Positivity-preserving well-balanced discontinuous Galerkin methods for the shallow water equations on unstructured triangular meshes

Yulong Xing[1] and Xiangxiong Zhang[2]

Abstract

The shallow water equations model flows in rivers and coastal areas and have wide applications in ocean, hydraulic engineering, and atmospheric modeling. In [36], the authors constructed high order discontinuous Galerkin methods for the shallow water equations 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. In this paper, we explore the extension of these methods on unstructured triangular meshes. The simple positivity-preserving limiter is reformulated, and we prove that the resulting scheme guarantees the positivity of the water depth. Extensive numerical examples are provided to verify the positivity-preserving property, well-balanced property, high-order accuracy, and good resolution for smooth and discontinuous solutions.

Keywords: shallow water equations; discontinuous Galerkin method; high order accuracy; well-balanced; positivity-preserving methods; wetting and drying treatment

---

[1]Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831 and Department of Mathematics, University of Tennessee, Knoxville, TN 37996. E-mail: xingy@math.utk.edu. Fax: (865) 241-4811. Research is sponsored by the National Science Foundation grant DMS-1216454, ORNL's Laboratory Directed Research and Development funds, and the U. S. Department of Energy, Office of Advanced Scientific Computing Research. The work was partially performed at ORNL, which is managed by UT-Battelle, LLC, under Contract No. DE-AC05-00OR22725.

[2]Department of Mathematics, Massachusetts Institute of Technology, Cambridge, MA 02139. E-mail: zhangxx@math.mit.edu.

1   Introduction

The main goal of this paper is to present high order accurate discontinuous Galerkin (DG) methods for the shallow water equations on unstructured triangular meshes, which are not only well-balanced for the still water steady state solutions, but also preserve the non-negativity of the water depth. The shallow water equations with a non-flat bottom topography play a critical role in the modeling and simulation of flows in rivers, lakes and coastal areas. They have wide applications in ocean, hydraulic engineering and atmospheric modeling. The two dimensional shallow water equations take the form

$$
\begin{cases}
h_t + (hu)_x + (hv)_y = 0 \\
(hu)_t + \left(hu^2 + \dfrac{1}{2}gh^2 \right)_x + (huv) = -ghb_x \\
(hv)_t + (huv)_x + \left(hv^2 + \dfrac{1}{2}gh^2 \right) = -ghb_y,
\end{cases}
\tag{1.1}
$$

where h denotes the water height, $(u, v)^T$ is the velocity vector, b represents the bottom topography and g is the gravitational constant. Other terms, such as a friction term, could also be added in (1.1).

Due to the large scientific and engineering applications of the shallow water equations, research on effective and accurate numerical methods for their solutions has attracted great attention in the past two decades. One difficulty encountered is the treatment of the source terms. An essential part for the shallow water equations and other conservation laws with source terms is that they often admit steady state solutions in which the flux gradients are exactly balanced by the source term. For the shallow water equations, people are particularly interested in the still water steady-state solution, which represents a still flat water surface, and often referred as "lake at rest" solution:

$$
u = 0 \quad \text{and} \quad h + b = const.
\tag{1.2}
$$

Traditional numerical schemes with a straightforward handling of the source term cannot balance the effect of the source term and the flux, and usually fail to capture the steady state well. They will introduce spurious oscillations near the steady state. The well-balanced

schemes are specially designed to preserve exactly these steady-state solutions up to machine error with relatively coarse meshes, and therefore it is desirable to design numerical methods which have the well-balanced property. The other major difficulty often encountered in the simulations of the shallow water equations is the appearance of dry areas in many engineering applications including the dam break problem and flood waves etc. Special attention needs to be paid near the dry/wet front, otherwise they may produce non-physical negative water height, which becomes problematic when calculating the eigenvalues $u \pm \sqrt{gh}$ to determine the time step size Δt, and renders the system not hyperbolic and not well posed.

In the past decade, many well-balanced numerical methods have been developed for the shallow water equations, see, e.g. [3, 1, 22, 2, 26, 25] and the references therein. There are also a number of positivity-preserving schemes [4, 14, 17, 7, 9, 6] proposed for (1.1), and a few of them [21, 1, 14, 8] can resolve both difficulties at the same time. Recently, high-order accurate DG methods have attracted increasing attention in many computational fields, including the geophysical fluid dynamics. DG method is a class of finite element methods using discontinuous piecewise polynomial space as the solution and test function spaces (see [10] for a historic review). 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, see the first work by Schwaneberg and Kongeter [29], followed by [15, 13, 18, 23] and others. Recently, several well-balanced DG methods have been proposed, by Xing and Shu [33, 34, 36], Ern et al. [14], Rhebergen et al. [27] and other researchers [20]. Also, some discussions on DG methods involving wetting and drying treatments for the shallow water equations can be found in [5, 14, 9].

In [36], high order accurate DG methods, which can maintain the still water steady state exactly, and at the same time can preserve the non-negativity of the water height, are developed for the shallow water equations on one-dimensional and two-dimensional rectangular meshes. Due to the complex geometry of the computational domains in many real-world applications, triangular meshes are often used. In this paper, we are interested in the ex-

tension of the positivity-preserving well-balanced methods developed in [36] on unstructured triangular meshes. A simple source term discretization will be presented, and shown to be balanced with the numerical fluxes at the steady state solution. With the introduction of a special designed Gaussian quadrature rule, we will demonstrate that the simple positivity-preserving limiter used in [39] is still plausible on triangular meshes, and does not affect the high order accuracy, as well as the mass conservation.

This paper is organized as follows. In Section 2, we present the well-balanced DG methods for the shallow water equations on triangular meshes, following the technique proposed in [34]. The positivity-preserving well-balanced DG methods are presented in Section 3, which involves a simple positivity-preserving limiter. Section 4 contains extensive numerical simulation results to demonstrate the behavior of our DG methods for two-dimensional shallow water equations on triangular meshes, 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 5.

2   Well-balanced DG methods

In this section, we first reinstate the classical Runge-Kutta discontinuous Galerkin (RKDG) methods applied for the shallow water equations. A few well-balanced DG methods have been developed recently, see the recent book chapter [25] for a review and the references therein. In this paper, we consider the approach developed by one of the authors in [34], where we observed that the classical RKDG methods are well-balanced for the still water solution (1.2), if a hydrostatic reconstruction is employed on the flux. The same technique is also used in [14, 20, 36] to derive well-balanced positivity-preserving methods. This is one of the simplest approaches to obtain a high order well-balanced scheme, and the extra computational cost due to the well-balanced property is negligible. Its extension to a triangulation will be introduced in this section, and this scheme will serve as the basis for the positivity-preserving technique presented in Section 3.

Let $\mathcal{T}_\tau$ be a family of partitions of the computational domain Ω parameterized by $\tau > 0$. For any triangle $K \in \mathcal{T}_\tau$, we define $\tau_K := \text{diam}(K)$ and $\tau := \max\limits_{K \in \mathcal{T}_\tau} \tau_K$. For each edge e_K^i $(i = 1, 2, 3)$ of K, we denote its length by l_K^i, and outward unit normal vector by ν_K^i. Let $K(i)$ be the neighboring triangle along the edge e_K^i and $|K|$ be the area of the triangle K.

For the ease of presentation, we denote the shallow water equations (1.1) by

$$U_t + f(U)_x + g(U)_y = s(h, b), \quad \text{or} \quad U_t + \nabla \cdot \mathbf{F}(U) = s(h, b),$$

where $U = (h, hu, hv)^T$ with the superscript T denoting the transpose, $f(U)$, $g(U)$ or $\mathbf{F}(U) = (f(U), g(U))$ are 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

$$V_\tau = V_\tau^k \equiv \{w \in L^2(\Omega); \ w|_K \in P^k(K) \ \forall K \in \mathcal{T}_\tau\}, \tag{2.1}$$

where $P^k(K)$ denotes the space of polynomials on K of degree at most k. We project the bottom function b into the same space V_τ, to obtain an approximation which is still denoted by b, again with an abuse of notation. Let $\mathbf{x}$ denote (x, y), then the numerical scheme is given by

$$\iint_K \partial_t U w \, d\mathbf{x} - \iint_K \mathbf{F}(U) \cdot \nabla w \, d\mathbf{x} + \sum_{i=1}^{3} \int_{e_K^i} \widehat{\mathbf{F}}|_{e_K^i} \cdot \nu_K^i w \, ds = \iint_K s(h, b) w \, d\mathbf{x}, \tag{2.2}$$

where $w(\mathbf{x})$ is a test function from the test space V_τ. The numerical flux $\widehat{\mathbf{F}}$ is defined by

$$\widehat{\mathbf{F}}|_{e_K^i} \cdot \nu_K^i = \mathcal{F}(U_i^{int(K)}, U_i^{ext(K)}, \nu_K^i). \tag{2.3}$$

where $U_i^{int(K)}$ and $U_i^{ext(K)}$ are the approximations to the values on the edge e_K^i obtained from the interior and the exterior of K.

We could, for example, use the simple global Lax-Friedrichs flux

$$\mathcal{F}(a_1, a_2, \nu) = \frac{1}{2}\left[\mathbf{F}(a_1) \cdot \nu + \mathbf{F}(a_2) \cdot \nu - \alpha(a_2 - a_1)\right], \quad \alpha = \max\left((|u| + \sqrt{gh}, |v| + \sqrt{gh}) \cdot \nu\right), \tag{2.4}$$

where the maximum is taken over the whole region. It satisfies the conservativity and consistency

$$\mathcal{F}(a_1, a_2, \nu) = -\mathcal{F}(a_2, a_1, -\nu), \quad \mathcal{F}(a_1, a_1, \nu) = \mathbf{F}(a_1) \cdot \nu. \qquad (2.5)$$

A simple Euler forward time discretization of (2.2) gives the fully discretized scheme

$$\iint_K \frac{U^{n+1} - U^n}{\Delta t} w \, \mathrm{d}\mathbf{x} - \iint_K \mathbf{F}(U) \cdot \nabla w \, \mathrm{d}\mathbf{x} + \sum_{i=1}^{3} \int_{e_K^i} \widehat{\mathbf{F}}|_{e_K^i} \cdot \nu_K^i w \, \mathrm{d}s = \iint_K s(h, b) w \, \mathrm{d}\mathbf{x}. \quad (2.6)$$

Total variation diminishing (TVD) high order Runge-Kutta time discretization [31] 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:

$$
\begin{aligned}
U^{(1)} &= U^n + \Delta t \mathcal{L}(U^n) \qquad\qquad\qquad\qquad (2.7)\\
U^{(2)} &= \frac{3}{4} U^n + \frac{1}{4} \left(U^{(1)} + \Delta t \mathcal{L}(U^{(1)}) \right)\\
U^{n+1} &= \frac{1}{3} U^n + \frac{2}{3} \left(U^{(2)} + \Delta t \mathcal{L}(U^{(2)}) \right),
\end{aligned}
$$

where $\mathcal{L}(U)$ 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. Following the technique presented in [34], our well-balanced numerical scheme, modified from the classical version (2.6), takes the form:

$$\iint_K \frac{U^{n+1} - U^n}{\Delta t} w \, \mathrm{d}\mathbf{x} - \iint_K \mathbf{F}(U) \cdot \nabla w \, \mathrm{d}\mathbf{x} + \sum_{i=1}^{3} \int_{e_K^i} \widehat{\mathbf{F}}^*|_{e_K^i} \cdot \nu_K^i w \, \mathrm{d}s = \iint_K s(h, b) w \, \mathrm{d}\mathbf{x}, \quad (2.8)$$

or equivalently,

$$
\begin{aligned}
&\iint_K \frac{U^{n+1} - U^n}{\Delta t} w \, \mathrm{d}\mathbf{x} - \iint_K \mathbf{F}(U) \cdot \nabla w \, \mathrm{d}\mathbf{x} + \sum_{i=1}^{3} \int_{e_K^i} \widehat{\mathbf{F}}|_{e_K^i} \cdot \nu_K^i w \, \mathrm{d}s\\
&= \iint_K s(h, b) w \, \mathrm{d}\mathbf{x} + \sum_{i=1}^{3} \int_{e_K^i} (\widehat{\mathbf{F}} - \widehat{\mathbf{F}}^*)|_{e_K^i} \cdot \nu_K^i w \, \mathrm{d}s \qquad\qquad (2.9)
\end{aligned}
$$

The left side of (2.9) is the classical RKDG scheme, and the right side is our approximation to the source term. The flux $\widehat{\mathbf{F}}^*$ is computed based on the hydrostatic reconstruction technique [1] and will be explained later. However we point out here that the difference

$\widehat{\mathbf{F}} - \widehat{\mathbf{F}}^*$ is a high order correction term at the level of $O(h^{k+1})$ regardless of the smoothness of the solution U. Therefore, the scheme (2.8) is a spatially $(k+1)$-th order conservative scheme and will converge to the weak solution.

