Available online at www.sciencedirect.com

ScienceDirect

Comput. Methods Appl. Mech. Engrg. 410 (2023) 116038

Computer methods in applied mechanics and engineering

www.elsevier.com/locate/cma

Development of POD-based Reduced Order Models applied to shallow water equations using augmented Riemann solvers

P. Solán-Fustero[a],[*], J.L. Gracia[b], A. Navas-Montilla[a], P. García-Navarro[a]

[a] I3A and Fluid Mechanics Department, University of Zaragoza, Spain
[b] IUMA and Applied Mathematics Department, University of Zaragoza, Spain

Received 22 December 2022; received in revised form 28 March 2023; accepted 29 March 2023
Available online 5 April 2023

Abstract

Reduced-order models (ROMs) based on the proper orthogonal decomposition have been proposed to reduce the computational resources required by the full-order models (FOMs) to approximate partial differential equations. In this paper a Roe-based ROM is developed to solve the shallow water equations in presence of source terms more efficiently than the Roe-based FOM. The well-balanced property and other numerical corrections such as the entropy fix and the wet–dry treatment are taken into account using augmented Riemann solvers to build the Roe-based FOM. In addition to this, a time averaging approach is necessary to develop the Roe-based ROM. This approach is validated by solving some cases and the computed solutions are compared with those ones of Lax–Friedrichs-based ROMs. It is also studied whether the ROM preserves or not the well-balancing, the entropy fix and the wet–dry treatment.

Keywords: Reduced-order modelling; POD methods; Lax–Friedrichs method; Roe method; Computational resources; Shallow water equations

1. Introduction

The shallow water equations (SWE) are widely used to model non-stationary free surface flows such as rivers, coastal flows, open channels and many other environmental flows where the dimensions in the vertical direction are negligible compared to the horizontal dimensions [1–3]. Unfortunately, SWE have no analytical solution and have to be solved by numerical methods. Shallow flow numerical modelling has been a very active field of research recently, supported by the increasing availability of laboratory and field data. Developments in numerical methods in recent years have led to a series of models applicable to a wide variety of geometries and flow types [4–7].

Among the options to solve 1D SWE numerically, a numerical scheme based on Godunov's method is considered here. Godunov-type numerical methods allow the numerical resolution of wave discontinuities in hyperbolic conservation systems by assuming a piece-wise constant distribution of the conserved variables within computational cells [8,9]. When trying to capture the dynamics of realistic shallow flows the inclusion of source terms in the simulation model that represent the effects of variable bed level and of the physical diffusion due to friction is mandatory. Several works indicate that the use of the numerical flux as defined for homogeneous equations is

[*] Corresponding author.
E-mail address: psolfus@unizar.es (P. Solán-Fustero).

https://doi.org/10.1016/j.cma.2023.116038

not adequate to solve situations involving source terms [10–13]. Of the many options that exist for constructing numerical schemes of the SWE, the upwind augmented Roe's method [11] has been chosen since it has been reported to produce robust and stable solutions in all kind of situations and is not as diffusive as other numerical methods such as the HLL in all its variants [12,14–16]. Furthermore, this first-order method has been chosen instead of other more accurate methods, such as methods based on WENO reconstructions [17,18], for the sake of simplicity.

In order to ensure stable, robust and accurate solutions in presence of bed variations and friction terms over dry beds, it is necessary to take into account numerical corrections, such as well-balancing, that keeps the discrete equilibrium with machine precision [4,5,19]. Augmented Riemann solvers are designed to preserve equilibrium in presence of source terms [20]. In addition to well-balancing, it is well known that other numerical corrections are necessary to fix some unphysical numerical solutions that may appear under certain circumstances, such as the entropy and the wet–dry front problems. The choice of Roe's method provides the possibility of carrying out these corrections and checking whether they are necessary or not in the ROM that is developed from it.

Furthermore, the numerical resolution of the SWE for realistic applications may involve large computational meshes and integration in time over long periods, hence requiring a high computational cost. It is therefore required to explore alternatives that help to speed up the calculations. Reduced-order models (ROMs) based on the proper orthogonal decomposition (POD) are a widely extended mathematical tool that offer saving computational costs without loss of accuracy. This is based on the fact that the variables of interest reside on a low-dimensional manifold within the infinite-dimensional solution space associated with the partial differential equation [21]. POD was originally introduced by Lumley [22] in 1967, and it has been given many names, such as Karhunen–Loève expansions [23,24] and principal component analysis [25,26]. It has been used in different frameworks and with many modifications [27–33]. The POD-based ROMs need to be trained using a set of high-fidelity solutions (snapshots) computed with numerical schemes, which are known as full-order models (FOMs). This training phase is called snapshot method [34]. After the training phase, in all numerical cases of this paper, the ROM is solved up to the same training time to obtain highly accurate solutions in shorter CPU times. The prediction of solutions beyond the training range is an issue of high complexity in advection-dominated problems, as shown in [35,36]. The authors of this paper worked on this aspect in an earlier paper [37] proposing a time extrapolation technique based on the method of the characteristics. Without the inclusion of this technique, ROMs are very inaccurate when solved outside the training range. Furthermore, this technique would require higher computational resources if applied to nonlinear problems with multiple shock and rarefaction waves.

In case of FOMs based on non-linear schemes such as Roe's method, it is necessary to look for alternatives to the standard Galerkin decomposition method [38]. In this direction, several techniques have been presented, such as the dynamic mode decomposition [39,40] o the discrete empirical interpolation method (DEIM) [41–43]. There are also multiple extensions of the standard POD method, such as the proper generalized decomposition (PGD), with recent works applied to frequency response analysis [44], space separation of transient wave propagation problem [45], 2D/3D flows and parametric Stokes problems [46] and the resolution of compressible/incompressible flows with different Reynolds numbers [47–50]; or such as the tensorial POD, that allow to reduce the online computational complexity of the reduced nonlinear terms for a SWEs model [51]. An interesting analysis of the POD-based ROM methodology in comparison with other non-intrusive methods applied to 2D SWE is presented in [39]. Other works on the combination of SWE and ROMs are focused on reducing the computational complexity by using DEIM with implicit numerical schemes [43], parametric sensitivity analysis [52,53] or meshless radial basis functions technique [47]. It is also worth mentioning interesting recent works investigating the combination of neural networks with POD methods applied to Navier–Stokes equations [54,55] or even deep-learning techniques [56] and Physics-informed neural networks [57]. However, in this work the time averaging approach in combination with the proper interval decomposition (PID, see [58]), which has been reported as a useful tool [28,33], is used. To validate the application of this approach to the Roe-based FOM used in this work, it has been first applied to a centred Lax–Friedrichs-based FOM without time averages.

In short, the main objective of this work is to develop a Roe-based ROM that allow to solve the SWE more efficiently than the FOM from which it is developed and with the minimum loss of accuracy. In addition, the interest is focused on studying whether it is necessary or not to take into account the numerical corrections aforementioned when designing the ROM and which properties of the FOM are preserved by the ROM or must be explicitly accounted for. It is also very interesting to study whether ROMs can be trained with solutions calculated with FOMs other than the one from which it was developed, i.e. the Roe-based ROM can be trained with the

Lax–Friedrichs-based FOM and vice versa. The study in the present work can be framed within the concept of ROM consistency as proposed by Ingimarson, Rebholz and Iliescu in a very recent work [59], in which they investigate theoretically and numerically how the performance of the ROM is affected by the discretization model used with respect to the FOM. Another very recent example dealing with the study of the concept of ROM consistency is that of García-Achilla, Novo and Rubino [60], in which the Navier–Stokes equations are considered. Finally, the sensitivity of the ROMs to specific parameters of the POD method is studied, such as the number of POD modes and the number of time steps per time window in terms of the differences between the solutions of the FOM and the ROM and the CPU times they require.

The remainder of the paper is organized as follows. Section 2 introduces the 1D SWE. Section 3 outlines the numerical methods used to solve them and Section 4, the POD-based ROM strategy and the different ROMs developed. A brief description of the corrections aforementioned is also contained in this section. Section 5 presents four numerical cases in which the ROMs developed and their characteristics are tested. Finally, concluding remarks are drawn in Section 6.

2. 1D shallow water equations

The study considered in this paper is focused on the resolution of the 1D SWE with source terms assuming a rectangular channel of unit width

$$\begin{aligned} &\frac{\partial h}{\partial t}+\frac{\partial q}{\partial x}=0, \\ &\frac{\partial q}{\partial t}+\frac{\partial}{\partial x}\left(\frac{q^2}{h}+\frac{1}{2}gh^2\right)=gh(S_0-S_f),\ \ (x,t)\in(0,L)\times(0,T], \end{aligned} \tag{1}$$

where $h(x,t)$ is the depth and $q(x,t)$ is the discharge, with $q(x,t)=h(x,t)u(x,t)$ and $u(x,t)$, the depth averaged velocity; the first source term is due to the gradient of the bed elevation $z(x)$

$$S_0=-\frac{\partial z}{\partial x}; \tag{2}$$

and the second source term is due to channel friction

$$S_f=\frac{n_b^2 u\,|u|}{h^{4/3}},$$

where n_b is the Manning coefficient. In addition, initial and boundary conditions for h and q have to be defined. The hyperbolic system of partial differential Eqs. (1) can be written in a conservative form

$$\frac{\partial \mathbf{U}}{\partial t}+\frac{\partial \mathbf{F}(\mathbf{U})}{\partial x}=\mathbf{S}, \tag{3}$$

where

$$\mathbf{U}=\begin{pmatrix} h\\ q\end{pmatrix},\quad \mathbf{F}(\mathbf{U})=\begin{pmatrix} q\\ \dfrac{q^2}{h}+\dfrac{1}{2}gh^2\end{pmatrix},\quad \mathbf{S}(\mathbf{U})=\begin{pmatrix} 0\\ gh(S_0-S_f)\end{pmatrix},$$

$\mathbf{U}$ is the vector of conserved variables, $\mathbf{F}(\mathbf{U})$ are the physical fluxes and $\mathbf{S}(\mathbf{U})$ are the source terms. System (3) can be written in quasi-linear form

$$\frac{\partial \mathbf{U}}{\partial t}+\mathbf{J}\frac{\partial \mathbf{U}}{\partial x}=\mathbf{S},$$

and the Jacobian matrix $\mathbf{J}(\mathbf{U})$ of the convective fluxes is given by

$$\mathbf{J}(\mathbf{U})=\frac{\partial \mathbf{F}(\mathbf{U})}{\partial \mathbf{U}}=\begin{pmatrix} 0 & 1\\ c^2-u^2 & 2u\end{pmatrix},$$

with $c=\sqrt{gh}$ the surface wave velocity. Assuming that this system of equations is strictly hyperbolic, the Jacobian matrix $\mathbf{J}(\mathbf{U})$ is diagonalizable with real eigenvalues. Its eigenvalues λ_1 and λ_2 and the corresponding eigenvectors $\mathbf{e}_1$ and $\mathbf{e}_2$ are

$$\begin{cases}\lambda_1=u-c,\\ \lambda_2=u+c,\end{cases}\quad \mathbf{e}_m=(1,\lambda_m)^T\,,\quad m=1,2.$$

3. Full-order model

The computational domain $[0, L]$ is discretized using the finite volume method (FV) by means of I_x volume cells of size $\Delta x = L/I_x$. The positions of the centre and left and right interfaces of ith cell are x_i, $x_{i-1/2}$ and $x_{i+1/2}$, respectively, for $i = 1, \dots, I_x$. The FOM is obtained by discretizing (1) with the Godunov-type scheme

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^n - \frac{\Delta t}{\Delta x}\left(\mathbf{F}_{i+1/2}^{n,-,*} - \mathbf{F}_{i-1/2}^{n,+,*}\right) - \Delta t \mathbf{S}_{f,i}^n, \tag{4}$$

where

$$\mathbf{U}_i^n = \begin{pmatrix} h_i^n \\ q_i^n \end{pmatrix} \quad \text{and} \quad \mathbf{S}_{f,i}^n = \begin{pmatrix} 0 \\ -g\dfrac{n_b^2 u_i^n \left|u_i^n\right|}{(h_i^n)^{1/3}} \end{pmatrix}.$$

Then, $h_i^n \approx h(x_i, t^n)$, $u_i^n \approx u(x_i, t^n)$ and $q_i^n \approx q(x_i, t^n)$ are the cell-averaged values of the water depth, velocity and discharge over the cell $\left(x_{i-1/2}, x_{i+1/2}\right)$. The numerical fluxes $\mathbf{F}_{i+1/2}^{n,-,*}$ and $\mathbf{F}_{i-1/2}^{n,+,*}$ are reconstructed following two different spatial discretizations: one based on the Roe numerical scheme and the other one based on the Lax–Friedrichs numerical scheme, which are explained in detail below.

Considering explicit schemes, the time step $\Delta t = t^{n+1} - t^n$ is selected dynamically using the Courant–Friedrichs–Lewy (CFL) condition [61]

$$\Delta t = CFL\frac{\Delta x}{\max\{\lambda_1, \lambda_2\}}, \tag{5}$$

where the CFL number satisfies $0 < CFL \leq 1$.

3.1. Roe-based FOM

The Roe-based FOM (Roe) is obtained by computing the numerical fluxes $\mathbf{F}_{i+1/2}^{n,-,*}$ and $\mathbf{F}_{i-1/2}^{n,+,*}$ as in [11,20]

$$\mathbf{F}_{i\pm1/2}^{n,\mp,*} = \mathbf{F}(\mathbf{U}_i^n) \pm \sum_{m=1}^{2}\left(\tilde{\lambda}_m^{\mp}\tilde{\gamma}_m\tilde{\mathbf{e}}_m\right)_{i\pm1/2}^n, \tag{6}$$

where

$$\left(\tilde{\lambda}_m^{\mp}\right)_{i+1/2}^n = \frac{1}{2}\left[\left(\tilde{\lambda}_m\right)_{i+1/2}^n \mp \left|\tilde{\lambda}_m\right|_{i+1/2}^n\right], \quad m = 1, 2.$$

The approximate eigenvalues $(\tilde{\lambda}_m)_{i+1/2}^n$ and eigenvectors $(\tilde{\mathbf{e}}_m)_{i+1/2}^n$ are

$$\begin{cases} \left(\tilde{\lambda}_1\right)_{i+1/2}^n = \tilde{u}_{i+1/2}^n - \tilde{c}_{i+1/2}^n, \\ \left(\tilde{\lambda}_2\right)_{i+1/2}^n = \tilde{u}_{i+1/2}^n + \tilde{c}_{i+1/2}^n, \end{cases} \quad (\tilde{\mathbf{e}}_m)_{i+1/2}^n = \left(1, (\tilde{\lambda}_m)_{i+1/2}^n\right)^T, \quad m = 1, 2,$$

with the following expressions for the Roe average velocity $\tilde{u}_{i+1/2}^n$ and celerity $\tilde{c}_{i+1/2}^n$

$$\tilde{u}_{i+1/2}^n = \frac{q_{i+1}^n\sqrt{h_{i+1}^n} + q_i^n\sqrt{h_i^n}}{\sqrt{h_{i+1}^n h_i^n}\left(\sqrt{h_{i+1}^n} + \sqrt{h_i^n}\right)}, \quad \tilde{c}_{i+1/2}^n = \sqrt{g\tilde{h}_{i+1/2}^n},$$

and $\tilde{h}_{i+1/2}^n = (h_{i+1}^n + h_i^n)/2$. The coefficients $(\tilde{\gamma}_m)_{i+1/2}^n$ in (6) are designed to preserve the well-balanced property

$$(\tilde{\gamma}_m)_{i+1/2}^n = \left(\tilde{\alpha}_m - \tilde{\beta}_m/\tilde{\lambda}_m\right)_{i+1/2}^n, \quad m = 1, 2,$$

with

$$\begin{cases} (\tilde{\alpha}_1)^n_{i+1/2} = \dfrac{\delta h^n_{i+1/2}\left(\tilde{\lambda}_2\right)^n_{i+1/2} - \delta q^n_{i+1/2}}{\left(\tilde{\lambda}_2\right)^n_{i+1/2} - \left(\tilde{\lambda}_1\right)^n_{i+1/2}}, \\ (\tilde{\alpha}_2)^n_{i+1/2} = \dfrac{\delta q^n_{i+1/2} - \delta h^n_{i+1/2}\left(\tilde{\lambda}_1\right)^n_{i+1/2}}{\left(\tilde{\lambda}_2\right)^n_{i+1/2} - \left(\tilde{\lambda}_1\right)^n_{i+1/2}}, \end{cases} \qquad \begin{cases} (\tilde{\beta}_1)^n_{i+1/2} = \dfrac{1}{2}\tilde{c}^n_{i+1/2}\delta z_{i+1/2}, \\ (\tilde{\beta}_2)^n_{i+1/2} = -(\tilde{\beta}_1)^n_{i+1/2}, \end{cases}$$

and $\delta z_{i+1/2} = z_{i+1} - z_i$.

