



Contents lists available at ScienceDirect

Advances in Water Resources

journal homepage: www.elsevier.com/locate/advwatres

Positivity-preserving high order well-balanced discontinuous Galerkin methods for the shallow water equations ^{☆,☆☆}

Yulong Xing ^{a,b,*}, Xiangxiong Zhang ^c, Chi-Wang Shu ^d

^a Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831, USA

^b Department of Mathematics, University of Tennessee, Knoxville, TN 37996, USA

^c Department of Mathematics, Brown University, Providence, RI 02912, USA

^d Division of Applied Mathematics, Brown University, Providence, RI 02912, USA

ARTICLE INFO

Article history:

Received 15 March 2010
 Received in revised form 12 August 2010
 Accepted 13 August 2010
 Available online xxx

Keywords:

Shallow water equations
 Discontinuous Galerkin method
 High order accuracy
 Well-balanced
 Positivity-preserving methods
 Wetting and drying treatment

ABSTRACT

Shallow water equations with a non-flat bottom topography have been widely used to model flows in rivers and coastal areas. An important difficulty arising in these simulations is the appearance of dry areas where no water is present, as standard numerical methods may fail in the presence of these areas. These equations also have still water steady state solutions in which the flux gradients are nonzero but exactly balanced by the source term. In this paper we propose a high order discontinuous Galerkin method which can maintain the still water steady state exactly, and at the same time can preserve the non-negativity of the water height without loss of mass conservation. A simple positivity-preserving limiter, valid under suitable CFL condition, will be introduced in one dimension and then extended to two dimensions with rectangular meshes. Numerical tests are performed to verify the positivity-preserving property, well-balanced property, high order accuracy, and good resolution for smooth and discontinuous solutions.

© 2010 Elsevier Ltd. All rights reserved.

1. Introduction

The shallow water equations with a non-flat bottom topography have been widely used to model flows in rivers and coastal areas. They have wide applications in ocean and hydraulic engineering: tidal flows in estuary and coastal water region; bore wave propagation; and river, reservoir, and open channel flows, among others. Many geophysical flows are modeled by the variants of the shallow water equations. This system describes the flow as a conservation law with additional source terms. In one space dimension, the shallow water equations take the form:

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

where h denotes the water height, u is the velocity of the fluid, b represents the bottom topography and g is the gravitational constant. Only the source term due to the bottom topography is taken into account in this system, but other terms could also be added in order to include effects such as friction on the bottom and on the surface as well as variations of the channel width.

Research on numerical methods for the solution of the shallow water system has attracted tremendous attention in the past two decades. A significant result in computing such solutions was given by Bermudez and Vazquez [2] in 1994. They proposed the idea of the “exact C-property”, which refers to the ability of the scheme to exactly preserve the still water at rest steady state solution:

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

which represents a still flat water surface. Such numerical methods are often regarded as well-balanced methods. Developing well-balanced methods for the shallow water equations is not a trivial task, especially for high order accurate methods. One key difficulty in developing high order well-balanced methods comes from the fact that we should not include any derivative of the unknown solution h and u in the approximation to the source term. Otherwise, conservation and convergence towards weak solutions will be problematic for discontinuous solutions. Recently, several high order well-balanced methods for the shallow water equations were successfully developed in [7–9,23,25,26,28,29,36–41].

* Research is sponsored by the Office of Advanced Scientific Computing Research, US Department of Energy. The work was performed at the ORNL, which is managed by UT-Battelle, LLC under Contract No. DE-AC05-00OR22725.

** Research supported by DOE Grant DE-FG02-08ER25863 and NSF Grant DMS-0809086.

* Corresponding author at: Department of Mathematics, University of Tennessee, Knoxville, TN 37996, USA. Tel.: +1 865 574 0247; fax: +1 865 574 0680.

E-mail addresses: xingy@math.utk.edu (Y. Xing), zhangxx@dam.brown.edu (X. Zhang), shu@dam.brown.edu (C.-W. Shu).

Another important difficulty often encountered in the simulations of the shallow water equations is the appearance of dry areas where no water is present. Many shallow water applications involve rapidly moving interfaces between wet and dry areas, such as dam breaks, flood waves and run-up phenomena over shores and sea defence structures. If no special attention is paid, standard numerical methods may fail near dry/wet front and may produce unacceptable negative water height.

There are many existing wetting and drying treatments for the continuous Galerkin based methods [24]. The first type is the mesh adaption technique which tracks the dry front by changing the meshes. It has the advantage in accuracy but is computationally expensive. The second type uses the mesh reduction technique, which removes the dry elements and restores them when they become wet later. It may cause oscillation and loss of mass and momentum (failure in conservation). Thin layer technique maintains a very thin layer in dry elements and includes these dry elements in the computation.

In recent years, high order accurate numerical schemes, including finite difference/volume WENO schemes, spectral methods and discontinuous Galerkin (DG) methods, have been developed to reduce the number of computational cells and minimize the computational time to achieve the desired resolution. Among these methods, DG method is a class of finite element methods using discontinuous piecewise polynomial space as the solution and test function spaces (see [12] for a historic review). It combines advantages of both finite element and finite volume methods, and has been successfully applied to a wide range of applications. Several advantages of the DG method, including its accuracy, high parallel efficiency, flexibility for hp-adaptivity and arbitrary geometry and meshes, make it particularly suited for the shallow water equations [15,17,20]. In the discontinuous Galerkin (DG) framework, mesh adaption technique was introduced in [4]. Ern et al. [16] employed a slope modification technique to keep the positivity of the water height. However, their method cannot preserve the mass conservation. A second order mass-conserving thin layer approach was presented in [6]. However, in this approach, a special treatment is needed in the flux computation to prevent instability due to excessive drying. Other methods involving wetting and drying treatments for the shallow water equations include [3,5,10,18,21,22].

The main objective of this paper is to develop positivity-preserving high order accurate well-balanced DG methods for the shallow water equations. Most existing wetting and drying treatments are focused on post-processing reconstruction of the data obtained from the numerical solution at each time level. Even though the post-processing can bring the reconstruction to satisfy non-negative water height, this alone usually does *not* guarantee that the solution (e.g., cell average from a finite volume or DG scheme) at the next time step still maintains the non-negative water height property. If negative cell averages for the water height are obtained at the next time level, the positivity reconstruction post-processing will destroy conservation. Thanks to the fact that the equation for the positivity variable h does not have a source term, following the approaches proposed in [30,42,43], we introduce a simple positivity-preserving limiter operator, which preserves the high order accuracy without losing local mass and momentum conservation. A rigorous proof of the non-negativity of the water height for the next time step, under a suitable CFL condition, provided the water height at the current time step is non-negative, will be given. The algorithm is first introduced in one dimension and then extended to two dimensions with rectangular meshes.

This paper is organized as follows. In Section 2, we give a brief review of the well-balanced DG methods for the shallow water equations proposed in [40]. The positivity-preserving limiter, which keeps the water height non-negative, preserves the mass

conservation and at the same time does not affect the high order accuracy for the general solutions is presented in Section 3. In Section 4, we combine the well-balanced technique and positivity-preserving limiter together, which involves a necessary change in the slope limiter procedure. Extension to two dimensions with rectangular meshes is introduced in Section 5. Section 6 contains extensive numerical simulation results to demonstrate the behavior of our DG methods for one- and two-dimensional shallow water equations, verifying high order accuracy, the well-balanced property, positivity-preserving property, and good resolution for smooth and discontinuous solutions. Concluding remarks are given in Section 7.

2. Well-balanced DG methods

Several well-balanced DG methods for the shallow water equations have been developed, see for example [27] for a list of references. In this paper, we consider the approach developed by two of the authors in [40], where we observed that the traditional DG methods are capable of maintaining the still water solution (1.2) exactly, if a small modification on the flux is provided. This is one of the simplest approaches to obtain a high order well-balanced scheme, and the computational cost to obtain such a well-balanced DG method is basically the same as the traditional DG method. In this section, we briefly review this well-balanced approach in one dimension, and refer to [40] for further details.

We discretize the computational domain into cells $I_j = [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$, and denote the size of the j -th cell by Δx_j and the maximum mesh size by $\Delta x = \max_j \Delta x_j$. For the ease of presentation, we denote the shallow water Eq. (1.1) by

$$U_t + f(U)_x = s(h, b),$$

where $U = (h, hu)^T$ with the superscript T denoting the transpose, $f(U)$ is the flux and $s(h, b)$ is the source term. In a high order DG method, we seek an approximation, still denoted by U with an abuse of notation, which belongs to the finite dimensional space:

$$V_{\Delta x} = V_{\Delta x}^k \equiv \{w : w|_{I_j} \in P^k(I_j), \quad j = 1, \dots, J\}, \quad (2.1)$$

where $P^k(I)$ denotes the space of polynomials in I of degree at most k and J is the total number of computational cells. We project the bottom function b into the same space $V_{\Delta x}$, to obtain an approximation which is still denoted by b , again with an abuse of notation. The numerical scheme is given by

$$\int_{I_j} \partial_t U v dx - \int_{I_j} f(U) \partial_x v dx + \hat{f}_{j+\frac{1}{2}} v(x_{j+\frac{1}{2}}^-) - \hat{f}_{j-\frac{1}{2}} v(x_{j-\frac{1}{2}}^+) = \int_{I_j} s(h, b) v dx, \quad (2.2)$$

where $v(x)$ is a test function from the test space $V_{\Delta x}$:

$$\hat{f}_{j+\frac{1}{2}} = F\left(U(x_{j+\frac{1}{2}}^-, t), U(x_{j+\frac{1}{2}}^+, t)\right), \quad (2.3)$$

and $F(a_1, a_2)$ is a numerical flux. We could, for example, use the simple Lax–Friedrichs flux:

$$F(a_1, a_2) = \frac{1}{2}(f(a_1) + f(a_2) - \alpha(a_2 - a_1)), \quad (2.4)$$

where $\alpha = \max(|u| + \sqrt{gh})$ and the maximum is taken over the whole region. A simple Euler forward time discretization of (2.2) gives the fully discretized scheme:

$$\int_{I_j} \frac{U^{n+1} - U^n}{\Delta t} v dx - \int_{I_j} f(U^n) \partial_x v dx + \hat{f}_{j+\frac{1}{2}}^n v(x_{j+\frac{1}{2}}^-) - \hat{f}_{j-\frac{1}{2}}^n v(x_{j-\frac{1}{2}}^+) = \int_{I_j} s(h^n, b) v dx. \quad (2.5)$$

Total variation diminishing (TVD) high order Runge–Kutta time discretization [35] is used in practice for stability and to increase temporal accuracy. For example, the third order TVD Runge–Kutta method is used in the simulation in this paper:

$$\begin{aligned} U^{(1)} &= U^n + \Delta t \mathcal{F}(U^n), \\ U^{(2)} &= \frac{3}{4} U^n + \frac{1}{4} (U^{(1)} + \Delta t \mathcal{F}(U^{(1)})), \\ U^{n+1} &= \frac{1}{3} U^n + \frac{2}{3} (U^{(2)} + \Delta t \mathcal{F}(U^{(2)})), \end{aligned} \quad (2.6)$$

where $\mathcal{F}(U)$ is the spatial operator.

In order to achieve the well-balanced property, we are interested in preserving the still water stationary solution (1.2) exactly. As mentioned in [40], our well-balanced numerical scheme, with a simple Euler forward time discretization, has the form:

$$\int_{I_j} \frac{U^{n+1} - U^n}{\Delta t} v dx - \int_{I_j} f(U^n) \partial_x v dx + \hat{f}_{j+\frac{1}{2}}^l v(x_{j+\frac{1}{2}}^-) - \hat{f}_{j-\frac{1}{2}}^r v(x_{j-\frac{1}{2}}^+) = \int_{I_j} s(h^n, b) v dx, \quad (2.7)$$

or equivalently:

$$\int_{I_j} \frac{U^{n+1} - U^n}{\Delta t} v dx - \int_{I_j} f(U^n) \partial_x v dx + \hat{f}_{j+\frac{1}{2}}^l v(x_{j+\frac{1}{2}}^-) - \hat{f}_{j-\frac{1}{2}}^r v(x_{j-\frac{1}{2}}^+) = \int_{I_j} s(h^n, b) v dx + (\hat{f}_{j+\frac{1}{2}}^l - \hat{f}_{j+\frac{1}{2}}^r) v(x_{j+\frac{1}{2}}^-) - (\hat{f}_{j-\frac{1}{2}}^l - \hat{f}_{j-\frac{1}{2}}^r) v(x_{j-\frac{1}{2}}^+). \quad (2.8)$$

The left side of (2.8) is the traditional RKDG scheme, and the right side is our approximation to the source term. The design of the left flux $\hat{f}_{j+\frac{1}{2}}^l$ and the right flux $\hat{f}_{j-\frac{1}{2}}^r$ will be explained later, however we point out here that $\hat{f}_{j+\frac{1}{2}}^l - \hat{f}_{j+\frac{1}{2}}^r$ and $\hat{f}_{j-\frac{1}{2}}^l - \hat{f}_{j-\frac{1}{2}}^r$ are high order correction terms at the level of $O(\Delta x^{k+1})$ regardless of the smoothness of the solution U . Therefore, the scheme (2.7) is a spatially $(k + 1)$ -th order conservative scheme and will converge to the weak solution.