After computing boundary values $U_i^{int(K)}$ and $U_i^{ext(K)}$ on the edge e_K^i, we set

$$
\begin{aligned}
h_i^{*,int(K)} &= \max\left(0, h_i^{int(K)} + b_i^{int(K)} - \max(b_i^{int(K)}, b_i^{ext(K)})\right) \\
h_i^{*,ext(K)} &= \max\left(0, h_i^{ext(K)} + b_i^{ext(K)} - \max(b_i^{int(K)}, b_i^{ext(K)})\right)
\end{aligned}
\tag{2.10}
$$

and redefine the interior and exterior values of U as:

$$
U_i^{*,int(K)} = \begin{pmatrix} h_i^{*,int(K)} \\ h_i^{*,int(K)} u_i^{int(K)} \\ h_i^{*,int(K)} v_i^{int(K)} \end{pmatrix} = \frac{h_i^{*,int(K)}}{h_i^{int(K)}} U_i^{int(K)},
$$

$$
U_i^{*,ext(K)} = \begin{pmatrix} h_i^{*,ext(K)} \\ h_i^{*,ext(K)} u_i^{ext(K)} \\ h_i^{*,ext(K)} v_i^{ext(K)} \end{pmatrix} = \frac{h_i^{*,ext(K)}}{h_i^{ext(K)}} U_i^{ext(K)},
\tag{2.11}
$$

Introducing the notations

$$
\delta_{i,x}^* = \left(0, \frac{g}{2}(h_i^{int(K)})^2 - \frac{g}{2}(h_i^{*,int(K)})^2, 0\right)^T, \qquad \delta_{i,y}^* = \left(0, 0, \frac{g}{2}(h_i^{int(K)})^2 - \frac{g}{2}(h_i^{*,int(K)})^2\right)^T
$$

on the edge e_K^i, the flux $\widehat{\mathbf{F}}^*$ is then given by:

$$
\widehat{\mathbf{F}}^*|_{e_K^i} \cdot \nu_K^i = \mathcal{F}(U_i^{*,int(K)}, U_i^{*,ext(K)}, \nu_K^i) + \langle \delta_{i,x}^*, \delta_{i,y}^* \rangle \cdot \nu_K^i.
\tag{2.12}
$$

We also require that all the integrals in formula (2.8) 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 w are polynomials at the still water state in each triangle K, hence $\mathbf{F}(U)$ and $s(h, b)$ are both polynomials. We can prove that the above methods (2.8), combined with the choice of fluxes (2.12), are actually well-balanced for the still water steady state of the shallow water equations. The key idea is to show that, at the still water steady state (1.2), the numerical fluxes $\widehat{\mathbf{F}}^*$ becomes $\mathbf{F}(U_i^{int(K)})$ on the edge e_K^i. We refer to [34] for the technical details of the proof.

When applied to problems which contain discontinuous solution, RKDG methods may generate oscillation and even nonlinear instability. We often apply nonlinear limiters to

7

control these oscillations. Many limiters have been studied in the literature. In this paper, we use the classical characteristic-wise total variation bounded (TVB) limiter in [12, 30], with a corrected minmod function defined by

$$\overline{m}(a_1, \cdots, a_n) = \begin{cases} a_1, & \text{if} \quad |a_1| \leq M\Delta x^2, \\ m(a_1, \cdots, a_n), & \text{otherwise}, \end{cases} \tag{2.13}$$

where M is the TVB parameter to be chosen adequately [11] and the minmod function m is given by

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

Usually, the limiter is applied on the function U after each inner stage in the Runge-Kutta time stepping. 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 $h + b = const$, since if the limiter is enacted, the resulting modified solution h may no longer satisfy this steady state relation. Therefore, following the idea presented in [33, 36], we present the following strategy to perform the limiter, which works well with the well-balance property. As explained in [36], we 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. We first check if the limiting is needed, based on $(h + b, hu, hv)^T$. If a certain cell is flagged by this procedure needing limiting, then the actual TVB limiter is implemented on the variables $(h, hu, hv)^T$. Note that if in a steady state region where $h + b = const$ and $u = v = 0$, we first check if the limiting is needed based on $(h + b, hu, hv)^T = (const, 0, 0)^T$, which demonstrates that limiting is not needed in this cell. Therefore the flat surface $h + b = const$ will not be affected by the limiter procedure and the well-balanced property is maintained. Also, we observe that this procedure will not destroy the conservativity of h, which will be maintained during the limiter process. 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 it works well with the

8

positivity-preserving limiter presented in the next section.

3 Positivity-preserving limiter

In this section, we present a simple positivity-preserving limiter on triangular meshes, and couple it with the well-balanced DG methods developed for the shallow water equations in Section 2. We will start by showing the positivity of a first order scheme with the well-balanced flux, and later generalize the idea to high order schemes. For the ease of presentation, Euler forward time discretization (2.8) will be discussed, but all the results hold for the TVD high order Runge-Kutta and multi-step time discretizations.

3.1 Preliminaries

For convenience, let $\mathcal{F}_1$ and $\widehat{\mathbf{F}}^*_1|_{e^i_K} \cdot \nu^i_K$ denote the first components of $\mathcal{F}$ and $\widehat{\mathbf{F}}^*|_{e^i_K} \cdot \nu^i_K$ respectively. Then $\widehat{\mathbf{F}}^*_1|_{e^i_K} \cdot \nu^i_K = \mathcal{F}_1(U_i^{*,int(K)}, U_i^{*,ext(K)}, \nu^i_K)$ by (2.12). Taking the test function as $w \equiv 1$ in (2.8), we get the the scheme satisfied by the cell averages for the water height h:

$$\overline{h}_K^{n+1} = \overline{h}_K^n - \frac{\Delta t}{|K|} \sum_{i=1}^{3} \int_{e^i_K} \mathcal{F}_1(U_i^{*,int(K)}, U_i^{*,ext(K)}, \nu^i_K) \, \mathrm{d}s, \tag{3.1}$$

where $\overline{h}_K^n$ stands for the average of h over the triangle K at time level n.

Suppose we use L-point Gaussian quadrature for the line integral in (2.8) and (3.1), and the subscript (i, β) will denote the point value at the β-th quadrature point of the i-th edge. Let w_β denote the Gauss quadrature weight on the interval $[-1/2, 1/2]$. Then (3.1) becomes

$$\overline{h}_K^{n+1} = \overline{h}_K^n - \frac{\Delta t}{|K|} \sum_{i=1}^{3} \sum_{\beta=1}^{L} \mathcal{F}_1(U_{i,\beta}^{*,int(K)}, U_{i,\beta}^{*,ext(K)}, \nu^i_K) w_\beta l^i_K. \tag{3.2}$$

3.2 First order schemes

To investigate the positivity of a high order scheme (3.2), we need to study its first order counterpart. Given piecewise constants U_K^n for the solution and b_K for the bottom on each triangle K at time level n, consider a first order scheme for the water height,

$$h_K^{n+1} = h_K^n - \frac{\Delta t}{|K|} \sum_{i=1}^{3} \mathcal{F}_1(U_i^{*,int(K)}, U_i^{*,ext(K)}, \nu^i_K) l^i_K, \tag{3.3}$$

9

where

$$U_i^{*,int(K)} = \frac{h_K^{*,i}}{h_K^n} U_K^n, \quad U_i^{*,ext(K)} = \frac{h_{K(i)}^{*,i}}{h_{K(i)}^n} U_{K(i)}^n,$$

with

$$h_K^{*,i} = \max\left(0, h_K^n + b_K - \max(b_K, b_{K(i)})\right),$$

$$h_{K(i)}^{*,i} = \max\left(0, h_{K(i)}^n + b_{K(i)} - \max(b_K, b_{K(i)})\right),$$

Lemma 3.1: Under the CFL condition $\frac{\Delta t}{|K|}\alpha \sum\limits_{i=1}^{3} l_K^i \leq 1$, with

$$\alpha = \max\left((|u| + \sqrt{gh}, |v| + \sqrt{gh}) \cdot \nu\right), \tag{3.4}$$

if h_K^n is non-negative for any K, then h_K^{n+1} is non-negative in the first order scheme (3.3).

Proof: By (2.4), the flux in (3.3) is

$$\mathcal{F}_1(U_i^{*,int(K)}, U_i^{*,ext(K)}, \nu_K^i)$$

$$= \frac{1}{2}\left[\frac{h_K^{*,i}}{h_K^n}\langle(hu)_K^n, (hv)_K^n\rangle \cdot \nu_K^i + \frac{h_{K(i)}^{*,i}}{h_{K(i)}^n}\langle(hu)_{K(i)}^n, (hv)_{K(i)}^n\rangle \cdot \nu_K^i - \alpha(h_{K(i)}^{*,i} - h_K^{*,i}).\right]$$

And the scheme (3.3) can be written as

$$h_K^{n+1} = \left[1 - \frac{1}{2}\frac{\Delta t}{|K|}\sum_{i=1}^{3} l_K^i \frac{h_K^{*,i}}{h_K^n}\left(\alpha + \langle u_K^n, v_K^n\rangle \cdot \nu_K^i\right)\right] h_K^n$$

$$+ \frac{1}{2}\frac{\Delta t}{|K|}\sum_{i=1}^{3} l_K^i \frac{h_{K(i)}^{*,i}}{h_{K(i)}^n}\left(\alpha - \langle u_{K(i)}^n, v_{K(i)}^n\rangle \cdot \nu_K^i\right) h_{K(i)}^n \tag{3.5}$$

Notice that $h_K^{*,i}/h_K^n$, $h_{K(i)}^{*,i}/h_{K(i)}^n \in [0,1]$. And we have $|\langle u_K^n, v_K^n\rangle \cdot \nu_K^i| \leq \alpha$ for any K by (3.4). Therefore, (3.5) is a linear combination of h_K^n and $h_{K(i)}^n$ with non-negative coefficients. Thus, h_K^{n+1} is non-negative if h_K^n and $h_{K(i)}^n$ are non-negative. $\square$

3.3 High order schemes

Following the approach in [39], the first step is to decompose the cell average $\overline{h}_K^n$ as a convex combination of point values of the DG polynomial $h_K(x,y)$ by a quadrature satisfying:

- The quadrature rule is exact for integration of $h_K(x,y)$ on K.

