A new well-balanced non-oscillatory central scheme for the shallow water equations on rectangular meshes

M. T. Capilla[a,*], A. Balaguer-Beser[a]

[a] *Department of Applied Mathematics, Universidad Politécnica de Valencia Valencia, Spain*

Abstract

This paper is concerned with the development of high-order well-balanced central schemes to solve the shallow water equations in two spatial dimensions. A Runge-Kutta scheme with a natural continuous extension is applied for time discretization. A Gaussian quadrature rule is used to evaluate time integrals and a three-degree polynomial which calculates point-values from cell averages or flux values by avoiding the increase in the number of solution extrema at the interior of each cell is used as reconstruction operator. That polynomial also guarantees that the number of extrema does not exceed the initial number of extrema and thus it avoids spurious numerical oscillations in the computed solution. A new procedure has been defined to evaluate the flux integrals and to approach the 2D source term integrals in order to verify the exact C-property, using the water surface elevation instead of the water depth as a variable. Numerical experiments have confirmed the high-resolution properties of our numerical scheme in 2D test problems. The well-balanced property of the resulting scheme has also been investigated.

Keywords: Central scheme, well-balanced, C-property, non-oscillatory, shallow water equations.
2000 MSC: 120600

1. Introduction

The shallow water equations with a non-flat bottom topography are used to model flows in rivers and coastal areas. They have wide applications in ocean and hydraulic engineering. For example, tidal flows in estuary and coastal water region, bore wave propagation, river flows, ... (see [25]). These equations have still water steady state solutions in which the flux gradients are balanced by the source term. It is desirable to develop numerical methods which preserve exactly these steady state solutions. The shallow water equations have initially been approached in the framework of upwind schemes. In this context, Bermúdez and Vázquez [3] proposed the idea of the exact C-property which means that the scheme is exact when applied to the stationary case. After the pioneer work of Greenberg and Leroux [11], schemes that preserve the lake at rest are called well-balanced. During the last years, several schemes have been proposed according to the well-balanced strategy (see [24], [4], [7], [8]). In [9], a high-order well-balanced finite volume upwind scheme was developed within a general nonconservative framework. It is a challenge to obtain

*Corresponding author
Email addresses: tcapilla@mat.upv.es (M. T. Capilla), abalague@mat.upv.es (A. Balaguer-Beser)

schemes that compute solutions with high-order accuracy (both in space and time) in the regions where they are smooth, while at the same time shock discontinuities are properly captured (see [10]).

The weighted essential non-oscillatory (WENO) schemes described in [12] uses the idea of adaptive stencils in the reconstruction procedure based on the local smoothness of the numerical solution to automatically achieve high order accuracy and a non-oscillatory property near discontinuities. The WENO method has been developed in recent years as a class of high order methods for the shallow water equations (see [23] or [16]), which gives non-oscillatory discontinuity transitions and high order accurate resolutions for the smooth part of the solution. In [16] the performance of the upwind WENO schemes has been investigated, with the objective of obtaining better performance for the shallow water equations by choosing suitable numerical fluxes. The main advantage of that upwind schemes is its strong physical basis, which allows to obtain good resolution and the maximum reduction of unphysical oscillations near shocks (see [21]). On the other hand, the required knowledge of the eigensystem is a strong limit to the versatility of the upwind finite volume schemes [7].

The alternative approach is the central schemes whose origin is the second-order Nessyahu-Tadmor staggered method [18]. Central schemes of Bianco et al. [5], Levy et al. [14], Pareschi et al. [19] or Balaguer-Beser [2] are particularly useful when the eigensystem of the equations is very complex or even unknown. Balaguer and Conde [1] compare the accuracy of upwind and central schemes to solve scalar conservation laws. They prove that upwind and central schemes have a similar order of accuracy when both numerical schemes use the same reconstruction operator. High-order well-balanced central WENO schemes have also been described for the shallow water equations by Caleffi et al. [6], which do not require to use the projection of the equations along characteristic directions. Capilla and Balaguer-Beser [8] have applied the methodology of [6] to solve the one-dimensional sediment transport equations using the same temporal integration scheme. However the WENO spatial reconstruction is replaced by a point value reconstruction algorithm based on average or flux values, which satisfies the monotonicity preserving property. The agreement between the results of [8] and what is reported in literature has been confirmed, showing the versatility and the capability of the proposed scheme that consists of a high-order finite volume scheme with simplified numerical flux evaluation.

In this paper, we present an extension of the central schemes described in [8] for the two-dimensional shallow water equations. We will define a new central finite volume scheme which maintains the exact C-property and it is high-order accurate. For space discretization we will extend the one-dimensional reconstruction procedure defined by Balaguer and Conde in [1] to the case of rectangular meshes, in the framework of the two-dimensional central schemes described in Levy et al. [14]. Interpolation procedure will use centered three-degree polynomials with a modification on the slope at the midpoint of the stencil, which has been designed so that the resulting polynomial has the same type of monotony of the interpolated data.

The outline of the paper is as follows. Section 2 shows the two dimensional shallow water systems, which have been manipulated in order to use the water surface elevation instead of the water depth. Section 3 describes the central scheme using a Runge-Kutta scheme with a natural continuous extension for time discretization. A special attention is paid to the reconstruction of point-values and averages at time t^n. The new procedure designed for the source term integration is presented in subsection 3.2. The reconstruction of the Runge-Kutta fluxes defined in [6] has been extended to two dimensions in space in subsection 3.3. Section 4 analyzes the well-balanced

properties of our scheme showing the C-property verification. Finally, some numerical tests are performed to verify the well-balanced property, high-order accuracy, and good resolution for smooth and discontinuous solutions.

2. Two dimensional shallow water systems

The shallow water system in two spatial dimensions takes the form:

$$\begin{cases} h_t + (q_1)_x + (q_2)_y = 0\,, \\ (q_1)_t + \left(\dfrac{q_1^2}{h} + \dfrac{1}{2}gh^2\right)_x + \left(\dfrac{q_1\,q_2}{h}\right)_y = -gh(Z_b)_x\,, \\ (q_2)_t + \left(\dfrac{q_1\,q_2}{h}\right)_x + \left(\dfrac{q_2^2}{h} + \dfrac{1}{2}gh^2\right)_y = -gh(Z_b)_y\,, \end{cases} \tag{1}$$

which are the equations governing the flow of a shallow layer of homogeneous fluid in a two dimensional domain $D \subset \mathbb{R}^2$. In the equations, $h(x,y,t)$ is the water depth; $q_j(x,y,t)$ is the component of the discharge $\vec{q}$ in the direction j, related to the velocity of the fluid $\vec{v} = (v_1(x,y,t), v_2(x,y,t))$ by the expression $q_j(x,y,t) = h(x,y,t)\cdot v_j(x,y,t)$; $Z_b(x,y)$ is the function that specifies the bottom topography and g is the gravity acceleration.

We manipulate the system in Equation (1) in order to use the water surface elevation $\eta(x,y,t) = h(x,y,t) + Z_b(x,y)$ instead of the water depth; then the shallow water system can be written as:

$$\begin{pmatrix} \eta \\ q_1 \\ q_2 \end{pmatrix}_t + \begin{pmatrix} q_1 \\ \frac{q_1^2}{\eta - Z_b} + \frac{1}{2}g(\eta - Z_b)^2 \\ \frac{q_1 q_2}{\eta - Z_b} \end{pmatrix}_x + \begin{pmatrix} q_2 \\ \frac{q_1 q_2}{\eta - Z_b} \\ \frac{q_2^2}{\eta - Z_b} + \frac{1}{2}g(\eta - Z_b)^2 \end{pmatrix}_y = \begin{pmatrix} 0 \\ -g(\eta - Z_b)(Z_b)_x \\ -g(\eta - Z_b)(Z_b)_y \end{pmatrix}. \tag{2}$$

System (2) can be expressed as:

$$u_t + f(u)_x + g(u)_y = s(x,y,u)\,. \tag{3}$$

Here, $u = (h, q_1, q_2)^{\mathrm{T}}$ is the vector of conservative variables, where the superscript $^{\mathrm{T}}$ denotes the transpose of a vector (or of a matrix); $f(u)$ and $g(u)$ are the flux vector valued functions and $s(x,y,u)$ is the source term which models the effects of the shape of the bottom on the flow.

We consider only flows over *wet* bed, $h(x,y,t) > \epsilon > 0$, for all x, y, t. The extension to *dry* flows ($h(x,y,t) \geq 0$) will be reported elsewhere. In the simplified case of flat bottom, in which we will assume $Z_b(x,y) = 0$ for all (x,y), system (2) expresses the conservation of the total water height $\eta = h$ and of the discharge $\vec{q}$ in presence of the gravity force.

In [8] we presented a high-order well-balanced numerical scheme for solving the one dimensional shallow water system. In this paper, the numerical scheme of [8] is extended to two spatial dimensions using a uniform rectangular grid and following the methodology developed in [2] and [14]. In the next sections we develop the numerical model adapting it to the resolution of the two dimensional shallow water system (2). This involves designing a new source term treatment to verify the exact C-property.

