Well-balanced bicharacteristic-based scheme for multilayer shallow water flows including wet/dry fronts[1]

M. Dudzinski[2], M. Lukáčová-Medviďová[3]

Abstract

The aim of this paper is to present a new well-balanced finite volume scheme for two-dimensional multilayer shallow water flows including wet/dry fronts. The ideas, presented here for the two-layer model, can be generalized to a multilayer case in a straightforward way. The method developed here is constructed in the framework of the Finite Volume Evolution Galerkin (FVEG) schemes. The FVEG methods couple a finite volume formulation with evolution operators. The latter are constructed using the bicharacteristics of multidimensional hyperbolic systems. However, in the case of multilayer shallow water flows the required eigenstructure of the underlying equations is not readily available. Thus we approximate the evolution operators numerically. This approximation procedure can be used for arbitrary hyperbolic systems. We derive a well-balanced approximation of the evolution operators and prove that the FVEG scheme is well-balanced for the multilayer lake at rest states even in the presence of wet/dry fronts. Several numerical experiments confirm the reliability and efficiency of the new well-balanced FVEG scheme.

1 Introduction

The shallow water equations are widely used for nearly horizontal flows with constant density. They can be derived as a depth-averaged form of the Navier-Stokes equations, see, e.g. [32]. However for modeling of stratified flows at least two layers of fluids have to

[1]The authors would like to thank A. Kurganov (Tulane University) for a fruitful discussion on the topic and to M.J. Castro (University of Malaga) for providing numerical results obtained by the path-consistent IFCP method. They are also grateful to referees for their remarks that helped to improve the presentation of results. This work was partially supported by the German Research Foundation DFG under the grants #INST 247/609-1 and #LU 1470/2-2.

[2]Institute of Numerical Simulation, Hamburg University of Technology, Schwarzenbergstraße 95 E, 21073 Hamburg, Germany e-mail: michael.dudzinski@tu-harburg.de

[3]Institute of Mathematics, Johannes Gutenberg University, Staudingerweg 9, 55128 Mainz, Germany, e-mail: lukacova@uni-mainz.de

be distinguished. Such flows arise typically in oceanography for example at the Strait of Gibraltar. Here the upper layer corresponds to the lighter Atlantic whereas the lower one to the denser Mediterranean water. The upper and lower layer are mixed developing an intermediate layer of mixed water. In a simplified mathematical model one can omit the intermediate layer that yields the two-layer shallow water system. This system belongs to a class of general balance laws in a non-conservative form, i.e.

$$W_t + \operatorname{div} F(W) = B(W) + C_0(W)\nabla\sigma(x) + \sum_{l=1}^{n} C_l(W)\nabla W_l. \tag{1}$$

The term $\sum_{l=1}^{n} C_l(W)\nabla W_l$ describes the non-conservative part. Indeed, it becomes conservative as soon as the columns of $\sum_l C_l$ have potentials. Moreover, if W is discontinuous the product does not have sense even in the distributional form and further clarification is necessary, cf. [19, 37]. We assume that $F := (f_1, \ldots, f_d)$, σ and $B, C_0, \ldots, C_n$ are smooth functions on their respective domains.

We are interested in a numerical approximation of the solution of (1) on $\Omega \subseteq \mathbb{R}^d$ in the time interval $T := (t_0, t_f]$ subject to some initial and boundary conditions. The conservative variables $W : \Omega \times T \to \mathbb{R}^n$ are considered to take their values in an open and convex subspace of $\mathbb{R}^n$. In the special case $\Omega \subseteq \mathbb{R}^2$ we will write the coordinate vector as $(x, y)^T$ instead of $(x_1, x_2)^T$.

Taking earth's rotation and bottom topography into account the aforementioned flows can be modeled by the following system of two-layer shallow water equations

$$W := \begin{pmatrix} h_1 \\ h_1u_1 \\ h_1v_1 \\ h_2 \\ h_2u_2 \\ h_2v_2 \end{pmatrix}, \qquad F(W) := \begin{bmatrix} h_1u_1 & h_1v_1 \\ h_1u_1^2 + \frac{g}{2}h_1^2 & h_1u_1v_1 \\ h_1u_1v_1 & h_1v_1^2 + \frac{g}{2}h_1^2 \\ h_2u_2 & h_2v_2 \\ h_2u_2^2 + \frac{g}{2}h_2^2 & h_2u_2v_2 \\ h_2u_2v_2 & h_2v_2^2 + \frac{g}{2}h_2^2 \end{bmatrix},$$

$$B(W) := \begin{pmatrix} 0 \\ fh_1v_1 \\ -fh_1u_1 \\ 0 \\ fh_2v_2 \\ -fh_2u_2 \end{pmatrix}, \qquad C_0(W) := \begin{bmatrix} 0 & 0 \\ -gh_1 & 0 \\ 0 & -gh_1 \\ 0 & 0 \\ -gh_2 & 0 \\ 0 & -gh_2 \end{bmatrix}, \tag{2}$$

$$C_1(W) := \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -rgh_2 & 0 \\ 0 & -rgh_2 \end{bmatrix}, \qquad C_4(W) := \begin{bmatrix} 0 & 0 \\ -gh_1 & 0 \\ 0 & -gh_1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

with $\sigma := b$ and $C_2 = C_3 = C_5 = C_6 = 0$. Denoting by $\mathcal{S}$ the physical source term due to the Coriolis forces and the bottom topography, $\mathcal{S} = B(W) + C_0(W)\nabla\sigma(x)$, and by

$\mathcal{C}(W)$ the coefficients of nonconservative terms, $(\mathcal{C}(W)\cdot\nabla)W = \sum_{l=1}^{n} C_l(W)\nabla W$, then the system (1) can be also rewritten in a more compact form

$$W_t + \operatorname{div} F(W) = (\mathcal{C}(W)\cdot\nabla)\, W + \mathcal{S}(x, W).$$

Equivalently, we have

$$\begin{cases} \partial_t h_1 + \operatorname{div}(h_1\vec{u}_1) = 0, \\ \partial_t(h_1\vec{u}_1) + \operatorname{div}\left(h_1\vec{u}_1\vec{u}_1^T\right) + gh_1\nabla(h_1 + h_2 + b) + fh_1\vec{u}_1^{\perp} = 0, \\ \partial_t h_2 + \operatorname{div}(h_2\vec{u}_2) = 0, \\ \partial_t(h_2\vec{u}_2) + \operatorname{div}\left(h_2\vec{u}_2\vec{u}_2^T\right) + gh_2\nabla(rh_1 + h_2 + b) + fh_2\vec{u}_2^{\perp} = 0, \end{cases} \tag{3}$$

Here $h_i, h_i\vec{u}_i := h_i \begin{pmatrix} u_i \\ v_i \end{pmatrix}$ represent the depth and discharge of i-th layer, respectively, whereas $\vec{u}_i^{\perp} := \begin{pmatrix} -v_i \\ u_i \end{pmatrix}$ stands for counterclockwise rotated vector orthogonal to $\vec{u}_i$. Subindex $i = 1$ stays for the upper, whereas $i = 2$ for the lower layer, respectively. Function b denotes the time independent bottom topography, f is the Coriolis parameter and g the gravitational constant. Set $r := \frac{\rho_1}{\rho_2}$ with ρ_i being the density of layer i and $\rho_1 < \rho_2$. The situation is depicted in Figure 1.
In (3) we can notice three types of source terms. The term $h_i\nabla b$ arise from a bottom

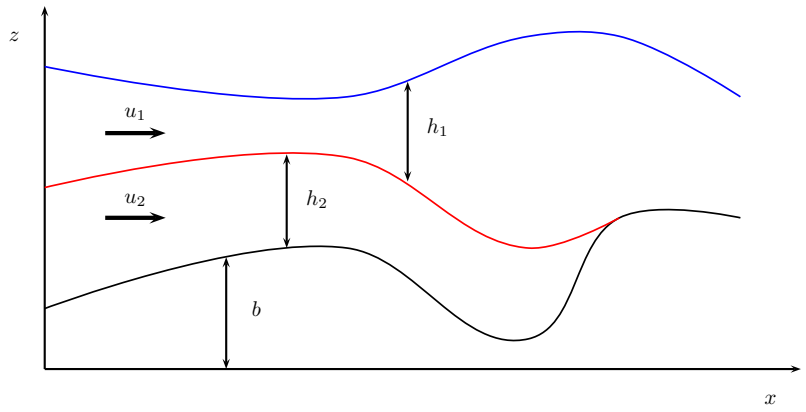


Figure 1: Two-layer shallow water model.

elevation. This term is well-defined as long as b is smooth enough. This means that the products $h_i\partial_x b$ and $h_i\partial_y b$ are independent of the path chosen to represent discontinuities in h_1 or h_2. Indeed, these products make then sense in the distributional framework. The non-conservative pressure terms $h_1\nabla h_2$, $h_2\nabla h_1$ are due to the presence of an additional layer. The effect of the Coriolis forces is modeled by $fh_i\vec{u}_i^{\perp}$.
Summing up the momentum equations one obtains a conservative equation for the total

momentum.

$$\partial_t(rh_1u_1 + h_2u_2) + \operatorname{div}(rh_1\vec{u}_1\vec{u}_1^T + h_2\vec{u}_2\vec{u}_2^T) + \frac{1}{2}g\nabla\left(rh_1^2 + rh_1h_2 + h_2^2\right) = \\ -g(rh_1 + h_2)\nabla b - f(rh_1\vec{u}_1^{\perp} + h_2\vec{u}^{\perp}). \qquad (4)$$

However, the non-conservativity of the individual layers makes it nontrivial to give a sense to the weak solution of (3). In [19] a definition of non-conservative products in terms of a family of paths chosen in the state space is given in one spatial dimension. Thus assuming a path is a-priori available a formulation of weak solutions and the Rankine-Hugoniot condition can be deduced. Note, however, that the problem to choose the right path, which should take additional physical considerations into account, can be rather involved.
Using the concept of a weak solution for non-conservative hyperbolic systems the so-called path-conservative schemes have been derived by Parés et al., cf. [34, 35, 16, 36] and the references therein. We would like to mention that the effect of numerical viscosity and its interplay with the non-conservative products is a very delicate problem, which has been recently discussed in the literature extensively, see, e.g., [2, 10, 36]. Further approaches to approximate numerically multilayer shallow water system can be found, e.g., in [1, 4, 5, 7, 8, 13, 23].
The paper is organized as follows. In the next section the structure of the two layer shallow water model will be investigated. Then we describe the numerical scheme in the Section 3 beginning with the Finite Volume Evolution Galerkin framework. Hereafter the discretization of the fluxes and sources are formulated. To have a self-consistent paper, a brief overview of the theory of bicharacteristics is given in Subsection 3.3. The Subsection 3.4 concentrates on the numerical approximation of the evolution operators. In the Subsections 3.5-3.8 the well-balancing and the positivity preserving property of our scheme is discussed. The last section is devoted to numerical experiments.

2 Two-layer shallow water equations

2.1 The system

For further investigations we rewrite (1) in a quasilinear form. After a possible change of variables, $W = T(V)$, this form reads

$$V_t + \sum_{i=1}^{d} \underbrace{\left(J_{f_i} - \left(\sum_{j=1}^{n} \tilde{C}_j(V) \right)_i \right)}_{A_i :=} V_{x_i} = \tilde{B}(V) + \tilde{C}(V)\nabla\sigma =: s(\nabla\sigma, V), \qquad (5)$$

where the non-conservative products are added to the Jacobian matrices of the fluxes. In the particular case of (3) we get

$$
\begin{aligned}
&V_t + A_1 V_x + A_2 V_y = s(\nabla b, V),\\
&V = \begin{pmatrix} \varepsilon \\ \vec{u}_1 \\ \omega \\ \vec{u}_2 \end{pmatrix}, \quad A_1 = \begin{pmatrix} u_1 & \varepsilon-\omega & 0 & u_2-u_1 & \omega-b & 0 \\ g & u_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & u_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & u_2 & \omega-b & 0 \\ rg & 0 & 0 & (1-r)g & u_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & u_2 \end{pmatrix},\\
&s = \begin{pmatrix} \vec{u}_2^T \nabla b \\ -f\vec{u}_1^{\perp} \\ \vec{u}_2^T \nabla b \\ -f\vec{u}_2^{\perp} \end{pmatrix}, \quad A_2 = \begin{pmatrix} v_1 & 0 & \varepsilon-\omega & v_2-v_1 & 0 & \omega-b \\ 0 & v_1 & 0 & 0 & 0 & 0 \\ g & 0 & v_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & v_2 & 0 & \omega-b \\ 0 & 0 & 0 & 0 & v_2 & 0 \\ rg & 0 & 0 & (1-r)g & 0 & v_2 \end{pmatrix}.
\end{aligned} \tag{6}
$$

Here $\omega := h_2 + b$, $\varepsilon := h_1 + \omega$ and the components of V are the so called equilibrium variables for the lake at rest. The reader should be aware that we do not assume ε to be small.
As long as V is smooth system (6) is equivalent to (3), however, (6) is more suitable for the derivation of evolution operators and their numerical approximation. In particular, for the lake at rest situation the velocities vanish. Thus the source term s becomes zero and we are able to show the well-balance property for our scheme in an elegant way, see Subsection 3.5.

2.1.1 Eigenvalues

System (5) is said to be (strictly) hyperbolic if for any vector $\eta := (\eta_1, \dots, \eta_d)^T \in S^{d-1}$ the matrix pencil $A(\eta) := \sum_{i=1}^{d} \eta_i A_i$ has only real (and distinct) eigenvalues and a full set of corresponding linearly independent eigenvectors. The eigenvalues are then the roots of the characteristic polynomial $\det(A(\eta) - \lambda I)$, with I the $n \times n$ identity matrix. In the particular case of (6) the characteristic polynomial reads

$$
\begin{aligned}
p(\lambda, \eta) = \quad & p_1(\lambda, \eta) \cdot p_2(\lambda, \eta) \\
= \quad & \underbrace{(\eta^T \vec{u}_1 - \lambda)(\eta^T \vec{u}_2 - \lambda)}_{=:p_1(\lambda,\eta)} \\
& \underbrace{\left[\left((\eta^T \vec{u}_1 - \lambda)^2 - \|\eta\|_2^2 g h_1\right)\left((\eta^T \vec{u}_2 - \lambda)^2 - \|\eta\|_2^2 g h_2\right) - \|\eta\|_2^4 r g^2 h_1 h_2\right]}_{=:p_2(\lambda,\eta)}
\end{aligned} \tag{7}
$$

The roots of p_1, $\lambda_{2/5} := \eta^T \vec{u}_{1/2}$, generate linearly degenerated characteristic fields (cf. (9)). For p_2 the roots have a complicated structure and are useless for most analytical considerations. However, it can be shown that p_2 has two real and two possibly complex roots. For this to see note that $\deg(p_2) = 4$ with leading order coefficient being one and

$p_2(\eta^T \vec{u}_i \pm \sqrt{gh_i}) < 0$. Next, consider $\lambda_{max}(\eta)$ to be the local maximum of p_2, then by elementary calculus we get

$$(6) \text{ is hyperbolic} \quad \Longleftrightarrow \quad \forall \eta \,:\, p_2(\lambda_{max}(\eta), \eta) > 0.$$

Since p_2 is a polynomial of degree four its roots can in general be computed, see e.g. [41]. Thus setting

$$\begin{aligned}
T &:= \frac{1}{3}\left(\left(\eta^T(\vec{u}_1 - \vec{u}_2)\right)^2 - g(h_1 + h_2)\right), \\
C_1 &:= T + g(h_1 + h_2) \geq 0, \\
C_2 &:= T^2 + \frac{4}{3}g^2 h_1 h_2 (1 - r) \geq 0, \\
C_3 &:= T^3 - 2g^2 h_1 h_2 \left[(2 + r)T + rg(h_1 + h_2)\right], \\
D &:= \sqrt{C_1 + \left(\sqrt{C_3^2 - C_2^3} + C_3\right)^{1/3} + \frac{C_2}{\left(\sqrt{C_3^2 - C_2^3} + C_3\right)^{1/3}}}, \\
K_{1/2} &:= \sqrt{3C_1 - D^2 \mp \frac{2g(h_1 - h_2)(\eta^T(\vec{u}_1 - \vec{u}_2))}{D}},
\end{aligned}$$

we get the roots of p_2 in a closed form

$$\begin{aligned}
\lambda_1 &= \frac{1}{2}\eta^T(\vec{u}_1 + \vec{u}_2) - \frac{1}{2}(D + K_1), \\
\lambda_3 &= \frac{1}{2}\eta^T(\vec{u}_1 + \vec{u}_2) + \frac{1}{2}(D + K_2), \\
\lambda_4 &= \frac{1}{2}\eta^T(\vec{u}_1 + \vec{u}_2) - \frac{1}{2}(D - K_1), \\
\lambda_6 &= \frac{1}{2}\eta^T(\vec{u}_1 + \vec{u}_2) + \frac{1}{2}(D - K_2).
\end{aligned}$$

The eigenvalues $\lambda_{4/6}$ are those that possibly become complex. Thus it is clear that

$$\Im(D) = -\Im\left(K_{1/2}\right),$$

where $\Im$ denotes the imaginary part of D. This proves the following

Lemma 2.1
The two-layer shallow water system (6) *is hyperbolic if and only if*

$$\forall \eta \,:\, \Im(D) = 0, D \neq 0.$$

There seems to be a well accepted indicator in one spatial dimension for when system (3) is hyperbolic

$$(u_1 - u_2)^2 < (1 - r)g(h_1 + h_2).$$

However, this inequality is derived by assuming $1 - r$ and $u_1 - u_2$ to be small and expanding p_2 using these quantities up to order one. In contrast Ovsjanikov [33] has analyzed the two-layer system by rewriting p_2 to a simpler form and revealed that it is indeed also hyperbolic when the difference in velocities is "big enough". In particular, if $r = 1$, it follows that p_2 has only three distinct roots if $\eta^T(\vec{u}_2 - \vec{u}_1) = 0$, two if $0 < |\eta^T(\vec{u}_2 - \vec{u}_1)| < C$ and four if $|\eta^T(\vec{u}_2 - \vec{u}_1)| \geq C$, for some constant $C > 0$. This behavior can also be seen when considering D.
The roots of p_1 describe the speed of advection waves. The two real roots of p_2 are connected to the fast external gravitational waves and bound the other four eigenvalues, provided the latter are all real. The possibly complex eigenvalues are connected to the internal gravitational waves and in the real case they are bounded by $\eta^T\vec{u}_i \pm \sqrt{gh_i}$. The situation where the internal eigenvalues become complex is linked to the Kelvin-Helmholtz instabilities. Indeed, in this case the distinct layers begin to mix. There are several approaches how to deal with such situations. In [7] the authors split the system to obtain two hyperbolic systems of shallow water equations that have only real roots. In [13] additional friction terms simulate the loss of momentum due to the friction effects between layers. In [12] the authors introduce an additional layer at the interface with density defined by a particular superposition of the densities of the starting layers to model the mixing effects. In the present paper we will only deal with the situation of immiscible layers and flows which are hyperbolic.
Generalization of the first order approximations from [38] in $\eta^T(\vec{u}_2 - \vec{u}_1)$ and $1 - r$ of the external eigenvalues are given by

$$\lambda_{ext}^{\pm} := u_m \pm \sqrt{g(h_1 + h_2)}, \qquad u_m := \eta^T \frac{h_1\vec{u}_1 + h_2\vec{u}_2}{h_1 + h_2}. \tag{8}$$

Note that these approximations are only valid when $\vec{u}_1 \approx \vec{u}_2$ and $r \approx 1$.
To complete the discussion we present without a proof the so-called ray velocities, that will be needed in the Section 3.3. Let us recall that the ray velocities are the velocities with which the point on the bicharacteristic moves in the (x, y) plane.

$$\begin{aligned}
\nabla_\eta\lambda_1 &= m_{P,1} - \frac{1}{2}\eta\,(D + K_1)\,,\\
\nabla_\eta\lambda_3 &= m_{P,3} + \frac{1}{2}\eta\,(D + K_2)\,,\\
\nabla_\eta\lambda_4 &= m_{P,4} - \frac{1}{2}\eta\,(D - K_1)\,,\\
\nabla_\eta\lambda_6 &= m_{P,6} + \frac{1}{2}\eta\,(D - K_2)\,.
\end{aligned}$$

Here we have used the following notation

$$
\begin{aligned}
m_{P,i} &:= \eta^{\perp} \cdot \left(\eta^{\perp}\right)^T \left(\nu_{1,i}\vec{u}_1 + \nu_{2,i}\vec{u}_2\right) + \eta \cdot \eta^T \left(\frac{1}{2}\vec{u}_1 + \frac{1}{2}\vec{u}_2\right), \\
\nu_{1,i} &:= -\frac{2\left\{\left(\eta^T\vec{u}_2 - \lambda_i\right)^2 - gh_2\right\}\left(\eta^T\vec{u}_1 - \lambda_i\right)}{\partial_\lambda\, p_2(\lambda,\eta)|_{\lambda=\lambda_i}}, \\
\nu_{2,i} &:= -\frac{2\left\{\left(\eta^T\vec{u}_1 - \lambda_i\right)^2 - gh_1\right\}\left(\eta^T\vec{u}_2 - \lambda_i\right)}{\partial_\lambda\, p_2(\lambda,\eta)|_{\lambda=\lambda_i}}.
\end{aligned}
$$