3.2. Numerical corrections

The numerical method as presented in Eqs. (4)–(6) may lead to solutions that have no physical meaning in situations such as the advance of water front over a dry bed. To circumvent such issues, the numerical fixes indicated in [10, Section 2.3] and [13, Section4.2.2] are applied in this work.

Source term correction to ensure positive water depths

As proposed in [13], two intermediate states are defined in the augmented Roe solver

$$(h^*)^n_{i+1/2} = h^n_i + (\tilde{\alpha}_1)^n_{i+1/2} - \left(\frac{\tilde{\beta}_1}{\tilde{\lambda}_1}\right)^n_{i+1/2},$$
$$(h^{**})^n_{i+1/2} = h^n_i - (\tilde{\alpha}_2)^n_{i+1/2} + \left(\frac{\tilde{\beta}_2}{\tilde{\lambda}_2}\right)^n_{i+1/2}.$$

The intermediate states have to satisfy that $(h^*)^n_{i+1/2}$, $(h^{**})^n_{i+1/2} \geq 0$; otherwise the coefficients $(\tilde{\beta}_m)^n_{i+1/2}$ are redefined as follows: If $(h^*)^n_{i+1/2} < 0$, then they are given by

$$(\tilde{\beta}_1)^n_{i+1/2} = (h^n_i + (\tilde{\alpha}_1)^n_{i+1/2})(\tilde{\lambda}_1)^n_{i+1/2}, \quad (\tilde{\beta}_2)^n_{i+1/2} = -(\tilde{\beta}_1)^n_{i+1/2};$$

and if $(h^{**})^n_{i+1/2} < 0$, then

$$(\tilde{\beta}_2)^n_{i+1/2} = (h^n_i - (\tilde{\alpha}_2)^n_{i+1/2})(\tilde{\lambda}_2)^n_{i+1/2}, \quad (\tilde{\beta}_1)^n_{i+1/2} = -(\tilde{\beta}_2)^n_{i+1/2}.$$

Wet–dry treatment

Taking into account the above intermediate states, the wet–dry treatment proposes the following algorithm

- If $h^n_{i+1} = 0$ and $(h^{**})^n_{i+1/2} < 0$, then

$$\mathbf{F}^{n,-,*}_{i+1/2} = \mathbf{F}^{n,-,*}_{i+1/2} + \mathbf{F}^{n,+,*}_{i+1/2} \quad \text{and} \quad \mathbf{F}^{n,+,*}_{i+1/2} = 0.$$

- If $h^n_i = 0$ and $(h^*)^n_{i+1/2} < 0$, then

$$\mathbf{F}^{n,+,*}_{i+1/2} = \mathbf{F}^{n,-,*}_{i+1/2} + \mathbf{F}^{n,+,*}_{i+1/2} \quad \text{and} \quad \mathbf{F}^{n,-,*}_{i+1/2} = 0.$$

Entropy fix

Non-physical values can also be generated with the Roe scheme in transcritical accelerations. To avoid this, the Harten–Hyman entropy fix [62] can be used. Following [13], the eigenvalues are decomposed $\left(\tilde{\lambda}_k\right)^n_{i+1/2} =$

$\left(\tilde{\lambda}_k^-\right)^n_{i+1/2} + \left(\tilde{\lambda}_k^+\right)^n_{i+1/2}$ where

$$\left(\tilde{\lambda}_k^-\right)^n_{i+1/2} = (\lambda_k)^n_i \, \frac{(\lambda_k)^n_{i+1} - \left(\tilde{\lambda}_k\right)^n_{i+1/2}}{\lambda^k_{i+1} - \lambda^k_i},$$

$$\left(\tilde{\lambda}_k^+\right)^n_{i+1/2} = (\lambda_k)^n_{i+1} \, \frac{\left(\tilde{\lambda}_k\right)^n_{i+1/2} - (\lambda_k)^n_i}{(\lambda_k)^n_{i+1} - (\lambda_k)^n_i}, \quad k = 1, 2, \; i = 1, \dots, I_x.$$

with $(\lambda_1)^n_i = u^n_i - c^n_i$ and $(\lambda_2)^n_i = u^n_i + c^n_i$. By construction, $\left(\tilde{\lambda}_k^-\right)^n_{i+1/2} < 0$ and $\left(\tilde{\lambda}_k^+\right)^n_{i+1/2} > 0$. This idea is applied also to the decomposition of the source terms $\tilde{\beta}^k_{i+1/2}$, as indicated in [13].

3.3. Lax–friedrichs-based FOM

The Lax–Friedrichs' method [63] is used in the present work to validate the development of the Roe-based ROM. This FOM is based on the numerical fluxes (4):

$$\mathbf{F}^{n,\mp,*}_{i+1/2} = \frac{1}{2}\left[\mathbf{F}\left(\mathbf{U}^n_i\right) + \mathbf{F}\left(\mathbf{U}^n_{i+1}\right) \mp \mathbf{S}^n_{0,i+1/2}\right] - \frac{1}{2}\nu\frac{\Delta x}{\Delta t}\left(\mathbf{U}^n_{i+1} - \mathbf{U}^n_i\right), \tag{7}$$

where

$$\mathbf{F}(\mathbf{U}^n_i) = \begin{pmatrix} q^n_i \\ (u^n_i)^2 h^n_i + \frac{1}{2} g (h^n_i)^2 \end{pmatrix}, \quad \mathbf{S}^n_{0,i+1/2} = \begin{pmatrix} 0 \\ -g\tilde{h}^n_{i+1/2}\delta z_{i+1/2} \end{pmatrix},$$

with $CFL \leq \nu \leq 1$. In the numerical experiments will take $\nu = 0.9$.

4. Reduced-order model strategy

The POD-based ROM strategy consists of two parts: (I) the off-line part, in which the ROM is trained using solutions computed with the FOM (4) up the training time; and (II) the on-line part, in which the ROM is solved up to the same training time.

The off-line part starts with the snapshot method [34], according to which a set of N_T numerical solutions computed with the FOM, also called snapshots, is used to construct the snapshot matrices, one for each variable

$$M_h = \begin{pmatrix} h^1_1 & \dots & h^{N_T}_1 \\ \vdots & \vdots & \vdots \\ h^1_{I_x} & \dots & h^{N_T}_{I_x} \end{pmatrix}, \quad M_u = \begin{pmatrix} u^1_1 & \dots & u^{N_T}_1 \\ \vdots & \vdots & \vdots \\ u^1_{I_x} & \dots & u^{N_T}_{I_x} \end{pmatrix}, \quad M_q = \begin{pmatrix} q^1_1 & \dots & h^{N_T}_1 \\ \vdots & \vdots & \vdots \\ q^1_{I_x} & \dots & q^{N_T}_{I_x} \end{pmatrix}.$$

In order to avoid loss of accuracy, the snapshot matrix is built using all the time levels computed with the FOM and the ROM is solved at the same time levels than the FOM.

The Galerkin decomposition method [38] is applied to each of the variables

$$h^n_i \approx \sum_{k=1}^{M_{\mathrm{POD}}} \hat{h}^n_k \phi_{i,k}, \quad u^n_i \approx \sum_{k=1}^{M_{\mathrm{POD}}} \hat{u}^n_k \xi_{i,k}, \quad q^n_i \approx \sum_{k=1}^{M_{\mathrm{POD}}} \hat{q}^n_k \eta_{i,k}, \tag{8}$$

where $\phi_{i,k}$, $\xi_{i,k}$ and $\eta_{i,k}$ are the functions of the basis of each variable, being M_{POD} a positive integer, also known as the number of POD modes, such that $M_{\mathrm{POD}} \ll \min\{N_T, I_x\}$. In other works, [28,33], the optimal value of the POD modes is obtained by means of a minimization problem taking into account the decay rate of the POD energy represented by these modes. However, in this work the interest is focused on analysing the performance of the different ROMs as a function of the number of POD modes they solve. As a final result, the optimal value is chosen attending to the accuracy of the solution computed as an *a posteriori* criterion.

To assess the accuracy of the solutions computed with the ROM, they have been compared with those of the corresponding FOM, by computing the L_1 norm of the water depth and water discharge at the final time T. Recall that, given a mesh function $z = (z_1, \dots, z_{I_x})$ with $\Delta x = L/I_x$ the diameter of the cells, the L_1 norm of u is given by:

$$\|z\|_1 = \Delta x \sum_{i=1}^{I_x} |z_i|,$$

and the differences in the L_1 norm for the depth and water discharge are denoted by $\|d_h\|_1$ and $\|d_q\|_1$, respectively.

The POD basis of functions is computed by applying the singular value decomposition to each snapshot matrix, as in the following case on the snapshot matrix of the water depth

$$M_h M_h^T = \Phi \Sigma \Sigma^T \Phi^T,$$

where $\Sigma \in \mathbb{R}^{I_x \times N_T}$ is a diagonal matrix whose entries are the singular values of M_h and matrix $\Phi = (\phi_1, \dots, \phi_{I_x}) \in \mathbb{R}^{I_x \times I_x}$ with $\phi_k = (\phi_{1,k}, \dots, \phi_{I_x,k})^T$ consists of the orthogonal eigenvectors of $M_h M_h^T$. The procedure for obtaining the basis of the snapshot matrices for water velocity and discharge is similar to that for water depth.

To complete the off-line part, these POD bases of h, u and q are used to train the ROM by building its training matrices. The following is the procedure for developing the ROM:

(i) introduce the Galerkin decomposition method (8) into the FOM, i.e. each time one of the variables h, u or q appears, it is replaced by its corresponding reduced variable $\hat{h}$, $\hat{u}$ or $\hat{q}$;
(ii) multiply each resulting equation by $\phi_{i,p}$ and $\eta_{i,p}$, respectively;
(iii) sum up over the cells to complete the projection of both equations into the reduced space.

This procedure requires some modifications in order to be successfully applied to the Roe method. These modifications are explained below.

4.1. Time windows and time averaging approach

The Galerkin decomposition method cannot be applied directly to some FOMs. This is, for example, the case of the SWE with the Roe method (6) which includes variables in square roots or in denominators of quotients of functions (see Section 3.1).

In addition to this, when solving the ROM, the conserved variables (water depth h and discharge q) are updated at each time step in the reduced space, but it is also necessary at each time step to calculate the variables in the physical space in order to update the water velocity u. This implies an extra cost of computational resources. This problem can be reduced by applying the Galerkin decomposition method only to the conserved variables h and q and excluding u.

For the above two reasons, to fully develop the Roe-based ROM, the time averaging approach introduced in [33] is applied as follows

$$\bar{h}_i^w = \frac{1}{N_s} \sum_{n=w_1}^{w_{N_s}} h_i^n, \quad \bar{u}_i^w = \frac{1}{N_s} \sum_{n=w_1}^{w_{N_s}} u_i^n, \; i = 1, \dots, I_x,$$

where N_s is the number of snapshots per time window and w_1 and w_{N_s} are the initial and final components of the wth time window, so that the time averaging approach is directly related to the PID method [27,30,33]. According to this method, the total simulation time T is divided into homogeneous N_w non-overlapping time windows

$$\left[0, t^{N_1}\right] \cup \left[t^{N_1+1}, t^{N_2}\right] \cup \dots \cup \left[t^{N_{w-1}+1}, t^{N_w}\right],$$

where $0 < t^{N_1} < t^{N_2} < \dots < t^{N_w} = t^{N_T} = T$. The snapshot matrices can also be divided into N_w time windows following the PID to improve the accuracy of the solutions computed with the ROM, so that a different snapshot matrix is built for each time window with its corresponding POD basis.

The time averaging approach is applied only to the water depth h and the water velocity u, because the ROM matrices depend on λ_1, λ_2 and c, which depend only on these variables, but not on the water discharge q.

In short, the Roe-based ROM is developed using time averages and validated by comparing the solutions of Lax–Friedrichs-based ROMs with and without time averages. These ROMs are shown below.

4.2. Roe-based ROM

First, a time-averaged ROM based on the Roe method (TRRoe) is considered

$$\begin{aligned}\hat{\mathbf{h}}^{n+1} &= \hat{\mathbf{h}}^{n} + \frac{1}{4}\frac{\Delta t}{\Delta x}A\hat{\mathbf{h}}^{n} + \frac{1}{4}\frac{\Delta t}{\Delta x}B\hat{\mathbf{q}}^{n} - \frac{1}{4}\frac{\Delta t}{\Delta x}\mathbf{C},\\ \hat{\mathbf{q}}^{n+1} &= \hat{\mathbf{q}}^{n} - \frac{1}{4}\frac{\Delta t}{\Delta x}D\hat{\mathbf{h}}^{n} + \frac{1}{4}\frac{\Delta t}{\Delta x}E\hat{\mathbf{q}}^{n} + \frac{1}{4}\frac{\Delta t}{\Delta x}\mathbf{F} - \Delta t g\mathbf{G},\end{aligned} \tag{9}$$

where the elements of the training matrices $A = \left(A_{kp}\right)$, $B = \left(B_{kp}\right)$, $C = \left(C_p\right)$, $D = \left(D_{kp}\right)$, $E = \left(E_{kp}\right)$, $F = \left(F_p\right)$ and $G = \left(G_p\right)$ are

$$A_{kp} = a_1\phi_{1,p} + \sum_{i=2}^{I_x-1}\left[\left(\phi_{i+1,k}-\phi_{i,k}\right)\bar{a}^{w}_{i+1/2} - \left(\phi_{i,k}-\phi_{i-1,k}\right)\bar{a}^{w}_{i-1/2}\right]\phi_{i,p} + a_{I_x}\phi_{I_x,p},$$

$$B_{kp} = b_1\phi_{1,p} + \sum_{i=2}^{I_x-1}\left[\left(\eta_{i+1,k}-\eta_{i,k}\right)\bar{b}^{w}_{i+1/2} + \left(\eta_{i,k}-\eta_{i-1,k}\right)\bar{b}^{w}_{i-1/2}\right]\phi_{i,p} + b_{I_x}\phi_{I_x,p},$$