3. Central scheme

We consider that the time interval is discretized into NT values, being Δt the time step, where $t^n = n \cdot \Delta t$ for $n = 0, 1, 2, ...,$ NT. The spatial discretization of the domain is based on the mesh sizes Δx and Δy in x and y directions respectively, so we use in the calculations the grid defined by the points $x_i = x_{i-1} + \Delta x$ and $y_j = y_{j-1} + \Delta y$, and the staggered grid defined by $x_{i+\frac{1}{2}} = x_i + \frac{\Delta x}{2}$ and $y_{j+\frac{1}{2}} = y_j + \frac{\Delta y}{2}$, for $i = 1, 2, ...,$ NX and $j = 1, 2, ...$ NY.

The central finite volume method integrates the system (2) with respect to the spatial and time variables over the control volume $I_{i+\frac{1}{2},j+\frac{1}{2}} \times [t^n, t^{n+1}]$, being $I_{i+\frac{1}{2},j+\frac{1}{2}} = [x_i, x_{i+1}] \times [y_j, y_{j+1}]$, resulting with

$$\overline{\overline{u}}^{n+1}_{i+\frac{1}{2},j+\frac{1}{2}} = \overline{\overline{u}}^{n}_{i+\frac{1}{2},j+\frac{1}{2}} - \frac{1}{\Delta x \Delta y} \int_{t^n}^{t^{n+1}} \left\{ \int_{y_j}^{y_{j+1}} \left[f_{i+1}(y,\tau) - f_i(y,\tau) \right] \, dy \right\} d\tau$$
$$- \frac{1}{\Delta x \Delta y} \int_{t^n}^{t^{n+1}} \left\{ \int_{x_i}^{x_{i+1}} \left[g_{j+1}(x,\tau) - g_j(x,\tau) \right] \, dx \right\} d\tau + \int_{t^n}^{t^{n+1}} \overline{\overline{s}}_{i+\frac{1}{2},j+\frac{1}{2}}(\tau) \, d\tau \,, \tag{4}$$

where for simplicity of notation we have denoted $f_i(y,t) = f(u(x_i, y, t))$ and $g_j(x,t) = g(u(x, y_j, t))$. The first integral on the right-hand side (rhs) of (4) is the cell average of the function $u(x, y, t^n)$ on the staggered cell $I_{i+\frac{1}{2},j+\frac{1}{2}}$, given by

$$\overline{\overline{u}}^{n}_{i+\frac{1}{2},j+\frac{1}{2}} = \frac{1}{\Delta x \Delta y} \int_{x_i}^{x_{i+1}} \int_{y_j}^{y_{j+1}} u(x, y, t^n) \, dy \, dx \,, \tag{5}$$

and the cell average in the last time integral on the rhs of (4) is

$$\overline{\overline{s}}_{i+\frac{1}{2},j+\frac{1}{2}}(\tau) = \frac{1}{\Delta x \Delta y} \int_{x_i}^{x_{i+1}} \int_{y_j}^{y_{j+1}} s(u(x, y, \tau)) \, dy \, dx \,. \tag{6}$$

In order to obtain the fourth order accuracy in time, a Gaussian quadrature with two integration nodes is selected to evaluate the time flux integrals in Equation (4), for example:

$$\int_{t^n}^{t^{n+1}} f(u(x_i, y_j, z)) dz = \frac{\Delta t}{2} \left(f(\hat{u}^{n+\beta_0}_{i,j}) + f(\hat{u}^{n+\beta_1}_{i,j}) \right) \,, \tag{7}$$

where $\hat{u}^{n+\beta_k}_{i,j} = u(x_i, y_j, t^n + \beta_k \Delta t)$, $k = 0, 1$, being

$$\beta_0 = \left(\frac{1 - 1/\sqrt{3}}{2} \right) \,, \qquad \beta_1 = \left(\frac{1 + 1/\sqrt{3}}{2} \right) \,. \tag{8}$$

According to [14], the following centered quadrature rule in space is used for the integrals in space:

$$\int_{y_j}^{y_{j+1}} f_i(y, t) dy = \frac{\Delta y}{24} \left[-f_i(y_{j+2}, t) + 13 f_i(y_{j+1}, t) + 13 f_i(y_j, t) - f_i(y_{j-1}, t) \right] \,. \tag{9}$$

In this way, the quadrature rule for approximating the integrals of the fluxes involves nodes on the segments $(x_i, y_j) \times [t^n, t^{n+1}]$ where the solution remains smooth.

In the next subsections we present a summary of the procedure involved in a computational time step to obtain the cell-averages $\overline{\overline{u}}^{n+1}_{i+\frac{1}{2},j+\frac{1}{2}}$ at the next time step t^{n+1}, starting from Equation (4).

3.1. Reconstruction at time t^n of point-values and averaged values

At time t^n we start the reconstruction with a given fourth order approximation of the following cell averages:

$$\overline{\overline{u}}^n_{i,j} = \frac{1}{\Delta x \Delta y} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} u(x,y,t^n)\, dy\; dx\,. \tag{10}$$

We will denote by $I_{i,j}$ the cell centered around the grid point (x_i, y_j): $I_{i,j} = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \times [y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}]$. A two-dimensional piecewise-polynomial reconstruction is computed from the data $\{\overline{\overline{u}}^n_{i,j}\}$ resulting with

$$R_{i,j}(x,y;\overline{\overline{u}}^{\,n}) \equiv R_{i,j}(x,y,t^n) = u(x,y,t^n) + O(h^4)\,, \qquad \forall\, x,y \in I_{i,j}\,, \tag{11}$$

where $h = \Delta x = \Delta y$ and $R_{i,j}(x,y,t^n)$ is a bicubic polynomial obtained simply through the tensor product of two one-dimensional interpolating polynomials, which were defined and analyzed in [1]. Also, we used these 1D reconstruction polynomials in the numerical model and applications presented in a previous work [8]. Those 1D polynomials take this form:

$$\begin{aligned} Q_{i,j}(x;\overline{\overline{u}}^n) = \overline{\overline{u}}^n_{i,j} + \theta^n_i \Bigg(& \frac{-\overline{\overline{u}}^n_{i-1,j} + 2\overline{\overline{u}}^n_{i,j} - \overline{\overline{u}}^n_{i+1,j}}{24} + \left(\frac{\overline{\overline{u}}^n_{i-1,j} - \overline{\overline{u}}^n_{i+1,j} + 10 d^n_i}{8}\right)\left(\frac{x-x_i}{\Delta x}\right) \\ & + \left(\frac{\overline{\overline{u}}^n_{i-1,j} - 2\overline{\overline{u}}^n_{i,j} + \overline{\overline{u}}^n_{i+1,j}}{2}\right)\left(\frac{x-x_i}{\Delta x}\right)^2 + \left(\frac{-\overline{\overline{u}}^n_{i-1,j} + \overline{\overline{u}}^n_{i+1,j} - 2d^n_i}{2}\right)\left(\frac{x-x_i}{\Delta x}\right)^3 \Bigg)\,. \end{aligned} \tag{12}$$

Parameter d^n_i is defined in order to verify some shape-preserving properties when the cell-averages form a monotone sequence (see [1]) and parameter θ^n_i is defined so as to eliminate spurious interface extrema (see [15]). The definition of d^n_i may cause that the resulting numerical scheme not to be of order $O((\triangle x)^4)$ in a small number of cells, especially near a local maximum, a local minimum, or a discontinuity. That will be commented below in numerical experiments. d^n_i takes a value equal to zero in the case where the solution is reconstructed from constant average values. This will be useful in the C-property verification (Section 4). Full details of the definition of θ^n_i and d^n_i are given in [8], pages 22-23.

The 2D-reconstruction polynomial is defined using (12) so that:

$$R_{i,j}(x,y;\overline{\overline{u}}^{\,n}) = Q_{i,j}(y; Q(x;\overline{\overline{u}}^{\,n}))\,. \tag{13}$$

The cell averages, $\overline{\overline{u}}^n_{i+\frac{1}{2},j+\frac{1}{2}}$ in Equation (5) can be approximated using the polynomials: $R_{i+1,j+1}(x,y,t^n)$, $R_{i,j+1}(x,y,t^n)$, $R_{i+1,j}(x,y,t^n)$ and $R_{i,j}(x,y,t^n)$ on the corresponding quarter cells, using a Gaussian quadrature rule. Let $I^m_{i,j}$, $m = 1, ..., 4$ denote the four quarters of the cell $I_{i,j}$, with $I^1_{i,j}$ being the upper-right quarter, while the other three quarters are numbered clockwise, as can bee seen in Figure 1.

A fourth-order computation of the cell averages in (5) is obtained through the averages of $R_{i,j}(x,y,t^n)$ over the four quarter cells:

$$\overline{\overline{R}}^{(m)}_{i,j} \simeq \frac{1}{\Delta x\, \Delta y} \int_{I^m_{i,j}} R_{i,j}(x,y,t^n) dx\, dy\,, \qquad m = 1, ..., 4\,. \tag{14}$$

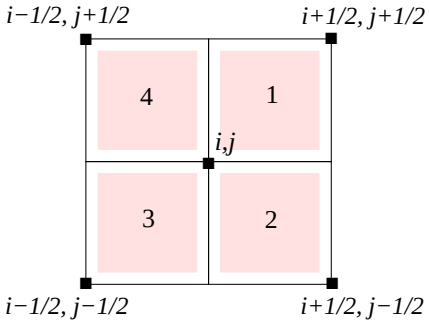

Figure 1: The quarter cells of the cell $I_{i,j}$.

