

Well-balanced central finite volume methods for the Ripa system

R. Touma^aC. Klingenberg^b

^a*Lebanese American University, Beirut, Lebanon*

^b*Würzburg University, Würzburg, Germany*

This paper is dedicated to the memory of Professor Paul Arminjon

Abstract

We propose a new well-balanced central finite volume scheme for the Ripa system both in one and two space dimensions. The Ripa system is a nonhomogeneous hyperbolic system with a non zero source term that is obtained from the shallow water equations system by incorporating horizontal temperature gradients. The proposed numerical scheme is a second-order accurate finite volume method that evolves a non-oscillatory numerical solution on a single grid, avoids the process of solving Riemann problems arising at the cell interfaces, and follows a well-balanced discretization that ensures the steady state requirement by discretizing the geometrical source term according to the discretization of the flux terms. Furthermore the proposed scheme mimics the surface gradient method and discretizes the water height according to the discretization of the water level. The proposed scheme is then applied and classical one and two-dimensional Ripa problems with flat or variable bottom topographies are successfully solved. The obtained numerical results are in good agreement with corresponding ones appearing in the recent literature, thus confirming the potential and efficiency of the proposed method.

Key words: Well-balanced central schemes; Ripa system; surface gradient method; non-oscillatory and gradients limiting.

1 Introduction

In this paper we consider a system of one- and two- dimensional of shallow water equations with horizontal temperature gradients called the Ripa system, see [8], [17] and [18].

We shall derive a second order accurate numerical method for this system that is able to maintain stationary states. This well-balanced property we found

to be essential even when computing non-stationary solutions. Without the well-balanced property the numerical solutions tend to become unstable also in cases where the underlying pde solution is stable; spurious oscillations and non-physical waves appear .

The discretization of the Ripa system has been studied in two recent previous papers. Chertok, Kurganov and Liu [6] build a central scheme coupled with an interface tracking method. In [10], the authors design a finite volume method that utilizes a new relaxation Riemann solver which is able to well balance the discretization.

Our proposed method is an alternative to these two approaches by using the surface gradient method introduced in [24]. Here the water surface level is chosen as the basis for data reconstruction. This is coupled with an unstaggered central scheme (UCS) previously developed in [21]. The unstaggered central schemes evolve a piecewise linear numerical solution on a single grid and follow the central finite volume schemes developed by [14], [2], [12], [11], and [20] to avoid the resolution of the Riemann problems arising at the cell interfaces thanks to a layer of staggered dual cells used at an intermediate step. A projection follows and generates the numerical solution at the centre of the original cells.

The Ripa system is a variant of the shallow water equations system in which the temperature of the flowing liquid is represented. We note that several numerical methods for the shallow water equations were developed in the recent literature and interested reader is referred to [1], [3], [4], [5], [13], [16], [23], [7], [15], and [21]. The Ripa system was first introduced in [8], [17], and [18] to model ocean currents. Written in its conservative form the two dimensional Ripa system is

$$\mathbf{u}_t + F(\mathbf{u})_x + G(\mathbf{u})_y = S(\mathbf{u}, \mathbf{x}), \quad (1)$$

with

$$\mathbf{u} = \begin{pmatrix} h \\ hu \\ hv \\ h\theta \end{pmatrix}, \quad F(\mathbf{u}) = \begin{pmatrix} hu \\ hu^2 + \frac{g}{2}h^2\theta \\ huv \\ uh\theta \end{pmatrix},$$

$$G(\mathbf{u}) = \begin{pmatrix} hv \\ huv \\ hv^2 + \frac{g}{2}h^2\theta \\ vh\theta \end{pmatrix}, \quad \text{and } S(\mathbf{u}, \mathbf{x}) = \begin{pmatrix} 0 \\ -gh\theta Z_x \\ -gh\theta Z_y \\ 0 \end{pmatrix}, \quad (2)$$

where h denotes the water height, (u, v) is the velocity field, θ is the temperature of the liquid, g is the gravitational constant, and $Z = Z(x, y)$ denotes

the bottom topography function.

The Ripa system is hyperbolic system with real eigenvalues and linearly independent eigenvectors. The Jacobian matrices $\partial F/\partial \mathbf{u}$ and $\partial G/\partial \mathbf{u}$ are

$$\frac{\partial F}{\partial \mathbf{u}} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ \frac{gh\theta}{2} - u^2 & 2u & 0 & \frac{gh}{2} \\ -uv & v & u & 0 \\ -u\theta & \theta & 0 & u \end{pmatrix} \quad \text{and} \quad \frac{\partial G}{\partial \mathbf{u}} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ -uv & v & u & 0 \\ \frac{gh\theta}{2} - v^2 & 0 & 2v & \frac{gh}{2} \\ -v\theta & 0 & \theta & v \end{pmatrix} \quad (3)$$

The eigenvalues of the Jacobian matrix $\partial F/\partial \mathbf{u}$ are $\lambda_1 = u - \sqrt{gh\theta}$, $\lambda_2 = \lambda_3 = u$, and $\lambda_4 = u + \sqrt{gh\theta}$, while the eigenvalues of the Jacobian matrix $\partial G/\partial \mathbf{u}$ are $\mu_1 = v - \sqrt{gh\theta}$, $\mu_2 = \mu_3 = v$, and $\mu_4 = v + \sqrt{gh\theta}$. The eigenvalues are needed to dynamically calculate the time steps of the numerical scheme and to ensure its stability. Note that when $u = \pm\sqrt{gh\theta}$ the Ripa system features a resonance phenomenon (the Jacobian matrix does not have a complete eigensystem), and thus the solution of the Riemann problems becomes very difficult to calculate. For this reason one would like to use numerical methods that avoids the resolution of the Riemann problems such as central schemes and others. Finally, we note that the solution of the Ripa system may be composed of shocks, contacts and rarefaction waves. For additional information on the structure of the solution of the Ripa system, one is referred to [6].

On the other hand, the Ripa system features the following two different steady states

$$\begin{cases} u = v = 0, \\ \theta = \text{constant}, \\ h + Z = \text{constant}, \end{cases} \quad (4)$$

and

$$\begin{cases} u = v = 0, \\ Z = \text{constant}, \\ h^2\theta = \text{constant}. \end{cases} \quad (5)$$

The goal of this paper is to develop a new well-balanced unstaggered central scheme (WB-UCS) for accurately solving one and two-dimensional Ripa systems while maintaining the steady states requirement at the discrete level. Here we note that under appropriate assumptions on the regularity of the solutions of the Ripa system, we can rewrite the Ripa system (2) as a hyperbolic system in terms of the conservative variables h , hu , hv and $h^2\theta$ (instead of $h\theta$

in (2)) as follows.

$$\begin{pmatrix} h \\ hu \\ hv \\ h^2\theta \end{pmatrix}_t + \begin{pmatrix} hu \\ hu^2 + \frac{g}{2}h^2\theta \\ huv \\ uh\theta \end{pmatrix}_x + \begin{pmatrix} hv \\ huv \\ hv^2 + \frac{g}{2}h^2\theta \\ vh\theta \end{pmatrix}_y = \begin{pmatrix} 0 \\ -gh\theta Z_x \\ -gh\theta Z_y \\ -h^2\theta u_x - h^2\theta v_y \end{pmatrix}, \quad (6)$$

As a consequence, any consistent finite volume scheme can easily satisfy the steady state (5); more precisely if the initial data are such that $u = v = 0$, $Z = \text{constant}$ and $h^2\theta = \text{constant}$, then the numerical solution will satisfy the requirements in (5) at any later time.

In this work we mainly focus on the steady state (4) and we develop a well-balanced scheme that satisfies the constraints in (4) at the discrete level both in one space dimension (Theorems (1) and (2)) and two dimensions (Theorems (3) and (4)). As for the steady state (5), we are still exploring/investigating appropriate numerical schemes that ensure the constraints in (5) at the discrete level. Ideally one would like to construct a numerical method that fulfills both equilibrium states (4) and (5) simultaneously. This indeed invites future research on this topic.

The rest of the paper is structured as follows: in section 2, the central finite volume scheme is developed coupling it with the surface gradient method. In section 3, one- and two-dimensional experiments are presented, showing excellent results. In the appendix proofs of our theorems are given.

2 Well-balanced central schemes for the Ripa system

In this section, we develop a new well-balanced central finite volume scheme for the Ripa system (1)-(2) in one and two space dimensions. We follow a classical central finite volume construction for hyperbolic systems of conservation laws with a source term. In order to satisfy the steady state requirement (4), we extend the surface gradient method to the case of the Ripa system and show that the steady state conditions in (4) are exactly maintained at the discrete level.

2.1 One-dimensional well-balanced central scheme for the Ripa system

We consider first the one-dimensional Ripa system written in its conservative form

$$\begin{cases} \partial_t \mathbf{u} + \partial_x f(\mathbf{u}) = S(\mathbf{u}, x), & t > 0, x \in \Omega \subset \mathbb{R} \\ u(x, 0) = u_0(x) \end{cases} \quad (7)$$

with $\mathbf{u}(x, t) = (h, hu, h\theta)$, $f(\mathbf{u}) = (hu, hu^2 + \frac{1}{2}gh^2, h\theta)$, and $S(\mathbf{u}, x) = (0, -gh\theta dz/dx, 0)$. Here $h(x, t)$ denotes the water height, $u(x, t)$ is the velocity in the x -direction, g is the gravitational constant, and $z(x)$ denotes the bottom topography function.

The computational domain Ω is first partitioned using the control cells $C_i = [x_{i-1/2}, x_{i+1/2}]$ centered at the nodes x_i . Without any loss of generality we assume that the numerical solution \mathbf{u}_i^n is known at time t^n and is defined at cell-centers x_i . A piecewise linear reconstruction of the solution values is then performed and the exact solution $\mathbf{u}(x, t)$ to system (7) is approximated at time t^n on the cells C_i by

$$\mathbf{u}(x, t^n) \approx \mathcal{L}_i(x, t^n) = \mathbf{u}_i^n + (x - x_i)\delta_i^n \quad \forall x \in C_i, \quad (8)$$

where δ_i^n approximates $\partial \mathbf{u}(x_i, t^n)/\partial x$ to a first order of accuracy. To avoid the process of solving the Riemann problems at the cell interfaces, the numerical solution to system (7) will be first calculated at time $t^{n+1} = t^n + \Delta t$ on the dual cells $D_{i+1/2} = [x_i, x_{i+1}]$. Then a back projection step will be applied to retrieve \mathbf{u}_i^{n+1} . We integrate system (7) over the domain $R_{i+1/2}^n = D_{i+1/2} \times [t^n, t^{n+1}]$ and then we apply Green's theorem to the left-hand side; taking into account that $u(x, t) \approx \mathcal{L}(x, t)$, we obtain

$$\begin{aligned} \mathbf{u}_{i+\frac{1}{2}}^{n+1} = \mathbf{u}_{i+\frac{1}{2}}^n - \frac{1}{\Delta x} \left[\int_{t^n}^{t^{n+1}} f(\mathbf{u}(x_{i+1}, t)) dt - \int_{t^n}^{t^{n+1}} f(\mathbf{u}(x_i, t)) dt \right] \\ + \frac{1}{\Delta x} \int_{t^n}^{t^{n+1}} \int_{x_i}^{x_{i+1}} S(\mathbf{u}, x) dx dt. \end{aligned} \quad (9)$$

Here $\mathbf{u}_{i+1/2}^n$ is the projected solution at time t^n on the dual cells $D_{i+1/2}$ and is obtained with the aid of a Taylor series expansion as follows

$$\mathbf{u}_{i+\frac{1}{2}}^n = \frac{1}{2} (\mathbf{u}_i^n + \mathbf{u}_{i+1}^n) + \frac{\Delta x}{8} (\delta_i^n - \delta_{i+1}^n) \quad (10)$$

The flux integrals in equation (9) are approximated with second-order of accuracy using the midpoint quadrature rule leading to

$$\begin{aligned} \mathbf{u}_{i+\frac{1}{2}}^{n+1} = \mathbf{u}_{i+1/2}^n - \frac{\Delta t}{\Delta x} \left[f(\mathbf{u}_{i+1}^{n+\frac{1}{2}}) - f(\mathbf{u}_i^{n+\frac{1}{2}}) \right] \\ + \frac{1}{\Delta x} \int_{t^n}^{t^{n+1}} \int_{x_i}^{x_{i+1}} S(\mathbf{u}(x, t), x) dx dt, \end{aligned} \quad (11)$$

where the required predicted values at the intermediate time $t^{n+1/2}$ are obtained using a first-order Taylor expansion in time and the balance law (7) as follows

$$\mathbf{u}_i^{n+1/2} = \mathbf{u}_i^n + \frac{\Delta t}{2\Delta x} \left(-f'_i + \Delta x S_i^n \right). \quad (12)$$

The term S_i^n discretizes the source term in system (7) and should be carefully discretized in order to ensure well-balancing as we will see later in this section.

Next we discretize the integral of the source term using centered differences and the midpoint quadrature rule, we obtain:

$$\begin{aligned} \int_{t^n}^{t^{n+1}} \int_{x_i}^{x_{i+1}} S(\mathbf{u}(x, t), x) dx dt &\approx S(\mathbf{u}_i^{n+1/2}, \mathbf{u}_{i+1}^{n+1/2}) \\ &= \Delta t \Delta x \begin{pmatrix} 0 \\ g \frac{h_i^{n+1/2} + h_{i+1}^{n+1/2}}{2} \frac{\theta_i^{n+1/2} + \theta_{i+1}^{n+1/2}}{2} \left(-\frac{z_{i+1} - z_i}{\Delta x} \right) \\ 0 \end{pmatrix}. \end{aligned} \quad (13)$$

The numerical solution at time t^{n+1} is obtained by substituting equation (13) in equation (11), and the computed values will be defined as the centers $x_{i+1/2}$ of the dual cells $D_{i+1/2}$; a back-projection step recovers the numerical solution on the original cells as follows

$$\mathbf{u}_i^{n+1} = \frac{1}{2} \left(\mathbf{u}_{i-\frac{1}{2}}^{n+1} + \mathbf{u}_{i+\frac{1}{2}}^{n+1} \right) + \frac{\Delta x}{8} \left((\mathbf{u}_{i-\frac{1}{2}}^{n+1})' - (\mathbf{u}_{i+\frac{1}{2}}^{n+1})' \right). \quad (14)$$

To ensure the constant-temperature lake at rest steady state (4) property of the Ripa system one should pay special attention to the way the predicted values are computed at time $t^{n+1/2}$ in equation (12), as well as to the back and forth projection steps in equations (10) and (14). Note that in the steady state case ($h + Z = \text{constant}$, $u = 0$, and $\theta = \text{constant}$) the Ripa system reduces to:

$$\partial_t \begin{pmatrix} h \\ 0 \\ h\theta \end{pmatrix} + \partial_x \begin{pmatrix} 0 \\ \frac{g}{2} h^2 \theta \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ -gh\theta Z_x \\ 0 \end{pmatrix}, \quad (15)$$

and therefore specific discretizations of the momentum's flux component and its corresponding source term component are required in order to ensure well-balancing in the steady state case. In this work we discretize the source term in equation (12) as follows:

$$S_i^n = S_{i,L}^n + S_{i,R}^n + S_{i,C}^n \quad (16)$$

with

$$\begin{aligned}
S_{i,L}^n &= \sigma_i^2 \frac{1 - \sigma_i}{6} (2 - \sigma_i) \begin{pmatrix} 0 \\ -gh_i^n \theta_i^n \Theta \frac{z_i - z_{i-1}}{\Delta x} \\ 0 \end{pmatrix}, \\
S_{i,R}^n &= \sigma_i^2 \frac{1 + \sigma_i}{2} (2 - \sigma_i) \begin{pmatrix} 0 \\ -gh_i^n \theta_i^n \Theta \frac{z_{i+1} - z_i}{\Delta x} \\ 0 \end{pmatrix}, \\
S_{i,C}^n &= \sigma_i \frac{(\sigma_i + 1)(\sigma_i - 1)}{6} \begin{pmatrix} 0 \\ -gh_i^n \theta_i^n \Theta \frac{z_{i+1} - z_{i-1}}{2\Delta x} \\ 0 \end{pmatrix}. \quad (17)
\end{aligned}$$

The parameter σ_i appearing in the discretized source term in equation (17) is a sensor function that forces the discretization of dZ/dx in $S(\mathbf{u}, x)$ to follow the discretization of $\partial h/\partial x$ and is defined as follows:

$$\sigma_i = \begin{cases} -1 & \text{if } h'_i = \Theta \frac{h_i^n - h_{i-1}^n}{\Delta x}, \\ 1 & \text{if } h'_i = \Theta \frac{h_{i+1}^n - h_i^n}{\Delta x}, \\ 0 & \text{if } h'_i = 0, \\ 2 & \text{if } h'_i = \frac{h_{i+1}^n - h_{i-1}^n}{2\Delta x}. \end{cases}$$

where $1 \leq \Theta \leq 2$ is the parameter of the MC- Θ limiter [22]. This leads us to a first main result.

