{-1}	0.13
–	Fitted	—	0.44	—	0.68	—	0.69
2 τ	800	1.39		2.86 10^{-1}		2.83 10^{-1}	
–	400	9.46 10^{-1}	0.55	9.59 10^{-1}	–1.74	1.12	–1.99
–	200	4.98 10^{-1}	0.93	5.46 10^{-1}	0.81	5.26 10^{-1}	1.09
–	100	3.76 10^{-1}	0.41	3.43 10^{-1}	0.67	3.44 10^{-1}	0.61
–	Fitted	—	0.66	—	0.00	—	0.02

of the numerical reflected waves from the extended computational domain. Consequently, time series are not displayed after $t = 40s$.

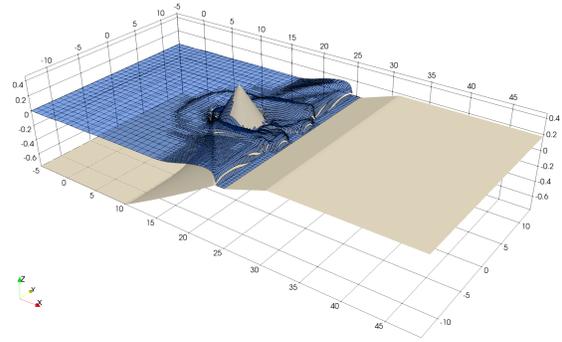
In Figure 18, the velocity time series measurements are compared with the numerical ones at various advection gauges locations. The SWE model performs reasonably matching the u velocities by accurately predicting the peak velocities and entire time trend in the u profiles. However, in ADV1, there are some u overshoots at $t = 6s$ and $t = 14s$, but fortunately, they don't affect the fluid speed time series since it matches experimental data after the event. Moreover, the numerical model accurately predicts the production of v at the ADV3 location.

7. CONCLUSIONS

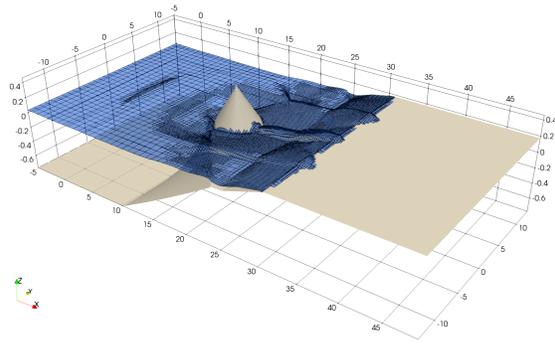
In this paper, Shallow Water Equations with dry areas and bathymetry are numerically solved using a Rung-Kutta Discontinuous Galerkin method modified to be well-balanced and Total Variations Diminishing in the Means. Nevertheless, the ability to solve problems with dry areas is not straightforward for such a method. Thus, three ways to treat flooding and drying areas are introduced and implemented. The first is based on Slope Modification, the second is based on p-adaptation, and the last, the least common, is at the interface between eXtended Finite Element and mesh adaptation. These positivity-preserving methods are compared on one-dimensional cases through numerical and experimental benchmarks, and a selected one is extended to two-dimensional problems. It can be concluded using numerical results that the Ghost Cell method performs the best, but its two-dimensional extension is not straightforward. \mathbb{P}^0 -adaptation performs the worst but is easy to implement. Lastly, Slope Modification performs well and can be easily applied to two-dimensional problems. Its performance is suitable for solving problems involving a moving



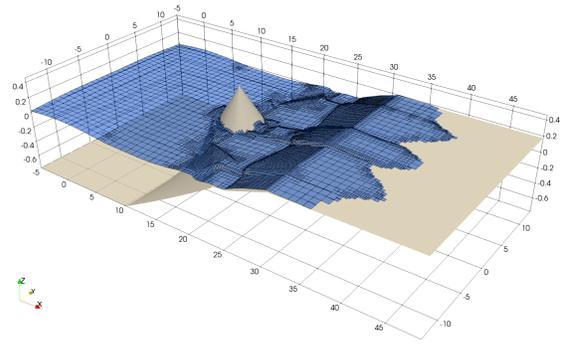
(A) $t = 6.5s$ and with 9345 elements



(B) $t = 8.5s$ and with 10194 elements



(C) $t = 11s$ and with 10968 elements



(D) $t = 14.5s$ and with 8877 elements

FIGURE 16. Swingler and Lynett test case, free surface representation at different times with $p = 1$, SSP-RK method of order 2 and with Slope Modification