10

- The quadrature points include all L-point Gauss quadrature points for each edge e_K^i.

- All the quadrature weights should be positive.

This particular quadrature rule can be constructed by a transformation of the tensor product of M-point Gauss-Lobatto and L-point Gauss quadrature, which is summarized below (see [39] for details).

Let $\{v^\beta : \beta = 1, \cdots, L\}$ denote the Gauss quadrature points on $[-\frac{1}{2}, \frac{1}{2}]$ with weights w_β, and $\{\widehat{u}^\alpha : \alpha = 1, \cdots, M\}$ denote the Gauss-Lobatto quadrature points on $[-\frac{1}{2}, \frac{1}{2}]$ with weights $\widehat{w}_\alpha$. In the barycentric coordinates, the set of quadrature points S_K can be written as

$$
\begin{aligned}
S_K \;=\; &\left\{ \left(\frac{1}{2} + v^\beta, (\frac{1}{2} + \widehat{u}^\alpha)(\frac{1}{2} - v^\beta), (\frac{1}{2} - \widehat{u}^\alpha)(\frac{1}{2} - v^\beta) \right), \right. \\
&\left((\frac{1}{2} - \widehat{u}^\alpha)(\frac{1}{2} - v^\beta), \frac{1}{2} + v^\beta, (\frac{1}{2} + \widehat{u}^\alpha)(\frac{1}{2} - v^\beta) \right), \\
&\left. \left((\frac{1}{2} + \widehat{u}^\alpha)(\frac{1}{2} - v^\beta), (\frac{1}{2} - \widehat{u}^\alpha)(\frac{1}{2} - v^\beta), \frac{1}{2} + v^\beta \right) : \alpha = 1, \cdots, M; \beta = 1, \cdots, L \right\}.
\end{aligned}
$$
(3.6)

In particular, for the P^2-DG method used in numerical tests of this paper, 4-point Gauss quadrature rule is needed so that the line integral in (2.8) is exactly calculated for the still water state. And the 3-point Gauss-Lobatto quadrature is sufficient to construct S_K. See Fig 3.1 for the quadrature points.

Let $h_K(x, y)$ denote the DG polynomial for the water height at time level n and $w_{\mathbf{x}}$ denote the quadrature weight for the point $\mathbf{x} \in S_K$ of the quadrature rule (3.6). Let $h_{i,\beta}^{int(K)}$ denote the point value of $h_K(\mathbf{x})$ at the β-th Gauss quadrature point of the i-th edge of K. Then the quadrature weight for $h_{i,\beta}^{int(K)}$ is $2w_\beta \widehat{w}_1 / 3$, see [39] for the detail.

The cell average $\overline{h}_K^n$ can now be written as a convex combination of point values of $h_K(\mathbf{x})$ via the quadrature rule S_K,

$$
\overline{h}_K^n = \iint_K h_K(\mathbf{x}) \, \mathrm{d}\mathbf{x} = \sum_{\mathbf{x} \in S_K} h_K(\mathbf{x}) w_{\mathbf{x}} = \sum_{i=1}^{3} \sum_{\beta=1}^{L} \frac{2}{3} w_\beta \widehat{w}_1 h_{i,\beta}^{int(K)} + \sum_{\mathbf{x} \in \overline{S}_K} h_K(\mathbf{x}) w_{\mathbf{x}}, \qquad (3.7)
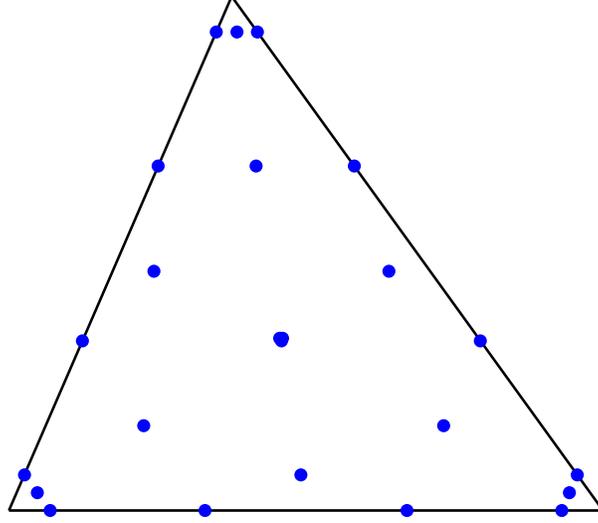$$

Figure 3.1: The quadrature points on a triangle for P^2 polynomials. There are 24 distinct points. Three points near the centroid of the triangle are very close to one another.

where $\overline{S}_K$ is the set of the points in S_K that lie in the interior of the triangle K.

Theorem 3.2: For the scheme (3.2) to be positivity preserving, i.e., $\overline{h}_K^{n+1} \geq 0$, a sufficient condition is that $h_K(\mathbf{x}) \geq 0$, $\forall \mathbf{x} \in S_K$ for all K, under the CFL condition

$$\alpha \frac{\Delta t}{|K|} \sum_{i=1}^{3} l_K^i \leq \frac{2}{3} \widehat{w}_1. \tag{3.8}$$

Here $h_K(\mathbf{x})$ denotes the polynomial for water height at time level n, $\widehat{w}_1$ is the quadrature weight of the M-point Gauss-Lobatto rule on $[-1/2, 1/2]$ for the first quadrature point. For $k = 2, 3$, $\widehat{w}_1 = 1/6$ and for $k = 4, 5$, $\widehat{w}_1 = 1/12$.

Proof: Rewrite the scheme (3.2) as

$$
\begin{aligned}
\overline{h}_K^{n+1} &= \overline{h}_K^n - \frac{\Delta t}{|K|} \sum_{i=1}^{3} \sum_{\beta=1}^{L} \mathcal{F}_1(U_{i,\beta}^{*,int(K)}, U_{i,\beta}^{*,ext(K)}, \nu_K^i) w_\beta l_K^i \\
&= \overline{h}_K^n - \frac{\Delta t}{|K|} \sum_{\beta=1}^{L} w_\beta \left(\sum_{i=1}^{3} \mathcal{F}_1(U_{i,\beta}^{*,int(K)}, U_{i,\beta}^{*,ext(K)}, \nu_K^i) l_K^i \right).
\end{aligned} \tag{3.9}
$$

Then decompose the flux term inside the bracket. Let

$$\sum_{i=1}^{3} \mathcal{F}_1(U_{i,\beta}^{*,int(K)}, U_{i,\beta}^{*,ext(K)}, \nu_K^i) l_K^i$$

$$= \mathcal{F}_1(U_{1,\beta}^{*,int(K)}, U_{1,\beta}^{*,ext(K)}, \nu_K^1) l_K^1 + \mathcal{F}_1(U_{2,\beta}^{*,int(K)}, U_{2,\beta}^{*,ext(K)}, \nu_K^2) l_K^2 + \mathcal{F}_1(U_{3,\beta}^{*,int(K)}, U_{3,\beta}^{*,ext(K)}, \nu_K^3) l_K^3$$

$$= \mathcal{F}_1(U_{1,\beta}^{*,int(K)}, U_{1,\beta}^{*,ext(K)}, \nu_K^1) l_K^1 + \mathcal{F}_1(U_{1,\beta}^{*,int(K)}, U_{2,\beta}^{*,int(K)}, -\nu_K^1) l_K^1$$

$$+ \mathcal{F}_1(U_{2,\beta}^{*,int(K)}, U_{1,\beta}^{*,int(K)}, \nu_K^1) l_K^1 + \mathcal{F}_1(U_{2,\beta}^{*,int(K)}, U_{2,\beta}^{*,ext(K)}, \nu_K^2) l_K^2 + \mathcal{F}_1(U_{2,\beta}^{*,int(K)}, U_{3,\beta}^{*,int(K)}, \nu_K^3) l_K^3$$

$$+ \mathcal{F}_1(U_{3,\beta}^{*,int(K)}, U_{2,\beta}^{*,int(K)}, -\nu_K^3) l_K^3 + \mathcal{F}_1(U_{3,\beta}^{*,int(K)}, U_{3,\beta}^{*,ext(K)}, \nu_K^3) l_K^3, \tag{3.10}$$

where we have used the conservativity of the flux (2.5).

Plugging (3.7) and (3.10) into (3.9), we get the monotone form

$$\overline{h}_K^{n+1} = \sum_{i=1}^{3} \sum_{\beta=1}^{L} \frac{2}{3} w_\beta \widehat{w}_1 h_{i,\beta}^{int(K)} + \sum_{\mathbf{x} \in \overline{S}_K} h_K(\mathbf{x}) w_\mathbf{x} - \frac{\Delta t}{|K|} \sum_{\beta=1}^{L} w_\beta \left(\sum_{i=1}^{3} \mathcal{F}_1(U_{i,\beta}^{*,int(K)}, U_{i,\beta}^{*,ext(K)}, \nu_K^i) l_K^i \right)$$

$$= \sum_{\mathbf{x} \in \overline{S}_K} h_K(\mathbf{x}) w_\mathbf{x} + \sum_{\beta=1}^{L} \frac{2}{3} w_\beta \widehat{w}_1 [H_{1,\beta} + H_{2,\beta} + H_{3,\beta}], \tag{3.11}$$

where $H_{1,\beta}, H_{2,\beta}$, and $H_{3,\beta}$ are

$$H_{1,\beta} = h_{1,\beta}^{int(K)} - \frac{3\Delta t}{2\widehat{w}_1 |K|} \left[\mathcal{F}_1(U_{1,\beta}^{*,int(K)}, U_{1,\beta}^{*,ext(K)}, \nu_K^1) l_K^1 + \mathcal{F}_1(U_{1,\beta}^{*,int(K)}, U_{2,\beta}^{*,int(K)}, -\nu_K^1) l_K^1 \right],$$

$$H_{2,\beta} = h_{2,\beta}^{int(K)} - \frac{3\Delta t}{2\widehat{w}_1 |K|} \left[\mathcal{F}_1(U_{2,\beta}^{*,int(K)}, U_{1,\beta}^{*,int(K)}, \nu_K^1) l_K^1 + \mathcal{F}_1(U_{2,\beta}^{*,int(K)}, U_{2,\beta}^{*,ext(K)}, \nu_K^2) l_K^2 \right.$$

$$\left. + \mathcal{F}_1(U_{2,\beta}^{*,int(K)}, U_{3,\beta}^{*,int(K)}, \nu_K^3) l_K^3 \right]$$

$$H_{3,\beta} = h_{3,\beta}^{int(K)} - \frac{3\Delta t}{2\widehat{w}_1 |K|} \left[\mathcal{F}_1(U_{3,\beta}^{*,int(K)}, U_{2,\beta}^{*,int(K)}, -\nu_K^3) l_K^3 + \mathcal{F}_1(U_{3,\beta}^{*,int(K)}, U_{3,\beta}^{*,ext(K)}, \nu_K^3) l_K^3 \right].$$

Following Lemma 3.1, under the CFL condition (3.8), $H_{2,\beta} \geq 0$ if $h_{i,\beta}^{int(K)}, h_{i,\beta}^{ext(K)} \geq 0$. The positivity of $H_{1,\beta}$ and $H_{3,\beta}$ follows the analysis of one-dimensional first order positivity-preserving methods presented in [36, Lemma3.1].