Theorem 1 *In the context of the one-dimensional Ripa system (7), if at a given time t^n , the numerical solution is such that $h_i^n + z_i = \text{constant}$, $u_i^n = 0$, and $\theta_i^n = \text{constant}$, for all i , then the equations (12), (16), and (17) lead to $\mathbf{u}_i^{n+\frac{1}{2}} = \mathbf{u}_i^n$ while equations (11) and (13) lead to $\mathbf{u}_{i+\frac{1}{2}}^{n+1} = \mathbf{u}_{i+\frac{1}{2}}^n$.*

PROOF. Highlights of the proof of theorem 1 are given in the appendix.

Remark

Theorem 1 states that the updated numerical solution $n_{i+1/2}^{n+1}$ obtained at time t^{n+1} shares the same properties of the forward projected solution $u_{i+1/2}^n$ obtained at time t^n on the cells $D_{i+1/2}$, but its projection u_i^{n+1} doesn't necessarily satisfy the steady state requirement on the cells C_i unless a special care of the

forward projection step and the backward projection step is taken into account.

In general, the central scheme fails to satisfy the steady state requirement when it is used to solve steady state Ripa systems problems. This is due to the fact that the riverbed function and the water level function are both linear on the original cells C_i , but not on the staggered cells $D_{i+1/2}$. An additional treatment of the scheme is needed to remedy this situation. In this work we extend the surface gradient method, previously adapted to the shallow water equations, to the case of the Ripa systems and calculate the numerical derivative of the water height $h(x, t)$ component in terms of the water level function $H(x, t) = h(x, t) + Z(x)$. We also note that the $h\theta$ component of the solution should also be projected forward and backward using the water level component by following the surface gradient method as is described below.

We first assume that the bottom topography function is defined at the cell interfaces (i.e., $z_{i+\frac{1}{2}}$ is given at $x_{i+\frac{1}{2}}$). The cell centered bottom elevation function values are then obtained using the equation $z_i = \frac{1}{2}(z_{i+\frac{1}{2}} + z_{i-\frac{1}{2}})$, and the linear interpolants $z(x)$ of the bottom topography function on the cells C_i are given by:

$$z(x) = z_i + \frac{1}{\Delta x}(z_{i+\frac{1}{2}} - z_{i-\frac{1}{2}})(x - x_i), \forall x \in C_i.$$

The linearization of the water height $h(x)$ inside each cell will be made indirectly by first linearizing the water level $H(x) = h(x) + z(x)$, and then by using the relation $h(x) = H(x) - z(x)$.

The linearization $H(x) = H_i + H'_i(x - x_i)$ on the cells C_i is obtained by using a limiting procedure of the numerical derivatives of $H_i = h_i + z_i$; the linearization of the water height $h(x)$ is finally obtained by calculating h'_i as follows:

$$(h_i^n)' = (H_i^n)' - \frac{1}{\Delta x}(z_{i+1/2} - z_{i-1/2}). \quad (18)$$

Similarly, in the forward projection step the linearization of the $h\theta$ component is made indirectly in terms of H using equation (18) as follows:

$$(h_i^n \theta_i^n)' = \left((H_i^n)' - \frac{1}{\Delta x}(z_{i+1/2} - z_{i-1/2}) \right) \theta_i^n + h_i^n (\theta_i^n)'. \quad (19)$$

Likewise, for the back projection step of the computed numerical solution $\mathbf{u}_{i+1/2}^{n+1}$ back onto the original cells C_i , we proceed again using the surface gradient method. We define the water level $H_{i+1/2}$ on the staggered nodes as follows:

$$\tilde{H}_{i+1/2}^{n+1} = h_{i+1/2}^{n+1} + \tilde{z}_{i+1/2} \quad (20)$$

where $\tilde{z}_{i+1/2} = z_{i+1/2} - \frac{1}{2}(z_{i+1/2} - (z_i + z_{i+1})/2)$ is the corrected bottom value due to the fact that the bottom function $z(x)$ is not linear inside the staggered

cell $D_{i+1/2}$.

Next, we obtain the limited discrete derivatives $(H_{i+1/2}^{n+1})'$ from the staggered water level values $\tilde{H}_{i+1/2}^{n+1}$ and obtain $(h_{i+1/2}^{n+1})'$ as follows:

$$(h_{i+1/2}^{n+1})' = (\tilde{H}_{i+1/2}^{n+1})' - \frac{1}{\Delta x}(z_{i+1} - z_i). \quad (21)$$

Similarly for the back-projection of the $h\theta$ component in $\mathbf{u}_{i+1/2}^{n+1}$, the linearization is performed in terms of $(\tilde{H}_{i+1/2}^{n+1})'$ and using equation (21) as follows

$$(h_{i+1/2}^{n+1}\theta_{i+1/2}^{n+1})' = \left((\tilde{H}_{i+1/2}^{n+1})' - \frac{1}{\Delta x}(z_{i+1} - z_i) \right) \theta_{i+1/2}^{n+1} + h_{i+1/2}^{n+1}(\theta_{i+1/2}^{n+1})'. \quad (22)$$

Theorem 2 *In the context of the one-dimensional Ripa system (7) and the central scheme (11) along with the prediction step (12), the projection steps (10), (18), (19), and the back projection steps (14), (21), (22), and if at a given time t^n the numerical solution \mathbf{u}_i^n satisfies the lake at rest with a constant temperature steady state (4), (i.e. $h_i^n + z_i = \text{constant}$, $u_i^n = 0$, and $\theta_i^n = \text{constant}$ for all i), then the numerical solution generated using the developed central scheme exactly satisfies the steady state 4 at the discrete level and $\mathbf{u}_i^{n+1} = \mathbf{u}_i^n$ holds for all i and all $n = 0, 1, \dots$.*

PROOF. Highlights of the proof of theorem 2 are given in the appendix of the paper.

Remark 1 *We finally note that the proposed well-balanced numerical scheme follows the same stability condition as the original NT scheme [14], since the numerical integration of the source term as well as the forward and backward projection steps of h and $h\theta$ components according to the surface gradient method don't add any restriction on the stability requirement of the numerical base scheme. The numerical results presented in section 4 are performed with a CFL number of magnitude 0.485, and the time step Δt is calculated dynamically in terms of the eigenvalues of the one-dimensional Ripa system $\lambda_i, i = 1, \dots, 3$ and the spacial step Δx according to the formula $\max(|\lambda_i|) \frac{\Delta t}{\Delta x} \leq 0.485$ for $i \in \{1, \dots, 3\}$. For further information regarding the stability of central schemes, one is referred to [14], [11].*

2.2 Two-dimensional well-balanced central scheme for the Ripa system

In this section we extend the one-dimensional well-balanced unstaggered central scheme to the case of the two-dimensional Ripa system with a variable

bottom topography function.

We consider the two-dimensional Ripa system (1)

$$\partial_t \mathbf{u} + \partial_x f(\mathbf{u}) + \partial_y g(\mathbf{u}) = S(\mathbf{u}, \mathbf{x}) \quad (23)$$

where \mathbf{u} , $f(\mathbf{u})$, $g(\mathbf{u})$, and $S(\mathbf{u}, \mathbf{x})$ are defined in system (2). We assume that the computational domain $\Omega \subset \mathbb{R}^2$ is uniformly discretized using the Cartesian cells $C_{i,j} = [x_{i-1/2}, x_{i+1/2}] \times [y_{j-1/2}, y_{j+1/2}]$ centered at the nodes (x_i, y_j) , and we define the dual cells $D_{i+1/2, j+1/2}$ to be the rectangles $[x_i, x_{i+1}] \times [y_j, y_{j+1}]$.

We assume further that the initial condition is defined at the centers of the cells C_{ij} . Without any loss of generality we assume that the numerical solution to the two-dimensional Ripa system is known at time t^n and is also defined at the center of the cells C_{ij} . We start by constructing the linear interpolants

$$L_{i,j}(x, y, t^n) = \mathbf{u}_{i,j}^n + (x - x_i)\delta_i^n + (y - y_j)\sigma_j^n \approx \mathbf{u}(x, y, t^n), \quad \forall (x, y) \in C_{i,j}, \quad (24)$$

that approximate the solution to system (23) on the cells C_{ij} , where (δ_i^n, σ_j^n) denotes a limited numerical gradient of the numerical solution $\mathbf{u}_{i,j}^n$. We then integrate equation (23) on the rectangular box $R_{i+1/2, j+1/2}^n = D_{i+1/2, j+1/2} \times [t^n, t^{n+1}]$ and we apply Green's theorem to the integral to the left-hand side to obtain

$$\int_{R_{i+1/2, j+1/2}^n} (\partial_t \mathbf{u} + \partial_x f(\mathbf{u}) + \partial_y g(\mathbf{u})) \, dV = \int_{R_{i+1/2, j+1/2}^n} (S(\mathbf{u}, x, y)) \, dV. \quad (25)$$

Just like the one-dimensional case, the staggering process will avoid solving Riemann problems, but the obtained solution at time t^{n+1} will be defined on the dual cells $D_{i+1/2, j+1/2}$. A back-projection step will be necessary to retrieve the solution values at centers of original cells $C_{i,j}$. Gradients limiting avoids spurious oscillations and Euler's midpoint quadrature rule ensures second-order of accuracy provided the integral of the source term is properly approximated. Furthermore, in order to ensure well-balancing and the steady-state requirement at the discrete level, the source term should be discretized according to the discretization of the flux terms. Taking into account that the numerical solution is piecewise linear over the dual cells $D_{i+1/2, j+1/2}$ and is defined at their centers $(x_{i+1/2}, y_{j+1/2})$, equation (25) reduces to

$$\begin{aligned} \mathbf{u}_{i+1/2, j+1/2}^{n+1} &= \mathbf{u}_{i+1/2, j+1/2}^n \\ &\quad - \frac{\Delta t}{2\Delta x} \left[(f_{i+1, j}^{n+1/2} - f_{i, j}^{n+1/2}) + (f_{i+1, j+1}^{n+1/2} - f_{i, j+1}^{n+1/2}) \right] \\ &\quad - \frac{\Delta t}{2\Delta y} \left[(g_{i, j+1}^{n+1/2} - g_{i, j}^{n+1/2}) + (g_{i+1, j+1}^{n+1/2} - g_{i+1, j}^{n+1/2}) \right] \\ &\quad + \frac{\Delta t}{\Delta x \Delta y} S(\mathbf{u}_{i, j}^{n+1/2}, \mathbf{u}_{i+1, j}^{n+1/2}, \mathbf{u}_{i, j+1}^{n+1/2}, \mathbf{u}_{i+1, j+1}^{n+1/2}) \end{aligned} \quad (26)$$

where $\mathbf{u}_{i+1/2,j+1/2}^n$ is the projected solution at time t^n on the staggered dual cells $D_{i+1/2,j+1/2}$, and is evaluated using a Taylor expansion in space as follows

$$\begin{aligned} \mathbf{u}_{i+1/2,j+1/2}^n &= \frac{1}{4} \left(\mathbf{u}_{i,j}^n + \mathbf{u}_{i+1,j}^n + \mathbf{u}_{i,j+1}^n + \mathbf{u}_{i+1,j+1}^n \right) \\ &\quad + \frac{1}{16} (\delta_{i,j} + \delta_{i,j+1} - \delta_{i+1,j} - \delta_{i+1,j+1}) \\ &\quad + \frac{1}{16} (\sigma_{i,j} - \sigma_{i,j+1} + \sigma_{i+1,j} - \sigma_{i+1,j+1}). \end{aligned} \quad (27)$$

The flux integral with respect to time is approximated to second order of accuracy using the midpoint quadrature rule and the term $f_{i,j}^{n+1/2} = f(\mathbf{u}_{i,j}^{n+1/2})$ is obtained using a prediction step at an intermediate time $t^{n+1/2}$. The solution at time t^{n+1} on the cells C_{ij} of the original grid is then obtained using a back projection step as follows:

$$\begin{aligned} \mathbf{u}_{i,j}^{n+1} &= \frac{1}{4} \left(\mathbf{u}_{i-\frac{1}{2},j-\frac{1}{2}}^{n+1} + \mathbf{u}_{i+\frac{1}{2},j-\frac{1}{2}}^{n+1} + \mathbf{u}_{i-\frac{1}{2},j+\frac{1}{2}}^{n+1} + \mathbf{u}_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} \right) \\ &\quad + \frac{1}{16} \left(\delta_{i-\frac{1}{2},j-\frac{1}{2}} + \delta_{i-\frac{1}{2},j+\frac{1}{2}} - \delta_{i+\frac{1}{2},j-\frac{1}{2}} - \delta_{i+\frac{1}{2},j+\frac{1}{2}} \right) \\ &\quad + \frac{1}{16} \left(\sigma_{i-\frac{1}{2},j-\frac{1}{2}} + \sigma_{i-\frac{1}{2},j+\frac{1}{2}} - \sigma_{i+\frac{1}{2},j-\frac{1}{2}} - \sigma_{i+\frac{1}{2},j+\frac{1}{2}} \right) \end{aligned} \quad (28)$$

where $(\delta, \sigma)_{i+1/2,j+1/2}$ denotes a limited numerical gradient of the numerical solution obtained at time t^{n+1} at the point $(x_{i+1/2}, y_{j+1/2})$. For a further detailed description of the unstaggered central scheme for homogeneous hyperbolic systems one is referred to [19].