After computing boundary values $U_{j+\frac{1}{2}}^{\pm}$, we set:

$$h_{j+\frac{1}{2}}^{*,\pm} = \max(0, h_{j+\frac{1}{2}}^{\pm} + b_{j+\frac{1}{2}}^{\pm} - \max(b_{j+\frac{1}{2}}^+, b_{j+\frac{1}{2}}^-)), \quad (2.9)$$

and redefine the left and right values of U as

$$U_{j+\frac{1}{2}}^{*,\pm} = \begin{pmatrix} h_{j+\frac{1}{2}}^{*,\pm} \\ h_{j+\frac{1}{2}}^{*,\pm} u_{j+\frac{1}{2}}^{\pm} \end{pmatrix}. \quad (2.10)$$

Then the left and right fluxes $\hat{f}_{j+\frac{1}{2}}^l$ and $\hat{f}_{j-\frac{1}{2}}^r$ are given by

$$\begin{aligned} \hat{f}_{j+\frac{1}{2}}^l &= F(U_{j+\frac{1}{2}}^{*,+}, U_{j+\frac{1}{2}}^{*,+}) + \begin{pmatrix} 0 \\ \frac{\alpha}{2} (h_{j+\frac{1}{2}}^-)^2 - \frac{\alpha}{2} (h_{j+\frac{1}{2}}^+)^2 \end{pmatrix}, \\ \hat{f}_{j-\frac{1}{2}}^r &= F(U_{j-\frac{1}{2}}^{*,+}, U_{j-\frac{1}{2}}^{*,+}) + \begin{pmatrix} 0 \\ \frac{\alpha}{2} (h_{j-\frac{1}{2}}^+)^2 - \frac{\alpha}{2} (h_{j-\frac{1}{2}}^-)^2 \end{pmatrix}. \end{aligned} \quad (2.11)$$

We also require that all the integrals in formula (2.7) should be calculated exactly at the still water state. This can be easily achieved by using suitable Gauss quadrature rules since the numerical solutions h , b and v are polynomials at the still water state in each cell I_j , hence $f(U)$ and $s(h, b)$ are both polynomials. We have proven in [40] that the above methods (2.7), combined with the choice of fluxes (2.11), are actually well-balanced for the still water steady state of the shallow water equations.

Another important ingredient for the DG methods is that a slope limiter procedure might be needed after each inner stage in the Runge–Kutta time stepping, when the solution contains discontinuities. We use the characteristic-wise total variation bounded (TVB) limiter in [14,33], with a corrected minmod function defined by

$$\bar{m}(a_1, \dots, a_n) = \begin{cases} a_1, & \text{if } |a_1| \leq M \Delta x^2, \\ m(a_1, \dots, a_n), & \text{otherwise,} \end{cases} \quad (2.12)$$

where M is the TVB parameter to be chosen adequately [13] and the minmod function m is given by

$$m(a_1, \dots, a_n) = \begin{cases} s \min_i |a_i|, & \text{if } s = \text{sign}(a_1) = \dots = \text{sign}(a_n), \\ 0, & \text{otherwise.} \end{cases}$$

For the shallow water system, we perform the limiting in the local characteristic variables. However, this limiter procedure might destroy the preservation of the still water steady state $h + b = \text{const}$. Therefore, following the idea presented in [1,45], we apply the limiter procedure on the function $(h + b, (hu))^T$ instead. The modified RKDG solution is then defined by $h^{mod} \equiv (h + b)^{mod} - b$. Since the average of h in cell I_j , denoted by \bar{h}_j , satisfies that $\bar{h}_j^{mod} = (\bar{h} + \bar{b})^{mod} - \bar{b}_j = (\bar{h} + \bar{b}) - \bar{b}_j = \bar{h}_j$, we observe that this procedure will not destroy the conservativity of h , which should be maintained during the limiter process.

3. Positivity-preserving limiter

We consider the Euler forward in time (2.7) first, and higher order time discretization will be discussed later in this section. By taking the test function $v = 1$ in (2.7), we obtain the scheme satisfied by the cell averages in the well-balanced DG methods:

$$\bar{U}_j^{n+1} = \bar{U}_j^n - \lambda (\hat{f}_{j+\frac{1}{2}}^r - \hat{f}_{j-\frac{1}{2}}^l) + \Delta t \int_{I_j} s(h^n, b) dx. \quad (3.1)$$

By plugging (2.10) and (2.11) into (3.1), the scheme satisfied by the cell averages of the height in the well-balanced DG methods (2.7) can be written as

$$\bar{h}_j^{n+1} = \bar{h}_j^n - \lambda [\hat{F}(h_{j+\frac{1}{2}}^{*,+}, u_{j+\frac{1}{2}}^-; h_{j+\frac{1}{2}}^{*,+}, u_{j+\frac{1}{2}}^+) - \hat{F}(h_{j-\frac{1}{2}}^{*,+}, u_{j-\frac{1}{2}}^-; h_{j-\frac{1}{2}}^{*,+}, u_{j-\frac{1}{2}}^+)], \quad (3.2)$$

where

$$\hat{F}(h_{j+\frac{1}{2}}^{*,+}, u_{j+\frac{1}{2}}^-; h_{j+\frac{1}{2}}^{*,+}, u_{j+\frac{1}{2}}^+) = \frac{1}{2} (h_{j+\frac{1}{2}}^{*,+} u_{j+\frac{1}{2}}^- + h_{j+\frac{1}{2}}^{*,+} u_{j+\frac{1}{2}}^+ - \alpha (h_{j+\frac{1}{2}}^{*,+} - h_{j+\frac{1}{2}}^{*,+})) \quad (3.3)$$

and $h_{j+\frac{1}{2}}^{*,\pm}$ are defined in (2.9).

We start by showing the positivity of a first order scheme with the well-balanced flux.

Lemma 3.1. Under the CFL condition $\lambda \alpha \leq 1$, with $\alpha = \max(|u| + \sqrt{gh})$, consider the following scheme:

$$h_j^{n+1} = h_j^n - \lambda [\hat{F}(h_j^{*,+}, u_j^n; h_{j+1}^{*,+}, u_{j+1}^+) - \hat{F}(h_{j-1}^{*,+}, u_{j-1}^-; h_j^{*,+}, u_j^n)], \quad (3.4)$$

with \hat{F} the same as in (3.3) and

$$h_j^{*,+} = \max(0, h_j^n + b_j - \max(b_j, b_{j+1}))$$

$$h_j^{*,+} = \max(0, h_j^n + b_j - \max(b_{j-1}, b_j)).$$

If $h_j^n, h_{j\pm 1}^n$ are non-negative, then h_j^{n+1} is also non-negative.

Proof. The scheme (3.4) can be written as

$$h_j^{n+1} = \left[1 - \frac{1}{2}\lambda(\alpha + u_j^n) \frac{h_j^{*+}}{h_j^n} - \frac{1}{2}\lambda(\alpha - u_j^n) \frac{h_j^{*-}}{h_j^n} \right] h_j^n + \left[\frac{1}{2}\lambda(\alpha + u_{j-1}^n) \frac{h_{j-1}^{*+}}{h_{j-1}^n} \right] h_{j-1}^n + \left[\frac{1}{2}\lambda(\alpha - u_{j+1}^n) \frac{h_{j+1}^{*-}}{h_{j+1}^n} \right] h_{j+1}^n.$$

Therefore, h_j^{n+1} is a linear combination of h_{j-1}^n , h_j^n and h_{j+1}^n and all the coefficients are non-negative since $0 \leq h_j^{*\pm} \leq h_j^n$. Thus, $h_j^{n+1} \geq 0$. \square

Next, we consider high order schemes. Let us first introduce the N -point Legendre Gauss–Lobatto quadrature rule on the interval $I_j = [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$, which is exact for the integral of polynomials of degree up to $2N - 3$, where we choose N such that $2N - 3 \geq k$. We denote these quadrature points on I_j as

$$S_j = \left\{ x_{j-\frac{1}{2}} = \hat{x}_j^1, \hat{x}_j^2, \dots, \hat{x}_j^{N-1}, \hat{x}_j^N = x_{j+\frac{1}{2}} \right\}.$$

Let \hat{w}_r be the quadrature weights for the interval $[-1/2, 1/2]$ such that $\sum_{r=1}^N \hat{w}_r = 1$. Recall $h_j^n(x)$ denotes the DG polynomial approximating the water height in the cell I_j . We have:

$$\begin{aligned} \bar{h}_j^n &= \frac{1}{\Delta x} \int_{I_j} h_j^n(x) dx = \sum_{r=1}^N \hat{w}_r h_j^n(\hat{x}_j^r) \\ &= \sum_{r=2}^{N-1} \hat{w}_r h_j^n(\hat{x}_j^r) + \hat{w}_1 h_{j-\frac{1}{2}}^+ + \hat{w}_N h_{j+\frac{1}{2}}^-, \end{aligned} \quad (3.5)$$

since the quadrature is exact for polynomials of degree k . Following the approaches in [30,42,43], we have the result:

Proposition 3.2. Consider the scheme (3.2) satisfied by the cell averages of the water height in our DG method. Let $h_j^n(x)$ be the DG polynomial for the water height in the cell I_j . If $h_{j-\frac{1}{2}}^+, h_{j+\frac{1}{2}}^+$ and $h_j^n(\hat{x}_j^r)$ ($r = 1, \dots, N$) are all non-negative, then \bar{h}_j^{n+1} is also non-negative under the CFL condition:

$$\lambda\alpha \leq \hat{w}_1. \quad (3.6)$$

Proof. Plug (3.5) into (3.2). We can rewrite (3.2) by adding and subtracting the term $\hat{F}(h_{j-\frac{1}{2}}^{*+}, u_{j-\frac{1}{2}}^+; h_{j+\frac{1}{2}}^{*-}, u_{j+\frac{1}{2}}^-)$:

$$\begin{aligned} \bar{h}_j^{n+1} &= \sum_{r=2}^{N-1} \hat{w}_r h_j^n(\hat{x}_j^r) + \hat{w}_1 h_{j-\frac{1}{2}}^+ + \hat{w}_N h_{j+\frac{1}{2}}^- - \lambda \left[\hat{F}(h_{j+\frac{1}{2}}^{*-}, u_{j+\frac{1}{2}}^-; h_{j+\frac{1}{2}}^{*+}, u_{j+\frac{1}{2}}^+) \right. \\ &\quad - \hat{F}(h_{j-\frac{1}{2}}^{*+}, u_{j-\frac{1}{2}}^+; h_{j+\frac{1}{2}}^{*-}, u_{j+\frac{1}{2}}^-) + \hat{F}(h_{j-\frac{1}{2}}^{*+}, u_{j-\frac{1}{2}}^+; h_{j+\frac{1}{2}}^{*-}, u_{j+\frac{1}{2}}^-) \\ &\quad \left. - \hat{F}(h_{j-\frac{1}{2}}^{*-}, u_{j-\frac{1}{2}}^-; h_{j-\frac{1}{2}}^{*+}, u_{j-\frac{1}{2}}^+) \right] \\ &= \sum_{r=2}^{N-1} \hat{w}_r h_j^n(\hat{x}_j^r) + \hat{w}_N H_N + \hat{w}_1 H_1 \end{aligned}$$

where

$$H_1 = h_{j-\frac{1}{2}}^+ - \frac{\lambda}{\hat{w}_1} \left[\hat{F}(h_{j-\frac{1}{2}}^{*+}, u_{j-\frac{1}{2}}^+; h_{j+\frac{1}{2}}^{*-}, u_{j+\frac{1}{2}}^-) - \hat{F}(h_{j-\frac{1}{2}}^{*-}, u_{j-\frac{1}{2}}^-; h_{j+\frac{1}{2}}^{*+}, u_{j+\frac{1}{2}}^+) \right] \quad (3.7)$$

$$H_N = h_{j+\frac{1}{2}}^- - \frac{\lambda}{\hat{w}_N} \left[\hat{F}(h_{j+\frac{1}{2}}^{*-}, u_{j+\frac{1}{2}}^-; h_{j+\frac{1}{2}}^{*+}, u_{j+\frac{1}{2}}^+) - \hat{F}(h_{j-\frac{1}{2}}^{*+}, u_{j-\frac{1}{2}}^+; h_{j+\frac{1}{2}}^{*-}, u_{j+\frac{1}{2}}^-) \right]. \quad (3.8)$$

Notice that (3.7) and (3.8) are both of the type (3.4), hence $H_1 \geq 0$ and $H_N \geq 0$ under the suitable CFL conditions, which are $\frac{\lambda}{\hat{w}_1} \alpha \leq 1$ and $\frac{\lambda}{\hat{w}_N} \alpha \leq 1$, respectively. Since $\hat{w}_1 = \hat{w}_N$, these two CFL conditions

are the same and become (3.6). Therefore $\bar{h}_j^{n+1} \geq 0$, since it is a convex combination of H_1 , H_N and $h_j^n(\hat{x}_j^r)$ ($r = 2, \dots, N - 1$). \square

Remark 3.3. Here we only discuss the Euler forward time discretization. TVD high order Runge–Kutta [35] and multistep [34] time discretizations will keep the validity of the proposition since TVD time discretizations are convex combinations of the Euler forward operators.

Remark 3.4. For $k = 2, 3$, the \hat{w}_1 of the Gauss–Lobatto quadrature is $1/6$, and the corresponding CFL condition in (3.6) is $\lambda\alpha \leq 1/6$. Recall that the CFL condition for linear stability for the DG methods is $\lambda\alpha \leq 1/5$ for $k = 2$, which are comparable to our CFL restriction. The CFL condition (3.6) using the flux (2.11) and (2.4), and the Gauss–Lobatto quadrature points for $k = 2, 3, 4, 5$ are listed in Table 3.1.

Remark 3.5. The Gauss–Lobatto quadrature only serves the purpose towards the proof of the Proposition 3.2. We only need these Gauss–Lobatto points when evaluating (3.10) later in the implementation of the positivity-preserving limiter. It has nothing to do with the computation of the cell integrals in (2.2), for which we can use any quadrature as long as the accuracy requirement is satisfied.

Remark 3.6. Note that although the well-balanced flux (2.11) is used throughout the proof, the result also holds for the traditional DG methods using the flux (2.3) without the correction (2.11). Any other positivity-preserving exact or approximate Riemann solver, including Godunov, Boltzmann type and Harten–Lax–Van Leer, will also work under the corresponding CFL condition. Also, although the equation for the positivity variable h does not have a source term, we would like to comment that combining the well-balanced scheme with the positivity-preserving limiter is still non-trivial. For example, the well-balanced discontinuous Galerkin method developed in [39] cannot be extended in the same fashion.