Therefore, if all the point values involved in (3.11), $h_{i,\beta}^{int(K)}, h_{i,\beta}^{ext(K)}$ and $h_K(\mathbf{x})$ for $\mathbf{x} \in \overline{S}_K$ are non-negative, which is equivalent to $h_K(\mathbf{x}) \geq 0, \forall \mathbf{x} \in S_K$ for all K, then we have the positivity $\overline{h}_K^{n+1} \geq 0$ in (3.11). □

Remark 3.3 As mentioned in [35, 38], for those points in $\overline{S}_K$, instead of requiring $h_K(\mathbf{x}) \geq 0, \forall \mathbf{x} \in \overline{S}_K$, it suffices to require $\sum_{\mathbf{x} \in \overline{S}_K} h_K(\mathbf{x}) w_\mathbf{x} \geq 0$ to have positivity of $\overline{h}_K^{n+1}$ in (3.11). No-

tice that $\sum_{\mathbf{x} \in \overline{S}_K} h_K(\mathbf{x}) w_\mathbf{x} / \sum_{\mathbf{x} \in \overline{S}_K} w_\mathbf{x}$ is a convex combination of point values of $h_K(\mathbf{x})$, thus by the Mean Value Theorem, there exists some point $\mathbf{x}^* \in K$ such that $h_K(\mathbf{x}^*) = \frac{1}{\sum_{\mathbf{x} \in \overline{S}_K} w_\mathbf{x}} \sum_{\mathbf{x} \in \overline{S}_K} h_K(\mathbf{x}) w_\mathbf{x}$.

By (3.7), we have

$$h_K(\mathbf{x}^*) = \frac{1}{\sum_{\mathbf{x} \in \overline{S}_K} w_\mathbf{x}} \sum_{\mathbf{x} \in \overline{S}_K} h_K(\mathbf{x}) w_\mathbf{x} = \frac{1}{1 - 2\widehat{w}_1} \left(\overline{h}_K^{n+1} - \sum_{i=1}^{3} \sum_{\beta=1}^{L} \frac{2}{3} w_\beta \widehat{w}_1 h_{i,\beta}^{int(K)} \right), \qquad (3.12)$$

where we use the fact $\sum_{\mathbf{x} \in \overline{S}_K} w_\mathbf{x} = 1 - \sum_{i=1}^{3} \sum_{\beta=1}^{L} \frac{2}{3} w_\beta \widehat{w}_1 = 1 - 2\widehat{w}_1$. So a more relaxed but less intuitive sufficient condition for (3.2) to satisfy $\overline{h}_K^{n+1} \geq 0$ is, $\overline{h}_K^{n}, h_{i,\beta}^{int(K)}, h_{i,\beta}^{ext(K)}, h_K(\mathbf{x}^*) \geq 0$ with the CFL condition (3.8), where $h_K(\mathbf{x}^*)$ is defined in (3.12).

3.4   The limiter

At time level n, given the water height DG polynomial $h_K(\mathbf{x})$ with its cell average $\overline{h}_K^{n} \geq 0$, to enforce the sufficient condition $h_K(\mathbf{x}) \geq 0$, $\forall \mathbf{x} \in S_K$, the limiter in [39] can be used directly, i.e., replacing $h_K(\mathbf{x})$ by a linear scaling around the cell average:

$$\widetilde{h}_K(\mathbf{x}) = \theta_K(h_K(\mathbf{x}) - \overline{h}_K^{n}) + \overline{h}_K^{n}, \qquad (3.13)$$

where $\theta_K \in [0, 1]$ is determined by

$$\theta_K = \min_{\mathbf{x} \in S_K} \theta_\mathbf{x}, \quad \theta_\mathbf{x} = \min \left\{ 1, \frac{\overline{h}_K^{n}}{\overline{h}_K^{n} - h_K(\mathbf{x})} \right\}. \qquad (3.14)$$

This limiter is conservative (the cell average of $\widetilde{p}_K$ is still $\overline{h}_K^{n}$), positivity-preserving ($\widetilde{h}_K(\mathbf{x}) \geq 0$, $\forall \mathbf{x} \in S_K$) and high order accurate. See [37, 39, 36] for the discussion of the limiter.

An alternative limiter is to enforce the relaxed condition in Remark 3.3. Let $\widetilde{S}_K$ denote the points in S_K which lie on the edges of K, then we can use (3.13) with

$$\theta_K = \min \left\{ \theta_{\mathbf{x}^*}, \min_{\mathbf{x} \in \widetilde{S}_K} \theta_\mathbf{x} \right\}, \quad \theta_\mathbf{x} = \min \left\{ 1, \frac{\overline{h}_K^{n}}{\overline{h}_K^{n} - h_K(\mathbf{x})} \right\}. \qquad (3.15)$$

Compared to (3.14), evaluating $h_K(\mathbf{x}), \mathbf{x} \in \overline{S}_K$ is avoided in (3.15) since $h_K(\mathbf{x}^*)$ can be obtained by (3.12), which is preferred since these point values are not Gaussian quadrature points on a triangle thus not used in the DG scheme (2.8).

14

Notice that the positivity-preserving limiter (3.13) or (3.14) will not take any effect if the DG polynomials satisfy (1.2). So the positivity-preserving limiter does not affect the well-balanced property.

3.5 The algorithm for Runge-Kutta time discretizations

At the end, we present the algorithm flowchart of our positivity-preserving well-balanced methods, when coupled with third order TVD Runge-Kutta methods.

First of all, one notices that, for Euler forward time discretization, the CFL constraint (3.8) is sufficient rather than necessary for preserving positivity. Second, for a Runge-Kutta time discretization, to enforce the CFL condition rigorously, we need to obtain an accurate estimation of (3.4) for all the stages of Runge-Kutta based only on the numerical solution at time level n, which is very difficult in most of test examples. So an efficient implementation is, if a preliminary calculation to the next time step produces negative water height, we restart the computation from the time step n with half of the time step size.

The algorithm of positivity-preserving well-balanced discontinuous Galerkin method with the third order TVD Runge-Kutta time discretization on triangular meshes can be summarized as below:

1. Given the DG polynomials $U_K(\mathbf{x})$ at time step n satisfying the cell average of h is non-negative and $h_K(\mathbf{x}) > 0, \forall \mathbf{x} \in \widetilde{S}_K$, calculate $\alpha = \max\left((|u| + \sqrt{gh}, |v| + \sqrt{gh}) \cdot \nu\right)$, where the maximum of u, v, h is taken over $\widetilde{S}_K$ and the maximum of ν is taken over ν_K^i for all K. Set the time step

$$\Delta t = \min \frac{2}{3} \frac{\widehat{w}_1 |K|}{\alpha \sum_{i=1}^3 l_K^i}.$$

2. Calculate the first stage with $U_K(\mathbf{x})$ based on (2.8) with the numerical fluxes (2.12). Let $U_K^1(\mathbf{x})$ denote the solution of the first stage. Modify it by first the TVB limiter then the positivity limiter (3.13) or (3.14) into $\widetilde{U}_K^1(\mathbf{x})$.

3. Calculate the second stage with $\widetilde{U}_K^1(\mathbf{x})$. Let $U_K^2(\mathbf{x})$ denote the solution of the second stage. If its cell average of water height is negative (by Theorem 3.2, this means that α calculated based on $U_K(\mathbf{x})$ is smaller than the one of $\widetilde{U}_K^1(\mathbf{x})$), then go back to step two and restart with half time step; otherwise, modify it by the limiters into $\widetilde{U}_K^2(\mathbf{x})$.

4. Calculate the third stage with $\widetilde{U}_K^2(\mathbf{x})$. Let $U_K^3(\mathbf{x})$ denote the solution of the third stage. If its cell average of water height is negative (by Theorem 3.2, this means that α calculated based on $U_K(\mathbf{x})$ is smaller than the one of $\widetilde{U}_K^2(\mathbf{x})$), then go back to step two and restart with half time step; otherwise, modify it by the limiters into $\widetilde{U}_K^3(\mathbf{x})$, which is the solution at time step $n+1$.

4 Numerical examples

In this section we present numerical results of our positivity-preserving well-balanced DG methods when applied to the two-dimensional shallow water equations on unstructured triangular meshes. The third order finite element DG method (i.e. $k = 2$), coupled with the third order TVD Runge-Kutta time discretization (2.7), is implemented in the examples. Global Lax-Friedrichs numerical flux is used, and the gravitation constant g is fixed as $9.812 \, m/s^2$. The time step is taken as indicated by the CFL condition (3.8). All the meshes are unstructured, and generated by EasyMesh [24].

4.1   Well-balanced test

The purpose of the first test problem is to verify the well-balanced property of our algorithm towards the steady-state solution.

We consider a rectangular computational domain $[0, 1] \times [0, 1]$. The bottom function is chosen as:

$$b(x, y) = \max\left(0, \, 1 - (10x - 5)^2 - (10y - 5)^2\right), \tag{4.1}$$

and the initial data is the stationary solution:

$$h(x, y, 0) = 2 - b(x, y), \qquad hu(x, y, 0) = hv(x, y, 0) = 0.$$

A periodic boundary condition is used. This steady state should be exactly preserved, and the surface should remain flat. We compute the solution until $t = 0.5$ on the triangular meshes with the mesh size $\tau_K = 0.025$. In order to demonstrate that the still water solution is indeed maintained up to round-off error, we use the 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 discharges hu, hv in Table 4.1. We can clearly see that all errors are at the level of round-off errors, which verifies the well-balanced property.

Table 4.1: L^1 and L^∞ errors for the stationary solution in Section 4.1.

L^1 error			L^∞ error		
h	hu	hv	h	hu	hv
4.72E-13	1.11E-12	1.13E-12	2.32E-12	2.32E-11	2.09E-11

4.2 Accuracy test

In this example we will test the high order accuracy of our schemes when applied to the following two dimensional problem. The computational domain is set as a unit square $[0, 1]^2$. The bottom topography and the initial data are given by:

$$b(x, y) = \sin^2 \left(\frac{\sqrt{2}\pi}{2}(x + y) \right), \qquad h(x, y, 0) = 5 + e^{\cos\left(\sqrt{2}\pi(x+y)\right)},$$

$$(hu)(x, y, 0) = (hv)(x, y, 0) = \frac{\sqrt{2}}{2} \sin \left(\cos \left(\sqrt{2}\pi(x + y) \right) \right),$$

which are obtained by rotating the setup of the one-dimensional accuracy test in [32, 35] by an angle of 45 degrees. Periodic boundary conditions are considered here for simplicity. The final time is set as $t = 0.05$ to avoid the appearance of shocks in the solution. Since the exact solution is also not known explicitly for this case, we use the one-dimensional well-balanced DG methods presented in [34], with a refined $12, 800$ uniform cells, to compute a reference solution. After rotating that solution by an angle of 45 degrees, this reference solution is treated as the exact solution in computing the numerical errors. The TVB constant M in the