The term $\Delta t S(\mathbf{u}_{i,j}^{n+1/2}, \mathbf{u}_{i+1,j}^{n+1/2}, \mathbf{u}_{i,j+1}^{n+1/2}, \mathbf{u}_{i+1,j+1}^{n+1/2})$ in equation (26) is used to approximate the spatial integral of the source term over the domain $R_{i+1/2,j+1/2}^n$ with a second-order of accuracy. In the case of the Ripa system, the spatial integral of the source term is discretized using centered differences and the midpoint quadrature rule as follows:

$$\begin{aligned}
S \left(\mathbf{u}_{i,j}^{n+\frac{1}{2}}, \mathbf{u}_{i+1,j}^{n+\frac{1}{2}}, \mathbf{u}_{i,j+1}^{n+\frac{1}{2}}, \mathbf{u}_{i+1,j+1}^{n+\frac{1}{2}} \right) = & \\
& \frac{\Delta x \Delta y}{2} \begin{pmatrix} 0 \\ -g \frac{(h\theta)_{i+1,j}^{n+\frac{1}{2}} + (h\theta)_{i,j}^{n+\frac{1}{2}}}{2} \frac{z_{i+1,j} - z_{i,j}}{\Delta x} \\ -g \frac{(h\theta)_{i,j+1}^{n+\frac{1}{2}} + (h\theta)_{i,j}^{n+\frac{1}{2}}}{2} \frac{z_{i,j+1} - z_{i,j}}{\Delta y} \\ 0 \end{pmatrix} \\
& + \frac{\Delta x \Delta y}{2} \begin{pmatrix} 0 \\ -g \frac{(h\theta)_{i+1,j+1}^{n+\frac{1}{2}} + (h\theta)_{i,j+1}^{n+\frac{1}{2}}}{2} \frac{z_{i+1,j+1} - z_{i,j+1}}{\Delta x} \\ -g \frac{(h\theta)_{i+1,j+1}^{n+\frac{1}{2}} + (h\theta)_{i+1,j}^{n+\frac{1}{2}}}{2} \frac{z_{i+1,j+1} - z_{i+1,j}}{\Delta y} \\ 0 \end{pmatrix}. \quad (29)
\end{aligned}$$

The predicted values $\mathbf{u}_{ij}^{n+1/2}$ at the intermediate time step $t^{n+1/2}$ in equations (26) and (29) are estimated using a first-order Taylor expansion in time and the RIPA system (23) as follows:

$$\mathbf{u}_{i,j}^{n+\frac{1}{2}} = \mathbf{u}_{i,j}^n + \frac{\Delta t}{2} \left(-\frac{f_{i,j}^x}{\Delta x} - \frac{g_{i,j}^y}{\Delta y} + S_{i,j}^n \right), \quad (30)$$

where $\frac{f_{i,j}^x}{\Delta x}$ is a limited numerical partial derivative of the flux function $f(\mathbf{u})$ with respect to the x variable, and is obtained using the Jacobian matrix $\partial f / \partial \mathbf{u}$. The term S_{ij}^n in the predictor step (30) discretizes the source term (in equation (23) at time t^n on the cells $C_{i,j}$ as follows

$$S_{i,j}^n = \begin{pmatrix} 0 \\ -g(h\theta)_{i,j}^n \frac{\partial z}{\partial x} \Big|_{ij} \\ -g(h\theta)_{i,j}^n \frac{\partial z}{\partial y} \Big|_{ij} \\ 0 \end{pmatrix} = \begin{pmatrix} S_1 = 0 \\ S_2 \\ S_3 \\ S_4 = 0 \end{pmatrix}. \quad (31)$$

The terms S_2 and S_3 in equation (31) are used to discretize the gradient of the bottom topography function according to the discretization of the water

height gradient, and are obtained with the aid of sensor functions as follows:

$$S_2 = \begin{cases} -g(h\theta)_{i,j}^n \Theta \frac{z_{i,j} - z_{i-1,j}}{\Delta x}, & \text{if } \sigma_2 = -1, \\ 0 & \text{if } \sigma_2 = 0, \\ -g(h\theta)_{i,j}^n \Theta \frac{z_{i+1,j} - z_{i,j}}{\Delta x}, & \text{if } \sigma_2 = 1, \\ -g(h\theta)_{i,j}^n \frac{z_{i+1,j} - z_{i-1,j}}{2\Delta x}, & \text{if } \sigma_2 = 2. \end{cases}$$

$$\text{and } S_3 = \begin{cases} -g(h\theta)_{i,j}^n \Theta \frac{z_{i,j} - z_{i,j-1}}{\Delta y}, & \text{if } \sigma_3 = -1, \\ 0 & \text{if } \sigma_3 = 0, \\ -g(h\theta)_{i,j}^n \Theta \frac{z_{i,j+1} - z_{i,j}}{\Delta y}, & \text{if } \sigma_3 = 1, \\ -g(h\theta)_{i,j}^n \frac{z_{i,j+1} - z_{i,j-1}}{2\Delta y}, & \text{if } \sigma_3 = 2. \end{cases} \quad (32)$$

The parameters σ_2 and σ_3 in equation (32) are two sensor parameters that direct the discretization of $\frac{\partial z}{\partial x}|_{ij}$ and $\frac{\partial z}{\partial y}|_{ij}$ according to the discretizations of $\partial_x h|_{ij}$ and $\partial_y h|_{ij}$, respectively. They are defined as follows:

$$\sigma_2 = \begin{cases} -1, & \text{if } \partial_x h|_{ij} \approx \Theta (h_{i,j}^n - h_{i-1,j}^n) / \Delta x, \\ 0, & \text{if } \partial_x h|_{ij} \approx 0, \\ 1, & \text{if } \partial_x h|_{ij} \approx \Theta (h_{i+1,j}^n - h_{i,j}^n) / \Delta x, \\ 2, & \text{if } \partial_x h|_{ij} \approx (h_{i+1,j}^n - h_{i-1,j}^n) / (2\Delta x). \end{cases}$$

$$\text{and } \sigma_3 = \begin{cases} -1, & \text{if } \partial_y h|_{ij} \approx \Theta (h_{i,j}^n - h_{i,j-1}^n) / \Delta y, \\ 0, & \text{if } \partial_y h|_{ij} \approx 0, \\ 1, & \text{if } \partial_y h|_{ij} \approx \Theta (h_{i,j+1}^n - h_{i,j}^n) / \Delta y, \\ 2, & \text{if } \partial_y h|_{ij} \approx (h_{i,j+1}^n - h_{i,j-1}^n) / (2\Delta y). \end{cases} \quad (33)$$

The parameter $1 \leq \Theta \leq 2$ appearing in the formulae for S_2 and S_3 is the MC- Θ limiter parameter.

Theorem 3 *In the context of the two-dimensional Ripa system (1)-(2), the central finite volume method (26), and the directed discretization formulas of the source term (29), (31), (32), and (33), if the steady state (4) is satisfied at the discrete level at time t^n (i.e., $h_{i,j}^n + z_{i,j} = \text{constant}$, $u_{i,j}^n = v_{i,j}^n = 0$, and $\theta_{i,j}^n = \text{constant}$), then the updated numerical solution verifies the equations $\mathbf{u}_{i,j}^{n+1/2} = \mathbf{u}_{i,j}^n$ and $\mathbf{u}_{i+1/2,j+1/2}^{n+1} = \mathbf{u}_{i+1/2,j+1/2}^n$ for all i, j .*

PROOF. The proof of theorem (3) can be constructed in a similar way the

the proof of theorem (1); highlights are presented in the appendix section.

Remark

Theorem 3 states that if the numerical solution at time t^n corresponds to the case of an equilibrium state, then the updated solution at time t^{n+1} is such that $\mathbf{u}_{i+1/2,j+1/2}^{n+1} = \mathbf{u}_{i+1/2,j+1/2}^n$, but the equality $\mathbf{u}_{i,j}^{n+1} = \mathbf{u}_{i,j}^n$ doesn't necessarily holds. The forward and backward projection steps (27) and (28) need to be adjusted according to the surface gradient method in order to ensure the steady state requirement. In the steady state case (5) the two-dimensional Ripa system becomes

$$\partial_t \begin{pmatrix} h \\ 0 \\ 0 \\ h\theta \end{pmatrix} + \frac{\partial}{\partial x} \begin{pmatrix} 0 \\ \frac{g}{2}h^2\theta \\ 0 \\ 0 \end{pmatrix} + \frac{\partial}{\partial y} \begin{pmatrix} 0 \\ 0 \\ \frac{g}{2}h^2\theta \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ -gh\theta Z_x \\ -gh\theta Z_y \\ 0 \end{pmatrix}, \quad (34)$$

and one can see that the first and fourth components (h and $h\theta$) of the projected solutions need to be carefully computed. As in the one-dimensional case, we assume that the bottom function is defined at the corners $(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}})$ of the original cells $C_{i,j}$ and we set the bottom topography function value at the cell centers to be

$$z_{i,j} = \frac{z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}}{4} \quad (35)$$

In the forward projection step (27) we linearize the water height h in the first and fourth components of the numerical solution $\mathbf{u}_{i,j}^n$ in terms of the water level $H_{i,j}^n = h_{i,j}^n + z_{i,j}$ by first constructing the linear interpolants for the water levels on each control cell $C_{i,j}$ as follows

$$H(x, y, t^n) = H_{i,j}^n + (x - x_i)(H_x)_{i,j}^n + (y - y_j)(H_y)_{i,j}^n, \quad \forall (x, y) \in C_{i,j}, \quad (36)$$

where $((H_x)_{i,j}^n, (H_y)_{i,j}^n)$ denotes a limited numerical gradient of the water level function $H_{i,j}^n$. The numerical gradient of the water height function $h(x, y, t^n)$ can now be calculated using the equations

$$(h_x)_{i,j}^n = (H_x)_{i,j}^n - (z_x)_{i,j} \quad \text{and} \quad (h_y)_{i,j}^n = (H_y)_{i,j}^n - (z_y)_{i,j}. \quad (37)$$

The gradient of the bottom topography function is discretized using centered differences as follows:

$$(z_x)_{i,j} = \left(\frac{z_{i+1/2,j-1/2} + z_{i+1/2,j+1/2}}{2} - \frac{z_{i-1/2,j-1/2} + z_{i-1/2,j+1/2}}{2} \right) / \Delta x \quad (38)$$

$$(z_y)_{i,j} = \left(\frac{z_{i-1/2,j+1/2} + z_{i+1/2,j+1/2}}{2} - \frac{z_{i-1/2,j-1/2} + z_{i+1/2,j-1/2}}{2} \right) / \Delta y \quad (39)$$

Equations (38) and (39) are used in the forward projection step (27) of the first and fourth components of $\mathbf{u}_{i,j}^n$ only.

As for the back projection step (28), we follow a similar procedure and linearize the water height in terms of the water level function. As in the one-dimensional case, we first correct the water bed function at the center of the cells $D_{i+1/2,j+1/2}$ as follows

$$\begin{aligned} \tilde{z}_{i+\frac{1}{2},j+\frac{1}{2}} &= z_{i+\frac{1}{2},j+\frac{1}{2}} \\ &- \frac{1}{2} \left(z_{i+\frac{1}{2},j+\frac{1}{2}} - \frac{z_{i+1/2,j+3/2} + z_{i+1/2,j-1/2} + z_{i-1/2,j+1/2} + z_{i+3/2,j+1/2}}{4} \right). \end{aligned} \quad (40)$$

Next, we calculate the water level $\widetilde{H}_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1}$ using corrected waterbed function values as follows:

$$\widetilde{H}_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} = h_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} + \tilde{z}_{i+\frac{1}{2},j+\frac{1}{2}}. \quad (41)$$

The projection step of the water height at time t^{n+1} back onto the original grid can be performed now using the limited gradient components of $h_{i+1/2,j+1/2}^{n+1}$ computed by discretizing the spatial partial derivatives in $h_x = H_x - z_x$ and $h_y = H_y - z_y$ as follows:

$$\begin{aligned} (h_x)_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} &= (\widetilde{H}_x)_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} - \frac{\frac{z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{3}{2},j+\frac{1}{2}}}{2} - \frac{z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}}{2}}{\Delta x} \\ (h_y)_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} &= (\widetilde{H}_y)_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} - \frac{\frac{z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{3}{2}}}{2} - \frac{z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}}{2}}{\Delta y}, \end{aligned} \quad (42)$$

where the discrete derivatives $(\widetilde{H}_x)_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1}$ and $(\widetilde{H}_y)_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1}$ are obtained from the staggered values $\widetilde{H}_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1}$ using a gradients limiting procedure.

Theorem 4 *In the context of the two-dimensional shallow water equations (23), and the proposed two-dimensional finite volume method (26) along with the forward and backward projection steps (27) and (28) obtained with the aid of the surface gradient method (37) and (42), if at a given time t^n , the numerical solution satisfies the steady state (4) at the discrete level (i.e., $h_{i,j}^n + z_{i,j} = \text{constant}$, $u_{i,j}^n = v_{i,j}^n = 0$, and $\theta_{i,j}^n = \text{constant}$), then we can show that the updated numerical solution $\mathbf{u}_{i,j}^{n+1}$ satisfies the steady state at the discrete level and the equation $\mathbf{u}_{i,j}^{n+1} = \mathbf{u}_{i,j}^n$ holds for all i, j .*

The detailed proof of theorem (4) is given in the appendix section.

3 Numerical experiments

3.1 One-dimensional numerical experiments:

The one and two-dimensional well-balanced schemes developed in this paper are now applied and used to solve classical Ripa problems.

3.1.1 One-dimensional Riemann problem over a flat bottom

Our first numerical experiment is a classical Riemann problem with two constant states across the point $x_0 = 0$ center of the computational domain $\Omega = [-1, 1]$. The initial conditions are as follows

$$(h, u, \theta) = \begin{cases} (5, 0, 3) & \text{if } x < 0, \\ (1, 0, 5) & \text{if } x > 0. \end{cases}$$

The computational domain Ω is discretized using 200 gridpoints and the numerical solution is calculated at the final time $t = 0.2$. Figure 1 shows the water height, the temperature, and the pressure obtained at the final time using the well-balanced scheme (dotted curve). The reference solution (solid curve) is obtained using 2000 gridpoints. The obtained results are in perfect match and in a good agreement with those appearing in [6] and [9]. Figure 2

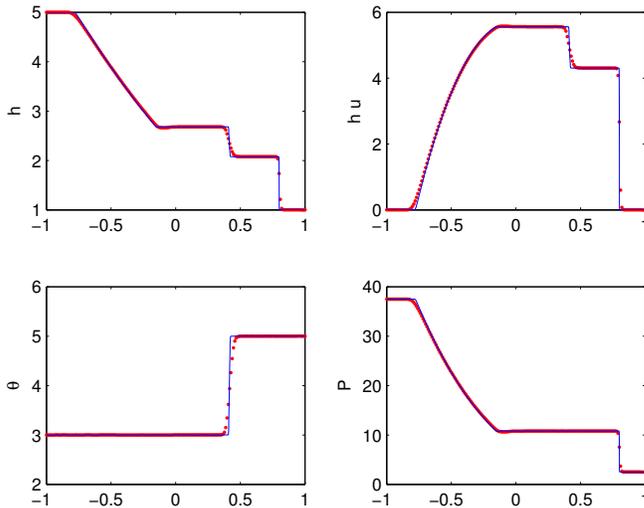


Figure. 1. 1D dam break problem: Water height h , momentum hu , temperature θ , and pressure P obtained at time $t = 2$ using the UCS scheme (dotted line) and the WB-UCS scheme (solid line).

shows the profile of $h\theta$ obtained using the unstaggered central scheme with

(dashed curve) and without (dotted curve) well-balancing on 200 gridpoints. The reference solution (solid curve) is obtained on 2000 gridpoints. The obtained results are in good agreement in this case and the UCS scheme is capable of generating the correct waves mainly because the source term of the RIPA system vanishes in this test case (flat bottom topography).

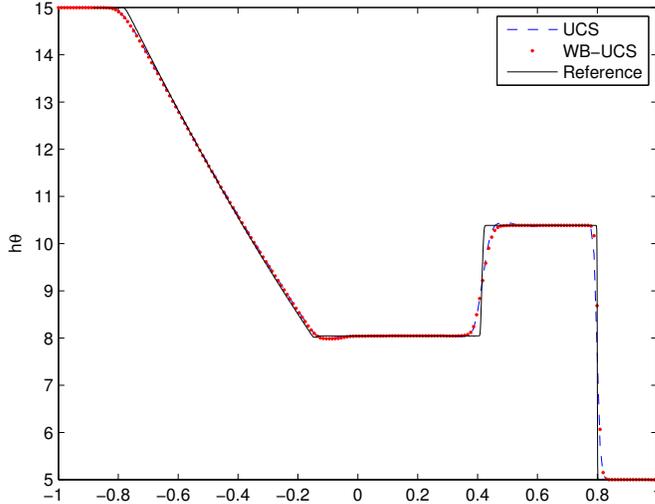


Figure. 2. 1D dam break problem: Water height at time $t = 2$ obtained using the UCS scheme (dashed line) and the WB-UCS scheme (dotted line); the solid line is the reference line obtained on 2000 gridpoints.

