Advanced Modeling and Simulation in Engineering Sciences

RESEARCH ARTICLE

Open Access

Reduced-order modeling for stochastic large-scale and time-dependent flow problems using deep spatial and temporal convolutional autoencoders

Azzedine Abdedou[1] and Azzeddine Soulaimani[1]*

*Correspondence: azzeddine.soulaimani@etsmtl.ca

[1] Department of Mechanical Engineering, Ecole de technologie superieure, 1100 Notre-Dame W., Montreal H3C 1K3, QC, Canada

Abstract

A non-intrusive reduced-order model based on convolutional autoencoders is proposed as a data-driven tool to build an efficient nonlinear reduced-order model for stochastic spatiotemporal large-scale flow problems. The objective is to perform accurate and rapid uncertainty analyses of the flow outputs of interest for which the input parameters are deemed uncertain. The data are constituted from a set of high-fidelity snapshots, collected using an inhouse high-fidelity flow solver, which correspond to a sample of the uncertain input parameters. The method uses a 1D-convolutional autoencoder to reduce the spatial dimension of the unstructured meshes used by the flow solver. Another convolutional autoencoder is used for the time compression. The encoded latent vectors, generated from the two compression levels, are then mapped to the input parameters using a regression-based multilayer perceptron. The proposed model allows for rapid predictions for unseen parameter values, allowing the output statistical moments to be computed efficiently. The accuracy of the proposed approach is compared to that of the linear reduced-order technique based on an artificial neural network through two benchmark tests (the one-dimensional Burgers and Stoker's solutions) and a hypothetical dam break flow problem, with an unstructured mesh and over a complex bathymetry river. The numerical results show that the proposed methods present strong predictive capabilities to accurately approximate the statistical moments of the outputs. In particular, the predicted statistical moments are oscillations-free, unlike those obtained with the traditional proper orthogonal decomposition method. The proposed reduction framework is simple to implement and can be applied to other parametric and time-dependent problems governed by partial differential equations, which are commonly encountered in many engineering and science problems.

Keywords: Uncertainty propagation, Reduced-order modeling, Deep learning, Convolutional autoencoders

Introduction

Computational science has made great gains in efficiency due to the increases in high-performance computational resources and the emergence of novel methods, with significant achievements in scientific and industrial areas. Most of the computational mechanics problems are described by time-dependent and parameterized nonlinear partial differential equations. Their discretization over fine spatial meshes and for a high number of time steps leads to the so-called high-fidelity solutions, whose evaluation renders the computational techniques prohibitively expensive when dealing with multi-resolutions, as in optimization and uncertainty quantification [1,2]. Reduced-order modeling (ROM) [3] has gained interest as a powerful methodology to reduce the cumbersome computational effort involved in high-fidelity solutions, while ensuring acceptable accuracy, particularly for situations that require real-time predictions.

One of the approaches most commonly adopted in reduced-order modeling is proper orthogonal decomposition (POD) [4,5], which allows an approximation of the outputs using a linear combination of a limited number of basis functions [6,7]. Many non-intrusive techniques have been proposed to compute the coefficients of a POD linear approximation using the data-driven concept without any modification of the governing equations [8–10]. Some of these methods are based on the stochastic framework and account for the variability stemming from the parametric domain by expressing the modal coefficients as a function of the stochastic basis functions, as in the polynomial chaos expansion method (POD-PCE) [11,12] and the B-splines Bézier-element method (POD-BSBEM) [13]. Recent studies have explored the combination of the POD basis function with artificial neural networks (ANN) to construct an efficient regression framework for linear reduced-order modeling for time-dependent problems by learning the mapping between the time-parameter inputs and the modal coefficients of the POD basis function [14–16]. Despite the wide adoption of linear reduced-order modeling in approximating the outputs for parametric and time-dependent problems, they still encounter difficulties in accurately capturing the dynamics of some complex physical problems (i.e., those with strong hyperbolic behavior or shock propagation waves) without considerably increasing the number of reduced basis functions and thus compromising the low dimensionality aspect [17,18].

To overcome the limitations of the techniques based on linear dimensionality reduction [19], recent methods based on nonlinear manifolds have gained interest in the field of dimensionality reduction. Some of these approaches are based on the recently developed algorithms of deep learning technology that are effective in learning sophisticated abstract features [20–23]. The autoencoder-based method (AE), a special type of neural network (NN) with two parts, the encoder and the decoder, trained jointly, has been successfully introduced as a nonlinear dimensionality reduction framework with an effective nonlinear relationship between variables [24,25]. Some limitations concerning classical autoencoders based on dense layers have been reported in the literature, due to the drastic increase in the number of trainable parameters when the dimension of the input dataset becomes large [19,26]. Convolutional autoencoders (CAEs) have emerged as an alternative to AEs in nonlinear compression manifolds. Their structure comprises several operations such as convolution-deconvolution, pooling-upsampling, and a multilayer perceptron layer (MLP) [27,28]. CAEs offer the possibility of sharing coefficients and local patch connections, allowing a significant reduction in the number of trainable parameters. It is also possible to impose conditions on the latent space as proposed in the β-variational autoencoder-CNN architecture [29] where the latent vectors are ranked based on their contribution to the construction and learned to be nearly non-orthogonal.

To make full use of temporal information, many sequence networks, including the recurrent neural network (RNN) [30] and its variants (e.g. LSTMs, GRUs, TCN, Transformer etc.) have been widely used. The idea behind recurrent neural networks (RNNs) is to apply the weight sharing operations at each time instant by involving the current state (previous history) as well as the value of the current input. In this way, a network can scale well (and rapidly) to sequences of different lengths. The key idea behind the popular recurrent network, long-short term memory, (LSTM) [31] is the so-called cell state, which helps to overcome the problems associated with the vanishing/exploding phenomena that are caused by the long-term dependencies within the network. Some configurations combine CAEs with LSTMs or TCNs [19,32–35] to provide a tool for time predictions over the latent space.

This paper proposes a non-intrusive reduced order modeling (NIROM) framework for parametrized and time-dependent flow problems. The main objective is to perform accurate and rapid uncertainty analyses of the outputs of fluid dynamic flows for which the input parameters are deemed uncertain. This data-driven approach relies on the attractive features of convolutional autoencoders (CAEs) to reduce the dimensionality of high-fidelity solutions collected from numerical solvers with large-scale meshes. The method, referred to as NIROM-CAEs, is based on two compression levels provided by 1D-autoencoders. The first 1D-encoder reduces the spatial dimension by encoding the input dataset along the spatial dimension to generate a latent space. A second 1D-encoder convolves the latent space along the temporal dimension to output the final spatiotemporal encoded latent vector, which is mapped to the input parameters values through a multilayer perceptron (MLP). Therefore, for a new set of unseen parameters, the online predictive stage allows a rapid and accurate reconstruction of the original spatiotemporal dynamics through the trained 1D decoders. The proposed time prediction method is simple to implement and does not suffer from the vanishing or the exploding phenomena caused by long-term dependencies. The framework is applied to a stochastic treatment of two benchmark cases with univariate meshes, and to a hypothetical dam break flow over a natural river with complex bathymetry and two-dimensional unstructured meshes, which may present a challenging task for convolution autoencoders. By arranging the data into a vector according to the nodes' numbering, a simple 1D-convolutinal autoencoder reveals sufficiently accurate results for the space compression step. The proposed method is compared with the more traditional approaches based on proper orthogonal decomposition. For solutions with strong gradients, the present method does not suffer from oscillations in the predicted statistical moments, unlike the POD-based methods. Therefore, the proposed framework offers interesting features in non-linear reduction modeling for physical problems with a high degree of complexity in their dynamics.

The paper is organized as follows: "Methodology" presents the fundamental framework of the proposed NIROM-CAEs, with detailed steps of the offline and online stages. Numerical test cases for assessing the performance and accuracy of the proposed method are provided in "Results and discussion", followed by a summary and concluding remarks in "Conclusion".

Methodology

This section describes the fundamental framework of the proposed non-intrusive reduced model based on 1D convolutional autoencoders for space and time dimensions. The method belongs to the data-driven approaches, where high-fidelity solutions to large-scale and time-dependent physical problems are obtained from numerical solvers and gathered for both time and parameter sequences to construct snapshot matrices.

Convolutional autoencoders (CAEs)

Convolutional autoencoders (CAEs) have gained interest as powerful nonlinear reduced-order modeling techniques with remarkable performances in the image recognition field. Convolutional layers have been introduced to overcome some of the limitations that classical autoencoders (AEs) based on dense layers may face when treating time-dependent problems with high dimensional inputs [19]. These layers are characterized by two properties: local connections and shared weights, thus allowing a feature map of the input and a limitation of the number of trainable parameters. Convolutional autoencoders have two distinct symmetrical parts: the first part, called the encoder ($\mathcal{F}_{enc}$), reduces the dimension of the input matrix by mapping with a latent space through a combination of successive convolutional, pooling, and fully connected layers; and the second part, the decoder ($\mathcal{F}_{dec}$), a combination of dense, upsampling, and convolutional layers, which maps the latent reduced-dimension vector to a larger-dimensional reconstruction of the input. A schematic representation of a 1D convolutional autoencoder architecture is presented in Fig. 1. It should be emphasized that the same structure of the 1D CAE is adopted in this study for the encoding and decoding processes for both space and time dimensions. Thus, the presentation of the framework of the proposed approach will concern mainly the 1D convolutional autoencoders.

A convolution layer represents a feature map where each unit in the layer is locally connected to a selected part of the previous layer via a kernel (filter) and an activation function. This operation allows the most dominant features from the input data to be extracted by applying a filter that moves with the vertical stride in the case of 1D convolution, as shown in Fig. 2. In this figure, a schematic representation of the 1D-convolution operation for a given layer n is detailed for both space and time directions. A mathematical formulation of a typical 1D-convolutional operation denoted by the symbol ($*$), shown in Figs. 1 and 2, can be expressed as follows [27,36]:

$$h_i^{\ell} = \sigma\left(\mathbf{H}^{\ell-1} * \mathbf{f}_i^{\ell} + b_i^{\ell}\right), \tag{1}$$

where $h_i^{\ell} \in \mathbb{R}^{D_\ell \times 1}$ denotes the i^{th} feature of the ℓ^{th} layer, σ is a nonlinear activation function, $\mathbf{H}^{\ell-1} = \left[h_1^{\ell-1}, h_2^{\ell-1}, \ldots, h_{N_{f_{\ell-1}}}^{\ell-1}\right]$ stands for the convolution layer $\ell - 1$, $\mathbf{f}_i^{\ell}$ is the kernel of the layer ℓ, and b_i^{ℓ} the bias parameter, with $i \in (1, N_{f_\ell})$ and $\ell \in (1, n)$. The number of feature column vectors in each layer N_{f_ℓ} corresponds to the number of kernels represented by different colors in Fig. 2, and the total number of layers n defines the depth of the convolution neural network (CNN). In addition to the convolutional operations, pooling layers (in this case, a max-pooling) are also integrated after each convolution layer to reduce the dimension of the convolved features by a fixed factor defined as a kernel size of the max-pooling layer, thus allowing the most dominant features of the local domain