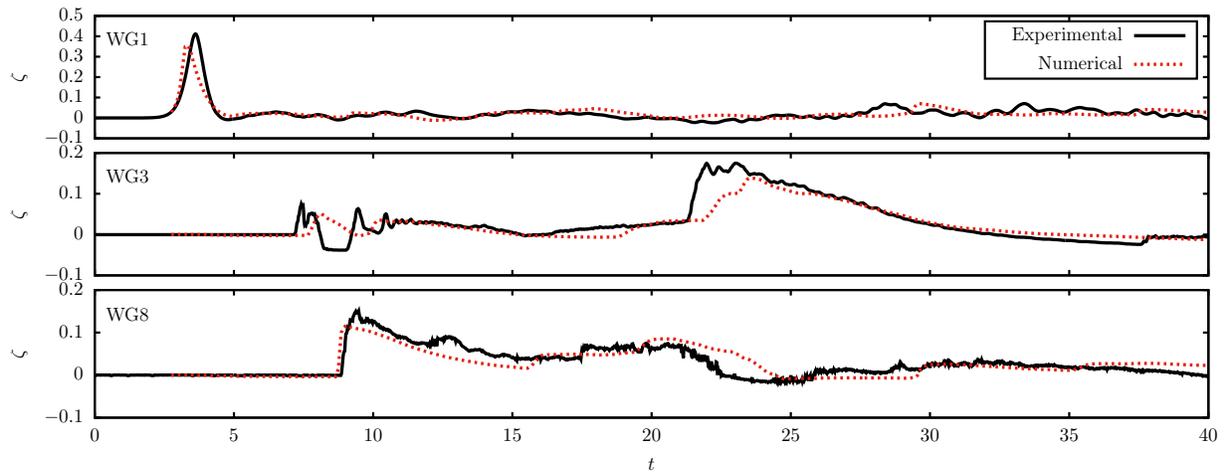


FIGURE 17. Swingler and Lynett test case, time series of the free surface compared to experimental data (WGs) with $p = 1$, SSP-RK method of order 2 and with Slope Modification

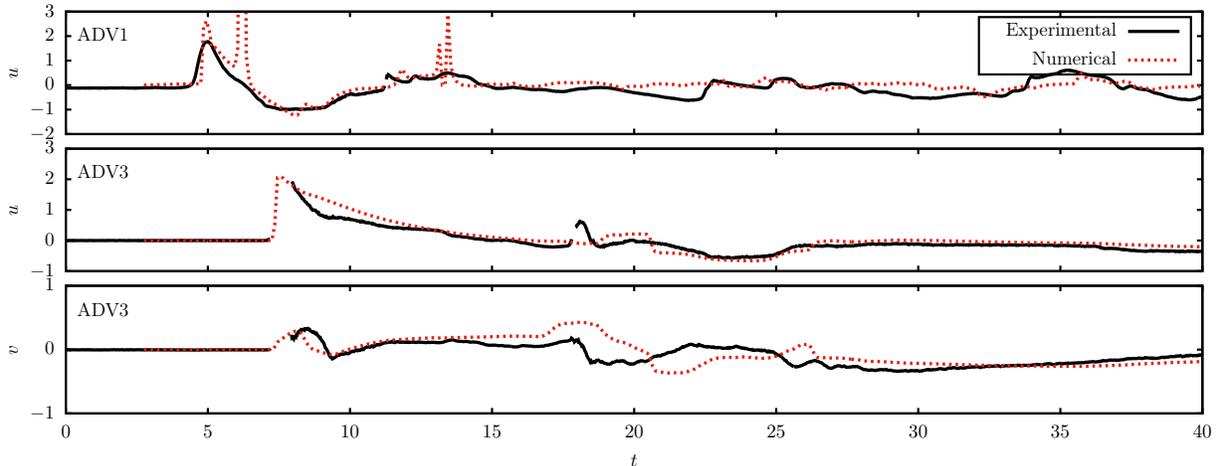


FIGURE 18. Swingler and Lynett test case, time series of the fluid speed compared to experimental data (ADVs) with $p = 1$, SSP-RK method of order 2 and with Slope Modification

shoreline in two dimensions. Based on its performance, the Ghost Cell method should be extended in two dimensions and compared to the Slope Modification method.

ACKNOWLEDGMENTS

This work has been supported financially by the région Provence-Alpes-Côte d’Azur, France (Emploi Jeunes Dctorants).