3.1.2 Perturbation of a lake at rest problem

Our next one-dimensional experiment is a small perturbation of a lake at rest problem as considered in [6]. The non-flat bottom topography function is given by

$$Z(x) = \begin{cases} 0.85\{\cos[10\pi(x + 0.9)] + 1\}, & \text{if } -1 \leq x \leq -0.8, \\ 1.25\{\cos[10\pi(x - 0.4)] + 1\}, & \text{if } 0.3 \leq x \leq 0.5, \\ 0, & \text{otherwise.} \end{cases}$$

The initial water level is $h + z = 6$, the initial velocity and temperature are $v = 0$ and $\theta = 4$, respectively. We first validate the well-balanced numerical scheme and compute the steady state solution until $t = 1$. Figure 3 (top left) shows the waterbed and the water level at rest at the final time. Figure 3 (top right) shows that the steady state remains satisfied at the discrete level at the final time. Next we perturb the lake at rest and take $h(x, 0) = h(x, 0) + \chi_{[-1.5, -1.4]}(x)$, with $\chi_{[-1.5, -1.4]}(x) = 1$ if $x \in [-1.5, -1.4]$ and $\chi_{[-1.5, -1.4]}(x) = 0$ elsewhere in the computational domain. As time evolves, the initial perturbation splits into two pulses moving in opposite directions,

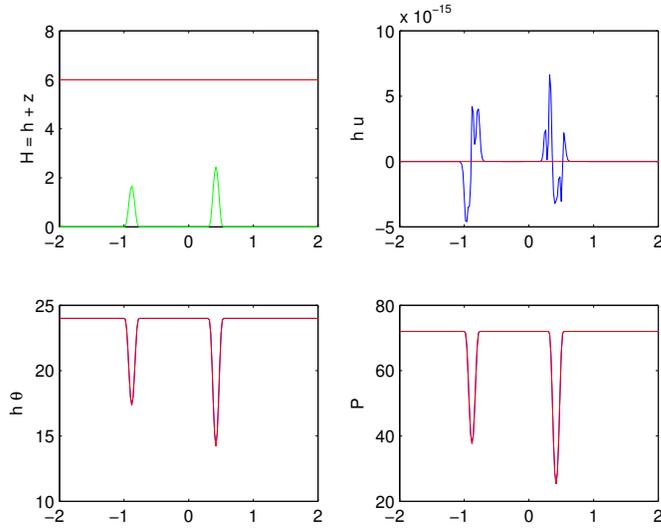


Figure. 3. Lake at rest problem: steady state computed at time $t = 1$.

and leave the computational domain from its endpoints. Figure 4 (top left) shows the water height at time $t = 0.4$.

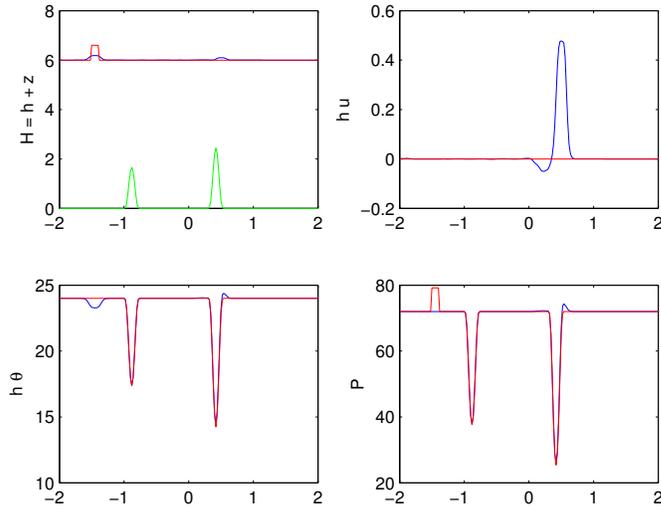


Figure. 4. Perturbation of a lake at rest problem: Numerical solution at the final time $t = 0.4$ showing the propagation of pulses.

3.1.3 Dam break problem over a rectangular bump

Here we extend the shallow water equations problem on a discontinuous waterbed presented in [7] to the case of the Ripa system. The waterbed features

of a rectangular bump defined as follows:

$$z(x) = \begin{cases} 8, & \text{if } |x - 300| < 75, \\ 0, & \text{otherwise.} \end{cases}$$

The initial conditions for (h, u, θ) are as follows

$$(h, u, \theta) = \begin{cases} (20 - z(x), 0, 10), & \text{if } x \leq 300, \\ (15 - z(x), 0, 5), & \text{otherwise.} \end{cases}$$

The numerical solution is computed at the final time $t = 12$ using the well-balanced scheme and the obtained numerical results are reported in figure 5 where we plot $H = h + z$, hu , and $h\theta$ obtained on 200 gridpoints (dotted curve) and 2000 gridpoints (reference solid curve).

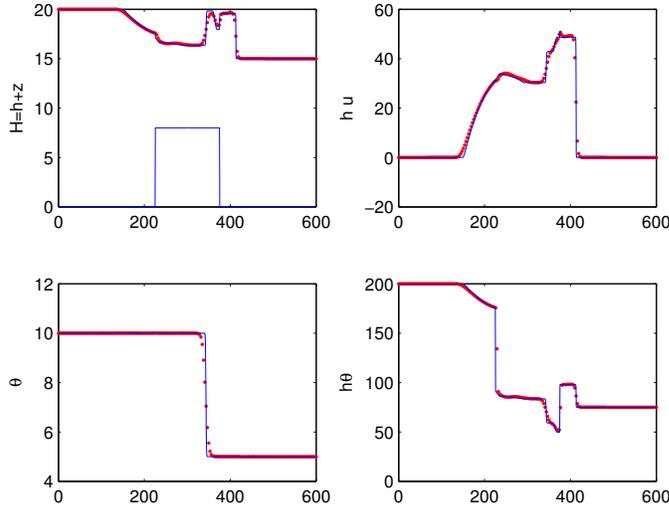


Figure. 5. 1D dam break problem over a rectangular bump: Water height obtained at time $t = 12$ using the well-balanced scheme on 200 grid points (dotted curve) and 2000 grid points (solid curve).

Figure 6 shows the profile of the pressure $P = h\theta^2/2$ at the final time obtained on 200, 400, and 2000 gridpoints. When the surface gradient method is not applied in the forward and backward projection steps, the numerical scheme generates spurious oscillations and non-physical waves due to the non well-balanced effects. Figure 7 shows the water level function obtained with and without the surface gradient method (solid and dashed curves, respectively); non-physical waves start in the neighborhood of waterbed jumps and propagate in the computational domain. The non-physical waves are invisible when the numerical solution is computed using the well-balanced scheme.

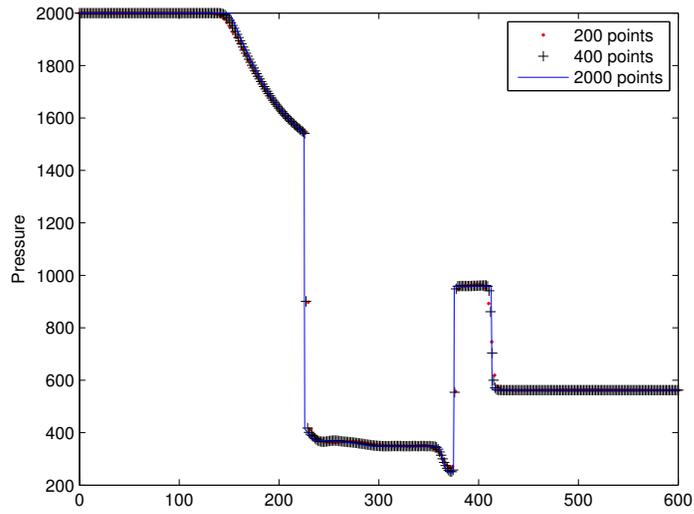


Figure. 6. 1D dam break problem over a rectangular bump: Profile of the pressure $P = h^2\theta/2$ at the final time.

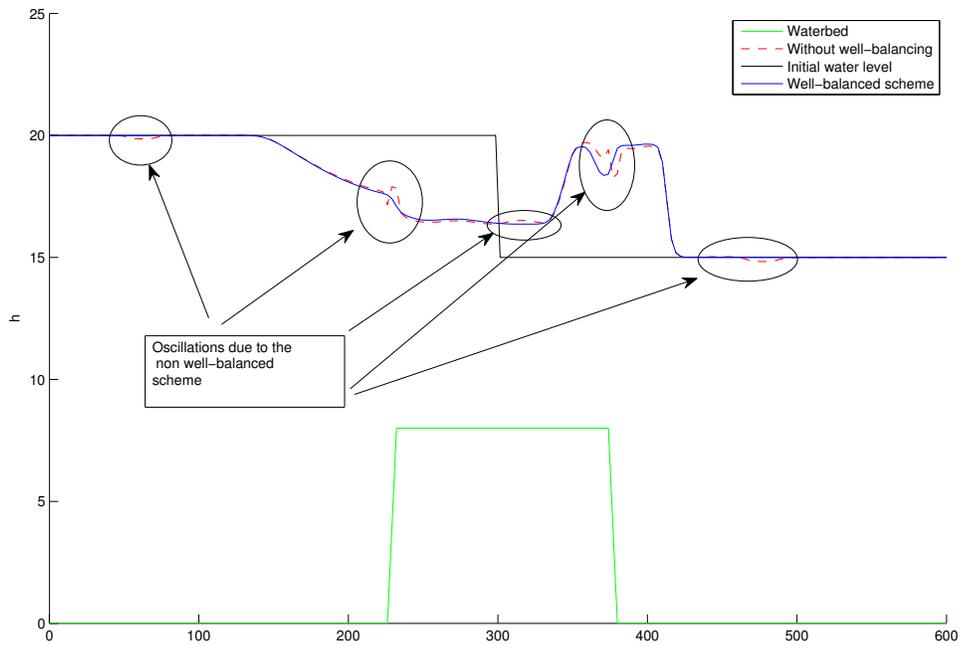


Figure. 7. 1D dam break problem over a rectangular bump: Water level obtained with and without surface gradient method for the forward and backward projection steps

3.1.4 Dam break problem over a flat bottom

Our final one-dimensional problem is a dam break problem over a flat bottom topography. The initial conditions feature two constant states defined as follows:

$$(h, u, \theta) = \begin{cases} (2, 0, 1), & \text{if } |x| \leq 0.5, \\ (1, 0, 1.5), & \text{otherwise.} \end{cases}$$

The computational domain $\Omega = [-1, 1]$ is discretized using 200, 400, and 2000 gridpoints and the numerical solution is calculated at the final time $t = 0.2$ using the well-balanced central scheme. Figure 8 shows the water height at the final time obtained using 200, 400, and 2000 gridpoints. We note

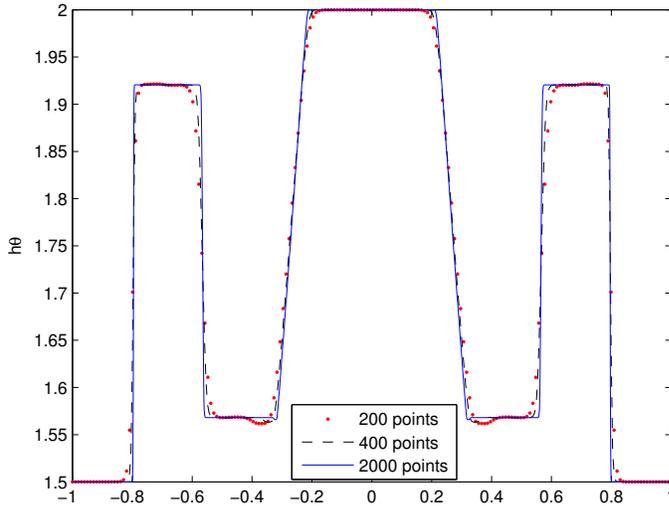


Figure. 8. 1D dam break problem over a flat waterbed: Water height at time $t = 0.2$ obtained using the UCS scheme (dotted line) and the WB-UCS scheme (solid line).

that when the waterbed is flat, the source term in the Ripa system vanishes and the resulting scheme is a homogeneous hyperbolic systems that can be easily solved using finite volume methods. This test case was solved using both the proposed well-balanced numerical scheme and the original unstaggered central scheme [21]. Figure 9 shows the profiles of h , hu , $h\theta$, and the pressure $P = h^2\theta/2$ obtained using the numerical base scheme with (solid curve) and without (dotted curve) well-balancing at time $t = 0.2$. A good agreement between the obtained numerical results is observed in figure 9, thus confirming the consistency of the well-balanced scheme.

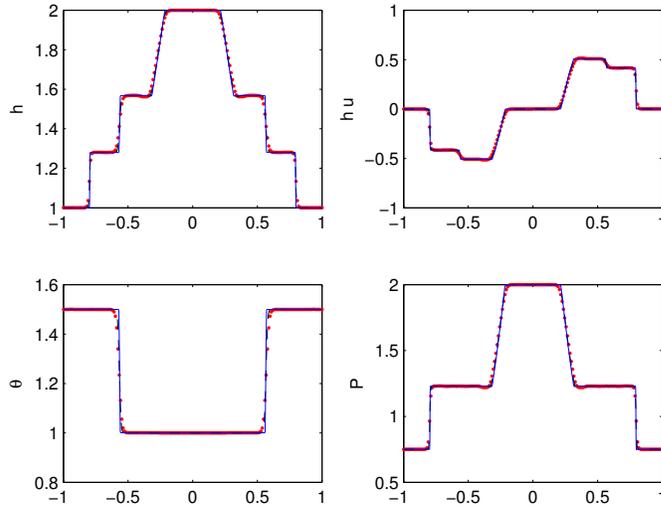


Figure. 9. 1D dam break problem over a flat waterbed: Profile of the water height h , momentum hu , temperature θ , and pressure P obtained at time $t = 0.2$ using the WB-UCS scheme on 200 grid points (dotted curve) and 2000 gridpoints (solid curve).

3.2 Two-dimensional numerical experiments:

Now we validate the two-dimensional well-balanced central schemes developed in this paper and we solve some classical Ripa problems.

3.2.1 Rectangular dam break problem

First, we consider the classical two-dimensional rectangular dam break problem over a flat bottom topography. The initial conditions feature two constant states defined as follows:

$$(h, u, v, \theta) = \begin{cases} (2, 0, 0, 1), & \text{if } -0.5 \leq x \leq 0.5, \\ (1, 0, 0, 1.5), & \text{otherwise.} \end{cases}$$

The computational domain $\Omega = [-1, 1]^2$ is discretized using 100^2 gridpoints and the numerical solution is calculated at the final time $t = 0.2$ using the well-balanced central scheme. Figure 10 shows the profile of the obtained water height at the final time, while figure 11 shows cross sections of the water height along the x -axis obtained at the final time on 50^2 , 100^2 , and 200^2 gridpoints and compared to the solution of the corresponding one-dimensional problem obtained on 2000 gridpoints. Figure 11 shows a good agreement between the obtained numerical results. The structure of the obtain solution features two shock waves and two contact waves propagating away from the y -axis and

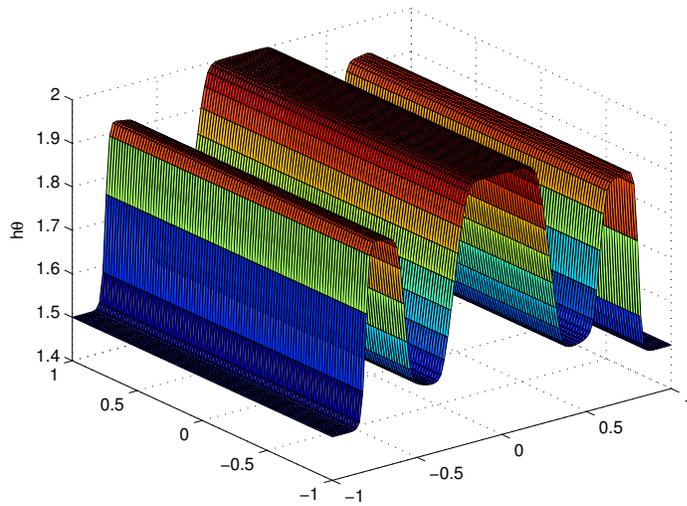


Figure. 10. Rectangular dam break problem: Water height at time $t = 0.2$ obtained using the UCS scheme (dotted line) and the WB-UCS scheme (solid line).

two rarefaction waves propagating towards the y -axis.