$$\begin{aligned}C_p &= c_1\phi_{1,1} + \sum_{i=2}^{I_x-1}\left[\delta z_{i+1/2}\bar{c}^{w}_{i+1/2} - \delta z_{i-1/2}\bar{c}^{w}_{i-1/2}\right]\phi_{i,p} + c_{I_x}\phi_{I_x,p},\\ D_{kp} &= d_1\eta_{1,p} + \sum_{i=2}^{I_x-1}\left[\left(\phi_{i+1,k}-\phi_{i,k}\right)\bar{d}^{w}_{i+1/2} + \left(\phi_{i,k}-\phi_{i-1,k}\right)\bar{d}^{w}_{i-1/2}\right]\eta_{i,p} + d_{I_x}\eta_{k,I_x},\\ E_{kp} &= e_1\eta_{1,1} + \sum_{i=2}^{I_x-1}\left[\left(\eta_{i+1,k}-\eta_{i,k}\right)\bar{e}^{w}_{i+1/2} + \left(\eta_{i,k}-\eta_{i-1,k}\right)\bar{e}^{w}_{i-1/2}\right]\eta_{i,p} + e_{I_x}\eta_{I_x,p},\\ F_p &= f_1\eta_{1,p} + \sum_{i=2}^{I_x-1}\left[\delta z_{i+1/2}\bar{f}^{w}_{i+1/2} + \delta z_{i-1/2}\bar{f}^{w}_{i-1/2}\right]\eta_{i,p} + f_{I_x}\eta_{I_x,p},\\ G_p &= \sum_{i=1}^{I_x}\frac{n_b^2 u^w_i\left|u^w_i\right|}{\left(h^w_i\right)^{1/3}}\eta_{i,p},\end{aligned} \tag{10}$$

with

$$\bar{a}^{w}_{i+1/2} = \left[\frac{\left|\tilde{\lambda}_1\right|\tilde{\lambda}_2 - \left|\tilde{\lambda}_2\right|\tilde{\lambda}_1}{\tilde{c}}\right]^{w}_{i+1/2}, \quad \bar{b}^{w}_{i\pm1/2} = \left[\frac{\left(\tilde{\lambda}_1 \mp \left|\tilde{\lambda}_1\right|\right) - \left(\tilde{\lambda}_2 \mp \left|\tilde{\lambda}_2\right|\right)}{\tilde{c}}\right]^{w}_{i\pm1/2},$$

$$\bar{c}^{w}_{i+1/2} = \left[\tilde{c}\left(\frac{\left|\tilde{\lambda}_1\right|}{\tilde{\lambda}_1} - \frac{\left|\tilde{\lambda}_2\right|}{\tilde{\lambda}_2}\right)\right]^{w}_{i+1/2},$$

$$\bar{d}^{w}_{i\pm1/2} = \left[\frac{\tilde{\lambda}_1\tilde{\lambda}_2}{\tilde{c}}\right]^{w}_{i\pm1/2}\left[\left(\tilde{\lambda}_1 \mp \left|\tilde{\lambda}_1\right|\right) - \left(\tilde{\lambda}_2 \mp \left|\tilde{\lambda}_2\right|\right)\right]^{w}_{i\pm1/2},$$

$$\bar{e}^{w}_{i\pm1/2} = \left[\frac{\left(\tilde{\lambda}_1 \mp \left|\tilde{\lambda}_1\right|\right)\tilde{\lambda}_1 - \left(\tilde{\lambda}_2 \mp \left|\tilde{\lambda}_2\right|\right)\tilde{\lambda}_2}{\tilde{c}}\right]^{w}_{i\pm1/2},$$

$$\bar{f}^{w}_{i\pm1/2} = \tilde{c}^{w}_{i\pm1/2}\left[\left(\tilde{\lambda}_1 \mp \left|\tilde{\lambda}_1\right|\right) - \left(\tilde{\lambda}_2 \mp \left|\tilde{\lambda}_2\right|\right)\right]^{w}_{i\pm1/2},$$

and the boundary terms $a_\ell, b_\ell, \ldots, f_\ell$, $\ell = 1, I_x$ depend on the type of the boundary conditions, as described in [37]. The following versions of ROM also calculate the boundary condition terms in the same way.

4.3. Lax–Friedrichs-based ROM

The Lax–Friedrichs-based ROM (RLF) is developed by applying the Galerkin decomposition to h, u and q and without using time averages for any of the variables. Its vector formulation is

$$\begin{aligned} \hat{\mathbf{h}}^{n+1} &= \hat{\mathbf{h}}^{n} - \frac{1}{2}\frac{\Delta t}{\Delta x} A \hat{\mathbf{q}}^{n} + \frac{1}{2}\nu B \hat{\mathbf{h}}^{n} + \frac{1}{2}\nu C, \\ \hat{\mathbf{q}}^{n+1} &= \hat{\mathbf{q}}^{n} - \frac{1}{2}\frac{\Delta t}{\Delta x} \left(\hat{\mathbf{q}}^{n}\right)^{T} D \hat{\mathbf{u}}^{n} - \frac{1}{4} g \frac{\Delta t}{\Delta x} \left(\hat{\mathbf{h}}^{n}\right)^{T} E \hat{\mathbf{h}}^{n} + \frac{1}{2}\nu F \hat{\mathbf{q}}^{n} - \frac{1}{4}\frac{\Delta t}{\Delta x} g G \hat{\mathbf{h}}^{n}, \end{aligned} \tag{11}$$

where the entries of the training matrices $A = \left(A_{kp}\right)$, $B = \left(B_{kp}\right)$, $C = \left(C_{p}\right)$, $D = \left(D_{qkp}\right)$, $E = \left(E_{qkp}\right)$, $F = \left(F_{kp}\right)$ and $G = \left(G_{kp}\right)$ are

$$\begin{aligned} A_{kp} &= a_1 \phi_{1,p} + \sum_{i=2}^{I_x - 1} \left(\eta_{i+1,k} - \eta_{i-1,k}\right) \phi_{i,p} + a_{I_x} \phi_{I_x,p}, \\ B_{kp} &= b_1 \phi_{1,p} + \sum_{i=2}^{I_x - 1} \left(\phi_{i+1,k} - 2\phi_{i,k} + \phi_{i-1,k}\right) \phi_{i,p} + b_{I_x} \phi_{I_x,p}, \\ C_{p} &= c_1 \phi_{1,p} + \sum_{i=2}^{I_x - 1} \left(z_{i+1} - 2z_{i} + z_{i-1}\right) \phi_{i,p} + c_{I_x} \phi_{I_x,p}, \\ D_{qkp} &= d_1 \eta_{1,p} + \sum_{i=2}^{I_x - 1} \left(\eta_{i+1,k}\xi_{q,i+1} - \eta_{i-1,k}\xi_{q,i-1}\right) \eta_{i,p} + d_{I_x} \eta_{I_x,p}, \\ E_{qkp} &= e_1 \eta_{1,p} + \sum_{i=2}^{I_x - 1} \left(\phi_{i+1,k}\phi_{q,i+1} - \phi_{i-1,k}\phi_{q,i-1}\right) \eta_{i,p} + e_{q,I_x} \eta_{I_x,p}, \\ F_{kp} &= f_1 \eta_{1,p} + \sum_{i=2}^{I_x - 1} \left(\eta_{i+1,k} - 2\eta_{i,k} + \eta_{i-1,k}\right) \eta_{i,p} + f_{I_x} \eta_{I_x,p}, \\ G_{kp} &= g_1 \eta_{1,p} + \sum_{i=2}^{I_x - 1} \left((\phi_{i+1,k} + \phi_{i,k})\delta z_{i+1/2} + (\phi_{i,k} + \phi_{i-1,k})\delta z_{i-1/2}\right) \eta_{i,p} + g_{I_x} \eta_{I_x,p}. \end{aligned} \tag{12}$$

4.4. Lax–Friedrichs-based ROM with time averages

The time-averaged Lax–Friedrichs-based ROM with time windows (TRLF) is developed by applying the Galerkin decomposition only to h and q and using time averaging for u (and h in the friction source term). Its vector formulation is

$$\begin{aligned} \hat{\mathbf{h}}^{n+1} &= \hat{\mathbf{h}}^{n} - \frac{1}{2}\frac{\Delta t}{\Delta x} A \hat{\mathbf{q}}^{n} + \frac{1}{2}\nu B \hat{\mathbf{h}}^{n} + \frac{1}{2}\nu C, \\ \hat{\mathbf{q}}^{n+1} &= \hat{\mathbf{q}}^{n} - \frac{1}{2}\frac{\Delta t}{\Delta x} D \hat{\mathbf{h}}^{n} - \frac{1}{4} g \frac{\Delta t}{\Delta x} \left(\hat{\mathbf{h}}^{n}\right)^{T} E \hat{\mathbf{h}}^{n} + \frac{1}{2}\nu F \hat{\mathbf{q}}^{n} - \frac{1}{4}\frac{\Delta t}{\Delta x} g G \hat{\mathbf{h}}^{n} - \Delta t g H, \end{aligned} \tag{13}$$

where the training matrices A, B, C, E, F and G are the same as in (12) and the entries of the new matrices $D = \left(D_{kp}\right)$ and $H = \left(H_{p}\right)$ are

$$\begin{aligned} D_{kp} &= d_{k,1} \eta_{1,p} + \sum_{i=2}^{I_x - 1} \left(\phi_{i+1,k}(u_{i+1}^{w})^2 - \phi_{i-1,k}(u_{i-1}^{w})^2\right) \eta_{i,p} + d_{I_x} \eta_{I_x,p}, \\ H_{p} &= \sum_{i=1}^{I_x} \frac{n_b^2 u_i^w \left|u_i^w\right|}{\left(h_i^w\right)^{1/3}} \eta_{i,p}. \end{aligned} \tag{14}$$

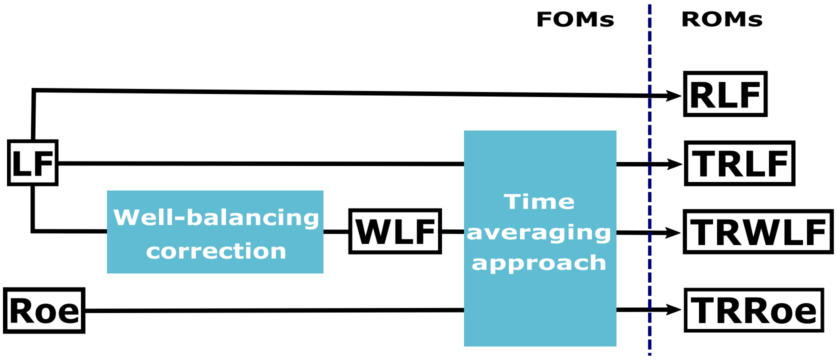


Fig. 1. Numerical schemes used.

Fig. 1 shows a chart where the numerical methods used in this work are presented, including both the FOMs and ROMs. It shows methods that are developed without any correction, such as the Lax–Friedrichs-based ROM (RLF) from the Lax–Friedrichs-based FOM (LF), but also the methods that include the time averaging approach, such as the time-averaged Lax–Friedrichs-based ROM (TRLF) and the time-averaged Roe-based ROM (TRRoe). The time-averaged well-balanced Lax–Friedrichs-based ROM (TRWLF) and its associated FOM, the well-balanced Lax–Friedrichs-based FOM (WLF), will be introduced in the next section.

5. Numerical results

Case 0. Validation of time averaging approach in a flat and frictionless case

The essential feature of an instantaneous dam-break is a large discontinuity that occurs when the water accumulated at a higher height is released and flows over the lower channel. When flow occurs in a channel, it can be assimilated to a one-dimensional character. In this work, the dam-break problem is modelled using the 1D SWE.

A dam-break over a flat and frictionless bed is solved in the domain $(x, t) \in [0, 12] \times [0, 0.99]$ with the following initial conditions (ICs)

$$h(x,0) = \begin{cases} 2, & \text{if } 0 \le x \le 6, \\ 1, & \text{if } 6 < x \le 12, \end{cases} \qquad q(x,0) = 0, \quad 0 \le x \le 12,$$

and free boundary conditions (BCs) are imposed.

Fig. 2 shows the computed approximations to h (left) and q (right) at the final time $T = 0.99$ using the LF, RLF, TRLF and TRRoe methods with $I_x = 203$, $N_w = 24$ and $M_{\text{POD}} = 5$. These numerical results validate the use of the time averaging approach, since the TRLF and TRRoe achieve results as good as the other two schemes.

5.1. Case 1. Validation of time averaging approach in presence of source terms

This case is set up to validate the ROMs developed with the time averaging approach applied to the 1D SWE with source terms and to study their solutions in terms of the differences between the FOM and the ROM and their efficiency. For this purpose, the problem is again a dam-break in the domain $[0, 12] \times [0, 1.02]$ with a mild slope, without friction and with ICs

$$h(x,0) = \begin{cases} 2 - z(x), & \text{if } 0 \le x \le 6, \\ 1 - z(x), & \text{if } 6 < x \le 12, \end{cases} \qquad q(x,0) = 0, \quad 0 \le x \le 12,$$

with $z(x) = 0.2(1 - x/12)$ and free BCs. The spatial domain is divided into $I_x = 100$, 201, 403 and 809 volume cells. The time step is computed for each case to satisfy the CFL condition (5) with $CFL = 0.9$.

In addition to the validation of the time averaging approach, the sensitivity of the ROM solutions to parameters specific to the method is studied in this case. On the one hand, the number of time windows N_w used is shown in Table 1, where N_w is arranged according to the number of snapshots per window N_s and per mesh refinement I_x.

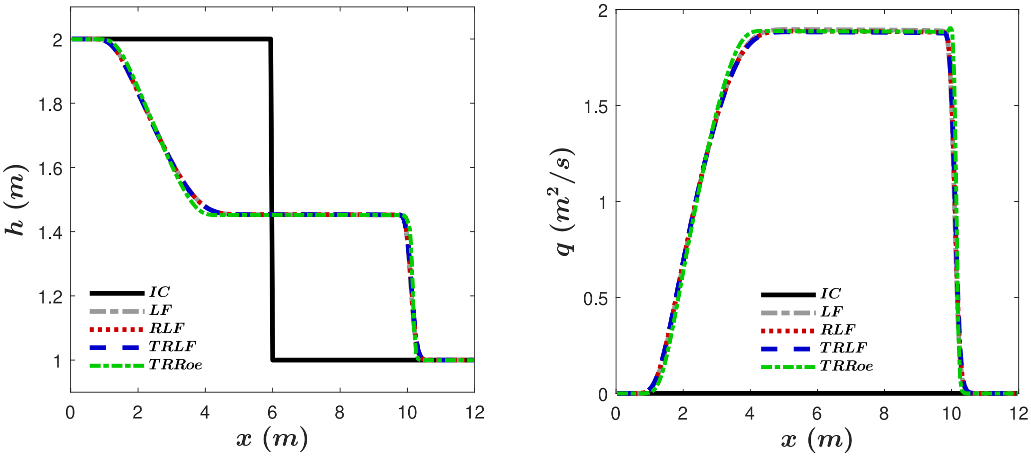


Fig. 2. Case 0: Approximations to h (left) and q (right) with LF, RLF, TRLF and TRRoe methods and taking $I_x = 203$, $N_w = 24$ and $M_{\text{POD}} = 5$.

Table 1
Number of time windows, N_w, for different values of N_t and N_s.

I_x	N_t	N_s			
		48	12	4	2
100	48	1	4	12	24
201	96	2	8	24	48
403	192	4	16	48	96
809	384	8	32	96	192

Table 2
Case 1: $\|d_h\|_1$ for different values of M_{POD} and N_w with $I_x = 201$.