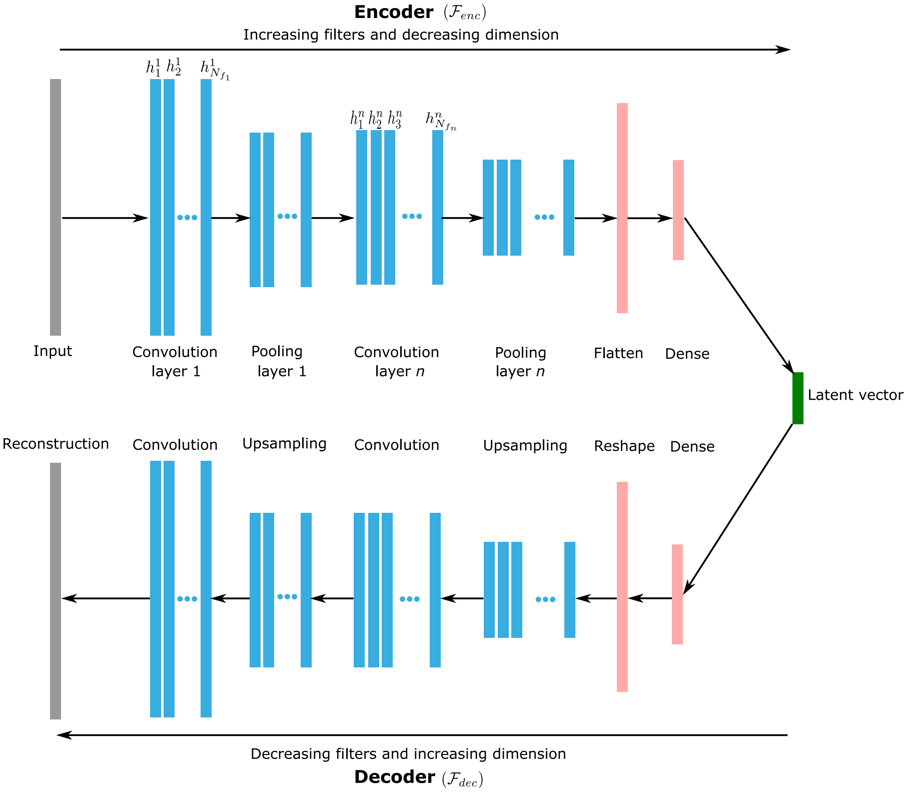

Encoder ($\mathcal{F}_{enc}$)
Increasing filters and decreasing dimension
h_1^1 h_2^1 $h_{N_{f_1}}^1$
h_1^n h_2^n h_3^n $h_{N_{f_n}}^n$
Input
Convolution layer 1
Pooling layer 1
Convolution layer *n*
Pooling layer *n*
Flatten
Dense
Latent vector
Reconstruction
Convolution
Upsampling
Convolution
Upsampling
Reshape
Dense
Decreasing filters and increasing dimension
Decoder ($\mathcal{F}_{dec}$)

Fig. 1 Schematic representation of 1D convolutional autoencoder architecture. The same structure of the 1D-CAE is adopted in the encoding and decoding processes for both the space and time dimensions

to be maintained [37]. The output features resulting from a max-pooling operation are illustrated in the encoder part of Fig. 1 for the n^{th} layer, and denoted by p_i^n with $i \in (1, N_{f_n})$.

Non-intrusive reduced order model based convolutional autoencoders (NIROM-CAEs)

As mentioned above, the proposed approach combines convolutional autoencoders with the reduced order modeling concept and concerns the parametrized time-dependent problems with large-scale computational domains. The framework of the non-linear surrogate model is mainly based on a succession of two compression levels through 1D convolutions. The first level concerns the space dimension while the second deals with the compression of the time dimension of the latent space obtained from the former spatial compression.

Dataset construction

The model proposed in this work belongs to the data-driven approaches where the input dataset, known as a snapshot matrix, is constructed through a collection of high-fidelity solutions from the numerical solver $\left\{\mathbf{u}(\eta_s, t_j) \in \mathbb{R}^{N_x \times 1}; j = 1, \ldots, N_t; s = 1, \ldots, N_s\right\}$, where η_s represents the s^{th} value of the random parameter η within its generated sample set with a size of N_s, following an appropriate probability density function $\varrho(\eta)$. t_j denotes the j^{th} time step within the temporal domain $t \in \mathcal{T} = [0, T]$ decomposed into N_t time steps. The high fidelity solutions are collected from the numerical solver as a column vector $\mathbf{u} \in \mathbb{R}^{N_x \times 1}$, where N_x is the total number of the meshing nodes that cover the computational domain. The snapshot matrix, obtained by gathering these solution vectors, can be expressed as a 3D global matrix:

$$\boldsymbol{\mathcal{U}} = \left[\, \mathcal{U}_1 \mid \ldots \mid \mathcal{U}_s \mid \ldots \mid \mathcal{U}_{N_s} \right] \in \mathbb{R}^{N_s \times N_x \times N_t}, \tag{2}$$

where $\mathcal{U}_s \in \mathbb{R}^{N_x \times N_t}$ is a 2D solutions matrix corresponding to the parameter value η_s over all the time steps, expressed as:

$$\mathcal{U}_s = \left[\mathbf{u}(\eta_s, t_1) \mid \ldots \mid \mathbf{u}(\eta_s, t_{N_t})\right] \in \mathbb{R}^{N_x \times N_t}. \tag{3}$$

It should be noted that the global snapshot matrix given by Eq. (2) is divided during the learning process into two sets: a training set (80 %) and a validation set (20 %) for the space and time autoencoders.

Spatial compression

The spatial compression represents the first compression level of the proposed technique; its goal is to reduce the dimension of the input dataset along the spatial dimension from N_x to L_x with $L_x \ll N_x$. Each snapshot matrix $\mathcal{U}_s \in \mathbb{R}^{N_x \times N_t}$, as given by Eq. (3), has its relation to the parameter value η_s reshaped by considering its transpose $\mathcal{U}_s^T \in \mathbb{R}^{N_t \times N_x}$ on which the space encoder $\mathcal{F}_{x_{enc}}$ is applied along the spatial dimension as follows:

$$\mathcal{V}_{x_s} = \mathcal{F}_{x_{enc}}\left(\mathcal{U}_s^T\right) \in \mathbb{R}^{N_t \times L_x}, \tag{4}$$

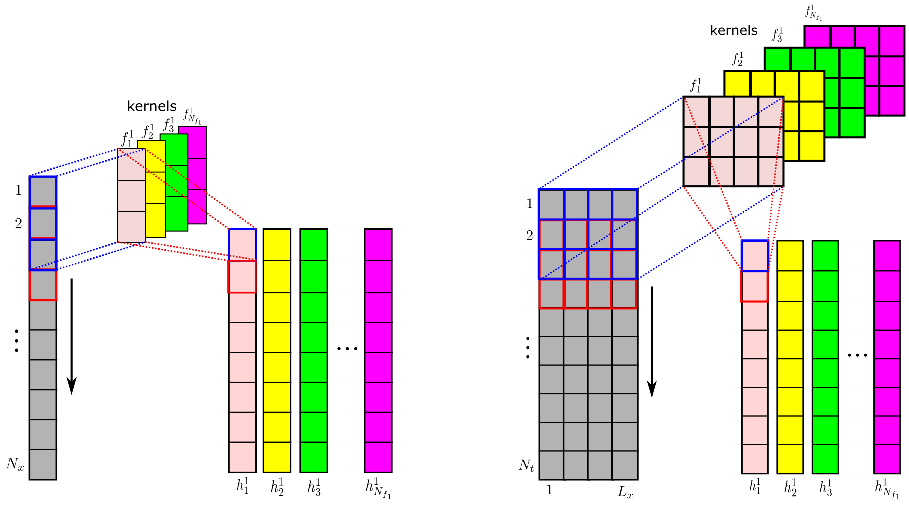
where $\mathcal{V}_{x_s} \in \mathbb{R}^{N_t \times L_x}$ represents the encoded latent space corresponding to the parameter value η_s, with $s = 1, \ldots, N_s$. It should be emphasized that for implementation purposes in the TensorFlow library [38], the dimension of the input dataset has been extended to a 3D tensor of a shape $(N_t, N_x, 1)$ through the use of a reshape operator: $\mathcal{R} : \mathbb{R}^{N_t \times N_x} \mapsto \mathbb{R}^{N_t \times N_x \times 1}$ [26], where the third dimension stands for the number of channels, which is equal to one in this case. A schematic representation of the 1D convolution operation performed on the first convolution layer by the space encoder is depicted in Fig. 2a, where N_{f_1} 1D-filters, whose width corresponds to the number of channels, convolve a local patch of the input dataset along the spatial dimension, represented by the vertical arrow, to generate the features of the convolved layer h_i^1, $i = 1, \ldots, N_{f_1}$. As shown in Fig. 2a, each filter is distinguished by the same color as its corresponding column features generated when it convolves along the spatial direction. The detailed structures of the space autoencoders adopted for different test cases are reported in Tables 2 and 4.

Temporal compression

Once the spatial dimension is encoded, a second compression level is performed on the obtained latent space $\mathcal{V}_{x_s} \in \mathbb{R}^{N_t \times L_x}$ along the temporal dimension to reduce its size from N_t to a given L_t through a 1D time encoder $\mathcal{F}_{t_{enc}}$ that has the same structure as that shown in Fig. 1. The encoding process can be mathematically expressed as follows:

$$\mathcal{V}_{t_s} \in \mathbb{R}^{L_t} = \mathcal{F}_{t_{enc}}\left(\mathcal{V}_{x_s}\right), \tag{5}$$

where $\mathcal{V}_{t_s} \in \mathbb{R}^{L_t}$ is the encoded latent vector corresponding to the single parameter value η_s. The convolution operation of the first layer 1D-time encoder, expressed by



(a) Convolution along the spatial dimension

(b) Convolution through the temporal dimension

Fig. 2 Schematic representation of the 1D-convolutional operation along the spatial and temporal dimensions. For interpretation of the references to color in this figure, the reader is referred to the web version of this article

Eq. (1), is presented in Fig. 2b, where N_{f_1} filters convolve local patches of the input dataset $\mathcal{V}_{x_s} \in \mathbb{R}^{N_t \times L_x}$ along the temporal dimension (N_t). L_x denotes the number of channels of the input dataset, which corresponds to the width of each filter. The detailed structures of the time autoencoders applied to the test cases addressed in this study are reported in Tables 3 and 5, along with information about the number of convolution layers, kernel sizes, and activation functions.

Regression based multilayer perceptron (MLP)

The third level of the proposed approach concerns the encoded latent vector $\mathcal{V}_{t_s} \in \mathbb{R}^{L_t}$, obtained through two compression levels of the input dataset $\mathcal{U}_s \in \mathbb{R}^{N_x \times N_t}$, which is mapped to the input parameter η_s via a multilayer perceptron (MLP) regression. This regression, which links the input parameter to the final latent vector, composed of multiple stacked fully-connected layers, can be expressed as follows [32]:

$$\mathcal{V}_{t_s} = \Pi_{MLP}\left(\eta_s\right). \tag{6}$$

The detailed architecture of the MLP used in this work for the addressed test cases is presented in Table 6. The mathematical framework presented in subsections through represents the main steps of the offline stage of the proposed NIROM-CAEs technique.