REFERENCES

- [1] T. Altazin, M. Ersoy, F. Golay, D. Sous, and L. Yushchenko. Numerical investigation of BB-AMR scheme using entropy production as refinement criterion. *International Journal of Computational Fluid Dynamics*, 30(3):256–271, Mar. 2016.
- [2] A. Balzano. Evaluation of methods for numerical simulation of wetting and drying in shallow water flow models. *Coastal Engineering*, 34(1):83–107, July 1998.
- [3] P. D. Bates and J.-M. Hervouet. A new method for moving–boundary hydrodynamic problems in shallow water. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 455(1988):3107–3128, Aug. 1999.
- [4] T. Belytschko and T. Black. Elastic crack growth in finite elements with minimal remeshing. *International Journal for Numerical Methods in Engineering*, 45(5):601–620, 1999.
- [5] O. Bokhove. Flooding and Drying in Discontinuous Galerkin Finite-Element Discretizations of Shallow-Water Equations. Part 1: One Dimension. *Journal of Scientific Computing*, 22(1):47–82, June 2005.
- [6] S. Bradford and B. Sanders. Finite-Volume Model for Shallow-Water Flooding of Arbitrary Topography. *Journal of Hydraulic Engineering-asce - J HYDRAUL ENG-ASCE*, 128, Mar. 2002.
- [7] M. Bristeau and B. Perthame. Kinetic schemes for solving Saint-Venant equations on unstructured grids. In *Innovative Methods for Numerical Solution of Partial Differential Equations*, pages 267–277. WORLD SCIENTIFIC, Dec. 2001.
- [8] S. Bunya, E. J. Kubatko, J. J. Westerink, and C. Dawson. A wetting and drying treatment for the Runge–Kutta discontinuous Galerkin solution to the shallow water equations. *Computer Methods in Applied Mechanics and Engineering*, 198(17-20):1548–1562, Apr. 2009.
- [9] G. F. Carrier and H. P. Greenspan. Water waves of finite amplitude on a sloping beach. *Journal of Fluid Mechanics*, 4(1):97–109, May 1958.
- [10] N. Chalmers and L. Krivodonova. A robust CFL condition for the discontinuous Galerkin method on triangular meshes. *Journal of Computational Physics*, 403:109095, Feb. 2020.

