



Positivity-Preserving Well-Balanced Arbitrary Lagrangian–Eulerian Discontinuous Galerkin Methods for the Shallow Water Equations

Weijie Zhang¹ · Yinhua Xia¹ · Yan Xu¹

Received: 5 March 2021 / Revised: 4 June 2021 / Accepted: 4 July 2021 / Published online: 21 July 2021
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract

In this paper, we develop well-balanced arbitrary Lagrangian–Eulerian discontinuous Galerkin (ALE-DG) methods for the shallow water equations, which preserve not only the still water equilibrium but also the moving water equilibrium. Based on the time-dependent linear affine mapping, the ALE-DG method for conservation laws maintains almost all mathematical properties of DG methods on static grids, such as conservation, geometric conservation law (GCL), entropy stability. The main difficulty to obtain the well-balanced property of the ALE-DG method for shallow water equations is that the grid movement and usual time discretization may destroy the equilibrium. By adopting the GCL preserving Runge–Kutta methods and the techniques of well-balanced DG schemes on static grids, we successfully construct the high order well-balanced ALE-DG schemes for the shallow water equations with still and moving water equilibria. Meanwhile, the ALE-DG schemes can also preserve the positivity property at the same time. Numerical experiments in different circumstances are provided to illustrate the well-balanced property, positivity preservation and high order accuracy of these schemes.

Keywords Shallow water equations · Arbitrary Lagrangian–Eulerian discontinuous Galerkin method · Geometric conservation law · Still and moving water equilibria · Well-balanced scheme · Positivity preservation

W. Zhang: Research supported by National Numerical Windtunnel Project NNW2019ZT4-B08; Y. Xia: Research supported by Science Challenge Project TZT2019-A2.3, NSFC Grant No. 11871449; Y. Xu: Research supported by National Numerical Windtunnel Project NNW2019ZT4-B08, NSFC Grant No. 12071455, 11722112.

✉ Yan Xu
yxu@ustc.edu.cn

Weijie Zhang
zhwj@mail.ustc.edu.cn

Yinhua Xia
yhxia@ustc.edu.cn

¹ School of Mathematical Sciences, University of Science and Technology of China, Hefei 230026, Anhui, People's Republic of China

1 Introduction

The shallow water equations have been widely used to model flows in the atmosphere, rivers, lakes and oceans as well as gravity waves in a smaller domain. They are especially suitable to model tides which have very large length scales. In one space dimension, the shallow water equations take the form:

$$\begin{cases} h_t + (hu)_x = 0, \\ (hu)_t + (hu^2 + \frac{1}{2}gh^2)_x = -ghb_x, \end{cases} \quad (1.1)$$

where h denotes the water height, u is the velocity of the fluid, b is the bottom topography and g is the gravitational constant. The shallow water equations exactly preserve the still water equilibrium:

$$u = 0 \quad \text{and} \quad h + b = \text{constant}, \quad (1.2)$$

which represents a still flat water surface. The shallow water equations also preserve the general moving water equilibrium:

$$m := hu = \text{constant} \quad \text{and} \quad E := \frac{1}{2}u^2 + g(h + b) = \text{constant}, \quad (1.3)$$

where m , E are the moving water equilibrium variables. The still water equilibrium is just a special case of the moving water equilibrium when the velocity equals zero.

The well-balanced schemes are introduced to preserve exactly, at a discrete level, these equilibrium solutions. An important advantage of the well-balanced schemes is that they can accurately resolve small perturbations to such steady state solutions with relatively coarse meshes. Research on well-balanced numerical methods for the shallow water system has attracted many attentions in the past two decades. Researchers have developed many well-balanced schemes for the shallow water equations to preserve the still water equilibrium, e.g. [3,25,31,34,36,38–40]. Most of the well-balanced methods which preserve the still water equilibrium cannot preserve the moving water equilibrium, and it is significantly more difficult to obtain well-balanced schemes for the moving water equilibrium. Some well-balanced schemes for the moving water equilibrium have been developed, see [8,10,21,27,30,31]. In a recent paper [35], Xing designed a well-balanced discontinuous Galerkin (DG) scheme for arbitrary equilibrium of the shallow water equations. The key idea of this paper is a special way to recover the moving water equilibrium. Another important difficulty in solving the shallow water equations is the appearance of dry areas. The water height may become negative near the wet-dry front and may cause the breakdown of numerical simulations. There are some existing work to overcome this difficulty. In [6,7], the authors cut off the outgoing numerical flux without reducing the global time step to maintain the positivity of the water height. In [11], a hydrostatic reconstruction scheme based on subcell reconstruction is introduced. Many other techniques can be found in [4,9,26]. A very popular technique for high order methods is a simple scaling positivity-preserving limiter, which can preserve the high order accuracy without losing local conservation [40].

The well-balanced schemes for shallow water equations were developed for static computational meshes, but in many applications, numerical methods with a deformable moving mesh are desirable for flexibility. In recent years, well-balanced moving mesh methods have been designed within several different frameworks, which preserve the still water equilibrium. In [44,45], the authors introduced a new interpolation in finite volume methods to remap the flow variables to the new mesh while keeping the well-balanced property and

height positive. In [1,2], finite volume methods has been coupled pre-balanced ALE form of shallow water equations. In [28], space-time discontinuous DG methods on moving meshes and a new weak form were developed to preserve the steady states. A two-step moving mesh DG scheme was presented in [42], where the well-balanced DG methods with hydrostatic reconstruction on static grids and a remapping were coupled. We will choose the arbitrary Lagrangian–Eulerian discontinuous Galerkin (ALE-DG) method developed by Klingenberg et al. [19,23] in this paper, which maintains mathematical properties of the DG methods on static grids, such as conservation, geometric conservation law (GCL), entropy stability and high order accuracy. Thereinto, the GCL property of the grid deformation method is essential for the development of well-balanced grid deformation schemes for the shallow water equations. It has been shown in [19] that the GCL property of the ALE-DG method can be satisfied for any time discretization methods with the order greater than or equal to the dimension. Thus the modified total variation diminishing Runge–Kutta (TVD-RK) methods has been introduced to guarantee the GCL property in each stage of Runge–Kutta methods for two or higher dimension cases. Besides, by adopting the time-dependent linear affine mapping, the ALE-DG algorithm requires only very mild Lipschitz continuity of the mesh movement function. Moreover, due to the GCL property, We can keep the water height average positive under suitable time steps, which is consistent with the one in DG methods [40] on static grids.

The DG method is a popular high order method for solving hyperbolic conservation laws. It has been developed and analyzed by Cockburn, Shu et al. in a series of papers, e.g. [12–16] and in a review article [33]. Researchers have developed many well-balanced DG schemes for the shallow water equations, e.g. [17,18,22,28,35,39,41], which can be classified as the hydrostatic reconstruction and special source term treatment schemes roughly. But well-balanced DG methods cannot be generalized to ALE-DG methods directly, due to the mesh movement and the time-dependent approximation space. We find that the GCL preserving fully discrete ALE-DG scheme with the modified TVD-RK methods can be adopted to develop high order accurate well-balanced schemes, also based on the techniques of well-balanced schemes on static grids. The main objective of this paper is to develop high-order accurate positivity-preserving well-balanced ALE-DG methods for the shallow water equations. We will describe the algorithms of the methods preserving the still and moving water equilibria and prove the well-balanced property of the schemes. We mainly concentrate on the formulation and show the feasibility of using ALE-DG methods for solving shallow water equations. We remark that the methods of the grid movement is not investigated in this work. The grid movement is prescribed explicitly and not derived from the computed solution in the numerical examples.

The rest of the paper is organized as follows. In Sect. 2, we give some important notations and properties for the ALE-DG method. In Sect. 3, we present the well-balanced ALE-DG methods for the shallow water equations which preserve the still water equilibrium in two space dimensions by adopting the modified TVD-RK time discretization. One-dimensional case is similar. In Sect. 4, we develop the well-balanced ALE-DG scheme for moving water equilibrium. In Sect. 5, we provide numerical examples both for still water equilibrium and moving water equilibrium to verify the well-balanced property, the positivity preserving property and high order accuracy. Finally, we will give some concluding remarks in Sect. 6. All the technical proof details in this paper are shown in “Appendices A–B”.

2 Notations

In this section, we present the time-dependent cells and finite element spaces in one and two dimensions respectively.

2.1 Mesh Grids for 1D and Notations

We discretize the computational space domain Ω into time-dependent cells. We assume that there are given points $\left\{x_{j-\frac{1}{2}}^n\right\}_{j=1}^{N+1}$ at time level t^n and $\left\{x_{j-\frac{1}{2}}^{n+1}\right\}_{j=1}^{N+1}$ at t^{n+1} , such that

$$\Omega = \bigcup_{j=1}^N \left[x_{j-\frac{1}{2}}^n, x_{j+\frac{1}{2}}^n \right], \quad \Omega = \bigcup_{j=1}^N \left[x_{j-\frac{1}{2}}^{n+1}, x_{j+\frac{1}{2}}^{n+1} \right].$$

Then we define straight lines connecting $x_{j-\frac{1}{2}}^n$ and $x_{j-\frac{1}{2}}^{n+1}$

$$x_{j-\frac{1}{2}}(t) := x_{j-\frac{1}{2}}^n + \omega_{j-\frac{1}{2}}(t - t^n), \text{ for } t \in [t^n, t^{n+1}], \tag{2.1}$$

where

$$\omega_{j-\frac{1}{2}} := \frac{x_{j-\frac{1}{2}}^{n+1} - x_{j-\frac{1}{2}}^n}{t^{n+1} - t^n}. \tag{2.2}$$

The straight lines (2.1) provide time-dependent cells $K_j(t) = \left[x_{j-\frac{1}{2}}(t), x_{j+\frac{1}{2}}(t) \right]$ at any time $t \in [t^n, t^{n+1}]$. The size of the j -th cell is denoted by

$$\Delta_j(t) := x_{j+\frac{1}{2}}(t) - x_{j-\frac{1}{2}}(t). \tag{2.3}$$

For any time $t \in [t^n, t^{n+1}]$ and cell $K_j(t)$, we define the grid velocity field ω

$$\omega(x, t) = \omega_{j+\frac{1}{2}} \frac{x - x_{j-\frac{1}{2}}(t)}{\Delta_j(t)} + \omega_{j-\frac{1}{2}} \frac{x_{j+\frac{1}{2}}(t) - x}{\Delta_j(t)}. \tag{2.4}$$

We define the following finite dimensional test function space

$$\mathcal{V}_h(t) := \left\{ \psi \in L^2(\Omega) \mid \psi|_{K_j(t)} \in P^k(K_j(t)), \forall 1 \leq j \leq N \right\}, \tag{2.5}$$

where k denotes the polynomial degree and

$$\mathbf{\Pi}_h(t) := \left\{ (\zeta, \psi)^T \mid \zeta, \psi \in \mathcal{V}_h(t) \right\}. \tag{2.6}$$

For $\psi \in \mathcal{V}_h(t)$, the values at the cell boundaries of $x_{j+\frac{1}{2}}$ is defined by

$$\psi_{j+\frac{1}{2}}^- := \lim_{\epsilon \rightarrow 0^+} \psi(x - \epsilon), \quad \psi_{j+\frac{1}{2}}^+ := \lim_{\epsilon \rightarrow 0^+} \psi(x + \epsilon). \tag{2.7}$$

2.2 Mesh Grids for 2D and Notations

Similarly, we present the time-dependent cells in two dimensions. We assume that there exist triangle meshes $\mathcal{T}^n = \bigcup_{j=1}^N \mathcal{K}_j^n$ at time level t^n and $\mathcal{T}^{n+1} = \bigcup_{j=1}^N \mathcal{K}_j^{n+1}$ at t^{n+1} which cover the convex polyhedron domain Ω such that

$$\overline{\Omega} = \bigcup_{j=1}^N \overline{\mathcal{K}_j^n} \quad \text{and} \quad \overline{\Omega} = \bigcup_{j=1}^N \overline{\mathcal{K}_j^{n+1}},$$

and $\{x_j^n\}_{j=1}^M$ and $\{x_j^{n+1}\}_{j=1}^M$ are the mesh grids at time t^n and t^{n+1} .

We define the straight lines between the points \mathbf{x}_j^n and \mathbf{x}_j^{n+1} :

$$\mathbf{x}_j(t) := \mathbf{x}_j^n + (t - t^n)\boldsymbol{\omega}_j^n, \quad \boldsymbol{\omega}_j^n := \frac{1}{t^{n+1} - t^n}(\mathbf{x}_j^{n+1} - \mathbf{x}_j^n). \tag{2.8}$$

For each pair of triangles \mathcal{K}_j^n and \mathcal{K}_j^{n+1} , assume $\mathbf{x}_a^n, \mathbf{x}_b^n, \mathbf{x}_c^n$ are three vertices of \mathcal{K}_j^n and $\mathbf{x}_a^{n+1}, \mathbf{x}_b^{n+1}, \mathbf{x}_c^{n+1}$ are corresponding vertices of \mathcal{K}_j^{n+1} . We can define the time-dependent triangles by

$$\mathcal{K}_j(t) = \text{conv}\{\mathbf{x}_a(t), \mathbf{x}_b(t), \mathbf{x}_c(t)\}, \tag{2.9}$$

where $\text{conv}(\cdot)$ denote the convex hull of a set. The area of cell $\mathcal{K}_j(t)$ is denoted by $\Delta_j(t)$.

We then define the grid velocity field $\boldsymbol{\omega}(\mathbf{x}, t)$ for $t \in [t^n, t^{n+1}]$. We assume that $\boldsymbol{\omega}$ is a vector with two components: ω_1, ω_2 . Each component is a piecewise linear polynomial, such that $\omega_1, \omega_2|_{\mathcal{K}_j(t)} \in P^1(\mathcal{K}_j(t))$. In each cell $\mathcal{K}_j(t)$, the components ω_1, ω_2 are defined as follows:

$$\begin{vmatrix} \mathbf{x} - \mathbf{x}_a(t) & \omega_1(\mathbf{x}, t) - \omega_{a,1}^n \\ \mathbf{x}_b(t) - \mathbf{x}_a(t) & \omega_{b,1}^n - \omega_{a,1}^n \\ \mathbf{x}_c(t) - \mathbf{x}_a(t) & \omega_{c,1}^n - \omega_{a,1}^n \end{vmatrix} = 0, \quad \begin{vmatrix} \mathbf{x} - \mathbf{x}_a(t) & \omega_2(\mathbf{x}, t) - \omega_{a,2}^n \\ \mathbf{x}_b(t) - \mathbf{x}_a(t) & \omega_{b,2}^n - \omega_{a,2}^n \\ \mathbf{x}_c(t) - \mathbf{x}_a(t) & \omega_{c,2}^n - \omega_{a,2}^n \end{vmatrix} = 0. \tag{2.10}$$

Here $|\cdot|$ denotes the matrix determinant. $\mathbf{x}_a^n, \mathbf{x}_b^n, \mathbf{x}_c^n$ are the vertexes of cell $\mathcal{K}_j(t)$ and $\boldsymbol{\omega}_a^n, \boldsymbol{\omega}_b^n, \boldsymbol{\omega}_c^n$ are the grid velocity of these vertexes defined in (2.8) respectively. $\omega_{i,1}^n, \omega_{i,2}^n$ are the two components of $\boldsymbol{\omega}_i^n$, for $i = a, b, c$. The equation (2.10) means that for $i = 1, 2$, the four points in three dimensional space, $(\mathbf{x}, \omega_i(\mathbf{x}, t)), (\mathbf{x}_a, \omega_{a,i}^n), (\mathbf{x}_b, \omega_{b,i}^n), (\mathbf{x}_c, \omega_{c,i}^n)$ coplanar. The two matrices in (2.10) are both 3 by 3 matrices.

We define the following finite dimensional test function space

$$\mathcal{V}_h^{2d}(t) := \left\{ \psi \in L^2(\Omega) \mid \psi|_{\mathcal{K}_j(t)} \in P^k(\mathcal{K}_j(t)), \forall 1 \leq j \leq N \right\}, \tag{2.11}$$

where k denotes the polynomial degree and

$$\boldsymbol{\Pi}_h^{2d}(t) := \left\{ (\zeta, \psi, \delta)^T \mid \zeta, \psi, \delta \in \mathcal{V}_h^{2d}(t) \right\}. \tag{2.12}$$

For $\psi \in \mathcal{V}_h^{2d}(t)$, the values on the boundary of a triangle \mathcal{K} is defined by

$$\psi^{int}(\mathbf{x}) := \lim_{\epsilon \rightarrow 0^+} \psi(\mathbf{x} - \epsilon \mathbf{n}), \quad \psi^{ext}(\mathbf{x}) := \lim_{\epsilon \rightarrow 0^+} \psi(\mathbf{x} + \epsilon \mathbf{n}), \tag{2.13}$$

where the vector \mathbf{x} is any point on the boundary of the triangle \mathcal{K} and the vector \mathbf{n} is the outer normal vector of the triangle \mathcal{K} at the point \mathbf{x} .

3 The Well-Balanced ALE-DG Methods for Still Water Equilibrium

In this section, we present the positivity-preserving well-balanced high-order ALE-DG methods for the shallow water equations which preserve the still water equilibrium. According to the previous work in [19,23], the ALE-DG methods can maintain almost all the mathematical properties of DG methods on static grids, such as conservation, geometric conservation law (GCL), entropy stability, and optimal error estimates. Our well-balanced ALE-DG methods for the still water equilibrium are mainly based on the GCL property on the moving mesh. We will develop two different well-balanced schemes and introduce their well-balanced property

and positivity-preserving property in two dimensions. The one-dimensional case is similar to the two-dimensional case.

3.1 Equations

We first write out the shallow water equations in two dimensions:

$$\begin{cases} h_t + (hu)_x + (hv)_y = 0, \\ (hu)_t + (hu^2 + \frac{1}{2}gh^2)_x + (huv)_y = -ghb_x, \\ (hv)_t + (huv)_x + (hv^2 + \frac{1}{2}gh^2)_y = -ghb_y, \end{cases} \tag{3.1}$$

where h represents the water height, (u, v) denotes the velocity vector of the fluid, and b is the bottom topography. The still water equilibrium takes the form as follows

$$u = 0, \quad v = 0, \quad h + b = \text{constant}. \tag{3.2}$$

The GCL property means that the numerical scheme can preserve constant states, which fit the still water equilibrium very well. Our motivation is to achieve the well-balanced property by using GCL. The first step is to change the variables from conservative variables

$$\mathbf{u} = (h, hu, hv)^T$$

to equilibrium variables

$$\mathbf{v} = (\eta, hu, hv)^T, \quad \text{with } \eta = h + b,$$

where η denotes the surface level. In this case, the time discretization will not destroy the well-balanced property on the moving mesh. For the ease of presentation, we rewrite the shallow water equations (3.1) as follows:

$$\mathbf{v}_t + \mathbf{f}(\mathbf{v})_x + \mathbf{h}(\mathbf{v})_y = \mathbf{s}(\mathbf{v}, b), \tag{3.3}$$

where

$$\mathbf{f}(\mathbf{v}) = \begin{pmatrix} hu \\ \frac{(hu)^2}{\eta-b} + \frac{1}{2}g(\eta-b)^2 \\ \frac{(hu)(hv)}{\eta-b} \end{pmatrix}, \quad \mathbf{h}(\mathbf{v}) = \begin{pmatrix} hv \\ \frac{(hu)(hv)}{\eta-b} \\ \frac{(hv)^2}{\eta-b} + \frac{1}{2}g(\eta-b)^2 \end{pmatrix},$$

$$\mathbf{s}(\mathbf{v}, b) = \begin{pmatrix} 0 \\ -g(\eta-b)b_x \\ -g(\eta-b)b_y \end{pmatrix}.$$

Because $b(\mathbf{x})$ is independent of t , we can easily prove that the solution for (3.1) is the solution for (3.3). The variables in \mathbf{v} are constants in the case of the still water equilibrium. We denote the numerical solution still by \mathbf{v} and numerical bottom topography by b for simplicity.