It is worth mentioning that the proposed model was built and trained using the open-source deep learning library TensorFlow [38] using the Adam optimizer with its default parameters. To accelerate the convergence and optimize the training processes, the data used during the training process of the offline stage are all normalized with the same min-max scaling [32,39]:

$$\tilde{u}^s = \frac{u^s - min(\mathbf{u}_j)}{max(\mathbf{u}_j) - min(\mathbf{u}_j)} - 0.5\ ; \quad j = 1, \ldots, N_t; \quad s = 1, \ldots, N_s, \tag{7}$$

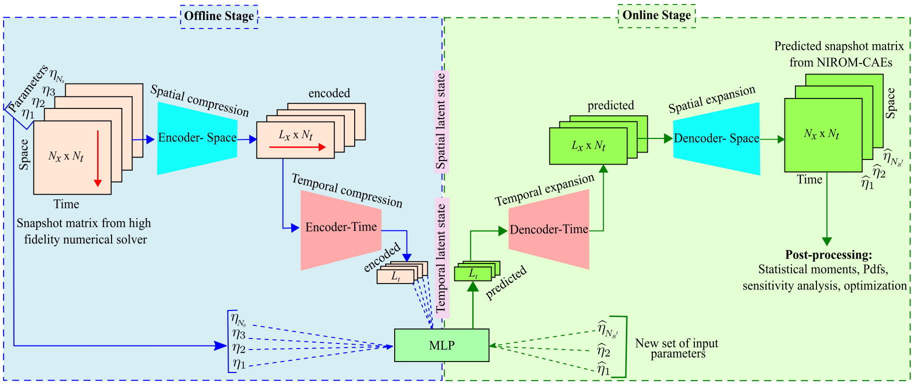


Fig. 3 Flowchart illustrating the offline training and online predictive stages of the NIROM-CAEs model. The red arrow indicates the encoding direction (for interpretation of the references to color in this figure, the reader is referred to the web version of this article)

with $\tilde{u}^s \in [-0.5, 0.5]$. The inverse transformation must also be applied to the predicted data matrix during the online stage to recover the original scaling.

Online surrogate prediction

Once the three models are trained during the offline stage described above, the online surrogate prediction becomes straightforward. A new set of the random parameters is generated following its probability distribution function $\widehat{\boldsymbol{\eta}} = \{\widehat{\eta}_1, \ldots, \widehat{\eta}_{N_{s'}}\}$, and for each unseen value $\widehat{\eta}_s$, the encoded Spatiotemporal latent vector $\widehat{\mathcal{V}}_{t_s}$ is predicted through the MLP regression model: $\widehat{\mathcal{V}}_{t_s} \in \mathbb{R}^{L_t} = \Pi_{MLP}\left(\widehat{\eta}_s\right)$. This predicted latent space is then decoded through the time decoder $\mathcal{F}_{t_{dec}}$ to approximate the latent spatial space: $\widehat{\mathcal{V}}_{x_s} \in \mathbb{R}^{N_t \times L_x} = \mathcal{F}_{t_{dec}}\left(\widehat{\mathcal{V}}_{t_s}\right)$. Finally, the obtained latent space is decoded using the space decoder $\mathcal{F}_{x_{dec}}$ to generate the predicted input dataset: $\widehat{\mathcal{U}}_s \in \mathbb{R}^{N_t \times N_x \times 1} = \mathcal{F}_{x_{dec}}\left(\widehat{\mathcal{V}}_{x_s}\right)$, which is transformed to its final shape through the inverse reshape operator $\mathcal{R}^{-1} : \mathbb{R}^{N_t \times N_x \times 1} \mapsto \mathbb{R}^{N_t \times N_x}$. Thus, the predicted snapshot matrix corresponding to the single value parameter $\widehat{\eta}_s$ throughout the online stage can be expressed as follows:

$$\widehat{\mathcal{U}}_s = \mathcal{F}_{x_{dec}}\left(\mathcal{F}_{t_{dec}}\left(\Pi_{MLP}\left(\widehat{\eta}_s\right)\right)\right). \tag{8}$$

The statistical moments can be estimated via the constructed surrogate model of the stochastic output response $\widehat{u}$ as follows:

$$\mu_{\widehat{u}} = \mathbb{E}\left[\widehat{u}\right] = \int_{\Xi} \widehat{u}\varrho(\widehat{\eta})d\widehat{\eta}, \tag{9}$$

and

$$\sigma_{\widehat{u}}^2 = \mathbb{E}\left[\widehat{u}^2\right] - \mu_{\widehat{u}}^2 = \left[\int_{\Xi} \left(\widehat{u}\right)^2 \varrho(\widehat{\eta})d\widehat{\eta}\right] - \mu_{\widehat{u}}^2 \tag{10}$$

The fundamental framework of the proposed NIROM-CAEs approach is summarized in the flowchart presented in Fig. 3, which presents the most relevant steps of the offline training and online predictive stages.

The predictive abilities of the proposed NIROM-CAEs are assessed in comparison with the Latin Hypercube Sampling approach (LHS), whose results are considered as reference solutions. Another method based on a proper orthogonal decomposition concept combined with an artificial neural network (POD-ANN) is retained as a linear non-intrusive reduced order modeling technique for comparison purposes. A detailed presentation of the development and implementation steps of POD-ANN is beyond the scope of this paper, and the reader is encouraged to consult the corresponding references [14–16] for a deeper insight into the modeling procedure. It is worth mentioning that the architecture choice and hyperparameters selection were performed after a series of preliminary settings and a trial-and-error approach by combining, among others, the number of convolution layers, the number of filters, the kernel sizes, the activation functions, and the number of epochs. Indeed, several computations have been performed to select the appropriate 1D-spatial autoencoder structures for both spatial and temporal dimensions, and a tradeoff was made between the accuracy, the targeted compression levels, and the computational cost. A detailed description of the architectures and hyperparameters settings for each test case is presented in Tables 2, 3, 4 and 5.

Results and discussion

This section presents the application of the proposed non-intrusive reduced order model to two well-known 1D benchmark test cases (Burgers' and Stoker's analytical problems) and to a hypothetical failure of a real dam case. The results are depicted to showcase the accuracy and efficiency of the proposed approach in approximating the statistical moments for time-dependent problems.

One-dimensional Burgers equation test case

The first test case concerns the one-dimensional viscous Burgers' equation for nonlinear convection-diffusion, expressed in its dimensionless form with the corresponding initial and Dirichlet boundary conditions as follows [40,41]:

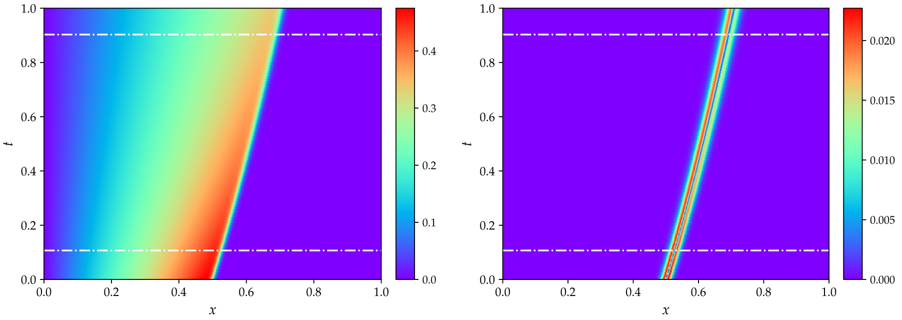
$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \frac{1}{Re}\frac{\partial^2 u}{\partial x^2}\,, \qquad x \in [0,\ 1], \qquad t \in [0,\ 1], \tag{11}$$

with: $u(x,0) = \frac{x}{1+exp\left(\frac{Re}{16}(4x^2-1)\right)}$ and $u(0,t) = u(1,t) = 0$. An analytical solution for the velocity field is given by [33]:

$$u(x,t) = \frac{\frac{x}{t+1}}{1+\sqrt{\frac{t+1}{t_0}}exp(Re\frac{x^2}{4t+4})}, \tag{12}$$

with $t_0 = exp(\frac{Re}{8})$. In this test case, the Reynolds number is considered as a random variable which follows a uniform distribution within its variability range $Re_{\mu=1000,\,\sigma=200} \in \mathcal{U}\,[654,\ 1\,346]$. The sample used in this test case has a size $N_s = 200$, randomly divided into two sets: one for training (80 %) and one for validation (20 %). For each selected value in the generated random parametric sets, the analytical solution given by Eq. (12) is evaluated over a discretized spatial domain with $N_x = 1000$ nodes for all the $N_t = 104$ time-steps to construct the training and validation high-fidelity snapshot matrices that are used to train the 1D spatial-CAE ($\mathcal{F}_x$) of the NIROM-CAEs approach, as described above.

The detailed architectures of the spatial encoder ($\mathcal{F}_{x_{enc}}$) and the spatial decoder ($\mathcal{F}_{x_{dec}}$), which consist of 1 033 773 trainable parameters, are given in Table 2. The spatial encoder



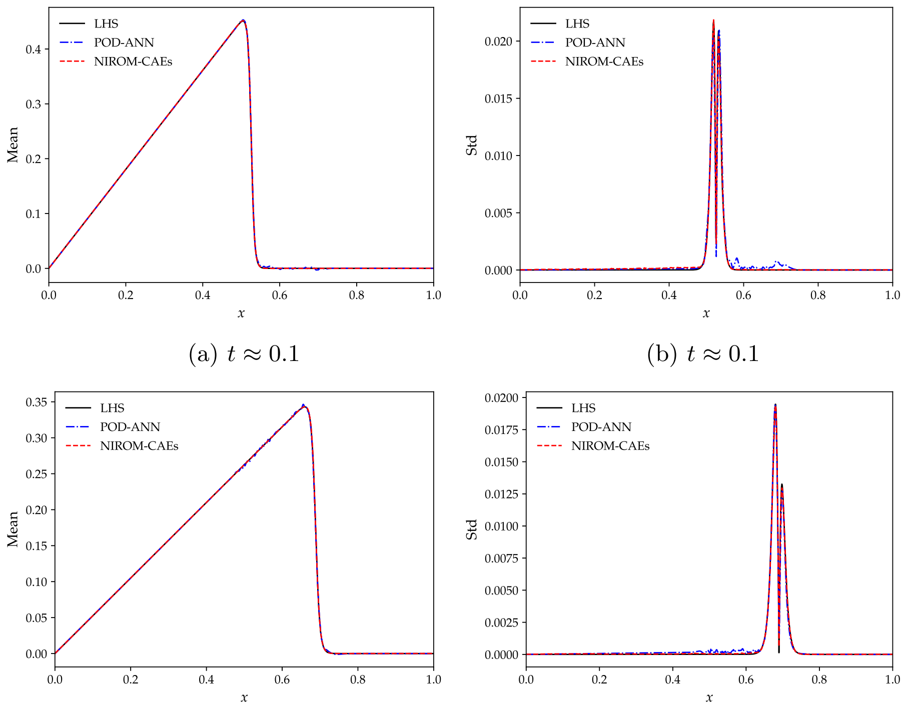
(a) Mean (b) Std

Fig. 4 Contour plots of the mean and standard deviation of the Burgers' analytical solution obtained from 5000 LHS realizations with uniform distribution $Re_{\mu=1000,\sigma=200} \in \mathcal{U}\,[654,\ 1346]$. The two horizontal dashed white lines indicate the times for which the results in Fig. 5 are presented. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

allows the compression of the spatial dimension from $N_x = 1000$ nodes to a reduced-order latent space of dimension $L_x = 50$ through a succession of 1D convolutional and max-pooling layers with associated nonlinear activation functions. The generated spatial latent space of dimension $N_t = 104 \times L_x = 50$ serves as input to the 1D-time autoencoder ($\mathcal{F}_t$), which reduces the temporal dimension from $N_t = 104$ to $L_t = 10$. The detailed structures of both the encoder ($\mathcal{F}_{t_{enc}}$) and the decoder ($\mathcal{F}_{t_{dec}}$), with 177 494 trainable parameters, are shown in Table 3. The spatiotemporal latent space thereby obtained is then mapped with the input parameters using a multi-layer perceptron whose structure is given in Table 6 and which consists of 34 570 trainable parameters. The CAE-space, CAE-time, and the MLP were trained for 500, 1000 and 3000 epochs, respectively, and the decay of the training and validation losses during the training phase is depicted in Fig. 18.