Then the cell average of the solution in (5) is the sum of the four quarter-cell averages defined in (14):

$$\overline{\overline{u}}^{n}_{i+\frac{1}{2},j+\frac{1}{2}} = \overline{\overline{R}}^{(1)}_{i,j} + \overline{\overline{R}}^{(2)}_{i,j+1} + \overline{\overline{R}}^{(3)}_{i+1,j+1} + \overline{\overline{R}}^{(4)}_{i+1,j} \, .$$

Using the reconstruction polynomials $R_{i,j}(x, y, t^n)$, we also approximate the point-values $\hat{u}^n_{i,j}$ at time t^n on the non-staggered grid.

3.2. Source term integration

The second component of the cell average of the source term in (6) is given by

$$\overline{\overline{s}}^{[2]}_{i+\frac{1}{2},j+\frac{1}{2}} = \frac{1}{\Delta y} \int_{y_j}^{y_{j+1}} \left(\frac{-1}{\Delta x} \int_{x_i}^{x_{i+1}} g(\eta - Z_b) \frac{\partial Z_b}{\partial x} \, dx \right) \, dy \, . \tag{15}$$

Following the procedure described in [8], an integration by parts with respect to the x variable is performed, in order to involve the spatial derivative of the free surface elevation instead of the bed elevation. This was suggested by Caleffi et al. in [6] for the one dimensional case. Thus, initially we compute:

$$\begin{aligned} \overline{s}^{[2]}_{i+\frac{1}{2}}(y) &= \frac{-1}{\Delta x} \int_{x_i}^{x_{i+1}} g(\eta(x,y) - Z_b(x,y)) \frac{\partial Z_b(x,y)}{\partial x} \, dx \\ &= \frac{g}{2\Delta x} \left[\hat{Z}^2_{b,i+1}(y) - \hat{Z}^2_{b,i}(y) - 2\hat{\eta}_{i+1}(y)\hat{Z}_{b,i+1}(y) + 2\hat{\eta}_i(y)\hat{Z}_{b,i}(y) \right] \\ &+ \frac{1}{\Delta x} \int_{x_i}^{x_{i+1}} \psi(x,y) \, dx \, , \end{aligned} \tag{16}$$

where $\psi(x, y) = g \cdot Z_b(x, y) \cdot \frac{\partial \eta(x,y)}{\partial x}$. We obtain a formulation which is preferable for the numerical integration of the bed slope source terms because it involves the spatial derivative of the free surface elevation instead of the spatial derivative of the bed elevation and, in general, the free surface profile is more regular than the bed profile, even if the solution is discontinuous. Moreover, the integration by parts used in equation (16) is essential to get the C-property verification, as it will be demonstrated below in Section 4.

The spatial integral in Equation (15) in the y variable is evaluated using a centered quadrature rule:

$$\overline{\overline{s}}^{[2]}_{i+\frac{1}{2},j+\frac{1}{2}} = \frac{1}{24} \left[-\overline{s}^{[2]}_{i+\frac{1}{2}}(y_{j+2}) + 13\,\overline{s}^{[2]}_{i+\frac{1}{2}}(y_{j+1}) + 13\,\overline{s}^{[2]}_{i+\frac{1}{2}}(y_j) - \overline{s}^{[2]}_{i+\frac{1}{2}}(y_{j-1}) \right] \, . \tag{17}$$

In a similar way we evaluate the third component of the averaged source term in (6), obtaining:

$$\begin{aligned}\overline{s}^{[3]}_{j+\frac{1}{2}}(x) &= \frac{g}{2\Delta y}\left[\hat{Z}^2_{b,j+1}(x) - \hat{Z}^2_{b,j}(x) - 2\hat{\eta}_{j+1}(x)\hat{Z}_{b,j+1}(x) + 2\hat{\eta}_j(x)\hat{Z}_{b,j}(x)\right] \\ &+ \frac{1}{\Delta y}\int_{y_j}^{y_{j+1}} \phi(x,y)\; dy\,,\end{aligned} \tag{18}$$

where $\phi(x,y) = g \cdot Z_b(x,y) \cdot \frac{\partial \eta(x,y)}{\partial y}$ and

$$\int_{y_j}^{y_{j+1}} \phi(x,y)\; dy = \int_{y_j}^{y_j+\frac{\triangle y}{2}} P_{j,x}(y;\hat{\phi})\; dy + \int_{y_j+\frac{\triangle y}{2}}^{y_{j+1}} P_{j+1,x}(y;\hat{\phi})\; dy,$$

where the interpolating three-degree polynomial $P_{j,x}(y;\hat{\phi})$ approximates point values starting from point-values (see [8]). Again, we apply a centered quadrature rule for the integral with respect to the x variable.

The source term time integral in (4) is also evaluated using a Gaussian quadrature rule with two nodes:

$$\int_{t^n}^{t^{n+1}} \overline{\overline{s}}_{i+\frac{1}{2},j+\frac{1}{2}}(\tau)\, d\tau = \frac{\Delta t}{2}\left(\overline{\overline{s}}^{n+\beta_0}_{i+\frac{1}{2},j+\frac{1}{2}} + \overline{\overline{s}}^{n+\beta_1}_{i+\frac{1}{2},j+\frac{1}{2}}\right)\,, \tag{19}$$

where

$$\overline{\overline{s}}^{n+\beta_k}_{i+\frac{1}{2},j+\frac{1}{2}} = \frac{1}{\Delta x \Delta y}\int_{x_i}^{x_{i+1}}\int_{y_j}^{y_{j+1}} s(u(x,y,t^n+\beta_k\Delta t))\; dy\, dx\,, \qquad k = 0,1\,, \tag{20}$$

and the constants β_0 and β_1 where defined in (8).

3.3. Reconstruction of Runge-Kutta fluxes

In order to evaluate the time integrals of the source term (19) and the time flux integrals in (7) we have to predict the point-values of the solution at two intermediate states: $\hat{u}^{n+\beta_k}_{i,j} \equiv u(x_i, y_j, t^n + \beta_k\,\Delta t)$, $k = 0, 1$. The prediction of these intermediate values at times $t^{n+\beta_0}$ and $t^{n+\beta_1}$ is obtained by means of a Runge-Kutta scheme coupled with the natural continuous extension (NCE) [5]:

$$\hat{u}^{n+\beta_k}_{i,j} \equiv u(x_i, y_j, t^n + \beta_k\,\Delta t) = \hat{u}^n_{i,j} + \Delta t \sum_{l=1}^{4} b_l(\beta_k)\, k^{(l)}_{i,j}\,, \tag{21}$$

where the constants $b_l(\beta_k)$ are given in [8] and $k^{(l)}_{i,j}$, $1 \le l \le 4$, are the Runge-Kutta fluxes, which coincide with a numerical evaluation of $(-f_x - g_y + s)$ in the shallow water system (3). We use the point-values of the solution $\{\hat{u}^{(l)}_{k_1,j}\}$ with $k_1 \in \{i-2, i-1, i, i+1, i+2\}$, and $\{\hat{u}^{(l)}_{i,k_2}\}$ with $k_2 \in \{j-2, j-1, j, j+1, j+2\}$, to calculate the functions F^j_i and G^i_j respectively, defined by

$$F^j_i(x_{k_1};\hat{u}) = -\left[f(\hat{u}_{k_1,j}) - f(\hat{u}_{i,j})\right] + \begin{bmatrix} 0 \\ \frac{1}{2}g\left[(\hat{\eta}_{i,j} - \hat{Z}_{b,k_1}(y_j))^2 - (\hat{\eta}_{i,j} - \hat{Z}_{b,i}(y_j))^2\right] \\ 0 \end{bmatrix}\,,$$

$$G^i_j(y_{k_2};\hat{u}) = -\left[g(\hat{u}_{i,k_2}) - g(\hat{u}_{i,j})\right] + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{2}g\left[(\hat{\eta}_{i,j} - \hat{Z}_{b,k_2}(x_i))^2 - (\hat{\eta}_{i,j} - \hat{Z}_{b,j}(x_i))^2\right] \end{bmatrix}\,, \tag{22}$$

where for simplicity we have omitted the index (l) from $\hat{u}_{i,j}^{(l)}$. Equation (22) is a two-dimensional extension of the definition presented in [6, 8]. This definition, (22), guarantees that the numerical scheme maintains the exact C-property. We use the computed point-values of the functions F_i^j and G_j^i to calculate the interpolating polynomials $P_i(x; F_i^j)$ and $P_j(y; G_j^i)$. The evaluation of the Runge-Kutta fluxes is approximated using these interpolating polynomials, by means of:

$$k_{i,j}^{(l)} \equiv k_{i,j} = \frac{dP_i(x; F^j)}{dx} + \frac{dP_j(y; G^i)}{dy}\,, \tag{23}$$

where:

$$\begin{aligned} P_i(x; F^j) &= F_i^j + \theta_i^j \left[dP_i^j \cdot \left(\frac{x - x_i}{\Delta x} \right) + \left(\frac{F_{i-1}^j - 2F_i^j + F_{i+1}^j}{2} \right) \cdot \left(\frac{x - x_i}{\Delta x} \right)^2 \right. \\ &+ \left. \left(\frac{-F_{i-1}^j + F_{i+1}^j - 2dP_i^j}{2} \right) \cdot \left(\frac{x - x_i}{\Delta x} \right)^3 \right]. \end{aligned} \tag{24}$$

Again, θ_i^j and dP_i^j are defined in order to avoid the appearance of false extrema on the numerical solution. For more details about these parameters we refer the reader to [1] and [8].