2.1.2 Eigenvectors

Since the eigenvectors of (6) can be formulated in a closed analytical form, we can compute them having obtained the corresponding eigenvalues. For the roots $\lambda_{2/5} := \eta^T\vec{u}_{1/2}$ of (7), i.e. the roots of p_1, the corresponding right and left eigenvectors are

$$
\left(l_\eta^2\right)^T = r_\eta^2 = \begin{pmatrix} 0 \\ -\eta^{\perp} \\ 0 \\ 0 \end{pmatrix}, \qquad \left(l_\eta^5\right)^T = r_\eta^5 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -\eta^{\perp} \end{pmatrix}. \tag{9}
$$

For the remaining eigenvalues λ_j, $j = 1, 3, 4, 6$, the right eigenvectors are

$$
r_\eta^j = \left(\left(\eta^T\vec{u}_2 - \lambda_j\right)^2 - gh_2\right) \begin{pmatrix} h_1 \\ -\eta\left(\eta^T\vec{u}_1 - \lambda_j\right) \\ 0 \\ 0 \end{pmatrix} + grh_1 \begin{pmatrix} h_2 \\ 0 \\ h_2 \\ -\eta\left(\eta^T\vec{u}_2 - \lambda_j\right) \end{pmatrix}, \tag{10}
$$

whereas the left eigenvectors l_η^j can be deduced to be

$$
\Lambda_j l_\eta^j = \frac{1}{h_1} \begin{pmatrix} -\left(\eta^T\vec{u}_1 - \lambda_j\right) \\ \eta h_1 \\ \left(\eta^T\vec{u}_1 - \lambda_j\right) \\ 0 \end{pmatrix}^T + \frac{\left(\eta^T\vec{u}_1 - \lambda_j\right)^2 - gh_1}{grh_1h_2} \begin{pmatrix} 0 \\ 0 \\ -\left(\eta^T\vec{u}_2 - \lambda_j\right) \\ \eta h_2 \end{pmatrix}^T \tag{11}
$$

with

$$
\begin{aligned}
\Lambda_j &:= \prod_{i\in\{1,3,4,6\}\setminus\{j\}} (\lambda_i - \lambda_j) = \partial_\lambda\, p_2(\lambda,\eta)|_{\lambda=\lambda_j} \\
&= -2\left(\left(\left(\eta^T\vec{u}_1 - \lambda_j\right)^2 - gh_1\right)\left(\eta^T\vec{u}_2 - \lambda_j\right) + \left(\left(\eta^T\vec{u}_2 - \lambda_j\right)^2 - gh_2\right)\left(\eta^T\vec{u}_1 - \lambda_j\right)\right).
\end{aligned}
$$

2.2 Steady state solutions

If $\partial_t(h_1, h_1\vec{u}_1^T, h_2, h_2\vec{u}_2^T)^T = 0$ we say that the solution of (3) is stationary. A scheme preserving a particular stationary solution is called well-balanced. For (3) written in

primitive variables the stationary solutions are those fulfilling

$$\begin{cases} \operatorname{div}(h_1\vec{u}_1) = \operatorname{div}(h_2\vec{u}_2) = 0, \\ (\frac{d}{dx}\vec{u}_1)\vec{u}_1 = -g\nabla(h_1 + h_2 + b) - f\vec{u}_1^{\perp}, \\ (\frac{d}{dx}\vec{u}_2)\vec{u}_2 = -g\nabla(rh_1 + h_2 + b) - f\vec{u}_2^{\perp}, \end{cases} \tag{12}$$

If we consider solutions to be constant along the streamlines $\partial_t + \vec{u}_i^T\nabla$, i.e. $(\partial_t + \vec{u}_i^T\nabla)h_i = (\partial_t + \vec{u}_i^T\nabla)\vec{u}_i = 0$ for $i = 1, 2$, then (3) simplifies to

$$\begin{cases} h_1 \operatorname{div}(\vec{u}_1) = h_2 \operatorname{div}(\vec{u}_2) = 0, \\ g\nabla(h_1 + h_2 + b) = -f\vec{u}_1^{\perp}, \\ g\nabla(rh_1 + h_2 + b) = -f\vec{u}_2^{\perp}, \end{cases} \tag{13}$$

These are the conditions of the geostrophic equilibrium.
Physically (13) means that in the particular domain the velocities are conserved. Moreover, the direction of steepest descent of the surface ε, respectively the modified surface $\hat{\varepsilon} := rh_1 + h_2 + b$, is orthogonal to the velocity fields $\vec{u}_1$, respectively $\vec{u}_2$. Additionally, since

$$g\vec{u}_1^T\nabla\varepsilon = -\vec{u}_1^T\vec{u}_1^{\perp}f = 0$$
$$g\vec{u}_2^T\nabla\hat{\varepsilon} = -\vec{u}_2^T\vec{u}_2^{\perp}f = 0$$

ε, as well as $\hat{\varepsilon}$, are constant in direction $\vec{u}_1$ and $\vec{u}_2$, respectively. This means that they are locally one dimensional. Note, that this is not the case for ω unless $\{\vec{u}_1, \vec{u}_2\}$ is a linearly dependent family.
Another special solution to (13), the so-called lake at rest solution,

$$\nabla\varepsilon = 0 = \nabla\hat{\varepsilon}, \qquad \vec{u}_1 = 0 = \vec{u}_2 \tag{14}$$

is of particular interest. Our aim is to construct a scheme that preserves important stationary states, see Section 3.5.

3 Numerical Scheme

The **F**inite **V**olume **E**volution **G**alerkin schemes, or FVEG schemes for short, are finite volume schemes consisting of two steps. In the predictor step an evolutionary step is performed in order to predict the solution at a new intermediate time step. This is done by the evolution operators. In the corrector step a standard finite volume update is done. The FVEG schemes were already derived successfully for several hyperbolic balance laws, for example for the shallow water flows [28], the wave equation system [3, 29], the Euler equations [30, 31] and the shallow water magnetohydrodynamics equations [25]. Questions of accuracy, stability and the well-balancing have been studied in [28, 29, 30, 31].

Assume a discretization of the computational domain Ω into finite volumes Ω_i, $i = 1, \ldots, N$, with $\bigcup_{i=1}^{N} \Omega_i \approx \Omega$ and denote by $|\Omega_i|$ the volume of Ω_i. Furthermore, assume that the boundary $\partial\Omega_i$ of Ω_i is a hyperrectangle aligned with the coordinates x and composed of faces $I_{i,k}$, $k = 1, \ldots, M_i$, see Figure 2. In the numerical experiments presented below $d = 2$ and $M_i = 4$ for every i. Now integrating (1) over $\Omega_i \times [t_n, t_{n+1}]$ and applying the Gauss theorem we get for any volume Ω_i the balance equation in integral form, i.e.

$$\int_{\Omega_i} W(x, t_{n+1}) - W(x, t_n)\, d^{(d)}\Omega_i + \int_{t_n}^{t_{n+1}} \int_{\partial\Omega_i} F(W)\, n\, d^{(d-1)}\partial\Omega_i\, dt = \int_{t_n}^{t_{n+1}} \int_{\Omega_i} B(W) + C_0(W)\nabla\sigma(x) + \sum_{l=1}^{n} C_l(W)\nabla W_l\, d^{(d)}\Omega_i\, dt.$$

Setting $\Delta t_n := t_{n+1} - t_n$ and $W_i^n := \frac{1}{|\Omega_i|}\int_{\Omega_i} W(x, t_n)\, d^{(d)}\Omega$ and approximating the time-integral by the midpoint rule we get the FVEG formulation of corrector step

$$W_i^{n+1} = W_i^n - \frac{\Delta t_n}{|\Omega_i|} \sum_{k=1}^{M_i} \int_{I_{i,k}} F^{n+1/2}\, n_{i,k}\, d^{(d-1)} I_{i,k} + \frac{\Delta t_n}{|\Omega_i|} \int_{\Omega_i} B^{n+1/2} + C^{n+1/2}\nabla\sigma(x) + \sum_{l=1}^{n} C_l^{n+1/2}\nabla W_l^{n+1/2}\, d^{(d)}\Omega_i, \quad (15)$$

where $n_{i,k}$ is the outer normal vector to the face $I_{i,k}$ and $(\,.\,)^{n+1/2} := (\,.\,)|_{t=t_n+\Delta t_n/2}$ denotes a function evaluated at time $t_n + \Delta t_n/2$. In what follows, we will restrict to $\Omega \subseteq \mathbb{R}^2$, thus $d = 2$.

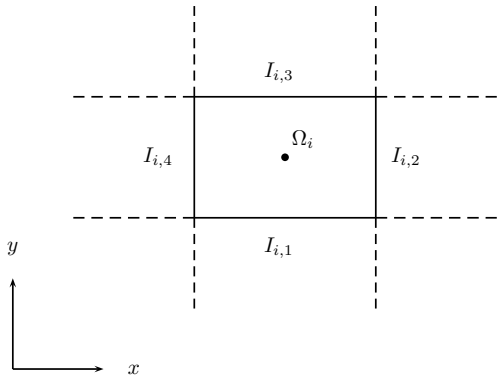


Figure 2: A finite volume Ω_i composed of faces $I_{i,1}, \ldots, I_{i,4}$.

3.1 Discretization of the fluxes

The discretization of the flux term in (15) is straightforward. For the approximation of the cell interface integral $\int_{I_{i,k}} (F^{n+1/2})^T n_{i,k}\, d^{(d-1)} I_{i,k}$ we use either the midpoint or the Simpson rule, i.e.

$$\int_{I_{i,k}} F^{n+1/2}\, n_{i,k}\, d^{(1)} I_{i,k} \approx |I_{i,k}| \sum_{l=1}^{K} \alpha_l F(W(x_{i,k}^l, t_n + \Delta t_n/2))\, n_{i,k}, \tag{16}$$

α_l being the weights of the quadrature rule. In particular, for the midpoint rule $K = 1$ and $x_{i,k}^1$ is the midpoint of $I_{i,k}$, whereas for the Simpson rule $K = 3$, $x_{i,k}^1, x_{i,k}^3$ are the end-points and $x_{i,k}^2$ the midpoint of $I_{i,k}$. The values of the solution $W(x_{i,k}^l, t_n + \Delta t_n/2)$ at quadrature points at a half time-step are predicted using the evolution operator, cf. Section 3.3.

3.2 Discretization of the source term

The correct discretization of the source term is crucial to obtain a correct solution. Since the system is not conservative the discretization not only influences the well-balancing but also the solution itself. Indeed, for a non-conservative system the shock speed depends on the path chosen to connect the left and right limiting values of the shock. Thus a certain numerical solution corresponds to a certain path. Indeed a discretization of source terms directly influences the shock speed.
The so called path-conservative schemes introduced by Parés [34] describe how to discretize the source term once a path is chosen. However, they have shown that choosing a different path than the one prescribed in the definition of a weak solution the discretization error may not vanish as $|\Omega_i|, \Delta t_n \to 0$.
Based on this considerations we use two different types of discretization of the source term $s(W)$. The first one is described in [20]. Here the operator splitting technique is used and the two-layer system is splitted into two one-layer shallow water equations. The path-consistent approximation of the nonconservative terms is done for each layer consecutively using piecewise linear curves connecting three predicted states: the left state, an intermediate value at a cell interface and the right state, see Figure 3. The predicted states at the time level $t_{n+1/2}$ are obtained using the corresponding evolution operator of each layer. This discretization fits into the framework of path-conservative schemes. Note that in the case when the signatures of eigenvalues of the full and the splitted system do not coincide large oscillations may appear in the solution. They decrease as the mesh is refined.
The second discretization is done for the unsplit two-layer equations. It is based on the idea that the total momentum is conservative. Considering the pressure terms in the equations of total momentum (4) we have the equality

$$rgh_1\nabla(b + h_2) + gh_2\nabla(b + rh_1) = rg\nabla(h_1h_2) + g(rh_1 + h_2)\nabla b.$$

As soon as b is smooth the right hand side is well-defined. Thus we want to discretize the left hand side terms in such a way that this equality holds. This leads to a discretization

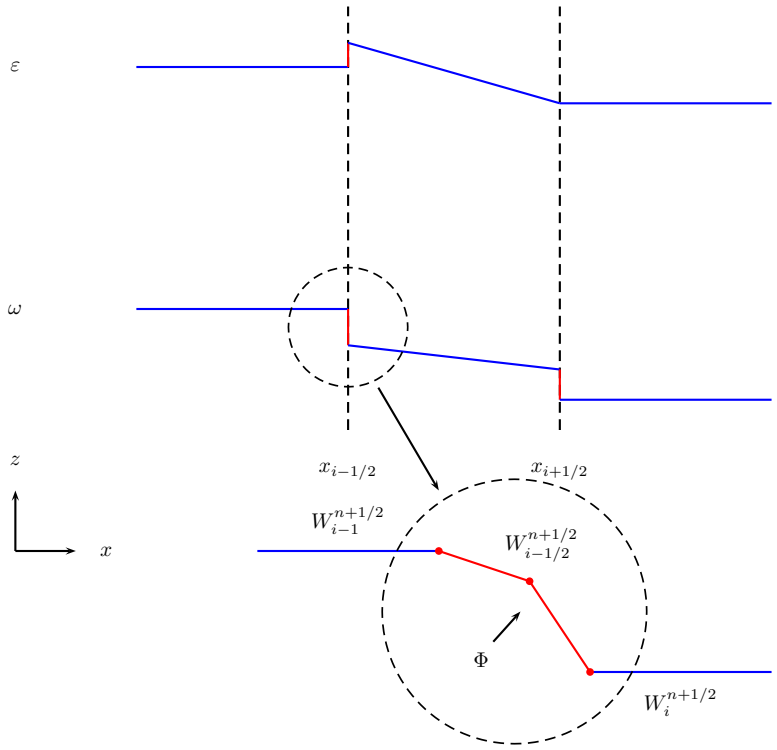


Figure 3: Path chosen for the discretization of source terms in the path-consistent FVEG scheme from [20].

already used for the shallow water equation in [28]. However, here, the source term reads

$$\Sigma = \int_{\Omega_i} B + C_0 \nabla \sigma + \sum_{l=1}^{n} C_l \nabla W_l \, d^{(1)}\Omega_i \tag{17}$$

with $W, B, C_0, \ldots, C_n$ given by (2) and $\sigma \equiv b$.
Now, since $C_0 = \frac{1}{r} C_1 + C_4$ and $C_2 = C_3 = C_5 = C_6 = 0$ we can further simplify (17)

$$\Sigma = \int_{\Omega_i} B + \frac{1}{r} C_1 \nabla \left(r h_1 + b \right) + C_4 \nabla \left(h_2 + b \right) \, d^{(2)}\Omega_i. \tag{18}$$

Let us decompose B as

$$B = C_4 \frac{f}{g} \vec{u}_1^{\perp} + C_1 \frac{f}{rg} \vec{u}_2^{\perp},$$

and set $\partial_x K_i = -\frac{f}{g} v_i$ and $\partial_x L_i = \frac{f}{g} u_i$, $i = 1, 2$. The source term (18) is now given by

$$\Sigma = \int_{\Omega_i} \frac{1}{r} C_1 \begin{pmatrix} \partial_x \left(K_2 + r h_1 + b \right) \\ \partial_y \left(L_2 + r h_1 + b \right) \end{pmatrix} + C_4 \begin{pmatrix} \partial_x \left(K_1 + h_2 + b \right) \\ \partial_y \left(L_1 + h_2 + b \right) \end{pmatrix} \, d^{(2)}\Omega_i. \tag{19}$$

Moreover, we approximate the matrices C_1, C_4 as follows

$$\overline{C_1} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -rg\mu_x h_2 & 0 \\ 0 & -rg\mu_y h_2 \end{bmatrix}, \overline{C_4} = \begin{bmatrix} 0 & 0 \\ -g\mu_x h_1 & 0 \\ 0 & -g\mu_y h_1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}.$$

Here $\mu_x f := \frac{1}{2} \left(f^{n+1/2}\big|_{I_{i,2}} + f^{n+1/2}\big|_{I_{i,4}} \right)$, for some function f, is the averaging operator in the x-direction at a half time-step with analogous notation for $\mu_y(\,.\,)$. Inserting the approximated matrices into (19) the Gauss theorem can be applied to reduce the integration over Ω_i to integration over the faces $I_{i,k}$, i.e.

$$\Sigma = \sum_{k=1}^{4} \int_{I_{i,k}} \frac{1}{r} \overline{C_1} \cdot \mathrm{diag}(n_{i,k}) \cdot \begin{pmatrix} K_2 + r h_1 + b \\ L_2 + r h_1 + b \end{pmatrix} + \overline{C_4} \cdot \mathrm{diag}(n_{i,k}) \cdot \begin{pmatrix} K_1 + h_2 + b \\ L_1 + h_2 + b \end{pmatrix} \, d^{(1)} I_{i,k}. \tag{20}$$

The face integrals are approximated in the same way as in the flux term (16). We use here again the predicted solution $W(x_{i,k}^l, t_n + \Delta t_n/2)$ at quadrature points. Note that this approximation of the nonconservative products in the cells is consistent with computing the following integrals

$$\begin{bmatrix} 0 \\ -g \int_0^1 \Phi_{h_1}(s) \Phi_{h_2}(s) ds \\ -g \int_0^1 \Phi_{h_2}(s) \Phi_{h_1}(s) ds \end{bmatrix}$$

using the family of straight segments

$$\Phi_{h_j}(s) = h^{n+1/2}_{j,i-1/2} + s(h^{n+1/2}_{j,i+1/2} - h^{n+1/2}_{j,i-1/2}), \quad j = 1, 2.$$

We would like to point out that our discretizations presented in [20] still do not solve the fundamental drawbacks of the path-consistent schemes. The second discretization of the nonconservative terms, cf. (20), is different and does not fit formally in the path-consistent framework. It is just based on the approximation of the total momentum and the Rankine-Hugoniot conditions hold for the total momentum.

3.3 Time evolution

A crucial step in the FVEG scheme is the use of theory of bicharacteristics. Here we will give a brief overview of this method, for further details see [18, 26]. Assume the balance law (5) to be hyperbolic and denote by λ^j_η, r^j_η and l^j_η its eigenvalues and the corresponding right and left eigenvectors, respectively.
In what follows we assume that A_i is either constant with respect to V or it is locally linearized at some suitable state $\overline{V}$, i.e. $A_i := A_i(\overline{V})$. In our numerical experiments presented below $\overline{V}$ is a local arithmetic mean of the cell averaged values of all cells that contain the particular integration point (x_e, t_s), cf. (23). Then λ^j_η, r^j_η and l^j_η do not depend on space and time and we can transform (5) further to characteristic variables $l^j_\eta V$. The system now reads

$$\partial_t(l^j_\eta V) + \sum_{i=1}^{n}\sum_{k=1}^{d} l^j_\eta A_k r^i_\eta \, \partial_{x_k}(l^i_\eta V) = (l^j_\eta s)(\nabla\sigma, V). \tag{21}$$

This equality can be rewritten as

$$\partial_t(l^j_\eta V) + \sum_{k=1}^{d} \partial_{\eta_k}\lambda^j_\eta \, \partial_{x_k}(l^j_\eta V) = (l^j_\eta s)(\nabla\sigma, V) + \sum_{k=1}^{d} l^j_\eta \left(\partial_{\eta_k}\lambda^j_\eta I - A_k\right) \partial_{x_k} V =: S_j.$$

Note that multidimensionality introduces additional source terms. Now, since A is constant with respect to V, we have

$$\frac{dx^j}{dt} = \nabla_\eta \lambda^j_\eta, \quad \frac{d(l^j_\eta V)}{dt} = S_j, \quad j = 1, \ldots, n. \tag{22}$$

The first term is a first order system of ordinary differential equations for the j-th bicharacteristic $Q_j(t, \eta) := (x^j(t, \eta), t)^T$. Since we search for the solution $V(P)$ at some point $P := (x_e, t_e)^T$ we need to integrate the second ordinary differential equation in (22) from time t_e down to time t_s, where the solution is known. We obtain

$$\begin{aligned} x^j(t_s, \eta) &= x_e - \nabla_\eta \lambda^j_\eta (t_e - t_s), \\ (l^j_\eta V)(P) &= (l^j_\eta V)(Q^j(t_s, \eta)) + \int_{t_s}^{t_e} S_j(Q_j(t, \eta), \eta)\, dt. \end{aligned} \tag{23}$$

From the theory of bicharacteristics [3] it follows that $l_\eta^j A_k r_\eta^j = \partial_{\eta_k} \lambda_\eta^j$, therefore

$$\lambda_\eta^j = l_\eta^j A r_\eta^j = l_\eta^j \left(\sum_{k=1}^{w} \eta_k A_k \right) r_\eta^j = \eta^T \nabla_\eta \lambda_\eta^j, \tag{24}$$

where the vectors $\nabla_\eta \lambda_\eta^j$ are the ray velocities. We choose the directional vector η as $\eta_1 := \cos(\theta)$ and $\eta_2 := \sin(\theta)$ parametrized by the angle θ, $\theta \in [0, 2\pi)$.
Now, to obtain back V variables, we multiply (23) by r_η^j from the left, sum over j and average over the unit sphere S^{d-1} in $\mathbb{R}^d$, see [26], i.e.

$$\begin{aligned} V(P) = \frac{1}{|S^{d-1}|} \int_{S^{d-1}} \sum_{j=1}^{n} r_\eta^j \, l_\eta^j \Big\{ & V(x_e - \nabla_\eta \lambda_\eta^j (t_e - t_s), t_s) \\ & + \int_{t_s}^{t_e} s(\nabla \sigma(x_e - \nabla_\eta \lambda_\eta^j (t_e - t)), V(x_e - \nabla_\eta \lambda_\eta^j (t_e - t), t)) \\ & + \sum_{k=1}^{d} \left(\partial_{\eta_k} \lambda_\eta^j I - A_k \right) \partial_{x_k} V(x_e - \nabla_\eta \lambda_\eta^j (t_e - t), t) \, dt \Big\} \, d\eta. \end{aligned} \tag{25}$$

Form (25) is used to predict the solution at a half time step. To simplify (25) one can furthermore use the extended lemma on bicharacteristics, cf. [39], in order to rewrite the last term for genuinely nonlinear fields, the so-called mantle integrals, as the integrals with the tangential derivatives of velocities. Let us remind that in the simpler case of wave equation system the bicharacteristics create a circular wave front and the so-called useful lemma, cf. [29], Lemma 2.1, easily yields integrals with the tangential derivatives after applying the integration by parts to the mantle integrals. From [40] we note that the mantle integral contains tangential derivatives of velocities for a general wavefront and hence the useful lemma holds not only for the case of circular wavefront but even for arbitrary curves.
Equation (25) gives an implicit representation of solution at a point P. If the eigenstructure of the system is known (25) gives an exact integral expression for the solution $V(P)$. For a particular representation of (25) for the wave and shallow water equations, see, e.g., [3] and [28]. Then it is enough to follow the bicharacteristics x^j, pick up the value of V at the foot of the bicharacteristic cone, i.e. at $(x_e - \nabla_\eta \lambda_\eta^j (t_e - t_s), t_s)^T$, and evaluate the time integral in (25).
Note that in the case that the matrices A_j are constant with respect to V the envelope of the bicharacteristics $x^j(t, \theta)$ with respect to t, θ either generates mantles of conoids or degenerates to straight lines. Figure 4 shows the envelopes of bicharacteristics for the particular case of two-layer shallow water flows with frozen Jacobian matrices A_i, $i = 1, 2$.
For a general system consisting of n equations also n bicharacteristics exist. Again restricting to two-layer shallow water flows the two real roots of p_2 generate the same bicharacteristic $x^1(t, \theta) \equiv x^3(t, \theta + \pi)$. The two possibly complex roots of p_2, provided they are real, generate $x^4(t, \theta) \equiv x^6(t, \theta + \pi)$, whereas the roots of p_1, corresponding to the linearly degenerated fields, generate the bicharacteristics x^2 and x^5, respectively.

