IOP Conference Series: Earth and Environmental Science

PAPER • **OPEN ACCESS**

A POD-based reduced order model applied to 1D shallow water equations

To cite this article: Pablo Solán-Fustero *et al* 2023 *IOP Conf. Ser.: Earth Environ. Sci.* **1136** 012036

A POD-based reduced order model applied to 1D shallow water equations

Pablo Solán-Fustero[1], **José Luis Gracia**[2], **Adrián Navas-Montilla**[1] **and Pilar García-Navarro**[1]

[1] Fluid Dynamic Technology, I3A, University of Zaragoza, Zaragoza, 50009, Spain
[2] IUMA and Department of Applied Mathematics, University of Zaragoza, Spain

psolfus@unizar.es

Abstract. Many environmental problems involving free surface flow can be solved using the shallow water equations (SWE) often involving high computational costs due to the large spatial and temporal scales of the events. In recent times, reduced order models (ROM) techniques are increasingly used to improve the computational efficiency of simulation models. The Proper Orthogonal Decomposition (POD) method provides an orthogonal basis for representing a given set of data and constructing the ROM by means of the method of snapshots. In this work, a POD-based intrusive ROM strategy is applied to the 1D SWE. The main goal of this work is to build a simulation model able to reproduce realistic scenarios. We analyse the computational improvement and the accuracy of the ROM results with respect to those of the full-order model (FOM).

1. Introduction

The shallow water equations (SWE) are widely used as mathematical model to represent the time evolution of free surface flows in channels. The SWE can be solved computationally following several numerical methods [1-3], here called full order models (FOM) that may involve high computational costs in realistic cases, thus requiring the use of mathematical tools to speed up the computations.

Among the various tools currently developed, reduced order models (ROM) based on proper orthogonal decomposition (POD) [4] allow the resolution of partial differential equations more efficiently than FOM and with little loss of accuracy [5-7]. This requires the proper interval decomposition (PID) method to structure the snapshot method in different time windows [8,9]. The ROM strategy has a first stage, off-line part, in which the FOM is solved. The solutions obtained by the FOM are used to train the ROM and then solve it (on-line part). In this work, two FOM are used: one based on Lax-Friedrichs numerical flux due to its simplicity [2] and another following Roe's method due to its robustness and good performance [1]. They are followed by an intrusive ROM using the Galerkin method [10]. Due to the non-linear character of the equations, it is necessary to make use of time averages of the variables of interest following [11].

With this objective, we first compare the results obtained by two ROMs developed from the Lax-Friedrichs FOM, one standard (version 1) and the other with averages of the water velocity (version 2). Then the time averaging approach is applied to the Roe's method (version 3).

The second objective of this work is to determine the best set of ROM parameters in terms of accuracy and CPU time. This will be done by measuring the errors of the computed solutions compared to the exact solution [12] and the CPU times required to obtain them.

2. Mathematical model and numerical method

As mentioned in the introduction, the cases considered in this paper involve the 1D SWE in the framework of a frictionless channel of rectangular cross-section and constant unit width [1]

$$\frac{\partial}{\partial t}\begin{pmatrix}h\\q\end{pmatrix}+\frac{\partial}{\partial x}\begin{pmatrix}q\\q^2/h+gh^2/2\end{pmatrix}=\begin{pmatrix}0\\0\end{pmatrix}, \tag{1}$$

where h is the water depth and $q = hu$ is the water discharge per unit width, with u, the cross-sectional average water velocity; g is the gravitational acceleration.

Regarding the FOM numerical method, the computational domain is discretized by means of N_x volume cells of uniform length Δx and the positions of the centre and left and right interfaces of the i-th cell are x_i, $x_{i-1/2}$ and $x_{i+1/2}$, respectively, with $i = 1, \dots, N_x$. The time step $\Delta t = t^{n+1} - t^n$ is selected dynamically using the Courant-Friedrichs-Lewy (CFL) condition [13].

Two different FOMs are used to solve the SWE system (1) using Godunov's scheme

$$\frac{\boldsymbol{U}_i^{n+1}-\boldsymbol{U}_i^n}{\Delta t}+\frac{\boldsymbol{F}_{i+1/2}^{n,*}-\boldsymbol{F}_{i-1/2}^{n,*}}{\Delta x}=\begin{pmatrix}0\\0\end{pmatrix}, \qquad i=1,\dots,N_x, \tag{2}$$

$$\boldsymbol{U}_i^n=\begin{pmatrix}h_i^n\\q_i^n\end{pmatrix}, \quad \boldsymbol{F}(\boldsymbol{U}_i^n)=\begin{pmatrix}q_i^n\\ \frac{(q_i^n)^2}{h_i^n}+\frac{1}{2}g(h_i^n)^2\end{pmatrix}, \tag{3}$$

where $h_i^n \approx h(x_i, t^n)$, $u_i^n \approx u(x_i, t^n)$ and $q_i^n \approx q(x_i, t^n)$ are the cell average values of the water depth, velocity and discharge over the cell $(x_{i-1/2}, x_{i+1/2})$.

In the first case, the numerical fluxes are given by the Lax-Friedrichs method [2]

$$\boldsymbol{F}_{i+1/2}^{n,*}=\frac{1}{2}[\boldsymbol{F}(\boldsymbol{U}_{i+1}^n)+\boldsymbol{F}(\boldsymbol{U}_i^n)]-\frac{1}{2}\nu\frac{\Delta x}{\Delta t}(\boldsymbol{U}_{i+1}^n-\boldsymbol{U}_i^n), \tag{4}$$

with $CFL \leq \nu \leq 1$.