4. *C*-property verification

In this Section we will prove that our numerical scheme satisfies the exact C-property. It can be verified that in case of quiescent flow, starting from $\overline{\overline{\eta}}_{i,j}^n = \eta^*$ constant and $\overline{\overline{q_k}}_{;i,j}^n = 0$, where $q_k = v_k h$, for $k = 1, 2$, gives $\overline{\overline{\eta}}_{i,j}^{n+2} = \hat{\eta}_{i,j}^{n+2} = \eta^*$ and $\overline{\overline{q_k}}_{;i,j}^{n+2} = \hat{q_k}_{;i,j}^{n+2} = 0$, $\forall\, n$. This means that, starting from the mentioned initial conditions, the scheme maintains the steadiness of the point-values and the cell averages of the solution.

4.1. C-property for point-values

In the case of quiescent flow, the scheme must preserve the steadiness of the point-value solution, i.e., $\hat{u}_{i,j}^{n+\beta_k} = \hat{u}_{i,j}^n$, for $k = 0, 1$. The time evolution of point-values is described by Equation (21), thus, to achieve this result, we have to demonstrate that the Runge-Kutta fluxes $k_{i,j} = 0$, $\forall\, i, j$, when $q_1 = v_1 h = 0$, $q_2 = v_2 h = 0$ and $\eta = \eta^* =$ constant. We will see that in case of quiescent flow, each component of the functions F_i^j and G_j^i defined in (22), is identically zero. From Equations (2), (3) and (22) it can be seen through direct calculations that the first and third components of F_i^j are identically zero. Analogously, the first and second components of G_j^i are zero.

For the second component of F_i^j we have:

$$F_i^{j;[2]}(x_k; \hat{u}) = -\left[f^{[2]}(\hat{u}_{k,j}) - f^{[2]}(\hat{u}_{i,j}) \right] + \frac{1}{2} g \left[(\hat{\eta}_{i,j} - \hat{Z}_{b,k}(y_j))^2 - (\hat{\eta}_{i,j} - \hat{Z}_{b,i}(y_j))^2 \right],$$

where $f^{[2]}(\hat{u}_{i,j})$ is the second component of the flux function $f(u)$ at $x = x_i$ and $y = y_j$. If $q_1 = q_2 = 0$ and $\eta = \eta^*$, then

$$-\left[f^{[2]}(\hat{u}_{k,j}) - f^{[2]}(\hat{u}_{i,j}) \right] = -\frac{1}{2} g \left[(\eta^* - \hat{Z}_{b,k}(y_j))^2 - (\eta^* - \hat{Z}_{b,i}(y_j))^2 \right],$$

and therefore the second component of F_i^j is also identically zero. Similarly, we obtain that the third component of G_j^i is zero, and then the point-values $\hat{F}_i^j(x_k; \hat{u}) = \hat{G}_j^i(y_k; \hat{u}) = 0$, $\forall\, i, j, k$. Consequently, the polynomials $P_i(x; F_i^j)$ and $P_j(y; G_j^i)$ in (23) are zero for any x and y, and then $k_{i,j} = 0$, $\forall\, i, j$.

4.2. C-property for cell averages

The steadiness of the cell averages of the solution, i.e., $\overline{\overline{u}}_{i,j}^{n+2} = \overline{\overline{u}}_{i,j}^{n}$, in the case of quiescent flow, must be verified. To achieve this result, it is sufficient to show that in the following equations the vector valued term in square brackets is zero, for water at rest:

$$\overline{\overline{u}}_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} = \overline{\overline{u}}_{i+\frac{1}{2},j+\frac{1}{2}}^{n} - \frac{1}{\Delta x \Delta y} \int_{t^n}^{t^{n+1}} \left[\int_{y_j}^{y_{j+1}} \left(f_{i+1}(y,\tau) - f_i(y,\tau) \right) \, dy \right.$$
$$\left. + \int_{x_i}^{x_{i+1}} \left(g_{j+1}(x,\tau) - g_j(x,\tau) \right) \, dx - \Delta x \Delta y \, \overline{\overline{s}}_{i+\frac{1}{2},j+\frac{1}{2}}(\tau) \right] d\tau \,, \tag{25}$$

$$\overline{\overline{u}}_{i,j}^{n+2} = \overline{\overline{u}}_{i,j}^{n+1} - \frac{1}{\Delta x \Delta y} \int_{t^{n+1}}^{t^{n+2}} \left[\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} \left(f_{i+\frac{1}{2}}(y,\tau) - f_{i-\frac{1}{2}}(y,\tau) \right) \, dy \right.$$
$$\left. + \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \left(g_{j+\frac{1}{2}}(x,\tau) - g_{j-\frac{1}{2}}(x,\tau) \right) \, dx - \Delta x \Delta y \, \overline{\overline{s}}_{i,j}(\tau) \right] d\tau \,, \tag{26}$$

so that

$$\overline{\overline{\eta}}_{i,j}^{n+2} = \overline{\overline{\eta}}_{i,j}^{n+1} = \overline{\overline{\eta}}_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} = \overline{\overline{\eta}}_{i+\frac{1}{2},j+\frac{1}{2}}^{n} = \eta^*$$

and

$$\overline{\overline{q_k}}_{;i,j}^{n+2} = \overline{\overline{q_k}}_{;i,j}^{n+1} = \overline{\overline{q_k}}_{;i+\frac{1}{2},j+\frac{1}{2}}^{n+1} = \overline{\overline{q_k}}_{;i+\frac{1}{2},j+\frac{1}{2}}^{n} = 0 \,.$$

We can notice that starting from $\overline{\overline{\eta}}_{i,j}^{n} = \eta^*, \forall i,j$, it can be verified that $R_{i,j}(x,y,t^n) = \eta^*, \forall (x,y) \in I_{i,j}^m, , \forall i,j, \ 1 \le m \le 4$ (see Figure 1). Then, according to equation (14), $\overline{\overline{R}}_{i,j}^{(m)} = \frac{\eta^*}{4}, \forall i,j, \ 1 \le m \le 4$ and:

$$\overline{\overline{\eta}}_{i+\frac{1}{2},j+\frac{1}{2}}^{n} = \overline{\overline{R}}_{i,j}^{(1)} + \overline{\overline{R}}_{i,j+1}^{(2)} + \overline{\overline{R}}_{i+1,j+1}^{(3)} + \overline{\overline{R}}_{i+1,j}^{(4)} = \eta^* \,.$$

In a similar way, we can prove that $\overline{\overline{q_k}}_{;i+\frac{1}{2},j+\frac{1}{2}}^{n} = 0$.

After applying equation (25) we get $\overline{\overline{\eta}}_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} = \overline{\overline{\eta}}_{i+\frac{1}{2},j+\frac{1}{2}}^{n} = \eta^*, \forall i,j$. In this case, we can check that $R_{i+\frac{1}{2},j+\frac{1}{2}}(x,y,t^n) = \eta^*, \forall (x,y) \in I_{i+\frac{1}{2},j+\frac{1}{2}}^{m}, \forall i,j, \ 1 \le m \le 4$. Then, we obtain that $\overline{\overline{R}}_{i+\frac{1}{2},j+\frac{1}{2}}^{(m)} = \frac{\eta^*}{4}, \forall i,j, \ 1 \le m \le 4$ so:

$$\overline{\overline{\eta}}_{i,j}^{n+1} = \overline{\overline{R}}_{i-\frac{1}{2},j-\frac{1}{2}}^{(1)} + \overline{\overline{R}}_{i-\frac{1}{2},j+\frac{1}{2}}^{(2)} + \overline{\overline{R}}_{i+\frac{1}{2},j+\frac{1}{2}}^{(3)} + \overline{\overline{R}}_{i+\frac{1}{2},j-\frac{1}{2}}^{(4)} = \eta^* \,.$$

Similarly it holds that $\overline{\overline{q_k}}_{;i,j}^{n} = 0$. From equation (26) we obtain that $\overline{\overline{\eta}}_{i,j}^{n+2} = \overline{\overline{\eta}}_{i,j}^{n+1} = \eta^*$ and $\overline{\overline{q_k}}_{;i,j}^{n+2} = \overline{\overline{q_k}}_{;i,j}^{n+1} = 0$.