TVB limiter (2.13) is taken as 100 here. The computational domain, and the unstructured triangular mesh with $\tau_K = 0.1, 0.025$, are shown in Figure 4.1. Table 4.2 contains the L^1 errors and orders of accuracy for the cell averages. We can clearly see that, in this two dimensional test case, third order accuracy is achieved for the RKDG scheme.
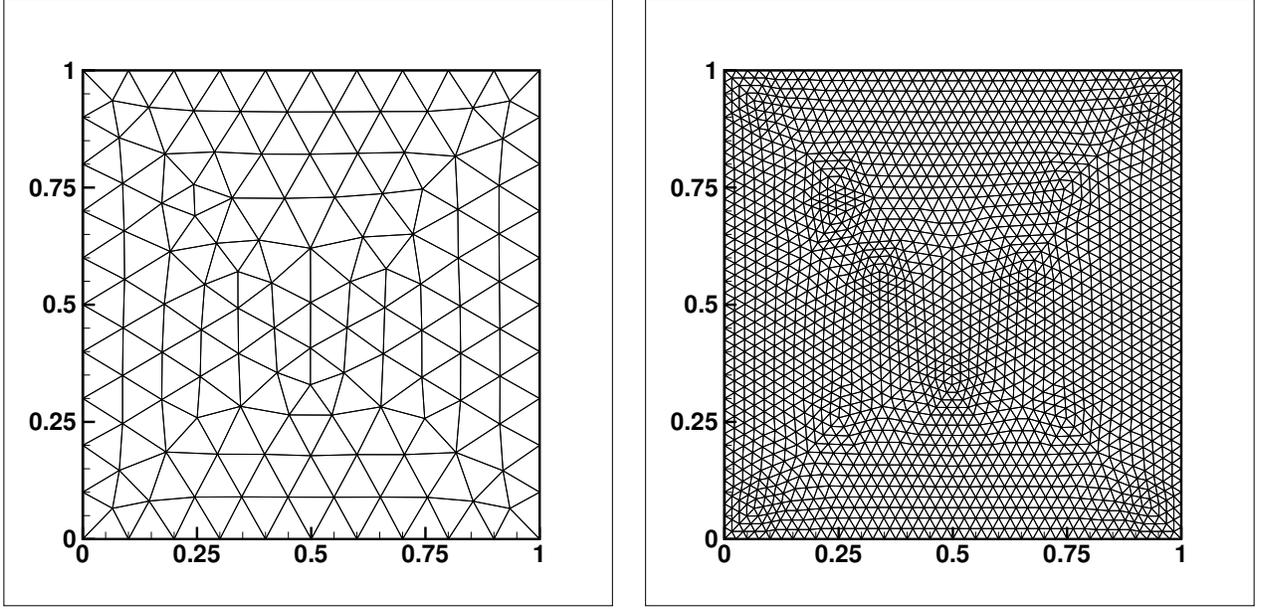

Figure 4.1: The computational domain and unstructured triangular meshes of the accuracy test problem in Section 4.2. Left: with mesh size $\tau_K = 0.1$; Right: with mesh size $\tau_K = 0.025$.

Table 4.2: FV scheme: L^1 errors and numerical orders of accuracy for the example in Section 4.2.

Meshsize	h		hu		hv	
τ_K	L^1 error	order	L^1 error	order	L^1 error	order
0.2	3.38E-03		2.84E-02		2.85E-02	
0.1	5.49E-04	2.61	4.23E-03	2.75	4.24E-03	2.75
5E-02	6.77E-05	3.02	5.65E-04	2.90	5.64E-04	2.91
2.5E-02	9.32E-06	2.86	6.87E-05	3.03	6.87E-05	3.03
1.25E-02	1.03E-06	3.17	1.15E-05	2.58	1.15E-05	2.58
6.25E-03	1.27E-07	3.02	1.85E-06	2.63	1.85E-06	2.63

4.3 A small perturbation of a steady-state solution

This is a classical example to show the capability of the proposed scheme for the perturbation of the stationary state. This test was given by LeVeque [22], and has also been used in [32, 34, 8].

We solve the system in the rectangular domain $[0, 2] \times [0, 1]$. The bottom topography is an isolated elliptical shaped hump:

$$b(x, y) = 0.8 \exp(-5(x - 0.9)^2 - 50(y - 0.5)^2). \tag{4.2}$$

The initial condition is given by:

$$h(x, y, 0) = \begin{cases} 1 - b(x, y) + 0.0001, & \text{if } 0.05 \leq x \leq 0.15, \\ 1 - b(x, y), & \text{otherwise,} \end{cases} \tag{4.3}$$
$$hu(x, y, 0) = hv(x, y, 0) = 0.$$

Initially, the surface is almost flat except for $0.05 \leq x \leq 0.15$, where h is perturbed upward by a small magnitude of 0.0001. Theoretically, this disturbance should split into two waves, propagating left and right at the characteristic speeds $\pm\sqrt{gh}$. We use the outlet boundary condition on the left and right, and reflection boundary conditions on the top and bottom sides. TVB constant M is taken as 10 in the test. Figure 4.2, left, displays the right-going disturbance as it propagates past the hump on the triangular meshes with $\tau_K = 0.00625$. The surface level $h + b$ is presented at different times. The results indicate that our schemes can resolve the complex small features of the flow very well. As a comparison, we refer to [8, Fig 5] for the output if a non-well-balanced method is used.

Next, we increase the height of the bottom topography to reach the water surface. The modified bottom topography takes the form:

$$b(x, y) = \exp(-5(x - 0.9)^2 - 50(y - 0.5)^2). \tag{4.4}$$

The other set up remains the same. We repeat the simulation and plot the surface level at different times in Figure 4.2, right. As one can see, the general structure of the solution is

well resolved. For small time when the right-going wave does not reach the bottom hump, the surface stays the same for these two simulations with different bottom topography. Obvious difference can be observed as the wave reaches and passes the hump.

4.4 Circular dam-break problem

This is a classical test case for testing the complete break of a circular dam separating a basin of water and dry bed. It has been previously tested in [6, 16].

We consider a square computatinal domain $[-100, 100] \times [-100, 100]$ with a flat bottom topography (i.e. $b = 0$). The dam is located at $r = \sqrt{x^2 + y^2} = 60$, and the water height h is initially set as 10 inside the dam and 0 outside. Both components of the velocity u and v are set to zero initially. At time $t = 0$, the circular wall forming the dam collapses. We discretize the domain with the triangular meshes and τ_K is set as 1. A 3D view and contour lines of the water height at time $t = 1.75$ are shown in Figure 4.3. We can observe an almost perfectly symmetric solution, and there is no oscillation ($\max(h) = 10$) in the numerical results.

4.5 Water drop problem

Next, we apply our methods to a numerical test case which simulates the water drop problem. Following the setup in [28], we consider the 2D Gaussian shaped peak initial condition given by:

$$
\begin{aligned}
h(x, y, 0) &= 1 + 0.1 \exp\left(-100((x - 0.5)^2 + (y - 0.5)^2)\right), \quad (4.5) \\
hu(x, y, 0) &= hv(x, y, 0) = 0,
\end{aligned}
$$

in the computational domain $[0, 1]^2$. The reflective boundary conditions are employed. The initial Gaussian shaped water drop generates a wave that reflects off the boundary. We have provided the evolution of water surface at various times in Figure 4.4, which shows that the wave is well simulated by our methods. As a comparison, we also repeat the test with a
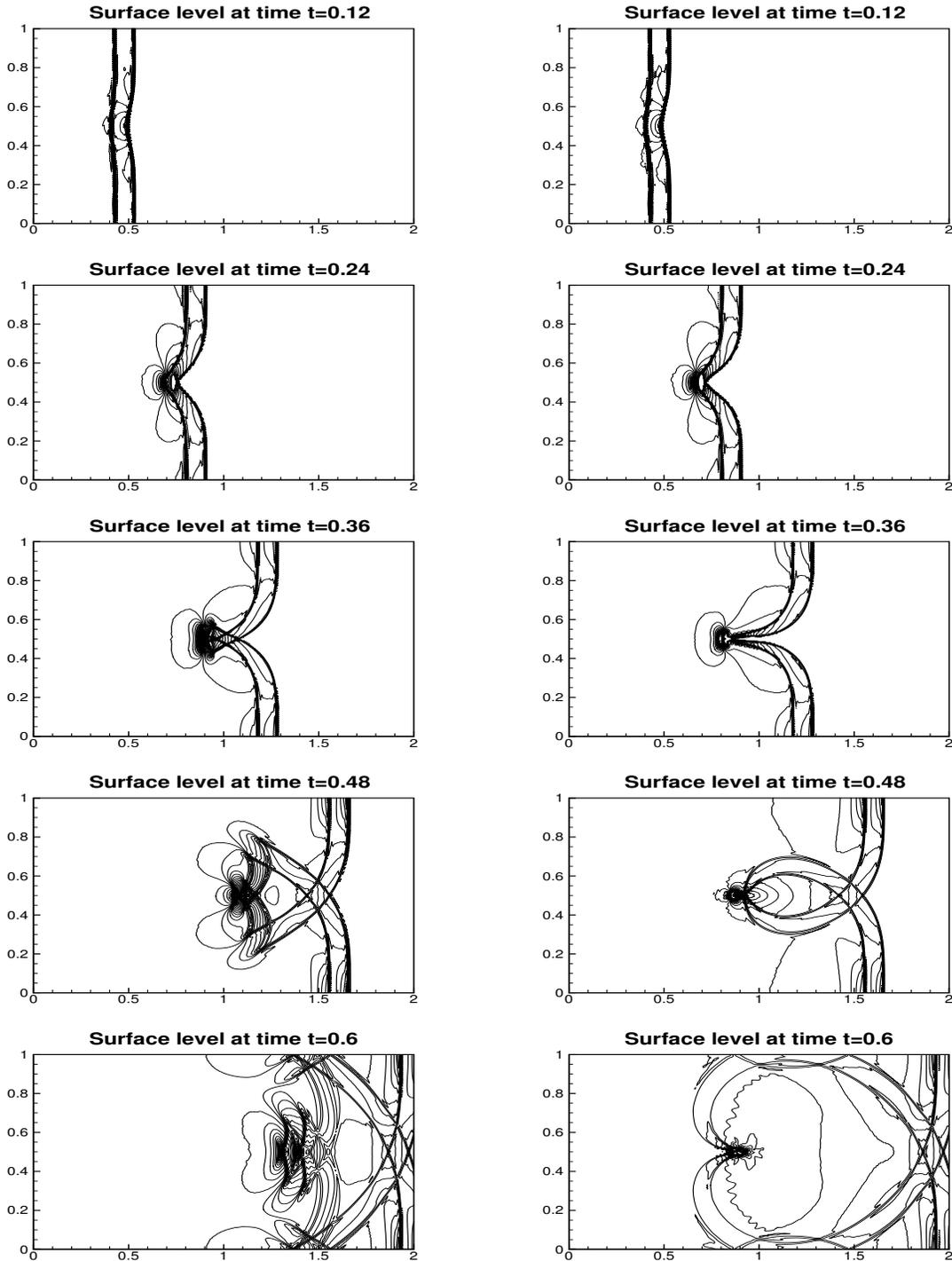
Figure 4.2: The contours of the surface level $h+b$ for the problem in Section 4.3. 40 uniformly spaced contour lines. From top to bottom: at time $t = 0.12$ from 0.999993 to 1.00007; at time $t = 0.24$ from 0.9999 to 1.0001; at time $t = 0.36$ from 0.9998 to 1.0002; at time $t = 0.48$ from 0.99985 to 1.00005; and at time $t = 0.6$ from 0.9999 to 1.0001. Left: results with the bottom (4.2). Right: results with the bottom (4.4).
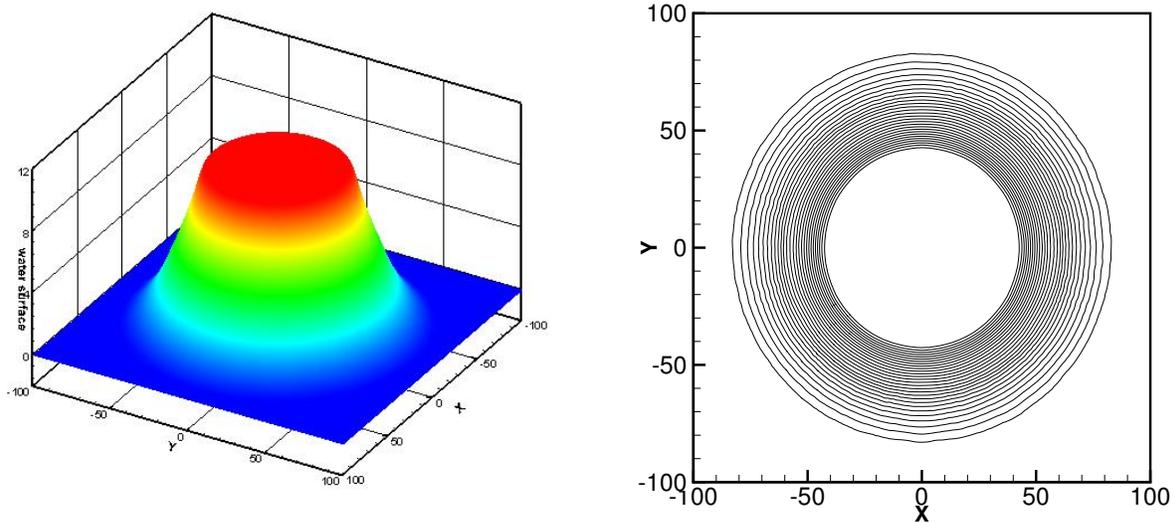
Figure 4.3: Numerical results at time $t = 1.75$ of the circular dam-break problem in Section 4.4. Left: 3D view of the surface level; Right: the contours of the surface level with 30 uniformly spaced contour lines between 0 and 10.