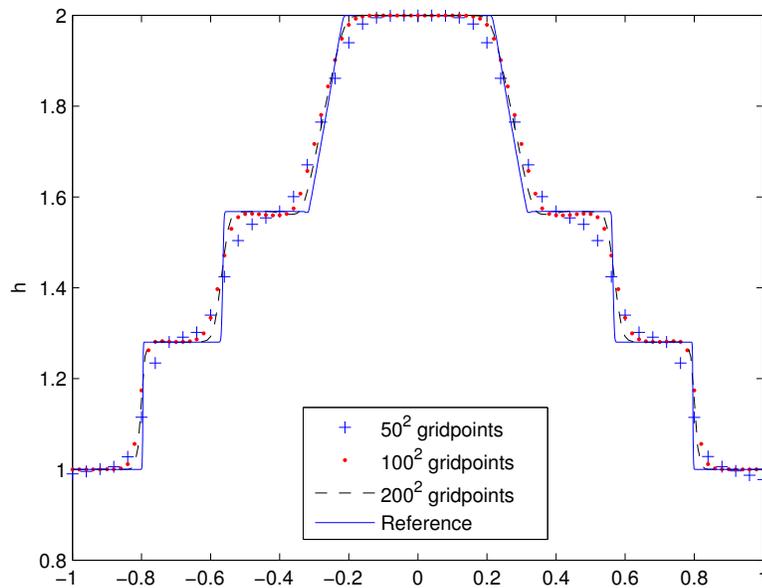


Figure. 11. Rectangular dam break problem: Profile of the water height h obtained on 50^2 , 100^2 , and 200^2 gridpoints and compared to the solution of the corresponding one-dimensional problem obtained on 2000 gridpoints.

3.2.2 Circular dam break problem

Next, we consider the classical two-dimensional circular dam break problem over a flat bottom topography. The initial conditions feature two constant states defined as follows:

$$(h, u, v, \theta) = \begin{cases} (2, 0, 0, 1), & \text{if } x^2 + y^2 \leq 0.25, \\ (1, 0, 0, 1.5), & \text{otherwise.} \end{cases}$$

The computational domain $\Omega = [-1, 1]^2$ is discretized using 100^2 gridpoints and the numerical solution is calculated at the final time $t = 0.2$ using the well-balanced central scheme. Figure 12 shows the profile of the water height obtained using the proposed numerical scheme. We see a shock and a contact wave propagating radially towards the exterior of the domain, and a circular rarefaction wave is propagating towards the center of the domain. Figures 13

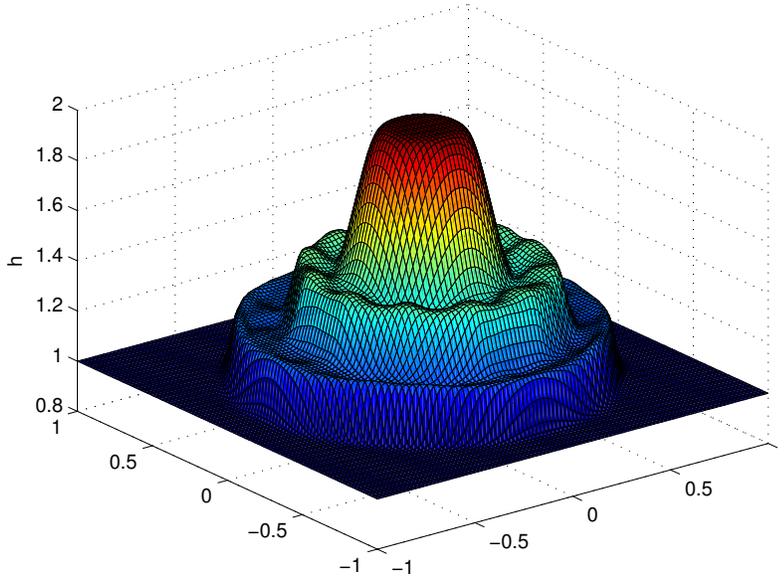


Figure. 12. Circular dam break problem: Profile of the water height h obtained time $t = 0.2$ on 100^2 gridpoints.

and 14 show scatter plots of the cross sections of the water height h along the diagonal line ($y = x$) and the y -axis ($x=0$), obtained on 50^2 , 100^2 , and 200^2 gridpoints.

3.2.3 Perturbation of a steady state on an irregular waterbed

For our next numerical experiment, we consider a small perturbation of a steady state problem, a variant of the problem presented in [6]. The initial

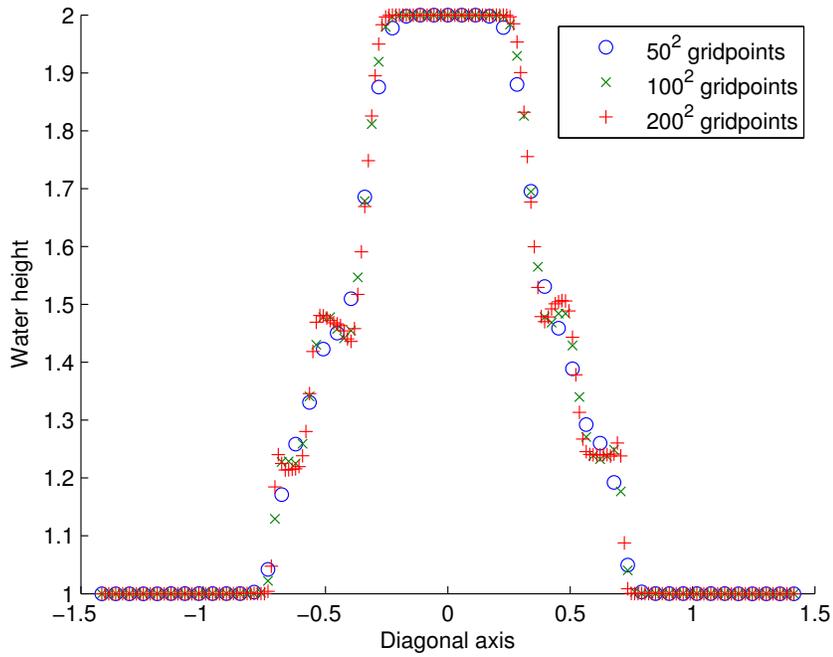


Figure. 13. Circular dam break problem: Scatter plot of the cross sections of the water height h at time $t = 0.15$ along the diagonal line $y = x$ obtained on 50^2 , 100^2 , and 200^2 gridpoints.

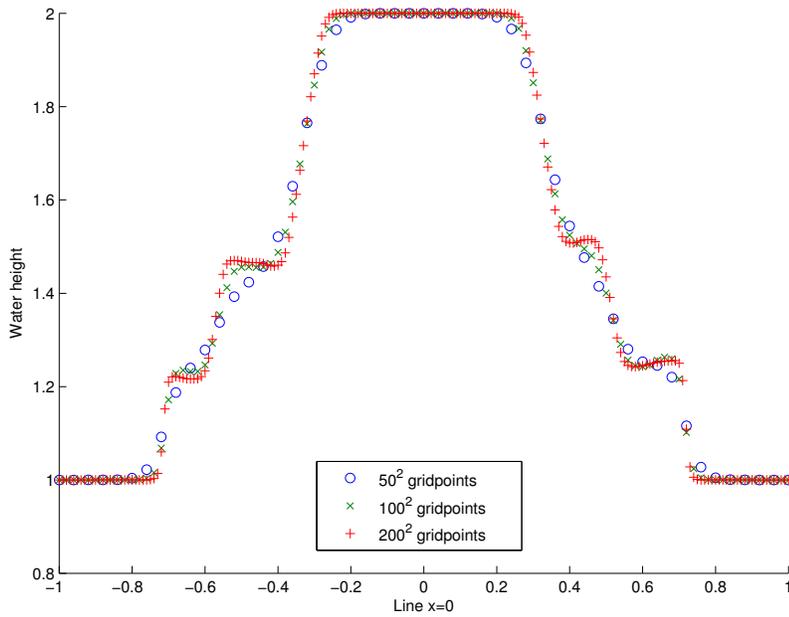


Figure. 14. Circular dam break problem: Scatter plot of the cross sections of the water height h at time $t = 0.15$ along the line $x = 0$ obtained on 50^2 , 100^2 , and 200^2 gridpoints.

data features the steady state $(h + z, u, v, \theta)(x, y, 0) = (3, 0, 0, 4/3)$ for all $(x, y) \in [-2, 2]^2$. The waterbed topography is defined as follows

$$z(x, y) = \begin{cases} 0.5 \exp[-100((x + 0.5)^2 + (y + 0.5)^2)], & \text{if } x \leq 0 \\ 0.6 \exp[-100((x - 0.5)^2 + (y - 0.5)^2)], & \text{if } x > 0 \end{cases}$$

The steady state solution is perturbed by introducing the variation of the water height $h(x, y, 0) = h(x, y, 0) + 0.1$ for all (x, y) inside the annulus $0.01 \leq x \leq 0.09$. The well-balanced numerical scheme is first validated and the numerical solution of the steady state (without perturbation) is computed until time $t = 0.8$ on a 100^2 gridpoints. The obtained water height is shown in figure (15) along with the waterbed function. Figure 16 shows the profile of the

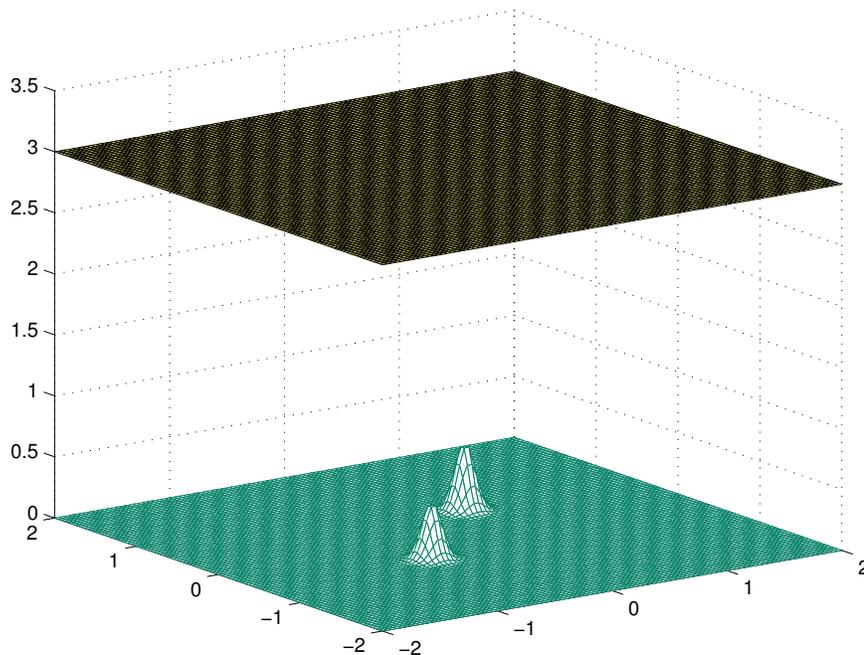


Figure. 15. Perturbation of a steady state problem: Profile of the water height at a steady state at time $t = 0.8$ obtained on 100^2 gridpoints obtained using the WB-UCS scheme.

water height obtained at time $t = 0.15$ and shows the propagation of the water perturbation across the computational domain. Figure 17 shows some contour lines of the water height at time $t = 0.15$ obtained using the well-balanced scheme; these results are in good agreement with corresponding ones presented in [6]. Figure 18 shows a scatter plot of the cross sections of $h\theta$ along the diagonal line $y = x$ obtained at the final time $t = 0.15$ on 50^2 , 100^2 , and 200^2 using the WB-UCS scheme.

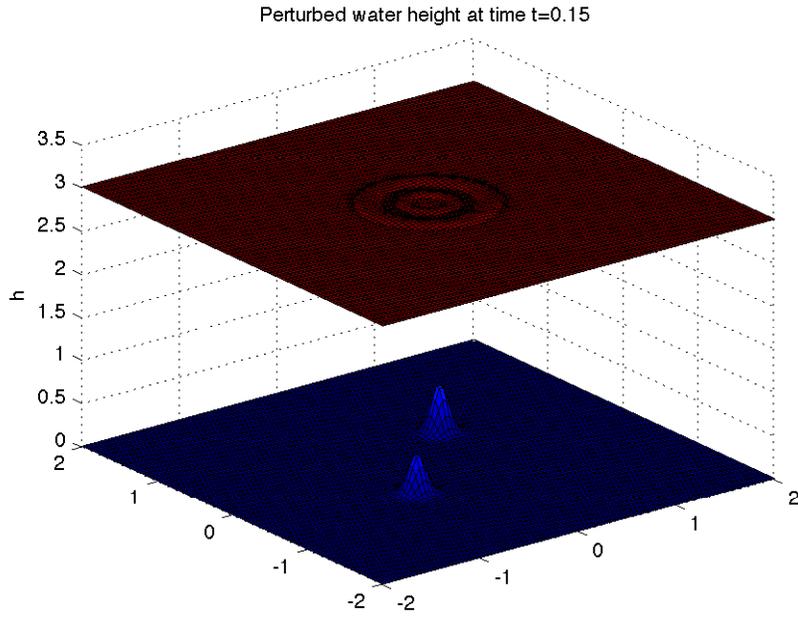


Figure. 16. Perturbation of a steady state problem: Profile of the perturbed water height at time $t = 0.15$ obtained on 100^2 gridpoints obtained using the WB-UCS scheme.

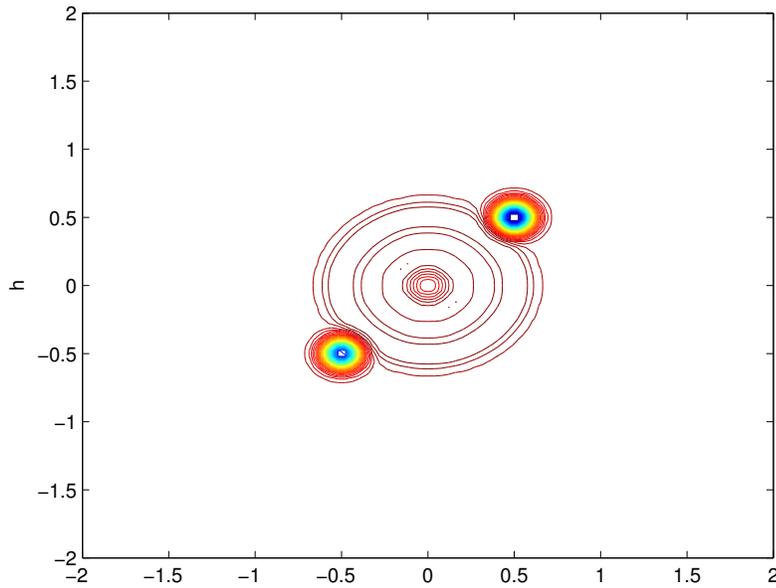


Figure. 17. Perturbation of a steady state problem: Contour lines of the water height at time $t = 0.15$ on 100^2 gridpoints obtained using the WB-UCS scheme.