It can be seen from Equation (2), that the first component of the term in square brackets in (25) (similarly in (26)) is identically zero. We have to prove that the second and third components of this term are zero. Remembering Equation (16) and assuming $\hat{\eta}_i(y) = \hat{\eta}_{i+1}(y) = \eta^* = \text{constant}$, the second component of the term in square brackets in (25) may be written as:

$$\int_{y_j}^{y_{j+1}} \left[f_{i+1}^{[2]}(y,\tau) - f_i^{[2]}(y,\tau) \right] \, dy - \Delta x \int_{y_j}^{y_{j+1}} \overline{s}_{i+\frac{1}{2}}^{[2]}(y,\tau) \, dy \tag{27}$$

$$= \int_{y_j}^{y_{j+1}} \frac{1}{2} g \left[(\eta^* - \hat{Z}_{b,i+1}(y))^2 - (\eta^* - \hat{Z}_{b,i}(y))^2 \right] dy$$
$$- \int_{y_j}^{y_{j+1}} \left[\frac{1}{2} g (\hat{Z}^2_{b,i+1}(y) - \hat{Z}^2_{b,i}(y)) - g\eta^* (\hat{Z}_{b,i+1}(y) - \hat{Z}_{b,i}(y)) + \Delta x \, \overline{\psi}_{i+\frac{1}{2}}(y) \right] dy$$
$$= -\Delta x \int_{y_j}^{y_{j+1}} \overline{\psi}_{i+\frac{1}{2}}(y) \, dy,$$

where $\overline{\psi}_{i+\frac{1}{2}}(y)$ is the cell average of the function $\psi(x,y) = g\, Z_b(x,y)\, \frac{\partial \eta}{\partial x}$ on the cell $[x_i, x_{i+1}]$.

In a similar way, it can be proved that the third component of the term in square brackets in (25) is equal to $-\Delta y \int_{x_i}^{x_{i+1}} \overline{\phi}_{j+\frac{1}{2}}(x)\, dx$, where $\phi(x,y) = g\, Z_b(x,y)\, \frac{\partial \eta}{\partial y}$, so if $\overline{\psi}_{i+\frac{1}{2}}(y)$ and $\overline{\phi}_{j+\frac{1}{2}}(x)$ are zero then the terms in square brackets are also zero. In the case of quiescent flow, if $\hat{\eta}_{i,j}$ are constants $\forall\, i,j$, then the free surface derivatives $\hat{\eta}'_{i,j} = 0$. Consequently, the point-values $\hat{\psi}_{i,j}$ and $\hat{\phi}_{i,j}$ are zero. Finally, the cell averages of the functions ψ and ϕ are approximated using the point-values $\hat{\psi}_{i,j}$ and $\hat{\phi}_{i,j}$, and therefore if $\hat{\psi}_{i,j} = \hat{\phi}_{i,j} = 0$ also the averaged values $\overline{\psi}_{i+\frac{1}{2}}(y)$ and $\overline{\phi}_{j+\frac{1}{2}}(x)$ are zero, and the C-property verification for cell averages is proved.

5. Applications

In this Section we examine the behaviour and accuracy of the numerical scheme, in several numerical tests. We show a numerical verification of the exact C-property and we present results for several standard tests proposed in the literature. In all cases, we consider $\Delta x = \Delta y$ for the spatial steps. In order to assure numerical stability, an adaptive time step satisfying the following CFL-like condition [14], which is related to the numerical stability of the Runge-Kutta scheme used, must be selected:

$$\Delta t < \frac{h}{2} \frac{1}{\max(|\sigma_x|, |\sigma_y|)},$$

where σ_x and σ_y are the largest (in modulus) eigenvalues of the Jacobian of f and g (3), and $h = \Delta x = \Delta y$. However, we focus the attention on the behaviour of the central scheme described in the paper. Thus, all calculations have been performed using a fixed mesh ratio, $\Delta t/h = 0.0025$, obtaining a good convergence and numerical stability. This makes the scheme particularly adaptable to other type of problems.

The numerical scheme developed in previous sections has been implemented into a FORTRAN code. All calculations have been performed on a personal computer Intel Core(TM) i5-3550 - 3.30 GHz with 8 Gb RAM.

5.1. Test for the exact C-property

This test, presented in [22], is used to verify numerically that our scheme maintains the exact C-property over a non flat bottom. The bottom topography is given by a two-dimensional hump:

$$Z_b(x,y) = 0.8\, e^{-50((x-0.5)^2+(y-0.5)^2)}, \qquad x, y \in [0,1].$$

As initial condition for the water depth we set $h(x,y,0) = 1 - Z_b(x,y)$ and the initial velocity is set to be zero: $q_1(x,y,0) = q_2(x,y,0) = 0$. This surface should remain flat. We consider a rectangular mesh with $\Delta x = \Delta y = 0.01$ which corresponds to a NX=NY=100 uniform mesh. Table 1 contains the L^1 and L^∞ errors at time $t = 0.05$ s, for the water height h and the two components of the discharge q_1 and q_2.

From these results, we can conclude that the exact C-property is verified.

Unknowns	h	q_1	q_2
L^1	0	0	0
L^∞	0	0	0

Table 1: L^1 and L^∞ errors for the C-property analysis

5.2. A two-dimensional accuracy test

In order to check the accuracy of the numerical scheme, we consider a test proposed in [22], that involves a smooth solution of the 2D shallow water system. The computational domain is the unit square and the bottom topography is defined as

$$Z_b(x,y) = \sin(2\pi x) + \cos(2\pi y)\,.$$

The initial conditions for the water depth and the discharge are given by

$$h(x,y,0) = 10 + e^{\sin(2\pi x)}\cos(2\pi y)\,,$$
$$q_1(x,y,0) = \sin(\cos(2\pi x))\sin(2\pi y)\,, \qquad q_2(x,y,0) = \cos(2\pi x)\cos(\sin(2\pi y))\,.$$

Periodic boundary conditions have been imposed. The numerical solution is computed at the ending time $t = 0.05$ s. We use our numerical scheme on a mesh with (1024×1024) grid points and we take this solution as the reference solution. Table 2 shows the L^∞ errors and orders of accuracy.

Cells	h		q_1		q_2	
	L^∞	Order	L^∞	Order	L^∞	Order
64×64	2.96E−02		4.84E−02		2.16E−01	
128×128	3.48E−03	3.09	4.53E−03	3.42	2.80E−02	2.95
256×256	3.40E−04	3.36	8.15E−04	2.48	2.85E−03	3.29

Table 2: L^∞ errors and numerical orders for the 2D accuracy test

The order of accuracy obtained with the L^∞ norm is about 3 because in this problem our reconstruction algorithm makes use of second-degree polynomials in some of the cells containing solution minima. This is due to the limiting used in the reconstruction (d_i^n value), which may pollute the accuracy especially in the L^∞ norm. We have obtained the same conclusion in the one-dimensional scheme as can be seen in [8] (test 4.2).

5.3. Transcritical break of a circular dam

This problem is taken from [20], and it involves flow over flat bed. We consider the break of a circular dam separating two basins with water levels $h = 10$ and $h = 0.5$. The radius of the initial discontinuity is $r = 60$ and the spatial domain of the problem is $[0, 100] \times [0, 100]$. Due to the difference in water height, the flow becomes rapidly transcritical. Then the initial condition for the water height is given by

$$h(x,y,0) = \begin{cases} 10\,, & \text{if } (x^2+y^2) \le 60^2\,, \\ 0.5\,, & \text{otherwise}\,, \end{cases}$$

and $q_1(x, y, 0) = q_2(x, y, 0) = 0$.

The simulations have been run with our numerical scheme until time $t = 3$ s on a uniform mesh with (200×200) nodes. It is useful to introduce the Froude number [20]:

$$Fr = \frac{\sqrt{\vec{v} \cdot \vec{v}}}{a}, \tag{28}$$

with a being the speed of propagation, $a = \sqrt{g\,h}$, and $\vec{v} = (v_1, v_2)$ the velocity of the fluid. Contour plots of computed water depth and Froude number are given in Figure 2.

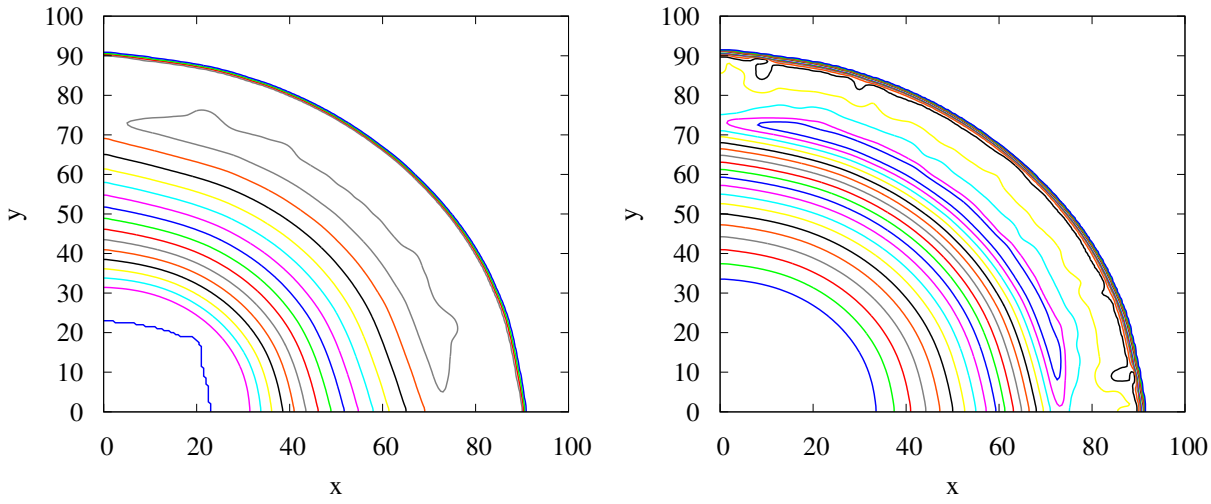

Figure 2: Transcritical dam-break problem: solution at $t = 3$. Left: water height contours with 20 uniformly spaced levels between 0.5 and 10. Right: Froude number contours with 20 uniformly spaced levels between 0 and 1.978.