non-zero bottom topography:

$$b(x, y) = 0.5 \exp\left(-10((x - 0.75)^2 + (y - 0.5)^2)\right). \tag{4.6}$$

The results are shown in Figure 4.5, where we can observe the effect of the bottom on the propagation of the wave.

4.6 Flooding on a channel with three mounds

In this test example, we consider the simulation of a flow through a channel which contains three mounds on its bottom [7, 17]. The length of the channel is 75 and width is 30. The bottom topography takes the form of

$$b(x, y) = \max(0, m_1(x, y), m_2(x, y), m_3(x, y)), \tag{4.7}$$

where

$$
\begin{aligned}
m_1(x, y) &= 1 - 0.1\sqrt{(x - 30)^2 + (y - 22.5)^2}, \\
m_2(x, y) &= 1 - 0.1\sqrt{(x - 30)^2 + (y - 7.5)^2}, \\
m_1(x, y) &= 12.8 - 0.28\sqrt{(x - 47.5)^2 + (y - 15)^2}.
\end{aligned}
$$

22
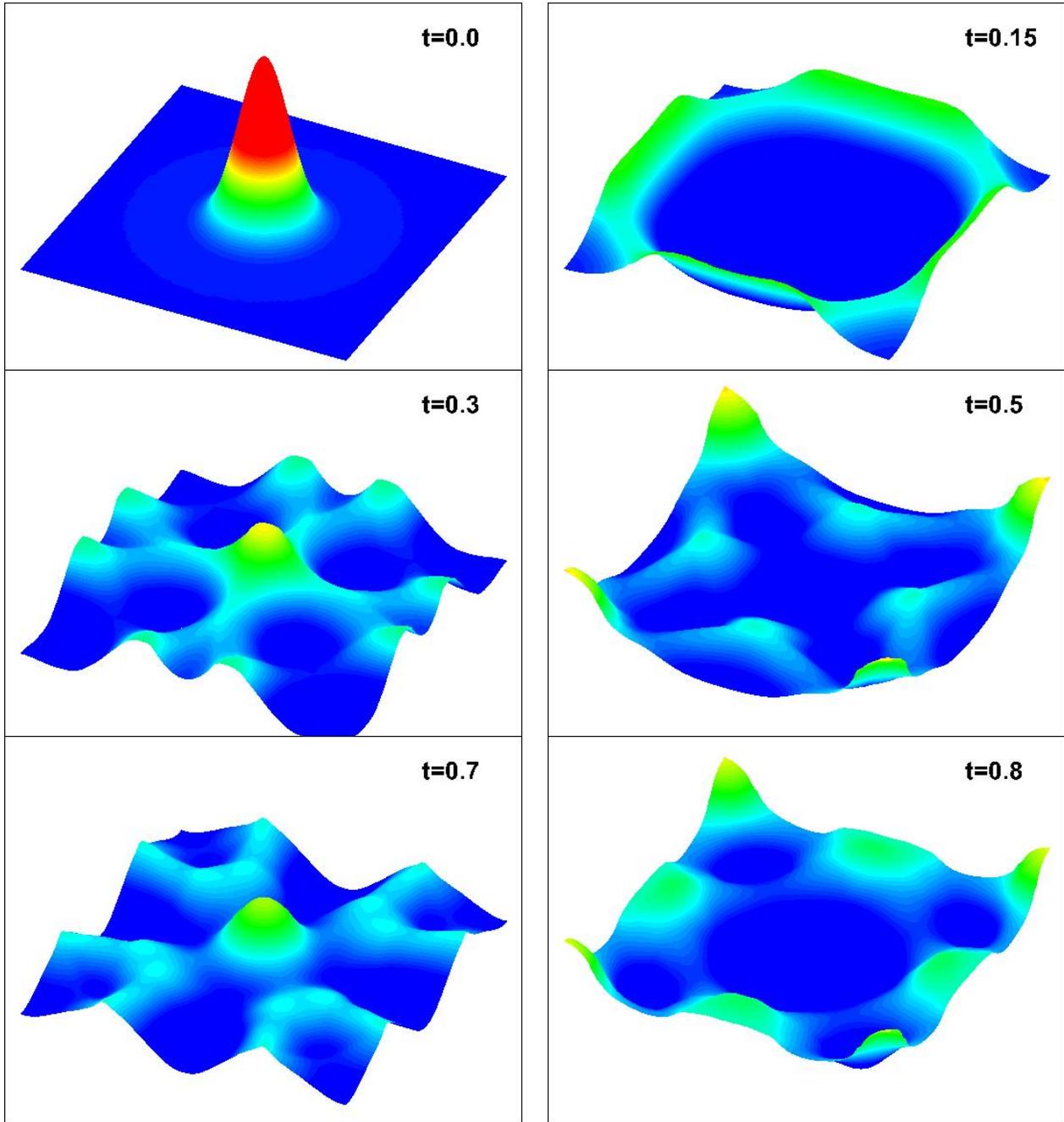
Figure 4.4: The water surface level in the water drop problem with the flat bottom topography at different times.
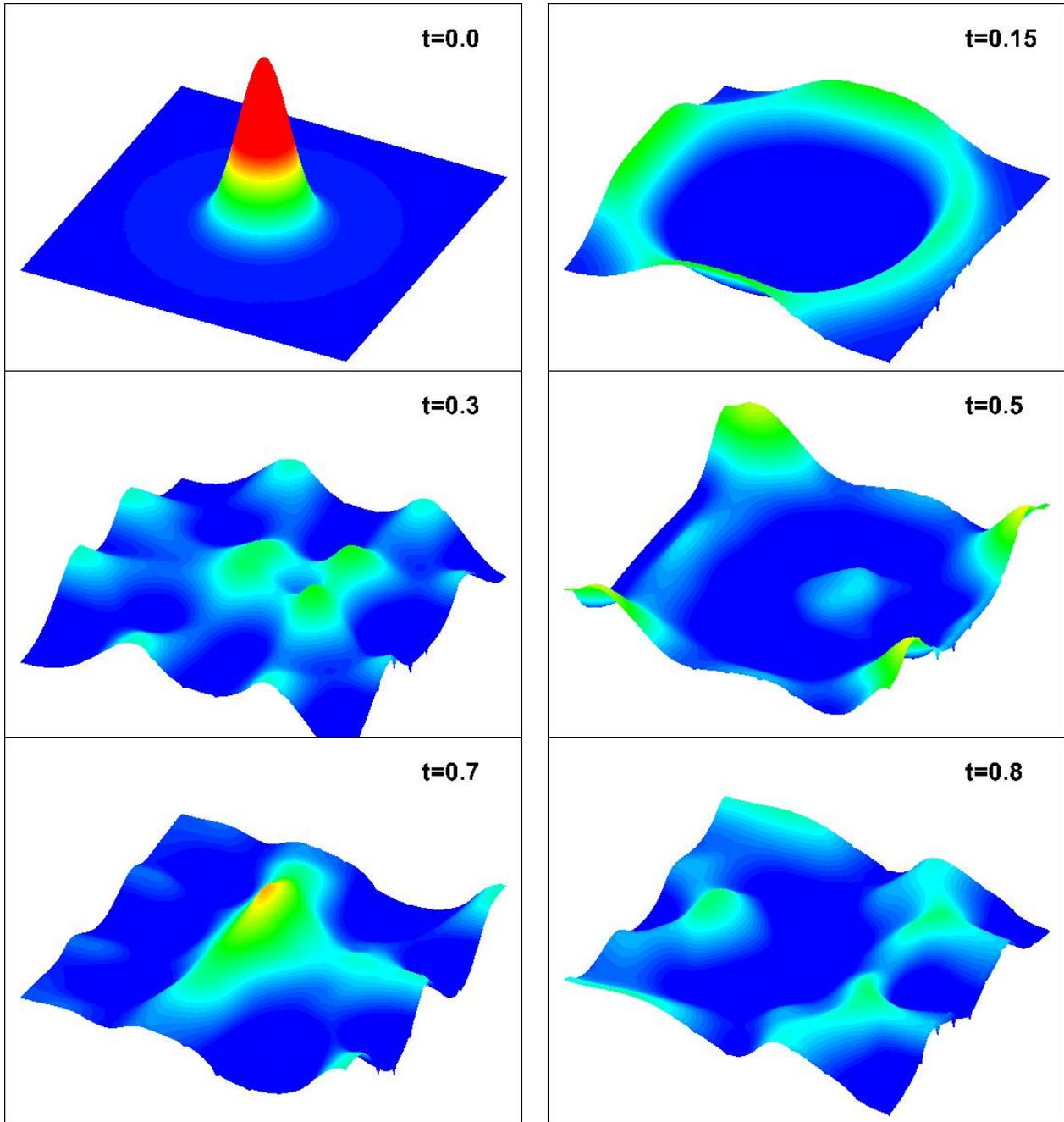
Figure 4.5: The water surface level in the water drop problem with the bottom topography (4.6) at different times.

Initially, the domain is set as dry, i.e. $h = hu = hv = 0$. We impose the reflecting boundary condition on the upper and lower boundaries in y-direction. The right x-boundary is a open-wall outflow boundary. At the left x-boundary, we impose an inflow of the form: $u = 1$, $v = 0$ and the water height $h = 0.5$ for the time $t \leq 300$, $h = 1$ when $t \geq 300$. We test our well-balanced positivity-preserving methods on this problem with triangulation of mesh size $\tau_K = 0.5$. The numerical results obtained at different times are shown in Figure 4.6.

4.7 Flows in converging-diverging channels

In the last example, we consider the water flow in an open converging-diverging channel. The test is first discussed in [19] and recently used in [8]. The gravitation constant g is taken as 1 in this test.