	M_{POD}	N_w			
		2	8	24	48
RLF	1	$9.76 \cdot 10^{-1}$	$2.65 \cdot 10^{-1}$	$7.35 \cdot 10^{-2}$	$1.36 \cdot 10^{-2}$
	5	$1.15 \cdot 10^{-1}$	$7.86 \cdot 10^{-3}$	$1.30 \cdot 10^{-2}$	$2.04 \cdot 10^{-2}$
	10	$3.60 \cdot 10^{-2}$	$3.10 \cdot 10^{-3}$	$1.30 \cdot 10^{-2}$	$1.85 \cdot 10^{-2}$
TRLF	1	$9.78 \cdot 10^{-1}$	$2.67 \cdot 10^{-1}$	$7.27 \cdot 10^{-2}$	$1.35 \cdot 10^{-2}$
	5	$1.75 \cdot 10^{-1}$	$3.42 \cdot 10^{-2}$	$1.57 \cdot 10^{-2}$	$2.63 \cdot 10^{-2}$
	10	$2.06 \cdot 10^{-1}$	$3.89 \cdot 10^{-2}$	$1.56 \cdot 10^{-2}$	$2.32 \cdot 10^{-2}$
TRRoe	1	$9.93 \cdot 10^{-1}$	$2.80 \cdot 10^{-1}$	$9.14 \cdot 10^{-2}$	$2.74 \cdot 10^{-2}$
	5	$1.21 \cdot 10^{-1}$	$2.26 \cdot 10^{-2}$	$3.27 \cdot 10^{-2}$	$5.09 \cdot 10^{-2}$
	10	$7.39 \cdot 10^{-2}$	$2.53 \cdot 10^{-3}$	$3.26 \cdot 10^{-2}$	$5.03 \cdot 10^{-2}$

Note that $N_w = N_t/N_s$. On the other hand, this problem has been solved with different number of POD modes, namely $M_{\text{POD}} = 1$, 5 and 10.

Tables 2 and 3 show the differences of the computed h and q with each version of the ROM and $I_x = 201$ for different values of M_{POD} and N_w. In general, we observe that similar computed approximations are obtained with the FOMs and ROMs in all the cases.

On the one hand, all three versions of the ROM show similar decreasing tendencies as M_{POD} grows in the differences of h and q, i.e., the water depth and discharge solutions are more accurate for $M_{\text{POD}} = 5$ and 10 than for $M_{\text{POD}} = 1$. The differences calculated with $M_{\text{POD}} = 5$ and $M_{\text{POD}} = 10$ become more similar as the number of time windows increases.

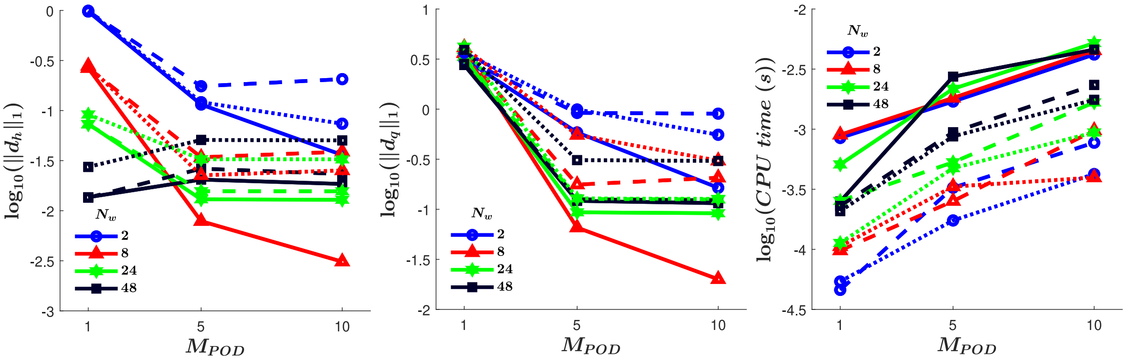
On the other hand, the differences are not monotonically decreasing with the number of time windows, which is due to the required change of basis that occurs when switching from one time window to the next. In each of

Table 3
Case 1: $\|d_q\|_1$ for different values of M_{POD} and N_w with $I_x = 201$.

	M_{POD}	N_w			
		2	8	24	48
RLF	1	3.16	3.66	3.31	2.76
	5	$5.85 \cdot 10^{-1}$	$6.56 \cdot 10^{-2}$	$9.32 \cdot 10^{-2}$	$1.21 \cdot 10^{-1}$
	10	$1.64 \cdot 10^{-1}$	$2.00 \cdot 10^{-2}$	$9.14 \cdot 10^{-2}$	$1.15 \cdot 10^{-1}$
TRLF	1	3.16	3.61	3.36	2.84
	5	$9.18 \cdot 10^{-1}$	$1.76 \cdot 10^{-1}$	$1.27 \cdot 10^{-1}$	$1.26 \cdot 10^{-1}$
	10	$9.01 \cdot 10^{-1}$	$2.07 \cdot 10^{-1}$	$1.25 \cdot 10^{-1}$	$1.23 \cdot 10^{-1}$
TRRoe	1	3.66	4.13	4.20	3.91
	5	$9.92 \cdot 10^{-1}$	$5.51 \cdot 10^{-1}$	$1.84 \cdot 10^{-1}$	$3.10 \cdot 10^{-1}$
	1	$5.55 \cdot 10^{-1}$	$3.06 \cdot 10^{-1}$	$1.27 \cdot 10^{-1}$	$3.04 \cdot 10^{-1}$

Table 4
Case 1: CPU times (s) of the FOMs with $I_x = 201$.

LF	Roe
$1.93 \cdot 10^{-2}$	$1.52 \cdot 10^{-2}$



(a) $\|d_h\|_1$.

(b) $\|d_q\|_1$.

(c) CPU times (s).

Fig. 3. Case 1: $\|d_h\|_1$, $\|d_q\|_1$ and CPU times (s) for RLF (solid), TRLF (dashed) and TRRoe (dotted) with $N_w = 2, 8, 24, 48$ and $I_x = 201$ for $M_{\text{POD}} = 1$, 5 and 10.

these changes, the approximation based on the number of POD modes can imply a small loss of accuracy, so, the bigger the number of time windows, the larger this loss of accuracy can be. To sum up, it can be concluded that the time averaging approach in combination with an appropriate refinement of time windows can be used in the construction of the Roe-based ROM without loss in the accuracy of the solutions.

Figs. 3(a) and 3(b) are included to highlight that, for high values of the number of time windows ($N_w = 24$ and 48), the differences of the solutions is of the same order of magnitude with 5 and 10 POD modes. This stagnation of the differences as the number of POD modes increases is due to the change of basis required at each time window. This is relevant when looking for the optimal set of values of M_{POD} and N_w in terms of efficiency.

Tables 4 and 5 show the CPU times required by each version of the FOM and the ROM to obtain the solution at the final time $T = 1.02$. As expected, the ROMs are faster than the FOMs and, in general, the TRLF and the TRRoe have better CPU times than the RLF. This is because the Galerkin decomposition is applied only to the water depth and discharge, but not to the water velocity. As expected, solving with 10 POD modes is more expensive in terms of CPU time than with 5 POD modes, as shown in Fig. 3(c). The CPU times required by the RLF are the most expensive, regardless of the number of time windows used, as can be seen from the clustering of the solid lines in Fig. 3(c).

Table 5
Case 1: CPU times (s) of the ROMs for different values of M_{POD} and N_w with $I_x = 201$.

	M_{POD}	N_w			
		2	8	24	48
RLF	1	$8.46 \cdot 10^{-4}$	$8.99 \cdot 10^{-4}$	$5.09 \cdot 10^{-4}$	$2.50 \cdot 10^{-4}$
	5	$1.70 \cdot 10^{-3}$	$1.82 \cdot 10^{-3}$	$2.17 \cdot 10^{-3}$	$2.74 \cdot 10^{-3}$
	10	$4.18 \cdot 10^{-3}$	$4.52 \cdot 10^{-3}$	$5.20 \cdot 10^{-3}$	$4.60 \cdot 10^{-3}$
TRLF	1	$4.60 \cdot 10^{-5}$	$9.80 \cdot 10^{-5}$	$2.53 \cdot 10^{-4}$	$2.30 \cdot 10^{-4}$
	5	$3.26 \cdot 10^{-4}$	$2.52 \cdot 10^{-4}$	$5.30 \cdot 10^{-4}$	$9.44 \cdot 10^{-4}$
	10	$7.72 \cdot 10^{-4}$	$9.76 \cdot 10^{-4}$	$1.67 \cdot 10^{-3}$	$2.34 \cdot 10^{-3}$
TRRoe	1	$5.40 \cdot 10^{-5}$	$1.06 \cdot 10^{-4}$	$1.13 \cdot 10^{-4}$	$2.08 \cdot 10^{-4}$
	5	$1.74 \cdot 10^{-4}$	$3.33 \cdot 10^{-4}$	$4.74 \cdot 10^{-4}$	$8.72 \cdot 10^{-4}$
	10	$4.23 \cdot 10^{-4}$	$3.96 \cdot 10^{-4}$	$9.39 \cdot 10^{-4}$	$1.76 \cdot 10^{-3}$

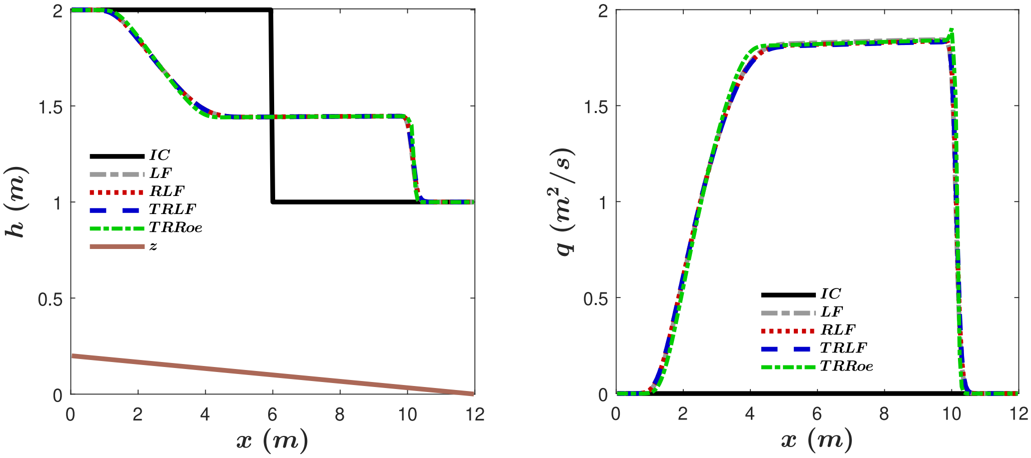
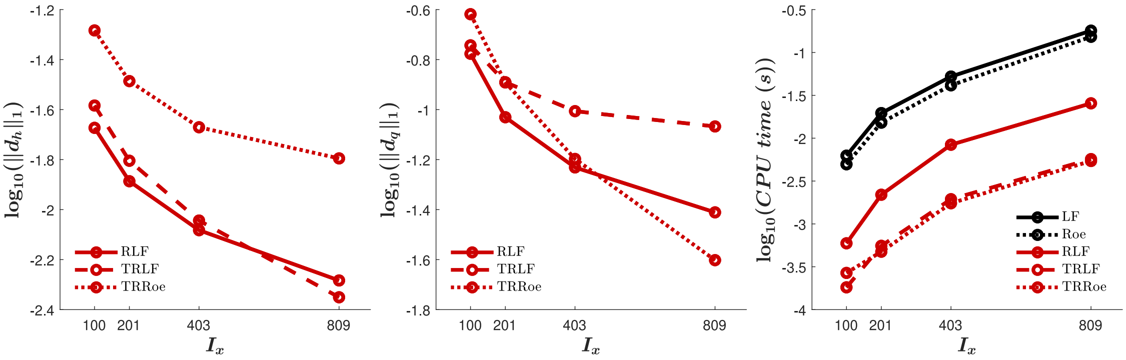


Fig. 4. Case 1: Solutions h (left) and q (right) with $I_x = 201$, $N_w = 24$ and $M_{\text{POD}} = 5$.

To obtain the optimal set of values of the number of POD modes and the number of time windows, it is necessary to take into account the combined action of both parameters and to compare the solutions computed in terms of their differences between the FOM and the ROM and CPU times. The differences for h using the TRRoe ROM with 8 time windows are slightly smaller than with 24 time windows; however, the differences for q with 24 time windows are much smaller than with 8 time windows, as can be seen in Figs. 3(a) and 3(b). Both values of the number of time windows require very similar CPU times, as shown in Fig. 3(c). In terms of the number of POD modes, there is hardly any difference between 5 and 10 POD modes for both h and q and, as shown in Fig. 3(c), the CPU time required by 10 POD modes is higher than that of 5. Taking all this into account, the optimal set of values of the parameters of study when using the TRRoe is 24 time windows (or 4 snapshots per time window) and 5 POD modes.

Fig. 4 shows the solutions for h and q at $T = 1.02$ computed with all versions of the ROM's with $I_x = 201$, and the optimal set of values, i.e., $N_w = 24$ and $M_{\text{POD}} = 5$. The ICs are also included. The TRRoe shows a sharper profile of the water depth, since the discontinuity of the shock wave is better captured than with the RLF and TRLF. Despite the spurious pike in the shock wave, the TRRoe ROM also captures properly the water discharge.

Finally, the same analysis of differences between the solutions of the FOM and the ROM and CPU times is performed with $N_w = 24$ and $M_{\text{POD}} = 5$ by refining the mesh according to the values given in Table 1. As shown in Figs. 5(a) and 5(b), the differences between the errors of the TRRoe and the other two reduced versions are not very large, being less than an order of magnitude, especially for q. Additionally, according to Fig. 5(c), the CPU times are clustered into three groups in decreasing order: (i) the FOMs (LF and Roe); (ii) the ROMs without time averages (RLF); and (iii) the ROMs with time averages (TRLF and TRRoe).



(a) $\|d_h\|_1$. (b) $\|d_q\|_1$. (c) CPU times (s).

Fig. 5. Case 1: $\|d_h\|_1$, $\|d_q\|_1$ and CPU times (s) for different mesh refinements.

Table 6
Case 1: Speed-ups of each version of the ROM.

I_x	100	201	403	809
RLF	10.53	8.96	6.24	7.04
TRLF	34.36	35.37	26.96	31.51
TRRoe	18.52	32.03	23.77	28.14

Table 6 shows the speed-ups of all versions of the ROM with respect to their corresponding FOMs. The TRLF and the TRRoe achieve higher speed-ups than the RLF due to the time averaging approach. This improvement reaches one order of magnitude for finer meshes.

5.2. Case 2. Equilibrium Riemann problem

The interest of Case 2 focuses on testing whether the considered FOMs and ROMs are actually well-balanced or not and on showing that the Lax–Friedrichs method is not able to reproduce results in equilibrium when there are source terms.

The numerical solution of the Riemann problem involved in this type of problems generates a contact wave due to the geometric source term. As explained in [20], augmented Riemann solvers were introduced in an attempt to represent the effect of the source term in the solution of the Riemann problem and automatically fulfil the Rankine–Hugoniot conditions [13].

The Roe method used in this work is defined as an augmented Riemann solver. On the contrary, the standard Lax–Friedrichs method proposed in [63] is not well-balanced. Therefore, a correction term must be added to reproduce properly equilibrium problems. The steady state solution satisfies $\partial q/\partial t = 0$, so from the momentum equation (1), one has

$$\frac{\partial h}{\partial x} = -\frac{\partial z}{\partial x} - \frac{1}{c^2}\frac{\partial (hu^2)}{\partial x},$$

where we used that $q = hu$ and $c^2 = gh$. This equality motivates the well-balanced Lax–Friedrichs-based FOM (WLF), which is obtained by introducing into (4) the following numerical fluxes

$$\mathbf{F}^{n,*,\mp}_{i+1/2} = \frac{1}{2}\left[\mathbf{F}\left(\mathbf{U}^n_i\right) + \mathbf{F}\left(\mathbf{U}^n_{i+1}\right) \mp \mathbf{S}^n_{0,i+1/2}\right] - \frac{1}{2}\nu\frac{\Delta x}{\Delta t}\left(\mathbf{U}^n_{i+1} - \mathbf{U}^n_i - \mathbf{L}^n_{i+1/2}\right), \tag{15}$$

with the following correction term

$$\mathbf{L}^n_{i+1/2} = \begin{pmatrix} -\delta\,(z)^n_{i+1/2} - \dfrac{\delta\left(hu^2\right)^n_{i+1/2}}{\left(c^n_{i+1}\right)^2} \\ 0 \end{pmatrix}.$$

Table 7
Subcases considered in Case 2.