To enforce the conditions of this proposition, we need to modify $h_j^n(x)$ such that it is non-negative for all $x \in S_j$. At time level n , given $\bar{h}_j^n \geq 0$, we introduce the following limiter on the DG polynomial $U_j^n(x) = (h_j^n(x), (hu)_j^n(x))$, which is a linear scaling around its cell average:

$$\tilde{U}_j^n(x) = \theta(U_j^n(x) - \bar{U}_j^n) + \bar{U}_j^n, \quad \theta = \min \left\{ 1, \frac{\bar{h}_j^n}{h_j^n - m_j} \right\}, \quad (3.9)$$

with

$$m_j = \min_{x \in S_j} h_j^n(x) = \min_{r=1, \dots, N} h_j^n(\hat{x}_j^r). \quad (3.10)$$

It is easy to observe that $\tilde{h}_j^n(\hat{x}_j^r) \geq 0$ ($r = 1, \dots, N$). We compute the modified polynomial $\tilde{U}_j^n(x)$ and use $\tilde{U}_j^n(x)$ instead of $U_j^n(x)$ in the scheme (2.7). Hence by this proposition, \bar{h}_j^{n+1} at time level $n + 1$ is

Table 3.1 The CFL condition (3.6) using the flux (2.11) and (2.4) for $2 \leq k \leq 5$ and the Gauss–Lobatto quadrature points on $[-\frac{1}{2}, \frac{1}{2}]$.

k	CFL	Quadrature points on $[-\frac{1}{2}, \frac{1}{2}]$
2	$\lambda\alpha \leq \frac{1}{6}$	$\{-\frac{1}{2}, 0, \frac{1}{2}\}$
3	$\lambda\alpha \leq \frac{1}{6}$	$\{-\frac{1}{2}, 0, \frac{1}{2}\}$
4	$\lambda\alpha \leq \frac{1}{12}$	$\{-\frac{1}{2}, -\frac{1}{\sqrt{20}}, \frac{1}{\sqrt{20}}, \frac{1}{2}\}$
5	$\lambda\alpha \leq \frac{1}{12}$	$\{-\frac{1}{2}, -\frac{1}{\sqrt{20}}, \frac{1}{\sqrt{20}}, \frac{1}{2}\}$

also non-negative, therefore (3.9) is indeed a positivity-preserving limiter for the well-balanced DG methods.

Note that this positivity-preserving limiter preserves the local conservation of water height h and also the momentum hu (i.e. $\bar{U}_j^n(x) = \bar{U}_j^n$). It can also be shown that this limiter does not destroy the high order accuracy, we refer to [42] for the detailed proof. We would like to mention that in wet region, where m_j is $O(1)$ above zero, the limiter does not take any effect, i.e. $\bar{U}_j^n(x) = U_j^n(x)$. Therefore this positivity-preserving limiter is active only in the dry or nearly dry region.

In this section, we proved a sufficient condition for the cell average \bar{U}_j^{n+1} in the well-balanced DG methods (2.7) and (2.11) to have non-negative water height. A very simple limiter can enforce this sufficient condition without destroying the high order accuracy and conservativity of the water height. Moreover, the positivity-preserving limiter will not take any effect if the DG polynomials $U_j^n(x)$ satisfy (1.2), since $h_j^n(x) = \text{const} - b_j(x) \geq 0$ for any x . Thus the limiter will not destroy the well-balanced property. Implementation details of the limiter will be shown in the next section.

4. Positivity-preserving high order well-balanced DG methods

In practical implementation, numerical tests show that there may be a conflict between the well-balanced TVB limiter (on $(h + b, (hu))^T$) and the positivity-preserving limiter if care is not taken in implementing these two limiters simultaneously. We may observe that the numerical time step becomes smaller and smaller as time evolves, and eventually the code stops. The same phenomenon has been reported in [6,16]. However, we observe that the traditional TVB limiter (on $(h, (hu))^T$), which is not well-balanced, works well with the positivity-preserving limiter.

We have tested various possible combinations of different limiters, and have come up with one which works well in the numerical tests, which we describe in detail below. Note that the TVB limiter procedure actually involves two steps: the first one is to check whether any limiting is needed in a specific cell; and, if the answer is yes, the second step is to apply the TVB limiter on the variables in this cell. In order to achieve both well-balanced and positivity-preserving property, we propose the following way to perform the TVB limiter. We first check if the limiting is needed, based on $(h + b, (hu))^T$ if in the wet region (i.e. θ in (3.9) equals to 1), or based on $(h, (hu))^T$ if in the dry or nearly dry region (i.e. $\theta < 1$). This is reasonable since $h + b$ is no longer constant in such dry or nearly dry region. If a certain cell is flagged by this procedure needing limiting, then the actual TVB limiter is implemented on $(h, (hu))^T$. Note that if in a steady state region where $h + b = \text{const}$ and $u = 0$ which is wet, we first check if the limiting is needed based on $(h + b, (hu))^T = (\text{const}, 0)^T$, which demonstrates that limiting is not needed in this cell. Therefore the flat surface $h + b = \text{const}$ will not be affected by the limiter procedure and the well-balanced property is maintained. When the limiting procedure is implemented this way, numerical results show that this choice of the TVB limiter does not destroy the well-balanced property, and also works well with the positivity-preserving limiter.

Again, we would like to mention that the change on the TVB limiter is purely for the purpose of the well-balanced property. If we solve for problems which are far from steady state, there is no need for using this new limiter procedure and we can switch back to the traditional limiter on $(h, (hu))^T$.

Another problem may occur in practical implementation when the water height is close to zero. In these nearly dry regions, the velocity $u = (hu)/h$ is not computed accurately and can achieve very large values even with a small numerical error in hu . This in turn leads to very small time steps with the CFL condition. Since the velocity in these nearly dry regions should be at the same

magnitude as the maximum of the velocity in wet regions, in numerical tests, we set $u = 0$ if $h \leq 10^{-6}$. The same treatment has been used in [31].

Given the DG polynomial $U_j^n(x)$ in interval I_j at time level n with a non-negative height cell average $\bar{h}_j^n \geq 0$, the algorithm flowchart of our high order well-balanced positivity-preserving DG method with Euler forward in time for the shallow water equations is

- Evaluate m_j by (3.10).
- In each cell I_j , check if the TVB limiter is needed based on $(h + b, (hu))^T$ if $m_j \geq 0$, or based on $(h, (hu))^T$ otherwise. If the answer is yes, perform the TVB limiter on $(h, (hu))^T$. The DG polynomial after the TVB limiting is still denoted as U_j^n .
- Evaluate m_j by (3.10) again and use the positivity-preserving limiter (3.9) to compute $\bar{U}_j^n(x)$.
- Use $\bar{U}_j^n(x)$ instead of $U_j^n(x)$ in the DG scheme (2.7) with the CFL condition (3.6).

For TVD high order time discretizations, we need to perform the algorithm above in each stage for a Runge–Kutta method or in each step for a multistep method.

5. Two-dimensional extension

In this section, we construct the positivity-preserving well-balanced DG scheme on rectangular meshes to solve the two-dimensional shallow water equations, which take the form:

$$\begin{cases} h_t + (hu)_x + (hv)_y = 0, \\ (hu)_t + \left(hu^2 + \frac{1}{2}gh^2\right)_x + (huv)_y = -ghb_x, \\ (hv)_t + (huv)_x + \left(hv^2 + \frac{1}{2}gh^2\right)_y = -ghb_y, \end{cases} \quad (5.1)$$

where (u, v) is the velocity of the fluid, and h, b and g follow the definitions below (1.1). For the ease of presentation, we denote this Eq. (5.1) by

$$U_t + f(U)_x + g(U)_y = s(h, b),$$

where $U = (h, hu, hv)^T$, and $f(U), g(U)$ are the fluxes. The still water stationary solution we are interested to preserve is

$$h + b = \text{const}, \quad hu = 0, \quad hv = 0. \quad (5.2)$$

We discretize the computational domain into cells $I_{ij} = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \times [y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}]$. For simplicity, we assume a uniform mesh is used, and $\lambda_1 = \Delta t/\Delta x$, $\lambda_2 = \Delta t/\Delta y$. The solution and test spaces are chosen as the space of two-variable polynomials of degree at most k in each cell I_{ij} . The numerical approximation is a piecewise polynomial, still denoted as U , and similarly we have the projection of b into the finite element space, still denoted as b .

We only discuss Euler forward in time for the same reason as in Section 3. The two-dimensional DG method is given by

$$\begin{aligned} & \int_{I_{ij}} \frac{U^{n+1} - U^n}{\Delta t} v dx dy - \int_{I_{ij}} f(U^n) \partial_x v dx dy + \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} \hat{f}_{i+\frac{1}{2}} v(x_{i+\frac{1}{2}}, y) dy \\ & - \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} \hat{f}_{i-\frac{1}{2}} v(x_{i-\frac{1}{2}}, y) dy - \int_{I_{ij}} g(U^n) \partial_y v dx dy \\ & + \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \hat{g}_{j+\frac{1}{2}} v(x, y_{j+\frac{1}{2}}) dx - \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \hat{g}_{j-\frac{1}{2}} v(x, y_{j-\frac{1}{2}}) dx \\ & = \int_{I_{ij}} s(h^n, b) v dx dy, \end{aligned}$$

where $\hat{f}_{i+\frac{1}{2}}(y) = F(U(x_{i+\frac{1}{2}}, y, t), U(x_{i-\frac{1}{2}}, y, t))$, with the Lax–Friedrichs flux

$$F(a_1, a_2) = \frac{1}{2}(f(a_1) + f(a_2) - \alpha_1(a_2 - a_1)),$$

$$\alpha_1 = \max(|u| + \sqrt{gh}),$$

and $\hat{g}_{j+\frac{1}{2}}(x) = G(U(x, y_{j+\frac{1}{2}}^-), U(x, y_{j+\frac{1}{2}}^+, t))$, with

$$G(a_1, a_2) = \frac{1}{2}(g(a_1) + g(a_2) - \alpha_2(a_2 - a_1)),$$

$$\alpha_2 = \max(|v| + \sqrt{gh}).$$

It is straightforward to extend our well-balanced RKDG schemes in Section 2 to two dimensions, and we refer to [40] for the details. Let $\hat{f}_{i+\frac{1}{2}}^l, \hat{f}_{i-\frac{1}{2}}^r, \hat{g}_{j+\frac{1}{2}}^l$ and $\hat{g}_{j-\frac{1}{2}}^r$ be the well-balanced fluxes defined similarly as in (2.11). Then the well-balanced DG method is

$$\begin{aligned} & \int_{I_{ij}} \frac{U^{n+1} - U^n}{\Delta t} v dx dy - \int_{I_{ij}} f(U^n) \partial_x v dx dy + \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} \hat{f}_{i+\frac{1}{2}}^l v(x_{i-\frac{1}{2}}^-, y) dy \\ & - \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} \hat{f}_{i-\frac{1}{2}}^r v(x_{i-\frac{1}{2}}^+, y) dy - \int_{I_{ij}} g(U^n) \partial_y v dx dy \\ & + \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \hat{g}_{j+\frac{1}{2}}^l v(x, y_{j+\frac{1}{2}}^-) dx - \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \hat{g}_{j-\frac{1}{2}}^r v(x, y_{j-\frac{1}{2}}^+) dx \\ & = \int_{I_{ij}} s(h^n, b) v dx dy, \end{aligned} \tag{5.3}$$

The integrals in (5.3) can be approximated by quadratures with sufficient accuracy. Let us assume that we use a Gauss quadrature with L points, which is exact for single variable polynomials of degree $2k + 1$ (see [11] for an analysis of the requirement of the numerical quadrature for the accuracy of the DG solution). We assume:

$$S_i^x = \{x_i^\beta : \beta = 1, \dots, L\}, \quad S_j^y = \{y_j^\beta : \beta = 1, \dots, L\}, \tag{5.4}$$

denote the Gauss quadrature points on $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ and $[y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}]$, respectively. For instance, $(x_{i-\frac{1}{2}}, y_j^\beta)$ ($\beta = 1, \dots, L$) are the Gauss quadrature points on the left edge of the (i, j) cell. We will still need to use the Gauss–Lobatto quadrature rule, and we distinguish the two quadrature rules by adding hats to the Gauss–Lobatto points, i.e.

$$\hat{S}_i^x = \{\hat{x}_i^r : r = 1, \dots, N\}, \quad \hat{S}_j^y = \{\hat{y}_j^r : r = 1, \dots, N\}, \tag{5.5}$$

will denote the Gauss–Lobatto quadrature points on $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ and $[y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}]$, respectively. Recall from Remark 3.5 that the Gauss–Lobatto quadrature (5.5) is introduced to prove the positivity, and Gauss quadrature (5.4) is used to compute the integrals in the DG scheme (5.3). Let \hat{w}_r be the Gauss–Lobatto quadrature weights for the interval $[-1/2, 1/2]$ such that $\sum_{r=1}^N \hat{w}_r = 1$, and we have:

Proposition 5.1. Consider the well-balanced DG scheme (5.3) solving (5.1). Let $U_{ij}^n(x, y) = (h_{ij}^n(x, y), (hu)_{ij}^n(x, y), (hv)_{ij}^n(x, y))^T$ be the DG polynomial in the cell I_{ij} , and \bar{h}_{ij}^n denotes the cell average of $h_{ij}^n(x, y)$ in I_{ij} . If $h_{ij}^n(x_i^\beta, y_j^\beta) \geq 0$ and $h_{ij}^n(\hat{x}_i^r, \hat{y}_j^r) \geq 0$ for all the r, β, i, j , then $\bar{h}_{ij}^{n+1} \geq 0$ under the CFL condition:

$$\frac{\Delta t}{\Delta x} \left(|u| + \sqrt{gh} \right)_{\infty} + \frac{\Delta t}{\Delta y} \left(|v| + \sqrt{gh} \right)_{\infty} \leq \hat{w}_1. \tag{5.6}$$

The proof is straightforward by using Lemma 3.1 and following the same lines as in [42,43].

The linear scaling limiter can enforce the sufficient conditions in the proposition above:

$$\tilde{U}_{ij}^n(x, y) = \theta(U_{ij}^n(x, y) - \bar{U}_{ij}^n) + \bar{U}_{ij}^n, \quad \theta = \min\left\{1, \frac{\bar{h}_{ij}^n}{h_{ij}^n - m_{ij}}\right\}, \tag{5.7}$$

where

$$m_{ij} = \min_{(x,y) \in S_{ij}^n} h_{ij}^n(x, y),$$

$$S_{ij} = \left\{ (x, y) : x \in S_i^x, y \in \hat{S}_j^y, \text{ or } x \in \hat{S}_i^x, y \in S_j^y \right\}. \tag{5.8}$$

Again, we can show this limiter does not destroy accuracy [42,43], and it keeps the conservativity of the water height. By the same argument as in Section 3, the positivity-preserving limiter does not destroy the well-balanced property.