3.2 The Standard ALE-DG Scheme

3.2.1 The Semi-discrete Scheme

For the shallow water equations (3.3), the semi-discrete ALE-DG scheme is the following: Find $\mathbf{v} \in \Pi_h^{2d}(t)$, for any $\phi \in \Pi_h^{2d}(t)$,

$$\frac{d}{dt} \int_{\mathcal{K}_j(t)} \mathbf{v} \cdot \phi \, dx = \mathcal{L}_j(\omega, \mathbf{v}, \phi, t) \tag{3.4}$$

with

$$\mathcal{L}_j(\omega, \mathbf{v}, \phi, t) = \int_{\mathcal{K}_j(t)} \mathbf{G}(\omega, \mathbf{v}) : \nabla \phi \, dx - \int_{\partial \mathcal{K}_j(t)} \widehat{\mathbf{G}} \cdot \phi^{int} \, ds + \int_{\mathcal{K}_j(t)} \mathbf{s}(\mathbf{v}, b) \cdot \phi \, dx,$$

where

- The operator $:$ is defined as follows: $A : B = \sum_{i,j} a_{ij} b_{ij}$ for any matrices $A = [a_{ij}]$ and $B = [b_{ij}] \in \mathbb{R}^{n \times m}$.
- $\omega = (\omega_1, \omega_2)^T$ is the grid velocity field defined in (2.10).
- Flux term $\mathbf{G}(\omega, \mathbf{v}) = (\mathbf{f}(\mathbf{v}) - \omega_1 \mathbf{v}, \mathbf{h}(\mathbf{v}) - \omega_2 \mathbf{v})$.
- The numerical flux $\widehat{\mathbf{G}} = \widehat{\mathbf{G}}(\omega, \mathbf{v}^{int}, \mathbf{v}^{ext}, \mathbf{n})$ is a monotone flux of $\mathbf{G}(\omega, \mathbf{v})$. We adopt the Lax-Friedrichs flux [19] in this paper:

$$\widehat{\mathbf{G}}(\omega, \mathbf{v}^{int}, \mathbf{v}^{ext}, \mathbf{n}) = (\mathbf{G}(\omega, \mathbf{v}^{int}) \cdot \mathbf{n} + \mathbf{G}(\omega, \mathbf{v}^{ext}) \cdot \mathbf{n} - \alpha(\mathbf{v}^{ext} - \mathbf{v}^{int}))/2, \tag{3.5}$$

where \mathbf{n} is the outer normal vector, $\alpha = \max(|((u, v)^T - \omega) \cdot \mathbf{n}| + \sqrt{gh})$ and the maximum is taken in the relevant domain.

3.2.2 The Runge–Kutta Time Discretization

In this paper, we will adopt the total variation diminishing Runge–Kutta (TVD-RK) time discretization method [20,32]. The time discretization for moving mesh is slightly different with the static mesh, since the test functions will change over time. We first introduce the relationship between the test functions at different time level.

We define the following time-dependent linear mapping, for any interval $K_j(t)$:

$$\mathcal{X}_j : [0, 1]^2 \rightarrow \mathcal{K}_j(t), \quad \mathcal{X}_j(\xi, t) = \mathbf{A}_{\mathcal{K}_j(t)} \xi + \mathbf{x}_{\mathcal{K}_j(t),a}, \tag{3.6}$$

where the matrix $\mathbf{A}_{\mathcal{K}_j(t)}$ is given by

$$\mathbf{A}_{\mathcal{K}_j(t)} := (\mathbf{x}_{\mathcal{K}_j(t),b} - \mathbf{x}_{\mathcal{K}_j(t),a}, \mathbf{x}_{\mathcal{K}_j(t),c} - \mathbf{x}_{\mathcal{K}_j(t),a}), \tag{3.7}$$

and $\mathbf{x}_{\mathcal{K}_j(t),a}, \mathbf{x}_{\mathcal{K}_j(t),b}, \mathbf{x}_{\mathcal{K}_j(t),c}$ are the three vertexes of cell $\mathcal{K}_j(t)$. Then we define corresponding basis functions. For time $t, \hat{t} \in [t^n, t^{n+1}]$, we call $\phi \in \mathcal{V}_h(t), \hat{\phi} \in \mathcal{V}_h(\hat{t})$ are corresponding basis functions, if they satisfy that

$$\hat{\phi}(\mathcal{X}_j(\xi, \hat{t})) = \phi(\mathcal{X}_j(\xi, t)), \quad \forall \xi \in [0, 1]^2. \tag{3.8}$$

Now, we write the third order TVD-RK method [32] as follows:

$$\int_{\mathcal{K}_j^{n,1}} \mathbf{v}^{n,1} \cdot \phi^{n,1} \, dx = \int_{\mathcal{K}_j^n} \mathbf{v}^n \cdot \phi^n \, dx + \Delta t \mathcal{L}_j(\omega^n, \mathbf{v}^n, \phi^n, t^n), \tag{3.9}$$

$$\int_{\mathcal{K}_j^{n,2}} \mathbf{v}^{n,2} \cdot \boldsymbol{\phi}^{n,2} dx = \frac{3}{4} \int_{\mathcal{K}_j^n} \mathbf{v}^n \cdot \boldsymbol{\phi}^n dx + \frac{1}{4} \left(\int_{\mathcal{K}_j^{n,1}} \mathbf{v}^{n,1} \cdot \boldsymbol{\phi}^{n,1} dx + \Delta t \mathcal{L}_j(\boldsymbol{\omega}^{n,1}, \mathbf{v}^{n,1}, \boldsymbol{\phi}^{n,1}, t^{n,1}) \right), \tag{3.10}$$

$$\int_{\mathcal{K}_j^{n+1}} \mathbf{v}^{n+1} \cdot \boldsymbol{\phi}^{n+1} dx = \frac{1}{3} \int_{\mathcal{K}_j^n} \mathbf{v}^n \cdot \boldsymbol{\phi}^n dx + \frac{2}{3} \left(\int_{\mathcal{K}_j^{n,2}} \mathbf{v}^{n,2} \cdot \boldsymbol{\phi}^{n,2} dx + \Delta t \mathcal{L}_j(\boldsymbol{\omega}^{n,2}, \mathbf{v}^{n,2}, \boldsymbol{\phi}^{n,2}, t^{n,2}) \right), \tag{3.11}$$

where $\mathcal{K}_j^{n,1} = \mathcal{K}_j^{n+1} = \mathcal{K}_j(t^{n+1})$, $\mathcal{K}_j^{n,2} = \mathcal{K}_j(\frac{t^n+t^{n+1}}{2})$; $\boldsymbol{\phi}^{n,1}$, $\boldsymbol{\phi}^{n+1}$, $\boldsymbol{\phi}^n$ and $\boldsymbol{\phi}^{n,2}$ are equivalent in the sense of basis functions defined in (3.8). The spatial discretization \mathcal{L}_j is defined in (3.4). Usually, it is obvious that the scheme (3.4) combined with (3.9)–(3.11) can not keep the well-balanced property. We need to have more treatments and will introduce two well-balanced schemes for the still water equilibrium below.

3.3 The Well-Balanced Semi-discrete ALE-DG Schemes

In this subsection, we will introduce the well-balanced modifications based on the standard ALE-DG scheme (3.4). We will introduce two different schemes in this subsection and show their properties in the following subsections.

3.3.1 The Semi-discrete ALE-DG Scheme with Hydrostatic Reconstruction

The semi-discrete scheme is the following: Find $\mathbf{v} \in \Pi_h^{2d}(t)$, for any $\boldsymbol{\phi} \in \Pi_h^{2d}(t)$,

$$\frac{d}{dt} \int_{\mathcal{K}_j(t)} \mathbf{v} \cdot \boldsymbol{\phi} dx = \mathcal{L}_j^h(\boldsymbol{\omega}, \mathbf{v}, \boldsymbol{\phi}, t) \tag{3.12}$$

with

$$\mathcal{L}_j^h(\boldsymbol{\omega}, \mathbf{v}, \boldsymbol{\phi}, t) = \int_{\mathcal{K}_j(t)} \mathbf{G}(\boldsymbol{\omega}, \mathbf{v}) : \nabla \boldsymbol{\phi} dx - \int_{\partial \mathcal{K}_j(t)} \widehat{\mathbf{G}}^* \cdot \boldsymbol{\phi}^{int} ds + \int_{\mathcal{K}_j(t)} \mathbf{s}(\mathbf{v}, b) \cdot \boldsymbol{\phi} dx,$$

where

- The well-balanced numerical flux is given by

$$\widehat{\mathbf{G}}^* = \widehat{\mathbf{G}}(\boldsymbol{\omega}, \mathbf{v}^{*,int}, \mathbf{v}^{*,ext}, \mathbf{n}) + \begin{pmatrix} \frac{1}{2}(\boldsymbol{\omega} \cdot \mathbf{n} - \alpha)(\eta^{*,int} - \eta^{int}) + \frac{1}{2}(\boldsymbol{\omega} \cdot \mathbf{n} + \alpha)(\eta^{*,ext} - \eta^{ext}) \\ \frac{g}{2}(\eta^{int} - b^{int})^2 - \frac{g}{2}(h^{*,int})^2 n_1 \\ \frac{g}{2}(\eta^{int} - b^{int})^2 - \frac{g}{2}(h^{*,int})^2 n_2 \end{pmatrix}, \tag{3.13}$$

where $\widehat{\mathbf{G}}(\boldsymbol{\omega}, \mathbf{v}^{int}, \mathbf{v}^{ext}, \mathbf{n})$ is defined in (3.5) and $\mathbf{n} = (n_1, n_2)^T$ is the outer normal vector.

- $\mathbf{v}^{*,int/ext}$ and $h^{*,int/ext}$ are the modified cell boundary values given by

$$\begin{aligned} \mathbf{v}^{*,int} &= (h^{*,int} + b^{int}, (hu)^{int}, (hv)^{int})^T, \\ \mathbf{v}^{*,ext} &= (h^{*,ext} + b^{ext}, (hu)^{ext}, (hv)^{ext})^T, \\ h^{*,int} &= \max(0, \eta^{int} - \max(b^{int}, b^{ext})), \\ h^{*,ext} &= \max(0, \eta^{ext} - \max(b^{int}, b^{ext})). \end{aligned} \tag{3.14}$$

We note that we change the definition of numerical flux (3.13) in well-balanced scheme (3.12) compared with (3.4).

3.3.2 The Semi-discrete ALE-DG Scheme with Special Source Term Treatment

The ALE-DG scheme with special source term treatment is defined in the following form: find $\mathbf{v} \in \Pi_h^{2d}(t)$, for any $\boldsymbol{\phi} = (\zeta, \psi, \delta)^T \in \Pi_h^{2d}(t)$,

$$\frac{d}{dt} \int_{\mathcal{K}_j(t)} \mathbf{v} \cdot \boldsymbol{\phi} \, d\mathbf{x} = \mathcal{L}_j^s(\boldsymbol{\omega}, \mathbf{v}, \boldsymbol{\phi}, t) \tag{3.15}$$

with

$$\mathcal{L}_j^s(\boldsymbol{\omega}, \mathbf{v}, \boldsymbol{\phi}, t) = \int_{\mathcal{K}_j(t)} \mathbf{G}(\boldsymbol{\omega}, \mathbf{v}) : \nabla \boldsymbol{\phi} \, d\mathbf{x} - \int_{\partial \mathcal{K}_j(t)} \widehat{\mathbf{G}}(\boldsymbol{\omega}, \mathbf{v}^{int}, \mathbf{v}^{ext}, \mathbf{n}) \cdot \boldsymbol{\phi}^{int} \, ds + s_j + r_j,$$

where s_j denotes the approximation of the source term $\int_{\mathcal{K}_j(t)} -ghb_x \psi \, d\mathbf{x}$ and r_j denotes the approximation of the source term $\int_{\mathcal{K}_j(t)} -ghb_y \delta \, d\mathbf{x}$. They are defined as follows:

$$\begin{aligned} s_j &= \int_{\partial \mathcal{K}_j(t)} \frac{1}{2} g \tilde{b}^2 \psi^{int} \, ds - \int_{\mathcal{K}_j(t)} \frac{1}{2} g b^2 \psi_x \, d\mathbf{x} \\ &\quad - g \bar{\eta}_j \left(\int_{\partial \mathcal{K}_j(t)} \tilde{b} \psi^{int} \, ds - \int_{\mathcal{K}_j(t)} b \psi_x \, d\mathbf{x} \right) - \int_{\mathcal{K}_j(t)} g(\eta - \bar{\eta}_j) b_x \psi \, d\mathbf{x}, \end{aligned} \tag{3.16}$$

$$\begin{aligned} r_j &= \int_{\partial \mathcal{K}_j(t)} \frac{1}{2} g \tilde{b}^2 \delta^{int} \, ds - \int_{\mathcal{K}_j(t)} \frac{1}{2} g b^2 \delta_y \, d\mathbf{x} \\ &\quad - g \bar{\eta}_j \left(\int_{\partial \mathcal{K}_j(t)} \tilde{b} \delta^{int} \, ds - \int_{\mathcal{K}_j(t)} b \delta_y \, d\mathbf{x} \right) - \int_{\mathcal{K}_j(t)} g(\eta - \bar{\eta}_j) b_y \delta \, d\mathbf{x}, \end{aligned} \tag{3.17}$$

where the notations are defined by

$$\bar{\eta}_j = \frac{1}{|\mathcal{K}_j(t)|} \int_{\mathcal{K}_j(t)} \eta \, d\mathbf{x}, \quad \tilde{b} = \frac{b^{int} + b^{ext}}{2}, \quad \tilde{b}^2 = \frac{(b^{int})^2 + (b^{ext})^2}{2}.$$

Compared with (3.4), we note that we change the definition of the source term approximation $\int_{\mathcal{K}_j(t)} s(\mathbf{v}, b) \cdot \boldsymbol{\phi} \, d\mathbf{x}$ to $s_j + r_j$ in well-balanced scheme (3.15).

3.3.3 Well-Balanced Property for Semi-discrete Schemes

For the well-balanced semi-discrete schemes (3.12) and (3.15), we have the following lemma as key of the well-balanced property.

Lemma 3.1 *If v is equal to the still water equilibrium, then the semi-discrete ALE-DG schemes (3.12) in Sect. 3.3.1 and (3.15) in Sect. 3.3.2 can both be simplified to*

$$\frac{d}{dt} \int_{\mathcal{K}_j(t)} v \cdot \phi \, dx = \int_{\mathcal{K}_j(t)} (\nabla \cdot (\omega \otimes v)) \cdot \phi \, dx, \tag{3.18}$$

where \otimes is the tensor product such that $\omega \otimes v = (\omega_1 v \ \omega_2 v)$ and

$$\nabla \cdot (\omega \otimes v) = (\omega_1 v)_x + (\omega_2 v)_y. \tag{3.19}$$

In order to show the main idea of the proofs, the one-dimensional cases for schemes (3.12) and (3.15) are given in ‘‘Appendices A.1 and A.2’’ respectively. The two-dimensional case is similar and we omit the corresponding proof in two dimensions.

3.4 The Fully Discrete Schemes and Well-Balanced Property

For the DG schemes on static grids, usually Lemma 3.1 indicate the well-balanced property of the corresponding fully-discrete schemes. But for the ALE-DG schemes, this is not the case. The schemes need the property so-called geometric conservation law (GCL) to maintain the well-balanced property in the temporal discretization.

The GCL is an important property to keep high order accuracy for the grid deformation method. We introduce the GCL briefly and refer [23] for details. If a method satisfies the GCL, that means it can preserve the constant states. In one dimension, [23] proved that for the ALE-DG method for conservation laws, semi-discrete ALE-DG scheme satisfies the GCL and the fully discrete scheme with TVD-RK methods (3.9)-(3.11) satisfies discrete geometric conservation law (dGCL). In two dimensions, a modified TVD-RK methods was presented in [19], where each equation of (3.9)–(3.11) adds a coefficient shown in [19]. For example, the Euler forward time discretization in two dimensions is given as follows

$$\beta \frac{|\mathcal{K}_j^n|}{|\mathcal{K}_j^{n+1}|} \int_{\mathcal{K}_j^{n+1}} v^{n+1} \cdot \phi^{n+1} \, dx = \int_{\mathcal{K}_j^n} v^n \cdot \phi^n \, dx + \Delta t \mathcal{L}_j(\omega^n, v^n, \phi^n, t^n), \tag{3.20}$$

where $\beta = 1 + \Delta t(\nabla \cdot \omega^n)$ and $|\mathcal{K}|$ represent the area of the cell \mathcal{K} . In one dimension, we note that (3.20) and the standard Euler forward scheme are the same, since coefficient

$$\beta \frac{|\mathcal{K}_j^n|}{|\mathcal{K}_j^{n+1}|} = (1 + \Delta t \omega_x^n) \frac{x_{j+\frac{1}{2}}^n - x_{j-\frac{1}{2}}^n}{x_{j+\frac{1}{2}}^{n+1} - x_{j-\frac{1}{2}}^{n+1}} = \left(1 + \Delta t \frac{\omega_{j+\frac{1}{2}}^n - \omega_{j-\frac{1}{2}}^n}{x_{j+\frac{1}{2}}^n - x_{j-\frac{1}{2}}^n} \right) \frac{x_{j+\frac{1}{2}}^n - x_{j-\frac{1}{2}}^n}{x_{j+\frac{1}{2}}^{n+1} - x_{j-\frac{1}{2}}^{n+1}} = 1.$$

The second equality is due to (2.4) and the third equality is due to (2.2). For the modified TVD-RK methods, the fully discrete scheme has the dGCL property and we refer [19] for its proof.

For the shallow water equations, which are conservation laws with source terms, we adopt the modified TVD-RK method in [19] and similar dGCL property can be proved for ALE-DG methods. This is essential for our proof about the well-balanced property. We show

the well-balanced property of the fully-discrete ALE-DG scheme with the Euler forward discretization (3.20) and the well-balanced property about the modified TVD-RK methods is the same.

Proposition 3.2 *The semi-discrete ALE-DG schemes (3.12) and (3.15) coupled with the Euler forward scheme (3.20) maintain the still water equilibrium (3.2) exactly.*

The main idea of the proof for Proposition 3.2 in one dimension is shown in “Appendix A.3”. The proof in two dimensions is similar and we omit the details for simplicity.

Remark 3.3 Spurious oscillations can occur in numerical solutions of ALE-DG methods when the solution contains discontinuities, so a slope limiter is needed after each time stage of TVD-RK methods. We use the total variation bounded (TVB) limiter presented in [40]. Such limiter on static mesh still works well on moving meshes for ALE-DG methods. Since we have changed the variables of the shallow water equations from u to v and v equals constants for still water equilibrium, so the TVB limiter won’t destroy the well-balanced property.

3.5 The Approximation of the Bottom Topography and Mass Conservation

We haven’t discussed the choice of the bottom topography b in our proposed schemes. Actually, different approximations of the bottom topography won’t affect the well-balanced property but may affect the mass conservation or the positivity-preserving property. We discuss how to choose the approximations of the bottom topography and the mass conservation property in this subsection and the positivity-preserving property will be introduced in the next subsection.