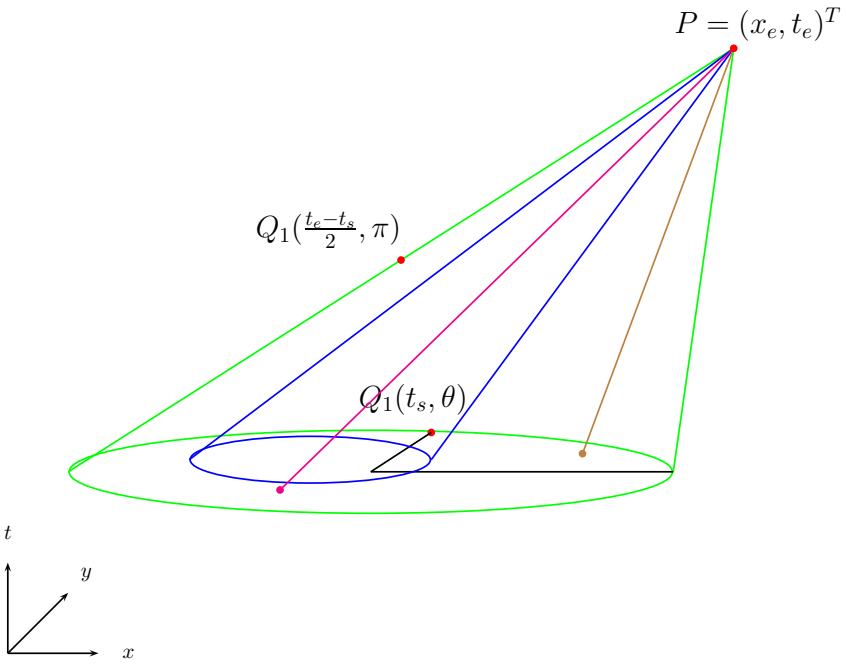


Figure 4: Mantels of conoids with apex P generated by envelopes of bicharacteristics x^1 (green), x^4 (blue) and the bicharacteritsics x^2 (magenta) and x^5 (brown).

3.4 Numerical approximation of the time evolution

The prediction step for the evaluation of the flux and source terms at a half time $t_n + \Delta t_n/2$ is done by the aforementioned theory of bicharacteristics, i.e. we set $t_s = t_n$, $t_e = t_n + \Delta t_n/2$ and P is a quadrature point $(x^l_{i,k}, t_n + \Delta t_n/2)$. Here we want to describe in detail how the exact evolution operator (25) is approximated.
Since the left and right eigenvectors (9), (10) and (11) for system (6) are known and the time integral in (25) is approximated by the rectangle rule at $t = t_n$, it suffices to describe the approximation of the line integral in (25) and how to obtain the ray velocities. The approximation of the line integral is tricky. Here it is crucial to consider the geometrical position of the sonic curve $Q_1(t_n, \theta), \theta \in [0, 2\pi)$ (intersection of a bicharacteristic conoid with the plane $t = t_n$) and thus the resulting intersection points between $Q_1(t_n, \theta)$ and the grid. The situation is depicted in Figure 5.

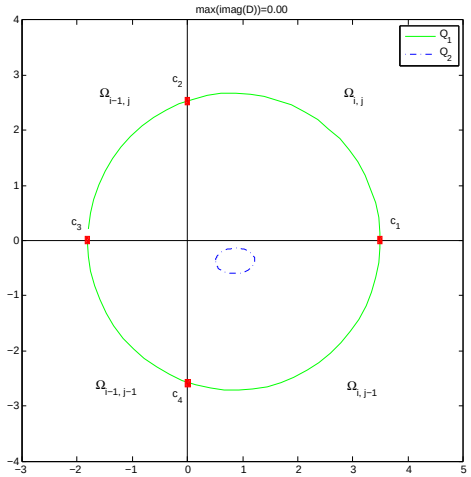


Figure 5: Intersection points between the grid and the circle $c(\theta) = Q_1(t_n, \theta)$ (solid curve). The dashed curve shows $Q_2(t_n, \theta)$, i.e. the perimeter of the sonic conoid corresponding to the internal eigenvalues.

Since the kernel of the line integral is a periodic function one would use for example a quadrature rule of the form $\frac{1}{N}\sum_{k=1}^{N} f(\theta_k)$. However, this quadrature in general introduces spurious oscillations since it might happen, that not all volumes of the domain of dependence are considered. To get a stable approximation we compute the intersection points with the grid $c_i := c(\theta_i)$, where $c(\theta)$ is a circle approximating the bicharacteristic x^1, i.e. the bicharacteristic corresponding to the external eigenvalues (remember, these bound the other eigenvalues). Next, we use the midpoint rule on each arc $[c(\theta)]_{\theta \in [\theta_i, \theta_{(i \mod 4)+1}]}$, $i = 1, \ldots, 4$, see again Figure 5. Now in the supersonic case we can have two distinct cases. When the integration point P lies on an edge the sonic curve is completely contained in one mesh cell. Then we take $\frac{1}{2}(f(0) + f(\pi))$. In the second case, when the integration point P lies on a vertex we have to distinguish two

sub-cases. The sonic curve lies in one cell or in two cells. For the former case we take $\frac{1}{4}\left(f(0)+f(\frac{\pi}{2})+f(\pi)+f(\frac{3\pi}{2})\right)$. Otherwise we compute the two intersection points with the grid and proceed analogously as in the subsonic case.
Let us note that in order to determine the corresponding angles θ_i it is enough to consider only x^1 since the domain of dependence of the numerical scheme is given by the bicharacteristic corresponding to the absolutely largest eigenvalue, i.e. by $|\lambda_{1/3}|$.
Further, to obtain a second order scheme the integration is realized with respect to bilinear functions. In particular we use the min-mod reconstruction, cf. Section 4, to get $\partial_{x_k} V$ that are needed at the foot of respective bicharacteristics in (25).
Now, it remains to compute the eigenvalues λ_j, $j = 1, 3, 4, 6$. This is done by a Newton-Raphson iteration to get λ_1, λ_3, i.e. the external eigenvalues. The starting values can be chosen as e.g. (8). However, since (8) are only valid for $r \approx 1$ and $\vec{u}_1 \approx \vec{u}_2$, a better choice for the starting values is to take the one dimensional lower and upper bounds

$$\min\left(u_1 - \sqrt{(1+\sqrt{r})gh_1},\ u_2 - \sqrt{(1+\sqrt{r})gh_2}\right),$$
$$\max\left(u_1 + \sqrt{(1+\sqrt{r})gh_1},\ u_2 + \sqrt{(1+\sqrt{r})gh_2}\right)$$

of the external eigenvalues, see [1] for a more detailed derivation. Next, we reduce p_2 to a second degree polynomial and compute the internal eigenvalues λ_4, λ_6.
$c(\theta)$ is computed as follows. We compute the eigenvalues for $\theta = 0$ and $\theta = \pi/2$. These information define uniquely the circle $c(\theta)$. All of our numerical experiments indicates that this is sufficient. The bicharacteristics corresponding to $\lambda_{2/5}$ are just straight lines. Note that unlike the case of shallow water and wave equations here the degenerated bicharacteristics are not the axes of the cones created by the other bicharacteristics.

3.5 Well-balancing for the lake at rest

It was already shown in [20], Theorem 1 that the finite volume update (15) preserves the lake at rest in the case of path-consistent approximation of the source term, cf. Figure 3. Here, we present the same result for the source term approximation in (20).

Lemma 3.1 (Well-balanced finite volume update)
The finite volume update (15) *with the discretization of the flux and source term* (16) *and* (20) *is third order well-balanced for the geostrophic equilibrium and exactly well-balanced for the lake at rest state, provided the EG-operator* (25) *preserves these states.*

Proof. First consider the face integral in the source term (20)

$$\begin{aligned}
&\frac{1}{|I_{i,k}|}\int_{I_{i,k}} \frac{1}{r}\overline{C_1}\cdot\operatorname{diag}(n_{i,k})\cdot\begin{pmatrix}K_2+rh_1+b\\L_2+rh_1+b\end{pmatrix}+\overline{C_4}\cdot\operatorname{diag}(n_{i,k})\cdot\begin{pmatrix}K_1+h_2+b\\L_1+h_2+b\end{pmatrix} d^{(1)}I_{i,k}\\
\approx&\sum_{l=1}^{K}\alpha_l\left(\frac{1}{r}\overline{C_1}\cdot\operatorname{diag}(n_{i,k})\cdot\begin{pmatrix}K_2+rh_1+b\\L_2+rh_1+b\end{pmatrix}\right.\\
&\qquad\left.+\overline{C_4}\cdot\operatorname{diag}(n_{i,k})\cdot\begin{pmatrix}K_1+h_2+b\\L_1+h_2+b\end{pmatrix}\right)(x_{i,k}^l,t_n+\frac{\Delta t_n}{2}),
\end{aligned}\tag{26}$$

and decompose the flux as

$$F(W)=\begin{bmatrix}h_1u_1 & h_1v_1\\ h_1u_1^2+\frac{g}{2}h_1^2 & h_1u_1v_1\\ h_1u_1v_1 & h_1v_1^2+\frac{g}{2}h_1^2\\ h_2u_2 & h_2v_2\\ h_2u_2^2+\frac{g}{2}h_2^2 & h_2u_2v_2\\ h_2u_2v_2 & h_2v_2^2+\frac{g}{2}h_2^2\end{bmatrix}=\underbrace{\begin{bmatrix}0&0\\ \frac{g}{2}h_1^2&0\\0&\frac{g}{2}h_1^2\\0&0\\ \frac{g}{2}h_2^2&0\\0&\frac{g}{2}h_2^2\end{bmatrix}}_{F_1(W):=}+\underbrace{\begin{bmatrix}h_1u_1&h_1v_1\\ h_1u_1^2&h_1u_1v_1\\ h_1u_1v_1&h_1v_1^2\\ h_2u_2&h_2v_2\\ h_2u_2^2&h_2u_2v_2\\ h_2u_2v_2&h_2v_2^2\end{bmatrix}}_{F_2(W):=}.$$

The flux term (16) then reads

$$\begin{aligned}
&\sum_{l=1}^{K}\alpha_l F(W(x_{i,k}^l,t_n+\frac{\Delta t_n}{2}))\,n_{i,k}\\
=&\sum_{l=1}^{K}\alpha_l\left(-\frac{1}{r}\overline{C_1}\cdot\operatorname{diag}(n_{i,k})\cdot\begin{pmatrix}h_2\\h_2\end{pmatrix}-\overline{C_4}\cdot\operatorname{diag}(n_{i,k})\cdot\begin{pmatrix}h_1\\h_1\end{pmatrix}+F_2(W)\,n_{i,k}\right)(x_{i,k}^l,t_n+\frac{\Delta t_n}{2}).
\end{aligned}\tag{27}$$

Now substituting (26) and (27) in (15) and, for convenience, dropping the evaluation points $(x_{i,k}^l,t_n+\frac{\Delta t_n}{2})$ we get

$$\begin{aligned}
W_i^{n+1}&=W_i^n+\frac{\Delta t_n}{|\Omega_i|}\sum_{k=1}^{4}|I_{i,k}|\sum_{l=1}^{K}\alpha_l\Big[-F_2(W)\,n_{i,k}\\
&\quad+\frac{1}{r}\overline{C_1}\cdot\operatorname{diag}(n_{i,k})\cdot\begin{pmatrix}h_2\\h_2\end{pmatrix}+\overline{C_4}\cdot\operatorname{diag}(n_{i,k})\cdot\begin{pmatrix}h_1\\h_1\end{pmatrix}\\
&\quad+\frac{1}{r}\overline{C_1}\cdot\operatorname{diag}(n_{i,k})\cdot\begin{pmatrix}K_2+rh_1+b\\L_2+rh_1+b\end{pmatrix}+\overline{C_4}\cdot\operatorname{diag}(n_{i,k})\cdot\begin{pmatrix}K_1+h_2+b\\L_1+h_2+b\end{pmatrix}\Big]\\
&=W_i^n+\frac{\Delta t_n}{|\Omega_i|}\sum_{k=1}^{4}|I_{i,k}|\sum_{l=1}^{K}\alpha_l\Big[-F_2(W)\,n_{i,k}\\
&\quad+\frac{1}{r}\overline{C_1}\cdot\operatorname{diag}(n_{i,k})\cdot\begin{pmatrix}K_2+rh_1+h_2+b\\L_2+rh_1+h_2+b\end{pmatrix}+\overline{C_4}\cdot\operatorname{diag}(n_{i,k})\cdot\begin{pmatrix}K_1+h_1+h_2+b\\L_1+h_1+h_2+b\end{pmatrix}\Big],
\end{aligned}$$

where $|I_{i,k}| = |I_{i,k+2}|$ and $n_{i,k} = -n_{i,k+2}$, $k = 1, 2$. By assumption the EG-operator preserves the geostrophic equilibrium and therefore

$$\begin{aligned} h_j \operatorname{div} \vec{u}_j &= 0, \qquad j = 1, 2, \\ \begin{pmatrix} K_2 + rh_1 + h_2 + b \\ L_2 + rh_1 + h_2 + b \end{pmatrix} &= const., \\ \begin{pmatrix} K_1 + h_1 + h_2 + b \\ L_1 + h_1 + h_2 + b \end{pmatrix} &= const. \end{aligned}$$

The last two equalities imply for $l = 1, \ldots, K$

$$\sum_{k=1}^{4} |I_{i,k}| \left[\frac{1}{r} \overline{C_1} \cdot \operatorname{diag}(n_{i,k}) \cdot \begin{pmatrix} K_2 + rh_1 + h_2 + b \\ L_2 + rh_1 + h_2 + b \end{pmatrix} + \overline{C_4} \cdot \operatorname{diag}(n_{i,k}) \cdot \begin{pmatrix} K_1 + h_1 + h_2 + b \\ L_1 + h_1 + h_2 + b \end{pmatrix} \right] = 0,$$

and thus

$$W_i^{n+1} = W_i^n - \frac{\Delta t_n}{|\Omega_i|} \sum_{k=1}^{4} |I_{i,k}| \sum_{l=1}^{K} \alpha_l \, F_2(W(x_{i,k}^l, t_n + \frac{\Delta t_n}{2})) \, n_{i,k}. \tag{28}$$

With $\mathbb{1} \in \mathbb{R}^6$ being the vector of ones we have, for example, for the cell integration by means of the Simpson rule

$$\begin{aligned} W_i^{n+1} &= W_i^n - \frac{\Delta t_n}{|\Omega_i|} \sum_{k=1}^{4} |I_{i,k}| \sum_{l=1}^{K} \alpha_l \, F_2(W(x_{i,k}^l, t_n + \frac{\Delta t_n}{2})) \, n_{i,k} \\ &= W_i^n - \frac{\Delta t_n}{|\Omega_i|} \int_{\Omega_i} \operatorname{div} F_2(W(x, t_n + \frac{\Delta t_n}{2})) \, d^{(2)}\Omega_i + \mathcal{O}(\Delta t_n \max_{i,k} |I_{i,k}|^2) \cdot \mathbb{1} \\ &= W_i^n - \frac{\Delta t_n}{|\Omega_i|} \int_{\Omega_i} \left. \begin{pmatrix} \partial_x(h_1 u_1) + \partial_y(h_1 v_1) \\ \partial_x(h_1 u_1^2) + \partial_y(h_1 u_1 v_1) \\ \partial_x(h_1 u_1 v_1) + \partial_y(h_1 v_1^2) \\ \partial_x(h_2 u_2) + \partial_y(h_2 v_2) \\ \partial_x(h_2 u_2^2) + \partial_y(h_2 u_2 v_2) \\ \partial_x(h_2 u_2 v_2) + \partial_y(h_2 v_2^2) \end{pmatrix} \right|_{t = t_n + \frac{\Delta t_n}{2}} d^{(2)}\Omega_i + \mathcal{O}(\Delta t_n \max_{i,k} |I_{i,k}|^2) \cdot \mathbb{1} \end{aligned}$$

For simplicity of presentation we use the standard notation δ_x, δ_y for central differences and denote by $\Delta x = \max_i (|I_{i,1}||I_{i,3}|)$ and $\Delta y = \max_i (|I_{i,2}||I_{i,4}|)$. We assume that the EG operator satisfies the equilibrium condition $(13)_1$ in the following way $h_j \operatorname{div} \vec{u}_j = h_j (\delta_x u_j + \delta_y v_j) + \mathcal{O}(\max_{i,k} |I_{i,k}|^3)$, $j = 1, 2$. Then it easy to show that the material derivatives are zero up to the third order, i.e. with $\mathcal{L}_i := \partial_t + \vec{u}_i^T \nabla$ we have

$$(\mathcal{L}_1 h_1, \mathcal{L}_1 u_1, \mathcal{L}_1 v_1, \mathcal{L}_2 h_2, \mathcal{L}_2 u_2, \mathcal{L}_2 v_2) = \mathcal{O}(\max_{i,k} |I_{i,k}|^3) \cdot \mathbb{1}.$$

Indeed, we have, for example, for the update of the first equation for h_1

$$\begin{aligned}(h_1)_i^{n+1} &= (h_1)_i^n - \frac{\Delta t_n}{|\Omega_i|}\int_{\Omega_i}\left(\partial_x(h_1u_1)+\partial_y(h_1v_1)\right)|_{t=t_n+\frac{\Delta t_n}{2}}\, d^{(2)}\Omega_i + \mathcal{O}(\max_{i,k}|I_{i,k}|^2\max_n \Delta t_n)\\ &= (h_1)_i^n - \Delta t_n\left(\delta_x(h_1u_1)+\delta_y(h_1v_1)\right)_i^{n+1/2} + \mathcal{O}(\max_{i,k}|I_{i,k}|^2\max_n \Delta t_n)\\ &= (h_1)_i^n - \Delta t_n\Big[(h_1\delta_x u_1 + h_1\delta_y v_1)_i^{n+1/2} - \Delta x\left((h_1)_x u\right)_i^{n+1/2}\\ &\qquad - \Delta y\left((h_1)_y v\right)_i^{n+1/2}\Big] + \mathcal{O}(\max_{i,k}|I_{i,k}|^2\max_n \Delta t_n).\end{aligned}$$

The CFL condition implies $\mathcal{O}(\max_{i,k}|I_{i,k}|) = \mathcal{O}(\Delta t_n)$. Therefore applying the discrete divergence condition we obtain

$$(h_1)_i^{n+1} = (h_1)_i^n - \Delta t_n\Delta x\left((h_1)_x u\right)_i^{n+1/2} - \Delta t_n\Delta y\left((h_1)_y v\right)_i^{n+1/2} + \mathcal{O}(\max_{i,k}|I_{i,k}|^3),$$

which is the desired condition $\mathcal{L}_1 h_1 = \mathcal{O}(\max_{i,k}|I_{i,k}|^3)$. The calculations for other components of W_i^{n+1} are analogous.
In particular, if we are at the lake at rest state additionally $\vec{u}_j = 0$, $j = 1, 2$ and therefore $F_2(W) = 0$. As a consequence it follows from (28) that $W_i^{n+1} = W_i^n$ exactly. □

In what follows we will show for the lake at rest state (14) that the evolutionary step meets the requirement of Lemma 3.1.