The second FOM is formulated using Roe's numerical flux

$$\boldsymbol{F}_{i\pm1/2}^{n,*}=\pm\sum_{j=1}^{2}\left(\tilde{\lambda}_j^{\mp}\tilde{\alpha}_j\tilde{\boldsymbol{e}}_j\right)_{i\pm1/2}^n, \qquad i=1,\dots,N_x, \tag{5}$$

where $\left(\tilde{\lambda}_j^{\mp}\right)_{i\pm1/2}^n=\frac{1}{2}\left(\tilde{\lambda}_j\mp\left|\tilde{\lambda}_j\right|\right)_{i\pm1/2}^n, j=1,2$, and where

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

$$\begin{cases}(\tilde{\alpha}_1)_{i+1/2}^n=\dfrac{\delta h_{i+1/2}^n\left(\tilde{\lambda}_2\right)_{i+1/2}^n-\delta q_{i+1/2}^n}{\left(\tilde{\lambda}_2\right)_{i+1/2}^n-\left(\tilde{\lambda}_1\right)_{i+1/2}^n}\\ (\tilde{\alpha}_2)_{i+1/2}^n=\dfrac{\delta q_{i+1/2}^n-\delta h_{i+1/2}^n\left(\tilde{\lambda}_1\right)_{i+1/2}^n}{\left(\tilde{\lambda}_2\right)_{i+1/2}^n-\left(\tilde{\lambda}_1\right)_{i+1/2}^n}\end{cases}, \tag{6}$$

with $\delta h_{i+1/2}^n = h_{i+1}^n - h_i^n$ and $\delta q_{i+1/2}^n = q_{i+1}^n - q_i^n$, and the average velocities

$$\tilde{c}_{i+1/2}^n=\sqrt{g\frac{1}{2}(h_{i+1}^n+h_i^n)}, \quad \tilde{u}_{i+1/2}^n=\frac{q_{i+1}^n\sqrt{h_{i+1}^n}+q_i^n\sqrt{h_i^n}}{\sqrt{h_{i+1}^n h_i^n}\left(\sqrt{h_{i+1}^n}+\sqrt{h_i^n}\right)}. \tag{7}$$

3. Reduced-order model strategy

The POD-based ROM in this work is based on the snapshot method [14], which consists of the computation of a set of N_t time numerical solutions of SWE, (h_i^n, u_i^n), also called snapshots, as numerical approximations to h and u using the FOM. The snapshots are used to construct the snapshot matrices $M_h = (\boldsymbol{h}^1, \ \cdots, \ \boldsymbol{h}^{N_T}), M_u = (\boldsymbol{u}^1, \ \cdots, \ \boldsymbol{u}^{N_T})$ and $M_q = (\boldsymbol{q}^1, \ \cdots, \ \boldsymbol{q}^{N_T})$. The POD basis of functions is computed by applying the singular value decomposition of these matrices. The Galerkin method [10] and these bases are used to reconstruct the numerical solutions

$$h_i^n \approx \sum_{k=1}^{N_{POD}} \hat{h}_k^n \phi_{i,k}, \quad u_i^n \approx \sum_{k=1}^{N_{POD}} \hat{u}_k^n \varphi_{i,k}, \quad q_i^n \approx \sum_{k=1}^{N_{POD}} \hat{q}_k^n \Phi_{i,k}, \tag{8}$$

where $\phi_{i,k}$, $\varphi_{i,k}$ and $\Phi_{i,k}$ are the functions of the basis of each variable, being the number of POD modes $N_{POD} \ll N_x$.

The ROM of (1) is obtained by: i) introducing the Galerkin method (8) into the FOM (2) or (5); ii) multiplying each resulting equation by $\phi_{i,k}$ and $\Phi_{i,k}$, respectively; and iii) summing up over the cells. This procedure meets difficulties when the FOM scheme is nonlinear, as in the case of Roe's scheme. Zokagoa and Soulaïmani in [11] propose to use time averages of some variables where appropriate. Before applying this approach, the performance of two ROMs developed from the same Lax-Friedrichs FOM with and without time averages of$u$are compared.

With the aim of improving the resolution of non-linear problems, Zokagoa and Soulaïmani in [11] make use of the PID method, originally introduced by [8]. Following this method, the total simulation time T is partitioned into N_w non-overlapping time windows $[0, t_1] \cup [t_2, t_3] \cup \cdots \cup \left[t_{N_t-1}, t_{N_t} = T\right]$, so that as many snapshot matrices (and POD bases) are generated as there are time windows $M_{h_w} = (\boldsymbol{h}^{1_w}, \ \cdots, \ \boldsymbol{h}^{N_{T_w}}), M_{u_w} = (\boldsymbol{u}^{1_w}, \ \cdots, \ \boldsymbol{u}^{N_{T_w}})$ and $M_{q_w} = (\boldsymbol{q}^{1_w}, \ \cdots, \ \boldsymbol{q}^{N_{T_w}})$, with $w = 1, \dots, N_w$. Hereafter the reference to time windows in the equations is omitted for the sake of clarity.