There are some different choices for the approximations of the bottom topography and in [1] has some discussion about it. Two main choices are:

1. $b \in \mathcal{V}_h(t)$ is the L^2 projection of the exact bottom topography.
2. Use the ALE-DG method to remap b on moving meshes, which means that according to equation $b_t = 0$, find $b \in \mathcal{V}_h(t)$, for any $\delta \in \mathcal{V}_h(t)$,

$$\frac{d}{dt} \int_{\mathcal{K}_j(t)} b \delta \, dx = - \int_{\mathcal{K}_j(t)} \omega b \cdot \nabla \delta \, dx + \int_{\partial \mathcal{K}_j(t)} \widehat{\omega b} \delta^{int} \, ds, \tag{3.21}$$

where $\widehat{\omega b}$ is the numerical flux based on our numerical schemes. In this work, based on the Lax-Friedrichs flux, we set

$$\widehat{\omega b} = (\omega(b^{int} + b^{ext}) \cdot \mathbf{n} + \alpha(b^{ext} - b^{int}))/2, \tag{3.22}$$

where we take the same α in (3.5). We note that (3.21) is semi-discrete scheme, and we choose the same time discretization as v , i.e. the modified TVD-RK methods in [19].

The first approach is shown in [1] and is thought to be more accurate than the second approach for bottom topography in poor condition. But the first approach don’t have the exactly mass conservation, which means that an accuracy enough integration quadrature must be adopted in computations and don’t have the weak positivity property, which is essential for the robustness of schemes and will be introduced in the next subsection.

We claim that our proposed well-balanced ALE-DG schemes have mass conservation coupled with the numerical bottom. First, we claim that the numerical flux of η is single-valued. This is obvious true for the scheme (3.15) and also true for the scheme (3.12) according

to the discussion in (A.3), which means that the scheme (3.12) has single-valued numerical flux for η . So we have that our proposed schemes conserve the surface level $\eta = h + b$. If our schemes conserve the bottom topography b , we can conclude that our schemes have mass conservation. This is obvious true for the scheme (3.21) of b with single-valued numerical flux.

3.6 Positivity-Preserving Property and Limiter

A simple positivity-preserving limiter has been proposed and implemented for the shallow water equations in [40] for the DG method. Such a limiter can keep the water height non-negative under suitable time step size and preserve the local conservation and does not affect the high-order accuracy. The basis of such limiter is a weak positivity property satisfied by the cell average of the water height. In our work, we will show how to couple this limiter with our proposed schemes in one dimension and the two-dimensional case is similar.

We begin with the quadrature rules. Let $\{\hat{x}_j^{(v)}\}_{1 \leq v \leq L}$ be the Gauss-Lobatto nodes in the interval K_j^n , and $\{\hat{\omega}_v\}_{1 \leq v \leq L}$ be the associated quadrature weight satisfying

$$\sum_{v=1}^L \hat{\omega}_v = 1, \quad \hat{\omega}_1 = \hat{\omega}_L = \frac{1}{L(L-1)}, \quad \text{with } L \geq (k+3)/2. \tag{3.23}$$

We note that such quadrature rule is only used for the following proof and we denote the set of quadrature points by

$$\mathbb{S}_{K_j^n} = \{\hat{x}_j^{(v)}, v = 1, \dots, L\}. \tag{3.24}$$

3.6.1 The Weak Positivity Property for the Well-Balanced Schemes

We will show the weak positivity property of our schemes (3.12) and (3.15) combined with the Euler forward time discretization (3.20) and the property for the high-order TVD-RK methods are similar with it.

Theorem 3.4 *If $\forall x \in \mathbb{S}_{K_j^n}, \eta_j^n(x) - b_j^n(x) > 0$ holds, we have*

$$\bar{\eta}_j^{n+1} > \bar{b}_j^{n+1},$$

for both two schemes (3.12) and (3.15) with the Euler forward discretization (3.20) under the CFL-type condition

$$\hat{\alpha}_0 \Delta t < \Delta x_j^n, \tag{3.25}$$

with

$$\hat{\alpha}_0 = \frac{\alpha}{\hat{\omega}_1},$$

where α is the parameter adopted in the Lax-Friedrichs flux (3.5) and $\hat{\omega}_1$ is the quadrature weight in (3.23).

The proof of Theorem 3.4 is shown in ‘‘Appendix A.4’’.

3.6.2 The Positivity-Preserving Limiter

For this part, we mainly refer the results in [40]. Since our numerical methods satisfy the weak positivity property in Theorem 3.4, we can apply a simple scaling limiter to make the values of height at points $x \in \mathbb{S}_{K_j^n}$ positive. We assume $\bar{h}_j^n > 0$, then the scaling limiter is given by

$$\tilde{h}_j^n(x) = \Theta(h_j^n - \bar{h}_j^n) + \bar{h}_j^n, \quad \Theta = \min \left(1, \frac{\bar{h}_j^n - \epsilon}{\bar{h}_j^n - \min_{x \in \mathbb{S}_{K_j^n}} h_j^n(x)} \right).$$

where ϵ is sufficiently small positive numbers to avoid the effect of the round-off error, for example $\epsilon = \min(10^{-11}, \bar{h}_j^n)$. Now we get $\tilde{h}_j^n(x) > 0$ for $x \in \mathbb{S}_{K_j^n}$.

We note that, to preserve the still water equilibrium, we do not change the surface level η . We use the idea in [42] and update the bottom $b_j^n(x)$ by

$$\tilde{b}_j^n(x) = \eta_j^n(x) - \tilde{h}_j^n(x). \tag{3.26}$$

Such an approach can preserve the still water equilibrium and keep the water height positive at the same time.

4 The Well-Balanced ALE-DG Method for Moving Water Equilibrium

In this section, we will present the well-balanced ALE-DG scheme for the moving water equilibrium. Since there is no general form of the moving water equilibrium states in two dimensions, we only discuss about the one-dimensional case (1.1) in this paper. For the ease of presentation, we denote the shallow water equations (1.1) as follows:

$$\mathbf{u}_t + \mathbf{f}(\mathbf{u})_x = \mathbf{s}(\mathbf{u}, b), \tag{4.1}$$

where

$$\mathbf{f}(\mathbf{u}) = \left(\begin{array}{c} hu \\ \frac{(hu)^2}{h} + \frac{1}{2}gh^2 \end{array} \right), \tag{4.2}$$

$$\mathbf{s}(\mathbf{u}, b) = \left(\begin{array}{c} 0 \\ -ghb_x \end{array} \right). \tag{4.3}$$

The numerical solution is still denoted by \mathbf{u} with an abuse of notation. Since the mesh grids are moving with time, it is difficult to preserve the moving water equilibrium. In [35], a well-balanced DG scheme has been developed based on hydrostatic reconstruction. Such approach can't extend to ALE-DG methods directly. Standard TVD-RK time discretization can't preserve the moving water equilibrium and conserve the mass at the same time. Inspired by the work on the well-balanced scheme for Euler equations with explicitly given equilibrium state in [24], we proposed a well-balanced ALE-DG scheme for moving water equilibrium.

4.1 Preliminary Preparation

4.1.1 Relation Between Conservative and Equilibrium Variables

The variables $\mathbf{u} = (h, hu)^T$ are called conservative variables and the equilibrium variables \mathbf{v} for moving water are defined as follows:

$$\mathbf{v} = (E, m)^T = \left(\frac{1}{2}u^2 + g(h + b), hu \right)^T. \tag{4.4}$$

For the still water case, the transformation between \mathbf{u} and \mathbf{v} is linear. But the transform functions are nonlinear and complicated for the moving water case. It is necessary to find the exactly transform functions between \mathbf{u} and \mathbf{v} .

- Given \mathbf{u} and the bottom function b , the value of \mathbf{v} can be computed directly. We denote this transform function by $\mathbf{v} = \mathbf{V}(\mathbf{u}, b)$.
- Given \mathbf{v} and the bottom function b , the value of \mathbf{u} can not be computed directly. We mainly follow the idea presented in [35] and present the main idea as follows.

If $m = 0$, the solution is trivial. Next, we assume that $m \neq 0$. From (4.4), we have

$$E = \frac{m^2}{2h^2} + g(h + b). \tag{4.5}$$

Thus we have

$$2gh^3 - 2(E - gb)h^2 + m^2 = 0. \tag{4.6}$$

This is a cubic polynomial of h with coefficients determined by \mathbf{v} .

We use the Cardan formula to get three roots for the polynomial. If

$$E \geq \frac{3}{2}(g|m|)^{\frac{2}{3}} + gb,$$

we can claim that there are two real roots greater than zero and one real root less than zero. By using this result, we can compute \mathbf{u} exactly. We denote the two real roots greater than zero by r_1 and r_2 , and assume that $r_1 \leq r_2$ without loss of generality.

We define the Froude number Fr and the sign function σ :

$$Fr := |u|/\sqrt{gh}, \quad \sigma := \text{sign}(Fr - 1). \tag{4.7}$$

Then we introduce the three different flow states

- sonic flow: $\sigma = 0, h = r_1 = r_2$;
- subsonic flow: $\sigma < 0, h = r_2$;
- supersonic flow: $\sigma > 0, h = r_1$.

Finally we get the following lemma.

Lemma 4.1 ([35]) *We assume that \mathbf{v} and b are given suitably such that*

$$E \geq \frac{3}{2}(g|m|)^{\frac{2}{3}} + gb.$$

If we also know the flow state (sonic, sub- or supersonic), then we can compute the unique solution h exactly. We denote this transform function by $h = H(\mathbf{u}, b, \sigma)$.

We denote the transform function $\mathbf{u} = \mathbf{U}(\mathbf{v}, b, \sigma) = (H(\mathbf{v}, b, \sigma), m)^T$.

4.1.2 Definition of Well-Balanced Property on Moving Meshes

Since the meshes are moving with time, the numerical solution must become different with different meshes. So it's hard to define the well-balanced property on moving meshes. We assume that the desired moving water equilibrium is explicitly given and is denoted by \mathbf{u}^d . We note that such equilibrium states can be obtained according to Sect. 4.1.1 by giving (\mathbf{v}, b, σ) . Set \mathbf{u}^e as an approximation of \mathbf{u}^d , which is independent of the flux term and source term of the shallow water equations. For example, the L^2 projection of \mathbf{u}^d is a choice that meet the conditions. We define the well-balanced property on moving meshes as: if the numerical solution \mathbf{u} equals \mathbf{u}^e for all time $t \geq 0$, we call that the scheme can preserve the moving water equilibrium. Such a definition is reasonable and in our numerical examples in Sect. 5, our proposed scheme can capture the small perturbation to the moving water equilibrium well. Like the still water equilibrium case, considering the mass conservation and the positivity-preserving property, we define \mathbf{u}^e by using the ALE-DG scheme of $(\mathbf{u}^d)_t$ instead of L^2 projection in our proposed scheme, which is introduced as follow:

For initialization, we define $\mathbf{u}^{e,0} = P\mathbf{u}^d \in \Pi_h(0)$. The way to update \mathbf{u}^e is by using the ALE-DG scheme of $\mathbf{u}_t^e = 0$: Find $\mathbf{u}^e \in \Pi_h(t)$, for any $\delta \in \Pi_h(t)$,

$$\frac{d}{dt} \int_{K_j(t)} \mathbf{u}^e \cdot \delta dx = \widehat{\omega \mathbf{u}^e}_{j+\frac{1}{2}} \cdot \delta^-_{j+\frac{1}{2}} - \widehat{\omega \mathbf{u}^e}_{j-\frac{1}{2}} \cdot \delta^+_{j-\frac{1}{2}} - \int_{K_j(t)} \omega \mathbf{u}^e \cdot \delta_x dx, \tag{4.8}$$

where $\widehat{\omega \mathbf{u}^e}$ is the numerical flux, which is given by

$$\widehat{\omega \mathbf{u}^e} = \omega \mathbf{u}^d. \tag{4.9}$$

We note that $\widehat{\omega \mathbf{u}^e} = 0$ at the cell boundary when \mathbf{u}^d is discontinuous, since we assume that the mesh grids must locate exactly at the discontinuity and won't move i.e. $\omega = 0$.

4.1.3 Hydrostatic Reconstruction

We will use the idea of hydrostatic reconstruction [35] to modify numerical fluxes. First, we define \mathbf{u}^r by

$$\mathbf{u}^r = \mathbf{u} - \mathbf{u}^e \in \Pi_h(t). \tag{4.10}$$

We can think that \mathbf{u} is divided into two parts, \mathbf{u}^e in (4.8) and \mathbf{u}^r . \mathbf{u}^e represents the equilibrium part and \mathbf{u}^r denotes the residue part, which means that if \mathbf{u}_{ex} is in equilibrium state, we have $\mathbf{u} = \mathbf{u}^e$ and $\mathbf{u}^r = 0$.

Secondly we will modify the cell boundary value of \mathbf{u} . There are two steps to get the modified boundary value $\mathbf{u}^{*,\pm}_{j+\frac{1}{2}}$:

1. Compute the cell boundary value $\mathbf{u}^{r,\pm}_{j+\frac{1}{2}}$ of \mathbf{u}^r .
2. Then the modified boundary value are defined as follow:

$$\mathbf{u}^{*,\pm}_{j+\frac{1}{2}} = \left(\max \left(0, h^{d,\pm}_{j+\frac{1}{2}} + h^{r,\pm}_{j+\frac{1}{2}} \right), m^{d,\pm}_{j+\frac{1}{2}} + m^{r,\pm}_{j+\frac{1}{2}} \right)^T. \tag{4.11}$$

We note that although the desired equilibrium states $\mathbf{u}^d \notin \Pi_h(t)$, it can be discontinuous at the cell boundary due different σ or discontinuous bottom topography. So in the definition, we distinguish its right and left limits as $\mathbf{u}^{d,\pm}$.

4.1.4 Treatment for the Source Term

As in [35], we define the approximation of the source term $\int_{K_j(t)} s(\mathbf{u}, b) \cdot \boldsymbol{\phi} dx$ by

$$s_j = - \int_{K_j(t)} \mathbf{f}(\mathbf{u}^e) \cdot \boldsymbol{\phi}_x dx + f(\mathbf{u}_{j+\frac{1}{2}}^{e,-}) \cdot \boldsymbol{\phi}_{j+\frac{1}{2}}^- - f(\mathbf{u}_{j-\frac{1}{2}}^{e,+}) \cdot \boldsymbol{\phi}_{j-\frac{1}{2}}^+ + \int_{K_j(t)} s(\mathbf{u}^r, b) \cdot \boldsymbol{\phi} dx, \tag{4.12}$$

where \mathbf{u}^e and \mathbf{u}^r is defined in (4.8) and (4.10); $\mathbf{u}_{j+\frac{1}{2}}^{e,\pm}$ are the boundary value of \mathbf{u}^e ; $\boldsymbol{\phi}$ is the test function vector. It can be proved that s_j has a high-order approximation (See [35] for details).

4.2 Semi-discrete ALE-DG Scheme

The semi-discrete scheme is the following: Find $\mathbf{u} \in \Pi_h(t)$, for any $\boldsymbol{\phi} \in \Pi_h(t)$,

$$\frac{d}{dt} \int_{K_j(t)} \mathbf{u} \cdot \boldsymbol{\phi} dx = \mathcal{L}_j^m(\omega, \mathbf{u}, \boldsymbol{\phi}, t) \tag{4.13}$$

with

$$\mathcal{L}_j^m(\omega, \mathbf{u}, \boldsymbol{\phi}, t) = \int_{K_j(t)} \mathbf{g}(\omega, \mathbf{u}) \cdot \boldsymbol{\phi}_x dx - \hat{\mathbf{g}}_{j+\frac{1}{2}}^l \cdot \boldsymbol{\phi}_{j+\frac{1}{2}}^- + \hat{\mathbf{g}}_{j-\frac{1}{2}}^r \cdot \boldsymbol{\phi}_{j-\frac{1}{2}}^+ + s_j,$$

where

- $\mathbf{g}(\omega, \mathbf{u}) = \mathbf{f}(\mathbf{u}) - \omega \mathbf{u}$;
- $\hat{\mathbf{g}}^l$ and $\hat{\mathbf{g}}^r$ are defined as

$$\hat{\mathbf{g}}_{j+\frac{1}{2}}^l = \hat{\mathbf{g}}(\omega_{j+\frac{1}{2}}, \mathbf{u}_{j+\frac{1}{2}}^{*,-}, \mathbf{u}_{j+\frac{1}{2}}^{*,+}) + \mathbf{f}(\mathbf{u}_{j+\frac{1}{2}}^-) - \mathbf{f}(\mathbf{u}_{j+\frac{1}{2}}^{*,-}), \tag{4.14}$$

$$\hat{\mathbf{g}}_{j-\frac{1}{2}}^r = \hat{\mathbf{g}}(\omega_{j-\frac{1}{2}}, \mathbf{u}_{j-\frac{1}{2}}^{*,-}, \mathbf{u}_{j-\frac{1}{2}}^{*,+}) + \mathbf{f}(\mathbf{u}_{j-\frac{1}{2}}^+) - \mathbf{f}(\mathbf{u}_{j-\frac{1}{2}}^{*,+}); \tag{4.15}$$

- $\mathbf{u}_{j+\frac{1}{2}}^{*,\pm}$ are given by (4.11);
- $\hat{\mathbf{g}}(\omega, \mathbf{u}^{*,-}, \mathbf{u}^{*,+})$ is a monotone numerical flux of $\mathbf{g}(\omega, \mathbf{u}^*)$. We use the Lax-Friedrichs flux, which is defined as follow:

$$\hat{\mathbf{g}}(\omega, \mathbf{u}^l, \mathbf{u}^r) = \frac{1}{2} \left(\mathbf{g}(\omega, \mathbf{u}^l) + \mathbf{g}(\omega, \mathbf{u}^r) - \alpha(\mathbf{u}^r - \mathbf{u}^l) \right), \tag{4.16}$$

$$\alpha = \max_{\mathbf{u} \in \{\mathbf{u}^l, \mathbf{u}^r\}} (|\mathbf{u} - \omega| + \sqrt{g\hat{h}}), \tag{4.17}$$

if \mathbf{u}^d is continuous at the cell boundary and we use the Roe’s flux in the scheme, which is defined as follow:

$$\begin{aligned} \hat{\mathbf{g}}(\omega, \mathbf{u}^l, \mathbf{u}^r) &= \frac{1}{2} \left(\mathbf{g}(\omega, \mathbf{u}^l) + \mathbf{g}(\omega, \mathbf{u}^r) - A_\omega(\mathbf{u}^r - \mathbf{u}^l) \right), \\ A_\omega &= R \begin{pmatrix} |\hat{q} - \hat{c}| & 0 \\ 0 & |\hat{q} + \hat{c}| \end{pmatrix} R^{-1}, \quad R = \begin{pmatrix} 1 & 1 \\ \hat{q} - \hat{c} & \hat{q} + \hat{c} \end{pmatrix}, \\ \hat{c} &= \sqrt{g\hat{h}}, \quad \hat{h} = \frac{1}{2} (h^l + h^r), \quad \hat{q} = \frac{u^l \sqrt{h^l} + u^r \sqrt{h^r}}{\sqrt{h^l} + \sqrt{h^r}} - \omega, \end{aligned}$$

if \mathbf{u}^d has a stationary shock at the cell boundary.

- s_j is defined in (4.12).

Next, similar to the case of the still water equilibrium, we will show the property of the semi-discrete ALE-DG scheme (4.13) is in Lemma 4.2 and its proof in ‘‘Appendix B.1’’.

Lemma 4.2 *If the numerical solution $\mathbf{u} = \mathbf{u}^e$ where \mathbf{u}^e is defined in (4.8) and the stationary shocks of \mathbf{u}^d are all located at the cell boundaries, the semi-discrete ALE-DG scheme (4.13) satisfies*

$$\mathcal{L}_j^m(\omega, \mathbf{u}, \phi, t) = \omega_{j+\frac{1}{2}} \mathbf{u}_{j+\frac{1}{2}}^{d,-} \cdot \phi_{j+\frac{1}{2}}^- - \omega_{j-\frac{1}{2}} \mathbf{u}_{j-\frac{1}{2}}^{d,+} \cdot \phi_{j-\frac{1}{2}}^+ - \int_{K_j(t)} \omega \mathbf{u}^e \cdot \phi_x dx. \quad (4.18)$$

4.3 Fully-Discrete ALE-DG Scheme and Well-Balanced Property

For the time discretization, we adopt the same TVD-RK method (3.9)–(3.11) for the semi-discrete scheme (4.13) of \mathbf{u} and semi-discrete scheme (4.8) of \mathbf{u}^e to keep the well-balanced property.