Subcase	FOM	ROM
2.1	LF	RLF
2.2	LF	TRWLF
2.3	LF	TRRoe
2.4	WLF	RLF
2.5	WLF	TRWLF
2.6	WLF	TRRoe
2.7	Roe	RLF
2.8	Roe	TRWLF
2.9	Roe	TRRoe

Table 8
ICs of Cases 2 and 3. The units are in m, m^2/s and s.

Case	h_L	h_R	q_L	q_R	z_L	z_R	T
2	1	0.6245627691	1	1	0	0.3	0.01
3	1	0.1614067989	0	0	0	0.05	0.02

The vector formulation of the corrected Lax–Friedrichs-based ROM (TRWLF) is very similar to the one of the TRLF (13), but with the new term $(\nu/g)J\hat{\mathbf{h}}^n$ in the mass equation due to the well-balancing correction

$$\begin{aligned}
\hat{\mathbf{h}}^{n+1} &= \hat{\mathbf{h}}^n - \frac{1}{2}\frac{\Delta t}{\Delta x}A\hat{\mathbf{q}}^n + \frac{1}{2}\nu B\hat{\mathbf{h}}^n + \frac{1}{2}\nu C + \frac{\nu}{g}J\hat{\mathbf{h}}^n,\\
\hat{\mathbf{q}}^{n+1} &= \hat{\mathbf{q}}^n - \frac{1}{2}\frac{\Delta t}{\Delta x}E\hat{\mathbf{h}}^n - \frac{1}{4}g\frac{\Delta t}{\Delta x}\left(\hat{\mathbf{h}}^n\right)^T F\hat{\mathbf{h}}^n + \frac{1}{2}\nu G\hat{\mathbf{q}}^n - \frac{1}{4}\frac{\Delta t}{\Delta x}gH\hat{\mathbf{h}}^n - \Delta t g I,
\end{aligned} \tag{16}$$

where the training matrices $A, B, \ldots, I$ are the same as in (13) and (14) and the entries of the matrix $J = \left(J_{kp}\right)$ are

$$J_{kp} = j_1\phi_{1,p} + \sum_{i=2}^{I_x-1}\left(\frac{\phi_{i+1,k}\left(\bar{u}^w_{i+1}\right)^2 - \phi_{i,k}\left(\bar{u}^w_i\right)^2}{\bar{h}^w_{i+1} + \bar{h}^w_i} - \frac{\phi_{i,k}\left(\bar{u}^w_i\right)^2 - \phi_{i-1,k}\left(\bar{u}^w_{i-1}\right)^2}{\bar{h}^w_i + \bar{h}^w_{i-1}}\right)\phi_{i,p} + j_{I_x}\phi_{I_x,p}.$$

Next, the performance of the ROMs is analysed when faced with these problems and whether the training solution must be also well-balanced or not. To this end, the same problem is solved by means of the 9 subcases shown in Table 7. According to [59], Subcases 2.1, 2.5 and 2.9 are consistent and the remaining subcases are inconsistent.

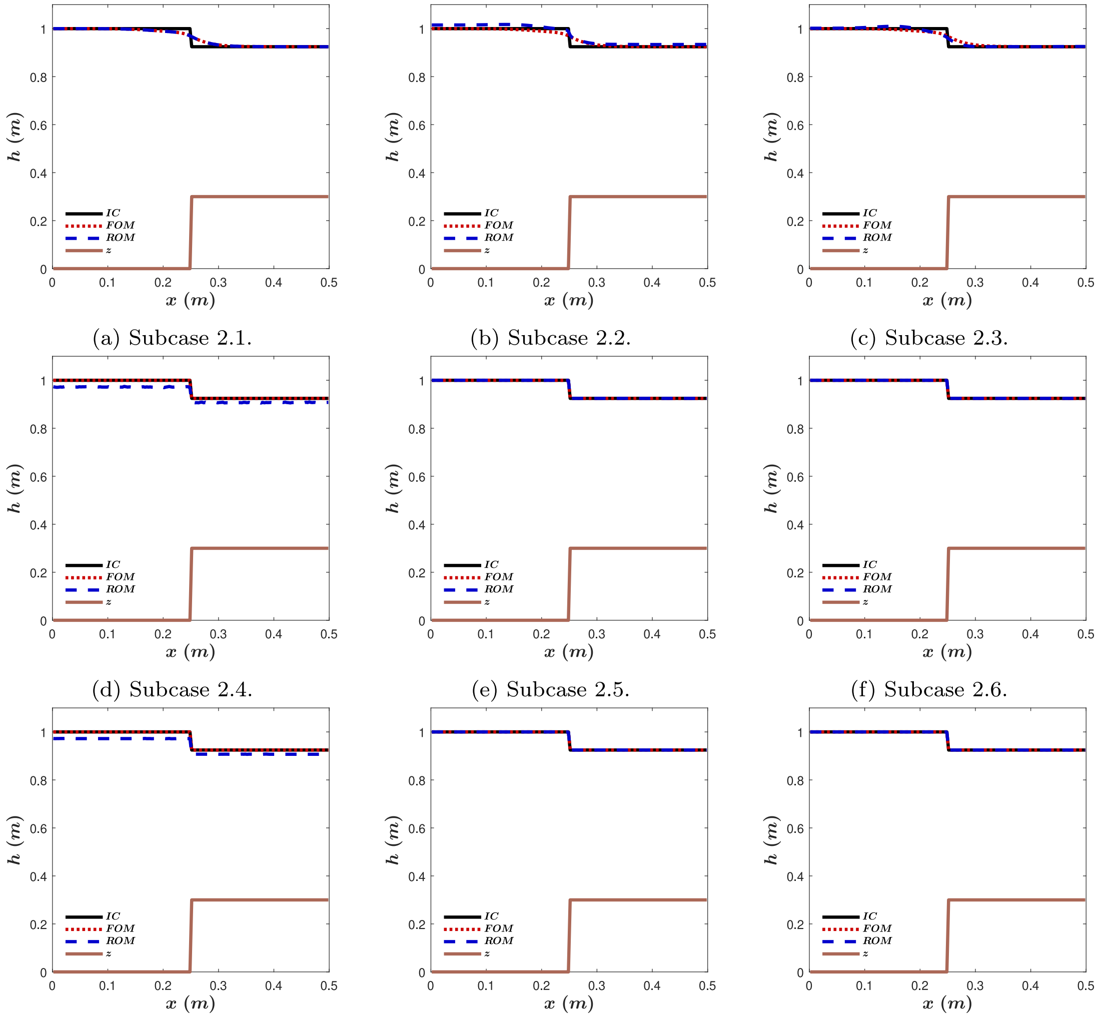
The equilibrium Riemann problem RP2 proposed in [20] is here replicated. The solution is discontinuous and it satisfies the Rankine–Hugoniot condition [13], whose ICs are shown in Table 8 and free BCs are imposed (see Figs. 6 and 7).

In essence, the ROMs reproduce properly the solutions of the FOM in Subcases 2.1, 2.5, 2.6, 2.8 and 2.9, but the solution of Subcase 2.1 is not well-balanced and therefore does not make physical sense and does not have to be taken into account. Similarly, Subcases 2.2 and 2.3 are trained by the LF and are not well-balanced. In the remaining Subcases 2.4 and 2.7 in which the RLF is trained by the WLF and the Roe, respectively, do not compute satisfactory results. In the end, it can be concluded that ROMs must have been developed and trained from balanced FOMs, but they need not be consistent.

5.3. Case 3. Transient Riemann problem

Case 3 takes a further step in the study of relationship between the well-balancing property and ROMs. To this end, the transient Riemann problem proposed in [20] as RP1 is considered. The ICs are shown in Table 8 and free BCs are considered.

Again, combinations of different FOMs and ROMs are studied, but, since the LF method is not well-balanced as seen in Case 2, it is suppressed in the analysis of Case 3 and only the combinations listed in the Table 9 are considered. Again, there are consistent Subcases, 3.1 and 3.4, and inconsistent Subcases, 3.2 and 3.3.



(a) Subcase 2.1.
(b) Subcase 2.2.
(c) Subcase 2.3.
(d) Subcase 2.4.
(e) Subcase 2.5.
(f) Subcase 2.6.
(g) Subcase 2.7.
(h) Subcase 2.8.
(i) Subcase 2.9.

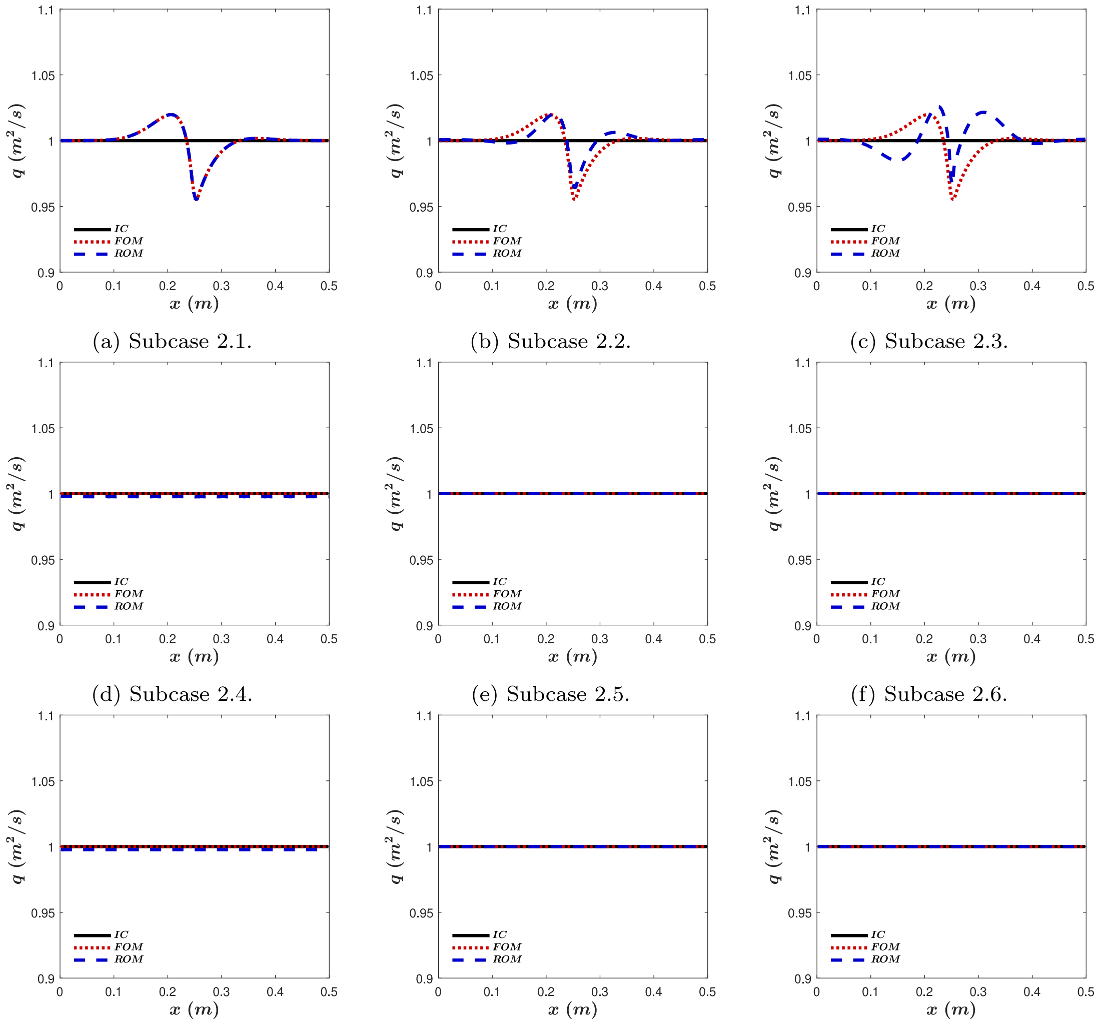
Fig. 6. Case 2: Solutions h for all subcases considered in Table 7.

Table 9
Subcases considered in Case 3.

Subcase	FOM	ROM
3.1	WLF	TRWLF
3.2	WLF	TRRoe
3.3	Roe	TRWLF
3.4	Roe	TRRoe

Moreover, the values of the parameters studied in the previous case are kept constant, namely $I_x = 320$, $N_w = 29$ ($N_T = 58$) and $M_{\text{POD}} = 5$. The CFL number is varied to test the performance of the ROMs. Figs. 8 and 9 show the solutions of h and q respectively computed with the different subcases and for different values of the CFL number: 0.1, 0.5 and 0.9. The exact solution is also included in these figures.

The most basic aspect that is worth mentioning in relation to these results is that the FOMs show different dependencies on the CFL number: while the WLF is highly dissipative for low CFL numbers, as shown in Subcases 3.1 and 3.2 in Figs. 8(a) and 9(a), the Roe shows hardly any numerical dissipation for different CFL values.



(a) Subcase 2.1. (b) Subcase 2.2. (c) Subcase 2.3.

(d) Subcase 2.4. (e) Subcase 2.5. (f) Subcase 2.6.

(g) Subcase 2.7. (h) Subcase 2.8. (i) Subcase 2.9.

Fig. 7. Case 2: Solutions q for all subcases considered in Table 7.

Regarding the water depth, all subcases obtain satisfactory results for high CFL numbers, as shown in Fig. 8, even the heterogeneous Subcases 3.2 and 3.3 in which the training FOM is different from the developing FOM. As the CFL number decreases, the results of Subcases 3.2 and 3.3 get worse, leading to solutions with no physical meaning with $CFL = 0.1$. These solutions underestimate the value of the water depth to the left of the rarefaction wave, where no perturbation is expected.

In terms of water discharge, the performance of the ROMs is slightly worse, as can be seen in Fig. 9. This difference with the water depth results may be due to the time averaging approach. Subcase 3.4 shows proper results of the water discharge, even though it underestimates the maximum when $CFL = 0.9$. This behaviour is explained in the following paragraph. The heterogeneous Subcases 3.2 and 3.3 are not able to ensure a proper calculation of the water discharge, as they under- or overestimate it for all values of the CFL number (except Subcase 3.2 with $CFL = 0.9$, which incidentally is correct). Finally, Subcase 3.1 shows good agreement between the ROM and FOM solutions, but both differ from the exact solution when the CFL is lowered.

The differences of the solutions of all subcases between the FOM and the ROM are included in Table 10. All subcases obtain accurate solutions, but, as expected, the differences in the homogeneous Subcases 3.1 and 3.4 are smaller than in the heterogeneous Subcases 3.2 and 3.3. Although Subcase 3.1 obtains better differences than

Table 10
Case 3: $\|d_h\|_1$ and $\|d_q\|_1$ computed with the ROM measured with respect to the FOM solution.