Lemma 3.2 (Well-balanced predictor step)
The time evolution step (25) *used for the system* (6) *is well-balanced for the lake at rest for* $r \in [0, 1)$.

Proof. If $r \in [0, 1)$ the lake at rest condition (14) implies $s = 0$ and

$$h_1 + h_2 + b = const. \wedge rh_1 + h_2 + b = const. \Rightarrow h_1 = const.$$

This means h_1 and thus ω is constant and, moreover, $\nabla V = 0$. Thus the evolution operator (25) reduces to

$$V(P) = \frac{1}{|S^{d-1}|}\int_{S^{d-1}}\sum_{i=1}^{n} r_\eta^i\, l_\eta^i V(x_e - \nabla_\eta\lambda_\eta^i(t_e - t_s), t_s)\, d\eta.$$

Next, since $V = const.$ we have,

$$\begin{aligned}V(P) &= \frac{1}{|S^{d-1}|}\int_{S^{d-1}}\sum_{i=1}^{n} r_\eta^i\, l_\eta^i V(x_e - \nabla_\eta\lambda_\eta^i(t_e - t_s), t_s)\, d\eta\\ &= \frac{1}{|S^{d-1}|}\int_{S^{d-1}}\underbrace{\sum_{i=1}^{n} r_\eta^i\, l_\eta^i}_{=RR^{-1}}\, d\eta\, V = \frac{1}{|S^{d-1}|}\underbrace{\int_{S^{d-1}} d\eta}_{=|S^{d-1}|}\, V = V.\end{aligned}$$

□

3.6 Positivity preservation of layer's height

Once the equilibrium values $V(P)$ at the volume face $I_{i,k}$ are calculated with the evolution operator (25) the conservative variables W_i^n are updated using (15). However, the multi-layer shallow water model indeed involves inequality constrains $h_i \geq 0$, $i = 1, 2$ that in general might not be fulfilled by a numerical discretization. Considering the time evolution (25), it seems to be very tedious to derive a constrain to ensure positivity of both layers. Therefore another technique is used here instead. Consider the first equation in (15)

$$\begin{aligned}(h_1)_i^{n+1} &= (h_1)_i^n - \frac{\Delta t_n}{|\Omega_i|} \sum_{k=1}^{M_i} |I_{i,k}| \underbrace{\sum_{l=1}^{K} \alpha_l F_1(W(x_{i,k}^l, t_n + \Delta t_n/2))\, n_{i,k}}_{=:F_i^k} \\ &= (h_1)_i^n - \frac{\Delta t_n}{|\Omega_i|} \sum_{k=1}^{M_i} |I_{i,k}|\, F_i^k, \end{aligned} \tag{29}$$

where F_1 denotes the first row of F. Recall that M_i is the number of faces belonging to the volume Ω_i. Next, split the fluxes into incoming and outgoing ones, i.e.

$$\left(F_i^k\right)^{in} := \min(0, F_i^k), \qquad \left(F_i^k\right)^{out} := \max(0, F_i^k).$$

Using these quantities the finite volume update (29) can be rewritten in the following way

$$(h_1)_i^{n+1} = (h_1)_i^n - \frac{(\Delta t_n)_{1,i}}{|\Omega_i|} \sum_{k=1}^{M_i} |I_{i,k}| \left(F_i^k\right)^{out} - \frac{1}{|\Omega_i|} \sum_{k=1}^{M_i} (\Delta t_n)_{1,k}^{in} |I_{i,k}| \left(F_i^k\right)^{in}.$$

Here $(\Delta t_n)_{1,i}$ is the time step for volume Ω_i and upper layer, whereas $(\Delta t_n)_{1,k}^{in}$ is the time step of a neighboring volume Ω such that $\partial\Omega \cap \partial\Omega_i = I_{i,k}$. Now since information propagates with a finite speed we will not allow that more mass flows out of the volume Ω_i during Δt_n than originally contained in Ω_i. Thus we have the constrain

$$(h_1)_i^n - \frac{(\Delta t_n)_{1,i}}{|\Omega_i|} \sum_{k=1}^{M_i} |I_{i,k}| \left(F_i^k\right)^{out} \geq 0. \tag{30}$$

Whenever (30) is not fulfilled, $(\Delta t_n)_{1,i}$ is adjusted to meet this constrain with equality, i.e.

$$(\Delta t_n)_{1,i} := \frac{|\Omega_i|\,(h_1)_i^n}{\sum_{k=1}^{M_i} |I_{i,k}| \left(F_i^k\right)^{out}} \geq 0. \tag{31}$$

Now this procedure is applied to every layer separately. The pseudo-code is given in Listing 1. Note that the global time step Δt_n will only be modified at nearly dry cells. Away from the shore line it still holds that $\Delta t_n = (\Delta t_n)_{1,i}$ for all cells Ω_i not being dry

or nearly dry.
Unlike [6], where a similar technique was used to treat nearly dry cells in the case of one layer shallow water flows, here, the adaption is done per volume without the necessity of an iterative adaption on the whole grid before the finite volume update is performed. Let us point out the procedure is applied to all volumes prior to the finite volume update.

Listing 1: Algorithm to ensure non-negativity after FV-update.

```
for i=1 to N in paralel do
  PredictSolution(i);
  ComputeFlux(i);
  ComputeSource(i);
  // adapt time step
  for k=1 to LAYERCOUNT do
    if OutflowMass(i,k)>MassInCell(i,k) then
      CalculateNewTime(i,k);
    endif
  endfor
endfor
...
for i=1 to N in paralel do
  FVupdate(i)
endfor
```

3.7 Modification of flux and source terms

The modifications described in Subsection 3.6 in general will break the well-balancing of a numerical scheme. This is due to the fact that different time steps for fluxes and sources are used. This behavior can be cured by adjusting the flux and source terms appropriately, cf. [6]. Consider again the proof of Lemma 3.1 for the lake at rest. There the crucial part was that the flux and source term should be treated together. With the modification described in the previous subsection we will have for the lake at rest state

$$W_i^{n+1} = W_i^n + \frac{(\Delta t_n)_{s,i}}{|\Omega_i|} \sum_{k=1}^{4} |I_{i,k}| \sum_{l=1}^{K} \alpha_l \left(\frac{1}{r} \underline{C_1} h_2 + \underline{C_4} h_1 \right) n_{i,k}$$
$$+ \frac{1}{|\Omega_i|} \sum_{k=1}^{4} (\Delta t_n)_{1,k}^{in} |I_{i,k}| \sum_{l=1}^{K} \alpha_l \left(\frac{1}{r} \underline{C_1} (r h_1 + b) + \underline{C_4} (h_2 + b) \right) n_{i,k}$$

instead. Note that the sources and fluxes are evaluated at the quadrature points $\left(x_{i,k}^l, t_n + \frac{\Delta t_n}{2}\right)$. More precisely only the pressure terms, i.e. the underlined terms in

the flux

$$F(W) = \begin{bmatrix} h_1u_1 & h_1v_1 \\ h_1u_1^2 + \underline{\frac{g}{2}h_1^2} & h_1u_1v_1 \\ h_1u_1v_1 & h_1v_1^2 + \underline{\frac{g}{2}h_1^2} \\ h_2u_2 & h_2v_2 \\ h_2u_2^2 + \underline{\frac{g}{2}h_2^2} & h_2u_2v_2 \\ h_2u_2v_2 & h_2v_2^2 + \underline{\frac{g}{2}h_2^2} \end{bmatrix}.$$

are critical. Thus rearranging these terms and combining then with the source terms will ensure well-balancing even in the case of nearly dry cells.

3.8 Adaption for lake at rest at wet/dry front

So far we have modified the FVEG scheme to treat nearly dry cells. However, the question of well-balancing at shore line is still open. Our aim is to obtain $\varepsilon = const.$ and $\omega = const.$, i.e. $\varepsilon_j = \varepsilon_{j+1}$ and $\omega_j = \omega_{j+1}$. Now, the problem arising here is depicted in Figure 6.

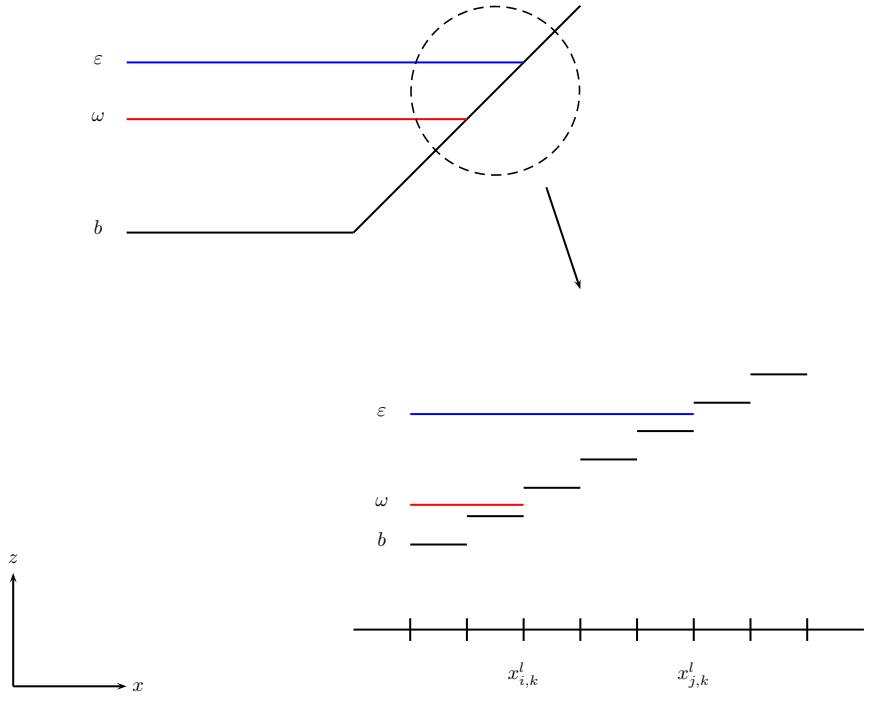


Figure 6: Situation at the shoreline.

The situation at the interface points $x^l_{i,k}$ and $x^l_{j,k}$ is of major interest. Let us consider the case where the water particles don't move, that is $\vec{u}_1 = \vec{u}_2 = 0$. In the discrete approximation due to jumps in ω and/or ε at wet/dry interfaces the evolution operator

will compute non vanishing velocities. This spurious movement must be trapped by the numerical method.

Unlike the situation at point $x^l_{j,k}$, where it is enough to modify the value of bottom topography b_{j+1} such that $b_{j+1} = \varepsilon_j$ to achieve well-balance, the situation at point $x^l_{i,k}$ is more difficult. Modifying b_{i+1} such that $\omega_i = b_{i+1}$ would break mass conservativity, since in general this would imply that $\varepsilon_i \neq \varepsilon_{i+1}$. A very similar approach has been proposed in [17], where the authors used a redefinition of discretization of the bottom topography to guarantee the lake at rest for the wet/dry fronts. However, they need to modify the bottom topography for each layer separately. Consequently there are two different values of bottom topography for wet/dry fronts. In our approach, on the other hand, we will not modify the bottom values but the values of ω and ε. Whenever the above situation arises we adapt as follows

$$\varepsilon_{j+1} := \varepsilon_j, \quad \omega_{i+1} := \omega_i.$$

Indeed, if $i \neq j$, we have at the point $x^l_{j,k}$ a situation where only one layer of fluid is not vanishing thus if its velocity is zero the adaption $\varepsilon_{j+1} := \varepsilon_j$ is done. This avoids a nonphysical removing of the shoreline since when a non-vanishing velocity is present in the layer no adaption is performed. On the other hand at the point $x^l_{i,k}$ we are at a interface between one and two layers. If here the velocities of both layers vanish the adaption $\omega_{i+1} := \omega_i$ is performed. Using a similar argumentation as above one realizes that also in this case no obstacles are removed.
The generalization of this technique to two spatial dimensions and multi-layer systems is straightforward. We apply the technique described for every neighboring edge and vertex. The pseudo-code is given in Listing 2. After the adaption we perform a bilinear recovery, the evolution step and finally the finite volume update.
Note that it is enough to apply the modification of ε and ω only prior to the evolution step. Since the evolution operator is well-balanced the finite volume update then works correctly without the necessity of any further adaption.

Listing 2: Algorithm for bottom adaption at wet/dry fronts.

```
for EveryUpperLayerNonDryVolume() do
  i=GetVolumeCoordinate();
  if VelocitiesVanish(i)
    for EveryUpperLayerDryNeighbor(i) do
      j=GetNeighborCoordinate();
      eps(j)=min(eps(i),eps(j));
    endfor
  endif
endfor

for EveryLowerLayerNonDryVolume() do
  i=GetVolumeCoordinate();
  if VelocitiesVanish(i)
```

```
    for EveryLowerLayerDryNeighbor(i) do
      j=GetNeighborCoordinate();
      w(j)=min(w(i),w(j));
    endfor
  endif
endfor
```

Let us point out that in the regions close to the wet/dry fronts very small values for the layer heights h_1 and h_2 may appear. To overcome the difficulties due to large round-off errors in computing the velocities $\vec{u}_i$ out of the mass fluxes $h_i\vec{u}_i$ for $i = 1, 2$ we define the velocities analogously as in [24], i.e.

$$\vec{u}_i := \frac{\sqrt{2}h_i\vec{u}_i}{\sqrt{h_i^4 + \max(h_i^4, \delta)}}, \qquad i = 1, 2,$$

where δ is a small a-priori chosen number. In our experiments $\delta := \min(\Delta x, \Delta y)^4$ was chosen.
Finally, in the case where either h_1 or h_2 vanishes we reduce our prediction step to that of the one-layer shallow water system.

4 Numerical results

In this section we demonstrate good performance and reliability of the new FVEG scheme through a series of numerical experiments. Note that no distinction between one- and two-dimensional test cases was done, i.e. all tests were computed using the full two-dimensional algorithm. For the discretization of the source terms we apply the well-balanced discretization from (20) as well as the adaption for dry states from the Sections 3.6 - 3.8. For the second order method we have used bilinear recovery with minmod limiter. Thus is we define a bilinear reconstruction $R_{\Delta x,\Delta y}$ that applies to a piecewise constant function V^n and computes a piecewise bilinear function $V_b := R_{\Delta x,\Delta y}V^n$ such that on any finite volume $\Omega_{i,j}$ with midpoint (x_i, y_j)

$$V_b(x, y)|_{\Omega_{i,j}} = f_0 + f_1(x - x_i) + f_2(y - y_j) + f_3(x - x_i)(y - y_j)$$

with coefficients

$$\begin{aligned}
f_0 &:= (1 - \mu_x^2\mu_y^2)V_{i+1,j+1} \\
f_1 &:= \text{minmod}\left(\delta_x\mu_y^2V^n_{i+1,j+1}, \delta_x\mu_y^2V^n_{i,j+1}\right) \\
f_2 &:= \text{minmod}\left(\delta_y\mu_x^2V^n_{i+1,j+1}, \delta_y\mu_x^2V^n_{i+1,j}\right) \\
f_3 &:= \text{minmod}\left(\delta_x\delta_y\frac{V^n_{i+1,j+1} + V^n_{i,j}}{2}, \delta_x\delta_y\frac{V^n_{i+1,j} + V^n_{i,j+1}}{2}\right).
\end{aligned}$$

Here $\mu_x V^n_{i,j} := \frac{1}{2}(V^n_{i,j} + V^n_{i-1,j})$ and $\delta_x V^n_{i,j} := \frac{1}{\Delta x}(V^n_{i,j} - V^n_{i-1,j})$ and analogous notation for y-direction. The minmod limiter is defined as

$$\operatorname{minmod}(x, y) := \frac{1}{2}\left(\operatorname{sgn}(x) + \operatorname{sgn}(y)\right)\min(|x|, |y|).$$

Hereafter, the prediction is computed by

$$V^{n+\frac{1}{2}} = EG_{\Delta t/2} V^n + EG_{\Delta t/2} R_{\Delta x, \Delta y} V^n$$

where $EG_{\Delta t/2}$ is the approximation of the evolution operator derived in Section 3.4, see also, e.g., [31].
Let us note that the evaluation of predicted solution at the intermediate time $t_{n+\frac{1}{2}}$ is needed to obtain the second order approximation in time as well. As already mentioned in the Section 3.1 the cell interface integrals are approximated by the Simpson rule in order to include multidimensional effects from the corner neighbors. The use of trapezoidal rule would yield unconditionally unstable scheme, that has been shown in [31]. The midpoint rule does not take corner neighbors into account, since the sonic curves lie just in the neighboring cells due to the CFL stability condition. However, the evolution operator includes an additional multidimensional source term (mantle integral). This has in particular impact for the second order scheme. Our experiments with the Gauss numerical quadratures showed the reduced stability of the whole FVEG scheme, cf. [22], and we are not using them here. Recall that the Jacobian matrices are frozen in the local average values of the neighboring cells connected to the integration point. We have also experimented with different linearizations and have seen that their influence on the quality of the whole solution of the nonconservative two-layer equations was in general negligible.

4.1 First dam break experiment

The first example was taken from [15]. We took $g = 9.81$, $r = 0.98$, $b = 0$ and choose the following initial data

$$h_1(0, x) = \begin{cases} 0.5, & \text{if } x < 0.5, \\ 0.55, & \text{if } x > 0.5, \end{cases}$$

$$h_2(0, x) = \begin{cases} 0.5, & \text{if } x < 0.5, \\ 0.45, & \text{if } x > 0.5, \end{cases}$$

$$u_1(0, x) = u_2(0, x) = 2.5.$$

This is a delicate test case. When trying to solve the problem using a splitting technique, e.g. by the path-consistent FVEG scheme from [20], spurious oscillations arise since the signature of the eigenvalues for the full and the splitted system differs. Thus wrong upwinding velocities appear. We solve the system for $x \in [0, 1]$ with 500 cells using CFL=0.9. The results presented in Figure 7 are comparable to those in [23]. Using the FVEG scheme no oscillations at the interface occur as it would be the case for splitting

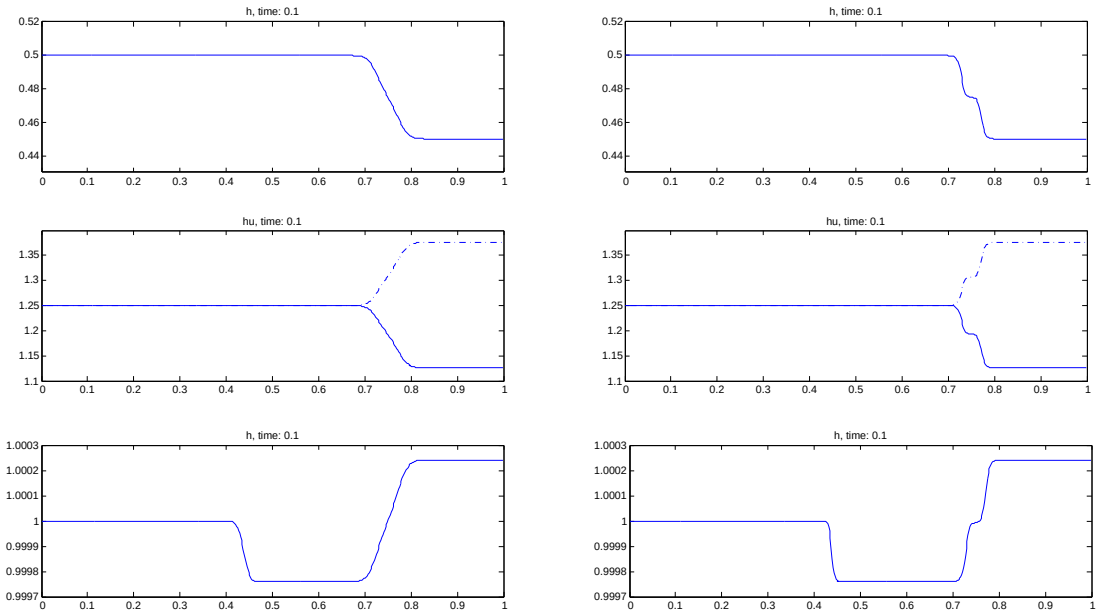


Figure 7: Results obtained for the first dam break problem 4.1 with first (left) and second order method (right) with $r = 0.98$. In the first row ω is plotted. The second row shows h_1u_1 (dashed-dotted) and h_2u_2 (solid). The surface ε is presented in the third row.

techniques.
To demonstrate robustness of our scheme also for $r \approx 1$ we present in Figure 8 the results for the case of $r = 0.99805$.