Remark 4.3 We claim that our ALE-DG scheme (4.13) coupled with TVD-RK time discretization conserves the mass. Since m^e always equals constant, we $m_{j+\frac{1}{2}}^{*, -} = m_{j+\frac{1}{2}}^{*, +}$. This leads to the numerical flux of h is single-valued. Then we can say our claim holds true.

Since TVD-RK time discretizations are convex combinations of the Euler forward operators, it’s enough to use the fully-discrete ALE-DG scheme with the Euler forward discretization for the proofs in the following two subsections:

$$\int_{K_j^{n+1}} \mathbf{u}^{n+1} \cdot \phi^{n+1} dx = \int_{K_j^n} \mathbf{u}^n \cdot \phi^n dx + \Delta t \mathcal{L}_j^m(\omega^n, \mathbf{u}^n, \phi^n, t^n), \quad (4.19)$$

combined with

$$\begin{aligned} & \int_{K_j^{n+1}} \mathbf{u}^{e,n+1} \cdot \delta^{n+1} dx \\ &= \int_{K_j^n} \mathbf{u}^{e,n} \cdot \delta^n dx + \Delta t \left(\widehat{\omega \mathbf{u}^e}_{j+\frac{1}{2}}^n \cdot \delta_{j+\frac{1}{2}}^{n,-} - \widehat{\omega \mathbf{u}^e}_{j-\frac{1}{2}}^n \cdot \delta_{j-\frac{1}{2}}^{n,+} - \int_{K_j^n} \omega^n \mathbf{u}^{e,n} \cdot \delta_x^n dx \right). \end{aligned} \quad (4.20)$$

Proposition 4.4 *The scheme described in (4.19) maintain the moving water equilibrium exactly, which means that*

$$\mathbf{u} = \mathbf{u}^e, \quad \mathbf{u}^r = 0, \quad (4.21)$$

hold exactly.

The proof of Proposition 4.4 is shown in ‘‘Appendix B.2’’.

Remark 4.5 We also want to apply the TVB limiter to the numerical solution after each time stage of TVD-RK methods. Different from the well-balanced scheme for still water equations, the TVB limiter may destroy the moving water equilibrium. We use the strategy in [35] on static mesh, which is the same for moving meshes. We divide the TVB limiter into two parts: First, we check if the TVB limiter is needed based on $\mathbf{u} - \mathbf{u}^e$; Second, we apply TVB limiter to the ‘trouble’ cells found in the first step.

4.4 Positivity Property of the Well-Balanced Scheme for the Moving Water Equilibrium

We want to adopt the same positivity-preserving limiter in [40] to ensure the robustness of our proposed scheme. As shown in [35,40], the positivity-preserving limiter is based on the weak positivity property of the cell average of water height h . In the following, we will show the weak positivity property of our proposed scheme coupled with Euler forward time discretization (4.19). We can obtain the equation for the cell average of h in the scheme (4.19) as follows

$$\frac{\Delta_j^{n+1}}{\Delta_j^n} \bar{h}_j^{n+1} = \bar{h}_j^n - \lambda \left(\hat{\mathbf{g}}_{j+\frac{1}{2}}^{l,[1]} - \hat{\mathbf{g}}_{j-\frac{1}{2}}^{r,[1]} \right), \tag{4.22}$$

where $\hat{\mathbf{g}}^{l/r,[1]}$ denote the first component of $\hat{\mathbf{g}}^{l/r}$ and $\lambda = \frac{\Delta t}{\Delta_j}$. Then we introduce the following weak positivity property with Lax-Friedrichs numerical flux (4.16) in (4.22) and its proof is shown in ‘‘Appendix B.3’’.

Theorem 4.6 *If $\forall x \in \mathbb{S}_{K_j^n}$, $h_j^n(x) > 0$ holds, for \bar{h}_j^{n+1} calculated from (4.22), we have*

$$\bar{h}_j^{n+1} > 0, \tag{4.23}$$

for the scheme (4.19) under CFL-type condition

$$\hat{\alpha}_0 \Delta t < \Delta_j^n, \tag{4.24}$$

with

$$\hat{\alpha}_0 = \frac{\alpha}{\hat{\omega}_1} \max_{x \in \{x_{j-\frac{1}{2}}^+, x_{j+\frac{1}{2}}^-\}} \frac{h_j^*(x)}{h_j^n(x)}, \tag{4.25}$$

where $h_{j+\frac{1}{2}}^{*,\pm}$ is defined in (4.11).

Now, we can apply the positivity-preserving limiter introduced in Sect. 3.6.2 for both \mathbf{u} and \mathbf{u}^e at each TVD-RK stage to keep the water height positive as before. Note that this positivity-preserving limiter preserves the local conservation and does not destroy the high order accuracy.

Moreover, we note that our proposed scheme can keep the water height h positive and preserve the moving water equilibrium at the same time. When the desired equilibrium \mathbf{u}^d involve low water height or dry area, the limiter may change the numerical solution $\mathbf{u}^n, \mathbf{u}^{e,n}$, but we still have $\mathbf{u}^n = \mathbf{u}^{e,n}$ and $\mathbf{u}^{r,n} = 0$ since we applied the same limiter to $\mathbf{u}^n, \mathbf{u}^{e,n}$. So we still can use Lemma 4.2 and Proposition 4.4 to get $\mathbf{u}^{n+1} = \mathbf{u}^{e,n+1}$ since the assumption of $\mathbf{u}^n = \mathbf{u}^{e,n}$ still holds true.

4.5 Algorithm

In summary, we will implement our scheme by following steps.

1. Initialization at time $t = 0$ to get \mathbf{u}^0 and $\mathbf{u}^{e,0} = P\mathbf{u}^d$.
2. We apply the same TVD-RK methods (3.9)–(3.11) for both \mathbf{u} and \mathbf{u}^e .
3. For each time stage in TVD-RK methods, we apply the positivity-preserving limiter Θ in Sect. 3.6.2 to both \mathbf{u} and \mathbf{u}^e ; We apply the TVB limiter in [35] to \mathbf{u} .
4. We compute \mathcal{L}_j^m in (4.13) for \mathbf{u} and (4.8) for \mathbf{u}^e to finish the spatial discretization.
5. Repeat the steps 2-4 to continue numerical simulations.

5 Numerical Examples

In this section we provide numerical results. We first give some basic settings. The gravitation constant g is taken as $9.812m/s^2$. In the following numerical examples, to verify the well-balanced property and the positivity-preserving property of the ALE-DG schemes on moving mesh, the grid movement is prescribed explicitly and is not derived from the computed solution. We adopt the moving grid

$$x_{j+\frac{1}{2}}(t) = x_{j+\frac{1}{2}}(0) + \frac{1}{3(x_r - x_l)^2} \sin\left(\frac{2\pi t}{t_{end}}\right) (x_{j+\frac{1}{2}}(0) - x_r) (x_{j+\frac{1}{2}}(0) - x_l), \tag{5.1}$$

where x_l and x_r are the endpoints of the computational domain in most 1D examples and

$$\begin{aligned} x_j(t) &= x_j(0) + 0.03 \sin\left(\frac{2\pi x_j(0)}{x_r - x_l}\right) \sin\left(\frac{2\pi y_j(0)}{y_r - y_l}\right) \sin\left(\frac{2\pi t}{t_{end}}\right), \\ y_j(t) &= y_j(0) + 0.02 \sin\left(\frac{2\pi x_j(0)}{x_r - x_l}\right) \sin\left(\frac{2\pi y_j(0)}{y_r - y_l}\right) \sin\left(\frac{4\pi t}{t_{end}}\right), \end{aligned} \tag{5.2}$$

where t_{end} is the stop time and x_r, x_l, y_r, y_l are the vertexes of the rectangle domain in most 2D examples and P^2 piecewise polynomials in most numerical examples, unless otherwise stated. The CFL number is set as 0.15 for 1D examples according to [35,40] and 0.1 for 2D examples according to [43].

In Sects. 5.1–5.3, numerical examples are shown to demonstrate the properties of the numerical schemes for the still water equilibrium in 1D and 2D, the moving water equilibrium in 1D respectively. Examples 5.1, 5.6 and 5.10 are well-balanced tests. Examples 5.3, 5.8 and 5.11 are small perturbation tests. Examples 5.2, 5.7 and 5.9 are accuracy tests. Examples 5.4 and 5.12 are positivity-preserving tests.

5.1 Still Water Equilibrium in 1D

Example 5.1 Test for the well-balanced property

The purpose of this example is to verify that our schemes indeed maintain the well-balanced property on moving grid with smooth or discontinuous bottom. We choose two different functions for the bottom topography for $0 \leq x \leq 1$.

- Smooth bottom

$$b(x) = 5e^{-40(x-0.5)^2}. \tag{5.3}$$

Table 1 (Example 5.1) The hydrostatic reconstruction scheme and the special source term scheme, L^1 and L^∞ errors for different precision with the smooth bottom (5.3)

Precision	Hydrostatic reconstruction				Special source term			
	L^1 error		L^∞ error		L^1 error		L^∞ error	
	η	hu	η	hu	η	hu	η	hu
Double	1.42E-11	1.11E-12	1.46E-11	4.76E-12	1.46E-11	1.21E-12	1.50E-11	6.26E-12
Quadruple	1.37E-29	8.28E-31	1.41E-29	2.91E-30	1.37E-29	9.58E-31	1.41E-29	3.74E-30

Table 2 (Example 5.1) The hydrostatic reconstruction scheme and the special source term scheme, L^1 and L^∞ errors for different precision with the discontinuous bottom (5.4)

Precision	Hydrostatic reconstruction				Special source term			
	L^1 error		L^∞ error		L^1 error		L^∞ error	
	η	hu	η	hu	η	hu	η	hu
Double	1.42E-11	1.12E-12	1.45E-11	3.89E-12	1.47E-11	1.32E-12	1.50E-11	5.53E-12
Quadruple	1.37E-29	7.80E-31	1.40E-29	2.74E-30	1.37E-29	1.04E-30	1.40E-29	4.14E-30

- Discontinuous bottom

$$b(x) = \begin{cases} 4, & \text{if } 0.4 \leq x \leq 0.8, \\ 0, & \text{otherwise.} \end{cases} \tag{5.4}$$

The initial data is

$$h + b = 10, \quad hu = 0.$$

We set stop time $t_{end} = 0.5$ and use $N = 200$ uniform cells at the beginning. We show the L^1 and L^∞ errors for two cases at different precision. We use the schemes hydrostatic reconstruction (3.12) and special source term (3.15). Table 1 shows the results of two schemes with the smooth bottom (5.3) and Table 2 shows the results of two schemes with the discontinuous bottom (5.4). We can see all results can reach the round-off error, which means the schemes are well-balanced.

Example 5.2 Accuracy test with moving boundary

In this example, we will show that our ALE-DG schemes can achieve high order accuracy for the problem with moving boundary. We choose the following bottom function and initial conditions

$$b(x) = \sin^2(\pi x), \quad h(x, 0) = 5 + e^{\cos(2\pi x)}, \quad (hu)(x, 0) = \sin(\cos(2\pi x)), \quad x \in [0, 1],$$

with periodic boundary conditions. Since the exact solution is not known explicitly, we use the fifth order finite volume WENO scheme with $N = 12800$ cells to compute a reference solution, and treat this reference solution as the exact solution in computing the numerical errors. We set stop time $t_{end} = 0.1$ and the solution is still smooth. A moving computational domain at each time step is used and the grid velocity is set as follows

$$\omega_{j+\frac{1}{2}}(t) = 1 + \sin\left(\frac{2\pi t}{t_{end}}\right) \left(x_{j+\frac{1}{2}}(0) - x_1\right) \left(x_{j+\frac{1}{2}}(0) - x_0\right).$$

Fig. 1 Example 5.2. The meshes with 25 cells at different time t for the moving domain problem. The boundaries move with speed $\omega = 1$

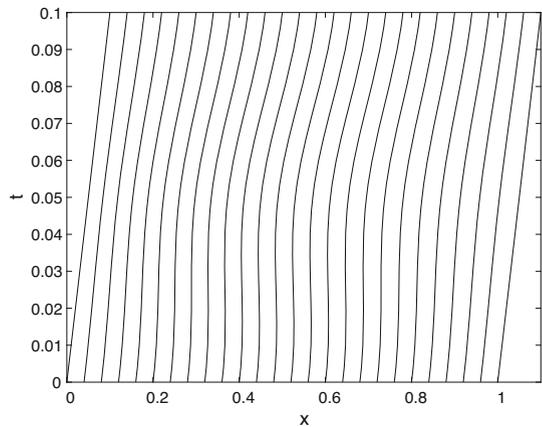


Table 3 (Example 5.2) The hydrostatic reconstruction scheme and the special source term scheme, L^1 error table for the moving boundary problem

N	Hydrostatic reconstruction				Special source term			
	η		hu		η		hu	
	L^1 error	Order	L^1 error	Order	L^1 error	Order	L^1 error	Order
25	9.71E-04	–	7.42E-03	–	9.71E-04	–	7.42E-03	–
50	9.16E-05	3.41	6.63E-04	3.48	9.16E-05	3.41	6.63E-04	3.48
100	9.41E-06	3.28	5.94E-05	3.48	9.41E-06	3.28	5.94E-05	3.48
200	1.19E-06	2.98	7.46E-06	2.99	1.19E-06	2.98	7.46E-06	2.99
400	1.53E-07	2.97	9.49E-07	2.97	1.53E-07	2.97	9.49E-07	2.97
800	1.94E-08	2.98	1.19E-07	2.99	1.94E-08	2.98	1.19E-07	2.99

The initial mesh is set as uniform mesh. So the mesh at every time step t^k can be defined as

$$x_{j+\frac{1}{2}}(t^k) = x_{j+\frac{1}{2}}(t^{k-1}) + (t^k - t^{k-1})\omega_{j+\frac{1}{2}}(t^{k-1}),$$

which is consist with the definition of the grid velocity. The meshes with 25 cells for different time are shown in Fig. 1 for example, where the boundaries move with the speed $\omega = 1$. Since this moving boundary problem is periodic, the boundary movement is easy to be implemented in the current algorithm framework. The L^1 errors are shown in Table 3 for both two schemes, hydrostatic reconstruction (3.12) and special source term (3.15). Figure 2 shows the numerical solutions with 200 cells compared with reference solution. We can see that our schemes can calculate the problem correctly in high-order accurate with moving boundaries.

Example 5.3 A small perturbation test

We will test the situation when a small perturbation added into the still water equilibrium. We first show that our ALE-DG methods can capture the perturbation correctly when involving moving meshes (5.1). Then we will adopt a specific adaptive mesh to show the efficiency of

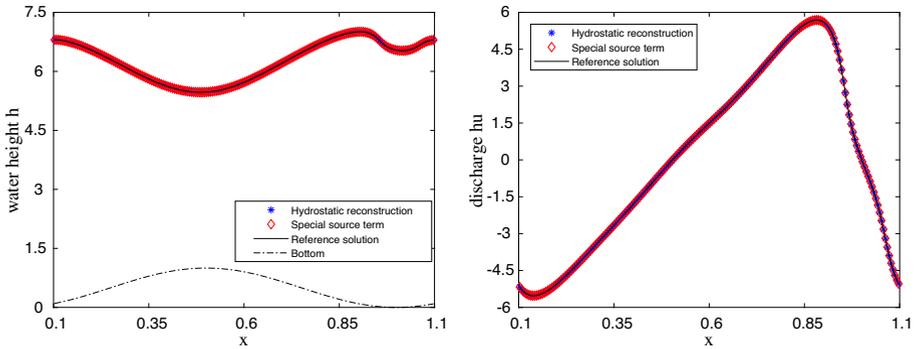


Fig. 2 (Example 5.2) The numerical solution of surface level η and discharge hu with 200 cells compared with reference solution for the moving domain problem

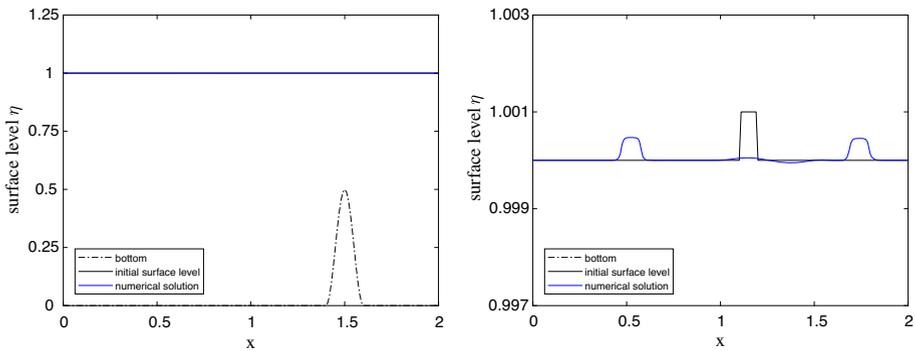


Fig. 3 (Example 5.3) The numerical solution of surface level η compared with initial data and bottom topography with $\epsilon = 0.001$. Right is the zoom in of the left figure for $\eta \in [0.997, 1.003]$

the ALE-DG methods. The initial conditions are

$$(hu)(x, 0) = 0, \quad h(x, 0) = \begin{cases} 1 - b(x) + \epsilon, & \text{if } 1.1 \leq x \leq 1.2, \\ 1 - b(x), & \text{otherwise,} \end{cases} \quad (5.5)$$

where ϵ is the perturbation constant, with the bottom topography

$$b(x) = \begin{cases} 0.25(\cos(10\pi(x - 1.5)) + 1), & \text{if } 1.4 \leq x \leq 1.6, \\ 0, & \text{otherwise.} \end{cases} \quad (5.6)$$

In this example, we consider a quiet small perturbation $\epsilon = 0.001$ and set the stop time as $t_{end} = 0.2$. The transmissive boundary conditions are used in our numerical tests. For the exact solution, the perturbation will propagate to the left and right at the characteristic speeds $\pm\sqrt{gh}$. The reference solutions computed by the hydrostatic reconstruction scheme with $N = 3000$ cells are plotted in Fig. 3, together with the initial surface level and bottom topography. In Fig. 4, we plot the numerical results by two ALE-DG schemes (3.12) and (3.15) on the coarse mesh with $N = 200$. It shows that both ALE-DG schemes can capture the small perturbation near the equilibrium on the relative coarse meshes.

Then we adopt a special moving grid to compare the ALE-DG method with the DG method on the uniform static grid. We use the hydrostatic reconstruction scheme (3.12) in this test.