	Subcase	CFL		
		0.1	0.5	0.9
$\|d_h\|_1$	3.1	$8.22 \cdot 10^{-5}$	$5.25 \cdot 10^{-5}$	$2.22 \cdot 10^{-4}$
	3.2	$3.35 \cdot 10^{-3}$	$1.01 \cdot 10^{-3}$	$8.76 \cdot 10^{-4}$
	3.3	$1.31 \cdot 10^{-2}$	$2.77 \cdot 10^{-3}$	$1.92 \cdot 10^{-3}$
	3.4	$1.89 \cdot 10^{-4}$	$4.52 \cdot 10^{-4}$	$3.22 \cdot 10^{-4}$
$\|d_q\|_1$	3.1	$5.51 \cdot 10^{-5}$	$5.74 \cdot 10^{-4}$	$3.28 \cdot 10^{-3}$
	3.2	$5.74 \cdot 10^{-2}$	$9.78 \cdot 10^{-3}$	$2.14 \cdot 10^{-3}$
	3.3	$5.10 \cdot 10^{-2}$	$1.54 \cdot 10^{-2}$	$1.19 \cdot 10^{-2}$
	3.4	$6.73 \cdot 10^{-4}$	$8.13 \cdot 10^{-4}$	$4.00 \cdot 10^{-3}$

Table 11
Case 3: CPU times (s) of the WLF and the Roe.

FOM	CFL		
	0.1	0.5	0.9
WLF	$5.75 \cdot 10^{-2}$	$1.91 \cdot 10^{-2}$	$1.33 \cdot 10^{-2}$
Roe	$7.41 \cdot 10^{-2}$	$1.94 \cdot 10^{-2}$	$1.52 \cdot 10^{-2}$

Table 12
Case 3: CPU times (s) of all subcases.

Subcase	CFL		
	0.1	0.5	0.9
3.1	$5.03 \cdot 10^{-3}$	$1.54 \cdot 10^{-3}$	$8.84 \cdot 10^{-4}$
3.2	$4.74 \cdot 10^{-3}$	$1.46 \cdot 10^{-3}$	$8.30 \cdot 10^{-4}$
3.3	$7.72 \cdot 10^{-3}$	$1.57 \cdot 10^{-3}$	$8.78 \cdot 10^{-4}$
3.4	$7.32 \cdot 10^{-3}$	$1.52 \cdot 10^{-3}$	$8.41 \cdot 10^{-4}$

Table 13
Case 3: Speed-ups of all subcases.

Subcase	CFL		
	0.1	0.5	0.9
3.1	11.43	12.43	15.02
3.2	12.14	13.12	16.00
3.3	9.61	12.35	17.27
3.4	10.13	12.77	18.03

Subcase 3.4, it must be recalled that these solutions are very different from the exact solution due to the high numerical dissipation. Subcase 3.4 shows an increasing tendency of $\|d_q\|_1$ when the CFL number increases. This may be related to the time averaging approach, because the larger the CFL number, the larger the time step and therefore the wider the time windows, which may imply a slight loss of accuracy in the solution. In spite of this, Subcase 3.4, with approximately constant differences, is the most suitable method to compute proper solutions to this problem, as seen in Figs. 8 and 9.

Tables 11 and 12 contain all the CPU times required by the FOMs and all subcases, respectively, to compute their solutions. The ROMs, in all subcases, are faster than their corresponding FOMs.

In terms of speed-up, all subcases reach at least one order of magnitude with respect to their corresponding FOMs, as can be seen in Table 13.

All in all, well-balanced FOMs such as the WLF and the Roe can be used to train their corresponding well-balanced ROMs, the TRWLF and the RRoe, to solve transient problems with high accuracy. However, the WLF is too dissipative, so that its solutions do not match up to what they should be. The numerical results of Case 3

Table 14
Case 4: Positions of the gauging points.

Point	G_4	G_10	G_{11}	G_{13}	G_{20}
Position (m)	19.5	25.5	26.5	28.5	35.5

Table 15
Case 4: CPU times required by the Roe and the TRRoe for different mesh refinements and the corresponding CPU times per time step and the corresponding speed-ups.

	I_x N_T	100 3212	200 6512	400 13 132	600 19 748
Roe	CPU time (s) CPU time/N_T (s)	0.14 $4.21 \cdot 10^{-5}$	0.45 $6.96 \cdot 10^{-5}$	1.54 $1.17 \cdot 10^{-4}$	3.37 $1.71 \cdot 10^{-4}$
TRRoe	CPU time (s) CPU time/N_T (s)	$6.05 \cdot 10^{-3}$ $1.88 \cdot 10^{-6}$	$2.65 \cdot 10^{-2}$ $4.07 \cdot 10^{-6}$	$8.71 \cdot 10^{-2}$ $6.63 \cdot 10^{-6}$	$1.92 \cdot 10^{-1}$ $9.72 \cdot 10^{-6}$
	Speed-up	**22.35**	**17.08**	**17.63**	**17.57**

show that the TRRoe obtains highly efficient results when trained with the solutions of the Roe independently of the CFL number. If trained with WLF solutions, non-physical results are obtained for low CFL numbers.

5.4. Case 4. Experimental data test: dam-break over a triangular obstacle

The last case is focused on studying the performance of the TRRoe on a realistic problem for which there are experimental data. This case serves to test the relationship of the TRRoe with the numerical corrections detailed in Section 3.2. All these corrections need to be used in this problem. In order to do so, the TRRoe will be trained with results computed by the Roe in which these corrections have been taken into account.

The problem used for this purpose has been proposed in [64] and [12, Section 8.2]. Experimental data was obtained from the Recherches Hydrauliques Lab. Châtelet together with the University of Bruxelles (Belgium) under the supervision of J.M. Hiver. The test case deals with the evolution of a dam-break wave over a triangular obstacle. The channel geometry is shown in Fig. 10. Glauker–Manning formula is used to compute friction losses with $n_b = 0.0125$ s/m$^{1/3}$. The ICs are defined as a water reservoir

$$h(x,0) = \begin{cases} 0.75 \text{ m}, & \text{if } 0 \text{ m} \le x < 15.5 \text{ m}, \\ 0 \text{ m}, & \text{if } 15.5 \text{ m} \le x \le 38 \text{ m}, \end{cases} \qquad q(x,0) = 0 \text{ m}^2/\text{s}, \quad 0 \text{ m} \le x \le 38 \text{ m},$$

and solid wall and free outlet BCs are considered upstream and downstream, respectively.

Experimental water depth data were measured at five gauging points along the channel. The positions of the gauging points G_4, G_{10}, G_{11}, G_{13} and G_{20} are shown in Fig. 10 and contained in Table 14.

Fig. 11 shows the results of h and q computed by the Roe and the TRRoe (with $I_x = 400$, $CFL = 0.1$, $N_w = 3283$ and $M_{\text{POD}} = 5$) along the channel at different times, namely $t = 3$ s, 5 s, 10 s and 20 s.

Fig. 12 shows the time evolution of h (left) and q (right) at the positions of the five gauging points G_4, G_{10}, G_{11}, G_{13} and G_{20}.

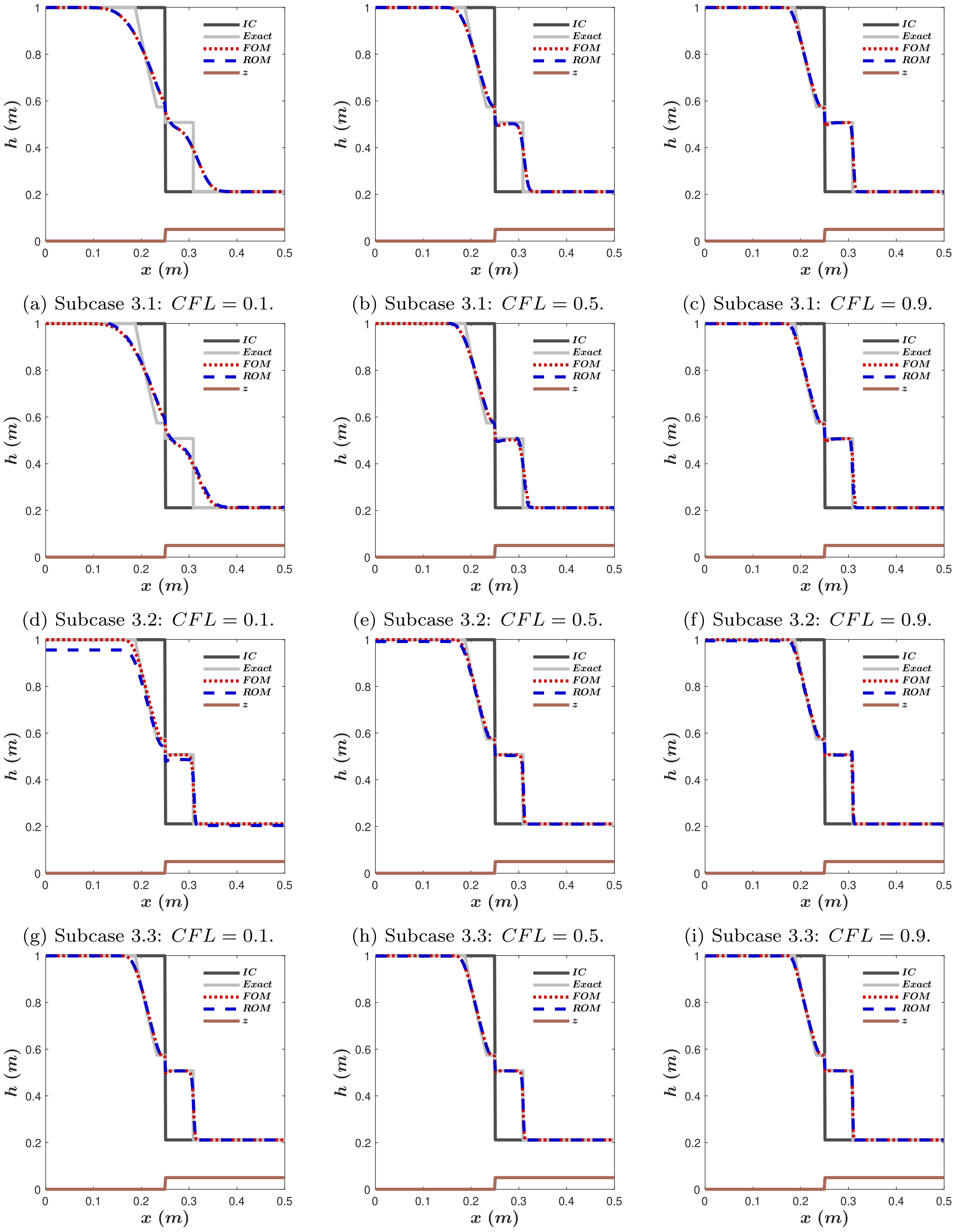
Table 15 contains the CPU times required by the Roe and the TRRoe to obtain their results for different mesh refinements. It also shows that the corresponding CPU times per time step and the speed-ups of the TRRoe with respect to the Roe achieve one order of magnitude.

The numerical results of Case 4 show that the TRRoe does not need to include entropy fix, wet–dry treatment or other corrections to successfully reproduce realistic problems when it has been trained using solutions computed with the Roe as FOM in which these corrections are taken into account.

Finally, an extension of this problem to the 2D case is carried out following [13] to show more clearly the computational advantages of ROMs. The ICs of the problem is similar to the 1D version

$$h(x,y,0) = \begin{cases} 0.75 \text{ m}, & \text{if } 0 \text{ m} \le x < 15.5 \text{ m}, \ 0 \text{ m} \le y \le 2 \text{ m}, \\ 0 \text{ m}, & \text{if } 15.5 \text{ m} \le x \le 38 \text{ m}, \ 0 \text{ m} \le y \le 2 \text{ m}, \end{cases}$$

$$q_x(x,y,0) = q_y(x,y,0) = 0 \text{ m}^2/\text{s}, \ \forall x, y,$$



(a) Subcase 3.1: $CFL = 0.1$.

(b) Subcase 3.1: $CFL = 0.5$.

(c) Subcase 3.1: $CFL = 0.9$.

(d) Subcase 3.2: $CFL = 0.1$.

(e) Subcase 3.2: $CFL = 0.5$.

(f) Subcase 3.2: $CFL = 0.9$.

(g) Subcase 3.3: $CFL = 0.1$.

(h) Subcase 3.3: $CFL = 0.5$.

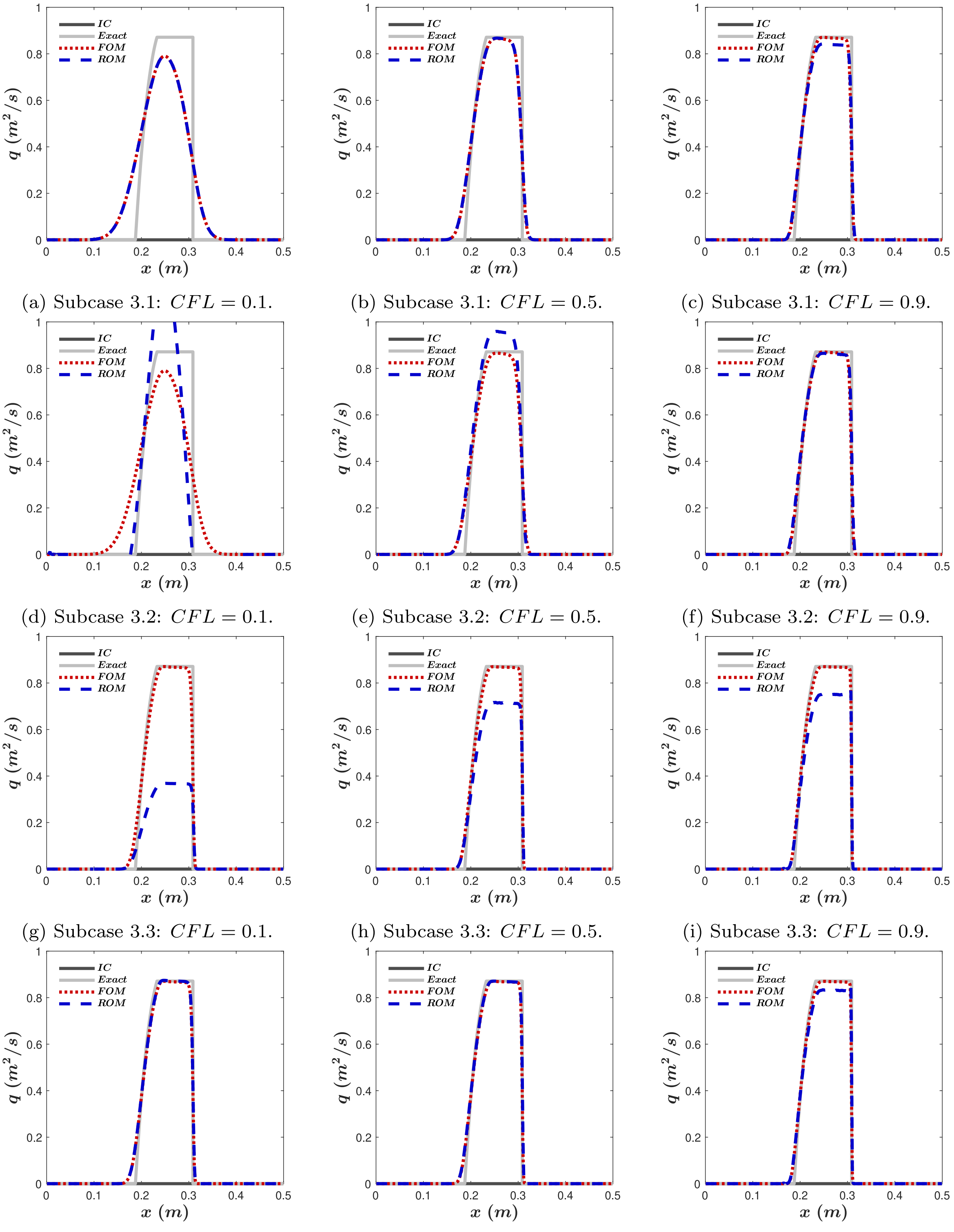
(i) Subcase 3.3: $CFL = 0.9$.