Given the DG polynomial $U_{ij}^n(x, y)$ in interval $I_{i,j}$ at time level n with a non-negative height cell average $\bar{h}_{ij}^n \geq 0$, the algorithm flowchart of our high order well-balanced positivity-preserving DG method with Euler forward in time for the shallow water equations is

- Evaluate $m_{i,j}$ by (5.8).
- In each cell $I_{i,j}$, check if the TVB limiter is needed based on $(h + b, (hu), (hv))^T$ if $m_{ij} \geq 0$, or based on $(h, (hu), (hv))^T$ otherwise. If the answer is yes, perform the TVB limiter on $(h, (hu), (hv))^T$. The DG polynomial after the TVB limiting is still denoted as $U_{ij}^n(x, y)$.
- Evaluate $m_{i,j}$ in (5.8) again, and use the positivity-preserving limiter (5.7) to compute $\tilde{U}_{ij}^n(x, y)$.
- Use $\tilde{U}_{ij}^n(x, y)$ instead of $U_{ij}^n(x, y)$ in the DG scheme (5.3) with the CFL condition (5.6).

For TVD high order time discretizations, we need to perform the algorithm above in each stage for a Runge–Kutta method or in each step for a multistep method.

6. Numerical examples

In this section we present numerical results of our well-balanced positivity-preserving DG methods when applied to the one- and two-dimensional shallow water equations. The third order finite element DG method (i.e. $k = 2$), coupled with the third order TVD Runge–Kutta time discretization (2.6), is implemented in the examples. The CFL number is taken as 0.16, and the TVB constant M in the TVB limiter (2.12) is taken as 0 in most numerical examples, unless otherwise stated. The gravitation constant g is fixed as 9.812 m/s².

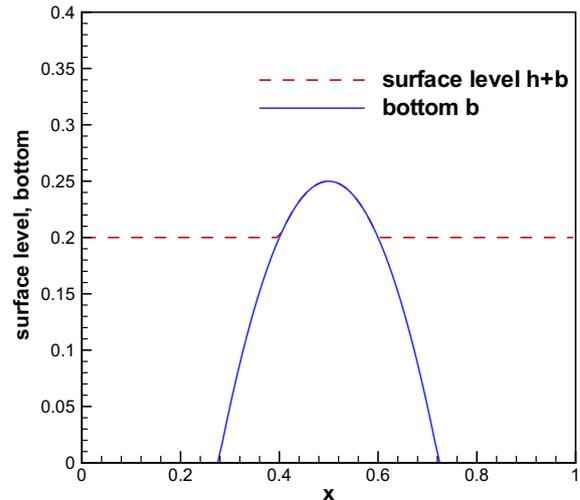


Fig. 6.1. The surface level $h + b$ and the bottom b for the stationary flow in Section 6.1.

Table 6.1
 L^1 and L^∞ errors for different precisions for the stationary solution in Section 6.1.

Precision	L^1 error		L^∞ error	
	h	hu	h	hu
Single	2.89E-07	1.14E-07	5.81E-07	4.20E-07
Double	7.16E-16	1.94E-16	1.11E-15	1.42E-15

Table 6.2
 L^1 errors and numerical orders of accuracy for the example in Section 6.2.

Number of cells	h		hu	
	L^1 error	Order	L^1 error	Order
25	2.12E-03		1.83E-02	
50	1.10E-04	4.27	9.73E-04	4.23
100	1.15E-05	3.26	1.02E-04	3.25
200	8.79E-07	3.72	7.72E-06	3.72
400	9.38E-08	3.23	8.26E-07	3.22
800	1.07E-08	3.13	9.41E-08	3.13

6.1. Test for the well-balanced property

The first test problem is chosen to verify that the DG methods indeed preserve the still water steady state with a non-flat bottom containing a wet/dry interface. The bottom topography is given by the depth function [22]:

$$b(x) = \max(0, 0.25 - 5(x - 0.5)^2), \quad 0 \leq x \leq 1. \quad (6.1)$$

The initial data is the stationary solution:

$$h + b = \max(0.2, b), \quad hu = 0,$$

and a periodic boundary condition is used. This steady state should be exactly preserved. We compute the solution until $t = 0.5$ using 200 uniform cells. The computed surface level $h + b$ and the bottom b are plotted in Fig. 6.1. In order to demonstrate that the still water solution is indeed maintained up to round-off error, we use single- and double-precision to perform the computation, and show the L^1 and L^∞ errors for the water height h (Note: h in this case is not a constant function!) and the discharge hu in Table 6.1 with different

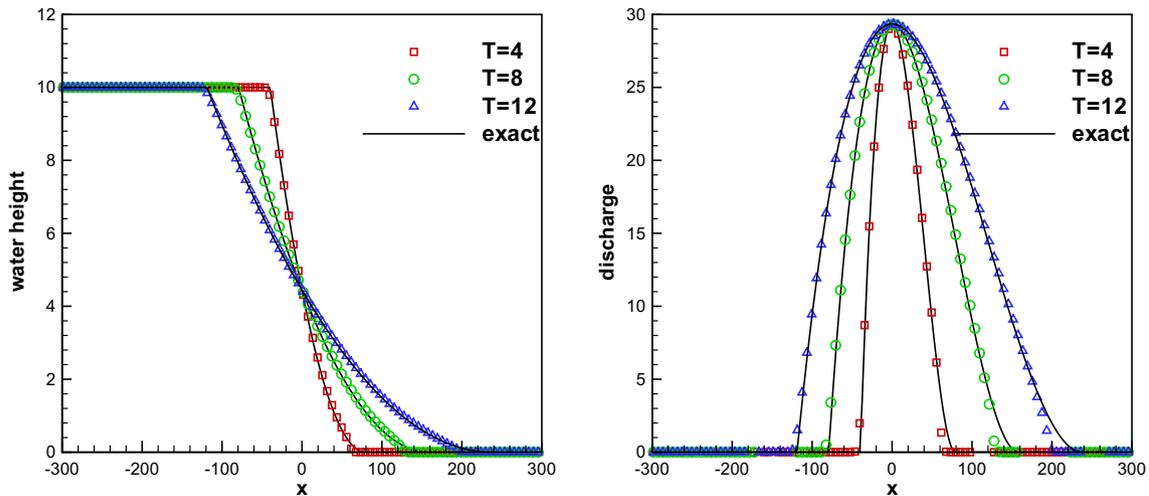


Fig. 6.2. The numerical and exact solutions of the first Riemann problem in Section 6.3 at different time with 200 uniform cells: left: the water height h ; right: the discharge hu .

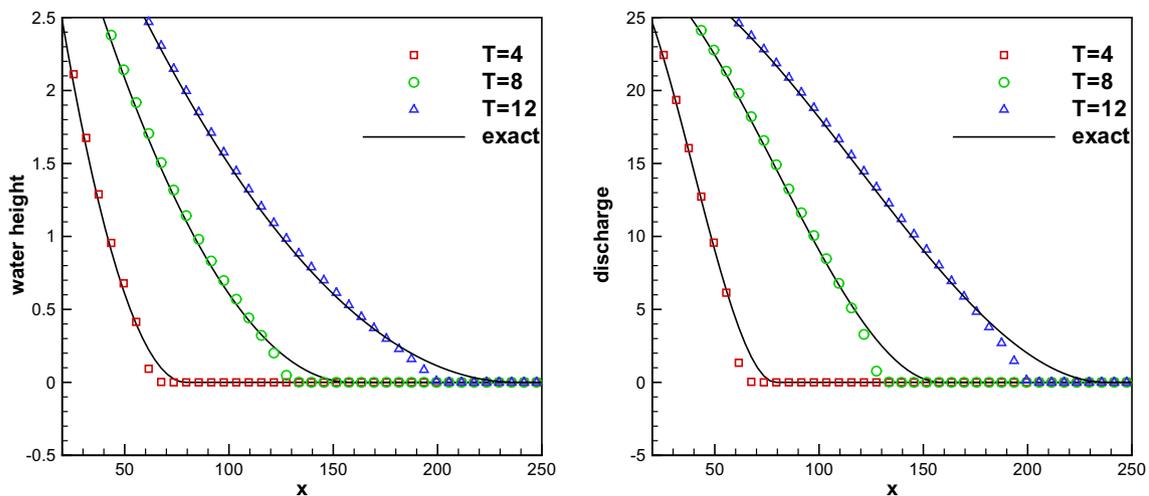


Fig. 6.3. Same as in Fig. 6.2, zoom-in of the wet/dry front.

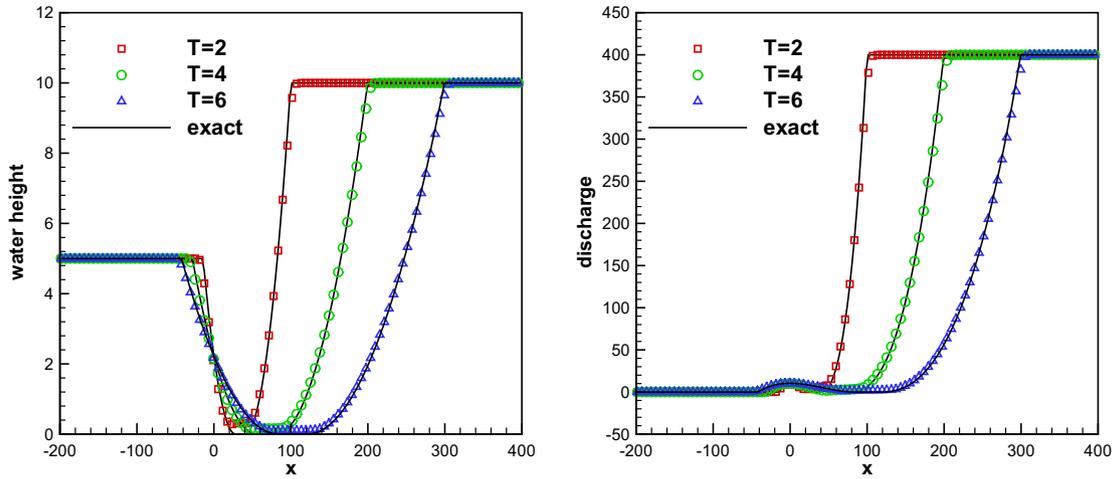


Fig. 6.4. The numerical and exact solutions of the second Riemann problem in Section 6.3 at different time with 200 uniform cells: left: the water height h ; right: the discharge hu .

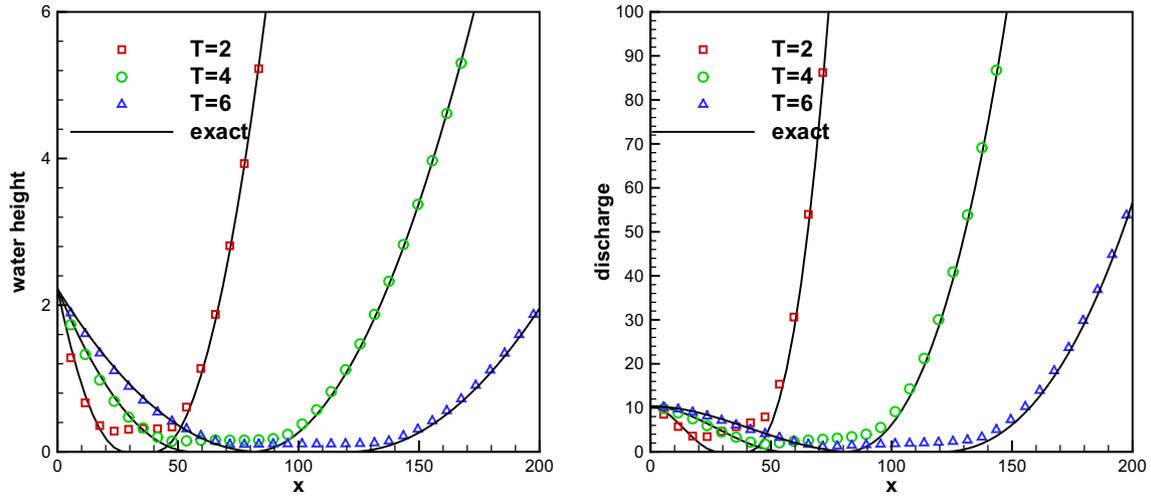


Fig. 6.5. Same as in Fig. 6.4, zoom-in of the wet/dry front.

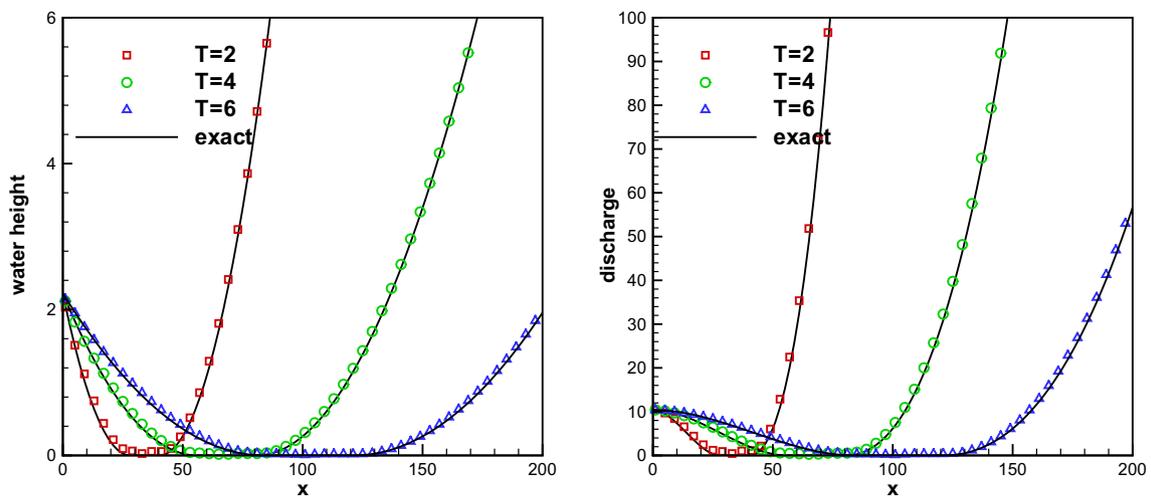


Fig. 6.6. Same as in Fig. 6.4, zoom-in of the wet/dry front, with 300 uniform cells employed.