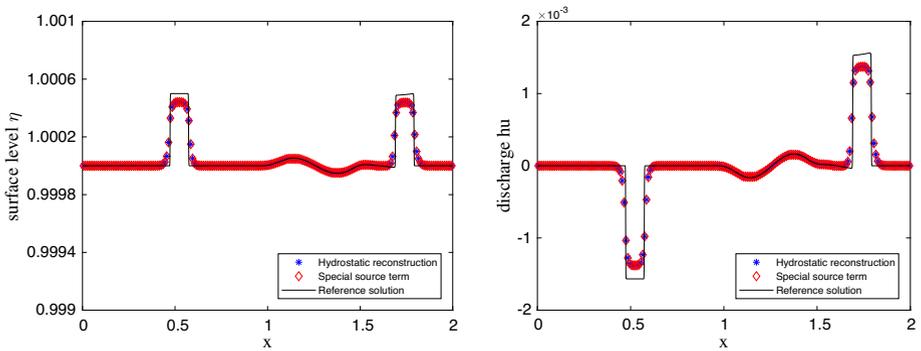


Fig. 4 (Example 5.3). $\epsilon = 0.001$ and $N = 200$. Left: the surface level η ; right: the discharge hu

With $N = 200$ cells, the moving grid is defined as follows. The initial grid is

$$x_{j+\frac{1}{2}}(0) = \begin{cases} \frac{j}{59}, & \text{if } 0 \leq j \leq 59, \\ 1 + 0.003(j - 59), & \text{if } 60 \leq j \leq 159, \\ 1.3 + \frac{7(j - 159)}{410}, & \text{if } 160 \leq j \leq 200, \end{cases} \quad (5.7)$$

and the final grid at the stop time $t_{end} = 0.2$ is

$$x_{j+\frac{1}{2}}(0.2) = \begin{cases} \frac{11}{650}j, & \text{if } 0 \leq j \leq 26, \\ 0.44 + 0.003(j - 26), & \text{if } 27 \leq j \leq 76, \\ 0.59 + 0.016875(j - 76), & \text{if } 77 \leq j \leq 140, \\ 1.67 + 0.003(j - 140), & \text{if } 141 \leq j \leq 190, \\ 1.82 + 0.018(j - 190), & \text{if } 191 \leq j \leq 200. \end{cases} \quad (5.8)$$

Then we define straight lines connecting two meshes and have

$$\omega_{j+\frac{1}{2}} = 5(x_{j+\frac{1}{2}}(0.2) - x_{j+\frac{1}{2}}(0)), \quad (5.9)$$

and for $0 < t < 0.2$

$$x_{j+\frac{1}{2}}(t) = x_{j+\frac{1}{2}}(0) + \omega_{j+\frac{1}{2}}t. \quad (5.10)$$

The moving grid is plotted in Fig. 5. The numerical solutions are shown in Fig. 6. The reference solution is computed by the hydrostatic DG scheme (i.e. the grid velocity $\omega \equiv 0$ in the ALE-DG scheme (3.12)) on the uniform grid with $N = 3000$ cells. We can see that the ALE-DG solution has better resolution than the DG solution. Additionally, we compute the CPU time in this test. For the solution on moving mesh with 200 cells, the CPU time is 45.28s, which is comparable to the CPU time 41.95s on static mesh with 200 cells. It is also found that with suitable moving mesh, the ALE-DG solution has less numerical dissipation than DG solutions on static mesh and the size of the small perturbation is almost the same as the reference solution.

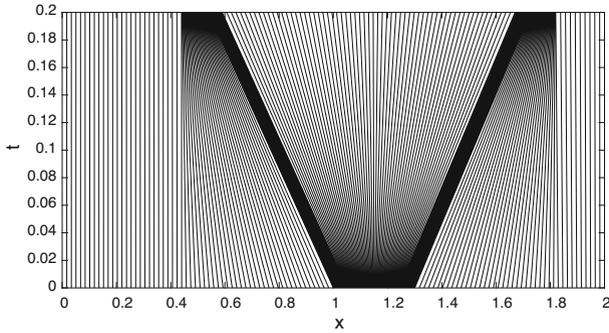


Fig. 5 (Example 5.3) The meshes at different time t

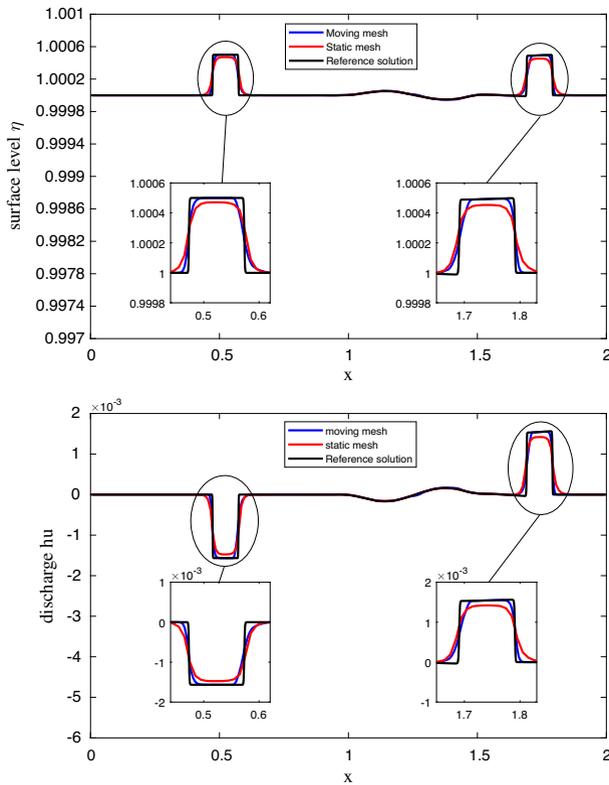


Fig. 6 (Example 5.3) The comparison with ALE-DG method and DG method for a small perturbation $\epsilon = 0.001$ of the still water equilibrium. We use the hydrostatic reconstruction scheme. Left: the surface level η ; right: the discharge hu

Example 5.4 Parabolic bowl

This example is used to demonstrate the positivity-preserving ability of our methods. The parabolic bottom is given by

$$b(x) = h_0(x/a)^2, \tag{5.11}$$

Table 4 (Example 5.4) The minimum of the water height h at time $t = 1000, 2000, 3000, 4000, 5000, 6000$ for two schemes

Time t	1000	2000	3000	4000	5000	6000
Hydrostatic reconstruction	1.21E-11	1.23E-11	1.22E-11	1.17E-11	1.20E-11	1.25E-11
Special source term	1.27E-11	1.12E-11	1.21E-11	1.23E-11	1.22E-11	1.20E-11

with $h_0 = 10$ and $a = 3000$. The computational domain is set as $[-5000, 5000]$. The analytical water surface level is given in [40]

$$\eta(x, t) = h_0 - \frac{B^2}{4g} \cos(2\kappa t) - \frac{B^2}{4g} - \frac{Bx}{2a} \sqrt{\frac{8h_0}{g}} \cos(\kappa t), \tag{5.12}$$

where $\kappa = \sqrt{2gh_0}/a$ and $B = 5$. The exact location of wet/dry front takes the form

$$x_0 = -\frac{B\kappa a^2}{2gh_0} \cos(\kappa t) \pm a. \tag{5.13}$$

We use the exact solution for the initial data and the boundary conditions. The CFL number is set as 0.15, which is little smaller than $\hat{\omega}_1 = \frac{1}{6}$ and the stop time is set as $t_{end} = 6000$ with 200 cells. In practical computation, in order to avoid the influence of round-off error, we set the minimum of h equals 10^{-11} in condition data. The sufficiently small number ϵ in positivity-preserving limiter is set as 10^{-11} too. The numerical solutions and the analytical solutions of the surface level are shown in Fig. 7. The minimum of the water height is shown in Table 4 and it certainly keeps the minimum of the water height. We can see that the numerical solution agree with the analytical solutions very well.

Example 5.5 The dam breaking problem over a rectangular bump

The purpose of this example is to show the capacity of the well-balanced schemes for the situations away from the steady states. The bottom function is

$$b(x) = \begin{cases} 8, & \text{if } \frac{4500}{8} \leq x \leq \frac{7500}{8}, \\ 0, & \text{otherwise,} \end{cases} \tag{5.14}$$

for $x \in [0, 1500]$. The initial conditions are:

$$(hu)(x, 0) = 0, \quad h(x, 0) = \begin{cases} 20 - b(x), & \text{if } x \leq 750, \\ 15 - b(x), & \text{otherwise.} \end{cases}$$

We test two cases: $t_{end} = 15$ and $t_{end} = 60$ and set $N = 400$. The reference solution is computed by the hydrostatic reconstruction scheme (3.12) on the uniform grid with $N = 4000$ cells. We plot the figure of surface level η in Figs. 8 and 9 with two schemes (3.12) and (3.15) respectively. It shows that our well-balanced methods still perform well when the solution is far away from the steady states.

5.2 Still Water Equilibrium in 2D

Example 5.6 Test for the well-balanced property

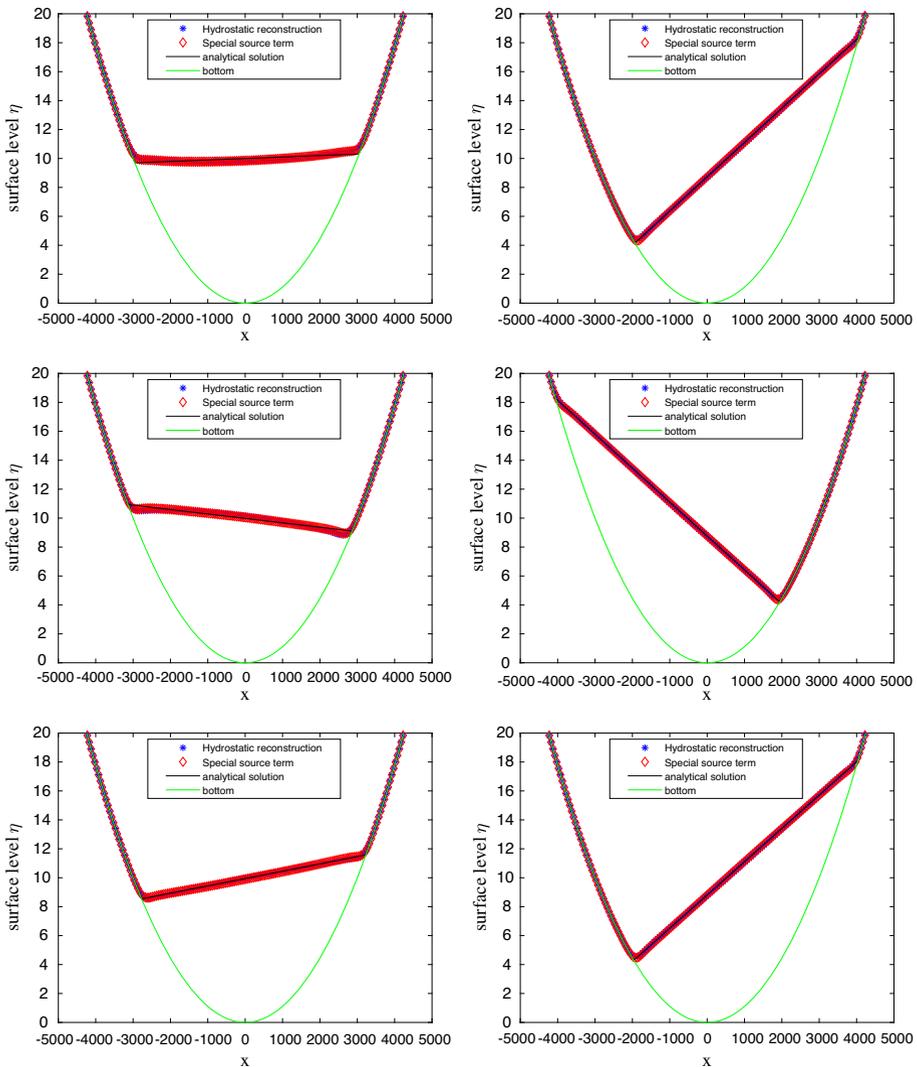


Fig. 7 (Example 5.4) The numerical surface level η using 200 cells compared with analytical solution and bottom b at different time. From top left to bottom right: $t = 1000, 2000, 3000, 4000, 5000, 6000$

This example is used to verify that our schemes indeed maintain the still water equilibrium for 2D shallow water equations. The two-dimensional bottom is given by

$$b(x, y) = 0.8e^{-50((x-0.5)^2+(y-0.5)^2)}, \quad x, y \in [0, 1] \tag{5.15}$$

The initial data is $h(x, y, 0) = 1 - b(x, y)$ and $hu(x, y, 0) = 0, hv(x, y, 0) = 0$. A static uniform criss-triangular mesh with 800 triangles is used at the beginning. We set stop time $t_{end} = 0.1$ and use double and quadruple precision separately. Tables 5 and 6 contain the L^1 errors for h and hu and hv . We can clearly see that the L^1 errors are at the level of round-off errors for different precision.

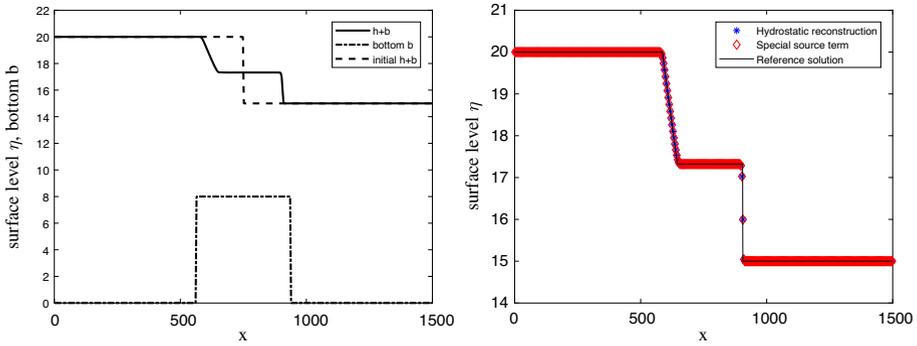


Fig. 8 (Example 5.5) The surface level η at time $t = 15$. Left: the numerical solution using hydrostatic reconstruction with $N = 400$, together with the initial condition and the bottom; right: the comparison between the numerical solution of two schemes and the reference solution

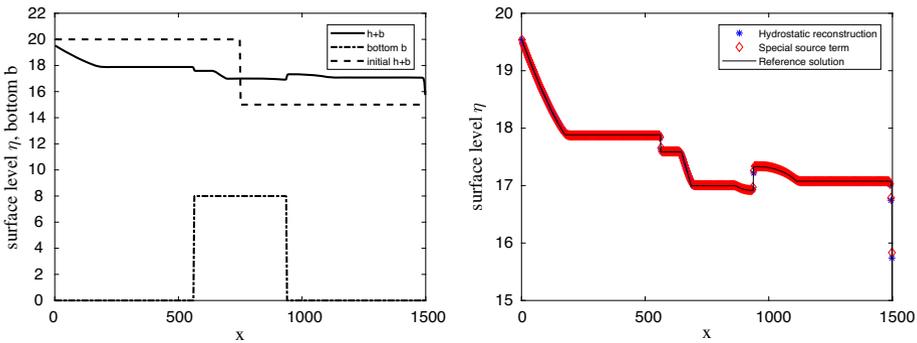


Fig. 9 (Example 5.5) The surface level η at time $t = 60$. Left: the numerical solution using hydrostatic reconstruction with $N = 400$, together with the initial condition and the bottom; right: the comparison between the numerical solution of two schemes and the reference solution

Table 5 (Example 5.6) Hydrostatic reconstruction, L^1 errors for different precision of the surface level η and the discharges hu and hv

Precision	L^1 error			L^∞ error		
	η	hu	hv	η	hu	hv
Double	3.87E-14	1.27E-13	7.14E-14	5.67E-12	1.94E-11	9.04E-12
Quadruple	7.23E-32	2.14E-31	1.38E-31	8.21E-30	3.19E-29	1.33E-29

Table 6 (Example 5.6) Special source term, L^1 errors for different precision of the surface level η and the discharges hu and hv

Precision	L^1 error			L^∞ error		
	η	hu	hv	η	hu	hv
Double	7.60E-14	2.17E-13	1.47E-13	9.30E-12	3.16E-11	1.75E-11
Quadruple	7.65E-32	2.26E-31	1.46E-31	8.21E-30	3.25E-29	1.42E-29

Table 7 (Example 5.7) L^1 errors and numerical orders of accuracy for hydrostatic reconstruction

N	η		hu		hv	
	L^1 error	Order	L^1 error	Order	L^1 error	Order
200	8.26E-03	–	4.16E-02	–	8.18E-02	–
800	9.79E-04	3.08	5.72E-03	2.86	1.02E-02	3.00
3200	8.12E-05	3.59	7.48E-04	2.94	1.06E-03	3.27
12800	7.97E-06	3.35	9.82E-05	2.93	1.21E-04	3.13
51200	9.14E-07	3.12	1.33E-05	2.88	1.52E-05	3.00

Table 8 (Example 5.7) L^1 errors and numerical orders of accuracy for special source term

N	η		hu		hv	
	L^1 error	Order	L^1 error	Order	L^1 error	Order
200	8.38E-03	–	5.06E-02	–	8.78E-02	–
800	1.01E-03	3.05	7.50E-03	2.76	1.11E-02	2.98
3200	7.53E-05	3.75	9.74E-04	2.94	1.07E-03	3.37
12800	6.86E-06	3.46	1.44E-04	2.76	1.20E-04	3.16
51200	7.78E-07	3.14	2.15E-05	2.74	1.59E-05	2.92

Example 5.7 Accuracy test in 2D

In this example we will show that our schemes have high order accuracy for a smooth solution. The bottom function and the initial data are given by:

$$\begin{aligned}
 b(x, y) &= \sin(2\pi x) + \cos(2\pi y), & h(x, y, 0) &= 10 + e^{\sin(2\pi x)} \cos(2\pi y), \\
 (hu)(x, y, 0) &= \sin(\cos(2\pi x)) \sin(2\pi y), & (hv)(x, y, 0) &= \cos(2\pi x) \cos(\sin(2\pi y)),
 \end{aligned}$$

where $(x, y) \in [0, 1] \times [0, 1]$ with periodic boundary conditions. We set stop time $t_{end} = 0.05$ when the solution is still smooth. We use the same fifth order WENO scheme with an extremely refined mesh consisting of 1600×1600 rectangle cells to compute a reference solution, according to [39]. Tables 7 and 8 contain the L^1 errors and orders. We see that ALE-DG schemes can achieve the optimal convergence rate.

Additionally, we compare the CPU time of our scheme (3.12) on the moving mesh defined in (5.2) and the static mesh in this example. We use 3200 triangle cells and the efficiency is comparable: the CPU time is 181.60s on the moving mesh and 178.75s on the static mesh.

Example 5.8 A small perturbation test

In this example, we will test the ability of our schemes to capture the small perturbation of the still water equilibrium in 2D. This is a classical example to show the capability of the well-balanced scheme for the perturbation of the equilibrium state, which was used in [25].

The initial data and the bottom function are given by:

$$h(x, y, 0) = \begin{cases} 1 - b(x, y) + 0.01, & \text{if } 0.05 \leq x \leq 0.15, \\ 1 - b(x, y), & \text{otherwise,} \end{cases} \tag{5.16}$$

$$hu(x, y, 0) = hv(x, y, 0) = 0, \tag{5.17}$$

Table 9 (Example 5.9) L^1 errors and numerical orders of accuracy of the ALE-DG solutions with piecewise P^2 polynomial basis

N	h		hu	
	L^1 error	Order	L^1 error	Order
25	8.43E-04	–	7.12E-03	–
50	9.95E-05	3.08	8.35E-04	3.09
100	9.89E-06	3.33	8.02E-05	3.38
200	1.14E-06	3.12	9.09E-06	3.14
400	1.41E-07	3.01	1.13E-06	3.00
800	1.78E-08	2.99	1.42E-07	3.00

$$b(x, y) = 0.5e^{-5(x-0.9)^2-50(y-0.5)^2} \tag{5.18}$$

where $(x, y) \in [0, 2] \times [0, 1]$ and the bottom is an isolated elliptical shaped hump. We show the numerical solution of surface level $h + b$ with $2 \times 160 \times 80$ criss-triangle cells at different times. Note that the wave speed is slower above the hump than elsewhere, leading to a distortion of the initially planar disturbance. In addition, different from the one-dimensional perturbation tests, the left-going pulse has already cleanly left the domain at the first time shown, due to the outflow boundary conditions. We can see that ALE-DG schemes can capture the small perturbation correctly (Fig. 10).