(j) Subcase 3.4: $CFL = 0.1$.

(k) Subcase 3.4: $CFL = 0.5$.

(l) Subcase 3.4: $CFL = 0.9$.

Fig. 8. Case 3: Solution h for $I_x = 320$, $N_s = 2$ and $M_{\text{POD}} = 5$.



(a) Subcase 3.1: $CFL = 0.1$.

(b) Subcase 3.1: $CFL = 0.5$.

(c) Subcase 3.1: $CFL = 0.9$.

(d) Subcase 3.2: $CFL = 0.1$.

(e) Subcase 3.2: $CFL = 0.5$.

(f) Subcase 3.2: $CFL = 0.9$.

(g) Subcase 3.3: $CFL = 0.1$.

(h) Subcase 3.3: $CFL = 0.5$.

(i) Subcase 3.3: $CFL = 0.9$.

(j) Subcase 3.4: $CFL = 0.1$.

(k) Subcase 3.4: $CFL = 0.5$.

(l) Subcase 3.4: $CFL = 0.9$.

Fig. 9. Case 3: Solution q for $I_x = 320$, $N_s = 2$ and $M_{\text{POD}} = 5$.

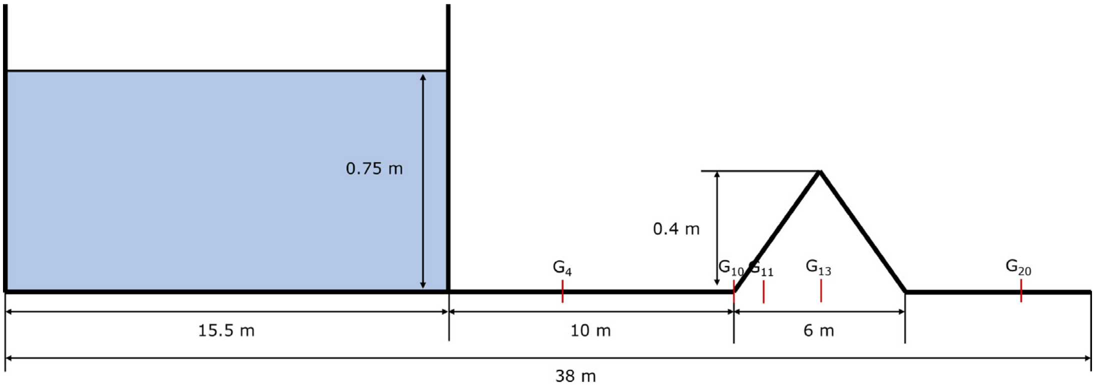
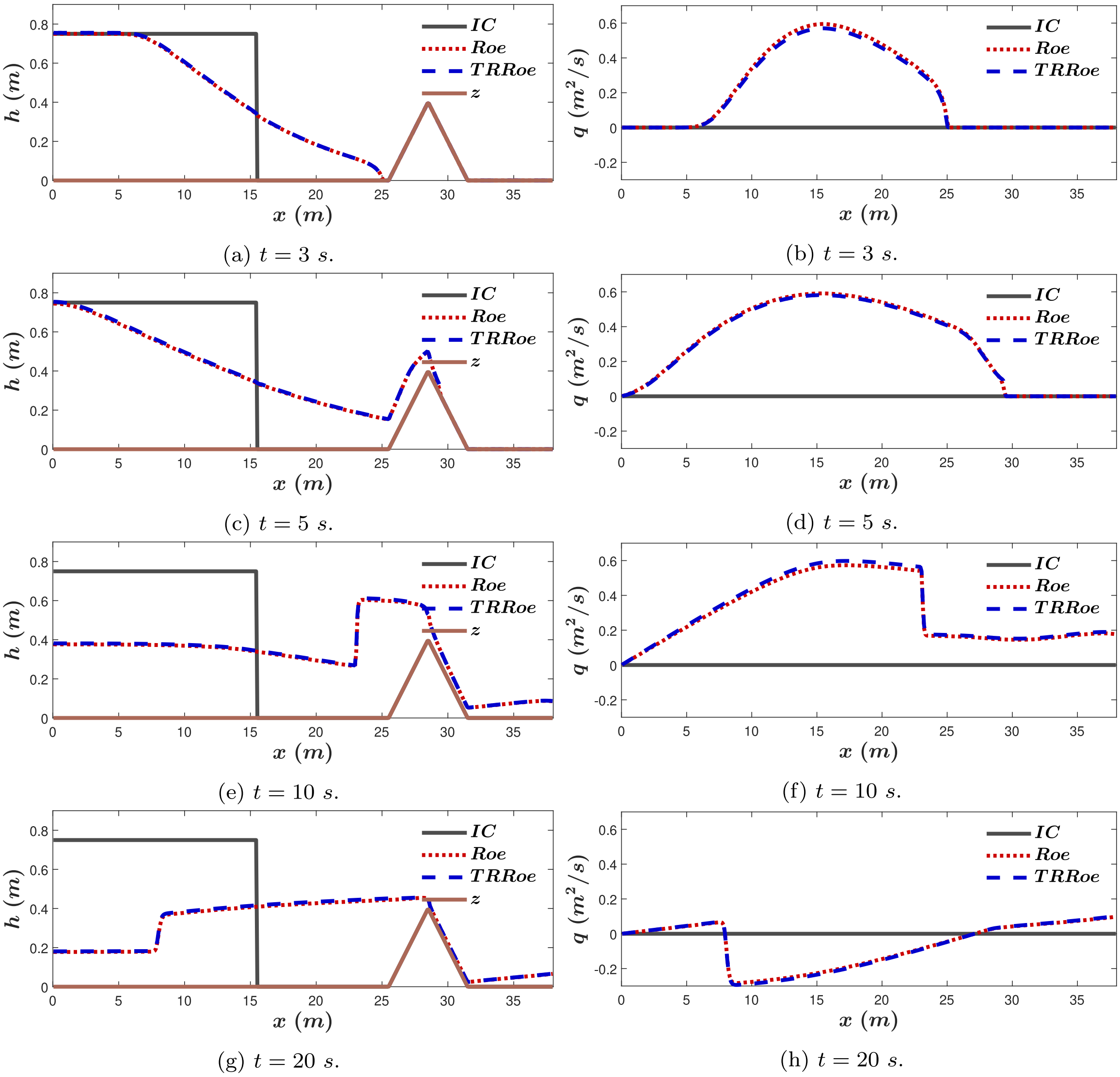


Fig. 10. Case 4: Geometry of the channel.



(a) $t = 3\ s$.

(b) $t = 3\ s$.

(c) $t = 5\ s$.

(d) $t = 5\ s$.

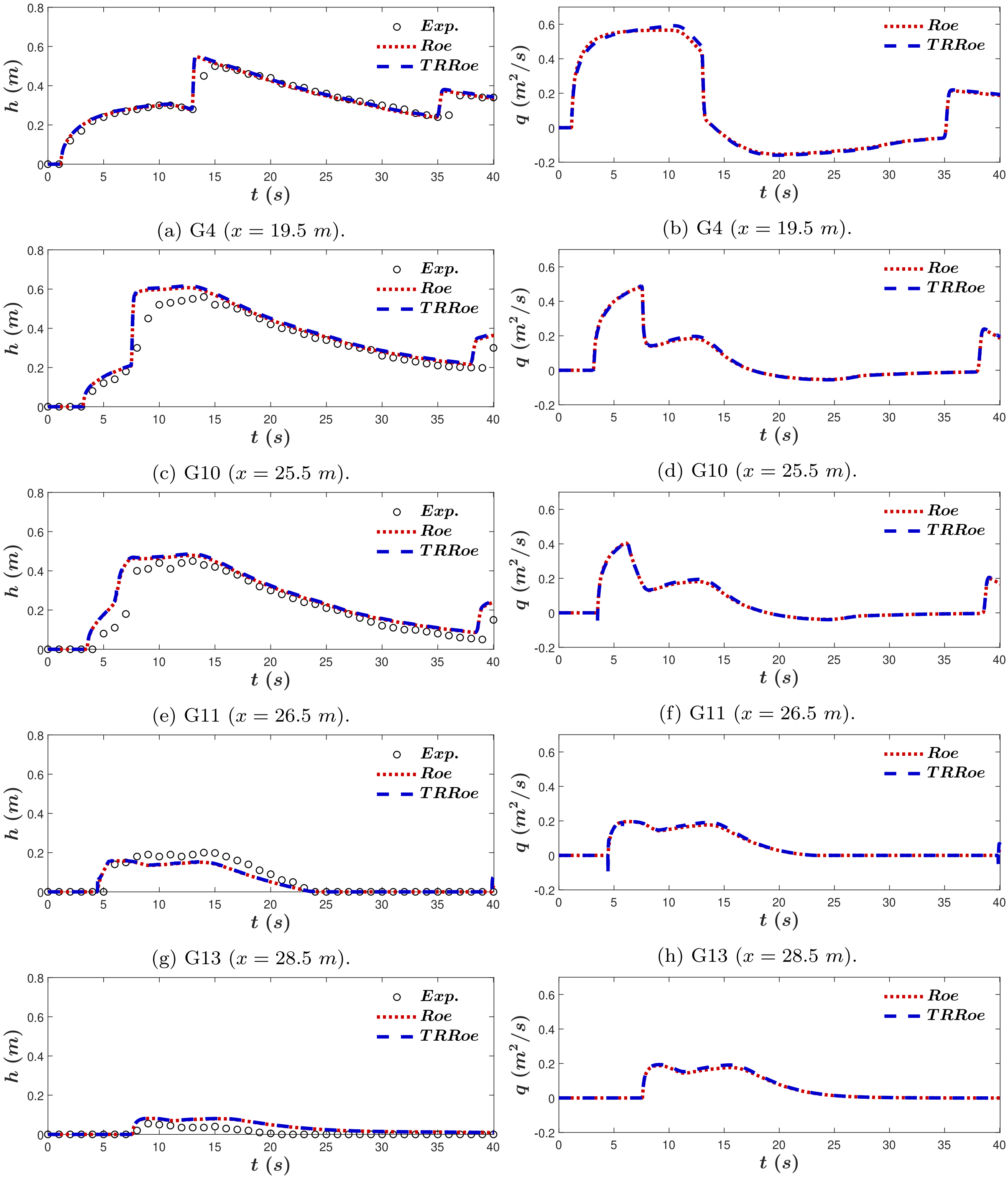
(e) $t = 10\ s$.

(f) $t = 10\ s$.

(g) $t = 20\ s$.

(h) $t = 20\ s$.

Fig. 11. Case 4: Results of h (left) and q (right) along the channel at different times computed with the Roe and the TRRoe with $I_x = 400$, $CFL = 0.1$, $N_w = 3283$ and $M_{\text{POD}} = 5$.



(a) G4 ($x = 19.5$ m).

(b) G4 ($x = 19.5$ m).

(c) G10 ($x = 25.5$ m).

(d) G10 ($x = 25.5$ m).

(e) G11 ($x = 26.5$ m).

(f) G11 ($x = 26.5$ m).

(g) G13 ($x = 28.5$ m).

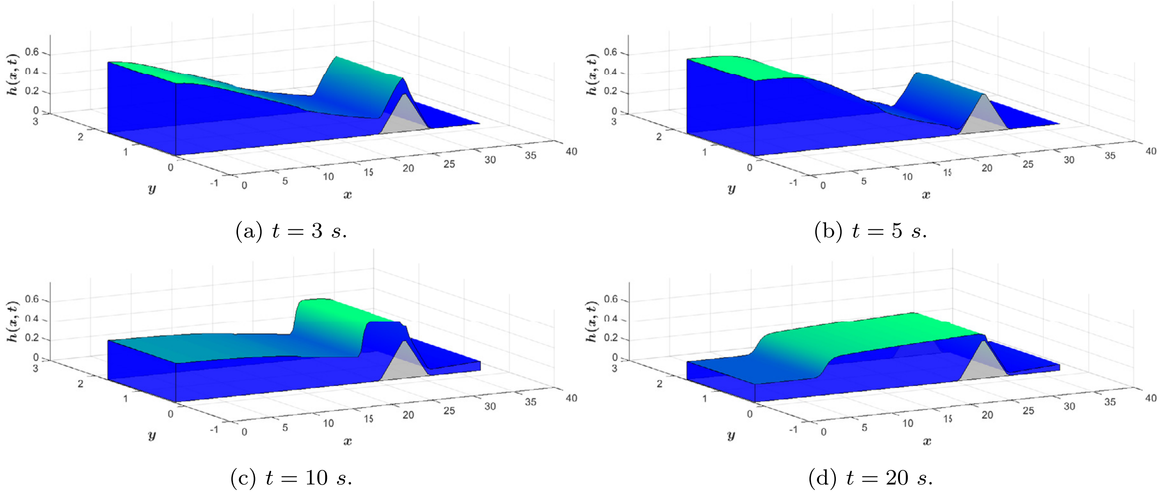
(h) G13 ($x = 28.5$ m).

(i) G20 ($x = 32.5$ m).

(j) G20 ($x = 32.5$ m).

Fig. 12. Case 4: Results of h (compared with the experimental data, left) and q (right) in the different gauging points computed with the Roe and the TRRoe with $I_x = 400$, $N_w = 3283$ and $M_{\text{POD}} = 5$.

where q_x and q_y are the water discharges in the x- and y-direction, respectively, and solid wall and free outlet BCs are considered upstream and downstream, respectively.



(a) $t = 3\ s$.

(b) $t = 5\ s$.

(c) $t = 10\ s$.

(d) $t = 20\ s$.

Fig. 13. Case 4: 2D SWE results of h along the channel at different times computed with the TRRoe with $I_x = 100$, $I_y = 4$, $N_w = 819$ and $M_{\text{POD}} = 5$.

Table 16
Case 4: CPU times required by the 2D version of the Roe and the TRRoe for different mesh refinements and the corresponding CPU times per time step and the corresponding speed-ups.

	$I_x \times I_y$ N_T	100 × 4 3274	200 × 4 6634	400 × 4 13 369	600 × 4 20 121
Roe	CPU time (s) CPU time/N_T (s)	1.87 $5.70 \cdot 10^{-4}$	6.61 $9.97 \cdot 10^{-4}$	25.71 $1.92 \cdot 10^{-3}$	59.06 $2.94 \cdot 10^{-3}$
TRRoe	CPU time (s) CPU time/N_T (s)	$3.06 \cdot 10^{-2}$ $9.35 \cdot 10^{-6}$	$1.20 \cdot 10^{-1}$ $1.82 \cdot 10^{-5}$	$4.75 \cdot 10^{-1}$ $3.55 \cdot 10^{-5}$	1.09 $5.41 \cdot 10^{-5}$
	Speed-up	**60.98**	**54.89**	**54.14**	**54.30**

Fig. 13 shows the results of h computed with the TRRoe taking $I_x = 100$, $I_y = 4$, $CFL = 0.1$, $N_w = 819$ and $M_{\text{POD}} = 5$ along the channel at $t = 3$ s, 5 s, 10 s and 20 s.

Table 16 contains the CPU times required by the Roe and the TRRoe to obtain their results for different mesh refinements for the 2D case. It also shows the corresponding CPU times per time step and the corresponding speed-ups of the TRRoe with respect to the Roe. The speed-ups of one order of magnitude are faster than those of the 1D case. As possible future work, it would be useful to study what is the most optimal set of values for the number of POD modes and the number of time windows in the 2D case to improve the CPU times required.