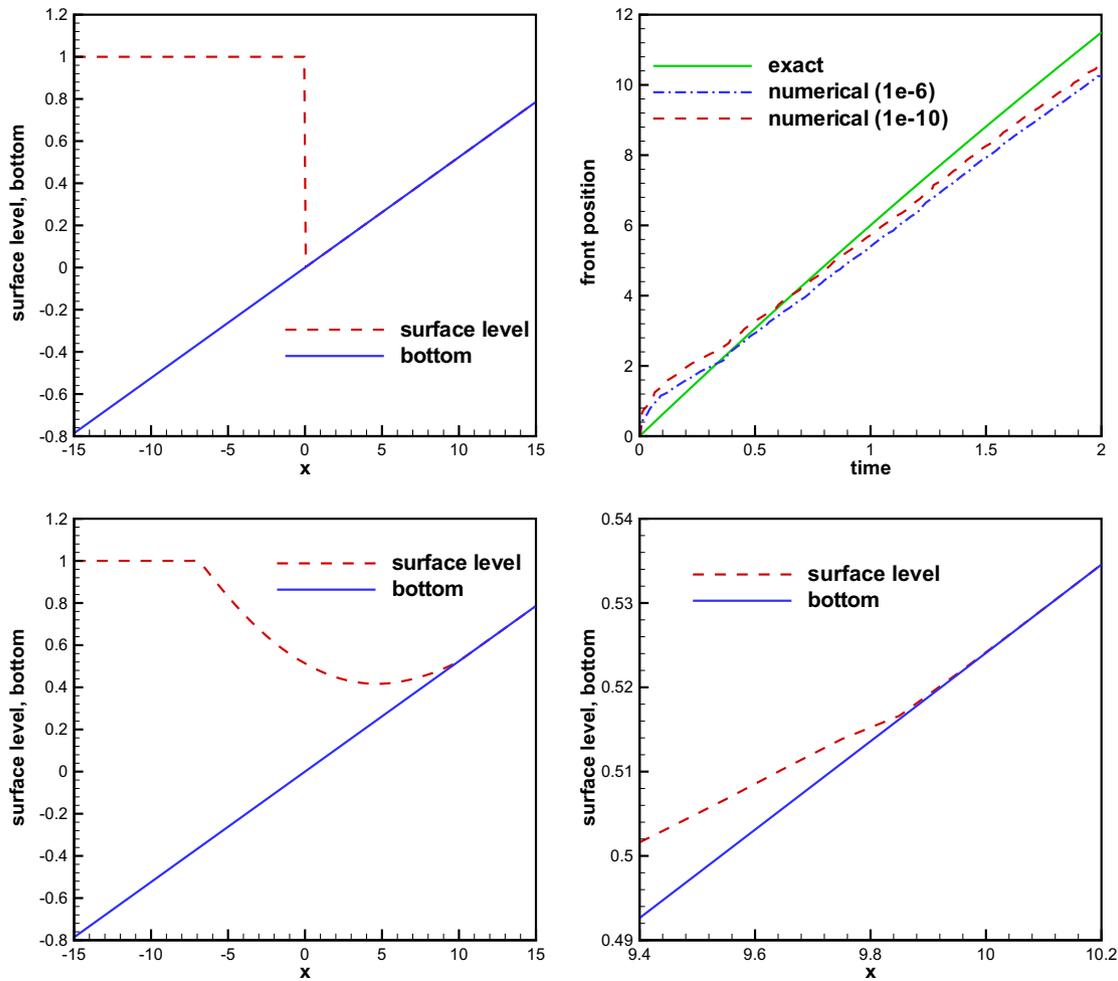


Fig. 6.7. The numerical results of the dambreak problem over an emerging topography with $\alpha = \pi/60$: top left: the initial condition; top right: time evolution of wet/dry front location, two different threshold 10^{-6} and 10^{-10} to determine the front location are used in numerics; bottom left: surface level at time $t = 2$; bottom right: zoom-in of surface level at $t = 2$.

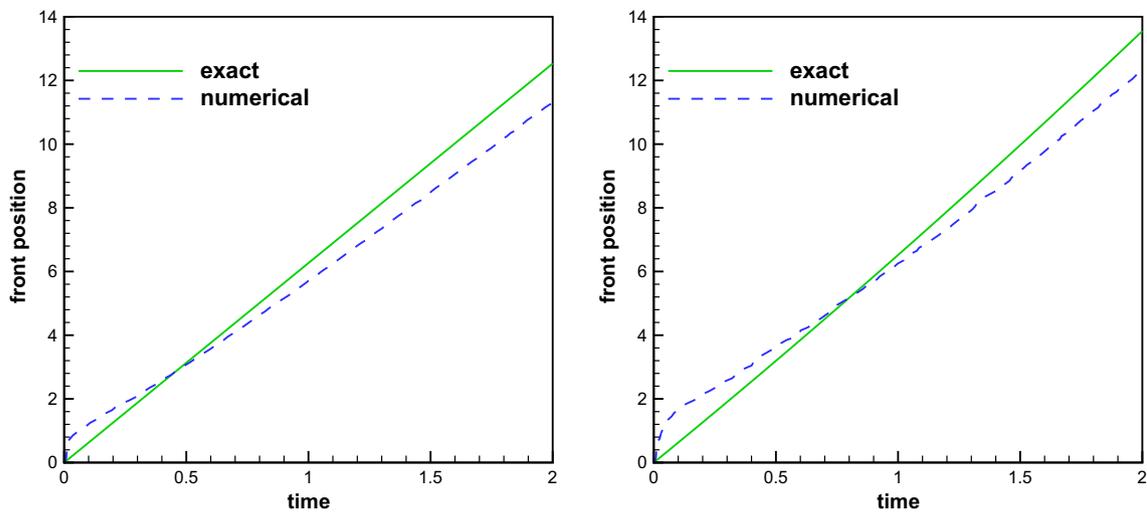


Fig. 6.8. Time evolution of wet/dry front location: left: flat bottom with $\alpha = 0$; right: bottom with decreasing depth $\alpha = -\pi/60$.

precisions. The errors are computed based on the numerical solutions at cell centers. We can clearly see that the L^1 and L^∞ errors

are at the level of round-off errors for different precisions, verifying the well-balanced property.

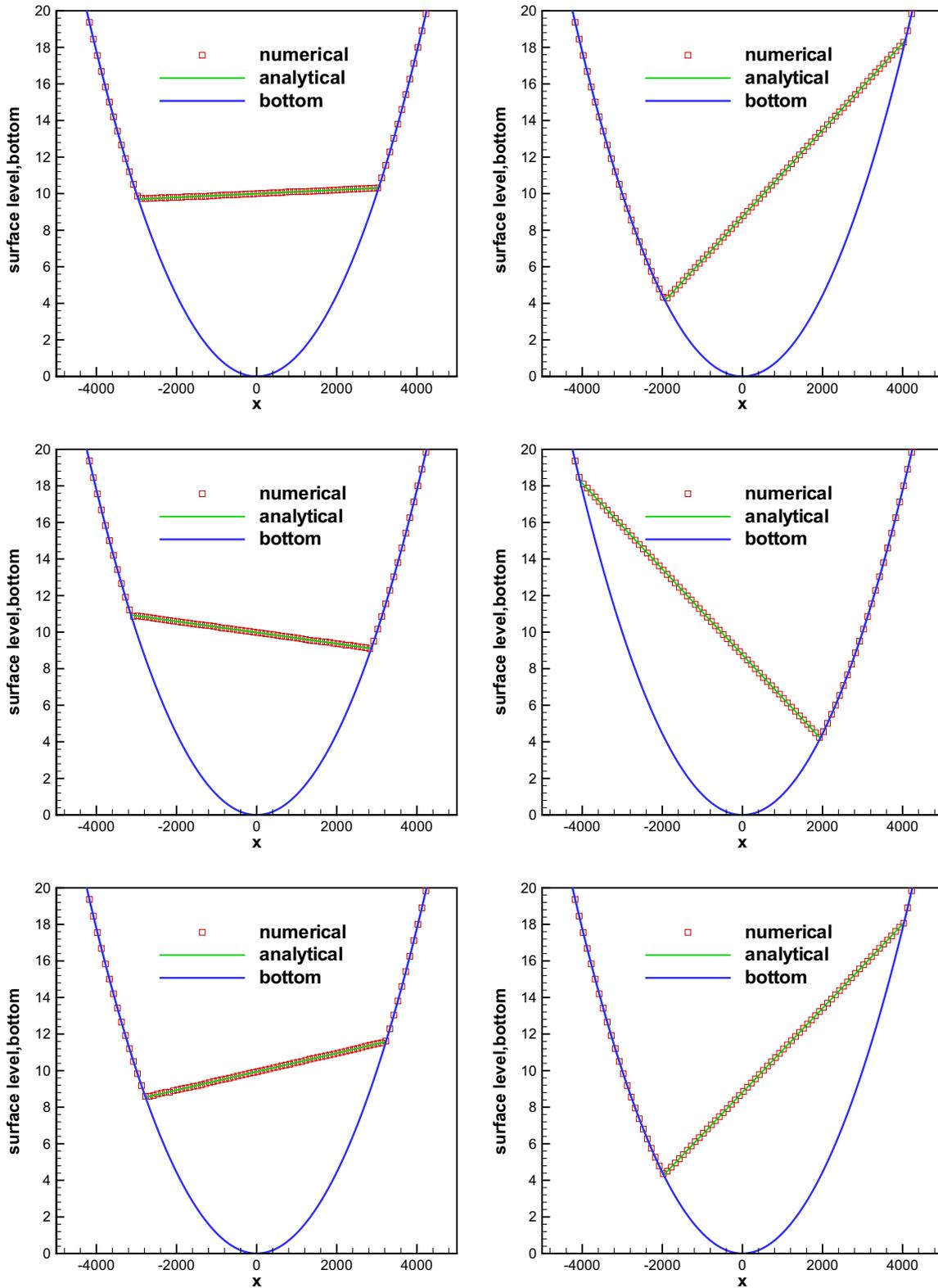


Fig. 6.9. The water surface level in the parabolic bowl problem at different time: top left: $t = 1000$; top right: $t = 2000$; middle left: $t = 3000$; middle right: $t = 4000$; bottom left: $t = 5000$; bottom right: $t = 6000$.

6.2. Accuracy test

In this example we will test the high order accuracy of our schemes for a smooth solution. We have chosen 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 same setup as in [40]. Since the exact solution is not known explicitly for this case, we use the

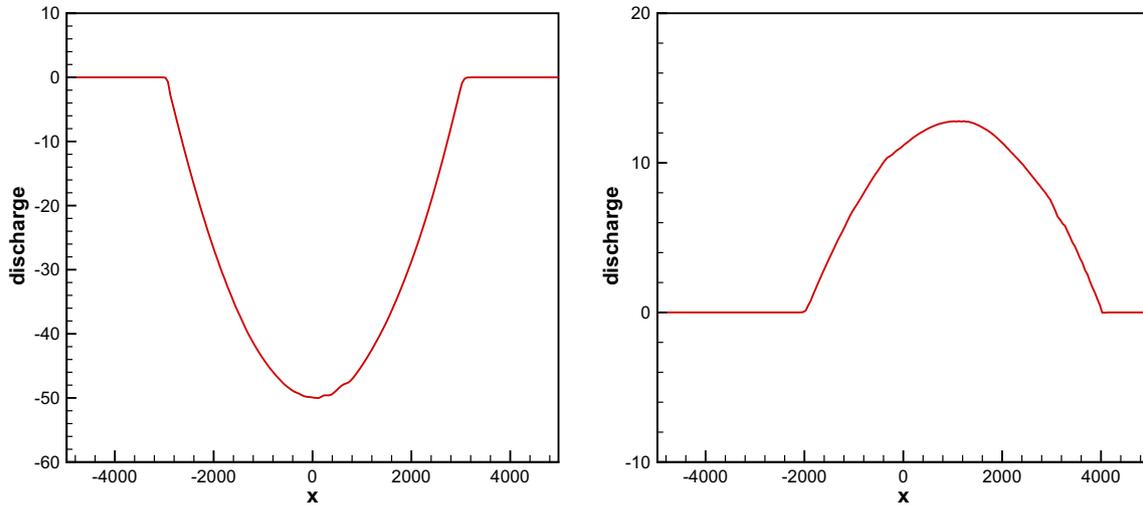


Fig. 6.10. The discharge hu in the parabolic bowl problem at different time: left: $t = 1000$; right: $t = 6000$.

fifth order finite volume WENO scheme from [39] with 12,800 cells to compute a reference solution, and treat this reference solution as the exact solution in computing the numerical errors. The TVB constant M in (2.12) is taken as 32 here, to avoid the accuracy order reduction near the extrema. In general, M should be chosen proportional to the size of the second derivative of the solution near smooth extrema, see [13]. We compute up to $t = 0.1$ when the solution is still smooth (shocks develop later in time for this problem). Table 6.2 contains the L^1 errors for the cell averages and numerical orders of accuracy for the DG scheme. We can clearly see that third order accuracy is achieved.

6.3. Riemann problem over a flat bottom

In this subsection, we consider two Riemann problems containing dry area over a flat bottom (i.e. $b(x) \equiv 0$). These examples were used in [6], and are chosen here to demonstrate the positivity-preserving ability of our methods.

The computational domain for the first test case is set as $[-300, 300]$, and the initial conditions are given by

$$hu(x, 0) = 0 \quad \text{and} \quad h(x, 0) = \begin{cases} 10 & \text{if } x \leq 0, \\ 0 & \text{otherwise.} \end{cases} \quad (6.2)$$

On the left side of 0, still water of height 10 is given, and the right side is dry region. The analytic solution for this problem can be found in [4]. We compute this problem using our well-balanced positivity-preserving methods with simple transmissive boundary conditions and 200 uniform cells. The solutions at time $t = 4, 8$ and 12 are shown in Fig. 6.2. We also plot the exact solutions in these figures to provide a comparison. The zoomed-in version near the wet/dry front at these times is presented in Fig. 6.3. From these figures, we observe that the exact solutions are well captured by the numerical results.