The computational domain is defined on the converging-diverging channel of length 3 with a half-cosine constriction centered at $x = 1.5$. It takes the form of $[0, 3] \times [-y_b(x), y_b(x)]$, where

$$y_b(x) = \begin{cases} 0.5 - 0.5(1 - d)\cos^2(\pi(x - 1.5)), & \text{if } |x - 1.5| \leq 0.5, \\ 0.5, & \text{otherwise,} \end{cases} \tag{4.8}$$

and d is the minimum channel breadth. Two values of the channel breadth d, 0.9 and 0.6, are tested in this example. The computational domain with $d = 0.6$ is shown in Figure 4.7, left.

We consider a bottom topography which consists of two elliptic Gausssian mounds:

$$b(x, y) = B\left(\exp(-10(x - 1.9)^2 - 50(y - 0.2)^2) + \exp(-20(x - 2.2)^2 - 50(y + 0.2)^2)\right), \tag{4.9}$$

where B will be specified later. This topography with $B = 1$ is shown in Figure 4.7, right. The initial conditions are given by:

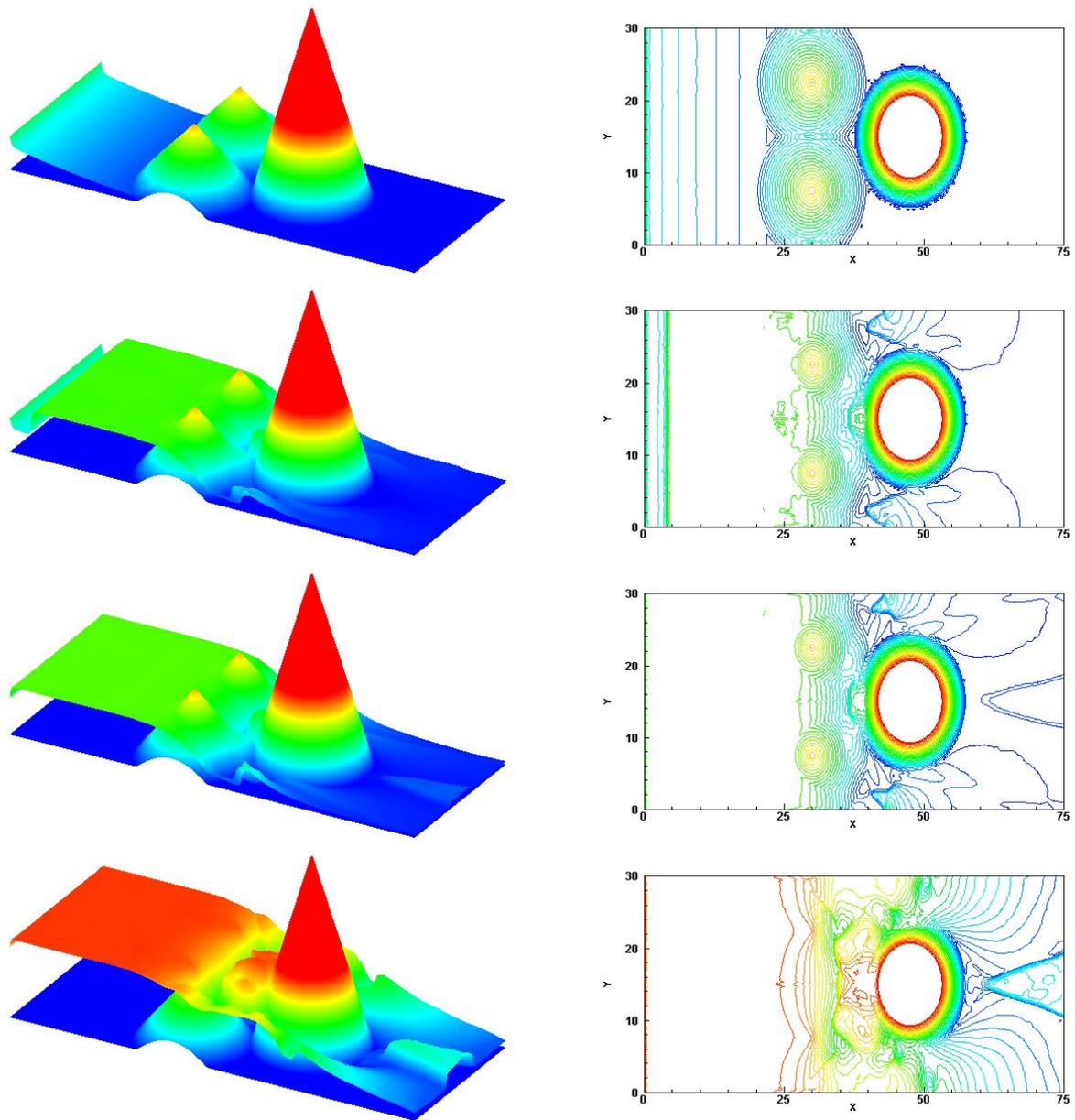$$h(x, y, 0) = \max(0, 1 - b(x, y)), \qquad hu(x, y, 0) = 2, \qquad hv(x, y, 0) = 0.$$

Figure 4.6: The water surface level in the flooding problem on a channel with three mounds at different times $t = 8, 30, 300$ and 540 (from top to bottom). Left: 3D view, color spanning between 0 and 1.2; Right: the 2D contours with 30 uniformly spaced contour lines between 0 and 1.2.

We impose the reflecting boundary condition on the upper and lower boundaries in y-direction. The left x-boundary is set as an inflow boundary with $u = 2$ and the right x-boundary is a zeroth-order outflow boundary.
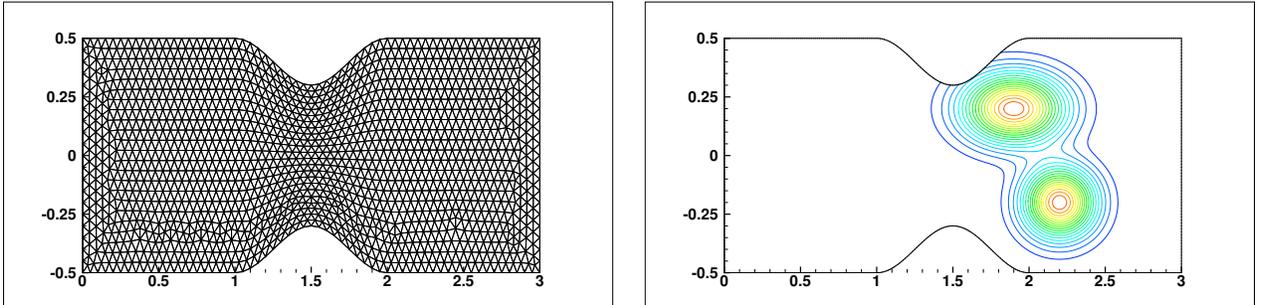

Figure 4.7: Initial setup of the converging-diverging channels problem in Section 4.7. Left: the computational domain with $d = 0.6$ and the unstructured triangular meshes with $\tau_K = 0.025$; Right: the contours of the bottom topography (4.9) with $B = 1$ and $d = 0.6$.

We first generate the triangular meshes on the rectangle domain $[0, 3] \times [-0.5, 0.5]$. Following the idea in [8], the triangulation on the converging-diverging channel is obtained through the mapping

$$(x, y) \rightarrow \begin{cases} (x, y - (1 - d)\cos^2(\pi(x - 1.5))y), & \text{if } |x - 1.5| \leq 0.5, \\ (x, y), & \text{otherwise.} \end{cases}$$

The resulting unstructured triangular mesh with $\tau_K = 0.025$ is shown in Figure 4.7, left.

Our well-balanced positivity-preserving methods are tested on this problem. In all tests, the simulations are carried out on two triangulation with mesh sizes $\tau_K = 0.025$ and $6.25E-03$ respectively. The stopping time is set as $t = 2$. We start with the parameters $d = 0.9$ and $B = 0$, i.e., a flat bottom. The numerical results are shown in Figure 4.8, which agree well with the solutions in [19, 8].

Next, we keep the computational domain ($d = 0.9$) and set the bottom topography as $B = 1$ in (4.8). The two mounds are set to reach the water surface. We repeat our simulations and show the results in Figure 4.9. They are in good agreement with the results shown in [8], and we can conclude that our well-balanced methods capture the complicated solution well.

We also modify the width of the channel by setting $d = 0.6$. The bottom topography is kept as $B = 1$. The numerical results are shown in Figure 4.10, which also agree well with the results in [8]. At the end, we increase the height of the mound to $B = 2$, to simulate the flows through two islands. Our positivity-preserving methods demonstrate to be robust and provide nice results, presented in Figure 4.11.
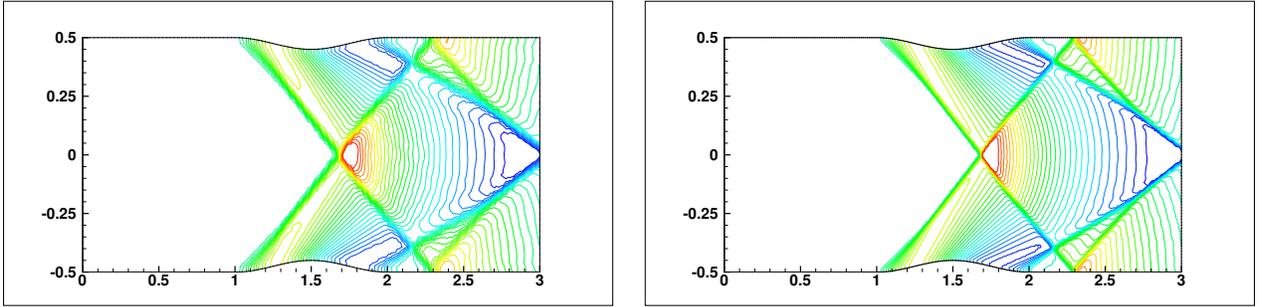

Figure 4.8: The contours of the water surface level of the converging-diverging channels problem in Section 4.7 with the parameters $d = 0.9$ and $B = 0$. Results are based on the triangulation with mesh sizes $\tau_K = 0.025$ (left) and $6.25E - 03$ (right), respectively.
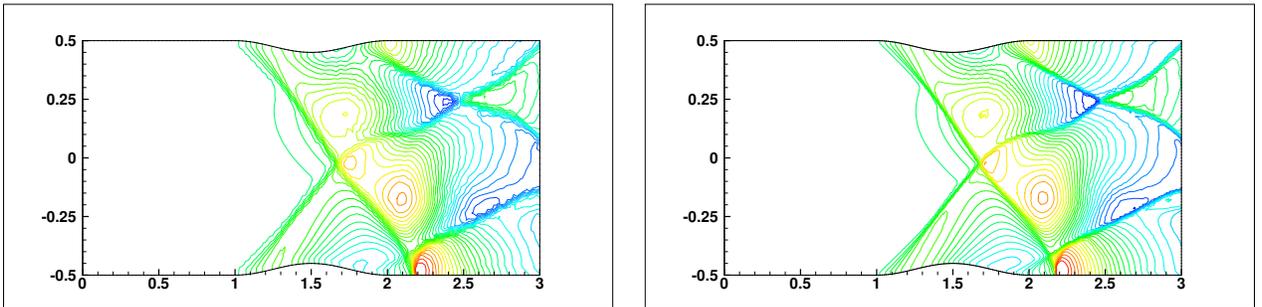

Figure 4.9: The contours of the water surface level of the converging-diverging channels problem in Section 4.7 with the parameters $d = 0.9$ and $B = 1$. Results are based on the triangulation with mesh sizes $\tau_K = 0.025$ (left) and $6.25E - 03$ (right), respectively.