The most relevant results for this test case obtained by the proposed NIROM-CAEs model are presented in terms of the statistical moments and relative L^2-error profiles, and compared with those from the POD-ANN and the LHS solutions. In Fig. 4, the 2D contours of the mean and standard deviation from the LHS method with 5000 realizations, considered as a reference solution, are presented in the space-time plane, where the discontinuity in the velocity field (Fig. 4a) and the variability surrounding its propagation with time (Fig. 4b) may present a challenging test case for the proposed reduced order modeling approach. The two horizontal dashed lines represent the time location for which the mean and the standard deviation profiles are displayed along with the spatial domain.

Figure 5 reports the comparison of the mean and standard deviation profiles as a function of the x-coordinate at different time-steps ($t \approx 0.1$ and 0.9) obtained with NIROM-CAEs (with $L_x = 50$ and $L_t = 10$) and POD-ANN (with $\epsilon_s = \epsilon_t = 10^{-10}$, which generates a latent space of dimension $L_{POD} = 77$ modes), with those from the reference LHS solutions (with $N_s = 5\,000$ realizations). These results show that the proposed non-intrusive surrogate model (NIROM-CAEs) predicts profiles of the mean (left column) and standard deviation (right column) that are in close agreement with those from the LHS reference solution. Conversely, an oscillatory behavior can be showcased in the statistical moments' curves obtained with the POD-ANN technique with their pronounced deviations from the LHS solution curves. It should be emphasized that the predicted statistical moments from the NIRON-CAEs approach during the predictive online phase were obtained with



(a) $t \approx 0.1$

(b) $t \approx 0.1$

(c) $t \approx 0.9$

(d) $t \approx 0.9$

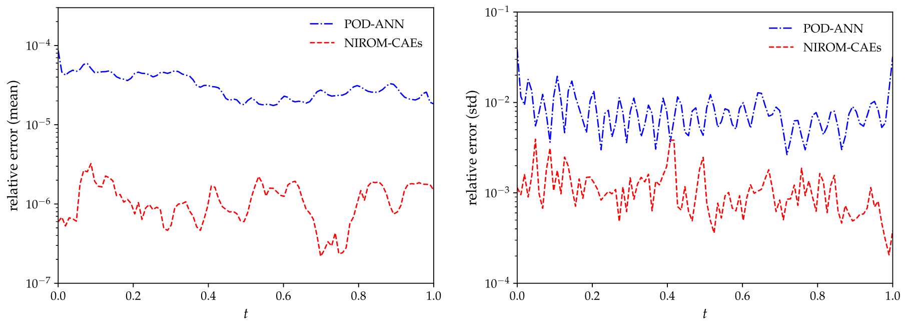
Fig. 5 Comparison of the mean (left) and standard deviation (right) of the Burgers analytical solutions obtained with the POD-ANN and NIROM-CAEs approaches with those of the LHS reference solution (5000 realizations) at different times: $t \approx 0.1$ (**a**, **b**) and $t \approx 0.9$ (**c**, **d**)

new values of the random input parameter that differ from those used in the offline training phase. As evidenced by the results displayed in Fig. 5, the surrogate NIROM-CAEs technique allows a better approximation of the output quantities of interest due to the nonlinearities that effectively capture the dynamics of the viscous advection shock in contrast with the linear-based POD approach.

To further investigate the performance of the proposed approach, the time trajectory of the spatial relative L^2-errors of the mean and standard deviation of the predicted solutions obtained by NIROM-CAEs are compared with those from the POD-ANN model. The relative errors are computed for the LHS reference solutions with 5 000 realizations over the whole set of computational nodes and for each time step. Its mathematical expression is defined as follows:

$$Err_{L^2,\Phi}^{Surr}(t) = \sqrt{\frac{\sum_{i=1}^{N_e} \left(\Phi_{i,Surr}(t) - \Phi_{i,LHS}(t)\right)^2}{\sum_{i=1}^{N_e} (\Phi_{i,LHS}(t))^2}}, \tag{13}$$

where Φ denotes either the mean or the standard deviation, and *Surr* stands for NIROM-CAEs or POD-ANN. The comparison of the mean and standard deviation error profiles highlights the predictive abilities of the NIROM-CAEs model, which presents lower error values compared to those from the POD-ANN approach as shown in Fig. 6. These plots confirm the capacity of the proposed nonlinear reduced-order model to accurately estimate the statistics of the outputs, even for challenging time-dependent physical problems such as the advective viscous shock Burgers'equation.



(a) Mean (b) Std

Fig. 6 Variation of the L^2-relative error of the mean (left) and standard deviation (right) as a function of time obtained with the POD-ANN (blue) and NIROM-CAEs (red). Errors are evaluated in reference to the LHS reference solution obtained with 5000 realizations

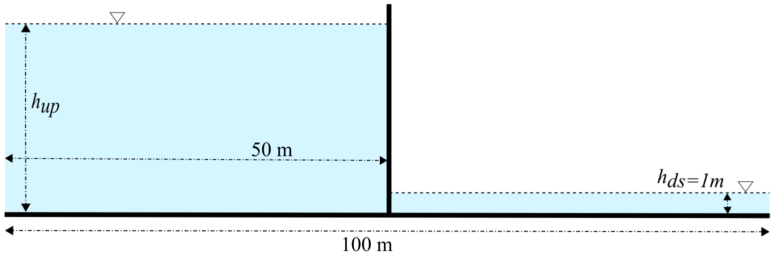


Fig. 7 Schematic representation of the initial condition of the Stoker's analytical solution of a 1D dam break over a wet flat bottom

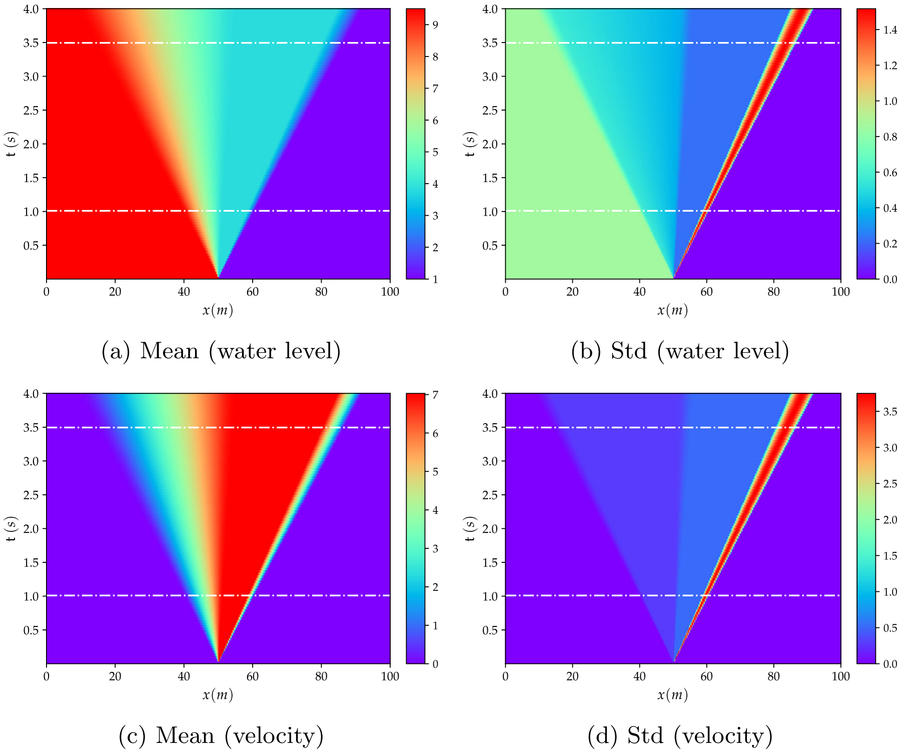
One dimensional dam break test case

The second test case concerns Stoker's analytical solution [42], which describes the propagation and rarefaction wave resulting from a one-dimensional dam break over a wet flat frictionless bottom. The dynamic state is initiated by unequal water levels of both the upstream and downstream sides located in the middle of the studied domain of 100 m, as shown in Fig. 7. The mathematical expressions for the water level and the velocity of the parametrized Stoker's analytical solution fields are given as follows [43,44]:

$$h(x,t) = \begin{cases} h_{up} \\ \frac{4}{9g}\left(\sqrt{g\,h_{up}} - \frac{x}{2t}\right)^2 \\ \frac{c_m^2}{g} \\ h_{ds} \end{cases}, \quad u(x,t) = \begin{cases} 0 \quad m/s & \text{if } x \leq x_A(t) \\ \frac{2}{3}\left(\frac{x}{t} + \sqrt{g\,h_{up}}\right) & \text{if } x_A(t) \leq x \leq x_B(t) \\ 2\left(\sqrt{g\,h_{up}} - c_m\right) & \text{if } x_B(t) \leq x \leq x_C(t) \\ 0 \quad m/s & \text{if } x_C(t) \leq x, \end{cases} \tag{14}$$

where x denotes the axial position, $x_A(t) = x_0 - t\sqrt{g\,h_{up}}$, $x_B(t) = x_0 + t(\sqrt{g\,h_{up}} - 3c_m)$ and $x_C = x_0 + t\frac{2c_m^2(\sqrt{g\,h_{up}} - c_m)}{c_m^2 - g\,h_{ds}}$, and where $c_m = \sqrt{gh_m}$ denotes the selected solution of $-8gh_{ds}c_m^2(g\,h_{up} - c_m^2)^2 + (c_m^2 + g\,h_{ds})(c_m^2 - g\,h_{ds})^2 = 0$.