5.3 Moving Water Equilibrium in 1D

Example 5.9 Test for the accuracy

In this example we will test the high order accuracy of our schemes for a smooth solution. We have the following bottom function and initial conditions

$$b(x) = \sin^2(\pi x), \quad h(x, 0) = 5 + e^{\cos(2\pi x)}, \quad (hu)(x, 0) = \sin(\cos(2\pi x)), \quad x \in [0, 1]$$

with periodic boundary conditions. The exact solution is not known explicitly and we use the fifth order finite volume WENO scheme with $N = 12800$ cells to compute a reference solution [35] in computing the numerical errors. We compute up to $t_{end} = 0.1$ when the solution is still smooth. Table 9 shows the L^1 errors and numerical orders for u . It is found that ALE-DG scheme has the third order accuracy for piecewise P^2 polynomial basis.

Example 5.10 Test for the well-balanced property

The purpose is to verify that our schemes maintain the well-balanced property. We choose three different moving water equilibrium which are classical cases and have been widely used. The bottom function is given by:

$$b(x) = \begin{cases} 0.2 - 0.05(x - 10)^2, & \text{if } 8 \leq x \leq 12, \\ 0, & \text{otherwise,} \end{cases} \tag{5.19}$$

for a channel of length 25 meters. Three cases, subcritical or transcritical flow with or without a steady shock are in exactly equilibrium for the initial condition, which means E and m are constants:

- (a) Subcritical flow: The initial condition is $E = 22.06605$ and $m = 4.42$. The boundary condition is $m = 4.42$ at $x = 0$ and $h = 2$ at $x = 25$.

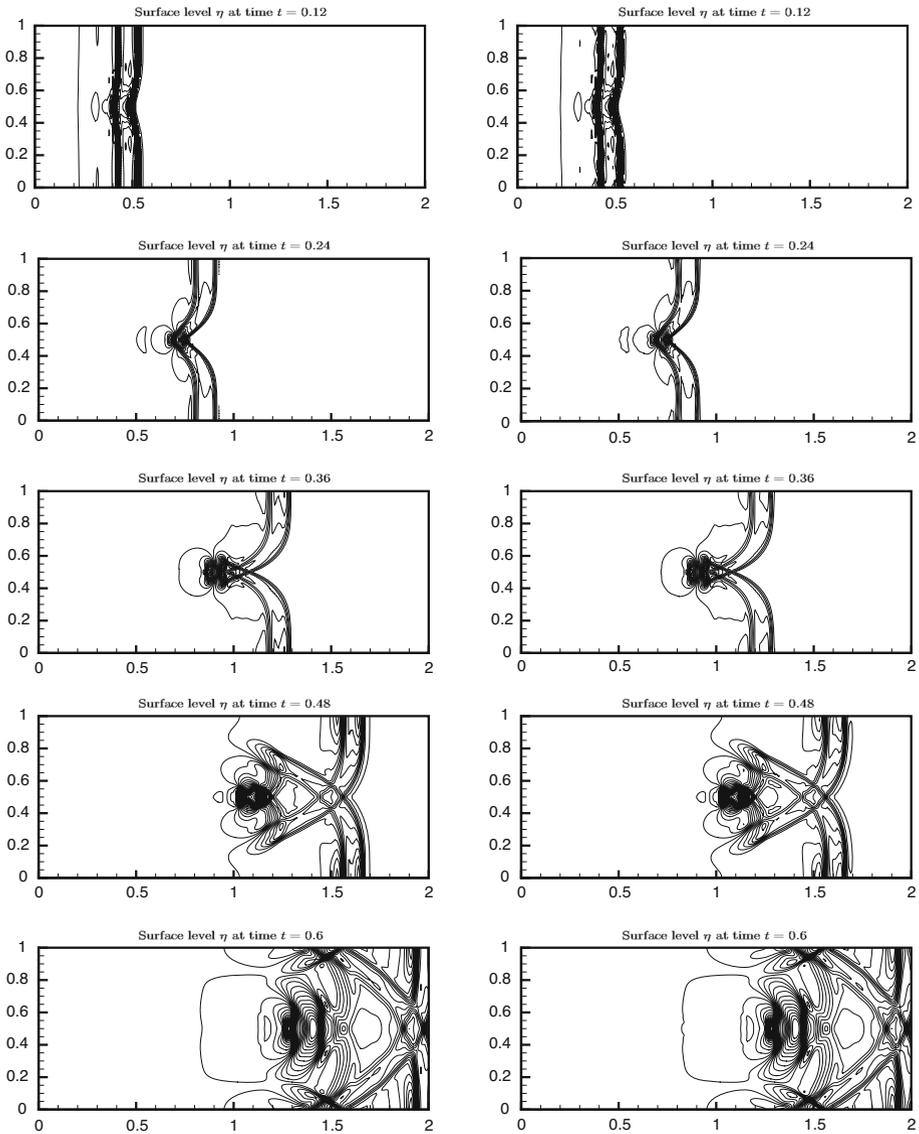


Fig. 10 (Example 5.8) Left: hydrostatic reconstruction; Right: special source term treatment. The contours of the surface level $h + b$ for Example 5.8 using a mesh of 25600 triangles. 30 uniformly spaced contour lines. From top to bottom: at time $t = 0.12$ from 0.99942 to 1.00656; at time $t = 0.24$ from 0.99318 to 1.01659; at time $t = 0.36$ from 0.98814 to 1.01161; at time $t = 0.48$ from 0.99023 to 1.00508; and at time $t = 0.6$ from 0.99514 to 1.00555

- (b) Transcritical flow without a shock: The initial condition is $E = 11.08071$ and $m = 1.53$. The boundary condition is $m = 1.53$ at $x = 0$ and $h = 0.66$ at $x = 25$ when the flow is subsonic.

Table 10 (Example 5.10) L^1 errors and L^∞ errors of the water height h and the discharge hu for moving water equilibrium

Test		L^∞ error		L^1 error	
		h	hu	h	hu
(a)	Double	2.34E-13	1.43E-12	4.86E-14	1.96E-13
	Quadruple	2.01E-31	1.32E-30	3.95E-32	1.66E-31
(b)	Double	1.51E-13	3.29E-13	5.75E-15	1.75E-14
	Quadruple	8.98E-32	3.36E-31	4.55E-33	1.52E-32
(c)	Double	5.82E-14	1.36E-13	2.99E-15	7.48E-15
	Quadruple	5.82E-32	1.84E-31	2.74E-33	7.03E-33

(c) Transcritical flow with a shock. The initial condition is $E = 4.15408$ when $x < 11.66550$ and $E = 3.38672$ when $x > 11.66550$ and $m = 0.18$. The boundary condition is $m = 0.18$ at $x = 0$ and $h = 0.33$ at $x = 25$.

Here we choose mesh to make sure that $x = 8, 12$ and $x = 11.66550$ if there is a shock are located at the cell boundary. For the first two cases, we choose uniform mesh at the beginning and

$$\begin{aligned}
 x_{j+\frac{1}{2}}(t) = & x_{j+\frac{1}{2}}(0) + \frac{1}{3(x_1 - x_0)^4} \sin\left(\frac{t}{t_{end}}\right) \\
 & (x_{j+\frac{1}{2}}(0) - 25) (x_{j+\frac{1}{2}}(0) - 8) (x_{j+\frac{1}{2}}(0) - 12) x_{j+\frac{1}{2}}(0).
 \end{aligned}
 \tag{5.20}$$

For the third case, since the equilibrium states on both sides of the shock are not the same, we recover two different equilibrium states and set the one of mesh grids exactly on the shock $x = 11.66550$ and assume this mesh grid unchanged over time. So we choose uniform mesh and shift the point near the shock to $x = 11.66550$ at the beginning and

$$\begin{aligned}
 x_{j+\frac{1}{2}}(t) = & x_{j+\frac{1}{2}}(0) + \frac{1}{3(x_1 - x_0)^5} \sin\left(\frac{t}{t_{end}}\right) \\
 & (x_{j+\frac{1}{2}}(0) - 25) (x_{j+\frac{1}{2}}(0) - 8) (x_{j+\frac{1}{2}}(0) - 11.66550) (x_{j+\frac{1}{2}}(0) - 12) x_{j+\frac{1}{2}}(0).
 \end{aligned}
 \tag{5.21}$$

We compute their solutions until $t_{end} = 5$ using $N = 200$ points. we show the L^1 and L^∞ errors for the water height h and the discharge hu in Table 10 with two different precision, where we can clearly see that the L^1 and L^∞ errors are at the level of round-off errors.

Example 5.11 A small perturbation test

In this example, we continue to test the three cases in Example 5.10 to demonstrate that our scheme can capture the small perturbation of moving water equilibrium correctly. Our initial conditions are almost the same as Example 5.10 that we additionally impose a small perturbation of size 0.05 on the h in the interval $[5.75, 6.25]$. We use transmissive boundary conditions. We take $N = 200$ cells and use the same meshes in Example 5.10. The stopping time $t_{end} = 1.5$ for the first two cases, $t_{end} = 3$ for the third case. We show the numerical results in Figs. 11, 12 and 13 and can clearly see that the propagated small perturbation is captured correctly for the both three cases.

Example 5.12 Riemann problem over a flat bottom

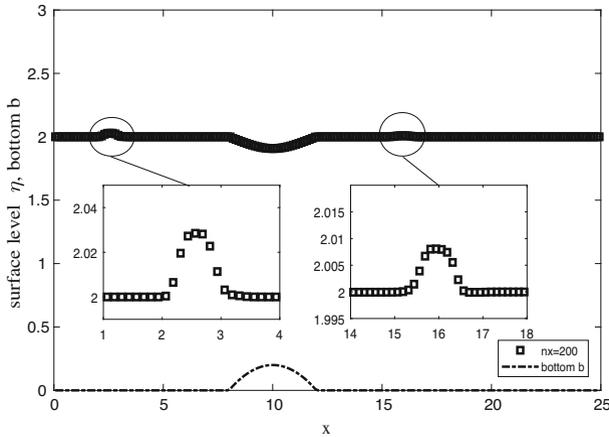


Fig. 11 (Example 5.11) Small perturbation of the subcritical flow (a) with 200 cells

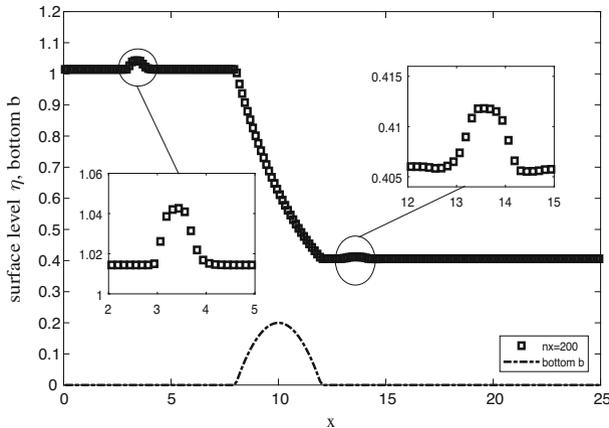


Fig. 12 (Example 5.11) Small perturbation of the transcritical flow without a shock (b) with 200 cells

In this example, we want to demonstrate that our ALE-DG well-balanced methods for moving water equilibrium have the positivity-preserving property. We consider Riemann problem containing dry area over a flat bottom, which means $b(x) \equiv 0$. This example was used in [40]. The initial conditions are given by

$$hu(x) = 0, \quad h(x) = \begin{cases} 10 & \text{if } x \leq 0, \\ 0 & \text{otherwise.} \end{cases} \tag{5.22}$$

We set the computational domain as $[-300,300]$ with transmissive boundary conditions and 200 uniform cells. The analytical solutions can be found in [5]. In practical computation, in order to avoid the influence of round-off error, we set the minimum of h equals 10^{-11} in condition data. The sufficiently small number ϵ in positivity-preserving limiter is set as 10^{-11} too. We plot the numerical solution of the water height h at time $t = 4, 8, 12$ compared with the analytical solutions in Fig. 14. We can see our scheme keeps the height positive and

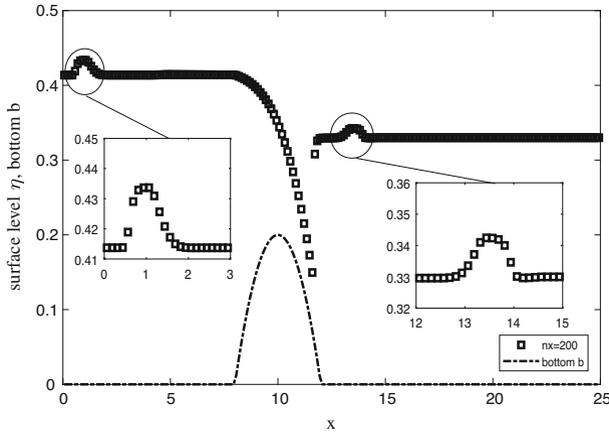


Fig. 13 (Example 5.11) Small perturbation of the transcritical flow with a shock (c) with 200 cells

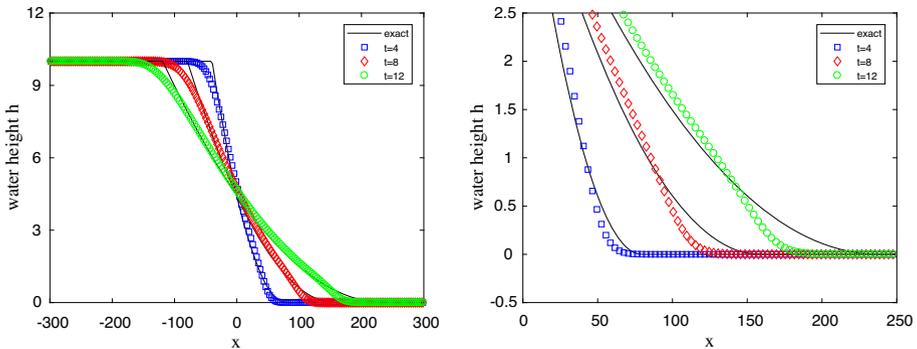


Fig. 14 (Example 5.12) The numerical solution of water height h compared with exact solutions for the Riemann problem. Left: water height h ; Right: the zoom in of the left figure for $x \in [0, 250]$, where water height h is near 0

Table 11 (Example 5.12) The minimum of the water height h at time $t = 4, 8, 12$

Time t	4	8	12
Minimum of h	1.00E-11	1.00E-11	1.00E-11

captures the exact solution well. The minimum of the water height is shown in Table 11 and it certainly keeps the minimum of the water height positive.

6 Conclusion

In this paper we developed the well-balanced ALE-DG methods for both still and moving water equilibria of the shallow water equations. For the ALE-DG methods, the well-balanced property is based on the GCL and corresponding modification of time discretization, besides the techniques of DG methods on static grids. For the still water equilibrium, the well-balanced ALE-DG method can be extended to the two-dimensional case. And the well-

balanced scheme for the moving water equilibrium is only for the one-dimensional case, since there is no general form of moving water equilibrium in two dimensions. Numerical examples have been given to show the well-balanced property, positivity-preserving property and high order accuracy. In this work, we mainly focus on the scheme design on the moving mesh. The grid movement in all the numerical examples is prescribed and is not derived from the computed solution. The performance of these schemes in applications is also dependent on the mesh optimization for the specific problem and its bottom topography due to the “time-dependent” approximation of the bottom. The methodology of mesh adaptation will be considered in our future work on applications. These techniques in moving water equilibrium part can also be extended to many other balance laws, such as Euler equations etc.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

A Proofs for the Still Water Equilibrium Schemes

In order to show the main idea of the proof, we only show the proofs of the properties in Sect. 3 in one dimensional case. The two-dimensional proofs are similar and we omit the detail for simplicity. In one dimensional case, the flux term is defined by

$$G(\omega, v) = f(v) - \omega v, \text{ with } f(v) = \left(\frac{(hu)^2}{\eta - b} + \frac{1}{2}g(\eta - b)^2 \right).$$

A.1 Proof of Lemma 3.1 for Scheme (3.12)

We first simplify the numerical flux $\hat{G}^{l,r}$. We denote the two components of the numerical flux \hat{G} by $\hat{G}^{[1]}$ and $\hat{G}^{[2]}$.

We write the specific form of $\hat{G}^{[1],l}$ as follows

$$\hat{G}^{[1],l} = \hat{G}^{[1]}(\omega, v^{*, -}, v^{*, +}) + \frac{1}{2}((\omega + \alpha)(\eta^{*, +} - \eta^+) + (\omega - \alpha)(\eta^{*, -} - \eta^-)) \tag{A.1}$$

$$\begin{aligned} &= \frac{1}{2}(hu^- - \omega\eta^{*, -} + hu^- - \omega\eta^{*, +} - \alpha(\eta^{*, +} - \eta^{*, -}) \\ &\quad + (\omega + \alpha)(\eta^{*, +} - \eta^+) + (\omega - \alpha)(\eta^{*, -} - \eta^-)) \\ &= \frac{1}{2}(hu^- - \omega\eta^- + hu^- - \omega\eta^+ - \alpha(\eta^+ - \eta^-)) = \hat{G}^{[1]}(\omega, v^-, v^+). \end{aligned} \tag{A.2}$$

Similarly we have

$$\hat{G}^{[1],r} = \hat{G}^{[1]}(\omega, v^-, v^+) = \hat{G}^{[1],r}. \tag{A.3}$$

Here without causing conflict, we have omitted the subscript $j + \frac{1}{2}$ and we note that we don't use the assumption of still water equilibrium.