The vector formulations of three versions of the ROM of SWE are presented below. The first one (version 1) is based on the Lax-Friedrichs method without time averaging of u

$$\widehat{\boldsymbol{h}}^{n+1} = \widehat{\boldsymbol{h}}^n - \frac{\Delta t}{2\Delta x} A \widehat{\boldsymbol{q}}^n + \frac{\nu}{2} B \widehat{\boldsymbol{h}}^n, \quad \widehat{\boldsymbol{q}}^{n+1} = \widehat{\boldsymbol{q}}^n - \frac{\Delta t}{2\Delta x} (\widehat{\boldsymbol{q}}^n)^T C \widehat{\boldsymbol{u}}^n - \frac{g\Delta t}{4\Delta x} g\left(\widehat{\boldsymbol{h}}^n\right)^T D \widehat{\boldsymbol{h}}^n + \frac{\nu}{2} E \widehat{\boldsymbol{q}}^n, \tag{9}$$

where the matrices, considering free boundary conditions (BCs), are

$$\begin{aligned}
A(k,p) &= \left(\Phi_{2,k} - \Phi_{1,k}\right)\phi_{1,p} + \sum_{i=2}^{N_x-1} \left(\Phi_{i+1,k} - \Phi_{i-1,k}\right)\phi_{i,p} + \left(\Phi_{N_x,k} - \Phi_{N_x-1,k}\right)\phi_{N_x,p}, \\
B(k,p) &= \left(\phi_{2,k} - \phi_{1,k}\right)\phi_{1,p} + \sum_{i=2}^{N_x-1} \left(\phi_{i+1,k} - 2\phi_{i,k} + \phi_{i-1,k}\right)\phi_{i,p} + \left(\phi_{N_x,k} - \phi_{N_x-1,k}\right)\phi_{N_x,p}, \\
C(q,k,p) &= \left(\Phi_{2,k}\varphi_{2,q} - \Phi_{1,k}\varphi_{1,q}\right)\Phi_{1,p} + \sum_{i=2}^{N_x-1} \left(\Phi_{i+1,k}\varphi_{i+1,q} - \Phi_{i-1,k}\varphi_{i-1,q}\right)\Phi_{i,p} + \\
&\quad \left(\Phi_{N_x,k}\varphi_{N_x,q} - \Phi_{N_x-1,k}\varphi_{N_x-1,q}\right)\Phi_{N_x,p}, \\
D(q,k,p) &= \left(\phi_{2,k}\phi_{2,q} - \phi_{1,k}\phi_{1,q}\right)\Phi_{1,p} + \sum_{i=2}^{N_x-1} \left(\phi_{i+1,k}\phi_{i+1,q} - \phi_{i-1,k}\phi_{i-1,q}\right)\Phi_{i,p} + \\
&\quad \left(\phi_{N_x,k}\phi_{N_x,q} - \phi_{N_x-1,k}\phi_{N_x-1,q}\right)\Phi_{N_x,p}, \\
E(k,p) &= \left(\Phi_{2,k} - \Phi_{1,k}\right)\Phi_{1,p} + \sum_{i=2}^{N_x-1} \left(\Phi_{i+1,k} - 2\Phi_{i,k} + \Phi_{i-1,k}\right)\Phi_{i,p} + \left(\Phi_{N_x,k} - \Phi_{N_x-1,k}\right)\Phi_{N_x,p}.
\end{aligned} \tag{10}$$

The second one (version 2) is based on the same method, but considers time averages of the water velocity u, so that the Galerkin method is only applied to the water depth h and the water discharge q

$$\widehat{\boldsymbol{h}}^{n+1} = \widehat{\boldsymbol{h}}^{n} - \frac{\Delta t}{2\Delta x} A \widehat{\boldsymbol{q}}^{n} + \frac{\nu}{2} B \widehat{\boldsymbol{h}}^{n}, \quad \widehat{\boldsymbol{q}}^{n+1} = \widehat{\boldsymbol{q}}^{n} - \frac{\Delta t}{2\Delta x} C \widehat{\boldsymbol{h}}^{n} - \frac{g\Delta t}{4\Delta x} g\left(\widehat{\boldsymbol{h}}^{n}\right)^{T} D \widehat{\boldsymbol{h}}^{n} + \frac{\nu}{2} E \widehat{\boldsymbol{q}}^{n}, \tag{11}$$

where the matrices are the same as in version 1, except for

$$\begin{aligned} C(q,k,p) = {} & \left(\phi_{2,k}(\bar{u}_2)^2 - \phi_{1,k}(\bar{u}_1)^2\right)\Phi_{1,p} + \sum\nolimits_{i=2}^{N_x-1} \left(\phi_{i+1,k}(\bar{u}_{i+1})^2 - \phi_{i-1,k}(\bar{u}_{i-1})^2\right)\Phi_{i,p} \\ & + \left(\phi_{N_x,k}\left(\bar{u}_{N_x}\right)^2 \right. \\ & \left. - \phi_{N_x-1,k}\left(\bar{u}_{N_x-1}\right)^2\right)\Phi_{N_x,p}. \end{aligned} \tag{12}$$

where $\bar{u}_{i+1} = \sum_{n=1}^{N_t} u_i^n / N_t$, with $i = 1, \dots, N_x$.