The upstream water level (h_{up}) is considered as an input random variable whose values are uniformly sampled within its plausible variability range $h_{up} \in \mathcal{U}\,[8,\ 11]$, whereas the



(a) Mean (water level)

(b) Std (water level)

(c) Mean (velocity)

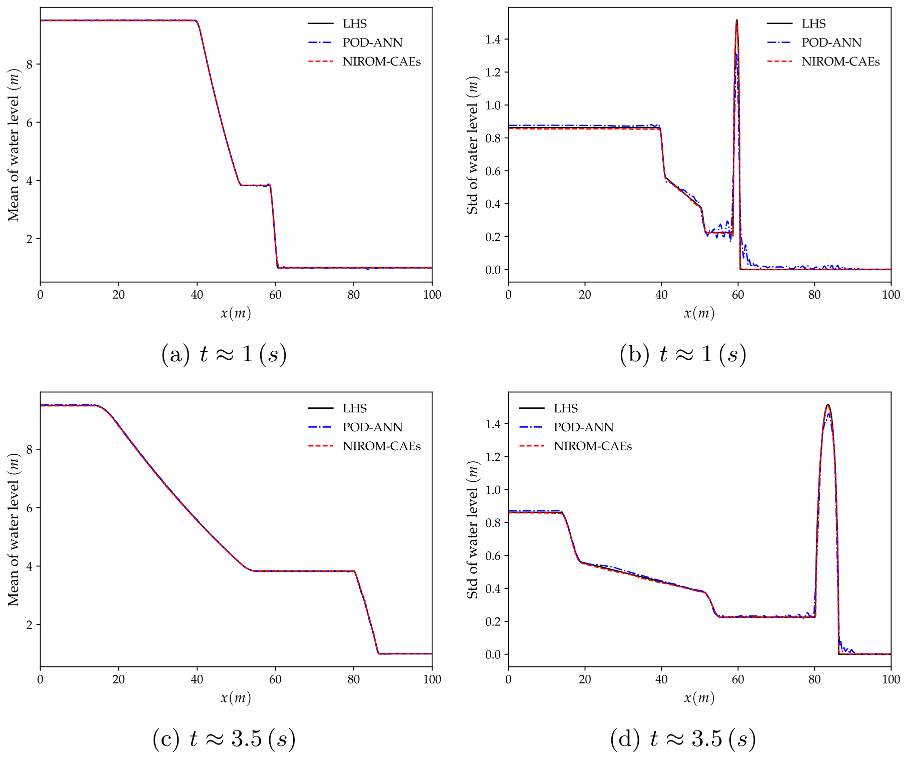
(d) Std (velocity)

Fig. 8 Contour plots of the mean (left column) and standard (right column) deviation of the Stoker's analytical solution of the water level (upper row) and the velocity (bottom row) obtained from the LHS solution with 5000 realizations. The two horizontal dashed lines represent the times for which the results in Figs. 9 and 10 are presented

downstream water depth is kept constant at a deterministic value $h_{ds} = 1\ m$. For each selected value in the generated sample set of the upstream water level, the analytical solution, given by Eq. (14), is evaluated over the $N_x = 1\,000$ nodes that contain the computational domain $x \in [0,\ 100]\ m$ for all the $N_t = 104$ time-steps of the temporal domain $t \in [1,\ 4]\ s$. The obtained solution vectors are then concatenated to construct the so-called high-fidelity snapshot matrix that trains the proposed non-intrusive reduced-order model (NIROM-CAEs) during the offline training phase.

Stoker's problem is considered among the most challenging benchmark test cases due to its strong hyperbolic behavior and the discontinuity accompanying the propagation of the front wave resulting from the initial breaking. The performance of the proposed NIROM-CAEs is assessed over this test case by comparing the obtained statistical moments to those from the LHS reference solution, whose contours for the water level and velocity fields are depicted in the spatiotemporal plane as shown in Fig. 8, where the longitudinal and transversal axes represent the spatial and temporal dimensions, respectively. The two horizontal lines show the time locations, $t \approx 1$ and $t \approx 3.5$ s, for which the evolution of the mean and standard deviation profiles over the channel length, respectively, are compared with the LHS and POD-ANN results.

Similar to the former numerical test case, the spatial and temporal autoencoders structures are composed of three 1D-convolutional layers with 32, 64 and 128 channels each, thereby reducing the spatial dimension from $N_x = 1\,000$ to $L_x = 50$, and then to $L_t = 10$, representing the reduced latent space. The obtained spatiotemporal latent space is then



(a) $t \approx 1\,(s)$

(b) $t \approx 1\,(s)$

(c) $t \approx 3.5\,(s)$

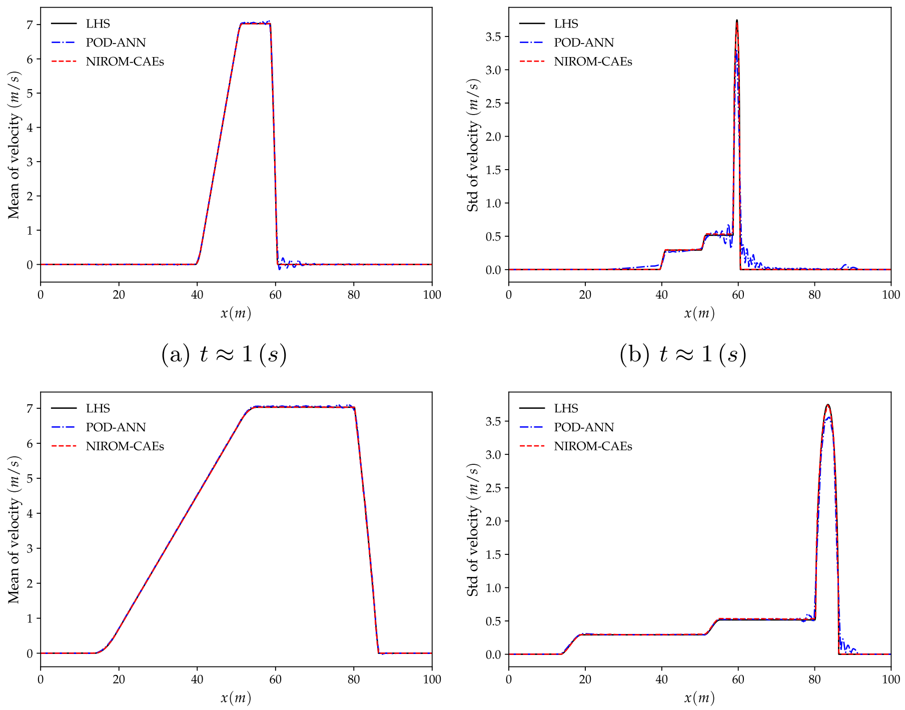
(d) $t \approx 3.5\,(s)$

Fig. 9 Distribution of the mean (left) and standard deviation (right) of the water level along the channel length at different time steps ($t \approx 1$ and $t \approx 3.5$ s). The results obtained from the POD-ANN and NIROM-CAEs techniques are compared with those from the LHS reference solution

linked with the input parameter vector via MLP mapping. The detailed architectures of the proposed NIROM-CAEs model with its CAE-space, CAE-time, and MLP are summarized in Tables 2, 3 and 6, respectively. The convergence history of the spatial, temporal CAEs and the MLP during the training phase are depicted in Fig. 19 with the number of epochs of 500, 1000, and 3000, respectively.

The results obtained by the proposed NIROM-CAEs model (presented in terms of the variation of the mean and standard deviation profiles as a function of the x-coordinate at different times ($t \approx 1,\ 3.5\ s$) are compared with those from the LHS reference solution (obtained with 5 000 realizations), as shown in Figs. 9 and 10. In addition to the LHS solution, the POD-ANN model is considered another solution with which to assess the predictive ability of the nonlinear NIROM-CAEs. The structure of the neural network introduced in the POD-ANN is constituted by three hidden layers, each of which contains 50 neurons. A sample set of $N_s = 300$ values of the upstream water level is randomly generated in its plausible variability range to build the snapshot matrices for both models by collecting the corresponding high-fidelity solutions, of which 80% is used for training and the remaining 20% for validation.

The comparison of the mean and standard deviation profiles of the water level presented in Fig. 9 shows that both the POD-ANN and NIROM-CAEs models accurately predict the distribution of the mean over the whole channel for the two simulation times, as a good agreement with the reference LHS profiles can be observed. However, the POD-ANN predictions for the standard deviation exhibit spurious oscillations in the area surrounding



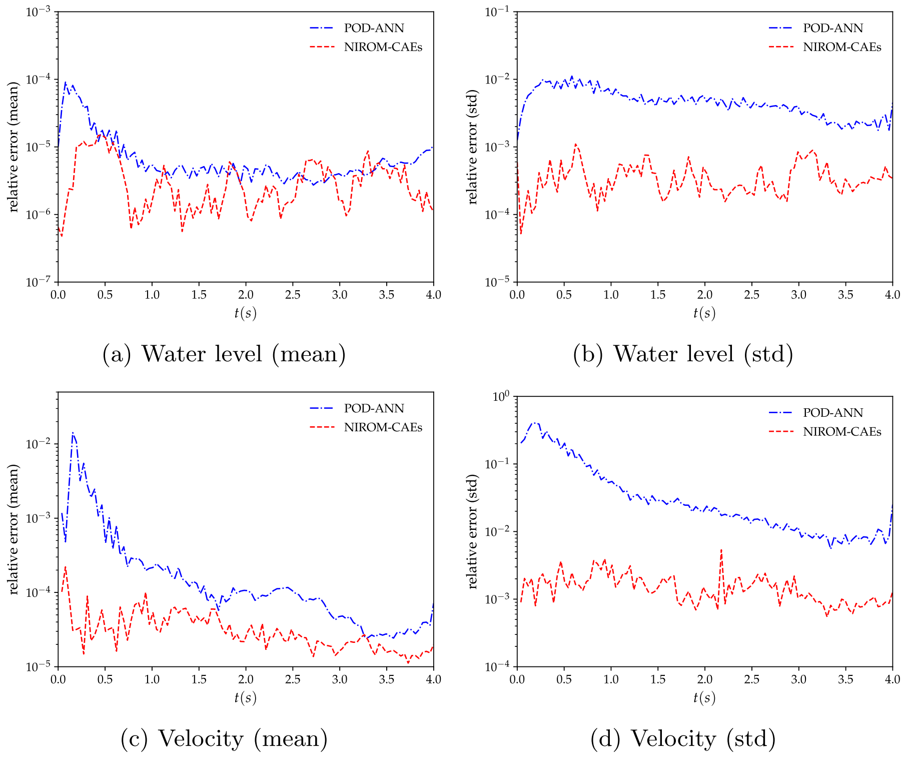
(a) $t \approx 1\,(s)$ (b) $t \approx 1\,(s)$

(c) $t \approx 3.5\,(s)$ (d) $t \approx 3.5\,(s)$

Fig. 10 Distribution of the mean (left) and standard deviation (right) of the velocity along the channel length at different time steps ($t \approx 1$ and $t \approx 3.5$ s). The results obtained from the POD-ANN and NIROM-CAEs techniques are compared with those from the LHS reference solution

the front shock wave, where significant deviations appear in comparison with the profiles from the LHS solution. In contrast, the predicted profiles from the proposed NIROM-CAEs show an excellent agreement with those from the high fidelity LHS solutions, even in the area where the discontinuity occurs. The same trend can be observed in the velocity statistical moments' profiles, as shown in Fig. 10, conversely to the approximated outputs from the POD-ANN, which are characterized by an oscillatory behavior. It can be concluded that, for this test case, the use of convolutional neural network autoencoders with nonlinear activation functions provides a powerful nonlinear compression model with high predictive abilities in capturing the dynamics of the outputs in contrast with the linear compression model based on the traditional POD approach.