Figure 3 displays the solution for the water height and Froude number at $t = 3$ along the line $x = y$, on two different uniform meshes with (200×200) points and (600×600) points for comparison.

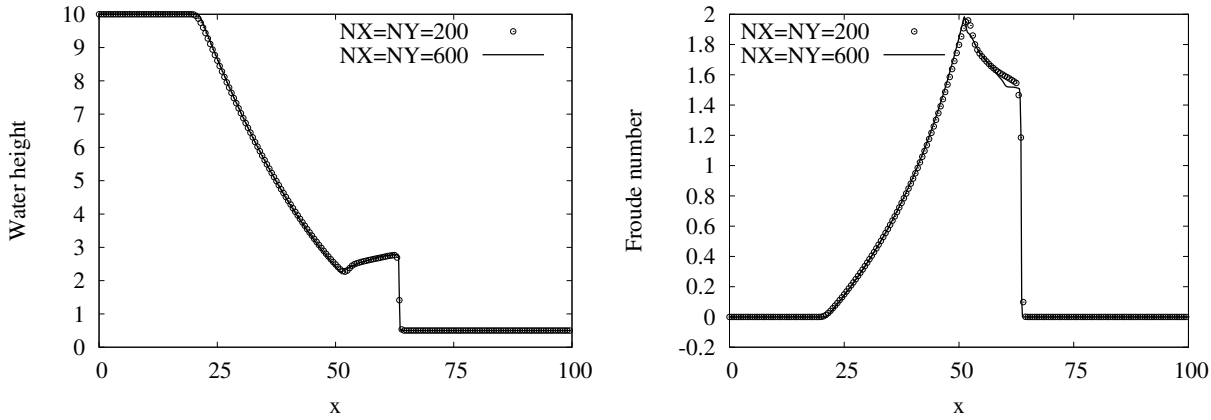

Figure 3: Transcritical dam-break problem: numerical solutions on two different meshes for the water height (left) and Froude number (right) at $t = 3$ s, along the line $y = x$.

5.4. Circular dam-break problem

We consider a test problem described in [10] and [17]. The domain is the square $[0,2] \times [0,2]$ and the bottom topography is given by the function:

$$Z_b(x,y) = \begin{cases} \frac{1}{8}(\cos(2\pi(x-0.5))+1)(\cos(2\pi y)+1)\,, & \text{if } (x-1.5)^2+(y-1)^2 \leq (0.5)^2\,, \\ 0\,, & \text{otherwise}\,. \end{cases}$$

The water depth is initially given by:

$$h(x,y,0) = \begin{cases} 1.1 - Z_b(x,y)\,, & \text{if } (x-1.25)^2+(y-1)^2 \leq (0.1)^2\,, \\ 0.6 - Z_b(x,y)\,, & \text{otherwise}\,, \end{cases}$$

and $q_1(x,y,0) = q_2(x,y,0) = 0$. We compute the numerical results using (200×200) grid points $(\Delta x = \Delta y = 0.01)$. Figure 4 shows the numerical solution for the free surface level at time $t = 0.15$ s. The analytical solution is not known for this problem, and therefore we have used our numerical scheme to obtain a reference solution in a mesh composed by (800×800) cells. In Figure 5 we compare the numerical and the reference solutions for the free surface level and the first component of the discharge q_1 at times $t = 0.05$, $t = 0.1$ and $t = 0.15$ s, in the longitudinal section $y = 1$.

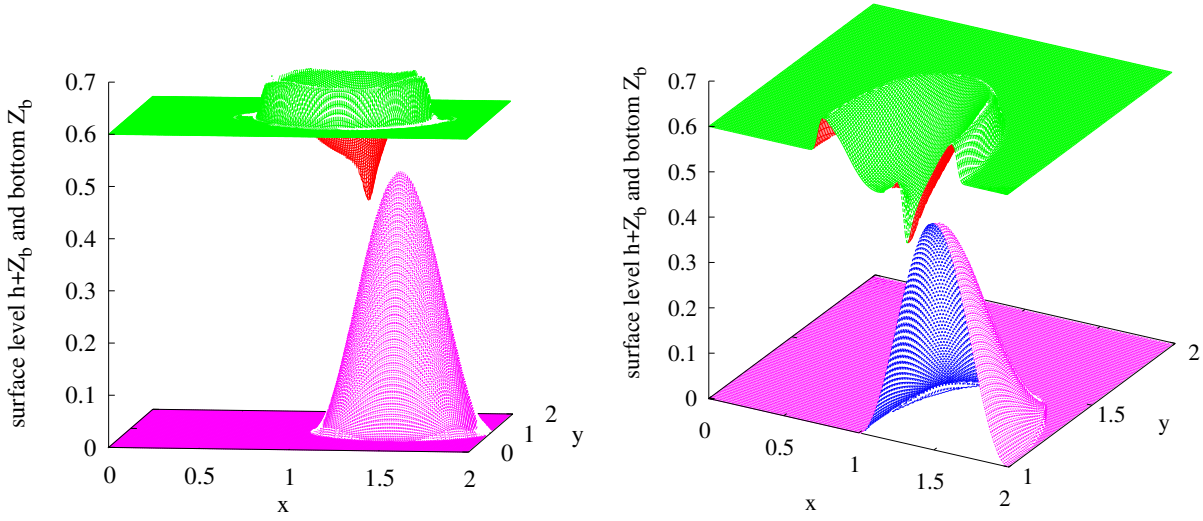

Figure 4: The circular dam-break problem: solution at $t = 0.15$ s. Left: free surface and topography. Right: a longitudinal section at $y = 1$.

These results agree with those in [10] and [17].

5.5. Perturbation of a lake at rest in 2D

In this section we consider a test initially proposed in [13], and recently used in [22, 24, 20, 17]. We show numerical results for a classical example of a small perturbation of a two-dimensional steady-state flow. The perturbation occupies a small portion of the computational domain. The domain is the rectangle $[0,2] \times [0,1]$ and the topography is an isolated elliptical shaped hump centered at $[0.9, 0.5]$:

$$Z_b(x,y) = 0.8\, e^{-5(x-0.9)^2-50(y-0.5)^2}\,.$$

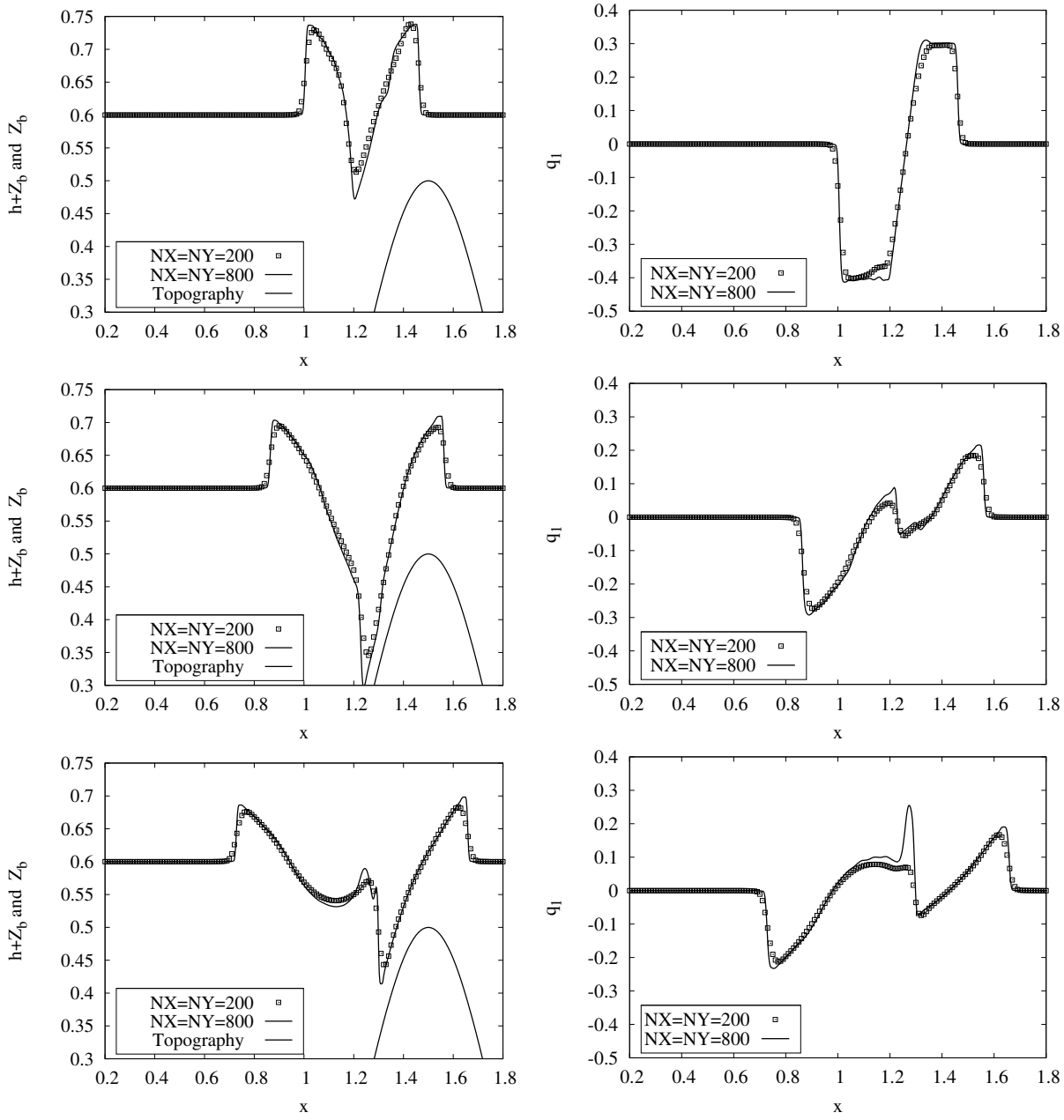

Figure 5: Circular dam-break problem. Comparison between a reference and numerical solution at times t =0.05, 0.1, 0.15 s (from top to bottom), along the line $y = 1$. Left: free surface level. Right: variable q_1.

At $t = 0$ s, the discharge is set to zero everywhere, $q_1(x, y, 0) = q_2(x, y, 0) = 0$ and the water depth is set to

$$h(x, y, 0) = \begin{cases} 1.01 - Z_b(x, y)\,, & \text{if } 0.05 \leq x \leq 0.15\,, \\ 1 - Z_b(x, y)\,, & \text{otherwise}\,. \end{cases}$$

So the surface is almost flat except for $0.05 \leq x \leq 0.15$, where h is perturbed upward by 0.01. We calculate the numerical solution at four different times: $t = 0.12$, $t = 0.24$, $t = 0.36$ and $t = 0.48$, on a uniform mesh with (200×100) nodes, which corresponds to a reference mesh size of

$\Delta x = \Delta y = 0.01$. In Figures 6 and 7 we show the results obtained with our scheme. On the top rows we have plotted the contours of the free surface level $\eta = h + Z_b$, and on the bottom rows, we have plotted a longitudinal section of the distribution of η at $y = 0.5$. The scaling of the bed $Z_b(x, y)$ used in the plots is reported in the figures.

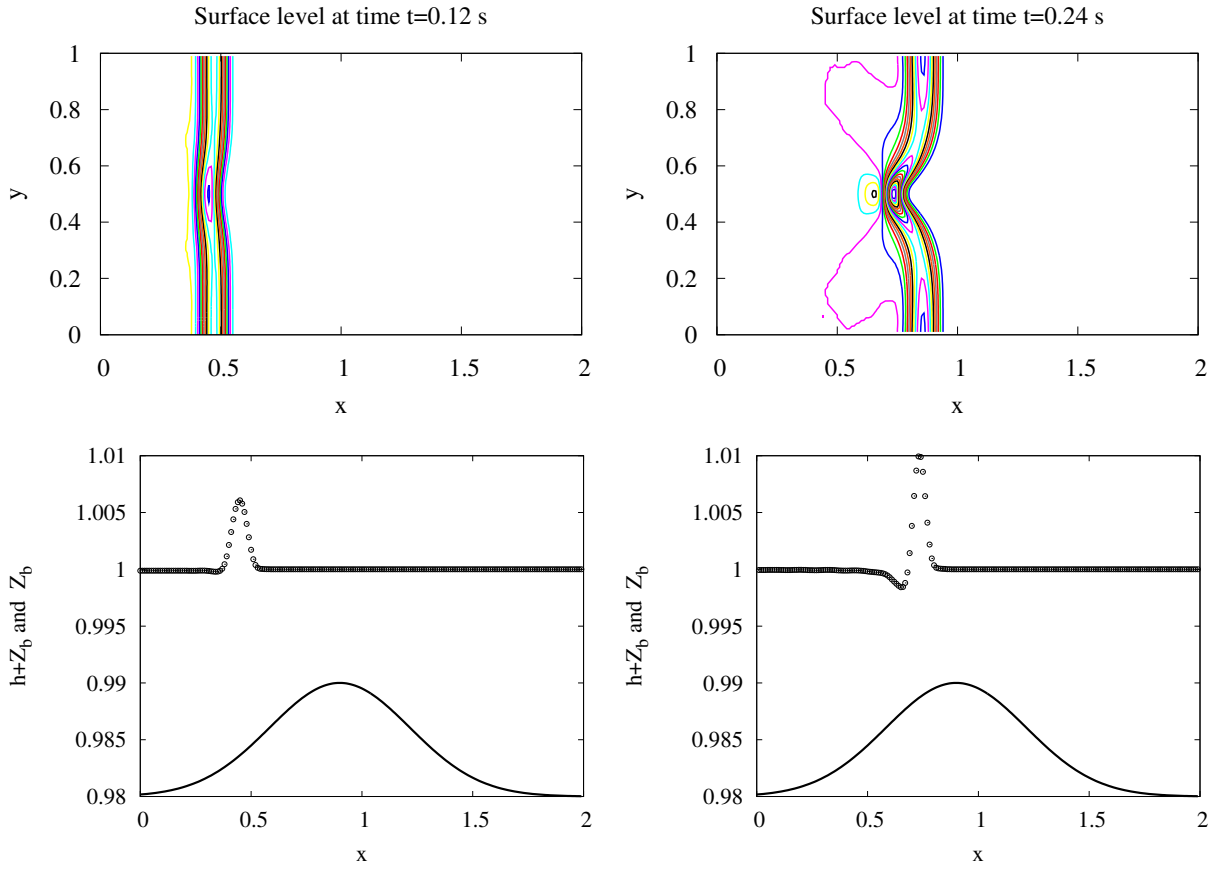

Figure 6: Perturbation of a lake at rest: solution at $t = 0.12$ (left) and $t = 0.24$ (right). Top: contours of the surface level $h + Z_b$ with 30 uniformly spaced contour lines. Bottom: solution along the line $y = 0.5$ ($Z_b = Z_b/80 + 0.98$).

In the region ahead the small perturbation, the exact solution is perfectly preserved by the scheme, as can be seen in Figures 6 and 7. In the region behind the perturbation, the solution quickly gets back to the lake-at-rest, and for completeness we have reported in Figure 8 the total water elevation and the Froude number (28) along the line $y = 0.5$, in the unperturbed region $x \in [0.5, 2]$, at time $t = 0.12$.

These results can be directly compared with those in [22, 24, 20, 17]. We observe that our results reproduce well the interaction and no oscillations are observed. The scheme can resolve consistently this problem.

6. Conclusions

In this paper we have extended the well-balanced central scheme described in [8] to solve the shallow water system in two spatial dimensions. The resulting scheme satisfies the exact C-property and it has been designed to have high-resolution and a non-oscillatory behaviour. Several