The third version (version 3) arises from Roe's method and considers time averages of water velocity u and the water depth h in denominators or square roots of (7)

$$\widehat{\boldsymbol{h}}^{n+1} = \widehat{\boldsymbol{h}}^{n} + \frac{\Delta t}{4\Delta x} A \widehat{\boldsymbol{h}}^{n} + \frac{\Delta t}{4\Delta x} B \widehat{\boldsymbol{q}}^{n}, \quad \widehat{\boldsymbol{q}}^{n+1} = \widehat{\boldsymbol{q}}^{n} - \frac{\Delta t}{4\Delta x} C \widehat{\boldsymbol{h}}^{n} + \frac{\Delta t}{4\Delta x} D \widehat{\boldsymbol{q}}^{n}, \tag{13}$$

where the matrices, considering free BC, are

$$\begin{aligned} A(k,p) = {} & \left(\phi_{2,k} - \phi_{1,k}\right)\bar{a}_{3/2}\phi_{1,p} + \sum_{i=2}^{N_x-1} \left[\left(\phi_{i+1,k} - \phi_{i,k}\right)\bar{a}_{i+1/2} - \left(\phi_{i,k} - \phi_{i-1,k}\right)\bar{a}_{i-1/2}\right]\phi_{i,p} \\ & + \left(\phi_{N_x,k} - \phi_{N_x-1,k}\right)\bar{a}_{N_x-1/2}\phi_{N_x,p}, \\ B(k,p) = {} & \left(\Phi_{2,k} - \Phi_{1,k}\right)\bar{b}_{3/2}\phi_{1,p} + \sum_{i=2}^{N_x-1} \left[\left(\Phi_{i+1,k} - \Phi_{i,k}\right)\bar{b}_{i+1/2} + \left(\Phi_{i,k} - \Phi_{i-1,k}\right)\bar{b}_{i-1/2}\right]\phi_{i,p} \\ & + \left(\Phi_{N_x,k} - \Phi_{N_x-1,k}\right)\bar{b}_{N_x-1/2}\phi_{N_x,p}, \\ C(k,p) = {} & \left(\phi_{2,k} - \phi_{1,k}\right)\bar{e}_{3/2}\Phi_{1,p} + \sum_{i=2}^{N_x-1} \left[\left(\phi_{i+1,k} - \phi_{i,k}\right)\bar{e}_{i+1/2} + \left(\phi_{i,k} - \phi_{i-1,k}\right)\bar{e}_{i-1/2}\right]\Phi_{i,p} \\ & + \left(\phi_{N_x,k} - \phi_{N_x-1,k}\right)\bar{e}_{N_x-1/2}\Phi_{N_x,p}, \\ D(k,p) = {} & \left(\Phi_{2,k} - \Phi_{1,k}\right)\bar{d}_{3/2}\Phi_{1,p} + \sum_{i=2}^{N_x-1} \left[\left(\Phi_{i+1,k} - \Phi_{i,k}\right)\bar{d}_{i+1/2} + \left(\Phi_{i,k} - \Phi_{i-1,k}\right)\bar{d}_{i-1/2}\right]\Phi_{i,p} \\ & + \left(\Phi_{N_x,k} - \Phi_{N_x-1,k}\right)\bar{d}_{N_x-1/2}\Phi_{N_x,p}, \end{aligned} \tag{14}$$

with

$$\begin{aligned} \bar{a}_{i+1/2} &= \left(\frac{|\bar{\lambda}_1|\bar{\lambda}_2 - |\bar{\lambda}_2|\bar{\lambda}_1}{\bar{c}}\right)_{i+1/2}, & \bar{b}_{i\pm1/2} &= \left(\frac{\bar{\lambda}_1^{\mp} - \bar{\lambda}_2^{\mp}}{\bar{c}}\right)_{i\pm1/2}, \\ \bar{e}_{i\pm1/2} &= \left(\frac{\bar{\lambda}_1\bar{\lambda}_2}{\bar{c}}\right)_{i\pm1/2} \left(\bar{\lambda}_1^{\mp} - \bar{\lambda}_2^{\mp}\right)_{i\pm1/2}, & \bar{d}_{i\pm1/2} &= \left(\frac{\bar{\lambda}_1\bar{\lambda}_1^{\mp} - \bar{\lambda}_2\bar{\lambda}_2^{\mp}}{\bar{c}}\right)_{i\pm1/2}, \end{aligned} \tag{15}$$

where $(\bar{\lambda}_1)_{i+1/2}$, $(\bar{\lambda}_2)_{i+1/2}$, $(\bar{\lambda}_1^{\mp})_{i+1/2}$, $(\bar{\lambda}_2^{\mp})_{i+1/2}$ and $\bar{c}_{i+1/2}$ are computed from the time averages $\bar{u}_i$ and $\bar{c}_i$ following (7).

4. Numerical results

4.1. Case 1

Consider the 1D shallow water equations (1), where the spatial domain is $[0, L]$, where $L = 12$, and the final time is $T = 1.02$.

The initial conditions (ICs) are defined as a dam break time as

$$h(x,0) = \begin{cases} 2, & if\ x \leq 6 \\ 1, & if\ x > 6 \end{cases}, \quad u(x,0) = 0, \quad 0 \leq x \leq L, \tag{16}$$

given in m and m/s, respectively; and free BCs are considered.

In this case, the two versions of the ROM, with and without time averages, developed from the Lax-Friedrichs method are compared in terms of accuracy and CPU time. The CFL number is 0.9 and $\nu = 0.9$.