To better illustrate the potential of nonlinear model reduction approaches to efficiently predict the statistical moments of the output responses of high-complexity problems, an evaluation of the relative L^2 error is performed for both the water level and velocity fields, as shown in Fig. 11. The mean and the standard deviation errors are calculated for each time step by evaluating the differences between the predicted statistical from the LHS reference over the entire calculation domain, as formulated in Eq. (13). The evolution of the relative errors as a function of time shows that the NIROM-CAEs approach presents a better approximation of the output responses, with errors on the order of 10^{-7} for the mean and 10^{-4} for the standard deviation, unlike the linear POD-ANN technique which presents higher error values that can reach up to two orders of magnitude in the case of the standard deviation for both velocity and water level. This quantitative analysis shows that



(a) Water level (mean) (b) Water level (std)

(c) Velocity (mean) (d) Velocity (std)

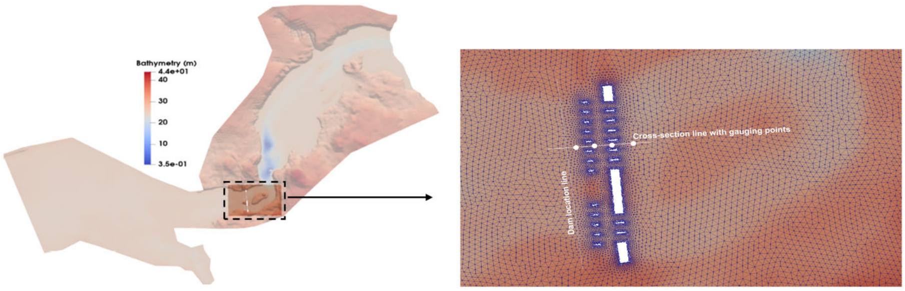
Fig. 11 Comparison of the relative L^2-error profiles of the mean (left) and standard deviation (right) of the water level (first row) and velocity (second row) as a function of time, obtained with the POD-ANN and NIROM-CAEs. Errors are evaluated with respect to the 5000-realization LHS reference solution

the introduction of non-linearities contributes to a better approximation of the statistical moments, allowing the construction of efficient spatiotemporal compression models for time-dependent complex problems.

Application to a hypothetical dam-break in a river

The proposed approach is applied to a third test case to evaluate its ability to perform an efficient uncertainty propagation analysis on a spatial domain with complex bathymetry. The test concerns a reach of the Milles-Iles River (province of Québec, Canada) including a dam as shown in Fig. 12. The data relating to the bathymetry and the roughness coefficient were provided by the Communeauté Métropolitaine de Montréal (CMM) from measurements and observations. The sub-domain of study is composed of an unstructured triangular mesh with 16 763 elements and $N_e = 10\,200$ nodes, over which high-fidelity solutions of the quantities of interest are collected from an in-house multi-GPU finite volume solver for shallow water equations [45]. A detailed description of the physical domain of this test case was addressed in [13].

A fictitious breaching process was initialized on both parts of a hypothetical dam with unequal water levels, located as indicated by a line in Fig. 12b. The downstream part of the dam is considered as dry whereas the free surface of the upstream part is considered as a random input parameter whose values are uniformly generated within its plausible variability range $\eta_{up} \in \mathcal{U}\,[29,\ 32]\ m$. The snapshot matrix is obtained by running the numerical solver for each value of the upstream free surface, selected randomly from



(a) Reach of the Mille-Iles river

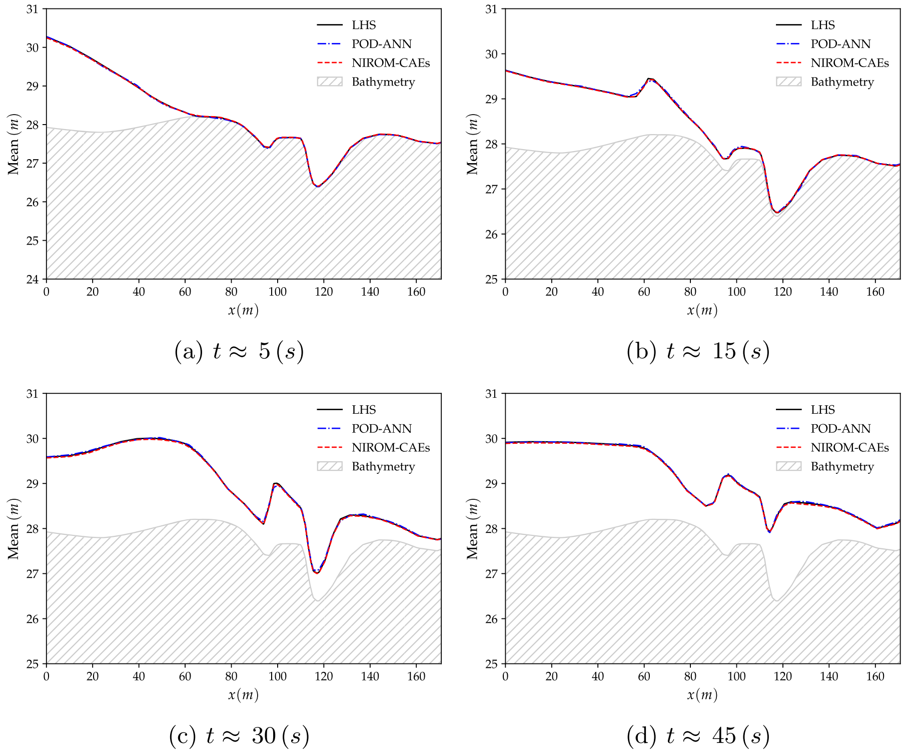
(b) Zoom of the studied sub-domain

Fig. 12 Sketch of the reach of the Mille Iles river with a close-up view of the studied zone. The cross-section line and the gauging points indicate locations where results are represented as a function of the longitudinal direction and time, respectively

the generated sample set, for the whole $N_t = 100$ simulation time steps that constitute the temporal domain ($t \in [0, 50]$ s). For each parameter-time combination, a so-called high-fidelity solution is stored in a vector of dimension $N_x = 10\,200$, representing the free surface values at each node of the computational domain. The concatenation of these solution vectors thus allows the construction of the snapshot matrix.

The spatial compression is performed by the proposed NIROM-CAEs using a space-autoencoder composed of two 1D-convolutional layers, one with 32 and the other one with 64 channels, followed by a max-pooling and dense layers, thereby reducing the initial spatial dimension from $N_x = 10\,200$ nodes to $L_x = 50$, representing the latent dimension. A second time-autoencoder is then applied to the generated spatial latent space to reduce the temporal dimension from $N_t = 100$, representing the number of time-steps, to $L_t = 10$. The time-autoencoder has a succession of three 1D-convolutional max-pooling layers followed by a dense layer that links the channels to the temporal latent space. The obtained latent space is then linked to the input parameter through an MLP of three layers, each with a width of 128 neurons, to map the values of the input parameter to the final latent space. A detailed description of the architectures of the space and time autoencoders and the MLP are presented in Tables 4, 5, and 6. The numbers of trainable parameters are 6 915 197, 153 230 and 34 570, respectively. The convergence histories of the space and time CAEs and the MLP during the training phase, performed on a Tesla P100 GPU with 32 GB of memory, are depicted in Fig. 20 with the number of epochs of 200, 1000 and 2000, respectively.

The results stemming from this test case are presented mainly in terms of the variation of the mean and standard deviation profiles of the water surface level over the studied computational sub-domain. The results obtained with the proposed NIROM-CAEs are compared with those from the LHS solutions, both obtained by running the deterministic numerical solver using a sample set of $N_s = 2000$ points of the uncertain initial upstream water level. The predictive model of the POD-ANN approach is based on a neural network composed of three hidden layers, each with of 50 neurons, that map the reduced-basis coefficient to the input parameter. Figures 13 and 14 present the variation of the mean and the standard deviation profiles, respectively, as a function of the longitudinal coordinate along the cross-section line, as shown in Fig. 12b. The comparison of the predicted mean profile of the water level at different simulation times shows that the NIROM-CAEs'predictions are in good agreement with the profiles from the LHS



(a) $t \approx 5\,(s)$

(b) $t \approx 15\,(s)$

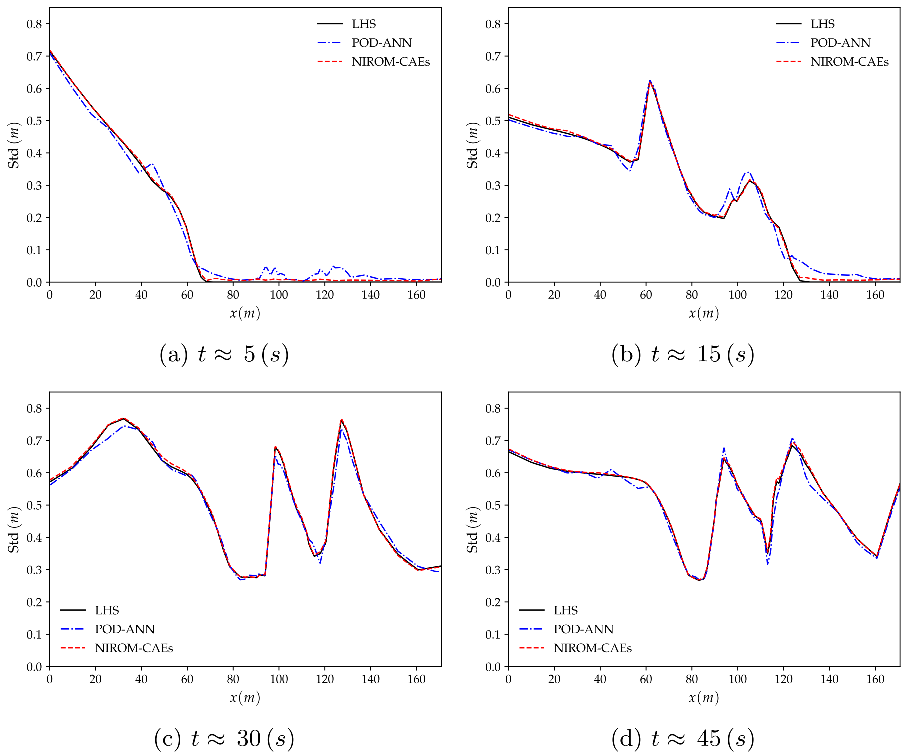
(c) $t \approx 30\,(s)$

(d) $t \approx 45\,(s)$

Fig. 13 The mean profiles of the water level over the cross-section line at various time-steps, obtained with the POD-ANN and NIROM-CAEs and compared with those from the LHS reference solution (with 2000 realizations). **a**: $t \approx 5\,s$, **b**: $t \approx 15\,s$, **c**: $t \approx 30\,s$ and (**d**): $t \approx 45\,s$

reference solutions. The POD-ANN model also presents satisfactory predictions for the mean. The shock wave propagation is visible as the water level over the cross-section line increases with time. Figure 13a through 13d show the hatched area that represents the bathymetry of the terrain. Despite the accurate results obtained by the POD-ANN approach for the mean profiles of the water level, its standard deviation predictions over the cross-section line reveal spurious oscillations, as shown in Fig. 14a through 14d. In contrast, the predictions from the NIROM-CAE approach show good agreement with the reference solution for all the presented simulation times. This further illustrates the good abilities of the proposed nonlinear reduced-order approach in approximating the outputs of high-dimensional time-dependent problems.