6. Concluding remarks

POD-based ROMs have proven to be very useful tools in solving transport problems, such as the 1D SWE [33]. When solving realistic problems modelled using the 1D SWE, the FOMs used need to take into account some numerical corrections, such as the well-balanced property, the entropy fix and the wet–dry treatment. This work is focused on studying the performance of a ROMs developed from a Roe-based FOM and what properties it preserves from them.

The time averaging approach has been essential to fully develop the Roe-based ROM, but at the same time it has proven to be useful for saving CPU time. Since it can be used to apply the Galerkin decomposition only to the water depth and discharge, the TRRoe do not need to recall the physical variables in each time step to compute the water velocity and finally they are faster. Case 1 has shown that this approach allows the development of ROMs that compute very efficient solutions by comparison between the RLF and the TRLF and the TRRoe itself.

Regarding the numerical corrections, the results of Cases 2 and 3 evidence that the ROM needs to be developed from a well-balanced FOM, as is the case for the TRRoe. And also it has to be trained with well-balanced solutions to accurately reproduce well-balance solutions, but does not require to be consistent.

On the contrary, the TRRoe does not need to take into account other corrections such as the entropy fix and the wet–dry treatment (provided it has been trained by corrected FOMS), as it has been proved in Case 4. This case also serves as an example of the ability of ROMs, if well designed, to reproduce realistic problems, since their solutions accurately approach the measurement data. The reason for this is the following.

For a scheme to be well-balanced, it is necessary to consider a correction term to the numerical scheme, as done with the Lax–Friedrichs method in Case 2. Because of this, the ROM that is developed from a scheme with a well-balanced correction is different from one that is developed from a scheme without such a correction. However, the other numerical corrections considered in this work lead to a change in the values of the variables if certain conditions that are evaluated at each time instant are met. In this way, the numerical scheme is not modified and the TRRoe does not have to take these corrections into account.

As possible future work, it would be interesting to apply these augmented Riemann solvers-based ROMs to problems with Dirichlet BCs following [65], to perform parametric sensitivity analysis as done in [52,53], and also extend them to 2D shallow water problems following works as [39]. In the field of boundary condition problems, the shifted boundary method applied in recent work such as [66,67] could be very useful. It would also be worth extending the study of consistency through the use of higher order schemes and analyse how they respond and whether they maintain the order of convergence or not.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request

Acknowledgements

This work was funded by the Spanish Ministry of Science and Innovation under the research project PGC2018-094341-B-I00. This work has also been partially funded by Gobierno de Aragón through Fondo Social Europeo (T32-20R and E24-17R, Feder 2019–2021 “Construyendo Europa desde Aragón”).

References

[1] P.K. Kundu, I.M. Cohen, Fluid Mechanics, Elsevier Science, 2010.

[2] J.A. Liggett, Fluid Mechanics, McGraw-Hill international editions, 1994.

[3] C.B. Vreugdenhil, Numerical Methods for Shallow-Water Flow, in: Water Science and Technology Library, Springer Netherlands, 1994.

[4] A. Bermudez, M.E. Vázquez-Cendón, Upwind methods for hyperbolic conservation laws with source terms, Comput. & Fluids 23 (8) (1994) 1049–1071.

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

[6] S. Khosh Bin Ghomash, D. Bachmann, D. Caviedes-Voullième, C. Hinz, Impact of rainfall movement on flash flood response: A synthetic study of a semi-arid mountainous catchment, Water 14 (12) (2022).

[7] G. Ran, Q. Zhang, L. Li, A depth-integrated non-hydrostatic model for nearshore wave modelling based on the discontinuous Galerkin method, Ocean Eng. 232 (2021) 108661.

[8] C. Hirsch, Numerical Computation of Internal and External Flows: The Fundamentals of Computational Fluid Dynamics, Elsevier Science, 2007.

[9] R.J. LeVeque, Finite Volume Methods for Hyperbolic Problems, in: Cambridge Texts in Applied Mathematics, Cambridge University Press, 2002.

[10] I. Echeverribar, M. Morales-Hernández, P. Brufau, P. García-Navarro, 2D numerical simulation of unsteady flows for large scale floods prediction in real time, Adv. Water Resour. 134 (2019) 103444.

[11] P. García-Navarro, M.E. Vázquez-Cendón, On numerical treatment of the source terms in the shallow water equations, Comput. & Fluids 29 (8) (2000) 951–979.

[12] J. Murillo, P. García-Navarro, Augmented versions of the HLL and HLLC Riemann solvers including source terms in one and two dimensions for shallow flow applications, J. Comput. Phys. 231 (20) (2012) 6861–6906.

[13] J. Murillo, A. Navas-Montilla, A comprehensive explanation and exercise of the source terms in hyperbolic systems using Roe type solutions. Application to the 1D-2D shallow water equations, Adv. Wat. Res. 98 (2016) 70–96.

[14] B. Einfeldt, On godunov-type methods for gas dynamics, SIAM J. Num. Anal. 25 (2) (1988) 294–318.

[15] A. Harten, P.D. Lax, B. Leer, On upstream differencing and godunov-type schemes for hyperbolic conservation laws, SIAM Rev. 25 (1) (1983) 35–61.

[16] E.F. Toro, M. Spruce, W. Speares, Restoration of the contact surface in the HLL-Riemann solver, Shock Waves 4 (1994) 25–34.

[17] G.S. Jiang, C.W. Shu, Efficient implementation of weighted ENO schemes, J. Comput. Phys. 126 (1) (1996) 202–228.

[18] X.D. Liu, S. Osher, T. Chan, Weighted essentially non-oscillatory schemes, J. Comput. Phys. 115 (1) (1994) 200–212.

[19] D.A. Haleem, G. Kesserwani, D Caviedes-Voullième, Haar wavelet-based adaptive finite volume shallow water solver, J. Hydroinformatics 17 (6) (2015) 857–873.

[20] A. Navas-Montilla, P. Solán-Fustero, J. Murillo, P. García-Navarro, Discontinuous Galerkin well-balanced schemes using augmented Riemann solvers with application to the shallow water equations, J. Hydroinformatics 22 (5) (2020) 1038–1058.

[21] C. Prud'Homme, D.V. Rovas, K. Veroy, L. Machiels, Y. Maday, A.T. Patera, G. Turinici, Reliable real-time solution of parametrized partial differential equations: Reduced-basis output bound methods, J. Fluids Eng. 124 (1) (2001) 70–80.

[22] J.L. Lumley, The structure of inhomogeneous turbulent flows, Atmos. Turb. Rad. Wave Prop. (1967) 166–176.

[23] K. Karhunen, Über lineare methoden in der wahrscheinlichkeitsrechnung, Ann. Acad. Sci. Fennicae. Ser. A. I. Math.-Phys. 37 (1947) 1–79.

[24] M. Loève, Probability Theory, 2nd ed., in: The University Series in Higher Mathematics, D. Van Nostrand Co., Inc., Princeton, N. J.-Toronto-New York-London, 1960, p. xvi+685.

[25] K. Pearson, LIII. On lines and planes of closest fit to systems of points in space, Lond., Edinb., Dublin Philos. Mag. J. Sci. 2 (11) (1901) 559–572.

[26] I.T. Jolliffe, Principal Component Analysis, in: Springer Series in Statistics, Springer, 2002.

[27] M. Ahmed, O. San, Stabilized principal interval decomposition method for model reduction of nonlinear convective systems with moving shocks, J. Comput. Appl. Math. 37 (2018) 6870–6902.

[28] J. Borggaard, A. Hay, D. Pelletier, Interval-based reduced-order models for unsteady fluid flow, Int. J. Numer. Anal. Model. 4 (2007) 353–367.

[29] E.A. Christensen, M. Brøns, J.N. Sørensen, Evaluation of proper orthogonal decomposition–based decomposition techniques applied to parameter-dependent nonturbulent flows, SIAM J. Sci. Comput. 21 (4) (1999) 1419–1434.

[30] W.L. IJzerman, Signal Representation and Modeling of Spatial Structures in Fluids (Ph.D. thesis), Universiteit Twente, 2000.

[31] D. Rim, B. Peherstorfer, K.T. Mandli, Manifold Approximations via Transported Subspaces: Model reduction for transport-dominated problems, 2020, ArXiv 1912.13024.

[32] A. Towne, O.T. Schmidt, T. Colonius, Spectral proper orthogonal decomposition and its relationship to dynamic mode decomposition and resolvent analysis, J. Fluid Mech. 847 (2018) 821–867.

[33] J.-M. Zokagoa, A. Soulaïmani, A POD-based reduced-order model for uncertainty analyses in shallow water flows, Int. J. Comput. Fluid Dyn. 32 (6–7) (2018) 278–292.

[34] L. Sirovich, Turbulence and the dynamics of coherent structures. I - Coherent structures. II - symmetries and transformations. III - Dynamics and scaling, Q. Appl. Math. 45 (1987) 561–571, 573–582, 583–590.

[35] H. Lu, D.M. Tartakovsky, Lagrangian dynamic mode decomposition for construction of reduced-order models of advection-dominated phenomena, J. Comput. Phys. 407 (2020) 109229.

[36] R. Mojgani, M. Balajewicz, Low-rank registration based manifolds for convection-dominated PDEs, Proc. AAAI Conf. Artif. Intell. 35 (1) (2021) 399–407.

[37] P. Solán-Fustero, J.L. Gracia, A. Navas-Montilla, P. García-Navarro, A POD-based ROM strategy for the prediction in time of advection-dominated problems, J. Comput. Phys. 471 (2022) 111672.

[38] B.G. Galerkin, Rods and plates. Series occurring in various questions concerning the elastic equilibrium of rods and plates, Eng. Bull. (Vestnik Inzhenerov) 19 (1915) 897–908.

[39] S.E. Ahmed, O. San, D.A. Bistrian, I.M. Navon, Sampling and resolution characteristics in reduced order models of shallow water equations: Intrusive vs nonintrusive, Internat. J. Numer. Methods Fluids 92 (8) (2020) 992–1036.

[40] P.J. Schmid, Dynamic mode decomposition of numerical and experimental data, J. Fluid Mech. 656 (2010) 5–28.

[41] M. Barrault, Y. Maday, N.C. Nguyen, A.T. Patera, An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations, C. R. Math. 339 (9) (2004) 667–672.

[42] S. Chaturantabut, D.C. Sorensen, Discrete Empirical Interpolation for nonlinear model reduction, in: Proceedings of the 48h IEEE Conference on Decision and Control (CDC) Held Jointly with 2009 28th Chinese Control Conference, 2009, pp. 4316–4321.

[43] R. Ştefănescu, I.M. Navon, POD/DEIM nonlinear model order reduction of an ADI implicit shallow water equations model, J. Comput. Phys. 237 (2013) 95–114.

[44] G.Y. Lee, K.C. Park, Y.H. Park, Reduced-order modeling via proper generalized decomposition for uncertainty quantification of frequency response functions, Comput. Methods Appl. Mech. Eng. 401 (2022) 115643.

[45] D. Goutaudier, L. Berthe, F. Chinesta, Proper Generalized Decomposition with time adaptive space separation for transient wave propagation problems in separable domains, Comput. Methods Appl. Mech. Eng. 380 (2021) 113755.

[46] C. Ghnatios, E. Hachem, A stabilized mixed formulation using the proper generalized decomposition for fluid problems, Comput. Methods Appl. Mech. Eng. 346 (2019) 769–787.

[47] M. Dehghan, M. Abbaszadeh, The use of proper orthogonal decomposition (POD) meshless RBF-FD technique to simulate the shallow water equations, J. Comput. Phys. 351 (2017) 478–510.

[48] J.W. Dossa-Fernandes, R.A. Kuche-Sanches, A. Barbarulo, A stabilized mixed space–time Proper Generalized Decomposition for the Navier–Stokes equations, Comput. Methods Appl. Mech. Eng. 386 (2021) 114102.

[49] S. Lorenzi, A. Cammi, L. Luzzi, G. Rozza, POD-Galerkin method for finite volume approximation of Navier–Stokes and RANS equations, Comput. Methods Appl. Mech. Eng. 311 (2016) 151–179.

[50] C.W. Rowley, T. Colonius, R.M. Murray, Model reduction for compressible flows using POD and Galerkin projection, Physica D 189 (1) (2004) 115–129.

[51] R. Ş.tefănescu, A. Sandu, I.M. Navon, Comparison of POD reduced order strategies for the nonlinear 2D shallow water equations, Int. J. Num. Met. Fluids 76 (8) (2014) 497–521.

[52] J.-M. Zokagoa, A. Soulaïmani, A POD-based reduced-order model for free surface shallow water flows over real bathymetries for Monte-Carlo-type applications, Comput. Methods Appl. Mech. Eng. 221–222 (2012) 1–23.

[53] J.-M. Zokagoa, A. Soulaïmani, Low-order modelling of shallow water equations for sensitivity analysis using proper orthogonal decomposition, Int. J. Comput. Fluid Dyn. 26 (5) (2012) 275–295.

[54] M. Milano, P. Koumoutsakos, Neural network modeling for near wall turbulent flow, J. Comput. Phys. 182 (1) (2002) 1–26.

[55] N. Peters, A. Wissink, J. Ekaterinaris, Machine learning-based surrogate modeling approaches for fixed-wing store separation, Aeros. Sci. Tech. 133 (2023) 108150.

[56] S. Fresca, A. Manzoni, POD-DL-ROM: Enhancing deep learning-based reduced order models for nonlinear parametrized PDEs by proper orthogonal decomposition, Comput. Methods Appl. Mech. Eng. 388 (2022) 114181.

[57] J. Yu, L. Lu, X. Meng, G.E. Karniadakis, Gradient-enhanced physics-informed neural networks for forward and inverse PDE problems, Comput. Methods Appl. Mech. Eng. 393 (2022) 114823.

[58] W.L. IJzerman, E. van Groesen, Low-dimensional model for vortex merging in the two-dimensional temporal mixing layer, Eur. J. Mech. - B/Fluids 20 (6) (2001) 821–840.

[59] S. Ingimarson, L.G. Rebholz, T. Iliescu, Full and reduced order model consistency of the nonlinearity discretization in incompressible flows, Comput. Methods Appl. Mech. Eng. 401 (2022) 115620.

[60] B. García-Archilla, J. Novo, S. Rubino, On the influence of the nonlinear term in the numerical approximation of Incompressible flows by means of proper orthogonal decomposition methods, Comput. Methods Appl. Mech. Eng. 405 (2023) 115866.

[61] R. Courant, K. Friedrichs, H. Lewy, On the partial difference equations of mathematical physics, IBM J. Res. Dev. 11 (2) (1967) 215–234.

[62] E.F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction, Springer Berlin Heidelberg, 2009.

[63] J. Burguete, P. García-Navarro, Improving simple explicit methods for unsteady open channel and river flow, Internat. J. Numer. Methods Fluids 45 (2) (2004) 125–156.

[64] P. Brufau, M.E. Vázquez-Cendón, P. García-Navarro, A numerical model for the flooding and drying of irregular domains, Internat. J. Numer. Methods Fluids 39 (3) (2002) 247–275.

[65] M. Gosses, W. Nowak, T. Wöhling, Explicit treatment for Dirichlet, Neumann and Cauchy boundary conditions in POD-based reduction of groundwater models, Adv. Water Resour. 115 (2018) 160–171.

[66] E.N. Karatzas, G. Stabile, L. Nouveau, G. Scovazzi, G. Rozza, A reduced basis approach for PDEs on parametrized geometries based on the shifted boundary finite element method and application to a Stokes flow, Comput. Methods Appl. Mech. Eng. 347 (2019) 568–587.

[67] X. Zeng, G. Stabile, E.N. Karatzas, G. Scovazzi, G. Rozza, Embedded domain reduced basis models for the shallow water hyperbolic equations with the shifted boundary method, Comput. Methods Appl. Mech. Eng. 398 (2022) 115143.