Next, since v equals the still water equilibrium, i.e. $h(x) + b(x) = \text{constant}$ and $hu(x) = 0$. We have $\eta_{j+\frac{1}{2}}^+ = \eta_{j+\frac{1}{2}}^-$ and $(hu)_{j+\frac{1}{2}}^\pm = 0$. For the numerical flux of the first equation, we

use (A.3) to get

$$\hat{\mathbf{G}}_{j+\frac{1}{2}}^{[1],l} = \hat{\mathbf{G}}_{j+\frac{1}{2}}^{[1],r} = \hat{\mathbf{G}} \left(\omega_{j+\frac{1}{2}}, \mathbf{v}_{j+\frac{1}{2}}^-, \mathbf{v}_{j+\frac{1}{2}}^+ \right) = \mathbf{G}^{[1]} \left(\omega_{j+\frac{1}{2}}, \mathbf{v}_{j+\frac{1}{2}}^- \right). \tag{A.4}$$

For the numerical flux of the second equation, we have

$$\begin{aligned} \hat{\mathbf{G}}_{j+\frac{1}{2}}^{[2],l} &= \hat{\mathbf{G}}^{[2]} \left(\omega_{j+\frac{1}{2}}, \mathbf{v}_{j+\frac{1}{2}}^{*,-}, \mathbf{v}_{j+\frac{1}{2}}^{*,+} \right) + \frac{g}{2} \left(\eta_{j+\frac{1}{2}}^- - b_{j+\frac{1}{2}}^- \right)^2 - \frac{g}{2} \left(h_{j+\frac{1}{2}}^{*,-} \right)^2 \\ &= \frac{1}{2} \left(\mathbf{G}(\omega_{j+\frac{1}{2}}, \mathbf{v}_{j+\frac{1}{2}}^{*,-}) + \mathbf{G}(\omega_{j+\frac{1}{2}}, \mathbf{v}_{j+\frac{1}{2}}^{*,+}) \right) + \frac{g}{2} \left(\eta_{j+\frac{1}{2}}^- - b_{j+\frac{1}{2}}^- \right)^2 - \frac{g}{2} \left(h_{j+\frac{1}{2}}^{*,-} \right)^2 \\ &= \frac{1}{2} \left(\frac{g}{2} \left(\eta_{j+\frac{1}{2}}^{*,-} - b_{j+\frac{1}{2}}^- \right)^2 + \frac{g}{2} \left(\eta_{j+\frac{1}{2}}^{*,+} - b_{j+\frac{1}{2}}^+ \right)^2 \right) + \frac{g}{2} \left(\eta_{j+\frac{1}{2}}^- - b_{j+\frac{1}{2}}^- \right)^2 - \frac{g}{2} \left(h_{j+\frac{1}{2}}^{*,-} \right)^2 \\ &= \frac{g}{2} \left(h_{j+\frac{1}{2}}^{*,-} \right)^2 + \frac{g}{2} \left(\eta_{j+\frac{1}{2}}^- - b_{j+\frac{1}{2}}^- \right)^2 - \frac{g}{2} \left(h_{j+\frac{1}{2}}^{*,-} \right)^2 \\ &= \frac{g}{2} \left(\eta_{j+\frac{1}{2}}^- - b_{j+\frac{1}{2}}^- \right)^2 = \mathbf{G}^{[2]} \left(\omega_{j+\frac{1}{2}}, \mathbf{v}_{j+\frac{1}{2}}^- \right). \end{aligned} \tag{A.5}$$

Here we use the fact that $hu_{j+\frac{1}{2}}^- = hu_{j+\frac{1}{2}}^+ = 0$. Similarly, we have

$$\hat{\mathbf{G}}_{j+\frac{1}{2}}^{[2],r} = \mathbf{G}^{[2]} \left(\omega_{j+\frac{1}{2}}, \mathbf{v}_{j+\frac{1}{2}}^+ \right). \tag{A.6}$$

So we combine the results (A.2)–(A.6) and have

$$\begin{aligned} \hat{\mathbf{G}}_{j+\frac{1}{2}}^l &= \mathbf{G} \left(\omega_{j+\frac{1}{2}}, \mathbf{v}_{j+\frac{1}{2}}^- \right) = \left(\begin{array}{c} \omega_{j+\frac{1}{2}} \eta_{j+\frac{1}{2}}^- \\ \frac{g}{2} \left(\eta_{j+\frac{1}{2}}^- - b_{j+\frac{1}{2}}^- \right)^2 \end{array} \right), \\ \hat{\mathbf{G}}_{j-\frac{1}{2}}^r &= \mathbf{G} \left(\omega_{j-\frac{1}{2}}, \mathbf{v}_{j-\frac{1}{2}}^+ \right) = \left(\begin{array}{c} \omega_{j-\frac{1}{2}} \eta_{j-\frac{1}{2}}^+ \\ \frac{g}{2} \left(\eta_{j-\frac{1}{2}}^+ - b_{j-\frac{1}{2}}^+ \right)^2 \end{array} \right). \end{aligned}$$

Thus we have

$$\begin{aligned} \mathcal{L}_j^h(\omega, \mathbf{v}, \phi, t) &= \int_{K_j(t)} \left(\begin{array}{c} -\omega \eta \\ \frac{g}{2} (\eta - b)^2 \end{array} \right) \cdot \phi_x dx - \left(\begin{array}{c} \omega_{j+\frac{1}{2}} \eta_{j+\frac{1}{2}}^- \\ \frac{g}{2} (\eta_{j+\frac{1}{2}}^- - b_{j+\frac{1}{2}}^-)^2 \end{array} \right) \cdot \phi_{j+\frac{1}{2}}^- \\ &\quad + \left(\begin{array}{c} \omega_{j-\frac{1}{2}} \eta_{j-\frac{1}{2}}^+ \\ \frac{g}{2} (\eta_{j-\frac{1}{2}}^+ - b_{j-\frac{1}{2}}^+)^2 \end{array} \right) \cdot \phi_{j-\frac{1}{2}}^+ + \int_{K_j(t)} \left(\begin{array}{c} 0 \\ -g(\eta - b)b_x \end{array} \right) \cdot \phi dx \\ &= - \int_{K_j(t)} \partial_x \left(\begin{array}{c} -\omega \eta \\ \frac{g}{2} (\eta - b)^2 \end{array} \right) \cdot \phi dx + \int_{K_j(t)} \left(\begin{array}{c} 0 \\ -g(\eta - b)b_x \end{array} \right) \cdot \phi dx \\ &= \int_{K_j(t)} \partial_x \left(\begin{array}{c} \omega \eta \\ 0 \end{array} \right) \cdot \phi dx \\ &= \int_{K_j(t)} (\omega \mathbf{v})_x \cdot \phi dx. \end{aligned}$$

where we use the integration by parts to cancel the boundary interface terms. This finishes the proof.

A.2 Proof of Lemma 3.1 for Scheme (3.15)

Since \mathbf{v} equals the still water equilibrium, i.e. \mathbf{v} is constant vector, we have

$$\mathbf{v}^-_{j+\frac{1}{2}} = \mathbf{v}^+_{j+\frac{1}{2}}.$$

Because $\widehat{\mathbf{G}}$ is a monotone flux, then we have

$$\begin{aligned} \widehat{\mathbf{G}}_{j+\frac{1}{2}} &= \frac{1}{2} \left(\mathbf{G} \left(\omega_{j+\frac{1}{2}}, \mathbf{v}^-_{j+\frac{1}{2}} \right) + \mathbf{G} \left(\omega_{j+\frac{1}{2}}, \mathbf{v}^+_{j+\frac{1}{2}} \right) \right) \\ &= \frac{1}{2} \left(\mathbf{f} \left(\mathbf{v}^-_{j+\frac{1}{2}} \right) + \mathbf{f} \left(\mathbf{v}^+_{j+\frac{1}{2}} \right) \right) - \omega_{j+\frac{1}{2}} \mathbf{v}^-_{j+\frac{1}{2}}. \end{aligned} \tag{A.7}$$

Similarly we have

$$\widehat{\mathbf{G}}_{j-\frac{1}{2}} = \frac{1}{2} \left(\mathbf{f} \left(\mathbf{v}^-_{j-\frac{1}{2}} \right) + \mathbf{f} \left(\mathbf{v}^+_{j-\frac{1}{2}} \right) \right) - \omega_{j-\frac{1}{2}} \mathbf{v}^+_{j-\frac{1}{2}}. \tag{A.8}$$

Then we can compute that

$$\begin{aligned} &\mathcal{L}_j^S(\omega, \mathbf{v}, \phi, t) \\ &= \int_{K_j(t)} -\omega \mathbf{v} \cdot \phi_x \, dx + \omega_{j+\frac{1}{2}} \mathbf{v}^-_{j+\frac{1}{2}} \cdot \phi^-_{j+\frac{1}{2}} - \omega_{j-\frac{1}{2}} \mathbf{v}^+_{j-\frac{1}{2}} \cdot \phi^+_{j-\frac{1}{2}} \\ &\quad + \int_{K_j(t)} \mathbf{f}(\mathbf{v}) \cdot \phi_x \, dx - \frac{1}{2} \left(\mathbf{f} \left(\mathbf{v}^-_{j-\frac{1}{2}} \right) \right. \\ &\quad \left. + \mathbf{f} \left(\mathbf{v}^+_{j-\frac{1}{2}} \right) \right) \cdot \phi^-_{j+\frac{1}{2}} + \frac{1}{2} \left(\mathbf{f} \left(\mathbf{v}^-_{j-\frac{1}{2}} \right) + \mathbf{f} \left(\mathbf{v}^+_{j-\frac{1}{2}} \right) \right) \cdot \phi^+_{j-\frac{1}{2}} + s_j \\ &= \int_{K_j(t)} (\omega \mathbf{v})_x \cdot \phi \, dx + \int_{K_j(t)} \frac{1}{2} g(\eta - b)^2 \psi_x \, dx \\ &\quad - \frac{g(\eta - b^-_{j+\frac{1}{2}})^2 + g(\eta - b^+_{j+\frac{1}{2}})^2}{4} \psi^-_{j+\frac{1}{2}} + \frac{g(\eta - b^-_{j-\frac{1}{2}})^2 + g(\eta - b^+_{j-\frac{1}{2}})^2}{4} \psi^+_{j-\frac{1}{2}} + s_j \\ &= \int_{K_j(t)} (\omega \mathbf{v})_x \cdot \phi \, dx + \int_{K_j(t)} \left(\frac{1}{2} g \eta^2 + \frac{1}{2} g b^2 - g \eta b \right) \psi_x \, dx \\ &\quad - \left(\frac{1}{2} g \eta^2 + \frac{1}{4} g \left(b^-_{j+\frac{1}{2}} \right)^2 + \frac{1}{4} g \left(b^+_{j+\frac{1}{2}} \right)^2 - \frac{1}{2} g \eta b^-_{j+\frac{1}{2}} - \frac{1}{2} g \eta b^+_{j+\frac{1}{2}} \right) \psi^-_{j+\frac{1}{2}} \\ &\quad + \left(\frac{1}{2} g \eta^2 + \frac{1}{4} g \left(b^-_{j-\frac{1}{2}} \right)^2 + \frac{1}{4} g \left(b^+_{j-\frac{1}{2}} \right)^2 - \frac{1}{2} g \eta b^-_{j-\frac{1}{2}} - \frac{1}{2} g \eta b^+_{j-\frac{1}{2}} \right) \psi^+_{j-\frac{1}{2}} \\ &\quad + \int_{K_j(t)} \left(-\frac{1}{2} g b^2 + g \eta b \right) \psi_x \, dx \end{aligned}$$

$$\begin{aligned}
 & - \left(-\frac{1}{4}g \left(b_{j+\frac{1}{2}}^- \right)^2 - \frac{1}{4}g \left(b_{j+\frac{1}{2}}^+ \right)^2 + \frac{1}{2}g\eta b_{j+\frac{1}{2}}^- + \frac{1}{2}g\eta b_{j+\frac{1}{2}}^+ \right) \psi_{j+\frac{1}{2}}^- \\
 & + \left(-\frac{1}{4}g \left(b_{j-\frac{1}{2}}^- \right)^2 - \frac{1}{4}g \left(b_{j-\frac{1}{2}}^+ \right)^2 + \frac{1}{2}g\eta b_{j-\frac{1}{2}}^- + \frac{1}{2}g\eta b_{j-\frac{1}{2}}^+ \right) \psi_{j-\frac{1}{2}}^+ \\
 & = \int_{K_j(t)} (\omega \mathbf{v})_x \cdot \boldsymbol{\phi} \, dx + \int_{K_j(t)} \frac{1}{2}g\eta^2 \psi_x \, dx - \frac{1}{2}g\eta^2 \psi_{j+\frac{1}{2}}^- + \frac{1}{2}g\eta^2 \psi_{j-\frac{1}{2}}^+ \\
 & = \int_{K_j(t)} (\omega \mathbf{v})_x \cdot \boldsymbol{\phi} \, dx + \int_{K_j(t)} \left(\frac{1}{2}g\eta^2 \right)_x \psi \, dx = \int_{K_j(t)} (\omega \mathbf{v})_x \cdot \boldsymbol{\phi} \, dx.
 \end{aligned}$$

The second equality is due to $hu = 0$ and $\eta = \text{constant}$. The sixth equality uses the fact $\eta = \text{constant}$.

A.3 Proof of Proposition 3.2

It is sufficient and necessary to prove that if \mathbf{v} are the still water equilibrium at time $t = 0$, i.e.

$$\eta^0(x) = \text{constant}, \quad \text{and} \quad (hu)^0(x) = 0, \tag{A.9}$$

then the numerical solution at any time t^n has to be

$$\eta^n(x) = \text{constant}, \quad \text{and} \quad (hu)^n(x) = 0. \tag{A.10}$$

Now we will use induction to prove the proposition.

Basic Step: Suppose that the initial data \mathbf{v}^0 are the still water equilibrium, i.e. $\eta^0(x) = \text{constant}$ and $(hu)^0(x) = 0$. So \mathbf{v}^0 is true.

Inductive Step: Now we assume the truth of \mathbf{v}^k , for some $k \in \mathbb{N}$ —that is, we assume that

$$\eta^k(x) = \text{constant}, \quad (hu)^k(x) = 0, \tag{A.11}$$

is a true state. From this assumption we want to deduce that

$$\eta^{k+1}(x) = \text{constant}, \quad (hu)^{k+1}(x) = 0. \tag{A.12}$$

In each time step, we use the Euler forward scheme (3.20). For both two schemes, we can use Lemma 3.1 to simplify the equation in (3.20) and get

$$\begin{aligned}
 \int_{K_j^{k+1}} \mathbf{v}^{k+1} \cdot \boldsymbol{\phi}^{k+1} \, dx & = \int_{K_j^k} \mathbf{v}^k \cdot \boldsymbol{\phi}^k \, dx + \Delta t \int_{K_j^k} (\omega^k \mathbf{v}^k)_x \cdot \boldsymbol{\phi}^k \, dx \\
 & = \mathbf{v}^k \cdot \left(\int_{K_j^k} \boldsymbol{\phi}^k \, dx + \Delta t \int_{K_j^k} (\omega^k)_x \boldsymbol{\phi}^k \, dx \right) \\
 & = \mathbf{v}^k \cdot \int_{K_j^{k+1}} \boldsymbol{\phi}^{k+1} \, dx \\
 & = \int_{K_j^{k+1}} \mathbf{v}^k \cdot \boldsymbol{\phi}^{k+1} \, dx,
 \end{aligned} \tag{A.13}$$

where the second equality holds since \mathbf{v}^k is a constant vector and the third equality is due to the dGCL in [23].

So we prove that $\mathbf{v}^{k+1} = \mathbf{v}^k$, which is also the still water equilibrium, i.e.

$$\eta^{k+1} = \text{constant}, \quad (hu)^{k+1} = 0.$$

This claim means that the numerical solution \mathbf{v}^n for $n \in \mathbb{N}$ is in the case of still water equilibrium, thus the fully discrete schemes are well-balanced.

A.4 Proof of Theorem 3.4

According to the results in (A.2) and (A.3), we simplify the numerical flux in (3.12) to get that the numerical scheme for the first equation in (3.12) and (3.15) are the same. Then according to the first equation of (3.12) or (3.15), we write the equation satisfied by the cell average of surface level η :

$$\frac{\Delta_j^{n+1}}{\Delta_j^n} \bar{\eta}_j^{n+1} = \bar{\eta}_j^n - \lambda \left(\hat{\mathbf{G}}_{j+\frac{1}{2}}^{[1]} + \hat{\mathbf{G}}_{j-\frac{1}{2}}^{[1]} \right). \tag{A.14}$$

where $\hat{\mathbf{G}}^{[1]}$ denote the first component of $\hat{\mathbf{G}}$:

$$\hat{\mathbf{G}}_{j+\frac{1}{2}}^{[1]} = \frac{1}{2} \left((hu)_{j+\frac{1}{2}}^{n,+} - \omega_{j+\frac{1}{2}} \eta_{j+\frac{1}{2}}^{n,+} + (hu)_{j+\frac{1}{2}}^{n,-} - \omega_{j+\frac{1}{2}} \eta_{j+\frac{1}{2}}^{n,-} - \alpha(\eta_{j+\frac{1}{2}}^{n,+} - \eta_{j+\frac{1}{2}}^{n,-}) \right), \tag{A.15}$$

and $\lambda = \frac{\Delta t}{\Delta_j^n}$. Moreover, we write out the equation satisfied by the cell average of bottom topography b :

$$\frac{\Delta_j^{n+1}}{\Delta_j^n} \bar{b}_j^{n+1} = \bar{b}_j^n + \lambda \left(\widehat{\omega b}_{j+\frac{1}{2}} - \widehat{\omega b}_{j-\frac{1}{2}} \right). \tag{A.16}$$

We denote $\eta_j - b_j$ by h_j . We firstly decompose \bar{h}_j^{n+1} in (A.14) into two parts:

$$\frac{\Delta_j^{n+1}}{\Delta_j^n} \bar{h}_j^{n+1} = W_1 + W_2, \tag{A.17}$$

with

$$\begin{aligned} W_1 &= \bar{b}_j^n - \frac{\Delta_j^{n+1}}{\Delta_j^n} \bar{b}_j^{n+1} + \lambda \left(\widehat{\omega b}_{j+\frac{1}{2}} - \widehat{\omega b}_{j-\frac{1}{2}} \right), \\ W_2 &= \bar{h}_j^n - \lambda \left(\left(\hat{\mathbf{G}}_{j+\frac{1}{2}}^{[1]} + \widehat{\omega b}_{j+\frac{1}{2}} \right) - \left(\hat{\mathbf{G}}_{j-\frac{1}{2}}^{[1]} + \widehat{\omega b}_{j-\frac{1}{2}} \right) \right). \end{aligned}$$

We note that $W_1 = 0$. This is because the definition of b in (A.16). So we only need to prove $W_2 > 0$.

Since we use the Lax-Friedrichs flux in this paper, we first simplify $\hat{\mathbf{G}}_{j+\frac{1}{2}}^{[1]} + \widehat{\omega b}_{j+\frac{1}{2}}$ by combining (A.15) and (3.22):

$$\begin{aligned} \hat{\mathbf{G}}_{j+\frac{1}{2}}^{[1]} + \widehat{\omega b}_{j+\frac{1}{2}} &= \frac{1}{2} \left((hu)_{j+\frac{1}{2}}^{n,+} - \omega_{j+\frac{1}{2}} h_{j+\frac{1}{2}}^{n,+} + (hu)_{j+\frac{1}{2}}^{n,-} - \omega_{j+\frac{1}{2}} h_{j+\frac{1}{2}}^{n,-} - \alpha(h_{j+\frac{1}{2}}^{n,+} - h_{j+\frac{1}{2}}^{n,-}) \right) \\ &= \frac{1}{2} \left(\left(u_{j+\frac{1}{2}}^{n,+} - \omega_{j+\frac{1}{2}} - \alpha \right) h_{j+\frac{1}{2}}^{n,+} + \left(u_{j+\frac{1}{2}}^{n,-} - \omega_{j+\frac{1}{2}} + \alpha \right) h_{j+\frac{1}{2}}^{n,-} \right) \end{aligned}$$

$$< \frac{1}{2} \left((\alpha - \alpha) h_{j+\frac{1}{2}}^{n,+} + (\alpha + \alpha) h_{j+\frac{1}{2}}^{n,-} \right) = \alpha h_{j+\frac{1}{2}}^{n,-}.$$

The first equality is due to $\hat{G}_{j+\frac{1}{2}}^{[1]}$ and $\widehat{\omega b}_{j+\frac{1}{2}}$ use the same α . The inequality is due to the definition of α :

$$-\alpha < u_{j+\frac{1}{2}}^{n,+} - \omega_{j+\frac{1}{2}} < \alpha.$$

Similarly we have

$$\hat{G}_{j-\frac{1}{2}}^{[1]} + \widehat{\omega b}_{j-\frac{1}{2}} > -\alpha h_{j-\frac{1}{2}}^{n,+}.$$

So we use the Gauss quadrature rule and above inequalities to get

$$\begin{aligned} W_2 &= \sum_{v=1}^L \hat{\omega}_v h_j^n(\hat{x}_j^{(v)}) - \lambda \left(\left(\hat{G}_{j+\frac{1}{2}}^{[1]} + \widehat{\omega b}_{j+\frac{1}{2}} \right) - \left(\hat{G}_{j-\frac{1}{2}}^{[1]} + \widehat{\omega b}_{j-\frac{1}{2}} \right) \right) \\ &> \sum_{v=1}^L \hat{\omega}_v h_j^n(\hat{x}_j^{(v)}) - \alpha \lambda h_{j-\frac{1}{2}}^{n,+} - \alpha \lambda h_{j+\frac{1}{2}}^{n,-} \\ &= \sum_{v=2}^{L-1} \hat{\omega}_v h_j^n(\hat{x}_j^{(v)}) + (\hat{\omega}_1 - \alpha \lambda) h_{j-\frac{1}{2}}^{n,+} - (\hat{\omega}_L - \alpha \lambda) h_{j+\frac{1}{2}}^{n,-} \\ &> 0. \end{aligned}$$

Notice that $\hat{\omega}_1 = \hat{\omega}_L$ and $\lambda < \frac{1}{\hat{\alpha}_0} = \frac{\hat{\omega}_1}{\alpha}$. Combined with the assumption $h(x) > 0$, for $x \in \mathbb{S}_{K^n}$, we proved that $W_2 > 0$.