In addition to the results presented above, four gauging points were chosen to depict the evolution of statistical moments as a function of time. These gauging points, whose approximate positions on the cross-section line are shown in Fig. 12b, were selected to show the ability of the proposed nonlinear technique to capture the temporal dynamics. In Figs. 15 and 16, the time variations of the mean and standard deviation are presented at the four selected gauging points. The results from NIROM-CAEs, POD-ANN, and the LHS solution can be compared. These plots show the increase in the water level from point 1 (close to the dam location) to point 4 (far from the dam), which allows an estimation of the arrival time of the flooding wave. Slight deviations can be observed in the mean profiles from the POD-ANN at almost all the gauging points, while the profiles from NIROM-CAEs present a good match with the reference solution, as shown in Fig. 15a



(a) $t \approx 5\,(s)$

(b) $t \approx 15\,(s)$

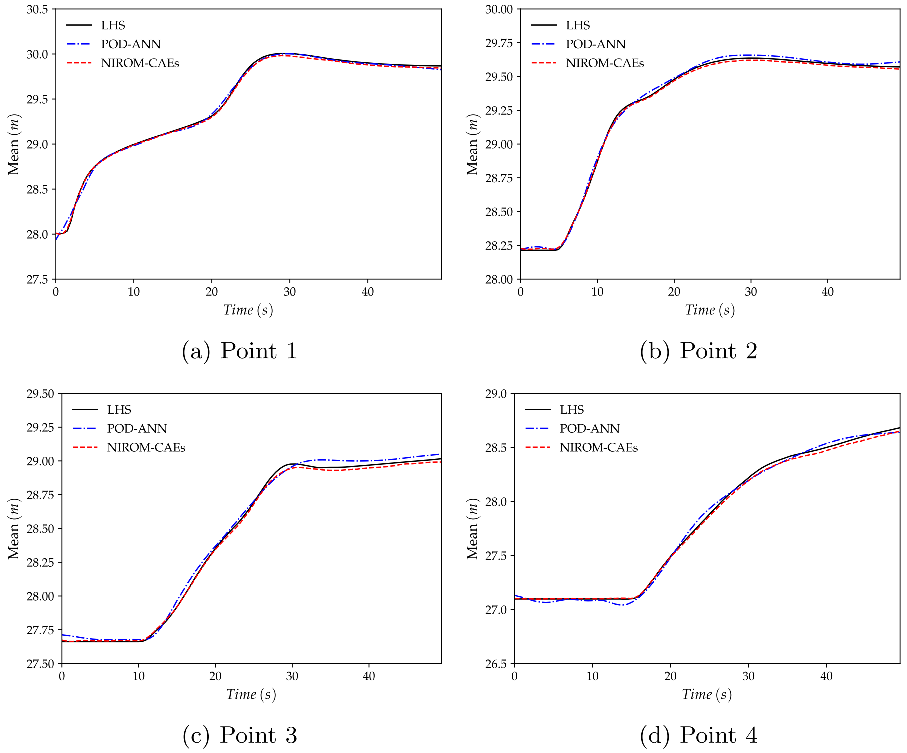
(c) $t \approx 30\,(s)$

(d) $t \approx 45\,(s)$

Fig. 14 Comparison of the standard deviation profiles of the water level over the cross-section line at various time-steps; the POD-ANN and NIROM-CAEs profiles vs those from the LHS reference solution (with 2000 realizations). **a**: $t \approx 5\,s$, **b**: $t \approx 15\,s$, **c**: $t \approx 30\,s$ and (**d**): $t \approx 45\,s$

through 15d. This tendency is further confirmed by the standard deviation profiles, where excellent superimposition can be observed between the profiles from the NIROM-CAEs and those from the LHS solution at all four gauging points. Conversely, the predictions from the linear POD-ANN technique present remarkable deviations with poor accuracy in the approximations of the standard deviation profiles of the water level at all gauging points, as indicated in Fig. 16a through 16d.

For a better insight into the accuracy of both reduced-order model approaches, namely POD-ANN and NIROM-CAEs, a quantitative evaluation is performed by computing the relative L^2-error over the whole computational domain (with the LHS reference sampling solution obtained with $N_s = 2000$ points). Thus, the relative error of the mean and standard deviation is represented as a function of time, as shown in Fig. 17. These plots show that the proposed NIROM-CAEs have much lower values of the relative error for both the mean and the standard deviation. The effect of the training set sample size (N_s) on the maximum relative error over time is reported in Table 1. Three values of the sample size $N_s = 30$, 90 and 300 were tested, and for each value, the maximum L^2-error was computed over time for both the mean and the standard deviation. It is clear that as the sample size increases, the relative error of the NIROM-CAEs decreases faster than that of the POD-ANN, particularly for the standard deviation, where a difference in the error values can reach almost two orders of magnitude. This quantitative analysis supports the above-mentioned observations concerning the predictive abilities of the proposed NIROM-CAEs. Another meaningful result from Table 1 is the dimension of



(a) Point 1

(b) Point 2

(c) Point 3

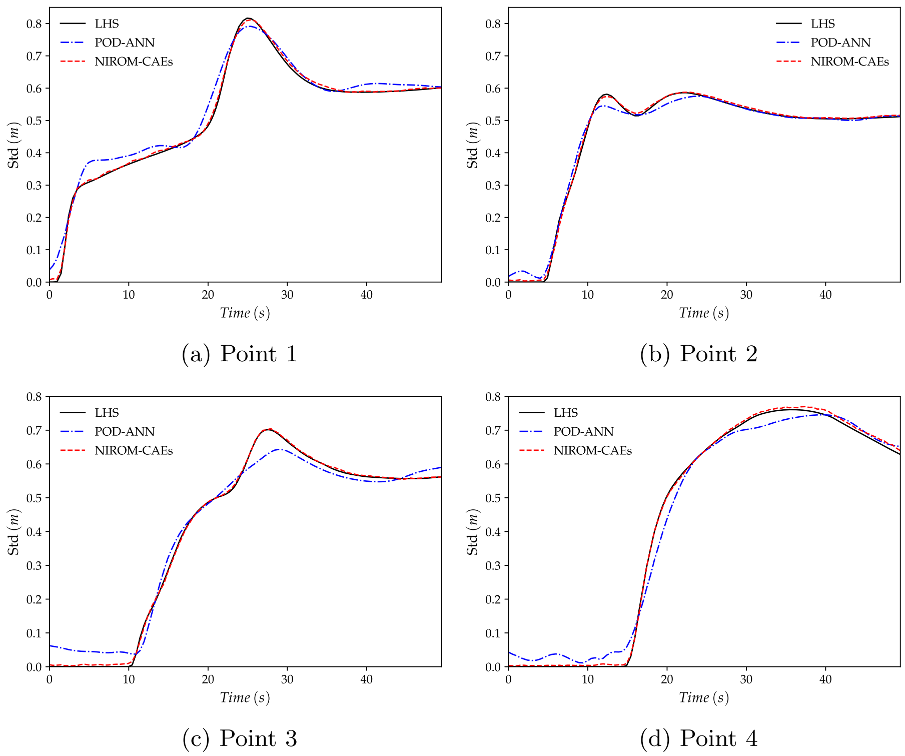
(d) Point 4

Fig. 15 Evolution of the mean profiles of the water level as a function of time at four gauging points. The profiles obtained with the NIROM-CAEs are compared to those from the POD-ANN technique and the LHS reference solution (with 2 000 realizations). **a**: Point 1, **b**: Point 2, **c**: Point 3 and **d**: Point 4

the generated latent space, which is considerably reduced for the NIROM-CAEs, on the order of $L_x = 50$ and $L_t = 10$, independently of the sample size, conversely to the POD-ANN, whose dimension of the reduced basis increases from $L_{POD} = 800$ for $N_s = 30$ to reach $L_{POD} = 3337$ for $N_s = 300$, which may present a challenging task for the available computational capacities.

Conclusion

This paper presents a non-intrusive reduced-order model based on convolutional autoencoders (NIROM-CAEs) for parameter-varying and time-dependent fluid problems. The model is fully data-driven and exploits the nonlinear framework provided by the convolutional autoencoders to tackle physical problems presenting a high degree of complexity. The construction of the offline training stage of the proposed approach consists of two successive compression levels along the spatial and temporal dimensions. The first encoder (a space encoder) performs compression along the spatial dimension of the input snapshot matrix to generate the encoded low-dimensional latent space. A second encoder (a time encoder), whose structure is similar to that of the former, applies convolutional operations along the temporal dimension of the encoded space to generate a second spatiotemporal latent vector with much lower dimensions. The encoded latent space thus obtained is used as the output data for a multilayer perceptron (MLP), deployed at the bottom level to train the mapping with the inputs from the parametric space. Once the offline training stage is constructed, the online predictive stage consists of the genera-



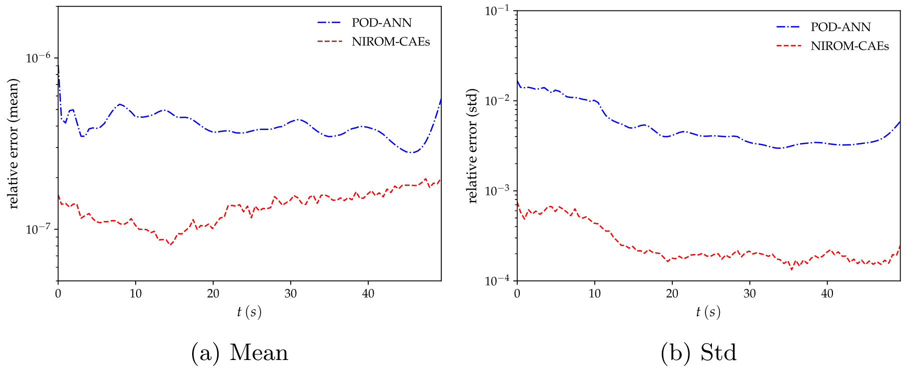
(a) Point 1

(b) Point 2

(c) Point 3

(d) Point 4

Fig. 16 Evolution of the standard deviation profiles of the water level as a function of time at four gauging points. The profiles obtained with the NIROM-CAEs are compared to those from the POD-ANN technique and the LHS reference solution (with 2000 realizations). **a**: Point 1, **b**: Point 2, **c**: Point 3 and **d**: Point 4



(a) Mean

(b) Std

Fig. 17 Comparison of the relative L^2-error of the mean and standard deviation of the water level as a function of time based on the POD-ANN technique and the proposed NIROM-CAEs approach. Errors are computed with respect to the reference LHS solution (with $N_s = 2000$ realizations)

tion of a new set of unseen points in the design space to provide the trained MLP at the bottom level. The resulting spatiotemporal latent vectors are then decoded through successive 1D decoders to predict solutions describing the high-dimensional spatiotemporal dynamics of the output quantities of interest. The proposed reduced-order model allows for rapid and accurate predictions of the statistical moments of the output distributions, thus enabling uncertainty propagation analyses.