- [11] J.-B. Clément, F. Golay, M. Ersoy, and D. Sous. An adaptive strategy for discontinuous Galerkin simulations of Richards' equation: Application to multi-materials dam wetting. *Advances in Water Resources*, 151:103897, 2021.
- [12] B. Cockburn, S.-Y. Lin, and C.-W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: One-dimensional systems. *Journal of Computational Physics*, 84(1):90–113, Sept. 1989.
- [13] B. Cockburn and C.-W. Shu. The Runge-Kutta local projection P^1 -discontinuous-Galerkin finite element method for scalar conservation laws. *ESAIM: Mathematical Modelling and Numerical Analysis*, 25(3):337–361, 1991.
- [14] B. Cockburn and C.-W. Shu. Runge–Kutta Discontinuous Galerkin Methods for Convection-Dominated Problems. *Journal of Scientific Computing*, 16:173–261, Sept. 2001.
- [15] T. Coupez and E. Hachem. Solution of high-Reynolds incompressible flow with stabilized finite element and adaptive anisotropic meshing. *Computer Methods in Applied Mechanics and Engineering*, 267:65–85, Dec. 2013.
- [16] A. de Saint-Venant. *Théorie Du Mouvement Non Permanent Des Eaux, Avec Application Aux Crues Des Rivières et à l'introduction Des Marées Dans Leur Lit*. Comptes Rendus Hebdomadaires Des Séances de l'Académie Des Sciences. Gauthier-Villars, 1871.
- [17] A. Duran. *Numerical Simulation of Depth-Averaged Flow Models : A Class of Finite Volume and Discontinuous Galerkin Approaches*. PhD thesis, Université Montpellier II, 30, Faculté des Sciences de Montpellier, Place E. Bataillon, 34095 Montpellier, Oct. 2014.
- [18] A. Duran and F. Marche. Recent advances on the discontinuous Galerkin method for shallow water equations with topography source terms. *Computers and Fluids*, 101:88–104, June 2014.
- [19] K. Dutt and L. Krivodonova. A high-order moment limiter for the discontinuous Galerkin method on triangular meshes. *Journal of Computational Physics*, 433:110188, May 2021.
- [20] A. Ern, S. Piperno, and K. Djadel. A well-balanced Runge–Kutta discontinuous Galerkin method for the shallow-water equations with flooding and drying. *International Journal for Numerical Methods in Fluids*, 58(1):1–25, 2007.
- [21] M. Ersoy, F. Golay, and L. Yushchenko. Adaptive multiscale scheme based on numerical density of entropy production for conservation laws. *Open Mathematics*, 11(8), Jan. 2013.
- [22] C. Eskilsson. An hp-adaptive discontinuous Galerkin method for shallow water flows. *International Journal for Numerical Methods in Fluids*, 67(11):1605–1623, 2011.
- [23] T.-P. Fries and T. Belytschko. The extended/generalized finite element method: An overview of the method and its applications. *International Journal for Numerical Methods in Engineering*, 84(3):253–304, 2010.
- [24] T. Gallouët, J.-M. Hérard, and N. Seguin. Some approximate Godunov schemes to compute shallow-water equations with topography. *Computers and Fluids*, 32(4):479–513, 2003.
- [25] J.-F. Gerbeau and a. B. Perthame. Derivation of viscous Saint-Venant system for laminar shallow water; Numerical validation. *Discrete and Continuous Dynamical Systems - B*, 1(1):89–102, 2001.
- [26] E. Godlewski and P.-A. Raviart. *Numerical Approximation of Hyperbolic Systems of Conservation Laws*. Springer New York, 1996.
- [27] S. K. Godunov and I. Bohachevsky. Finite difference method for numerical computation of discontinuous solutions of the equations of fluid dynamics. *Matematičeskij sbornik*, 47(89)(3):271–306, 1959.
- [28] F. Golay, M. Ersoy, L. Yushchenko, and D. Sous. Block-based adaptive mesh refinement scheme using numerical density of entropy production for three-dimensional two-fluid flows. *International Journal of Computational Fluid Dynamics*, 29(1):67–81, Jan. 2015.
- [29] T. C. Gopalakrishnan. A moving boundary circulation model for regions with large tidal flats. *International Journal for Numerical Methods in Engineering*, 28(2):245–260, 1989.
- [30] S. Gottlieb, C.-W. Shu, and E. Tadmor. Strong Stability-Preserving High-Order Time Discretization Methods. *SIAM Review*, 43(1):89–112, 2001.
- [31] A. Harten, P. D. Lax, and B. V. Leer. On Upstream Differencing and Godunov-Type Schemes for Hyperbolic Conservation Laws. *SIAM Review*, 25(1):35–61, 1983.
- [32] B. L. Karihaloo and Q. Z. Xiao. Modelling of stationary and growing cracks in FE framework without remeshing: A state-of-the-art review. *Computers & Structures*, 81(3):119–129, Feb. 2003.
- [33] M. Kazolea, A. I. Delis, and C. E. Synolakis. Numerical treatment of wave breaking on unstructured finite volume approximations for extended Boussinesq-type equations. *Journal of Computational Physics*, 271:281–305, Aug. 2014.
- [34] L. Krivodonova. Limiters for high-order discontinuous Galerkin methods. *Journal of Computational Physics*, 226(1):879–896, Sept. 2007.