4.2 Second dam break experiment

The second test is taken from [8]. We set $g = 9.81$, $b = 0$ and choose the initial conditions in the following way

$$h_1(0,x) = \begin{cases} 0.2, & \text{if } x < 5, \\ 1.8, & \text{if } x > 5, \end{cases}$$
$$h_2(0,x) = \begin{cases} 1.8, & \text{if } x < 5, \\ 0.2, & \text{if } x > 5, \end{cases}$$
$$u_1(0,x) = u_2(0,x) = 0.$$

Let us point out that this test case is more sensitive than the previous one due to larger initial jump. For this test we study several cases of the density fraction r: $r = 0.7, 0.98$ and 0.99805. Again CFL=0.9 is used and the computation is done for $x \in [0, 10]$ using 500 cells. The results can be seen in Figures 9, 10 and 11. For both solvers, the hydrostatic and kinetic solver, from [8] only the test case of $r = 0.7$ was successfully solved. Our solution in this case is similar to that from [8] but we are also able to solve for higher r. A similar test was already considered in an earlier paper by Castro et

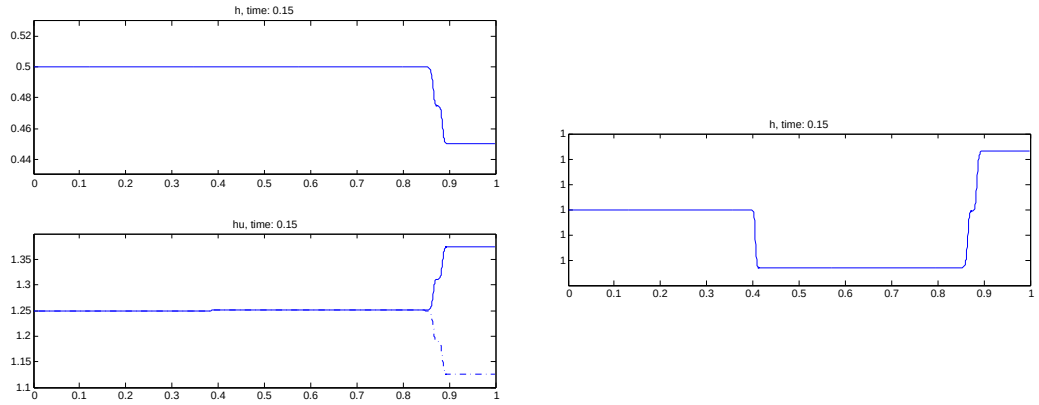


Figure 8: Results obtained for the first dam break problem 4.1 with $r = 0.98805$ *for the second order method on a* 2000 *grid. In the left column the interface* ω *(upper image) and the discharges* h_1u_1 *(dashed-dotted) and* h_2u_2 *(solid) are plotted. The right column show the water surface* ε*.*

al. [16]. We can notice very good comparison of our method also in the case $r = 0.99805$, see Figure 11.

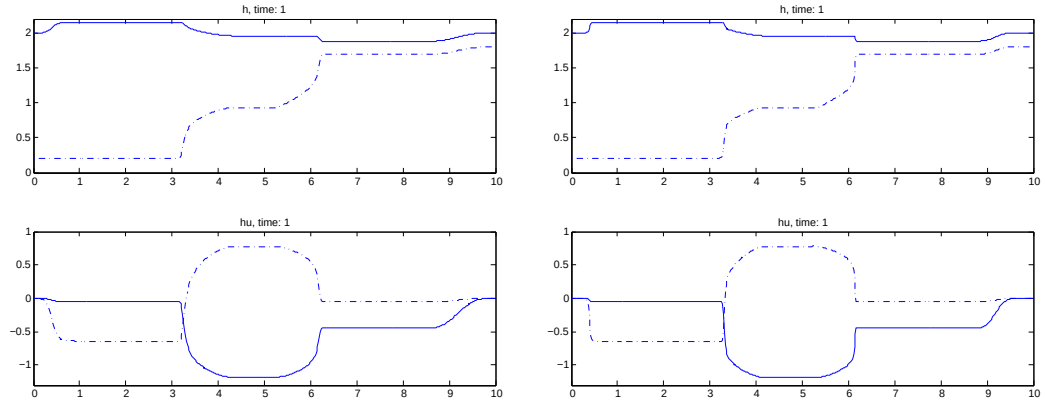


Figure 9: Results obtained for the second dam break problem 4.2 using $r = 0.7$ *with first (left) and second order method, left (right). First row shows the surface* ε *(solid) and the interface* ω *(dashed-dotted). In the second row* h_1u_1 *(dashed-dotted) and* h_2u_2 *(solid) is presented.*

4.3 Well-balancing test

The third test case illustrates the well-balanced behavior of the new FVEG scheme. If we do not approximate the source terms in a well-balanced manner, small perturbation

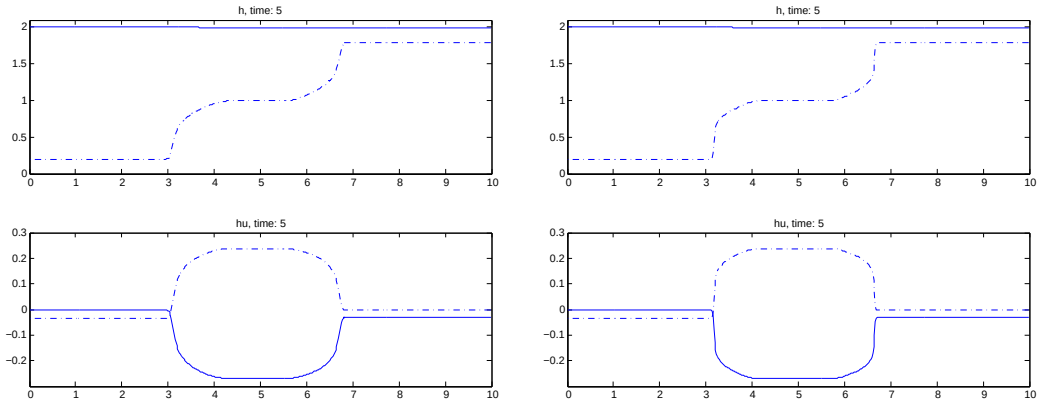


Figure 10: Results obtained for the second dam break problem 4.2 using $r = 0.98$ with first (left) and second order method (right). First row shows the surface ε (solid) and the interface ω (dashed-dotted). In the second row h_1u_1 (dashed-dotted) and h_2u_2 (solid) is presented.

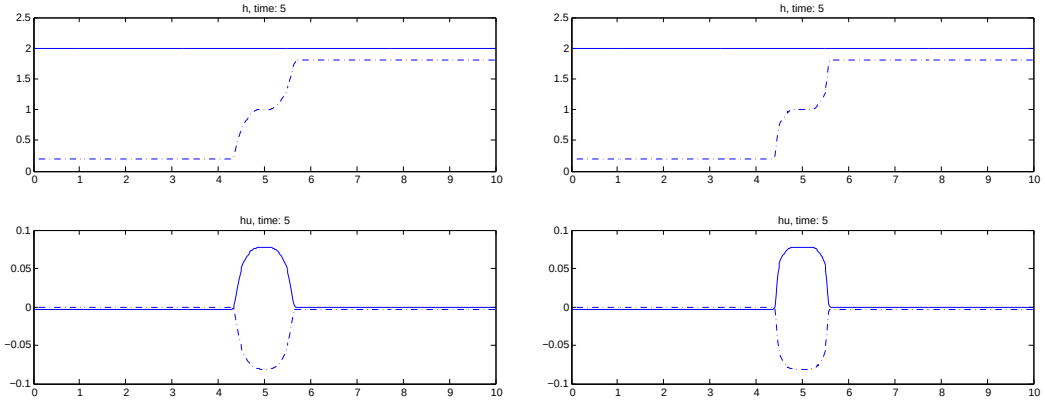


Figure 11: Results obtained for the second dam break problem 4.2 using $r = 0.99805$ with first (left) and second order method (right). First row shows the surface ε (solid) and the interface ω (dashed-dotted). In the second row h_1u_1 (dashed-dotted) and h_2u_2 (solid) is presented.

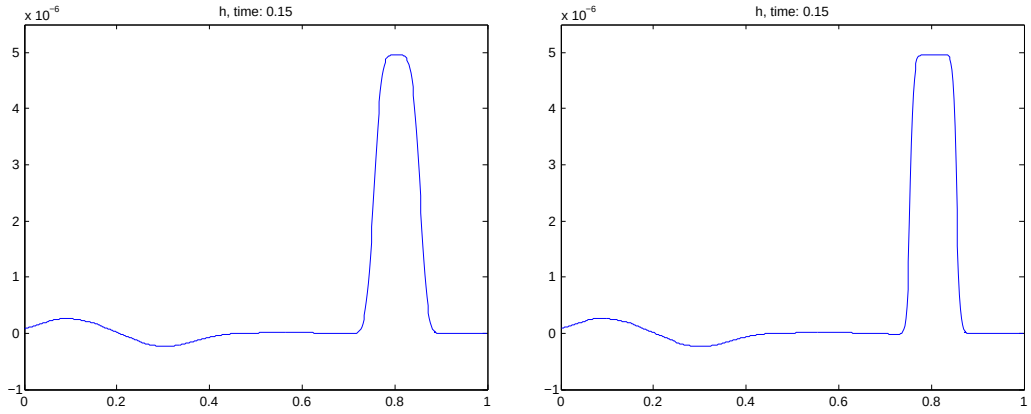


Figure 12: Surface ε obtained for the well-balance test 4.3 with smooth bottom topography for the first (left) and second order method (right).

in h_1 would yield strong oscillations of the solution. For this test we take

$$h_1(0,x) = \begin{cases} 1.00001, & \text{if } 0.1 < x < 0.2,, \\ 1.0, & \text{else,} \end{cases}$$
$$h_2(0,x) + b(0,x) = -1.0$$
$$u_1(0,x) = u_2(0,x) = 0,$$
$$b_s(x) = \begin{cases} \frac{1}{4}\left[\cos(10\pi(x-\frac{1}{2})) + 1\right] - 2, & \text{if } 0.4 < x < 0.6, \\ -2.0, & \text{else,} \end{cases}$$
$$b_d(x) = \begin{cases} -1.5, & \text{if } x > 0.5, \\ -2.0, & \text{else.} \end{cases}$$

We investigate both smooth b_s as well as discontinuous bottom topography b_d for the density fraction $r = 0.98$. The plot of the water surface ε can be seen in Figures 12 and 13. The solution is captured up to time $t = 0.15$ using a mesh with 500 cells and CFL=0.9. The initial perturbation generates two waves that propagate out of the computational domain and the solution converges to a steady state. Our results are comparable with those obtained in [23].

4.4 Geostrophic adjustment

Now we consider a two dimensional problem. It is a geostrophic adjustment simulation adapted from the analogous problem for the one-layer shallow water equations, see [11, 20]. The initial conditions are

$$h_1(0,x,y) = 1 + \frac{A_0}{2}\left(1 - \tanh\left(\frac{\sqrt{(\sqrt{\lambda}x)^2 + (y/\sqrt{\lambda})^2} - R_i}{R_E}\right)\right),$$
$$h_2(0,x,y) = 1, \qquad u_1(0,x,y) = v_1(0,x,y) = u_2(0,x,y) = v_2(0,x,y) = 0,$$

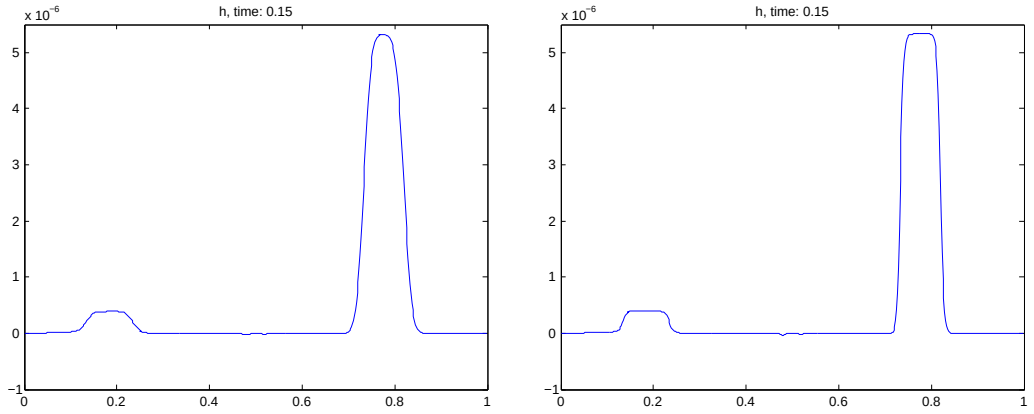


Figure 13: Surface ε obtained for the well-balance test 4.3 with discontinuous bottom topography for the first (left) and second order method (right).

where the parameters are $A_0 = 0.5$, $\lambda = 2.5$, $R_E = 0.1$ and $R_i = 1$. The gravity constant and the Coriolis parameter are set to $g = f = 1$. The fraction of the layer densities is $r = 0.98$. No bottom topography is considered. The computational domain $\Omega = [-10, 10]^2$ has been divided into 400×400 mesh cells. The CFL number was set to 0.6. In Figure 14 we see the time evolution of initial perturbation of water surface. Two circular shock waves propagating in the first and second layer can be recognized nicely. Behind the shocks there is elevation rotating clockwise. In order to represent free flow boundary conditions extrapolation has been implemented on the boundary of computational domain. Moreover, as time advances the solution tends to the geostrophic equilibrium. In Figure 15 one dimensional cuts obtained using the FVEG scheme with the source term discretization (20) and the path-consistent FVEG scheme [20] can be seen. We can clearly notice that both well-balanced source approximations yield same results.

4.5 Internal circular dam break

The next example simulates a two dimensional internal circular dam break. This test is taken from [14]. The initial conditions are

$$h_1(0, x, y) := \begin{cases} 1.8, & \text{if } x^2 + y^2 > 4 \\ 0.2, & \text{else} \end{cases},$$

$h_2(0, x, y) := 2 - h_1(0, x, y)$ and $\vec{u}_1 = \vec{u}_2 = 0$. The computational domain $\Omega := [-5, 5]^2$ is discretized by a 200×200 mesh. The CFL number is set to 0.9, $r = 0.98$, $f = 0 = b$ and $g = 9.81$. In Figure 16 we compare the results obtained by our FVEG scheme and by the second order path-conservative scheme based on the IFCP (**I**ntermediate **F**ield **C**apturing **P**arabola) Riemann solver from [21]. The results obtained by the IFCP method were kindly provided by M.J. Castro Díaz. We can clearly notice that both results are very similar showing well resolved solutions with the same shock speeds.

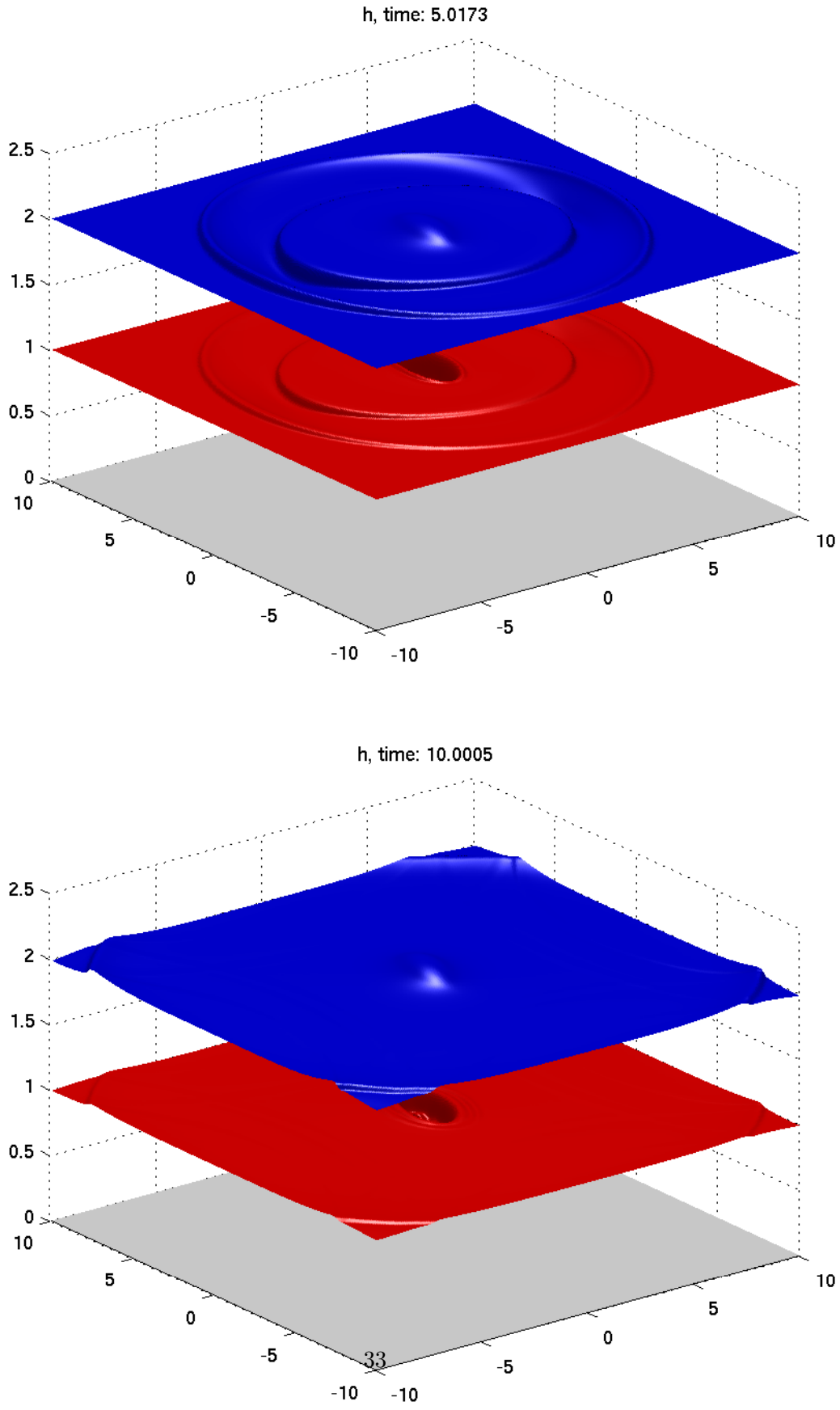


Figure 14: Water surface ε (blue) and interface ω (red) obtained for test 4.4.

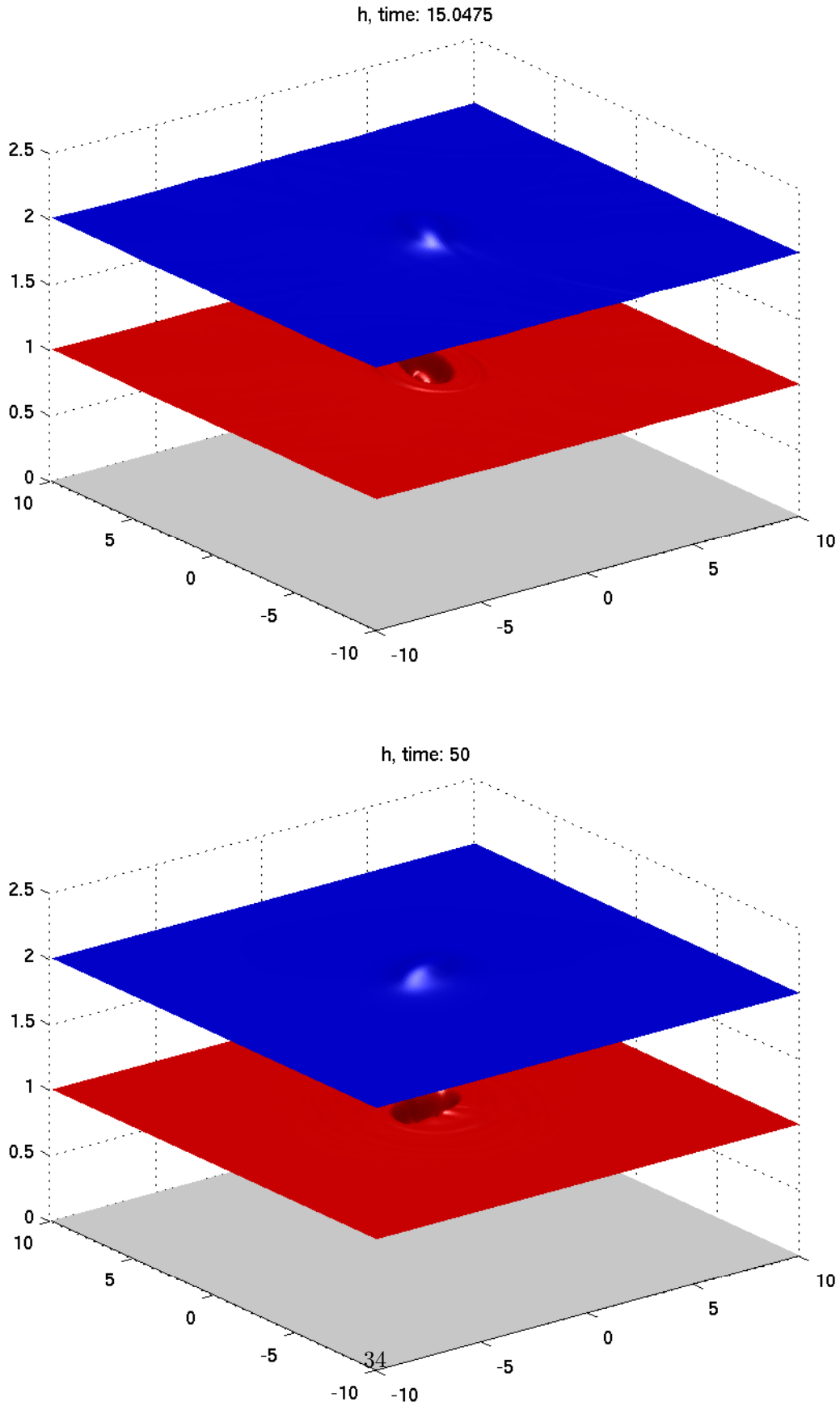


Figure 14: Water surface ε (blue) and interface ω (red) obtained for test 4.4 (cont.).