The second test case is on the computational domain $[-200, 400]$. The initial conditions have nonzero velocity, and are given by

$$h(x, 0) = \begin{cases} 5 & \text{if } x \leq 0, \\ 10 & \text{otherwise,} \end{cases} \quad \text{and} \quad u(x, 0) = \begin{cases} 0 & \text{if } x \leq 0, \\ 40 & \text{otherwise,} \end{cases} \quad (6.3)$$

which do not contain dry area. But as the constant initial conditions meet the drying criterion $\sqrt{gh_l} + \sqrt{gh_r} + u_l - u_r < 0$, a dry region emerges and this makes the problem numerically difficult. Two

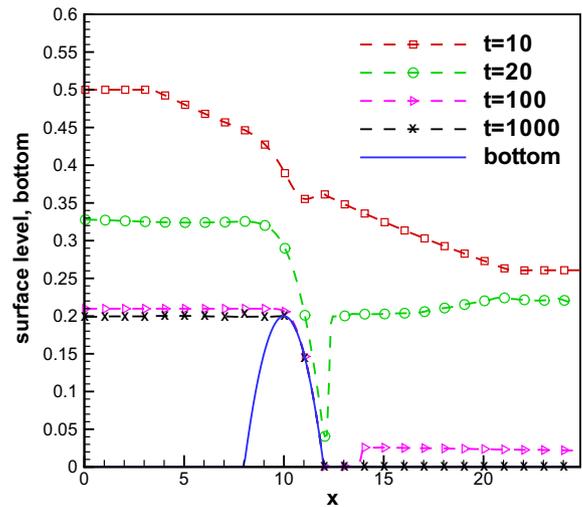


Fig. 6.11. Drain on a non-flat bottom with initial conditions (6.11) and 250 cells. The surface level at different time.

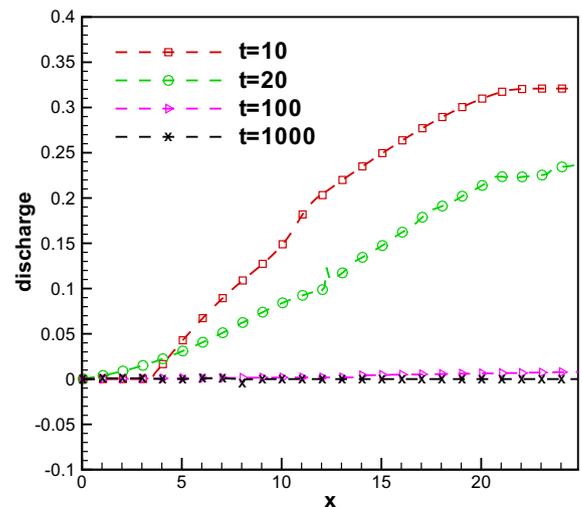


Fig. 6.12. Drain on a non-flat bottom with initial conditions (6.11) and 250 cells. The discharge at different time.

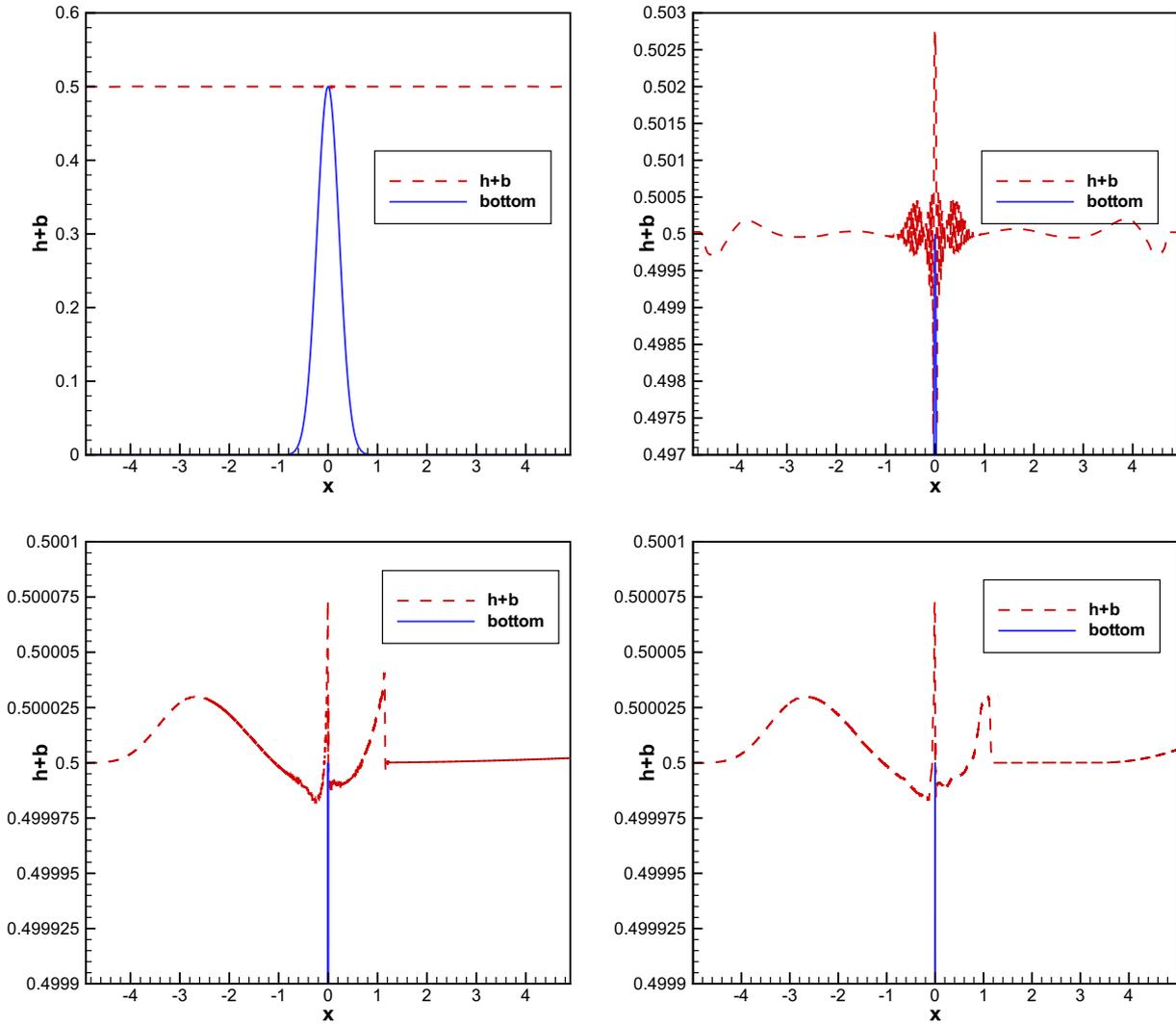


Fig. 6.13. The water surface level and bottom in the small perturbation test with different mesh sizes: top left: 250 cells; top right: zoomed-in version with 250 cells; bottom left: zoomed-in version with 6250 cells; bottom right: zoomed-in version with 6250 cells for the numerical scheme which applies the limiter on $(h + b, hu)^T$.

expansion waves then propagate away from each other. The analytic solution for this problem can be found in [4]. We compute this problem using our well-balanced positivity-preserving methods with simple transmissive boundary conditions and 200 uniform cells. The numerical solutions, as well as the exact solutions, at time $t = 2, 4$ and 6 are shown in Fig. 6.4. We can observe that the numerical solutions agree well with the exact solutions. The comparison near the wet/dry front are shown in Fig. 6.5. There exists observable error near the dry region. We repeat the test with 300 uniform cells and the corresponding solutions are plotted in Fig. 6.6, where such error are significantly reduced and a nice agreement between the numerical and exact solutions is observed.

We have also run this test case using the well-balanced DG methods without the positivity-preserving limiter. Negative water height was generated during the computation, which caused blow-up immediately. This confirms the positivity-preserving property of our method.

6.4. Dam break over a plane

In this subsection, we consider some examples which contain a non-flat bottom, and test the performance of the positivity-preserving methods under this condition. These test cases have been previously considered in [9].

The bottom topography is chosen as

$$b(x) = 1 - x \tan(\alpha), \tag{6.4}$$

with some angle α which will be defined later. The computational domain is set as $[-15, 15]$ and the initial conditions are given by

$$hu(x, 0) = 0 \quad \text{and} \quad h(x, 0) = \begin{cases} 1 - b(x) & \text{if } x \leq 0, \\ 0 & \text{otherwise,} \end{cases} \tag{6.5}$$

i.e., an initial still water on the left side of 0, and a dry region on the right. The discharge $q = 0$ is imposed at the left boundary $x = -15$ and a free boundary condition is considered at the right boundary $x = 15$.

For this particular problem, the position of the wet/dry front and its velocity can be exactly computed, as shown in [9]. They are given by

$$x_f(t) = 2t\sqrt{gh_0 \cos(\alpha)} - \frac{1}{2}gt^2 \tan(\alpha), \tag{6.6}$$

$$u_f(t) = 2\sqrt{gh_0 \cos(\alpha)} - gt \tan(\alpha),$$

where α is the angle defined in (6.4) and $h_0 = 1$ in this experiment.

Different values of the coefficient α produce different forms of bottom, and then different kinds of wet/dry fronts will appear. We consider three typical cases: an emerging topography $\alpha = \pi/$

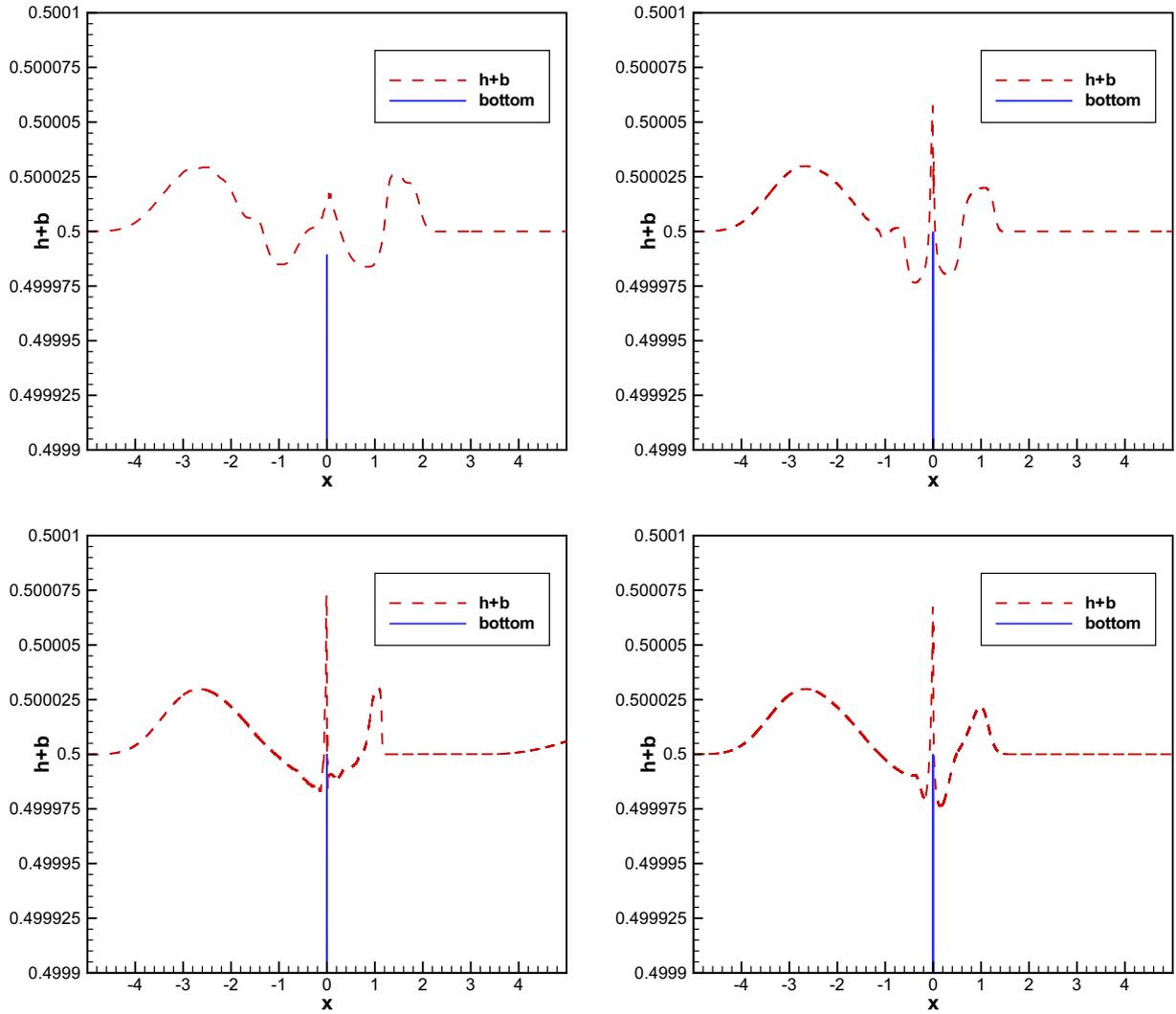


Fig. 6.14. The water surface level and bottom in the small perturbation test with different mesh size: top left: 250 cells; top right: 1250 cells; bottom left: 6250 cells; bottom right: 38400 cells.

60, the flat bottom $\alpha = 0$ and bottom with decreasing depth $\alpha = -\pi/60$.

We run the simulation until the stopping time $t = 2$, with 300 uniform cells. Numerical results of the case $\alpha = \pi/60$ are shown in Fig. 6.7. The initial condition is plotted on the top left, and the time evolution of the wet/dry front location is on the top right. The wet/dry front is defined as the first place where the water height exceeds 10^{-6} if counted from the right to the left. We also show the exact location (6.6) in the figure to provide a comparison, and observe that the numerical front moves faster at first and slower as time increases. To test the sensitivity of the front position from the threshold 10^{-6} , we repeat the test with a new threshold 10^{-10} , and the new front location is also shown in the figure. The surface level at time $t = 2$ is presented on the bottom left, and its zoom-in version near the front is on the bottom right. The time evolutions of the wet/dry front location for the cases $\alpha = 0$ and $\alpha = -\pi/60$ are shown in Fig. 6.8, together with the exact front location.

6.5. Parabolic bowl

For one-dimensional shallow water equations with a parabolic bottom topography, analytic solutions have been derived by Samp-

son et al. [32]. This provides a good test case for our numerical methods. This example has been used in [22] for the shallow water equations with the friction source term.