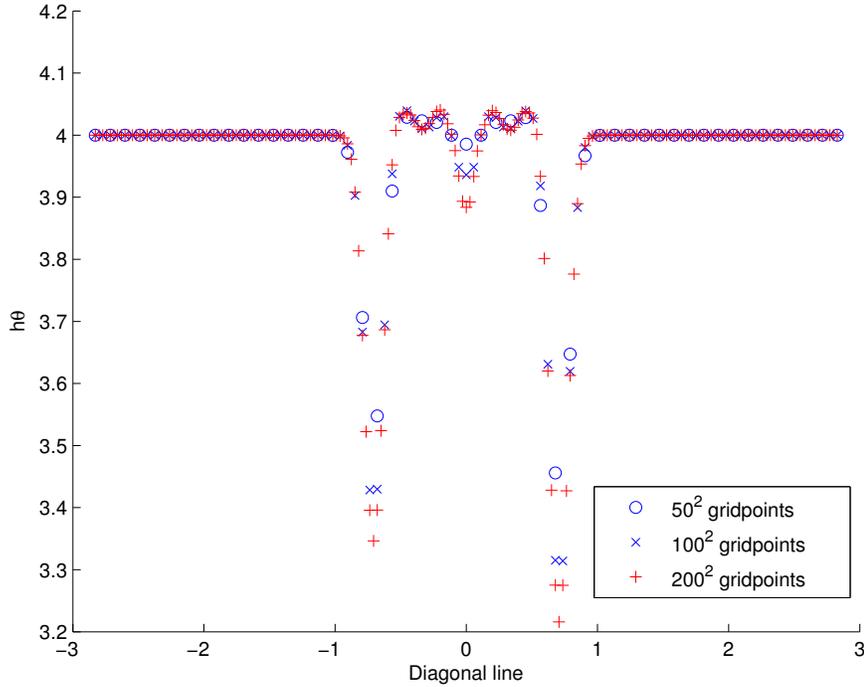


Figure. 18. Perturbation of a steady state problem: Scatter plot of the cross sections of $h\theta$ along the diagonal line $y = xs$ at time $t = 0.15$ obtained on 50^2 , 100^2 , and 200^2 gridpoints using the WB-UCS scheme.

4 Conclusion

In this paper we have developed a new well-balanced finite volume method for the Ripa system within the framework of one and two-dimensional unstaggered central schemes.

The proposed scheme has the advantages of being an unstaggered central scheme, second-order accurate, and it evolves the numerical solution on a single Cartesian grid.

To satisfy the well-balanced constraint in the case of the Ripa system we discretize the source term according to the discretization of the flux divergence term with the aid of sensor functions. Furthermore, to ensure that the equilibrium solution is exactly satisfied at the discrete level, we carefully adapt the surface gradient method to the case of the Ripa system by linearizing of the water height function, on both the original and staggered grids, according to the linearization of the water level function. The temperature component is linearized in a similar fashion.

The proposed scheme is then validated and classical problems from the recent literature are successfully solved. The numerical simulations we carried out deal with dam-breaks and hydraulic jumps with variable bed topography and varying temperature gradients. The numerical results obtained for both one

and two-dimensional Ripa system problems are in perfect agreement with corresponding results appearing in the literature, thus confirming the efficiency and potential of the proposed numerical scheme to accurately solve Ripa system problems.

Appendix

We now present the proofs of theorems (1)-(4).

Proof of theorem 1

First we show the predicted solution as time $t^{n+1/2}$ is invariant in time (i.e., is equal to the solution at time t^n) $\mathbf{u}_i^{n+1/2} = \mathbf{u}_i^n$.

Recall that $\mathbf{u}_i^{n+1/2}$ is computed using a Taylor series expansion in time as follows

$$\mathbf{u}_i^{n+1/2} = \mathbf{u}_i^n + \frac{\Delta t}{2} \left(-f'_i + S_i^n \Delta x \right),$$

and the source term S_i^n is discretized according to the discretization of $(h_i^n)'$.

If h'_i is discretized using the backward difference, i.e., $h'_i = \Theta \frac{h_i^n - h_{i-1}^n}{\Delta x}$, then the sensor function becomes $\sigma_i = -1$ and we obtain

$$S_{i,L}^n = \begin{pmatrix} 0 \\ -gh_i^n \theta_i^n \Theta \frac{z_i - z_{i-1}}{\Delta x} \\ 0 \end{pmatrix}, S_{i,R}^n = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \text{ and } S_{i,C}^n = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

Therefore,

$$S_i^n = S_{i,L}^n + S_{i,C}^n + S_{i,R}^n = \begin{pmatrix} 0 \\ -gh_i^n \theta_i^n \Theta \frac{z_i - z_{i-1}}{\Delta x} \\ 0 \end{pmatrix}$$

and in the context of the steady state (4), $f(u) = \begin{pmatrix} 0 \\ \frac{1}{2}gh^2\theta \\ 0 \end{pmatrix}$ so $(f_i^n)' =$

$$\begin{pmatrix} 0 \\ gh_i^n \theta_i^n \cdot (h_i^n)' \\ 0 \end{pmatrix} \text{ because } \theta_i^n \text{ is constant in the steady state case.}$$

Therefore the prediction step becomes,

$$\begin{aligned}
\mathbf{u}_i^{n+\frac{1}{2}} &= \mathbf{u}_i^n + \frac{\Delta t}{2} \left[\begin{pmatrix} 0 \\ -gh_i^n \theta_i^n (h_i^n)' \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ -gh_i^n \theta_i^n \Theta \frac{z_i - z_{i-1}}{\Delta x} \\ 0 \end{pmatrix} \right] \\
&= \mathbf{u}_i^n + \frac{\Delta t}{2} \begin{pmatrix} 0 \\ -gh_i^n \theta_i^n [(h_i^n)' + \Theta \frac{z_i - z_{i-1}}{\Delta x}] \\ 0 \end{pmatrix} \\
&= \mathbf{u}_i^n + \frac{\Delta t}{2\Delta x} \begin{pmatrix} 0 \\ -gh_i^n \theta_i^n [\Theta(h_i + z_i) - \theta(h_{i-1} + z_{i-1})] \\ 0 \end{pmatrix}
\end{aligned}$$

But since $h_i^n + z_i = H = \text{constant}$ for all i then $\Theta(h_i^n + z_i) - \Theta(h_{i-1}^n + z_{i-1}) = 0$ leading to

$$\mathbf{u}_i^{n+\frac{1}{2}} = \mathbf{u}_i^n$$

A similar proof can be done for the values of the sensor function σ_i . ■

Next we show that $\mathbf{u}_{i+\frac{1}{2}}^{n+1} = \mathbf{u}_{i+\frac{1}{2}}^n$. Recall that

$$\mathbf{u}_{i+\frac{1}{2}}^{n+1} = \mathbf{u}_{i+\frac{1}{2}}^n - \frac{\Delta t}{\Delta x} \left(f(\mathbf{u}_{i+1}^{n+\frac{1}{2}}) - f(\mathbf{u}_i^{n+\frac{1}{2}}) \right) + \frac{1}{\Delta x} S(\mathbf{u}_i^{n+\frac{1}{2}}, \mathbf{u}_{i+1}^{n+\frac{1}{2}})$$

where the source term is discretized using the formula

$$S(\mathbf{u}_i^{n+\frac{1}{2}}, \mathbf{u}_{i+1}^{n+\frac{1}{2}}) = \Delta t \Delta x \begin{pmatrix} 0 \\ g \frac{(h\theta)_i^{n+\frac{1}{2}} + (h\theta)_{i+1}^{n+\frac{1}{2}}}{2} \left(-\frac{z_{i+1} - z_i}{\Delta x} \right) \\ 0 \end{pmatrix}$$

In the steady state case (4) we have $\theta_i^n = \text{constant}$ and $\mathbf{u}_i^n = 0$ (zero velocity), and from the first part of theorem (1) just established above, we have

$$\mathbf{u}_i^{n+\frac{1}{2}} = \mathbf{u}_i^n, \text{ thus we obtain } u = \begin{pmatrix} h \\ 0 \\ h\theta \end{pmatrix} \text{ and } f(u) = \begin{pmatrix} 0 \\ \frac{1}{2}gh^2\theta \\ 0 \end{pmatrix}, \text{ therefore the}$$

discretized flux comes

$$f(\mathbf{u}_{i+1}^{n+\frac{1}{2}}) = \begin{pmatrix} 0 \\ \frac{1}{2}g(h_{i+1}^{n+\frac{1}{2}})^2\theta_{i+1}^{n+\frac{1}{2}} \\ 0 \end{pmatrix}, \text{ and } f(\mathbf{u}_i^{n+\frac{1}{2}}) = \begin{pmatrix} 0 \\ \frac{1}{2}g(h_i^{n+\frac{1}{2}})^2\theta_i^{n+\frac{1}{2}} \\ 0 \end{pmatrix}$$

leading to:

$$\begin{aligned} \mathbf{u}_{i+\frac{1}{2}}^{n+1} = & \mathbf{u}_{i+\frac{1}{2}}^n - \frac{\Delta t}{\Delta x} \left[\begin{pmatrix} 0 \\ \frac{1}{2}g(h_{i+1}^{n+\frac{1}{2}})^2\theta_{i+1}^{n+\frac{1}{2}} \\ 0 \end{pmatrix} - \begin{pmatrix} 0 \\ \frac{1}{2}g(h_i^{n+\frac{1}{2}})^2\theta_i^{n+\frac{1}{2}} \\ 0 \end{pmatrix} \right] \\ & + \Delta t \begin{pmatrix} 0 \\ g \frac{(h\theta)_i^{n+\frac{1}{2}} + (h\theta)_{i+1}^{n+\frac{1}{2}}}{2} \left(-\frac{z_{i+1} - z_i}{\Delta x} \right) \\ 0 \end{pmatrix} \end{aligned}$$

Performing basic algebra operations, and taking into account that $\theta_i^{n+1/2} = \theta_{i+1}^{n+1/2} = \theta_i^n = \text{constant}$, we obtain

$$\mathbf{u}_{i+\frac{1}{2}}^{n+1} = \mathbf{u}_{i+\frac{1}{2}}^n - \frac{\Delta t}{\Delta x} \theta_i^n \begin{pmatrix} 0 \\ \frac{1}{2}g(h_{i+1}^{n+\frac{1}{2}} + h_i^{n+\frac{1}{2}}) \left[(h_{i+1}^{n+\frac{1}{2}} + z_{i+1}) - (h_i^{n+\frac{1}{2}} + z_i) \right] \\ 0 \end{pmatrix}$$

But since $\mathbf{u}_i^{n+\frac{1}{2}} = \mathbf{u}_i^n$ and $h_i^n + z_i = H = \text{constant}$ for all i , then $(h_i^n + z_i) - (h_{i+1}^n + z_{i+1}) = 0$. Therefore, $(h_i^{n+\frac{1}{2}} + z_i) - (h_{i+1}^{n+\frac{1}{2}} + z_{i+1}) = (h_i^n + z_i) - (h_{i+1}^n + z_{i+1}) = 0$, leading to $\mathbf{u}_{i+\frac{1}{2}}^{n+1} = \mathbf{u}_{i+\frac{1}{2}}^n$ which means that if the steady state requirement was satisfied at time t^n on the dual cells, it will remain as such at time t^{n+1} . ■

Proof of theorem 2

To show that the steady state (4) is maintained at time t^{n+1} provided it was maintained at time t^n we will proceed component wise and show that $\mathbf{u}_i^{n+1} = \mathbf{u}_i^n$, for all i . Below we will present the proof for the $h\theta$ component and the prove can be easily generalized the the h component. We note that in the steady state case (4) the hu component doesn't change in time because $u = 0$ and because of the well-balanced discretization.

We use the surface gradient method for both the forward and backward pro-

jection step; for the forward projection we have

$$(h\theta)_{i+1/2}^n = \frac{1}{2} \left((h\theta)_i^n + (h\theta)_{i+1}^n \right) + \frac{\Delta x}{8} \left[(h\theta)_i^{n|'} - (h\theta)_{i+1}^{n|'} \right] \quad (43)$$

We discretize the derivatives of $h\theta$ using the product rule and with the aid of the water level function $H_i^n = h_i^n + z_i$ as follows

$$(h\theta)_i^{n|'} = (h_i^n)' \theta_i^n + h_i^n (\theta_i^n)' = \left[(H_i^n)' - \frac{z_{i+1/2} - z_{i-1/2}}{\Delta x} \right] \theta_i^n + h_i^n (\theta_i^n)'$$

Substituting in equation (43) one obtains (while taking into account that $H_i^n = H = \text{constant}$ and θ_i^n is constant for all i)

$$(h\theta)_{i+1/2}^n = \left[\frac{1}{2} (h_i^n + h_{i+1}^n) - \frac{1}{2} \left[z_{i+1/2} - \frac{z_i + z_{i+1}}{2} \right] \right] \theta_i^n \quad (44)$$

Similarly, for the backward projection step we apply the surface gradient method and one obtains (while taking into account that $\theta_{i+1/2}^{n+1/2}$ is constant and $\widetilde{H}_{i+1/2}^{n+1/2} = h_{i+1/2}^{n+1/2} + \widetilde{z}_{i+1/2}$ is constant for all i)

$$(h\theta)_i^{n+1} = \left[\frac{1}{2} (h_{i-1/2}^{n+1} + h_{i+1/2}^{n+1}) - \frac{1}{8} [-(z_i - z_{i-1}) + (z_{i+1} - z_i)] \right] \theta_i^n \quad (45)$$

Substituting equation (44) in (45) we obtain $(h\theta)_i^{n+1} = (h\theta)_i^n$. The proof for $h_i^{n+1} = h_i^n$ is performed in a similar way. Therefore we conclude that if the steady state (4) was satisfied at time t^n , then it will remain as such at the next time t^{n+1} . ■

Proof of theorem 3

The proof of theorem 3 follows the same strategy as the one of theorem 1. We assume that $\mathbf{u}_{i,j}^n$ satisfies the steady state (4) and we show first that $\mathbf{u}_{i,j}^{n+1/2} = \mathbf{u}_{i,j}^n$ and the $\mathbf{u}_{i+1/2,j+1/2}^{n+1}$.