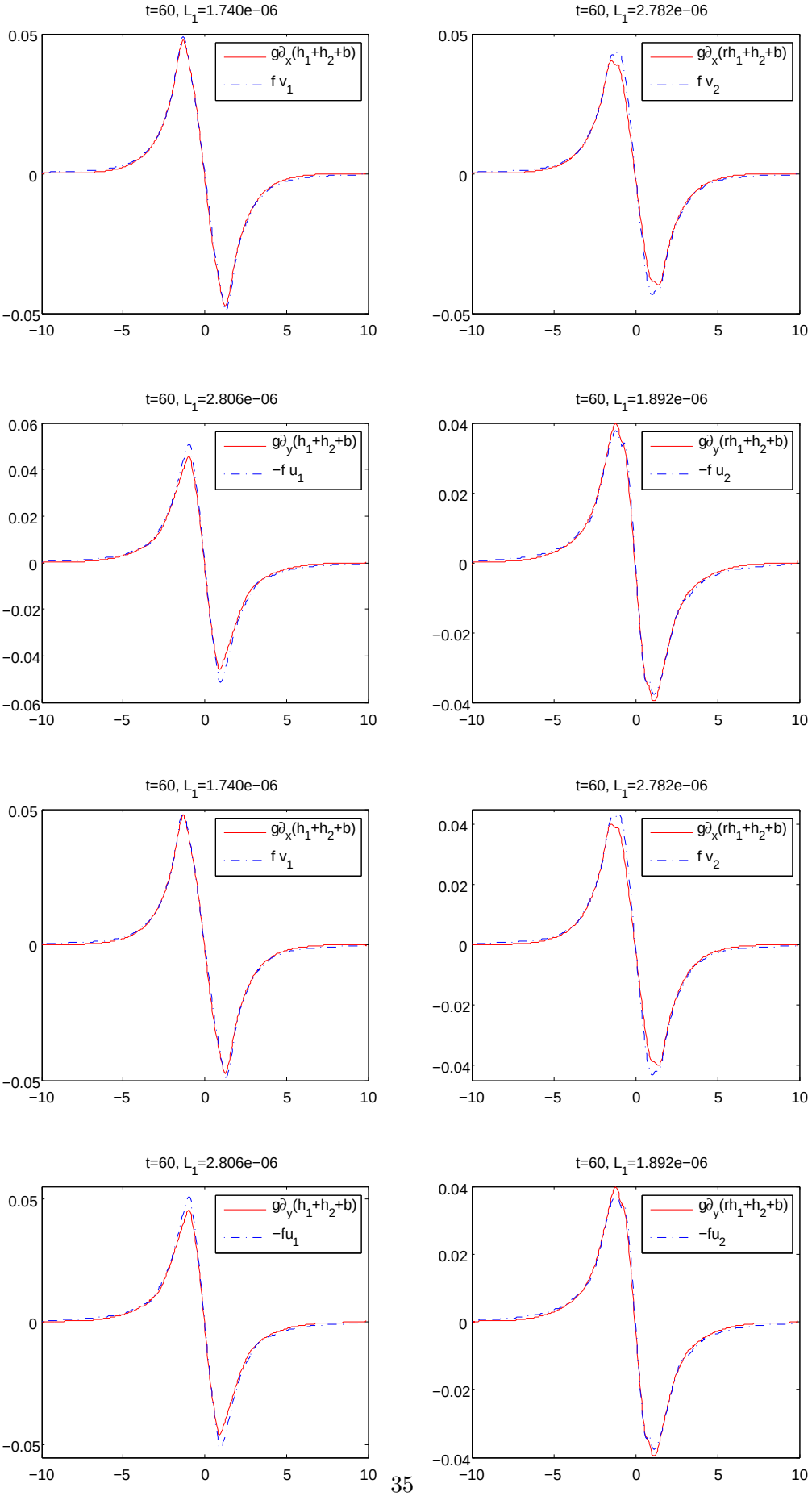


Figure 15: Geostrophic equilibrium. One dimensional cuts at $x = 0$ (left column) and $y = 0$ (right column), respectively. Source term discretization (20) *(upper group) and the path-consistent [20] (lower group).*

This is, in fact, quite surprising since shocks in nonconservative systems are expected to be sensible to the numerical viscosity. Consequently, this comparison indicates that for the two-layer shallow water system numerical methods based on different principles may lead to very similar discontinuous solutions. We have also performed an extensive testing of computational time for the FVEG and IFCP methods. Our numerical experiments show very similar computational costs, too.

In Figure 17 we have moreover compared the results obtained by our new second order FVEG scheme derived in this paper and the path-consistent FVEG scheme from [20]. We can notice that the the path-consistent FVEG scheme, based on the operator splitting, produces more numerical viscosity but again the influence on the shock speed is negligible.

4.6 Two dimensional interface propagation

This test is a generalization of the first dam break experiment in two spatial dimensions, see also [23]. The initial conditions are as follows

$$
\begin{aligned}
h_1(0,x,y) &= \begin{cases} 0.5, & \text{if } x \in \Omega, \\ 0.45, & \text{else}, \end{cases} \\
h_2(0,x,y) &= \begin{cases} 0.5, & \text{if } x \in \Omega, \\ 0.55, & \text{else}, \end{cases} \\
u_1(0,x,y) &= u_2(0,x,y) = v_1(0,x,y) = v_2(0,x,y) = 2.5, \\
b(x,y) &= 0,
\end{aligned}
$$

where Ω is given by

$$\Omega := \{(x+0.5)^2 + (y+0.5)^2 < 0.25\} \cup \{x < -0.5, y < 0.0\} \cup \{x < 0.0, y < -0.5\}.$$

The shallow water system was solved for $(x,y) \in [-1,1]^2$ on different meshes using 200×200, 400×400 and 800×800 mesh cells and CFL=0.6. The results at time $t = 0.1$ are given in Figures 18(a)-(f) and 19. In analogy to Experiment 4.1 we can notice the development of two shocks and the intermediate state in h_1. Comparing our results with the results obtained in [23] we can notice slightly more accurate resolutions and less oscillations in the FVEG scheme.

4.7 Two dimensional interface propagation with varying bottom topography

This test is analogous to the previous one except that we choose a spatially varying bottom topography

$$b(x,y) = 0.05\, e^{-100\cdot(x^2+y^2)} - 1.$$

We calculated the results up to $t = 0.1$ for $(x,y) \in [0,1]^2$ on meshes using 200×200, 400×400 and 800×800 cells and set CFL=0.6. The numerical solution depicted in Figures 21 (a)-(f) and 22 shows good agreement with the results obtained in [23].

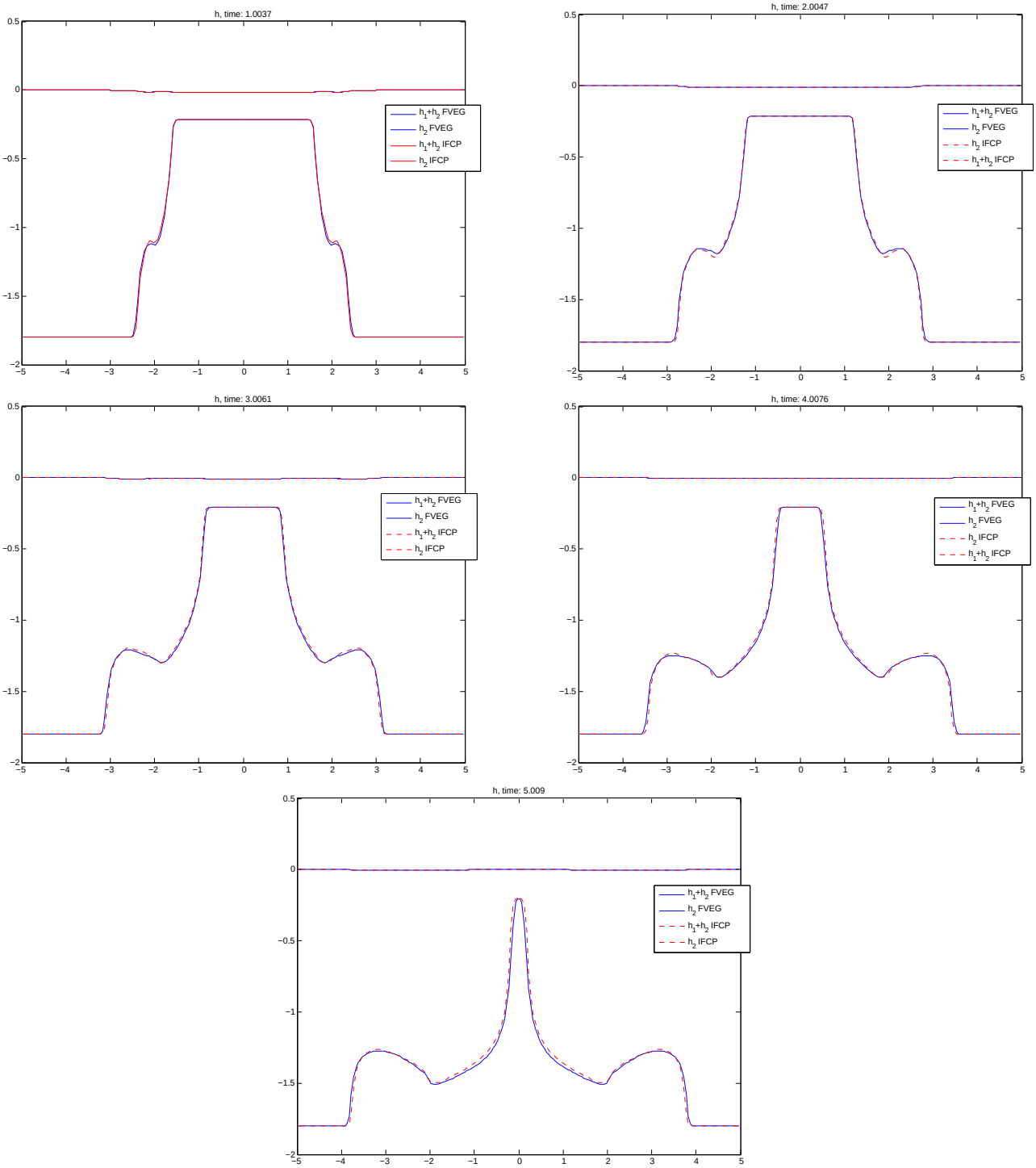


Figure 16: Comparison of the second order FVEG scheme and the IFCP scheme [14] for the circular dam break problem 4.5.

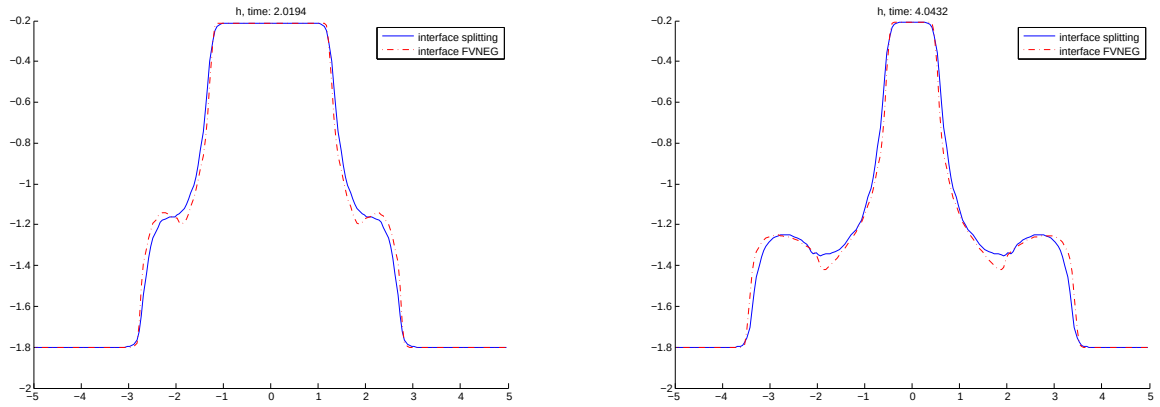


Figure 17: Comparison of the new FVEG scheme with the source discretization (20) *(dash line) and the path-consistent FVEG scheme from [20] (solid line) for the circular dam break problem 4.5.*

4.8 Experimental order of convergence

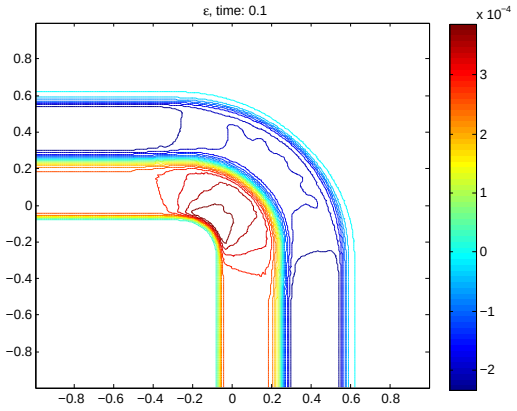
In order to test accuracy of our FVEG scheme we choose smooth initial conditions

$$\begin{aligned} h_1(0,x,y) &= 10 + e^{\sin(2\pi x)} \cdot \cos(2\pi y), \\ h_1u_1(0,x,y) &= \sin(\cos(2\pi x)) \cdot \sin(2\pi y), \\ h_1v_1(0,x,y) &= \cos(2\pi x) \cdot \cos(\sin(2\pi y)), \\ h_2(0,x,y) &= 2, \\ u_2(0,x,y) &= v_2(0,x,y) = 0, \\ b(x,y) &= \sin(2\pi x) + \cos(2\pi y). \end{aligned}$$

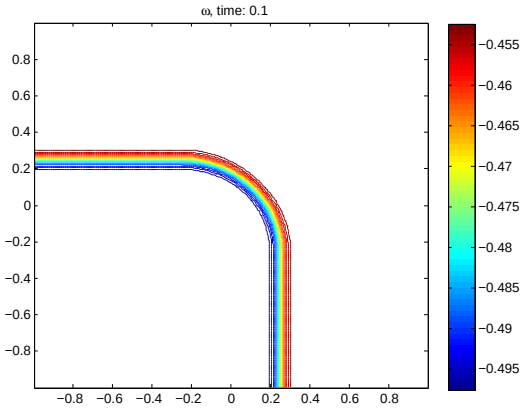
The computational domain is $[0,1]^2$ and the numerical solution was calculated up to $t = 0.05$ using CFL=0.6. For cell interface integrals the Simpson rule was used and no limiters were applied. The bilinear recovery was computed by the central differences for slopes, cf. [31]. Table 1 confirms the expected second order of accuracy.

4.9 Two dimensional lake at rest experiment

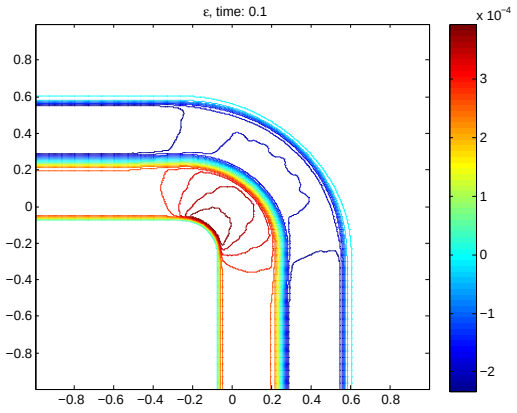
In this test we want to check the ability of the numerical scheme to preserve the lake at rest equilibrium. Indeed, we want to verify numerically Lemma 3.1 and 3.2. The initial



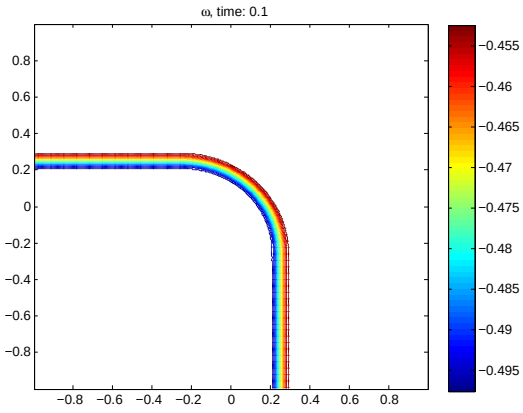
(a) ε



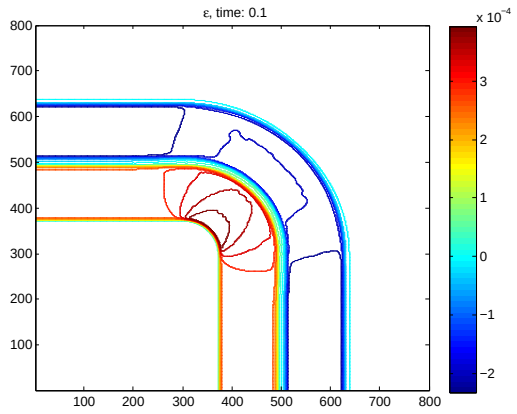
(b) ω



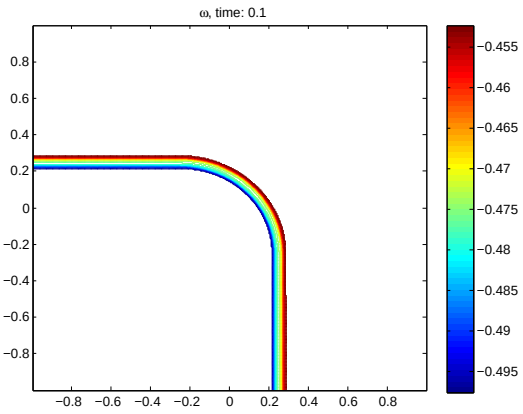
(c) ε



(d) ω



(e) ε



(f) ω

Figure 18: Results obtained for test 4.6. The upper to lower row are the results of the second order algorithm for 200×200, 400×400 *and* 800×800 *cells, respectively.*

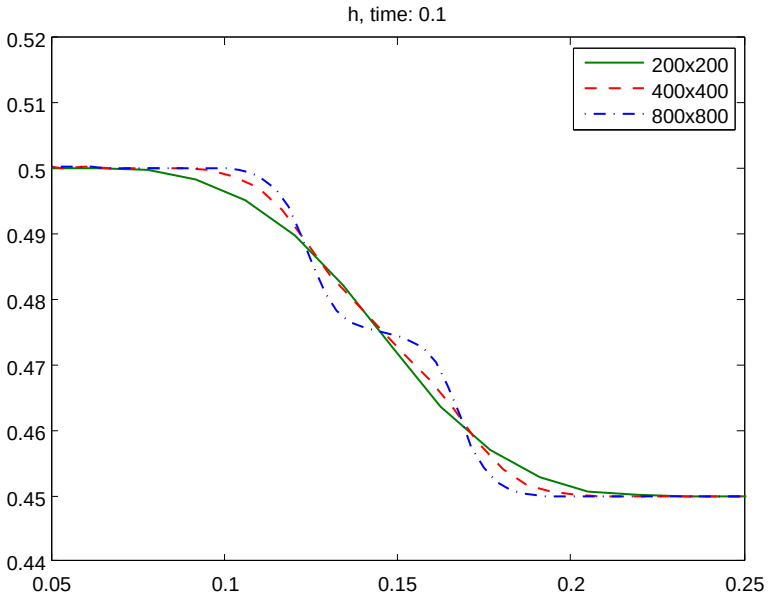


Figure 19: Problem 4.6. Cut $x = y$ of ω on meshes 200×200, 400×400 and 800×800.