B Proofs for the Moving Water Equilibrium Scheme

B.1 Proof of Lemma 4.2

We first prove that $\hat{g}(\omega_{j+\frac{1}{2}}, \mathbf{u}_{j+\frac{1}{2}}^{*,-}, \mathbf{u}_{j+\frac{1}{2}}^{*,+}) = \mathbf{g}(\omega_{j+\frac{1}{2}}, \mathbf{u}_{j+\frac{1}{2}}^{d,-}) = \mathbf{g}(\omega_{j+\frac{1}{2}}, \mathbf{u}_{j+\frac{1}{2}}^{d,-})$. Under such assumption, we have

$$\mathbf{u}^r = 0, \quad \mathbf{u}^* = \mathbf{u}^d. \tag{B.1}$$

With stationary shocks of \mathbf{u}^d and the assumption $\omega_{j+\frac{1}{2}} = 0$ at the shock, we can calculate that

$$\hat{g} \left(\omega_{j+\frac{1}{2}}, \mathbf{u}_{j+\frac{1}{2}}^{*,-}, \mathbf{u}_{j+\frac{1}{2}}^{*,+} \right) = \hat{g} \left(\omega_{j+\frac{1}{2}}, \mathbf{u}_{j+\frac{1}{2}}^{d,-}, \mathbf{u}_{j+\frac{1}{2}}^{d,+} \right) = \mathbf{g} \left(\omega_{j+\frac{1}{2}}, \mathbf{u}_{j+\frac{1}{2}}^{d,-} \right) = \mathbf{g} \left(\omega_{j+\frac{1}{2}}, \mathbf{u}_{j+\frac{1}{2}}^{d,-} \right). \tag{B.2}$$

The first equality is due to (B.1). The second and the third equality is due to $\omega_{j+\frac{1}{2}} = 0$, the property of Roe’s flux that it can exactly solve the Riemann problem [35], and the shock at point $x_{j+\frac{1}{2}}$ is a stationary shock.

With the continuity of \mathbf{u}^d , we have $\mathbf{u}_{j+\frac{1}{2}}^{*,\pm} = \mathbf{u}_{j+\frac{1}{2}}^{d,\pm}$ and we have the same conclusion with (B.2) according to the property of the Lax-Friedrichs flux. This means that for all the

numerical fluxes,

$$\hat{g}_{j+\frac{1}{2}}^l = f(u_{j+\frac{1}{2}}^-) - \omega_{j+\frac{1}{2}} u_{j+\frac{1}{2}}^{d,-} \quad \text{and} \quad \hat{g}_{j-\frac{1}{2}}^r = f(u_{j-\frac{1}{2}}^+) - \omega_{j-\frac{1}{2}} u_{j-\frac{1}{2}}^{d,+}.$$

Now, we have

$$\begin{aligned} \mathcal{L}_j^m(\omega, \mathbf{u}, \phi, t) &= \int_{K_j(t)} \mathbf{g}(\omega, \mathbf{u}) \cdot \phi_x dx - \left(f(u_{j+\frac{1}{2}}^-) - \omega_{j+\frac{1}{2}} u_{j+\frac{1}{2}}^{d,-} \right) \cdot \phi_{j+\frac{1}{2}}^- \\ &\quad + \left(f(u_{j-\frac{1}{2}}^+) - \omega_{j-\frac{1}{2}} u_{j-\frac{1}{2}}^{d,+} \right) \cdot \phi_{j-\frac{1}{2}}^+ - \int_{K_j(t)} f(u^e) \cdot \phi_x dx \\ &\quad + f(u_{j+\frac{1}{2}}^{e,-}) \cdot \phi_{j+\frac{1}{2}}^- - f(u_{j-\frac{1}{2}}^{e,+}) \cdot \phi_{j-\frac{1}{2}}^+ \\ &= - \int_{K_j(t)} \omega \mathbf{u} \cdot \phi_x dx + \omega_{j+\frac{1}{2}} u_{j+\frac{1}{2}}^{d,-} \cdot \phi_{j+\frac{1}{2}}^- - \omega_{j-\frac{1}{2}} u_{j-\frac{1}{2}}^{d,+} \cdot \phi_{j-\frac{1}{2}}^+. \end{aligned}$$

This is because of the assumption $\mathbf{u} = \mathbf{u}^e$ and we finish the proof.

B.2 Proof of Proposition 4.4

Similar to previous proof in Proposition 3.2, we will use induction to prove the proposition.

Basic Step: Suppose that \mathbf{u}^d is the initial condition. Then we have $\mathbf{u}^0 = P\mathbf{u}^d$ and $\mathbf{u}^{e,0} = P\mathbf{u}^d$, which will lead to $\mathbf{u}^0 = \mathbf{u}^{e,0}$. So \mathbf{u}^0 is true.

Inductive Step: Now we assume the truth of \mathbf{u}^k , for some $k \in \mathbb{N}$ —that is, we assume that $\mathbf{u}^k = \mathbf{u}^{e,k}$ is a true state. From this assumption we want to deduce that $\mathbf{u}^{k+1} = \mathbf{u}^{e,k+1}$.

Since $\mathbf{u}^k = \mathbf{u}^{e,k}$, we have $\mathbf{u}^{r,k} = 0$. Use Lemma 4.2, we have

$$\mathcal{L}_j^m(\omega^k, \mathbf{u}^k, \phi^k, t^k) = \omega_{j+\frac{1}{2}}^k u_{j+\frac{1}{2}}^{d,-} \cdot \phi_{j+\frac{1}{2}}^- - \omega_{j-\frac{1}{2}}^k u_{j-\frac{1}{2}}^{d,+} \cdot \phi_{j-\frac{1}{2}}^+ - \int_{K_j^k} \omega^k \mathbf{u}^{e,k} \cdot (\phi^k)_x dx. \tag{B.3}$$

By comparing (4.19) and (4.20), we have

$$\int_{K_j^{k+1}} (\mathbf{u}^{k+1} - \mathbf{u}^{e,k+1}) \cdot \phi^{k+1} dx = 0. \tag{B.4}$$

This means $\mathbf{u}^{k+1} = \mathbf{u}^{e,k+1}$. This finishes the proof that for $n \in \mathbb{N}$ the numerical solution \mathbf{u}^n is in the case of moving water equilibrium, which means our scheme do preserve the moving water equilibrium.

B.3 Proof of Theorem 4.6

Since we use the Lax-Friedrichs numerical flux, we can write the specific form of $\hat{g}_{j+\frac{1}{2}}^{l,[1]}$:

$$\begin{aligned} \hat{g}_{j+\frac{1}{2}}^{l,[1]} &= \hat{g}^{[1]}(\omega_{j+\frac{1}{2}}, \mathbf{u}_{j+\frac{1}{2}}^{*,-}, \mathbf{u}_{j+\frac{1}{2}}^{*,+}) \\ &= \frac{1}{2} \left(m_{j+\frac{1}{2}}^{*,+} - \omega_{j+\frac{1}{2}} h_{j+\frac{1}{2}}^{*,+} - \alpha h_{j+\frac{1}{2}}^{*,+} + m_{j+\frac{1}{2}}^{*,-} - \omega_{j+\frac{1}{2}} h_{j+\frac{1}{2}}^{*,-} + \alpha h_{j+\frac{1}{2}}^{*,-} \right) \end{aligned}$$

$$\begin{aligned}
 &= \frac{1}{2} \left(\left(u_{j+\frac{1}{2}}^{*,+} - \omega_{j+\frac{1}{2}} - \alpha \right) h_{j+\frac{1}{2}}^{*,+} + \left(u_{j+\frac{1}{2}}^{*,-} - \omega_{j+\frac{1}{2}} + \alpha \right) h_{j+\frac{1}{2}}^{*,-} \right) \\
 &< \frac{1}{2} \left((\alpha - \alpha) h_{j+\frac{1}{2}}^{*,+} + (\alpha + \alpha) h_{j+\frac{1}{2}}^{*,-} \right) = \alpha h_{j+\frac{1}{2}}^{*,-}.
 \end{aligned}
 \tag{B.5}$$

The first equality is due to $m^d = m^e$ is constant, which is same as the proof of mass conservation and the inequality is due the definition of α in (4.17). Similar we simplify $\hat{g}_{j-\frac{1}{2}}^{r,[1]}$ to get

$$\hat{g}_{j-\frac{1}{2}}^{r,[1]} > -\alpha h_{j-\frac{1}{2}}^{*,+}.
 \tag{B.6}$$

Now we combine (B.5), (B.6) and Gauss-Lobatto quadrature rule, which introduced in (3.6) to simplify (4.22):

$$\begin{aligned}
 \frac{\Delta_j^{n+1}}{\Delta_j^n} \bar{h}_j^{n+1} &= \bar{h}_j^n - \lambda \left(\hat{g}_{j+\frac{1}{2}}^{l,[1]} - \hat{g}_{j-\frac{1}{2}}^{r,[1]} \right) > \sum_{v=1}^L \hat{\omega}_v h_j^n(\hat{x}_j^{(v)}) - \alpha \lambda \left(h_{j+\frac{1}{2}}^{*,-} + h_{j-\frac{1}{2}}^{*,+} \right) \\
 &= \sum_{v=2}^{L-1} \hat{\omega}_v h_j^n(\hat{x}_j^{(v)}) + \left(\hat{\omega}_1 h_{j-\frac{1}{2}}^{n,+} - \alpha \lambda h_{j-\frac{1}{2}}^{*,+} \right) + \left(\hat{\omega}_L h_{j+\frac{1}{2}}^{n,-} - \alpha \lambda h_{j+\frac{1}{2}}^{*,-} \right) > 0.
 \end{aligned}$$

This is because the assumption of the theorem that $h_j^n(\hat{x}_j^{(v)}) > 0$ and (4.25):

$$1 > \hat{\alpha}_0 \lambda = \frac{\alpha \lambda}{\hat{\omega}_1} \max_{x \in \{x_{j-\frac{1}{2}}^+, x_{j+\frac{1}{2}}^-\}} \frac{h_j^*(x)}{h_j^n(x)} \rightarrow \hat{\omega}_L h_j^n(x) > \alpha \lambda h_j^*(x), \text{ for } x \in \{x_{j-\frac{1}{2}}^+, x_{j+\frac{1}{2}}^-\},
 \tag{B.7}$$

and the assumption of Gauss-Lobatto quadrature rule: $\hat{x}_1^{(v)}$ and $\hat{x}_L^{(v)}$ are the cell boundaries of K_j^n and $\hat{\omega}_1 = \hat{\omega}_L$. Now we have proved $\bar{h}_j^{n+1} > 0$ and finish the proof.

References

1. Arpaia, L., Ricchiuto, M.: R-adaptation for Shallow Water flows: conservation, well balancedness, efficiency. *Comput. Fluids* **160**, 175–203 (2018)
2. Arpaia, L., Ricchiuto, M.: Well balanced residual distribution for the ALE spherical shallow water equations on moving adaptive meshes. *J. Comput. Phys.* **405**, 109173 (2020)
3. Audusse, E., Bouchut, F., Bristeau, M.-O., Klein, R., Perthame, B.: A fast and stable well-balanced scheme with hydrostatic reconstruction for shallow water flows. *SIAM J. Sci. Comput.* **25**, 2050–2065 (2004)
4. Berthon, C., Marche, F.: A positive preserving high order VFRoe scheme for shallow water equations: a class of relaxation schemes. *SIAM J. Sci. Comput.* **30**, 2587–2612 (2008)
5. Bokhove, O.: Flooding and drying in discontinuous Galerkin finite-element discretizations of shallow-water equations. Part: 1 one dimension. *J. Sci. Comput.* **22.1–3**, 47–82 (2005)
6. Bollermann, A., Noelle, S., Lukáčová-Medvid'ová, M.: Finite volume evolution Galerkin methods for the shallow water equations with dry beds. *Commun. Comput. Phys.* **10**(2), 371–404 (2011)
7. Bollermann, A., Chen, G., Kurganov, A.: A well-balanced reconstruction of wet/dry fronts for the shallow water equations. *J. Sci. Comput.* **56**(2), 267–290 (2013)
8. Bouchut, F., Morales, T.: A subsonic-well-balanced reconstruction scheme for shallow water flows. *SIAM J. Numer. Anal.* **48**, 1733–1758 (2010)
9. Brufau, P., Vázquez-Cendón, M.E., García-Navarro, P.: A numerical model for the flooding and drying of irregular domains. *Int. J. Numer. Methods Fluids* **39**, 247–275 (2002)
10. Castro, M.J., Pardo Milanés, A., Parés, C.: Well-balanced numerical schemes based on a generalized hydrostatic reconstruction technique. *Math. Models Methods Appl. Sci.* **17**(12), 2055–2113 (2007)

11. Chen, G., Noelle, S.: A new hydrostatic reconstruction scheme based on subcell reconstructions. *SIAM J. Numer. Anal.* **55**(2), 758–784 (2017)
12. Cockburn, B., Hou, S., Shu, C.-W.: The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: the multidimensional case. *Math. Comput.* **54**, 545–581 (1990)
13. Cockburn, B., Lin, S.-Y., Shu, C.-W.: TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one-dimensional systems. *J. Comput. Phys.* **84**, 90–113 (1989)
14. Cockburn, B., Shu, C.-W.: TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework. *Math. Comput.* **52**, 411–435 (1989)
15. Cockburn, B., Shu, C.-W.: The Runge-Kutta local projection P1-discontinuous Galerkin finite element method for scalar conservation laws. *ESAIM Math. Modell. Numer. Anal.* **25**, 337–361 (1991)
16. Cockburn, B., Shu, C.-W.: The Runge-Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems. *J. Comput. Phys.* **141**, 199–224 (1998)
17. Duran, A., Marche, F.: Recent advances on the discontinuous Galerkin method for shallow water equations with topography source terms. *Compute. Fluids* **101**, 88–104 (2014)
18. Ern, A., Piperno, S., Djamel, K.: A well-balanced Runge-Kutta discontinuous Galerkin method for the shallow-water equations with flooding and drying. *Int. J. Numer. Methods Fluids* **58**, 1–25 (2008)
19. Fu, P., Schnücke, G., Xia, Y.: Arbitrary Lagrangian-Eulerian discontinuous Galerkin method for conservation laws on moving simplex meshes. *Math. Comput.* **88**, 2221–2255 (2019)
20. Gottlieb, S., Shu, C.-W., Tadmor, E.: Strong stability-preserving high-order time discretization methods. *SIAM Rev.* **43**, 89–112 (2001)
21. Jin, S., Wen, X.: Two interface type numerical methods for computing hyperbolic systems with geometrical source terms having concentrations. *SIAM J. Sci. Comput.* **26**, 2079–2101 (2005)
22. Kesserwani, G., Liang, Q.: Well-balanced RKDG2 solutions to the shallow water equations over irregular domains with wetting and drying. *Comput. Fluids* **39**, 2040–2050 (2010)
23. Klingenberg, C., Schnücke, G., Xia, Y.: Arbitrary Lagrangian-Eulerian discontinuous Galerkin method for conservation laws: analysis and application in one dimension. *Math. Comput.* **86**, 1203–1232 (2017)
24. Klingenberg, C., Puppo, G., Semplice, M.: Arbitrary order finite volume well-balanced schemes for the Euler equations with gravity. *SIAM J. Sci. Comput.* **41**(2), A695–A721 (2019)
25. LeVeque, R.J.: Balancing source terms and flux gradients on high-resolution Godunov methods: the quasi-steady wave-propagation algorithm. *J. Comput. Phys.* **146**, 346–365 (1998)
26. Nielsen, C., Apelt, C.: Parameters affecting the performance of wetting and drying in a two-dimensional finite element long wave hydrodynamic model. *J. Hydraul. Eng.* **129**, 628–636 (2003)
27. Noelle, S., Xing, Y., Shu, C.-W.: High-order well-balanced finite volume WENO schemes for shallow water equation with moving water. *J. Comput. Phys.* **226**, 29–58 (2007)
28. Rhebergen, S., Bokhove, O., Van der Vegt, J.J.W.: Discontinuous Galerkin finite element methods for hyperbolic nonconservative partial differential equations. *J. Comput. Phys.* **227**, 1887–1922 (2008)
29. Rhebergen, S.: Well-balanced R-adaptive and moving mesh space-time discontinuous Galerkin method for the shallow water equations. *Oxford Univ. Math. Inst. Tech. Rep.* **1757** (2013)
30. Russo, G.: A well-balanced flux-vector splitting scheme designed for hyperbolic systems of conservation laws with source terms. *Comput. Math. Appl.* **39**, 135–159 (2000)
31. Russo, G.: Central schemes for conservation laws with application to shallow water equations, pp. 225–246. *Milano, Trends and Applications of Mathematics to Mechanics*. Springer (2005)
32. Shu, C.-W., Osher, S.: Efficient implementation of essentially non-oscillatory shock-capturing schemes. *J. Comput. Phys.* **77**, 439–471 (1988)
33. Shu, C.-W.: High order WENO and DG methods for time-dependent convection-dominated PDEs: a brief survey of several recent developments. *J. Comput. Phys.* **316**, 598–613 (2016)
34. Vázquez-Cendón, M.E.: Improved treatment of source terms in upwind schemes for the shallow water equations in channels with irregular geometry. *J. Comput. Phys.* **148**, 497–526 (1999)
35. Xing, Y.: Exactly well-balanced discontinuous Galerkin methods for the shallow water equations with moving water equilibrium. *J. Comput. Phys.* **257**, 536–553 (2014)
36. Xing, Y., Shu, C.-W.: High order finite difference WENO schemes with the exact conservation property for the shallow water equations. *J. Comput. Phys.* **208**, 206–227 (2005)
37. Xing, Y., Shu, C.-W.: High order well-balanced finite volume WENO schemes and discontinuous Galerkin methods for a class of hyperbolic systems with source terms. *J. Comput. Phys.* **214**, 567–598 (2006)
38. Xing, Y., Shu, C.-W.: High-order well-balanced finite difference WENO schemes for a class of hyperbolic systems with source terms. *J. Sci. Comput.* **27**, 477–494 (2006)
39. Xing, Y., Shu, C.-W.: A new approach of high order well-balanced finite volume WENO schemes and discontinuous Galerkin methods for a class of hyperbolic systems with source terms. *Commun. Comput. Phys.* **214**, 567–598 (2006)

40. Xing, Y., Zhang, X., Shu, C.-W.: Positivity-preserving high order well-balanced discontinuous Galerkin methods for the shallow water equations. *Adv. Water Resour.* **33**, 1476–1493 (2010)
41. Xing, Y., Shu, C.-W., Noelle, S.: On the advantage of well-balanced schemes for moving-water equilibria of the shallow water equations. *J. Sci. Comput.* **48**, 339–349 (2011)
42. Zhang, M., Huang, W., Qiu, J.: A high-order well-balanced positivity-preserving moving mesh DG method for the shallow water equations with non-flat bottom topography. *J. Sci. Comput.* **87**(3), 1–43 (2021)
43. Zhang, X., Xia, Y., Shu, C.-W.: Maximum-principle-satisfying and positivity-preserving high order discontinuous Galerkin schemes for conservation laws on triangular meshes. *J. Sci. Comput.* **50**, 29–62 (2012)
44. Zhou, F., Chen, G., Noelle, S., Guo, H.: A well-balanced stable generalized Riemann problem scheme for shallow water equations using adaptive moving unstructured triangular meshes. *Int. J. Numer. Methods Fluids* **73**(3), 266–283 (2013)
45. Zhou, F., Chen, G., Huang, Y., Yang, J.Z., Feng, H.: An adaptive moving finite volume scheme for modeling flood inundation over dry and complex topography. *Water Resour. Res.* **49**(4), 1914–1928 (2013)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.