The accuracy is measured by computing the error of h and q at the final time in the L_1 norm with respect to the exact solution [12]. They are denoted by $L_1(h)$ and $L_1(q)$, respectively. The CPU time required by each ROM is also measured to study their efficiency.

Following the PID strategy approach, the construction of the snapshot matrices has been divided into time windows with the same number of snapshots. Table 1 shows the number of time windows N_w arranged according to the number of snapshots per window N_s and per mesh refinement N_x. Each N_x corresponds to a value of N_t according to the CFL condition.

Table 1. Number of time windows $N_w = N_t/N_s$ for different values of N_t and N_s.

N_x	N_t	N_s			
		16	**8**	**4**	**2**
100	48	3	6	12	24
203	96	6	12	24	48
409	192	12	24	48	96
822	384	24	48	96	192

First, the influence of N_w and N_{POD} on the computation of the solution is studied. The goal is to find the optimal set of values of these ROM parameters.

Figures 1-4 show the errors in the L_1 norm of the solutions computed with version 1 (left) and version 2 (right) of the ROM in terms of h and q as a function of N_x.

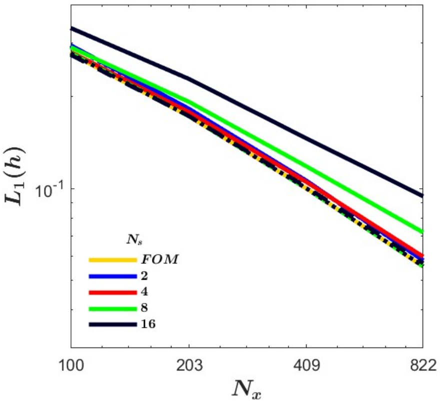

Figure 1. $L_1(h)$ of ROM v.1.

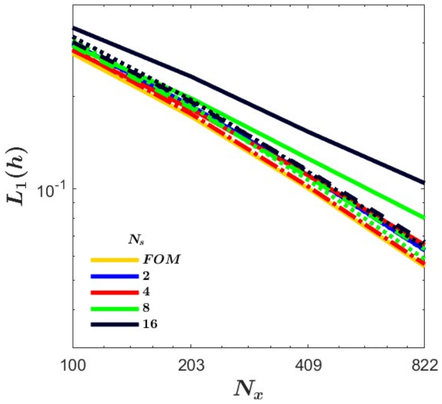

Figure 2. $L_1(h)$ of ROM v.2.

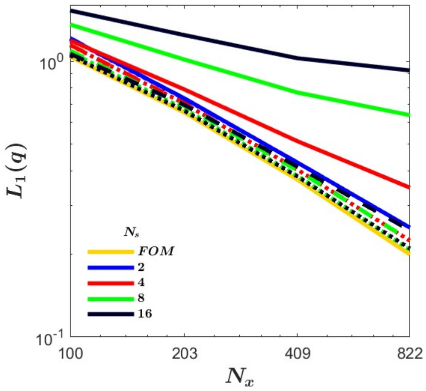

Figure 3. $L_1(q)$ of ROM v.1.

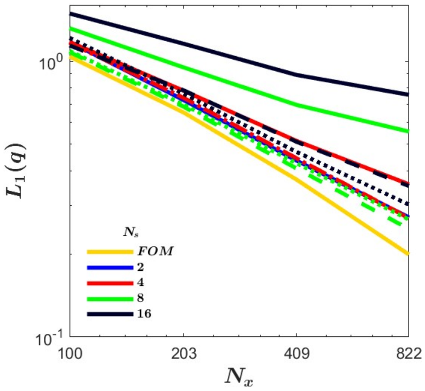

Figure 4. $L_1(q)$ of ROM v.2.

The yellow line represents the errors of the FOM solutions with respect to the exact solution. The rest of the colours represent the errors of the ROM solutions according to the number of snapshots in each time window N_s. The line type indicates the number of modes of the ROM: $N_{POD} = 2$ (solid line), 5 (dashed line) and 10 (dotted line).

As can be seen by comparing the figures of ROM version 1 with those of version 2, the ROM version 1 solutions are slightly more accurate for both variables h and q. In both cases, all ROM solutions are equally or less accurate than those of the FOM. On the other hand, the solutions obtained by solving only 2 POD modes (solid lines) are less accurate. The rest of the solutions with $N_{POD} = 5$ and 10 presented are slightly more accurate in ROM version 1.

Figures 5 and 6 show the CPU times required by versions 1 (left) and 2 (right) of the ROM. ROM version 1 results are clustered according to the number of modes solved N_{POD} regardless of the number of time windows, with $N_{POD} = 2$ (solid lines) being the best in terms of CPU time, achieving an order of magnitude improvement over FOM CPU times. However, the results of ROM version 2 do not show such clusters and are mixed without any pattern related to N_{POD} or N_s. In this case, the CPU time improvements reach two orders of magnitude. Nevertheless, the above study shows that the fastest results are the least accurate, so the CPU times against errors are plotted to obtain the optimal values of N_{POD} and N_w.

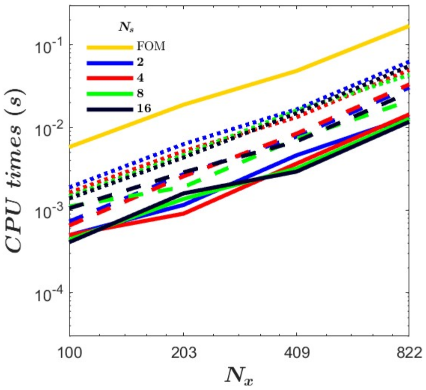

Figure 5. CPU times of ROM v.1.