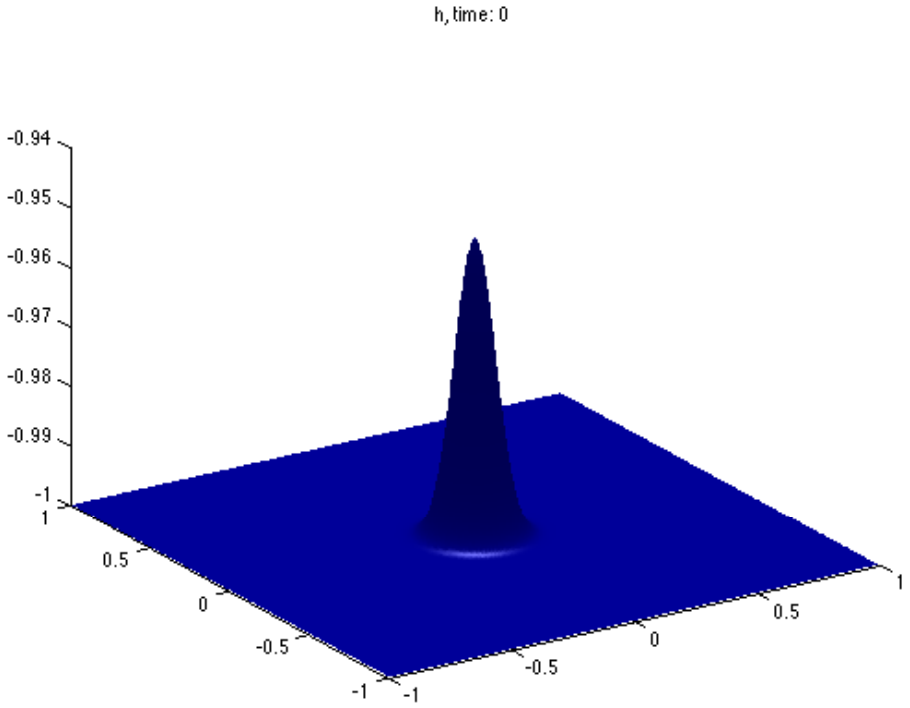
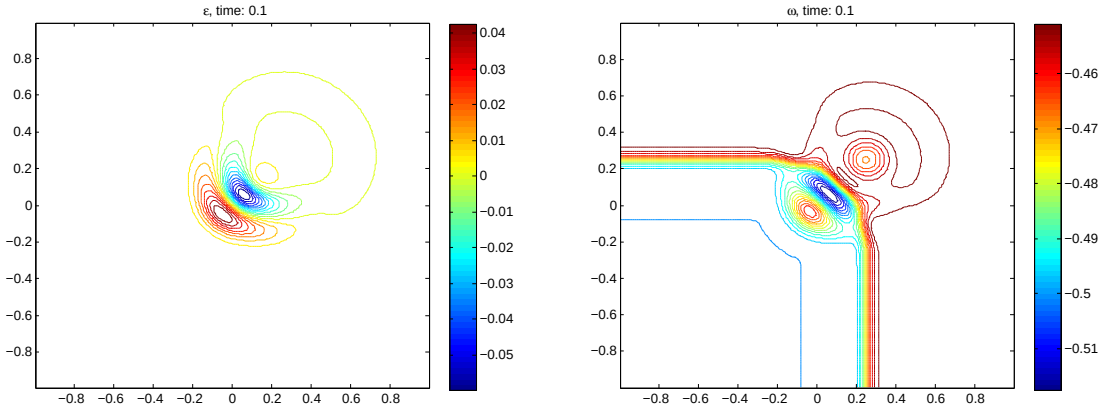
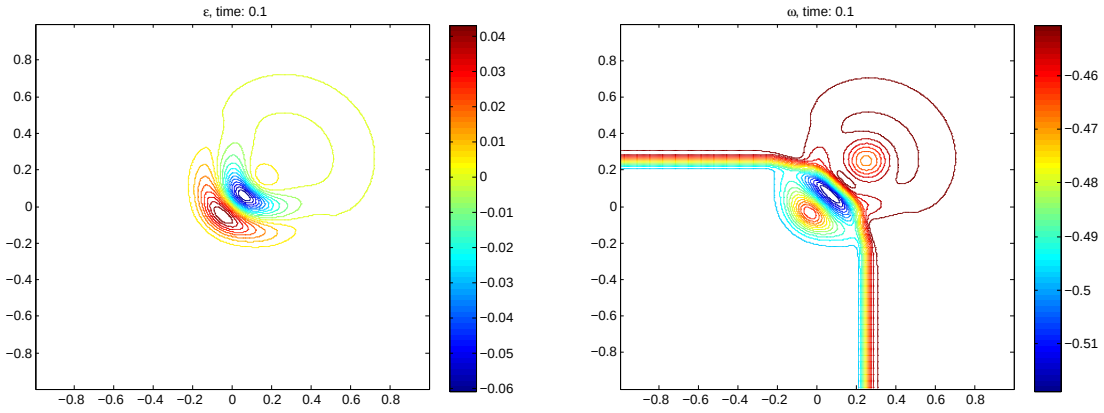


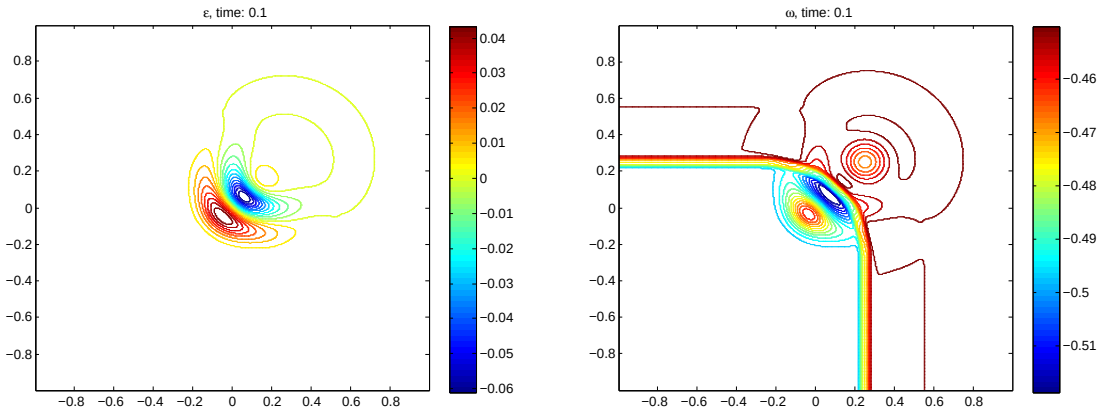
Figure 20: Bottom topography for the test 4.7.



(a) ε (b) ω



(c) ε (d) ω



(e) ε (f) ω

Figure 21: Results obtained for test 4.7. The upper to lower row are the results of the second order algorithm for 200×200, 400×400 *and* 800×800 *cells, respectively.*

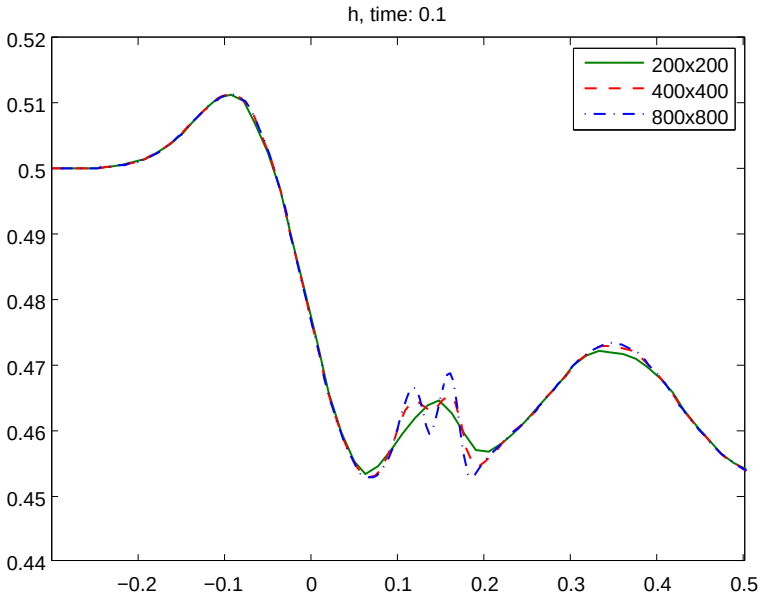


Figure 22: Problem 4.6. Cut $x = y$ of ω on meshes 200×200, 400×400 and 800×800.

No. cells	h_1	EOC	h_2	EOC
20×20	2.5155e-02		9.5881e-03	
40×40	5.7887e-03	2.1195	2.5155e-03	1.9304
80×80	1.2775e-03	2.1800	6.3063e-04	1.9960
160×160	2.8574e-04	2.1605	1.5581e-04	2.0170
320×320	6.1849e-05	2.2079	3.8325e-05	2.0234
No. cells	u_1	EOC	u_2	EOC
20×20	7.8012e-02		2.0035e-02	
40×40	1.5628e-02	2.3196	4.3540e-03	2.2022
80×80	3.3066e-03	2.2407	1.0165e-03	2.0987
160×160	7.0742e-04	2.2247	2.4500e-04	2.0528
320×320	1.5367e-04	2.2027	5.5853e-05	2.1331
No. cells	v_1	EOC	v_2	EOC
20×20	1.6859e-01		3.1914e-02	
40×40	3.3968e-02	2.3113	6.4839e-03	2.2992
80×80	6.9749e-03	2.2839	1.3617e-03	2.2515
160×160	1.5272e-03	2.1913	3.0344e-04	2.1659
320×320	3.6685e-04	2.0576	7.2801e-05	2.0594

Table 1: Results obtained for the EOC-test 4.8.

data is chosen in the following way

$$
\begin{aligned}
h_1(0,x,y) &= \frac{K_2 - K_1}{r-1} \\
h_2(0,x,y) &= \frac{1}{1-r}\left(K_2 - r \cdot K_1\right) - b \\
b(x,y) &= \begin{cases} 0.2, & \text{if } \|(x,y)\|_\infty < 0.5, \\ 0.1, & \text{else,} \end{cases} \\
u_1(0,x,y) &= u_2(0,x,y) = v_1(0,x,y) = v_2(0,x,y) = 0,
\end{aligned}
$$

where $\|.\|_\infty$ denotes the maximum norm and $r := \frac{\rho_1}{\rho_2} = 0.5$. The parameters are $K_1 = 1.0$, $K_2 = 0.7$, $g = 9.81$ and $f = 0$. Thus we have

$$
h_1 + h_2 + b = K_1, \qquad r \cdot h_1 + h_2 + b = K_2.
$$

Experimental tests were done for several grids using $5 \times 5, 10 \times 10, 20 \times 20, \ldots, 500 \times 500$ mesh cells. Here we have used different CFL numbers from $(0, 1]$. The results always yield

$$
\|h_1 + h_2 + b - K_1\|_{L^1} = 0, \qquad \|r \cdot h_1 + h_2 + b - K_2\|_{L^1} = 0.
$$

In order to calculate the L^1-norms the double precision arithmetic was applied.
Now, let us perturb h_1 by a small disturbance wave, i.e.

$$
h_1(0,x,y) = \frac{K_2 - K_1}{r-1} + \begin{cases} 10^{-3}, & \text{if } \|(x,y)\|_\infty < 0.25, \\ 0.0, & \text{else.} \end{cases} \tag{32}
$$

The initial perturbation develops waves that propagate out of the computational domain and the solution converges to a steady state. We have used free flow boundary conditions. One dimensional cuts are plotted in time $0.1, 0.2, 0.3$ in Figure 23 (a)-(f).

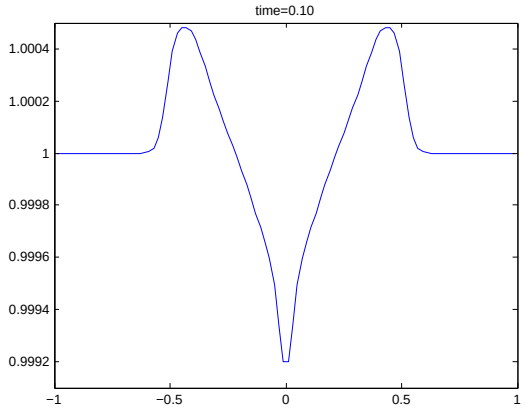
4.10 Run-up on canonical island

In the last experiment a solitary wave in each layer running up a canonical island is simulated. This test is a generalization of the one performed in [9] for one layer flow. The computation is done on the domain $\Omega = [0, 25] \times [0, 30]$ with a canonical island present at the center of Ω

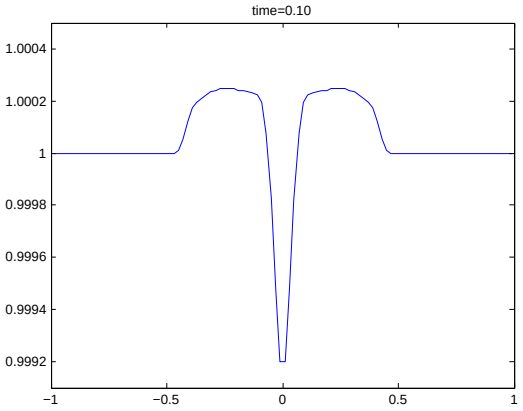
$$
b(x,y) := \begin{cases} 0.625, & (x-12.5)^2 + (y-15)^2 \le 1.21, \\ \frac{3.6 - \sqrt{(x-12.5)^2 + (y-15)^2}}{4}, & (x-12.5)^2 + (y-15)^2 \le 12.96, \\ 0, & \text{else.} \end{cases}
$$

Layer depths are chosen as

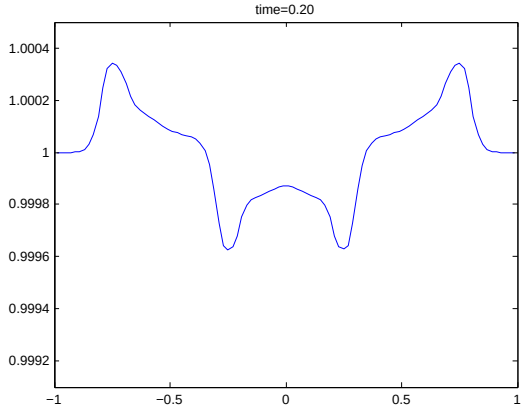
$$
\begin{aligned}
h_2(0,x,y) &= \max(0, H_2 - b(x,y)), \\
h_1(0,x,y) &= \max(0, H_1 - h_2(0,x,y) - b(x,y)),
\end{aligned}
$$



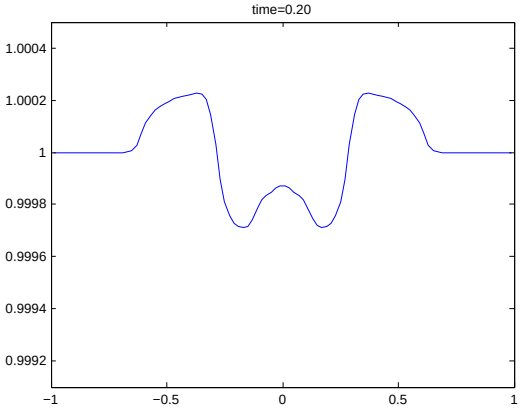
(a) Cut at $y = 0$



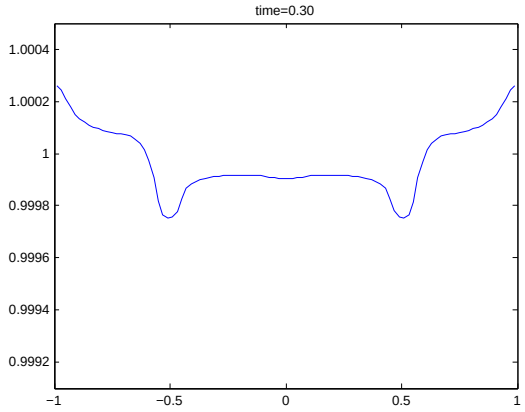
(b) Cut at $x = y$



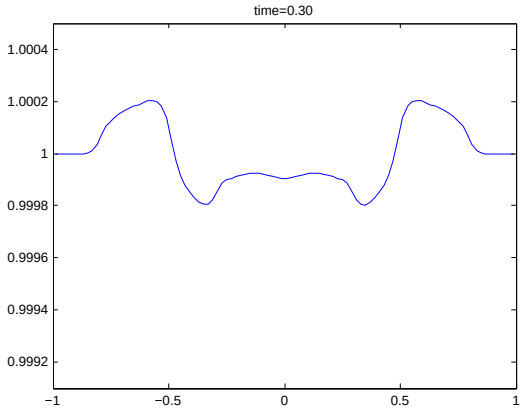
(c) Cut at $y = 0$



(d) Cut at $x = y$



(e) Cut at $y = 0$



(f) Cut at $x = y$

Figure 23: Surface ε obtained for test 4.9 with perturbed h_1 given by (32).

where $H_1 = 0.35$ and $H_2 = 0.15$ and the velocities vanish initially. The height of solitary wave entering the domain at time $t = 0$ through the left boundary in the i-th layer is given by

$$w_i(t, 0, y) = \alpha H_i \left(\frac{1}{\cosh((t - 3.5)\xi_i \sqrt{g H_i}/L)} \right)^2$$

with $L = 15$, $\alpha = 0.1$ and $\xi_i = \sqrt{\frac{3\alpha(1+\alpha)L^2}{4H_i^2}}$. The density ratio is taken as $r = 0.7$. Figure 24 shows the time evolution of h_1 and h_2 up to time 30 using CFL= 0.6. Upon hitting the island the amplitude of both solitary waves increases. As a consequence the island is flooded and reflecting waves are generated propagating in the opposite direction.

Conclusions

We have derived a new well-balanced **F**inite **V**olume **E**volution **G**alerkin scheme for two-dimensional multilayer shallow water flows including wet/dry fronts. The FVEG method is a predictor-corrector scheme. A multidimensional evolution operator for the multilayer shallow water system is derived from the theory of bicharacteristics. The evolution operator, needed in the predictor step, is constructed in such a way that all of the infinitely many directions of wave propagation are taken into account, see also [29], [31], [28] for other applications. Since for the two-layer shallow water equations the eigenstructure is not readily available we have derived analytical formulae for the eigenstructure of this non-conservative system. Due to efficiency reasons the Newton-Raphson iterative method has been used to compute the eigenvalues that are needed for the evolution operator. However, if a numerical method relying on the eigenvalues is to be developed for e.g. the GPU the analytical formulation is superior to the iterative method. Moreover, the FVEG method is well-balanced. We have proven that the corresponding approximation of the source terms yields the exactly well-balanced scheme for the lake at rest equilibrium, cf. Lemmas 3.1 and 3.2. In the case of two-layer geostrophic equilibrium the source term approximation gives third order well-balanced finite volume update, cf. Lemma 3.1. Furthermore, we have presented a relatively simple modification of the finite volume update to obtain a higher order positivity preserving method, cf. also [6] for a similar approach in the case of one-layer shallow water system. This is particularly important in the case of costal flow, see Figure 24.