- [35] E. J. Kubatko, S. Bunya, C. Dawson, and J. J. Westerink. Dynamic p-adaptive Runge–Kutta discontinuous Galerkin methods for the shallow water equations. *Computer Methods in Applied Mechanics and Engineering*, 198(21):1766–1774, May 2009.
- [36] P. J. Lynett, D. Swigler, S. Son, D. Bryant, and S. Socolofsky. EXPERIMENTAL STUDY OF SOLITARY WAVE EVOLUTION OVER A 3D SHALLOW SHELF. *Coastal Engineering Proceedings*, (32):1–1, 2010.
- [37] F. Marche. Derivation of a new two-dimensional viscous shallow water model with varying topography, bottom friction and capillary effects. *European Journal of Mechanics - B/Fluids*, 26(1):49–63, Jan. 2007.
- [38] N. Moës and T. Belytschko. X-FEM, de nouvelles frontières pour les éléments finis. In *5e Colloque National En Calcul Des Structures*, volume 11, pages 305–318, Giens, France, May 2001. CSMA.
- [39] N. Moës, J. Dolbow, and T. Belytschko. A finite element method for crack growth without remeshing. *International Journal for Numerical Methods in Engineering*, 46(1):131–150, 1999.
- [40] P. Mossier, A. Beck, and C.-D. Munz. A p-Adaptive Discontinuous Galerkin Method with hp-Shock Capturing. *Journal of Scientific Computing*, 91(1):4, Feb. 2022.
- [41] C. Nielsen and C. Apelt. Parameters Affecting the Performance of Wetting and Drying in a Two-Dimensional Finite Element Long Wave Hydrodynamic Model. *Journal of Hydraulic Engineering*, 129, Aug. 2003.
- [42] S. Osher. Convergence of Generalized MUSCL Schemes. *SIAM Journal on Numerical Analysis*, 22(5):947–961, 1985.
- [43] B. Perthame and C. Simeoni. A kinetic scheme for the Saint-Venant system with a source term. *CALCOLO*, 38(4):201–231, Dec. 2001.
- [44] D. A. D. Pietro and A. Ern. *Mathematical Aspects of Discontinuous Galerkin Methods*. Springer Berlin Heidelberg, 2012.
- [45] K. Pons. *Modélisation des tsunamis : propagation et impact*. PhD thesis, Université de Toulon, Av. de l’Université, 83130 La Garde, Dec. 2018.
- [46] V. Roeber and K. F. Cheung. Boussinesq-type model for energetic breaking waves in fringing reef environment. *Coastal Engineering*, 70:1–20, Dec. 2012.
- [47] V. Roeber, K. F. Cheung, and M. H. Kobayashi. Shock-capturing Boussinesq-type model for nearshore wave processes. *Coastal Engineering*, 57(4):407–423, Apr. 2010.
- [48] F. Shi, J. T. Kirby, J. C. Harris, J. D. Geiman, and S. T. Grilli. A high-order adaptive time-stepping TVD solver for Boussinesq modeling of breaking waves and coastal inundation. *Ocean Modelling*, 43–44:36–51, Jan. 2012.
- [49] C.-W. Shu. Total-Variation-Diminishing Time Discretizations. *SIAM Journal on Scientific and Statistical Computing*, 9(6):1073–1084, Nov. 1988.
- [50] C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of Computational Physics*, 77(2):439–471, Aug. 1988.
- [51] H. Tang and G. Warnecke. A Runge–Kutta discontinuous Galerkin method for the Euler equations. *Computers & Fluids*, 34(3):375–398, Mar. 2005.
- [52] G. Tumolo, L. Bonaventura, and M. Restelli. A semi-implicit, semi-Lagrangian, p-adaptive discontinuous Galerkin method for the shallow water equations. *Journal of Computational Physics*, 232(1):46–67, Jan. 2013.
- [53] S. Vater, N. Beisiegel, and J. Behrens. 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, Nov. 2015.
- [54] S. Vater, N. Beisiegel, and J. Behrens. A limiter-based well-balanced discontinuous Galerkin method for shallow-water flows with wetting and drying: Triangular grids. *International Journal for Numerical Methods in Fluids*, 91(8):395–418, Aug. 2019.
- [55] L. Wang and D. J. Mavriplis. Adjoint-based h–p adaptive discontinuous Galerkin methods for the 2D compressible Euler equations. *Journal of Computational Physics*, 228(20):7643–7661, Nov. 2009.
- [56] T. Weiyang, editor. *Chapter 2 Properties of the 2-D System of Shallow-Water Equations (2-D SSWE)*, volume 55 of *Shallow Water Hydrodynamics*. Elsevier, Jan. 1992.
- [57] Y. Xing and X. Zhang. Positivity-Preserving Well-Balanced Discontinuous Galerkin Methods for the Shallow Water Equations on Unstructured Triangular Meshes. *Journal of Scientific Computing*, 57(1):19–41, Oct. 2013.
- [58] Y. Xing, X. Zhang, and C.-W. Shu. Positivity-preserving high order well-balanced discontinuous Galerkin methods for the shallow water equations. *Advances in Water Resources*, 33(12):1476–1493, 2010.
- [59] X. Zhang, Y. Xia, and C.-W. Shu. Maximum-Principle-Satisfying and Positivity-Preserving High Order Discontinuous Galerkin Schemes for Conservation Laws on Triangular Meshes. *Journal of Scientific Computing*, 50(1):29–62, Jan. 2012.

IMATH, UNIVERSITÉ DE TOULON, LA GARDE 83130, FRANCE

Email address: pousselcamille@outlook.fr, ersoy@univ-tln.fr, golay@univ-tln.fr