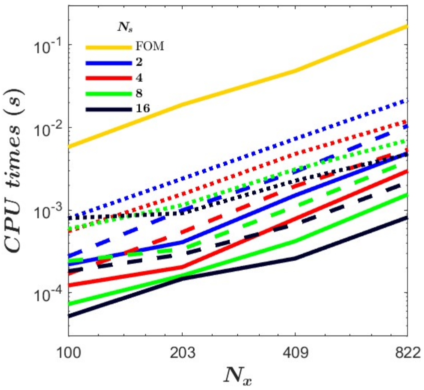

Figure 6. CPU times of ROM v.2.

Figures 7-10 show the errors in the norm against CPU times of versions 1 (left) and 2 (right) of the ROM. In general, the CPU times of ROM version 2 are better than those of version 1; otherwise, the errors obtained by both versions are very similar.

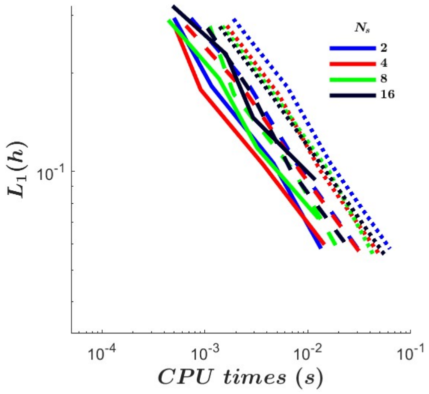

Figure 7. $L_1(h)$ vs CPU times of ROM v.1.

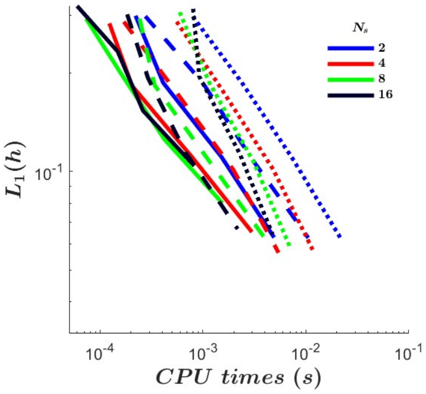

Figure 8. $L_1(h)$ vs CPU times of ROM v.2.

The best set of ROM parameter values is $N_S = 4$ and $N_{POD} = 5$, as it obtains very accurate final time solutions with the ROM, as can be seen in Figures 11 and 12, where the FOM and the ROM version 2 solutions at the final time T are represented together with the exact solution and the IC. Other ROM parameter values obtain results with inaccurate wave fronts.

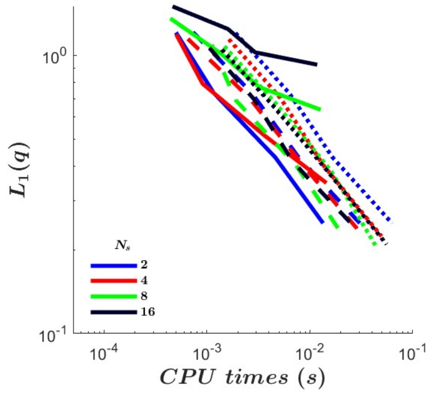

Figure 9. $L_1(q)$ vs CPU times of ROM v.1.

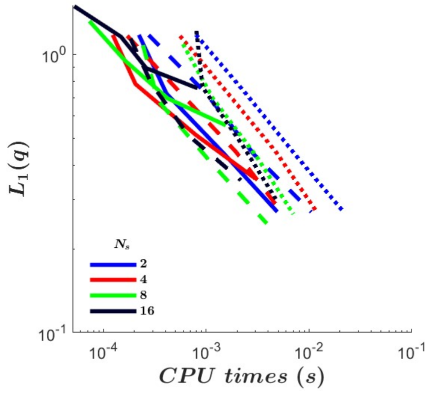

Figure 10. $L_1(q)$ vs CPU times of ROM v.2.

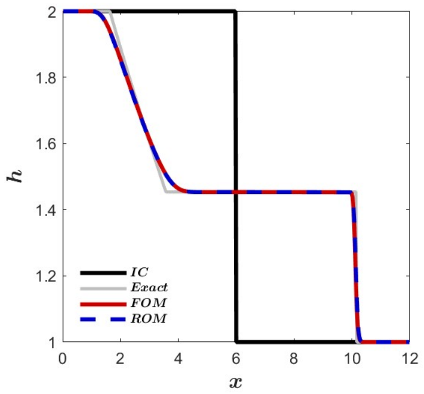

Figure 11. h computed with ROM v.2.

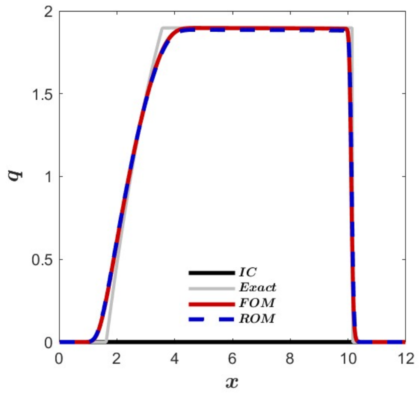

Figure 12. q computed with ROM v.2.