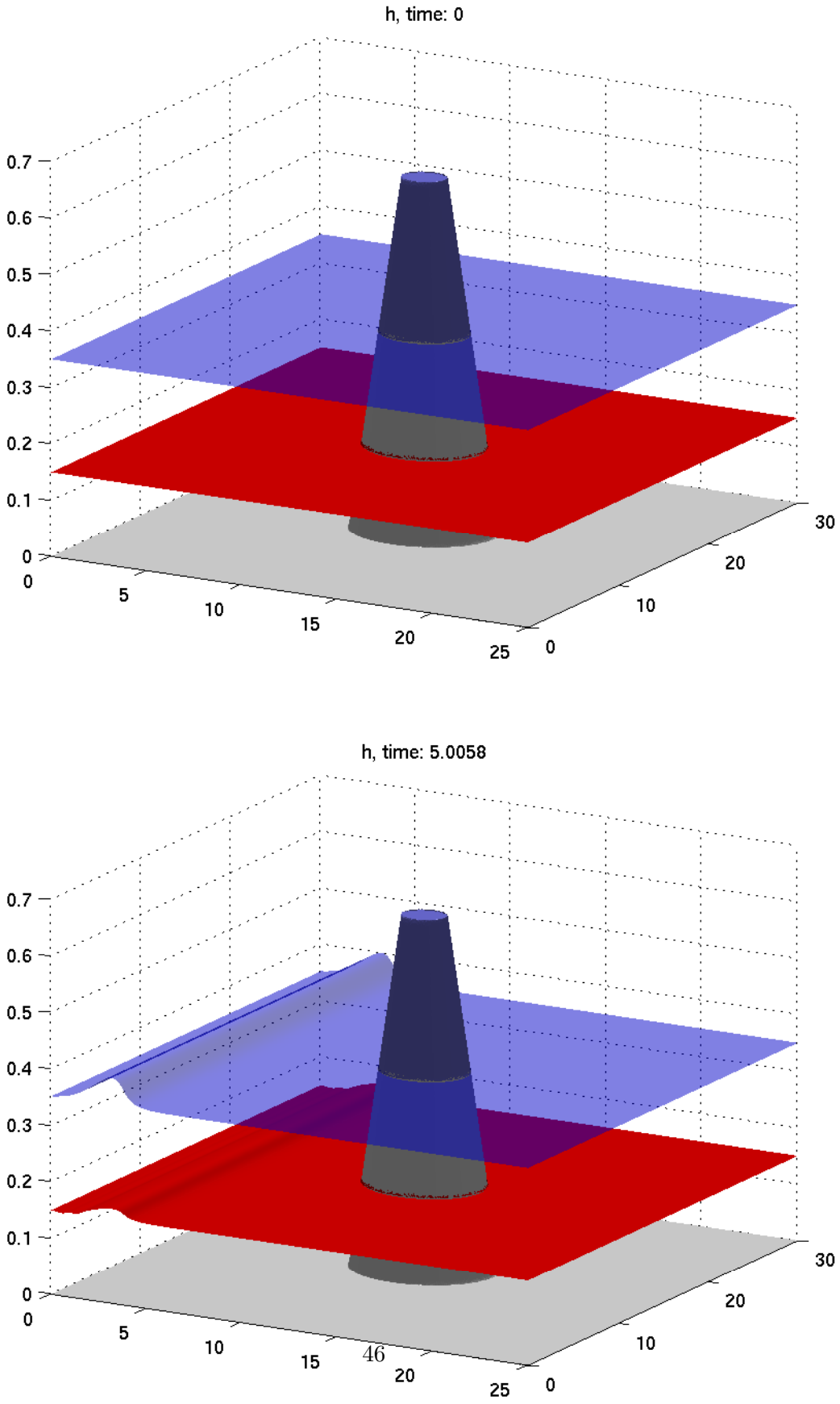
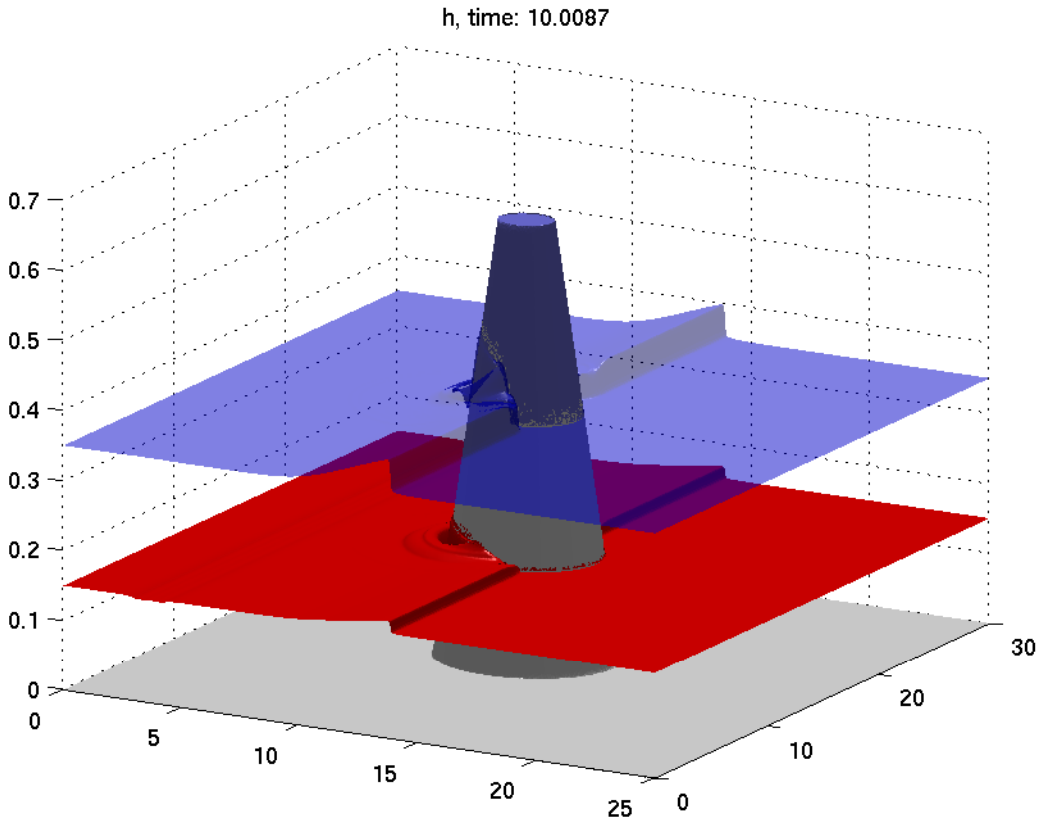


Figure 24: Results obtained for test 4.10.



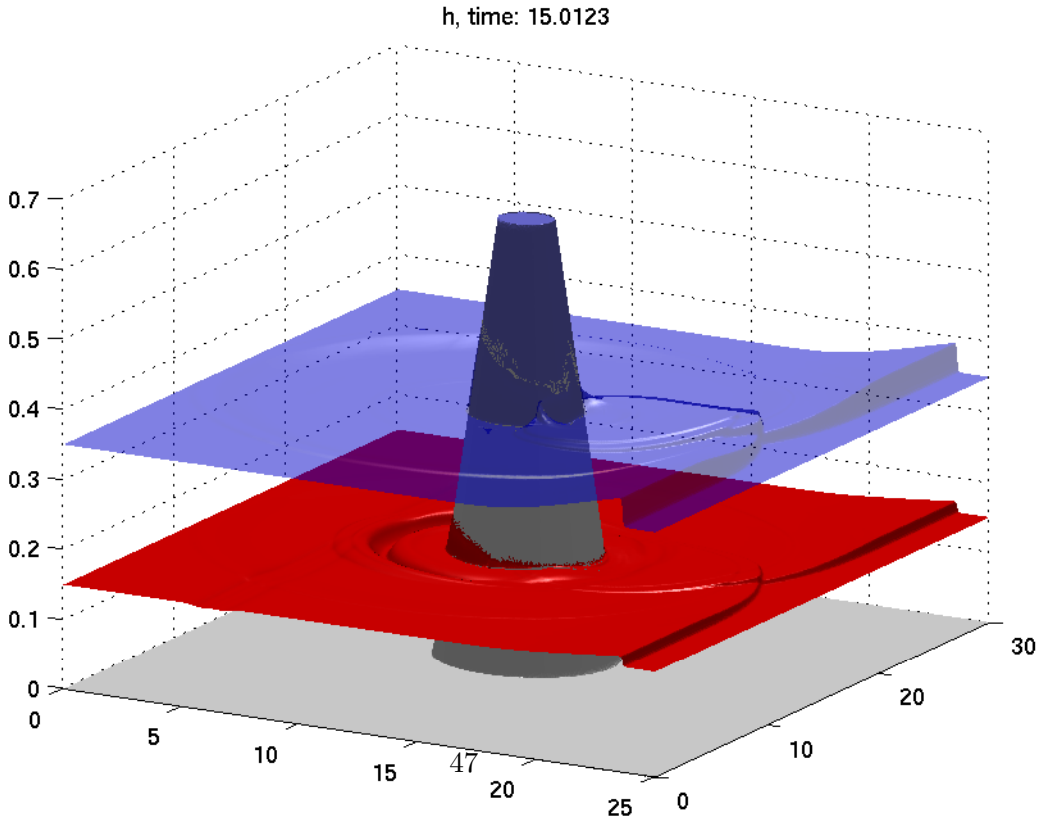
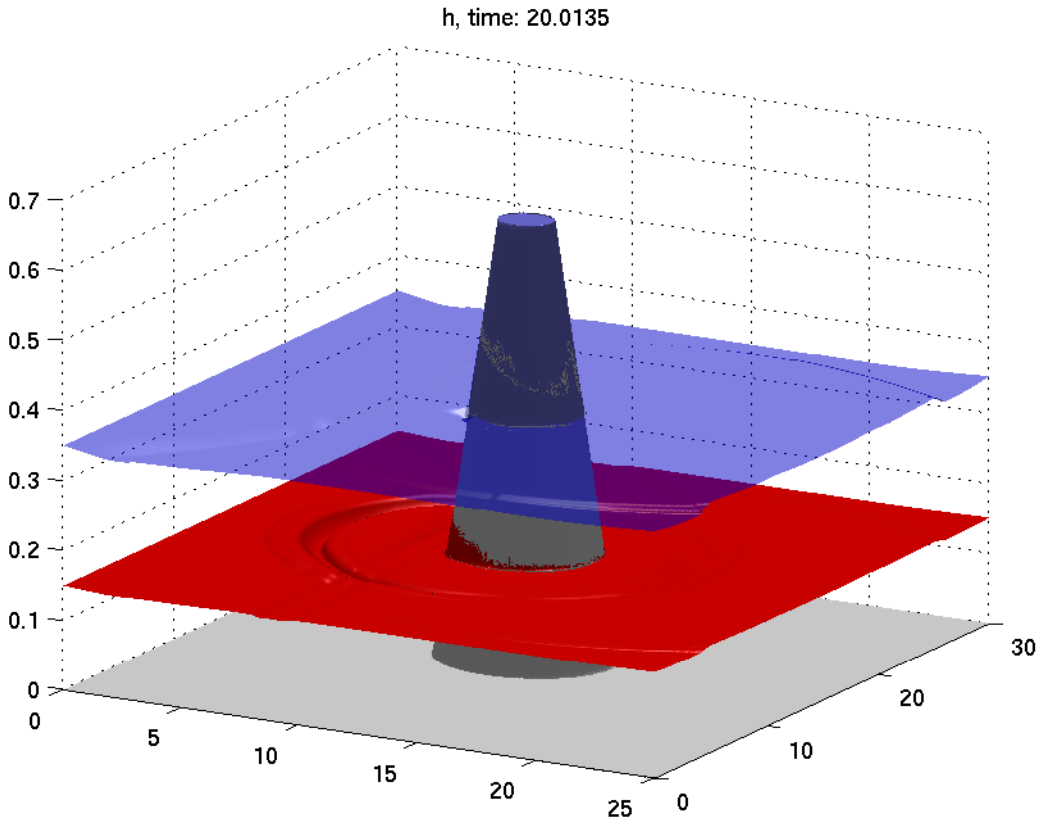


Figure 24: Results obtained for test 4.10 (cont.).



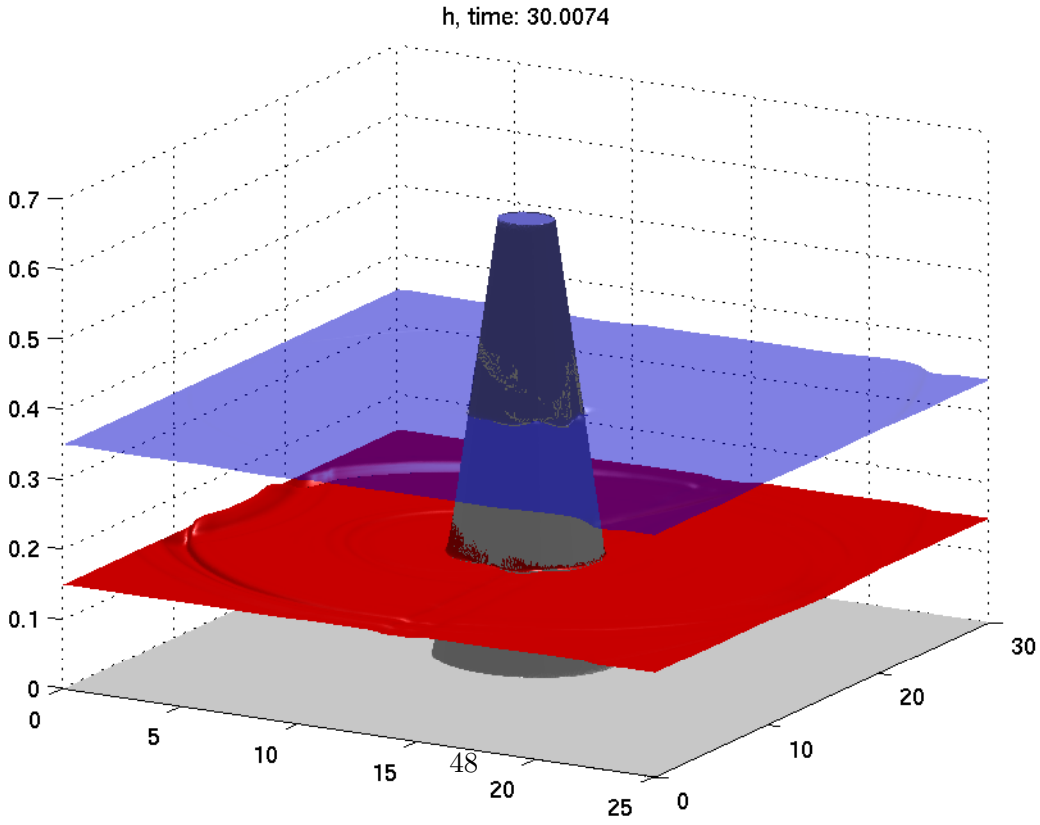


Figure 24: Results obtained for test 4.10 (cont.).

References

[1] R. Abgrall and S. Karni. Two-layer shallow water systems: a relaxation approach. *SIAM J. Sci. Comp.*, 31(3):1603–1627, 2009.

[2] R. Abgrall and S. Karni. A comment on the computation of non-conservative products. *J. Comput. Phys.*, 229(8):2759–2763, 2010.

[3] K.R. Arun, M. Kraft, M. Lukáčová-Medvid'ová, and Phoolan Prasad. Finite volume evolution Galerkin method for hyperbolic conservation laws with spatially varying flux functions. *J. Comput. Phys.*, 228(2):565–590, 2009.

[4] E. Audusse. A multilayer Saint-Venant model: derivation and numerical validation. *Discrete Continuous Dyn. Syst. B*, 5:189–214, 2005.

[5] E. Audusse, F. Benkhaldoun, J. Sainte-Marie, and M. Seaid. Multilayer Saint-Venant equations over movable beds. *Discrete Continuous Dyn. Syst. B*, 15(4):917–934, 2011.

[6] A. Bollermann, S. Noelle, and M. Lukáčová-Medvid'ová. Finite volume evolution Galerkin methods for the shallow water equations with dry beds. *Commun. Comput. Phys.*, 10:371–404, 2011.

[7] F. Bouchut and V. Zeitlin. A robust well-balanced scheme for multi-layer shallow water equations. *Discrete Continuous Dyn. Syst. B*, 13(4):739–758, 2010.

[8] F. Bouchut and T. Morales de Luna. An entropy satisfying scheme for two-layer shallow water equations with uncoupled treatment. *M2AN*, 42(4):683–698, 2008.

[9] M.J. Briggs, C.E. Synolakis, G.S. Harkins and D.R. Green. Laboratory experiments of tsunami runup on a circular island. *Pure Appl. Geophys.*, 144(3):569–593, 1995.

[10] M.J. Castro, Ph.G. LeFloch, M.L Muoz-Ruiz and C. Parés. Why many theories of shock waves are necessary. Convergence error in formally path-consistent schemes. *J. Comput. Phys.*, 227(17):8107-8129, 2008.

[11] M.J. Castro, J.A. López, and C. Parés. Finite volume simulation of the geostrophic adjustment in a rotating shallow-water system. *J. Sci. Comput.*, 31(1):444–477, 2008.

[12] M. Castro, J.T. Frings, S. Noelle, C. Parés, and G. Puppo. On the hyperbolicity of two- and three-layer shallow water equations. *Proceedings: Hyperbolic Problems: Theory, Numerics, Applications*, 2010.

[13] M.J. Castro-Díaz, E.D. Fernández-Nieto, J.M. González-Vida and C. Parés-Madroñal. Numerical treatment of the loss of hyperbolicity of the two-layer shallow-water system. *J. Sci. Comput.*, 48(1):16-40, 2011.

[14] M.J. Castro, E.D. Fernández-Nieto, A.M. Ferreiro, J.A. García-Rodríguez and C. Parés. High order extensions of Roe schemes for two-dimensional nonconservative hyperbolic systems *J. Sci. Comput.*, 39:67–114, 2009.

[15] M. Castro, J. Macías and C. Parés. A q-scheme for a class of systems of coupled conservation laws with source term. application to a two-layer 1-d shallow water system. *M2AN*, 35(1):107-127, 2001.

[16] M. Castro, J.M. Gallardo, and C. Parés. High order finite volume schemes based on reconstruction of states for solving hyperbolic systems with non conservative products. Applications to shallow water systems. *Math. Comp.*, 75:1103-1134, 2006.

[17] M.J. Castro, A. Ferreiro, J.A. García, J.M. Gonzáles, J.M. Macías, C. Parés and M.E. Vázquez. On the numerical treatment of wet/dry fronts in shallow flows: applications to one-layer and two-layer systems. *Math. Comp. Model.*, 42(3-4):419–439, 2005.

[18] R. Courant and D. Hilbert. *Methods of Mathematical Physics*, Vol. 2, Wiley, 1989.

[19] G. Dal Maso, P. LeFloch, and F. Murat. Definition and weak stability of nonconservative products. *J. Math. Pures Appl.*, 74:483–548, 1995.

[20] M. Dudzinski and M. Lukáčová-Medvid'ová. Well-balanced path-consistent finite volume EG schemes for the two-layer shallow water equations. In *Computational Science and High Performance Computing IV*, Notes on Numerical Fluid Mechanics and Multidisciplinary Design, Vol. 115, The 4th Russian-German Advanced Research Workshop, E. Krause et al. (eds.), pp. 121-135, Springer 2011.

[21] E.D. Fernández-Nieto, M.J. Castro Díaz and C. Parés. On an intermediate field capturing Riemann solver based on a parabolic viscosity matrix for the two-layer shallow water system. *J. Sci. Comput.*, 48:117-140, 2011.

[22] A. Hempel. *On the Application of Gaussian Quadrature for the Finite Volume Evolution Galerkin Schemes*, Study work, Hamburg University of Technology, 2009.

[23] A. Kurganov and G. Petrova. Central-upwind schemes for two-layer shallow water equations. *SIAM J. Sci. Comp.*, 31(3):1742–1773, 2009.

[24] A. Kurganov and G. Petrova. A second-order well-balanced positivity preserving central-upwind scheme for the Saint-Venant system *Commun. Math. Sci.*, 5(1):133–160, 2007.

[25] T. Kröger and M. Lukáčová-Medvid'ová. An evolution Galerkin scheme for the shallow water magnetohydrodynamic equations in two space dimensions. *J. Comput. Phys.*, 206:122–149, 2005.

[26] T. Kröger. *Multidimensional systems of hyperbolic conservation laws, numerical schemes, and characteristic theory : connections, differences, and numerical comparison.* PhD thesis, RWTH Aachen, 2004.

[27] M. Lukáčová-Medvid'ová, J. Saibertova, G. Warnecke and Y. Zahaykah. On evolution Galerkin Methods for the Maxwell and the Linearized Euler Equations. *Appl. Math.*, 49(5):415–439, 2004.

[28] M. Lukáčová-Medvid'ová, S. Noelle and M. Kraft. Well-balanced finite volume evolution Galerkin methods for the shallow water equations. *J. Comput. Phys.*, 221(1):122–147, 2007.

[29] M. Lukáčová-Medvid'ová, K.W. Morton and G.Warnecke. Evolution Galerkin methods for hyperbolic systems in two space dimensions. *Math. Comp.*, 69:1355-1384, 2000.

[30] M. Lukáčová-Medvid'ová, J. Saibertová and G.Warnecke. Finite volume evolution Galerkin methods for nonlinear hyperbolic systems. *J. Comput. Phys.*, 183(2):533562, 2002.

[31] M. Lukáčová-Medvid'ová, K.W. Morton and G.Warnecke. Finite volume evolution Galerkin methods for hyperbolic systems. *SIAM J. Sci. Comp.*, 26(1):1–30, 2004.

[32] F. Marche. Derivation of a new two-dimensional viscous shallow water model with varying topography, bottom friction and capillary effects. *Eur. J. Mech. B. Fluids*, 26(1):49–63, 2007.

[33] L.V. Ovsyannikov. Two-layer shallow water model. *J. Appl. Mech. Tech. Phys.*, 20(2):127–135, 1979.

[34] C. Parés. Numerical methods for nonconservative hyperbolic systems: a theoretical framework. *J. Math. Anal*, 44(1):300–321, 2006.

[35] C. Parés. Path-conservative numerical schemes for nonconservative hyperbolic systems. S. B. Benzoni-Gavage, D. Serre (eds.) *Hyperbolic Problems: Theory, Numerics, Applications*, pp. 817 - 824, Springer, 2008.

[36] C. Parés and M. Castro. On the well-balance property of Roe's method for nonconservative hyperbolic systems. Applications to shallow-water systems. *M2AN*, 38(5):821–852, 2004.

[37] J.P. Raymond. A new definition of nonconservative products and weak stability results. *Boll. Un. Mat. Ital. B*, 10:681–699, 1996.

[38] J.B. Schijf and J.C. Schonfeld. Theoretical considerations on the motion of salt and fresh water. In *Proceedings of the Minnesota International Hydraulics Convention*, Joint meeting International Association Hydraulic Research and Hydraulic Division American Society Civil Engineer, pp. 321–333, 1953.

[39] Phoolan Prasad. *Nonlinear Hyperbolic Waves in Multi-dimensions*, Chapman & Hall/CRC, 2001.

[40] Phoolan Prasad. A general version of the extended useful lemma, 2007, private communication.

[41] Gerolamo Cardano. *Ars magna or The rules of algebra. Transl. from the Latin and ed. by T. Richard Witmer. With a foreword by Oystein Ore. Reprint of the 1968 ed.*, New York, NY: Dover Publications. xxiv, 267 p. $ 8.95, 1993.