We take the parabolic bottom:

$$b(x) = h_0(x/a)^2, \quad (6.7)$$

with constants h_0 and a to be specified later. The computational domain is set as $[-5000, 5000]$. The analytical water surface, for the shallow water equations without the friction source term, is given by

$$h(x, t) + b(x) = h_0 - \frac{B^2}{4g} \cos(2\omega t) - \frac{B^2}{4g} - \frac{Bx}{2a} \sqrt{\frac{8h_0}{g}} \cos(\omega t), \quad (6.8)$$

where $\omega = \sqrt{2gh_0}/a$ and B is a given constant. The exact location of the wet/dry front takes the form:

$$x_0 = -\frac{B\omega a^2}{2gh_0} \cos(\omega t) \pm a. \quad (6.9)$$

We fix these coefficients to be $a = 3000$, $B = 5$ and $h_0 = 10$ for our test case. The initial condition is then defined by (6.8) (for the water height) and the zero discharge. Because the flow cannot reach the boundaries, we can pick any boundary conditions and they have

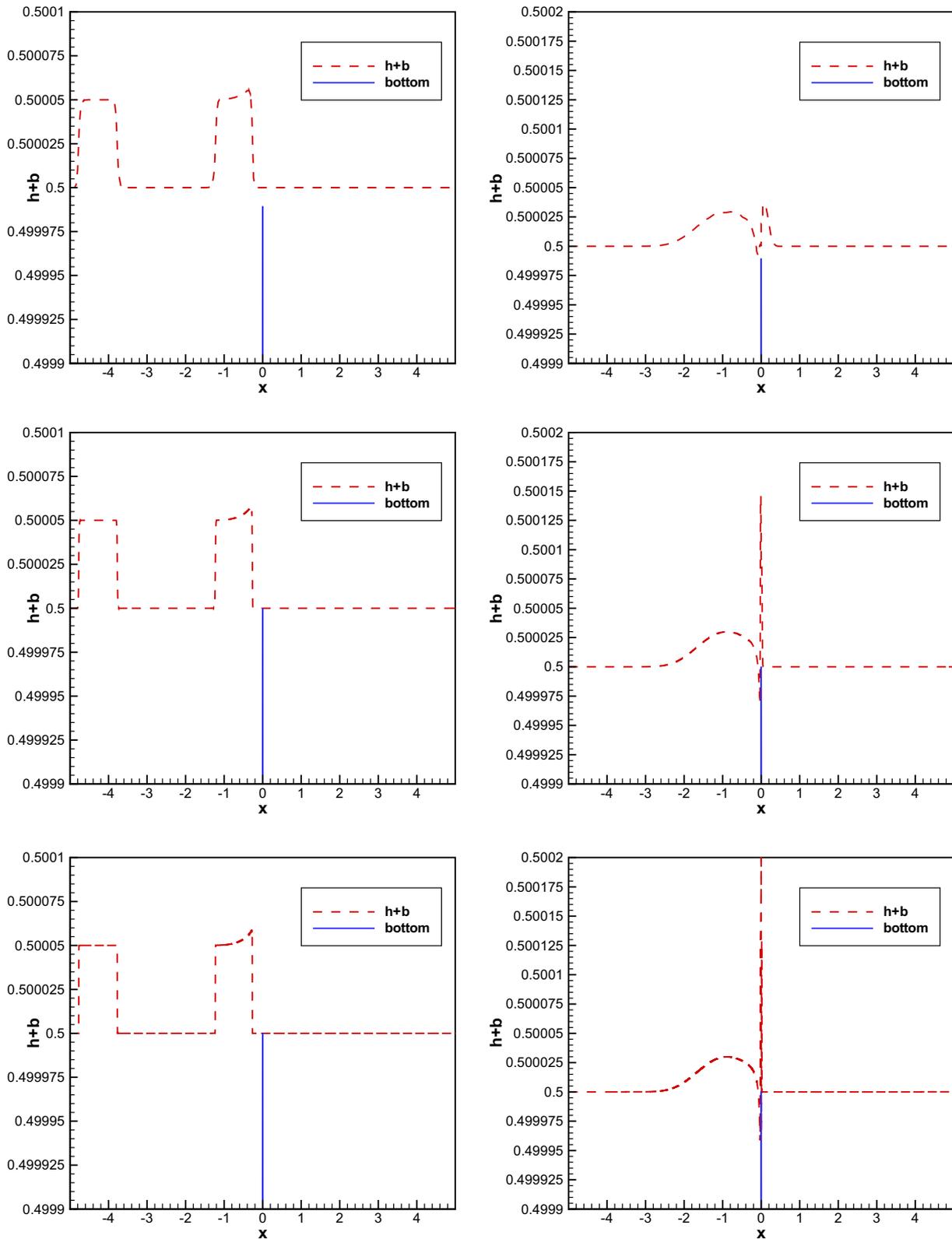


Fig. 6.15. The water surface level and bottom in the small perturbation test with different mesh size and stopping time: left: time $t = 0.8$; right: time $t = 1.6$; top: 250 cells; middle: 1250 cells; bottom: 6250 cells.

no impact on the numerical solutions. We run the simulation until $T = 6000$ with 200 uniform cells, and plot the numerical water surface at different times in Fig. 6.9. We also include the analytical solu-

tion to provide a comparison, and a nice agreement can be observed. This confirms the positivity-preserving property of our methods. The discharges hu at time 1000 and 6000 are also plotted in Fig. 6.10.

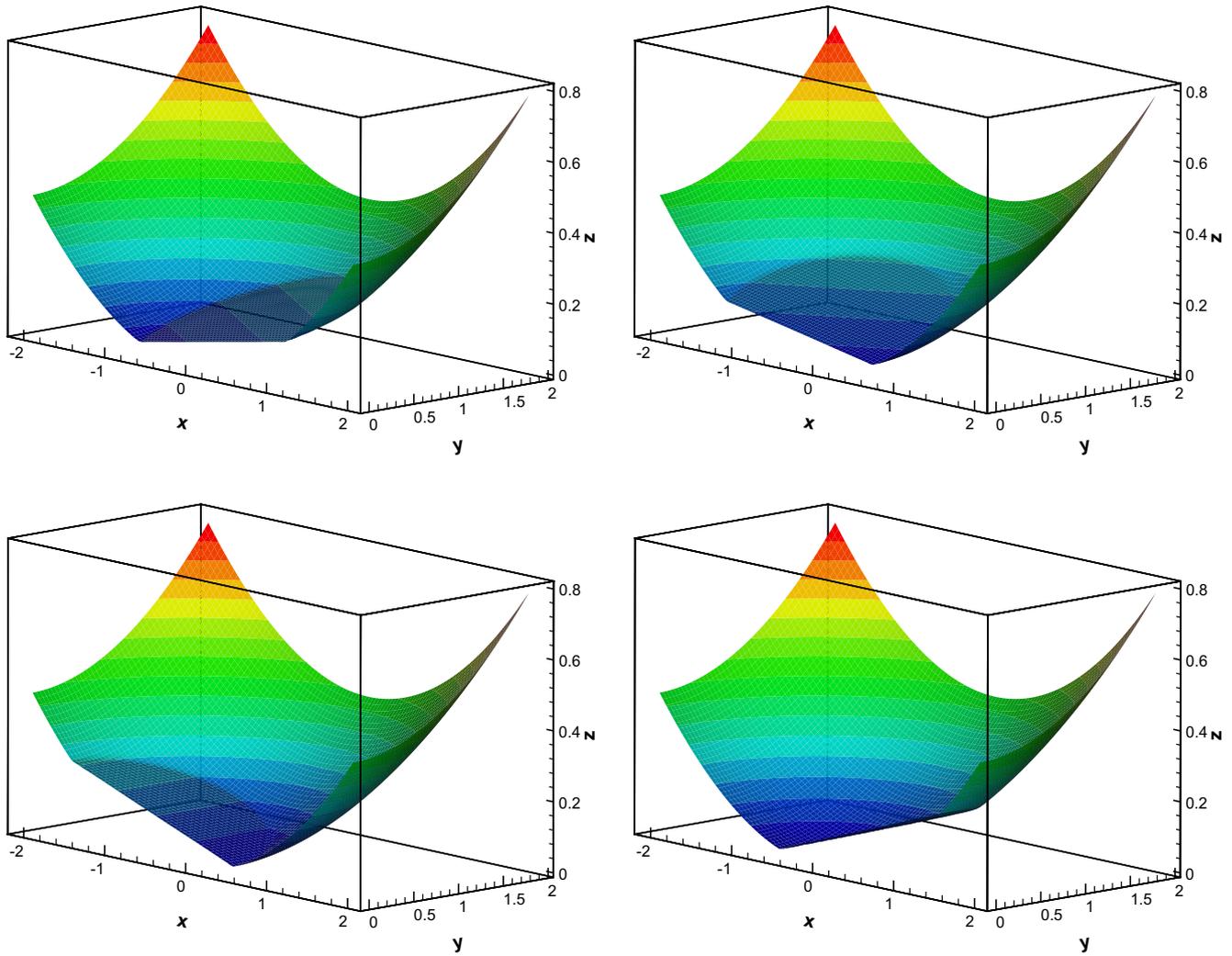


Fig. 6.16. The water surface level in the two-dimensional oscillating lake problem at different time: top left: $t = T/6$; top right: $t = T/3$; bottom left: $t = T/2$; bottom right: $t = 2T$.

6.6. Drain on a non-flat bottom

In this example, taken from [19], we solve the shallow water equations on the bottom:

$$b(x) = \begin{cases} 0.2 - 0.05(x - 10)^2 & \text{if } 8 \leq x \leq 12, \\ 0 & \text{otherwise,} \end{cases} \quad (6.10)$$

in the computational domain $[0, 25]$. The initial data is a still flat water:

$$h(x, 0) = 0.5 - b(x), \quad hu(x, 0) = 0. \quad (6.11)$$

The left boundary condition is a free condition on h and zero on hu . The right boundary condition is an outlet condition on a dry bed (refer to [19] for the details).

Two-hundred and fifty uniform cells are used in the computation. The solutions at different times $T = 10, 20, 100$ and 1000 are shown in Figs. 6.11 and 6.12. The outlet boundary condition on the right allows the water to freely flow out of the domain on the right, and a dry region is developed near the right side of the bump first. After a long time, the solution reaches a steady state, which is a still water on the left of the bump, and a dry state on the right. The numerical solutions reflect this pattern well and converge to the expected steady state.

6.7. Small perturbation test

In this test case, we study a very small perturbation of a steady state problem with a non-flat exponential bottom. In a channel $[-5, 5]$, we consider the bottom function:

$$b(x) = 0.5 \exp(-10x^2),$$

with the initial condition given by

$$(hu)(x, 0) = 0 \quad \text{and} \quad h(x, 0) = \begin{cases} 0.5 - b(x) + 0.0001 & \text{if } -3 \leq x \leq -2, \\ 0.5 - b(x) & \text{otherwise.} \end{cases}$$

Theoretically, this disturbance should split into two waves, propagating left and right at the characteristic speeds $\pm\sqrt{gh}$. The results at time $t = 2.4$, when the right travelling wave passes the bottom, are shown in Fig. 6.13. Top left is the plot for the water surface and bottom with 250 cells. A zoomed-in version on the horizontal interval is shown in top right, where we can observe that the bottom tip is very singular at this scale. The numerical scheme cannot resolve such singular tip with a coarse mesh of 250 points, which corresponds to less than one point in the tip, resulting in the artifact (oscillations) at this scale. As we increase the resolution to 6250 cells, such that the mesh size Δx is comparable to the width of bottom tip at the zoomed-in scale, the solution is well resolved and the spurious oscillation disappears, as shown in bottom left.

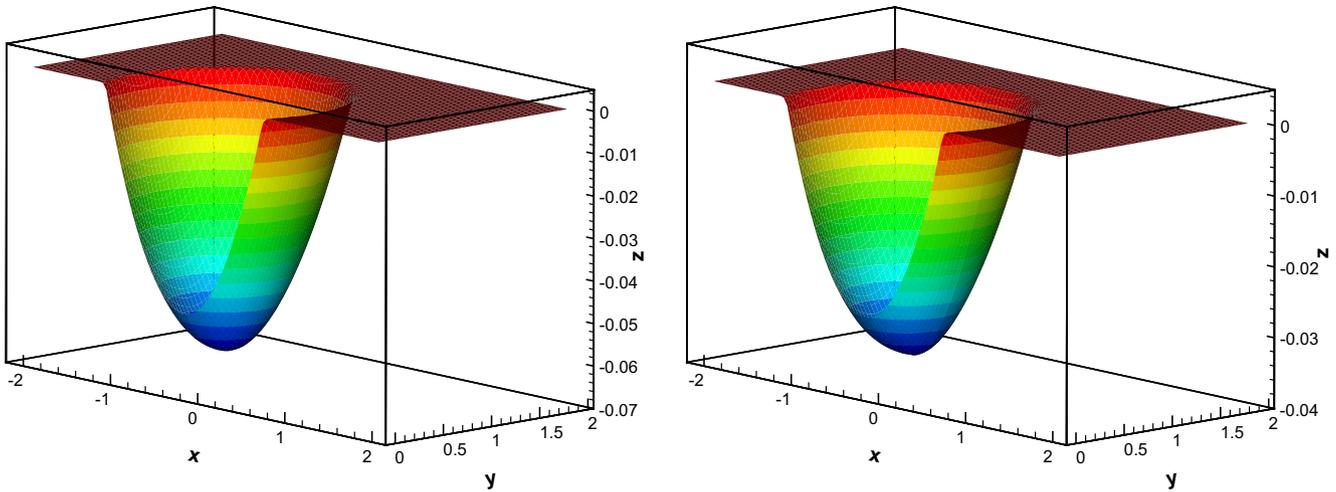


Fig. 6.17. The discharge in the two-dimensional oscillating lake problem at time $t = T/3$: left: h_u ; right: h_v .