In conclusion, it can be said that ROM version 2 is more efficient with hardly any loss of accuracy despite the use of time averages of u.

4.2. Case 2

Once the validity of the time averaging approach has been checked, the same problem as in Case 1 is solved with the version 3 ROM developed from the Roe's method.

Figures 13 and 14 show the errors in the L_1 norm of h (left) and q (right) as a function of N_x. First of all, it should be noted that the solution computed by the FOM following the upwind method is more accurate than the FOM of the Lax-Friedrichs method. This makes some solutions obtained by version 3 of the ROM even better than those of the Lax-Friedrichs FOM.

In terms of water depth, the accuracy of solutions obtained with version 3 of the ROM becomes worse when only 2 POD modes are solved. It can also be observed that when two snapshots per time window are solved, worse results are obtained than in the case of $N_s > 2$. Considering the water discharge, the errors show more diversity. Altogether, $N_s = 4$ with $N_{POD} = 5$ and 10 are the most accurate.

Figure 13 shows the CPU times required by version 3 of the ROM, which reach two orders of magnitude better than the CPU times of the FOM. The CPU times required by the FOM based on the upwind method are practically the same as those of the FOM based on the Lax-Friedrichs. Some sets of ROM parameter values reach two orders of magnitude improvement over the FOM.

In terms of the measured error in the water depth, it could be considered that the best values of the ROM parameters would be $N_s = 8$ or 16 with $N_{POD} = 5$, as can be seen in Figure 16. However, these values present worse values of the measured error in the flow rate, as can be seen in Figure 17. Considering both figures, it is concluded that, as in case 1, the best set of ROM parameter values are $N_s = 4$ and $N_{POD} = 5$, as these give very high accuracy solutions, as can be seen in Figures 18 and 19.

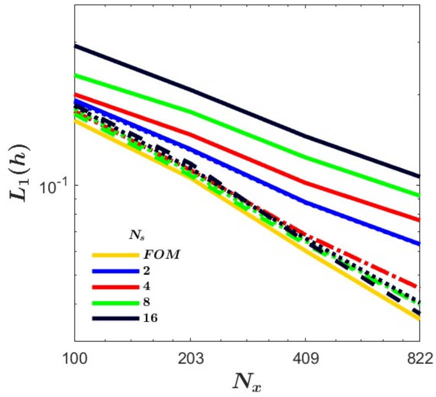

Figure 13. $L_1(h)$ of ROM v.3.

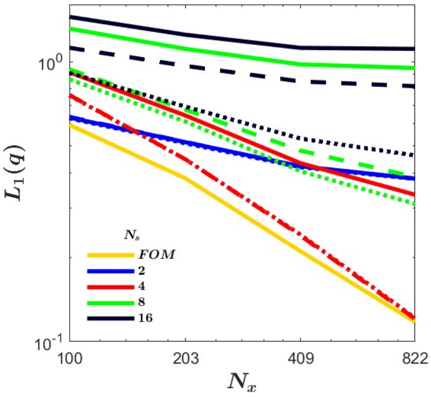

Figure 14. $L_1(q)$ of ROM v.3.

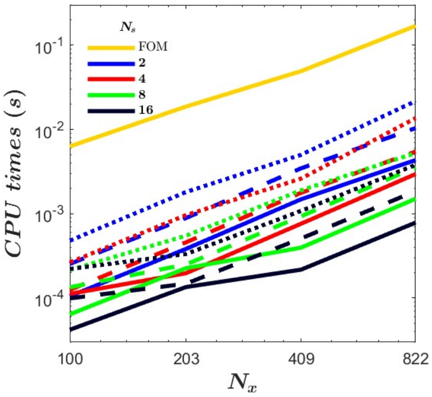

Figure 15. CPU times of ROM v.3.

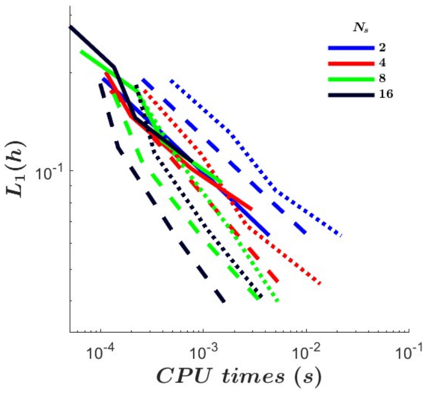

Figure 16. $L_1(h)$vs CPU times of ROM v.3

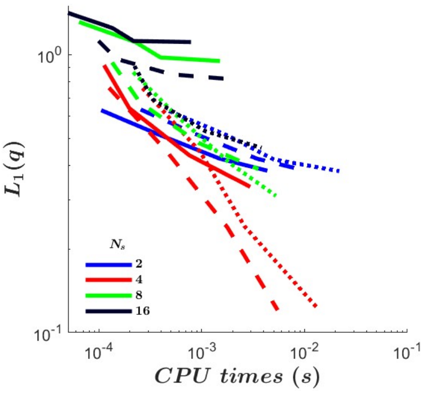

Figure 17. $L_1(q)$vs CPU times of ROM v.3.