When the steady state case (4) is satisfied at time t^n at the discrete level, the Ripa system reduces to

$$\partial_t \begin{pmatrix} h \\ 0 \\ 0 \\ h\theta \end{pmatrix} + \partial_x \begin{pmatrix} 0 \\ \frac{1}{2}gh^2\theta \\ 0 \\ 0 \end{pmatrix} + \partial_y \begin{pmatrix} 0 \\ 0 \\ \frac{1}{2}gh^2\theta \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ gh\theta \left(-\frac{\partial z}{\partial x} \right) \\ gh\theta \left(-\frac{\partial z}{\partial y} \right) \\ 0 \end{pmatrix}$$

The predicted solution at time $t^{n+1/2}$ is calculated using a Taylor expansion in time and the balance law as follows

$$\mathbf{u}_{i,j}^{n+\frac{1}{2}} = \mathbf{u}_{i,j}^n + \frac{\Delta t}{2} \left(-\frac{(f_x)_{i,j}}{\Delta x} - \frac{(g_y)_{i,j}}{\Delta y} + S_{i,j}^n \right) \quad (46)$$

where $S_{i,j}^n$ is the discretized source term obtained with the aid of sensor functions. If the partial derivatives of the flux functions are approximated using backward differences, then the sensor functions take the values $\sigma_2 = -1$ and $\sigma_3 = -1$. Using the product rule and taking into account that $\theta_{i,j}^n$ is constant, the numerical partial derivatives of $h^2\theta$ required in equation (46) become

$$\begin{aligned} (h^2\theta)_x|_{i,j}^n &= 2(h\theta)_{i,j}^n \Theta \left(\frac{h_{i,j}^n - h_{i-1,j}^n}{\Delta x} \right) \\ (h^2\theta)_y|_{i,j}^n &= 2(h\theta)_{i,j}^n \Theta \left(\frac{h_{i,j}^n - h_{i,j-1}^n}{\Delta y} \right) \end{aligned}$$

The source term is then discretized according to f_x and g_y as follows

$$S_{i,j}^n = \begin{pmatrix} S_1 = 0 \\ S_2 = -g(h\theta)_{i,j}^n \Theta \frac{z_{i,j} - z_{i-1,j}}{\Delta x} \\ S_3 = -g(h\theta)_{i,j}^n \Theta \frac{z_{i,j} - z_{i,j-1}}{\Delta y} \\ S_4 = 0 \end{pmatrix},$$

where $1 \leq \Theta \leq 2$ is the parameter of the MC- Θ limiter.

Substituting in (46) and performing elementary algebra operations, we obtain

$$\begin{aligned} \mathbf{u}_{i,j}^{n+\frac{1}{2}} &= \mathbf{u}_{i,j}^n + \frac{\Delta t}{2} \begin{pmatrix} 0 \\ -g(h\theta)_{i,j}^n \Theta \left(\frac{h_{i,j} - h_{i-1,j}}{\Delta x} + \frac{z_{i,j} - z_{i-1,j}}{\Delta x} \right) \\ -g(h\theta)_{i,j}^n \Theta \left(\frac{h_{i,j} - h_{i,j-1}}{\Delta y} + \frac{z_{i,j} - z_{i,j-1}}{\Delta y} \right) \\ 0 \end{pmatrix} \\ &= \mathbf{u}_{i,j}^n + \frac{\Delta t}{2} \begin{pmatrix} 0 \\ -g(h\theta)_{i,j}^n \frac{\Theta}{\Delta x} [(h_{i,j} + z_{i,j}) - (h_{i-1,j} + z_{i-1,j})] \\ -g(h\theta)_{i,j}^n \frac{\Theta}{\Delta y} [(h_{i,j} + z_{i,j}) - (h_{i,j-1} + z_{i,j-1})] \\ 0 \end{pmatrix} \end{aligned}$$

And since $h_{i,j} + z_{i,j} = H = \text{constant}$ for all (x_i, y_j) , then we obtain $\mathbf{u}_{i,j}^{n+\frac{1}{2}} = \mathbf{u}_{i,j}^n$. Similar proofs can be done to show that $\mathbf{u}_{i,j}^{n+\frac{1}{2}} = \mathbf{u}_{i,j}^n$ for the other values of the sensor functions σ_i and t_j . ■

Next we show to show that $\mathbf{u}_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} = \mathbf{u}_{i+\frac{1}{2},j+\frac{1}{2}}^n$

The solution at time t^{n+1} on the dual cells $D_{i+1/2,j+1/2}$ is obtained using equation (26) as follows

$$\mathbf{u}_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} = \mathbf{u}_{i+\frac{1}{2},j+\frac{1}{2}}^n \quad (47)$$

$$\begin{aligned} & -\frac{\Delta t}{2\Delta x} \left[f(u_{i+1,j}^{n+\frac{1}{2}}) + f(u_{i+1,j+1}^{n+\frac{1}{2}}) - f(u_{i,j}^{n+\frac{1}{2}}) - f(u_{i,j+1}^{n+\frac{1}{2}}) \right] \\ & -\frac{\Delta t}{2\Delta y} \left[g(u_{i,j+1}^{n+\frac{1}{2}}) + g(u_{i+1,j+1}^{n+\frac{1}{2}}) - g(u_{i,j}^{n+\frac{1}{2}}) - g(u_{i+1,j}^{n+\frac{1}{2}}) \right] \\ & + \frac{\Delta t}{\Delta x \Delta y} S \left(\mathbf{u}_{i,j}^{n+\frac{1}{2}}, \mathbf{u}_{i+1,j}^{n+\frac{1}{2}}, \mathbf{u}_{i,j+1}^{n+\frac{1}{2}}, \mathbf{u}_{i+1,j+1}^{n+\frac{1}{2}} \right) \end{aligned} \quad (48)$$

were the integral of the source term is approximated with second-order of accuracy as follows

$$\begin{aligned} & S \left(\mathbf{u}_{i,j}^{n+\frac{1}{2}}, \mathbf{u}_{i+1,j}^{n+\frac{1}{2}}, \mathbf{u}_{i,j+1}^{n+\frac{1}{2}}, \mathbf{u}_{i+1,j+1}^{n+\frac{1}{2}} \right) = \\ & \frac{\Delta x \Delta y}{2} \left[\left(\begin{array}{c} 0 \\ -g \frac{(h\theta)_{i+1,j} + (h\theta)_{i,j}}{2} \frac{z_{i+1,j} - z_{i,j}}{\Delta x} \\ -g \frac{(h\theta)_{i,j+1} + (h\theta)_{i,j}}{2} \frac{z_{i,j+1} - z_{i,j}}{\Delta y} \\ 0 \end{array} \right) + \left(\begin{array}{c} 0 \\ -g \frac{(h\theta)_{i+1,j+1} + (h\theta)_{i,j+1}}{2} \frac{z_{i+1,j+1} - z_{i,j+1}}{\Delta x} \\ -g \frac{(h\theta)_{i+1,j+1} + (h\theta)_{i+1,j}}{2} \frac{z_{i+1,j+1} - z_{i+1,j}}{\Delta y} \\ 0 \end{array} \right) \right] \end{aligned}$$

To show that $\mathbf{u}_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} = \mathbf{u}_{i+\frac{1}{2},j+\frac{1}{2}}^n$, we shall proceed component wise. Since $\mathbf{u}_{i,j}^n$ satisfies the steady state (4), then

$$\mathbf{u} = \begin{pmatrix} h \\ 0 \\ 0 \\ h\theta \end{pmatrix}, f = \begin{pmatrix} 0 \\ \frac{1}{2}gh^2\theta \\ 0 \\ 0 \end{pmatrix}, g = \begin{pmatrix} 0 \\ 0 \\ \frac{1}{2}gh^2\theta \\ 0 \end{pmatrix}, \text{ and } S = \begin{pmatrix} 0 \\ gh\theta \left(-\frac{\partial z}{\partial x}\right) \\ gh\theta \left(-\frac{\partial z}{\partial y}\right) \\ 0 \end{pmatrix}$$

and we also have from part 1 of theorem 3 $\mathbf{u}_{i,j}^{n+\frac{1}{2}} = \mathbf{u}_{i,j}^n$

- The fluxes and the source term that correspond to the h component are

zeros, then the h component is updated in equation (47) as follows

$$h_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} = h_{i+\frac{1}{2},j+\frac{1}{2}}^n$$

- The hu component is updated using equation (47)

$$\begin{aligned} (hu)_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} &= (hu)_{i+\frac{1}{2},j+\frac{1}{2}}^n \\ &- g \frac{\Delta t}{4\Delta x} \left[(h^2\theta)_{i+1,j}^{n+\frac{1}{2}} - (h^2\theta)_{i,j}^{n+\frac{1}{2}} + (h^2\theta)_{i+1,j+1}^{n+\frac{1}{2}} - (h^2\theta)_{i,j+1}^{n+\frac{1}{2}} \right] \\ &+ \frac{\Delta t}{2} \left[-g \frac{(h\theta)_{i+1,j}^{n+\frac{1}{2}} + (h\theta)_{i,j}^{n+\frac{1}{2}}}{2} \frac{z_{i+1,j} - z_{i,j}}{\Delta x} \right] \\ &+ \frac{\Delta t}{2} \left[-g \frac{(h\theta)_{i+1,j+1}^{n+\frac{1}{2}} + (h\theta)_{i,j+1}^{n+\frac{1}{2}}}{2} \frac{z_{i+1,j+1} - z_{i,j+1}}{\Delta x} \right] \end{aligned} \quad (49)$$

Taking into account that $\theta_{i,j}^n = \text{constant}$, $h_{i,j}^n + z_{i,j} = H = \text{constant}$, and $\mathbf{u}_{i,j}^{n+\frac{1}{2}} = \mathbf{u}_{i,j}^n$ for all i, j , then (49) reduces to $(hu)_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} = (hu)_{i+\frac{1}{2},j+\frac{1}{2}}^n$.

- Similarly we can show that $(hv)_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} = (hv)_{i+\frac{1}{2},j+\frac{1}{2}}^n$.
- The fluxes and source term that correspond to the $h\theta$ component are zeros, then $h\theta$ is updated as follows

$$(h\theta)_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} = (h\theta)_{i+\frac{1}{2},j+\frac{1}{2}}^n$$

Thus we conclude that $\mathbf{u}_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} = \mathbf{u}_{i+\frac{1}{2},j+\frac{1}{2}}^n$ for all i, j . ■

Proof of theorem 4

Now we want to show that the equality $\mathbf{u}_{i,j}^{n+1} = \mathbf{u}_{i,j}^n$ holds for all i, j whenever the steady state 4 is satisfied at the discrete level at time t^n . We shall proceed componentwise; we note that the equalities $hu_{i,j}^{n+1} = hu_{i,j}^n$ and $hv_{i,j}^{n+1} = hv_{i,j}^n$ are immediate from theorem 3 since we have $u^n_{i,j} = v^n_{i,j} = 0$ for all i, j . We start by showing the equality $h_{i,j}^{n+1} = h_{i,j}^n$; we recall that for both the first and fourth components of $\mathbf{u}_{i,j}^n$, the forward projection step (27) is performed using the surface gradient method as follows

$$\begin{aligned}
h_{i+\frac{1}{2},j+\frac{1}{2}}^n &= \frac{1}{4} \left(h_{i,j}^n + h_{i+1,j}^n + h_{i,j+1}^n + h_{i+1,j+1}^n \right) \\
&+ \frac{\Delta x}{16} \left(H_{x|i,j}^n - \frac{\frac{z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}}{2} - \frac{z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{1}{2}}}{2}}{\Delta x} \right) \\
&+ \frac{\Delta x}{16} \left(H_{x|i,j+1}^n - \frac{\frac{z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{3}{2}}}{2} - \frac{z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{3}{2}}}{2}}{\Delta x} \right) \\
&- \frac{\Delta x}{16} \left(H_{x|i+1,j}^n - \frac{\frac{z_{i+\frac{3}{2},j-\frac{1}{2}} + z_{i+\frac{3}{2},j+\frac{1}{2}}}{2} - \frac{z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}}{2}}{\Delta x} \right) \\
&- \frac{\Delta x}{16} \left(H_{x|i+1,j+1}^n - \frac{\frac{z_{i+\frac{3}{2},j+\frac{1}{2}} + z_{i+\frac{3}{2},j+\frac{3}{2}}}{2} - \frac{z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{3}{2}}}{2}}{\Delta x} \right) \\
&+ \frac{\Delta y}{16} \left(H_{y|i,j}^n - \frac{\frac{z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}}{2} - \frac{z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j-\frac{1}{2}}}{2}}{\Delta y} \right) \\
&+ \frac{\Delta y}{16} \left(H_{y|i+1,j}^n - \frac{\frac{z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{3}{2},j+\frac{1}{2}}}{2} - \frac{z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{3}{2},j-\frac{1}{2}}}{2}}{\Delta y} \right) \\
&- \frac{\Delta y}{16} \left(H_{y|i,j+1}^n - \frac{\frac{z_{i-\frac{1}{2},j+\frac{3}{2}} + z_{i+\frac{1}{2},j+\frac{3}{2}}}{2} - \frac{z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}}{2}}{\Delta y} \right) \\
&- \frac{\Delta y}{16} \left(H_{y|i+1,j+1}^n - \frac{\frac{z_{i+\frac{1}{2},j+\frac{3}{2}} + z_{i+\frac{3}{2},j+\frac{3}{2}}}{2} - \frac{z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{3}{2},j+\frac{1}{2}}}{2}}{\Delta y} \right) \tag{50}
\end{aligned}$$

Since at time t^n the steady state constraint ($H_{i,j}^n = h_{i,j}^n + z_{i,j} = \text{constant}$) is maintained, then all numerical partial derivatives of $H_{i,j}^n$ in equation (50) are zero. Furthermore, equation (35) leads to

$$\begin{aligned}
&(z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}) - (z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{1}{2}}) \\
&\quad = 2 \left((z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}) - 2z_{i,j} \right) \\
&(z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{3}{2}}) - (z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{3}{2}}) \\
&\quad = 2 \left((z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{3}{2}}) - 2z_{i,j+1} \right) \\
&(z_{i+\frac{3}{2},j-\frac{1}{2}} + z_{i+\frac{3}{2},j+\frac{1}{2}}) - (z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}) \\
&\quad = 2 \left(2z_{i+1,j} - (z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}) \right) \\
&(z_{i+\frac{3}{2},j+\frac{1}{2}} + z_{i+\frac{3}{2},j+\frac{3}{2}}) - (z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{3}{2}}) \\
&\quad = 2 \left(2z_{i+1,j+1} - (z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{3}{2}}) \right) \tag{51}
\end{aligned}$$

and similarly we have the following equations

$$\begin{aligned}
& (z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}) - (z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j-\frac{1}{2}}) \\
& \qquad \qquad \qquad = 2 \left((z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}) - 2z_{i,j} \right) \\
& (z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{3}{2},j+\frac{1}{2}}) - (z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{3}{2},j-\frac{1}{2}}) \\
& \qquad \qquad \qquad = 2 \left((z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{3}{2},j+\frac{1}{2}}) - 2z_{i+1,j} \right) \\
& (z_{i-\frac{1}{2},j+\frac{3}{2}} + z_{i+\frac{1}{2},j+\frac{3}{2}}) - (z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}) \\
& \qquad \qquad \qquad = 2 \left(2z_{i,j+1} - (z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}) \right) \\
& (z_{i+\frac{1}{2},j+\frac{3}{2}} + z_{i+\frac{3}{2},j+\frac{3}{2}}) - (z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{3}{2},j+\frac{1}{2}}) \\
& \qquad \qquad \qquad = 2 \left(2z_{i+1,j+1} - (z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{3}{2},j+\frac{1}{2}}) \right)
\end{aligned} \tag{52}$$

Substituting the identities we obtain

$$\begin{aligned}
h_{i+\frac{1}{2},j+\frac{1}{2}}^n &= \frac{1}{4} \left(h_{i,j}^n + h_{i+1,j}^n + h_{i,j+1}^n + h_{i+1,j+1}^n \right) \\
& - \frac{2}{32} \left((z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}) - 2z_{i,j} \right) \\
& - \frac{2}{32} \left((z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{3}{2}}) - 2z_{i,j+1} \right) \\
& + \frac{2}{32} \left(2z_{i+1,j} - (z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}) \right) \\
& + \frac{2}{32} \left(2z_{i+1,j+1} - (z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{3}{2}}) \right) \\
& - \frac{2}{32} \left((z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}) - 2z_{i,j} \right) \\
& - \frac{2}{32} \left((z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{3}{2},j+\frac{1}{2}}) - 2z_{i+1,j} \right) \\
& + \frac{2}{32} \left(2z_{i,j+1} - (z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}) \right) \\
& + \frac{2}{32} \left(2z_{i+1,j+1} - (z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{3}{2},j+\frac{1}{2}}) \right)
\end{aligned}$$

which simplifies leading to

$$\begin{aligned}
h_{i+\frac{1}{2},j+\frac{1}{2}}^n &= \frac{1}{4} \left(h_{i,j}^n + h_{i,j+1}^n + h_{i+1,j}^n + h_{i+1,j+1}^n \right) \\
&+ \frac{1}{4} \left(z_{i,j} + z_{i,j+1} + z_{i+1,j} + z_{i+1,j+1} \right) \\
&- \frac{1}{8} \left(4z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{3}{2}} + z_{i+\frac{1}{2},j-\frac{1}{2}} \right. \\
&\left. + z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{3}{2},j+\frac{1}{2}} \right)
\end{aligned} \tag{53}$$