5   Concluding remarks

Positivity-preserving well-balanced discontinuous Galerkin methods have been presented in [36] for the shallow water equations on one-dimensional and two dimensional problems with rectangular meshes. In this paper, we showed that such methods can be naturally extended to unstructured triangular meshes, with the introduction of a special quadrature rule from [39]. We have demonstrated that this positivity-preserving limiter can keep the water height
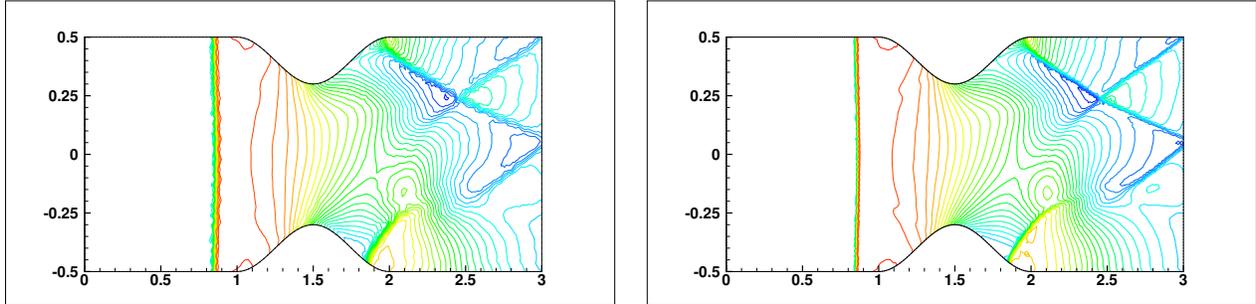
28

Figure 4.10: The contours of the water surface level of the converging-diverging channels problem in Section 4.7 with the parameters $d = 0.6$ and $B = 1$. Results are based on the triangulation with mesh sizes $\tau_K = 0.025$ (left) and $6.25E - 03$ (right), respectively.
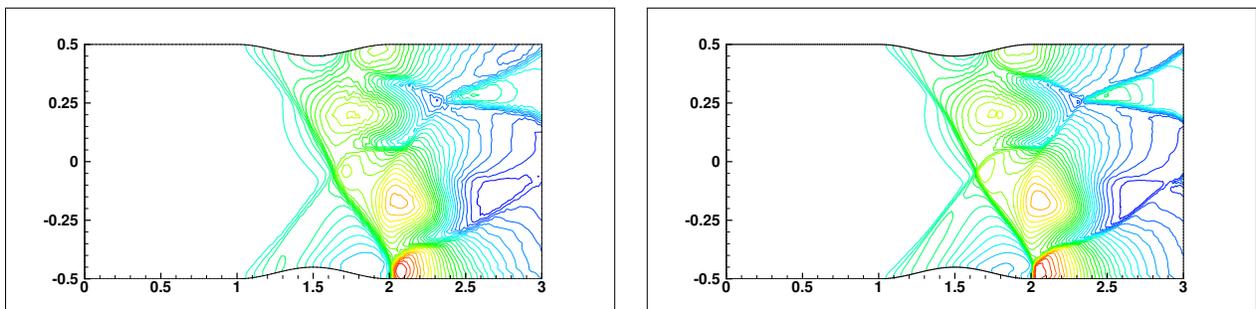

Figure 4.11: The contours of the water surface level of the converging-diverging channels problem in Section 4.7 with the parameters $d = 0.9$ and $B = 2$. Results are based on the triangulation with mesh sizes $\tau_K = 0.025$ (left) and $6.25E - 03$ (right), respectively.

non-negative under suitable CFL condition, can preserve the mass conservation, is easy to implement, and at the same time does not affect the high order accuracy for the general solutions. Extensive numerical examples are provided at the end to demonstrate the well-balanced property, accuracy, positivity-preserving property, and non-oscillatory shock resolution of the proposed numerical methods. The proposed methods are highly parallelizable and our ongoing work is to investigate their performance on high performance computers.

References

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

[2] D. S. Bale, R. J. LeVeque, S. Mitran, and J. A. Rossmanith. A wave propagation method for conservation laws and balance laws with spatially varying flux functions. *SIAM Journal on Scientific Computing*, 24:955–978, 2002.

[3] A. Bermudez and M. E. Vazquez. Upwind methods for hyperbolic conservation laws with source terms. *Computers and Fluids*, 23:1049–1071, 1994.

[4] C. Berthon and F. Marche. A positive preserving high order VFRoe scheme for shallow water equations: A class of relaxation schemes. *SIAM Journal on Scientific Computing*, 30:2587–2612, 2008.

[5] O. Bokhove. Flooding and drying in discontinuous Galerkin finite-element discretizations of shallow-water equations. Part 1: one dimension. *Journal of Scientific Computing*, 22:47–82, 2005.

[6] A. Bollermann, S. Noelle, and M. Lukácová-Medviová. Finite volume evolution Galerkin methods for the shallow water equations with dry beds. *Communications in Computational Physics*, 10:371–404, 2010.

[7] P. Brufau, M. E. Vázquez-Cendón, and P. García-Navarro. A numerical model for the flooding and drying of irregular domains. *Internation Jounral for Numerical Methods in Fluids*, 39:247–275, 2002.

[8] S. Bryson, Y. Epshteyn, A. Kurganov, and G. Petrova. Well-balanced positivity preserving central-upwind shceme on triangular grids for the Saint-Venant system. *ESAIM: Mathematical Modelling and Numerical Analysis*, 45:423–446, 2011.

[9] S. Bunya, E. J. Kubatko, J. J. Westerink, and C. Dawson. A wetting and drying treatment for the Runge-Kutta discontinuous Galerkin solution to the shallow water equations. *Computer Methods in Applied Mechanics and Engineering*, 198:1548–1562, 2009.

[10] B. Cockburn, G. Karniadakis, and C.-W. Shu. The development of discontinuous galerkin methods. In B. Cockburn, G. Karniadakis, and C.-W. Shu, editors, *Discontinuous Galerkin Methods: Theory, Computation and Applications*, pages 3–50. Lecture Notes in Computational Science and Engineering, Part I: Overview, volume 11, Springer, 2000.

[11] B. Cockburn and C.-W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework. *Mathematics of Computation*, 52:411–435, 1989.

[12] B. Cockburn and C.-W. Shu. The Runge-Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems. *Journal of Computational Physics*, 141:199–224, 1998.

[13] C. Dawson and J. Proft. Discontinuous and coupled continuous/discontinuous Galerkin methods for the shallow water equations. *Computer Methods in Applied Mechanics and Engineering*, 191:4721–4746, 2002.

[14] A. Ern, S. Piperno, and K. Djadel. A well-balanced Runge-Kutta discontinuous Galerkin method for the shallow-water equations with flooding and drying. *International Journal for Numerical Methods in Fluids*, 58:1–25, 2008.

[15] C. Eskilsson and S. J. Sherwin. A triangular spectral/hp discontinuous Galerkin method for modelling 2D shallow water equations. *International Journal for Numerical Methods in Fluids*, 45:605–623, 2004.

[16] J.-H. Feng, L. Cai, and W.-X. Xie. CWENO-type central-upwind schemes for multidimensional Saint-Venant system of shallow water equations. *Applied numerical mathematics*, 56:1001–1017, 2006.

[17] J. M. Gallardo, C. Parés, and M. Castro. On a well-balanced high-order finite volume scheme for shallow water equations with topography and dry areas. *Journal of Computational Physics*, 227:574–601, 2007.

[18] F. X. Giraldo, J. S. Hesthaven, and T. Warburton. Nodal high-order discontinuous Galerkin methods for the spherical shallow water equations. *Journal of Computational Physics*, 181:499–525, 2002.

[19] M.E. Hubbard. On the accuracy of one-dimensional models of steady converging/diverging open channel flows. *International Journal for Numerical Methods in Fluids*, 35:785–808, 2001.

[20] G. Kesserwani and Q. Liang. Well-balanced RKDG2 solutions to the shallow water equations over irregular domains with wetting and drying. *Computers & Fluids*, 39:2040–2050, 2010.

[21] A. Kurganov and D. Levy. Central-upwind schemes for the Saint-Venant system. *Mathematical Modelling and Numerical Analysis*, 36:397–425, 2002.

[22] R. J. LeVeque. Balancing source terms and flux gradients on high-resolution Godunov methods: the quasi-steady wave-propagation algorithm. *Journal of Computational Physics*, 146:346–365, 1998.

[23] R. D. Nair, S. J. Thomas, and R. D. Loft. A discontinuous Galerkin global shallow water model. *Monthly Weather Review*, 133:876–888, 2005.

[24] B. Niceno. EasyMesh version 1.4: a two-dimensional quality mesh generator. available from: http://www-dinma.univ.trieste.it/nirftc/research/easymesh/, 2001.

[25] S. Noelle, Y. Xing, and C.-W. Shu. High-order well-balanced schemes. In G. Russo and G. Puppo, editors, *Numerical Methods for Relaxation Systems and Balance Equations*. Quaderni di Matematica, Dipartimento di Matematica, Seconda Universita di Napoli, Italy, 2009.

[26] B. Perthame and C. Simeoni. A kinetic scheme for the Saint-Venant system with a source term. *Calcolo*, 38:201–231, 2001.

[27] S. Rhebergen, O. Bokhove, and J.J.W. van der Vegt. Discontinuous galerkin finite element methods for hyperbolic nonconservative partial differential equations. *Journal of Computational Physics*, 227:1887–1922, 2008.

[28] O. San and K. Kara. High-order acurate spectral difference method for shallow water equations. *International Journal of Research and Reviews in Applied Sciences*, 6:41–54, 2011.

[29] D. Schwanenberg and J. Köngeter. A discontinuous Galerkin method for the shallow water equations with source terms. In B. Cockburn, G. Karniadakis, and C.-W. Shu, editors, *Discontinuous Galerkin Methods: Theory, Computation and Applications*, pages 289–309. Lecture Notes in Computational Science and Engineering, Part I: Overview, volume 11, Springer, 2000.

[30] C.-W. Shu. Tvb uniformly high-order schemes for conservation laws. *Mathematics of Computation*, 49:105–121, 1987.

[31] C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of Computational Physics*, 77:439–471, 1988.

[32] Y. Xing and C.-W. Shu. High order finite difference WENO schemes with the exact conservation property for the shallow water equations. *Journal of Computational Physics*, 208:206–227, 2005.

[33] Y. Xing and C.-W. Shu. High order well-balanced finite volume WENO schemes and discontinuous Galerkin methods for a class of hyperbolic systems with source terms. *Journal of Computational Physics*, 214:567–598, 2006.

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

[35] Y. Xing and C.-W. Shu. High-order finite volume WENO schemes for the shallow water equations with dry states. *Advances in Water Resources*, 34:1026–1038, 2011.

[36] Y. Xing, X. Zhang, and C.-W. Shu. Positivity-preserving high order well-balanced discontinuous Galerkin methods for the shallow water equations. *Advances in Water Resources*, 33:1476–1493, 2010.

[37] X. Zhang and C.-W. Shu. On maximum-principle-satisfying high order schemes for scalar conservation laws. *Journal of Computational Physics*, 229:3091–3120, 2010.

[38] X. Zhang and C.-W. Shu. Maximum-principle-satisfying and positivity-preserving high order schemes for conservation laws: Survey and new developments. *Proceedings of the Royal Society A*, 467:2752–2776, 2011.

[39] X. Zhang, Y. Xia, and C.-W. Shu. Maximum-principle-satisfying and positivity-preserving high order discontinuous galerkin schemes for conservation laws on triangular meshes. *Journal of Scientific Computing*, 50:29–62, 2012.