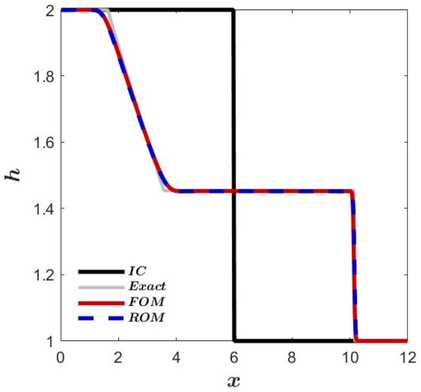

Figure 18. h of ROM v.3.

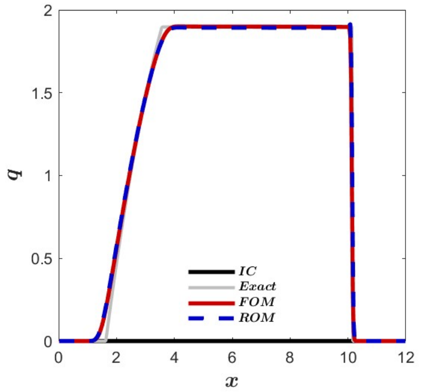

Figure 19. q of ROM v.3.

Table 2 shows the speed-ups of each of the ROM versions with respect to FOM. As can be seen, the versions with time averages 2 and 3 present higher speed-ups than version 1, which exceed the order of magnitude.

Table 2. Speed-ups of each version of the ROM.

N_x	100	203	409	822
ROM v.1	9.69	7.34	5.99	5.20
ROM v.2	37.18	34.77	25.86	31.85
ROM v.3	50.55	41.22	28.56	31.58

5. Conclusions

In this work a dam break problem has been solved with the 1D SWE using the ROM strategy. For this purpose, two numerical methods (FOM) of discretisation of the mathematical model have been considered, one based on the Lax-Friedrichs method and the other based on Roe's method. From these, three reduced order models have been studied.

With the two ROMs developed from the Lax-Friedrichs-based FOM, the validity of the time-averaging approach proposed in [9] has been tested. The results obtained show that the time-averaged ROM produces accurate solutions, as much as the standard ROM, and in a more efficient way. Furthermore, this time averaging approach has been applied to the development of the ROM from the FOM based on the Roe's method, obtaining satisfactory results.

The performance of the different ROMs has been studied against the variation of their own parameters, number of time windows (following the PID method) and number of POD modes solved, as well as the number of cells in the spatial domain. It can be concluded that the best values to obtain accurate and efficient results from the ROM are $N_s = 4$ and $N_{POD} = 5$, as they achieve good error rates and have low CPU times.

Acknowledgments
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] García-Navarro P and Vázquez-Cendón ME 2000 On numerical treatment of the source terms in the shallow water equations *Comput. Fluids* **29**(8) 951–979 https://doi.org/10.1016/S0045-7930(99)00038-9

[2] Burguete J and García-Navarro P 2004 Improving simple explicit methods for unsteady open channel and river flow *Int. J. Numer. Methods in Fluids* **45**(2) 125–156 https://doi.org/10.1002/fld.619

[3] Murillo J and Navas-Montilla A 2016 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. Water Resour.* **98** 70–96 https://doi.org/10.1016/j.advwatres. 2016. 10.019

[4] Lumley JL 1967 The structure of inhomogeneous turbulent flows *Atmospheric Turbulence and Radio Wave Propagation* ed AM Yaglom and VI Tartarsky (Moscow: Publishing House “Nauka”) pp 166–176

[5] Ahmed M and San O 2018 Stabilized principal interval decomposition method for model reduction of nonlinear convective systems with moving shocks *Comput. Appl. Math.* **37** 6870–6902 https://doi.org/10.1007/s40314-018-0718-z

[6] Ahmed SE, San O, Bistrian DA and Navon IM 2020 Sampling and resolution characteristics in reduced order models of shallow water equations: Intrusive vs nonintrusive *Int. J. Numer. Methods Fluids* **92**(8) 992–1036 https://doi.org/10.1002/fld.4815

[7] Akhtar I, Wang Z, Borggaard J and Iliescu T 2012 A new closure strategy for proper orthogonal decomposition reduced-order models *J. Comput. Nonlinear Dyn.* **7**(3) 034503 https://doi.org/10.1115/1.4005928

[8] IJzerman WL 2000 Signal Representation and Modeling of Spatial Structures in Fluids *PhD Thesis* University of Twente (Enschede, Netherlands) https://www.persistent-identifier.nl/urn:nbn:nl:ui:28-29660

[9] Borggaard J, Hay A and Pelletier D 2007 Interval-based reduced-order models for unsteady fluid flow *Int. J. Numer. Anal. Model.* **4**(3-4) 353–367 https://global-sci.org/intro/article_ detail/ijnam/866.html

[10] Galerkin BG 1915 Rods and plates. Series occurring in various questions concerning the elastic equilibrium of rods and plates *Engineers Bulletin* **19** 897–908

[11] Zokagoa JM and Soulaïmani A 2018 A POD-based reduced-order model for uncertainty analyses in shallow water flows *Int. J. Comput. Fluid Dynamics* **32**(6-7) 278–292 https://doi.org/10.1080/10618562.2018.1513496

[12] Stoker JJ 2011 *Water Waves: The Mathematical Theory with Applications* (Wiley Classics Library)

[13] Courant R, Friedrichs K and Lewy H 1928 Uber die partiellen Differenzengleichungen der mathematischen Physik *Math. Ann.* **100**(1) 32–74 https://doi.org/10.1007/BF01448839

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