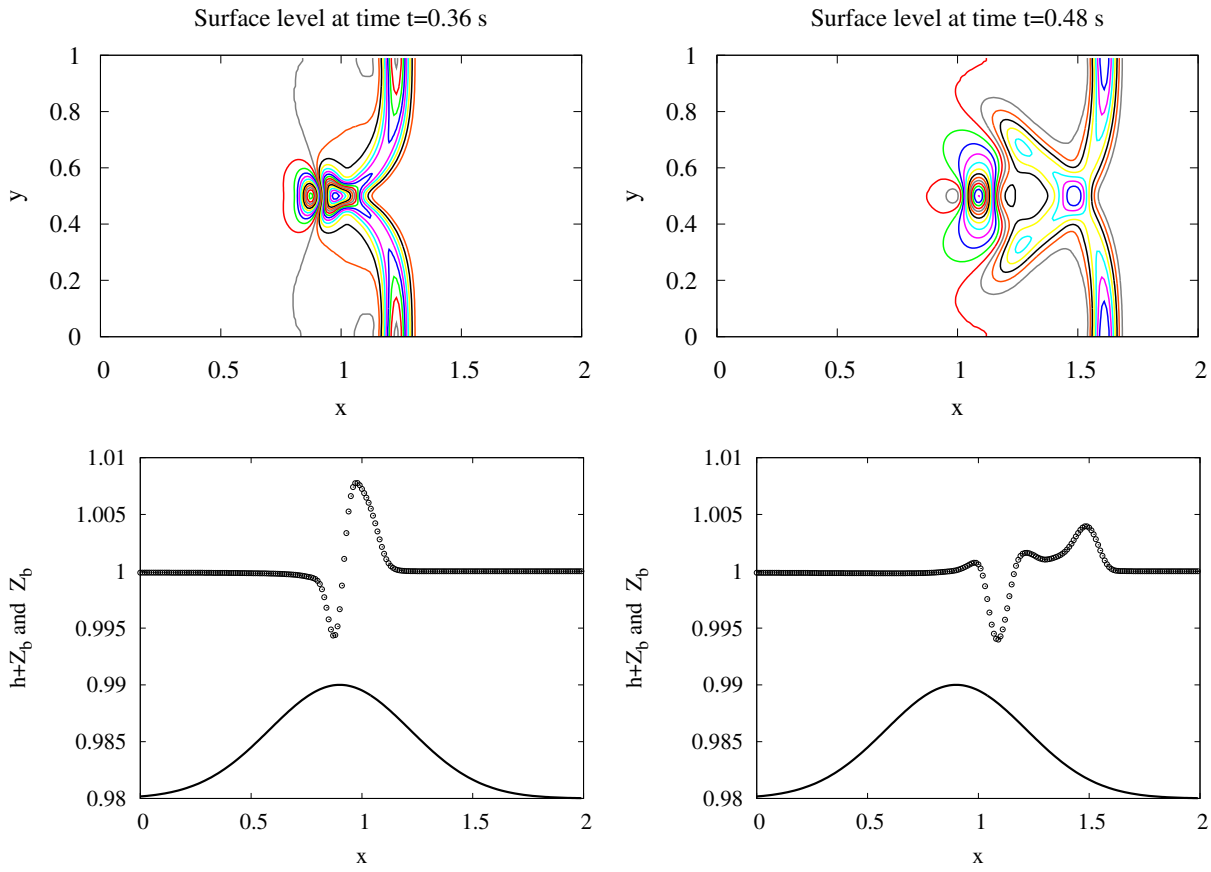

Figure 7: Perturbation of a lake at rest: solution at $t = 0.36$ (left) and $t = 0.48$ (right). Top: contours of the surface level $h + Z_b$ with 30 uniformly spaced contour lines. Bottom: solution along the line $y = 0.5$ ($Z_b = Z_b/80 + 0.98$).

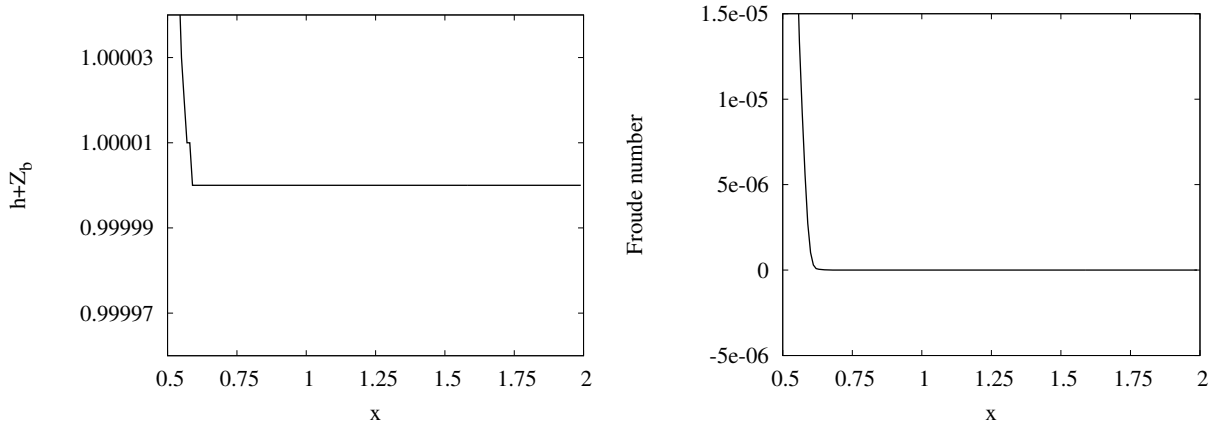

Figure 8: Perturbation of a lake at rest: total surface level (left) and Froude number (right) in the unperturbed region along the line $y = 0.5$, at time $t = 0.12$ s.

2D test problems are used to check the accuracy and non-oscillatory resolution of our numerical method. The exact conservation C-property have been demonstrated and we have tested the order of accuracy of our scheme. We have solved a problem which consists of a transcritical flow over

a flat bottom. Also, a circular dam-break problem and a test with small perturbation of a lake at rest are solved to demonstrate the theoretical properties of the scheme. The capability of the numerical scheme in reproducing the space-time evolution of the variables has been proved, and the agreement between our results and those reported in literature has been confirmed.

Acknowledgements

This work was partially funded by the "Programa de Apoyo a la Investigación y Desarrollo" (PAID-06-10) and (PAID-05-12) of the Universidad Politécnica de Valencia. Angel Balaguer-Beser thanks the support of the Spanish Ministry of Education and Science in the framework of the Projects CGL2009-14220-C02-01 and CGL2010-19591. The authors express their gratitude to the anonymous reviewers for their helpful comments.

References

[1] A. Balaguer and C. Conde, 2005. Fourth-Order non-oscillatory upwind and central schemes for hyperbolic conservation laws. SIAM J. Numer. Anal., 43(2): 455–473.

[2] A. Balaguer-Beser, 2011. A new reconstruction procedure in central schemes for hyperbolic conservation laws. Int. J. Numer. Meth. Engng., 86: 1481–1506.

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

[4] C. Berthon and F. Marche, 2008. A positive preserving high-order VFROE scheme for shallow water equations: a class of relaxation schemes. SIAM J. Sci. Comput., 30(5): 2587–2612.

[5] F. Bianco, G. Puppo and G. Russo, 1999. High-order central schemes for hyperbolic systems of conservation laws. SIAM J. Sci. Comput., 21(1): 294–322.

[6] V. Caleffi, A. Valiani and A. Bernini, 2006. Fourth-order balanced source term treatment in central WENO schemes for shallow water equations. J. Comput. Phys., 218: 228–245.

[7] V. Caleffi and A. Valiani, 2009. Well-balanced bottom discontinuities treatment for high-order shallow water equations WENO scheme. J. Engng. mechanics, 135(7): 684–696.

[8] M. T. Capilla and A. Balaguer-Beser, 2012. A well-balanced high-resolution shape-preserving central scheme to solve one-dimensional sediment transport equations. Adv. Eng. Softw., 50: 19–28.

[9] M. Castro, J. M. Gallardo and C. Parés, 2006. High order finite volume schemes based on reconstruction of states for solving hyperbolic systems with nonconservative products. Applications to shallow water systems. Math. Comput., 75: 1103–1134.

[10] M. J. Castro, E. D. Fernández-Nieto, A. M. Ferreiro, J. A. García-Rodríguez and C. Parés, 2008. High order extensions of Roe schemes for two dimensional nonconservative hyperbolic systems. J. Sci. Comput., 39: 67–114.

[11] J. M. Greenberg and A. Y. Leroux, 1996. A well-balanced scheme for the numerical processing of source terms in hyperbolic equations. SIAM J. Numer. Anal., 33: 1–16.

[12] G. S. Jiang and C. W. Shu, 1996. Efficient implementation of weighted ENO schemes. J. Comput. Phys., 126: 202–228.

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

[14] D. Levy, G. Puppo and G. Russo, 2002. A fourth-order central WENO scheme for multidimensional hyperbolic systems of conservation laws. SIAM J. Sci. Comput., 24: 480–506.

[15] X.D. Liu and E. Tadmor, 1998. Third order nonoscillatory central scheme for hyperbolic conservation laws. Numer. Math., 79, 397–425.

[16] C. Lu, J. Qiu and R. Wang, 2010. A numerical study for the performance of the WENO schemes based on different numerical fluxes for the shallow water equations. J. Comput. Math., 28(6): 807–825.

[17] A. Martínez-Gavara and R. Donat, 2011. A hybrid second order scheme for shallow water flows. J. Sci. Comput., 48: 241–257.

[18] H. Nessyahu and E. Tadmor, 1990. Non-oscillatory central differencing for hyperbolic conservation laws. J. Comput. Phys., 87: 408–463.

[19] L. Pareschi, G. Puppo and G. Russo, 2005. Central Runge-Kutta schemes for conservation laws. SIAM J. Sci. Comput., 26: 979–999.
[20] M. Ricchiuto, R. Abgrall and H. Deconinck, 2007. Application of conservative residual distribution schemes to the solution of the shallow water equations on unstructured meshes. J. Comput. Phys., 222: 287–331.
[21] E. F. Toro, 2001. Shock-capturing methods for free-surface shallow flows. John Wiley & Sons, (2001).
[22] Y. Xing and C. W. Shu, 2005. High order finite difference WENO schemes with the exact conservation property for the shallow water equations. J. Comput. Phys., 208: 206–227.
[23] Y. Xing and C. W. Shu, 2006. High order well-balanced finite volume WENO schemes and discontinuous Galerkin methods for a class of hyperbolic systems with source terms. J. Comput. Phys., 214: 567–598.
[24] Y. Xing and C. W. Shu, 2006. 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., 1(1): 100–134.
[25] Y. Xing and C. W. Shu, 2011. High-order finite volume WENO schemes for the shallow water equations with dry states. Adv. Water Resour., 34: 1026–1038.