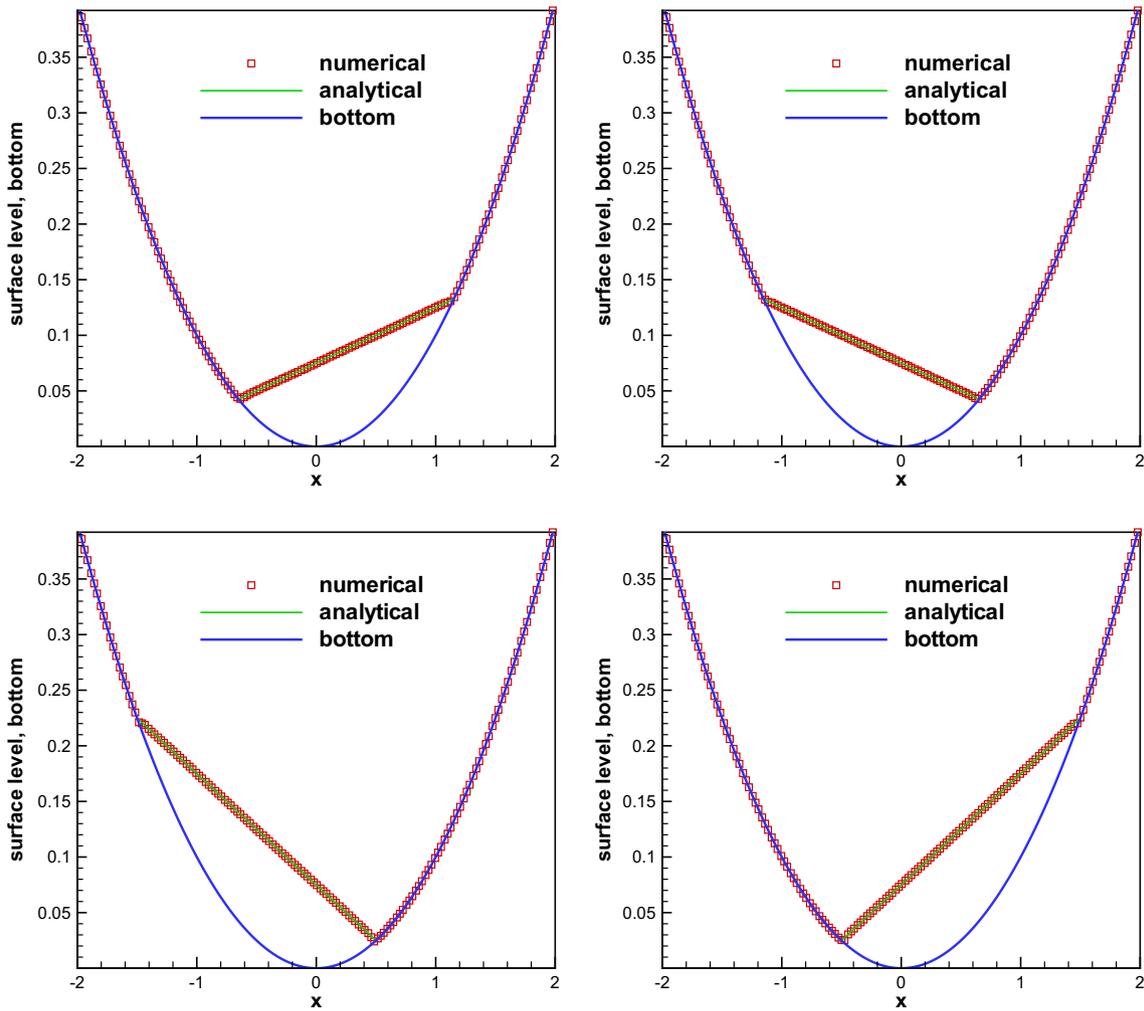


Fig. 6.18. The 2D plot of the water surface level in the two-dimensional oscillating lake problem along the line $y = 0$ at different time: top left: $t = T/6$; top right: $t = T/3$; bottom left: $t = T/2$; bottom right: $t = 2T$.

For this particular example, the numerical solution has better resolution if the TVB limiter is implemented on $(h + b, hu)^T$, as shown in

bottom right of Fig. 6.13 for the numerical result with 6250 points. Because of the strong singularity of the bottom tip at the zoomed-in

scale for this particular example, $h + b$ is a much smoother function than h , explaining the better performance when the limiter is implemented on $h + b$ rather than h .

Next, we will show more numerical results with different mesh sizes of 250, 1250, 6250 and 38400, respectively, for the case of applying the limiter on $(h + b, hu)^T$, in Fig. 6.14. We can clearly observe numerical convergence with grid refinement, starting at 1250 cells. To demonstrate that the “overshoot” right on top of the bottom tip is physical, we run the solution with a very refined mesh of 38400 cells, which agrees with the solution of 1250 cells well. In Fig. 6.15, we plot the numerical results at two different times, namely at $t = 0.8$ (before the right travelling wave reaches the bottom tip) and at $t = 1.6$ (when the right travelling wave just hits the bottom tip).

6.8. Two-dimensional oscillating lake

In this last test, we study a two-dimensional example proposed in [18]. The main purpose is to check the performance of the positivity-preserving limiter in two dimensions.

We consider a rectangular computational domain $[-2, 2] \times [-2, 2]$. The parabolic bottom topography takes the form:

$$b(x, y) = h_0 \frac{x^2 + y^2}{a^2}, \quad (6.12)$$

with constants h_0 and a to be specified later. The analytical solutions, for the two-dimensional shallow water equations without the friction source term, are given by [18]:

$$\begin{aligned} h(x, y, t) &= \max \left(0, \frac{\sigma h_0}{a^2} (2x \cos(\omega t) + 2y \sin(\omega t) - \sigma + 0.1 - b(x, y)) \right), \\ u(x, y, t) &= -\sigma \omega \sin(\omega t), \quad v(x, y, t) = \sigma \omega \cos(\omega t), \end{aligned} \quad (6.13)$$

which are periodic with the period $T = 2\pi/\omega$ and $\omega = \sqrt{2gh_0}/a$.

We fix these coefficients to be $a = 1$, $\sigma = 0.5$ and $h_0 = 0.1$ for our test case. The initial conditions are then defined by (6.13) with $t = 0$. Because the flow cannot reach the boundaries, we can pick any boundary conditions and they have no impact on the numerical solutions. We run the simulation until time $2T$ with 100×100 uniform cells, and plot the numerical water surface at different times in Fig. 6.16. The discharge at time $t = T/3$ are shown in Fig. 6.17. We also plot the water surface along the line $y = 0$, and compare the results with the analytical solution to provide a comparison in Fig. 6.18, where a nice agreement can be observed.

7. Concluding remarks

In this paper we have presented a simple positivity-preserving limiter based on DG methods for the shallow water equations, which can keep the water height non-negative under suitable CFL condition, can preserve the mass conservation and at the same time does not affect the high order accuracy for the general solutions. We then incorporate this limiter into a well-balanced DG method presented in [40], with a corresponding change in the slope limiter procedure. This method has been extended to the two-dimensional problem with rectangular meshes. Compared with other positive preserving methods in the literature, our approach has the advantage of simplicity, high order accuracy for smooth solutions and well-balanced property. Extensive numerical examples are provided to demonstrate the well-balanced property, accuracy, positivity-preserving property, and non-oscillatory shock resolution of the proposed numerical method.

In this paper we have only shown a straightforward extension of the one dimensional algorithm to two-dimensional DG scheme on a rectangular mesh. For triangular meshes, the idea of rewriting

the scheme as a convex combination of monotone schemes is still plausible with the introduction of a special quadrature rule, see [44] for such an extension for Euler equations. This, together with generalization to high order accurate finite difference positivity-preserving methods, constitute the ongoing work.

References

- [1] 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* 2004;25:2050–65.
- [2] Bermudez A, Vazquez ME. Upwind methods for hyperbolic conservation laws with source terms. *Comput Fluids* 1994;23:1049–71.
- [3] Berthon C, Marche F. A positive preserving high order VFRoe scheme for shallow water equations: a class of relaxation schemes. *SIAM J Sci Comput* 2008;30:2587–612.
- [4] Bokhove O. Flooding and drying in discontinuous Galerkin finite-element discretizations of shallow-water equations. Part 1: one dimension. *J Sci Comput* 2005;22:47–82.
- [5] Brufau P, Vázquez-Cendón ME, García-Navarro P. A numerical model for the flooding and drying of irregular domains. *Int J Numer Meth Fluids* 2002;39:247–75.
- [6] Bunya S, Kubatko EJ, Westerink JJ, Dawson C. A wetting and drying treatment for the Runge–Kutta discontinuous Galerkin solution to the shallow water equations. *Comput Meth Appl Mech Eng* 2009;198:1548–62.
- [7] Caleffi V, Valiani A, Bernini A. Fourth-order balanced source term treatment in central WENO schemes for shallow water equations. *J Comput Phys* 2006;218:228–45.
- [8] Canestrelli A, Siviglia A, Dumbser M, Toro EF. Well-balanced high-order centred schemes for non-conservative hyperbolic systems. Applications to shallow water equations with fixed and mobile bed. *Adv Water Resour* 2009;32:834–44.
- [9] Castro MJ, Gallardo JM, Parés C. High order finite volume schemes based on reconstruction of states for solving hyperbolic systems with nonconservative products. Applications to shallow-water systems. *Math Comput* 2006;75:1103–34.
- [10] Castro MJ, González-Vida JM, Parés C. Numerical treatment of wet/dry fronts in shallow flows with a modified Roe scheme. *Math Mod Meth Appl Sci* 2006;16:897–931.
- [11] Cockburn B, Hou S, Shu C-W. The Runge–Kutta local projection discontinuous Galerkin nite element method for conservation laws IV: the multidimensional case. *Math Comput* 1990;54:545–81.
- [12] Cockburn B, Karniadakis G, Shu C-W. The development of discontinuous Galerkin methods. In: Cockburn B, Karniadakis G, Shu C-W, editors. *Discontinuous Galerkin methods: theory, computation and applications*. Lecture notes in computational science and engineering, part I: overview, vol. 11. Springer; 2000. p. 3–50.
- [13] Cockburn B, Shu C-W. TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework. *Math Comput* 1989;52:411–35.
- [14] Cockburn B, Shu C-W. The Runge–Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems. *J Comput Phys* 1998;141:199–224.
- [15] Dawson C, Proft J. Discontinuous and coupled continuous/discontinuous Galerkin methods for the shallow water equations. *Comput Meth Appl Mech Eng* 2002;191:4721–46.
- [16] Ern A, Piperno S, Djadel K. A well-balanced Runge–Kutta discontinuous Galerkin method for the shallow-water equations with flooding and drying. *Int J Numer Meth Fluids* 2008;58:1–25.
- [17] Eskilsson C, Sherwin SJ. A triangular spectral/hp discontinuous Galerkin method for modelling 2D shallow water equations. *Int J Numer Meth Fluids* 2004;45:605–23.
- [18] Gallardo JM, Parés C, Castro M. On a well-balanced high-order finite volume scheme for shallow water equations with topography and dry areas. *J Comput Phys* 2007;227:574–601.
- [19] Gallouët T, Hérard J-M, Seguin N. Some approximate Godunov schemes to compute shallow-water equations with topography. *Comput Fluids* 2003;32:479–513.
- [20] Giraldo FX, Hesthaven JS, Warburton T. Nodal high-order discontinuous Galerkin methods for the spherical shallow water equations. *J Comput Phys* 2002;181:499–525.
- [21] Kurganov A, Levy D. Central-upwind schemes for the Saint–Venant system. *Math Model Numer Anal* 2002;36:397–425.
- [22] Liang Q, Marche F. Numerical resolution of well-balanced shallow water equations with complex source terms. *Adv Water Resour* 2009;32:873–84.
- [23] Lukáčová-Medviiová M, Noelle S, Kraft M. Well-balanced finite volume evolution Galerkin methods for the shallow water equations. *J Comput Phys* 2007;221:122–47.
- [24] 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* 2003;129:628–36.
- [25] Noelle S, Pankratz N, Puppo G, Natvig JR. Well-balanced finite volume schemes of arbitrary order of accuracy for shallow water flows. *J Comput Phys* 2006;213:474–99.

- [26] 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* 2007;226:29–58.
- [27] Noelle S, Xing Y, Shu C-W. High-order well-balanced schemes. In: Russo G, Puppo G, editors. Numerical methods for relaxation systems and balance equations. Italy: Quaderni di Matematica, Dipartimento di Matematica, Seconda Università di Napoli; 2009.
- [28] Parés C. Numerical methods for nonconservative hyperbolic systems. A theoretical framework. *SIAM J Numer Anal* 2006;44:300–21.
- [29] Parés C, Castro M. On the well-balance property of Roe's method for nonconservative hyperbolic systems. Applications to shallow-water systems. *Math Model Numer Anal (M²AN)* 2004;38:821–52.
- [30] Perthame B, Shu C-W. On positivity preserving finite volume schemes for Euler equations. *Numer Math* 1996;273:119–30.
- [31] Ricchiuto M, Bollermann A. Stabilized residual distribution for shallow water simulations. *J Comput Phys* 2009;228:1071–115.
- [32] Sampson J, Easton A, Singh M. Moving boundary shallow water flow above parabolic bottom topography. *ANZIAM J* 2006;47:C373–8.
- [33] Shu C-W. TVB uniformly high-order schemes for conservation laws. *Math Comput* 1987;49:105–21.
- [34] Shu C-W. Total-variation-diminishing time discretizations. *SIAM J Sci Stat Comput* 1988;9:1073–84.
- [35] Shu C-W, Osher S. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *J Comput Phys* 1988;77:439–71.
- [36] Vukovic S, Sopta L. ENO and WENO schemes with the exact conservation property for one-dimensional shallow water equations. *J Comput Phys* 2002;179:593–621.
- [37] Xing Y, Shu C-W. High order finite difference WENO schemes with the exact conservation property for the shallow water equations. *J Comput Phys* 2005;208:206–27.
- [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* 2006;27:477–94.
- [39] 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* 2006;214:567–98.
- [40] 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* 2006;1:100–34.
- [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*; in press. doi:10.1007/s10915-010-9377-y.
- [42] Zhang X, Shu C-W. On maximum-principle-satisfying high order schemes for scalar conservation laws. *J Comput Phys* 2010;229:3091–120.
- [43] Zhang X, Shu C-W. On positivity preserving high order discontinuous Galerkin schemes for compressible euler equations on rectangular meshes. *J Comput Phys*; in press. doi:10.1016/j.jcp.2010.08.016.
- [44] 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*; submitted for publication.
- [45] Zhou JG, Causon DM, Mingham CG, Ingram DM. The surface gradient method for the treatment of source terms in the shallow-water equations. *J Comput Phys* 2001;168:1–25.