Similarly, for the back-projection step, we follow the surface gradient method and discretize the water height in terms of the water level $\widetilde{H}_{i+1/2,j+1/2}^{n+1} = h_{i+1/2,j+1/2}^{n+1} + \widetilde{z}_{i+1/2,j+1/2}$ as follows.

$$\begin{aligned}
h_{i,j}^{n+1} &= \frac{1}{4} \left(h_{i-\frac{1}{2},j-\frac{1}{2}}^{n+1} + h_{i+\frac{1}{2},j-\frac{1}{2}}^{n+1} + h_{i-\frac{1}{2},j+\frac{1}{2}}^{n+1} + h_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} \right) \\
&+ \frac{\Delta x}{16} \left(\left(\widetilde{H}_x \right)_{i-\frac{1}{2},j-\frac{1}{2}}^{n+1} - \frac{z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j-\frac{1}{2}}}{2} - \frac{z_{i-\frac{3}{2},j-\frac{1}{2}} + z_{i-\frac{1}{2},j-\frac{1}{2}}}{2} \right) \\
&+ \frac{\Delta x}{16} \left(\left(\widetilde{H}_x \right)_{i-\frac{1}{2},j+\frac{1}{2}}^{n+1} - \frac{z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}}{2} - \frac{z_{i-\frac{3}{2},j+\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{1}{2}}}{2} \right) \\
&- \frac{\Delta x}{16} \left(\left(\widetilde{H}_x \right)_{i+\frac{1}{2},j-\frac{1}{2}}^{n+1} - \frac{z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{3}{2},j-\frac{1}{2}}}{2} - \frac{z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j-\frac{1}{2}}}{2} \right) \\
&- \frac{\Delta x}{16} \left(\left(\widetilde{H}_x \right)_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} - \frac{z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{3}{2},j+\frac{1}{2}}}{2} - \frac{z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}}{2} \right) \\
&+ \frac{\Delta y}{16} \left(\left(\widetilde{H}_y \right)_{i-\frac{1}{2},j-\frac{1}{2}}^{n+1} - \frac{z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{1}{2}}}{2} - \frac{z_{i-\frac{1}{2},j-\frac{3}{2}} + z_{i-\frac{1}{2},j-\frac{1}{2}}}{2} \right) \\
&+ \frac{\Delta y}{16} \left(\left(\widetilde{H}_y \right)_{i+\frac{1}{2},j-\frac{1}{2}}^{n+1} - \frac{z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}}{2} - \frac{z_{i+\frac{1}{2},j-\frac{3}{2}} + z_{i+\frac{1}{2},j-\frac{1}{2}}}{2} \right) \\
&- \frac{\Delta y}{16} \left(\left(\widetilde{H}_y \right)_{i-\frac{1}{2},j+\frac{1}{2}}^{n+1} - \frac{z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{3}{2}}}{2} - \frac{z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{1}{2}}}{2} \right) \\
&- \frac{\Delta y}{16} \left(\left(\widetilde{H}_y \right)_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} - \frac{z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{3}{2}}}{2} - \frac{z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}}{2} \right)
\end{aligned} \tag{54}$$

Note that when $H_{i,j} = h_{i,j}^n + z_{i,j} = H = \text{constant}$, then $\widetilde{H}_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} = h_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} + \widetilde{z}_{i+\frac{1}{2},j+\frac{1}{2}} = H$ remains constant and therefore $(\widetilde{H}_x)_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} = (\widetilde{H}_y)_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} = 0$; equation (54) becomes.

$$\begin{aligned}
h_{i,j}^{n+1} = & H - \frac{1}{4} \left(\tilde{z}_{i-\frac{1}{2},j-\frac{1}{2}} + \tilde{z}_{i+\frac{1}{2},j-\frac{1}{2}} + \tilde{z}_{i-\frac{1}{2},j+\frac{1}{2}} + \tilde{z}_{i+\frac{1}{2},j+\frac{1}{2}} \right) \\
& - \frac{1}{32} \left[(z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j-\frac{1}{2}}) - (z_{i-\frac{3}{2},j-\frac{1}{2}} + z_{i-\frac{1}{2},j-\frac{1}{2}}) \right] \\
& - \frac{1}{32} \left[(z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}) - (z_{i-\frac{3}{2},j+\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{1}{2}}) \right] \\
& + \frac{1}{32} \left[(z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{3}{2},j-\frac{1}{2}}) - (z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j-\frac{1}{2}}) \right] \\
& + \frac{1}{32} \left[(z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{3}{2},j+\frac{1}{2}}) - (z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}) \right] \\
& - \frac{1}{32} \left[(z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{1}{2}}) - (z_{i-\frac{1}{2},j-\frac{3}{2}} + z_{i-\frac{1}{2},j-\frac{1}{2}}) \right] \\
& - \frac{1}{32} \left[(z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}) - (z_{i+\frac{1}{2},j-\frac{3}{2}} + z_{i+\frac{1}{2},j-\frac{1}{2}}) \right] \\
& + \frac{1}{32} \left[(z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{3}{2}}) - (z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{1}{2}}) \right] \\
& + \frac{1}{32} \left[(z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{3}{2}}) - (z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}}) \right]
\end{aligned} \tag{55}$$

which reduces to

$$\begin{aligned}
h_{i,j}^{n+1} = & H - \frac{1}{4} \left(\tilde{z}_{i-\frac{1}{2},j-\frac{1}{2}} + \tilde{z}_{i+\frac{1}{2},j-\frac{1}{2}} + \tilde{z}_{i-\frac{1}{2},j+\frac{1}{2}} + \tilde{z}_{i+\frac{1}{2},j+\frac{1}{2}} \right) \\
& + \frac{1}{32} \left[z_{i-\frac{3}{2},j-\frac{1}{2}} + z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i-\frac{3}{2},j+\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{1}{2}} \right] \\
& + \frac{1}{32} \left[z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{3}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{3}{2},j+\frac{1}{2}} \right] \\
& - \frac{1}{32} \left[z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}} \right] \\
& - \frac{1}{32} \left[z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}} \right] \\
& - \frac{1}{32} \left[z_{i-\frac{1}{2},j-\frac{3}{2}} + z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j-\frac{3}{2}} + z_{i+\frac{1}{2},j-\frac{1}{2}} \right] \\
& + \frac{1}{32} \left[z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{3}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{3}{2}} \right] \\
& - \frac{1}{32} \left[z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}} \right] \\
& - \frac{1}{32} \left[z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}} \right]
\end{aligned}$$

Taking into account that

$$\begin{aligned}
z_{i,j} &= \frac{1}{4} \left[z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}} \right] \\
z_{i-1,j} &= \frac{1}{4} \left[z_{i-\frac{3}{2},j-\frac{1}{2}} + z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i-\frac{3}{2},j+\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{1}{2}} \right] \\
z_{i+1,j} &= \frac{1}{4} \left[z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{3}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{3}{2},j+\frac{1}{2}} \right]
\end{aligned}$$

$$z_{i,j-1} = \frac{1}{4} \left[z_{i-\frac{1}{2},j-\frac{3}{2}} + z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j-\frac{3}{2}} + z_{i+\frac{1}{2},j-\frac{1}{2}} \right]$$

$$z_{i,j+1} = \frac{1}{4} \left[z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{3}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{3}{2}} \right]$$

Thus we obtain

$$h_{i,j}^{n+1} = H - \frac{1}{4} \left(\tilde{z}_{i-\frac{1}{2},j-\frac{1}{2}} + \tilde{z}_{i+\frac{1}{2},j-\frac{1}{2}} + \tilde{z}_{i-\frac{1}{2},j+\frac{1}{2}} + \tilde{z}_{i+\frac{1}{2},j+\frac{1}{2}} \right) + \frac{1}{8} (z_{i-1,j} + z_{i+1,j} + z_{i,j-1} + z_{i,j+1} - 4z_{i,j}) \quad (56)$$

On the other hand we know from equation (40) that

$$\tilde{z}_{i+1/2,j+1/2} = \frac{1}{8} \left(4z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{3}{2}} + z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{3}{2},j+\frac{1}{2}} \right)$$

$$\tilde{z}_{i-1/2,j-1/2} = \frac{1}{8} \left(4z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i-\frac{1}{2},j-\frac{3}{2}} + z_{i-\frac{3}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j-\frac{1}{2}} \right)$$

$$\tilde{z}_{i-1/2,j+1/2} = \frac{1}{8} \left(4z_{i-\frac{1}{2},j+\frac{1}{2}} + z_{i-\frac{1}{2},j+\frac{3}{2}} + z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i-\frac{3}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}} \right)$$

$$\tilde{z}_{i+1/2,j-1/2} = \frac{1}{8} \left(4z_{i+\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{1}{2},j+\frac{1}{2}} + z_{i+\frac{1}{2},j-\frac{3}{2}} + z_{i-\frac{1}{2},j-\frac{1}{2}} + z_{i+\frac{3}{2},j-\frac{1}{2}} \right)$$

Adding together and using equations (35), we obtain

$$\tilde{z}_{i-\frac{1}{2},j-\frac{1}{2}} + \tilde{z}_{i+\frac{1}{2},j-\frac{1}{2}} + \tilde{z}_{i-\frac{1}{2},j+\frac{1}{2}} + \tilde{z}_{i+\frac{1}{2},j+\frac{1}{2}} = \frac{1}{8} [4z_{i,j-1} + 4z_{i-1,j} + 4z_{i,j+1} + 4z_{i+1,j} + 16z_{i,j}] \quad (57)$$

Substituting, we obtain

$$h_{i,j}^{n+1} = H - \frac{1}{4} \frac{1}{8} [4z_{i,j-1} + 4z_{i-1,j} + 4z_{i,j+1} + 4z_{i+1,j} + 16z_{i,j}] + \frac{1}{8} (z_{i,j-1} + z_{i-1,j} + z_{i,j+1} + z_{i+1,j} - 4z_{i,j}) = H - z_{i,j} = h_{i,j}^n. \blacksquare \quad (58)$$

$$(59)$$

Note that the proof of $(h\theta)_{i,j}^{n+1} = (h\theta)_{i,j}^n$ follows exactly the same steps as above since in the steady state 4 we have $\theta_{i,j}^n = \text{constant}$ for all i, j .

Acknowledgement

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions.

References

- [1] F. Alcrudo, P. Garcia-Navarro, A high resolution godunov-type scheme in finite volumes for the 2d shallow water equations, *Int. J. Numer. Methods Fluids* 16 (1993) 489–505.
- [2] A. Arminjon, M. Viallon, A. Madrane, A finite volume extension of the lax-friedrichs and nessyahu-tadmor schemes for conservation laws on unstructured grids, *Int. J. Comp. Fluid Dynamics* 9 (1997) 1–22.
- [3] F. Bouchut, *Nonlinear Stability of Finite Volume Methods for Hyperbolic Conservation Laws: And Well-Balanced Schemes for Sources*, Springer, 2004.
- [4] A. Canestrelli, A. Siviglia, M. Dumbser, E. Toro, Well-balanced high-order centered schemes for non-conservative hyperbolic systems. applications to shallow water equations with fixed and mobile bed, *Advances in Water Resources* 36 (2) (2009) 834–844.
- [5] M. Castro, J. Gallardo, C. Parés, High order finite volume schemes based on reconstruction of states for solving hyperbolic systems with nonconservative products. applications to shallow-water systems, *Math. Comp* 75 (1) (2006) 1103–1134.
- [6] A. Chertock, A. Kurganov, Y. Liu, Central-upwind schemes for the system of shallow water equations with horizontal temperature gradients, *Numerische Mathematik* (2013) 1–45.
- [7] N. Crnjarić-žic, S. Vuković, L. Sopta, Balanced central nt schemes for the shallow water equations, *Conf. App. Math. Sc. Comp* 75 (1) (2005) 171.
- [8] P. Dellar, Common hamiltonian structure of the shallow water equations with horizontal temperature gradients and magnetic fields, *Phys. Fluids* 303 (2003) 292–297.
- [9] V. Desveaux, *Contribution à l’approximation numérique des systèmes hyperboliques*, Ph.D. thesis, Université de Nantes, France (2013).
- [10] V. Desveaux, C. Zenk, C. Berthon, C. Klingenberg, Well-balanced scheme to capture non-explicit steady states. part 1: Ripa model, submitted to *Mathematics of Computation* (2014).
- [11] G. Jiang, D. Levy, C. Lin, S. Osher, E. Tadmor, High-resolution non-oscillatory central schemes with non-staggered grids for hyperbolic conservation laws, *SIAM Journal on Numerical Analysis* 35 (1998) 2147–2168.

- [12] G. Jiang, E. Tadmor, Non-oscillatory central schemes for multidimensional hyperbolic conservation laws, *Siam J. Sc. Comp.* 19 (1998) 1892–1917.
- [13] S. F. Liotta, V. Romano, G. Russo, Central schemes for the balance laws of relaxation type, *SIAM J. Num. Anal.* 38 (4) (2000) 1337–1356.
- [14] H. Nessyahu, E. Tadmor, Non-oscillatory central differencing for hyperbolic conservation laws, *J. Comp. Phys.* 87 (2) (1990) 408–463.
- [15] S. Noelle, N. Pankratz, G. Puppo, J. Natvig, Well-balanced finite volume schemes of arbitrary order of accuracy for shallow water flows, *J. Comp. Phys.* 213 (2006) 474–499.
- [16] S. Noelle, Y. Xing, C. Shu, High order well-balanced finite volume weno schemes for shallow water equation with moving water, *J. Comp. Phys.* 226 (2007) 29–58.
- [17] P. Ripa, Conservation laws for primitive equations models with inhomogeneous layers, *Geophys. Astrophys. Fluid Dynam.* 70 (1993) 85–111.
- [18] P. Ripa, On improving a one-layer ocean model with thermodynamics, *J. Fluid Mech.* 303 (1995) 169–201.
- [19] R. Touma, Central unstaggered finite volume schemes for hyperbolic systems: Applications to unsteady shallow water equations, *App. Math. Comp* 213 (1) (2009) 47–59.
- [20] R. Touma, P. Arminjon, Central finite volume schemes with cts magnetic field divergence treatment for 2- and 3-d ideal mhd, *J. Comp. Phys.* 212 (2006) 617–636.
- [21] R. Touma, S. Khankan, Well-balanced unstaggered central schemes for one and two-dimensional shallow water equation systems, *App. Math. Comp* 218 (1) (2012) 5948–5960.
- [22] B. Van Leer, Towards the ultimate conservative difference scheme iii. upstream-centered finite-difference schemes for ideal compressible flow, *J. Comp. Phys.* 23 (3) (1977) 263–275.
- [23] M. E. Vázquez-Cédon, Improved treatment of source terms in upwind schemes for the shallow water equations in channel with irregular geometry, *J. Comp. Phys.* 148 (1999) 497–526.
- [24] J. G. Zhou, D. M. Causon, C. G. Mingham, D. M. Ingram, The surface gradient method for the treatment of source terms in the shallow water equations, *J. Comp. Phys.* 168 (2001) 1–25.