The performance and accuracy of the proposed reduced-order model (NIROM-CAEs) are demonstrated on three nonlinear examples parametrized by a random variable with

Table 1 Effect of the sample size N_s on the maximum relative error in the L^2-norm for the mean and standard deviation obtained from the NIROM-CAEs ($L_x = 50$, $L_t = 10$) and POD-ANN ($\epsilon_s = \epsilon_t = 10^{-8}$) approaches. Errors are computed with respect to the LHS reference solution (with $N_s = 2000$ realizations)

N_s	L_{POD}	$Err^{max}_{L^2, Mean}$		$Err^{max}_{L^2, Std}$	
		POD-ANN	NIROM-CAEs	POD-ANN	NIROM-CAEs
30	860	8.2855E−07	2.1896E−06	0.024537	0.010349
90	2112	6.0781E−07	9.0018E−07	0.013303	0.005838
300	3387	9.1037E−07	1.9986E−07	0.016687	7.4981E−04

an appropriate variability range. The two first test cases concern the Burgers and Stoker's solutions, known as challenging test cases presenting a high hyperbolic behavior or a discontinuity accompanying the shock wave. The numerical results show that the model offers accurate approximations of the statistical moments in comparison to those of the reference solutions from the Latin Hypercube Sampling approach (LHS) method, unlike the linear POD-ANN model which shows some limitations in reproducing the dynamics of the outputs where an oscillatory behavior is observed in the predicted profiles. The results also reveal the low relative error of the proposed NIROM-CAEs, further highlighting its abilities. The model is then applied to a third case to analyze uncertainty propagation in a dam break flow over a real terrain. The NIROM-CAEs proved that its accuracy in the reconstruction of the mean and the standard deviation profiles, where good concordances with those from the LHS reference solutions were observed. The predictions from the POD-ANN show spurious oscillations, particularly for the temporal evolution at gauging points. Thus, the proposed non-intrusive reduced-order model based on convolutional autoencoders presents a powerful tool for nonlinear dimensionality reduction for parameter-varying, time-dependent, and large-scale problems characterized by strong hyperbolic behavior or even with the presence of a discontinuity. The model offers the construction of accurate surrogate predictions at a low cost.

A Training convergence history and models' configurations

See Tables 2, 3, 4, 5, 6 and Figs. 18, 19, 20.

Table 2 CAE-space architecture for Burger's and Stoker's test cases

Layer	Nb of filters	Kernel size	Activation function	Output shape
Encoder-space				
Input	–	–	–	1000×1
Conv-pool	32	3–2	PReLU	500×32
Conv-pool	64	3–2	PReLU	250×64
Conv-pool	128	3–5	PReLU	50×128
Flatten	–	–	–	6400
Dense	–	–	PReLU	60
Dense (output)	-	-	PReLU	$L_x = 50$
Decoder-space				
Input	–	–	–	$L_x = 50$
Dense	–	–	PReLU	60
Dense	–	–	PReLU	6400
Reshape	–	–	–	50×128
Conv-Upsamp	128	3–5	PReLU	250×128
Conv-Upsamp	64	3–2	PReLU	500×64
Conv-Upsamp	32	3–2	PReLU	1000×32
Conv (output)	1	3	PReLU	1000×1

Table 3 CAE-time architecture for Burger's and Stoker's test cases

Layer	Nb of filters	Kernel size	Activation function	Output shape
Encoder-time				
Input	–	–	–	$104 \times L_x$
Conv-pool	32	3–2	PReLU	52×32
Conv-pool	64	3–2	PReLU	26×64
Conv-pool	128	3–2	PReLU	13×128
Flatten	–	–	–	1 664
Dense (output)	–	–	PReLU	$L_t = 10$
Decoder-time				
Input	–	–	–	$L_t = 10$
Dense	–	–	PReLU	1664
Reshape	–	–	–	13×128
Conv-Upsamp	128	3–2	PReLU	26×128
Conv-Upsamp	64	3–2	PReLU	52×64
Conv-Upsamp	32	3–2	PReLU	104×32
Conv (output)	1	3	PReLU	$104 \times L_x$

Table 4 CAE-space architecture for hypothetical dam break of Miles-Iles river

Layer	Nb of filters	Kernel size	Activation function	Output shape
Encoder-space				
Input	–	–	–	$10,200 \times 1$
Conv-pool	32	3–5	PReLU	2040×32
Conv-pool	64	3–5	PReLU	408×64
Flatten	–	–	–	26112
Dense	-	–	PReLU	120
Dense (output)	–	–	PReLU	$L_x = 50$
Decoder-space				
Input	–	–	–	$L_x = 50$
Dense	–	–	PReLU	120
Dense	–	–	PReLU	26 112
Reshape	–	–	–	408×64
Conv-Upsamp	64	3–5	PReLU	2040×64
Conv-Upsamp	32	3–5	PReLU	10200×32
Conv (output)	1	3	PReLU	10200×1

Table 5 CAE-time architecture for hypothetical dam break of Miles-Iles river

Layer	Nb of filters	Kernel size	Activation function	Output shape
Encoder-time				
Input	–	–	–	$100 \times L_x$
Conv-pool	32	3–2	PReLU	50×32
Conv-pool	64	3–2	PReLU	25×64
Conv-pool	128	3–5	PReLU	5×128
Flatten	–	–	–	640
Dense (output)	–	–	PReLU	$L_t = 10$
Decoder-time				
Input	–	–	–	$L_t = 10$
Dense	–	–	PReLU	640
Reshape	–	–	–	5×128
Conv-Upsamp	128	3–5	PReLU	25×128
Conv-Upsamp	64	3–2	PReLU	50×64
Conv-Upsamp	32	3–2	PReLU	100×32
Conv (output)	1	3	PReLU	$100 \times L_x$

Table 6 MLP architecture for Burgers, Stoker's test cases and hypothetical dam break

Layer	Activation function	Output shape
MLP		
Input	ReLU	1
Dense	ReLU	128
Dense	ReLU	128
Dense	ReLU	128
Dense (output)	Linear	$L_t = 10$

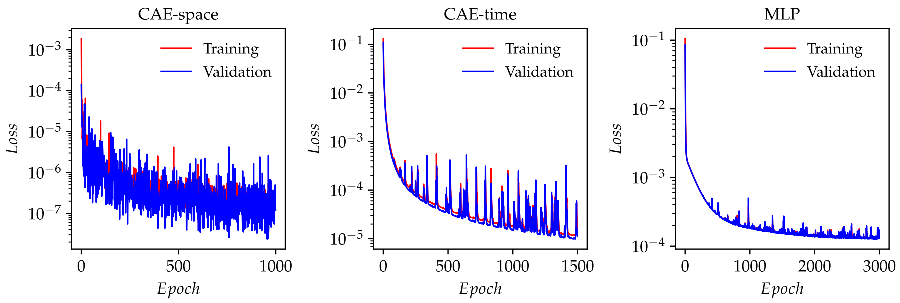


Fig. 18 Evolution of the training and validation losses of CAEs and MLP for the Burgers equation test case

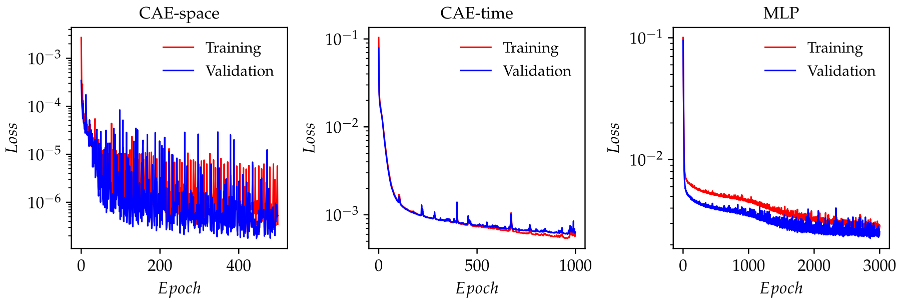


Fig. 19 Evolution of the training and validation losses of CAEs and MLP for the Stoker's analytical solution test case

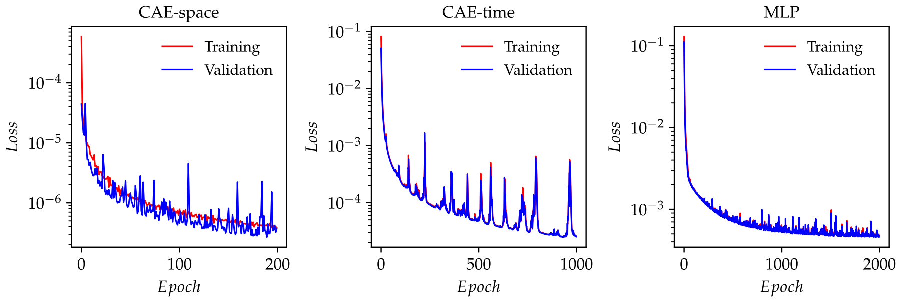


Fig. 20 Evolution of the training and validation losses of CAEs and MLP for the hypothetical dam break of the Mille-Iles river

Author contributions

AA: Methodology, Software, Validation, Writing an original draft. AS: Methodology, review and editing, Project administration, Funding acquisition, Validation, Supervision. Both authors read and approved the final manuscript.

Funding

This work is supported by the Natural Sciences and Engineering Research Council of Canada (Grant number RGPIN/2693-2021).

Availability of data and materials

The software can be shared upon request.

Declarations

Competing interests

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.

Received: 5 January 2023 Accepted: 27 April 2023
Published online: 19 May 2023

References

1. Rezaeiravesh S, Vinuesa R, Schlatter P. An uncertainty-quantification framework for assessing accuracy, sensitivity, and robustness in computational fluid dynamics. J Comput Sci. 2022;62: 101688.
2. Kalinina A, Spada M, Vetsch DF, Marelli S, Whealton C, Burgherr P, Sudret B. Metamodeling for uncertainty quantification of a flood wave model for concrete dam breaks. Energies. 2020;13(14):3685.
3. Quarteroni A, Manzoni A, Negri F. Reduced basis methods for partial differential equations: an introduction, vol. 92. Berlin: Springer; 2015.
4. Berkooz G, Holmes P, Lumley JL. The proper orthogonal decomposition in the analysis of turbulent flows. Annu Rev Fluid Mech. 1993;25(1):539–75.
5. Sirovich L. Turbulence and the dynamics of coherent structures. i. coherent structures. Quart Appl Math. 1987;45(3):561–71.
6. Chatterjee A. An introduction to the proper orthogonal decomposition. Curr Sci. 2000;78:808–17.
7. Zokagoa J-M, Soulaïmani A. 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. 2012;221:1–23.
8. Xiao D, Fang F, Pain CC, Navon IM. A parameterized non-intrusive reduced order model and error analysis for general time-dependent nonlinear partial differential equations and its applications. Comput Methods Appl Mech Eng. 2017;317:868–89.
9. Walton S, Hassan O, Morgan K. Reduced order modelling for unsteady fluid flow using proper orthogonal decomposition and radial basis functions. Appl Math Model. 2013;37(20–21):8930–45.
10. Xiao M, Breitkopf P, Coelho RF, Knopf-Lenoir C, Sidorkiewicz M, Villon P. Model reduction by cpod and kriging: application to the shape optimization of an intake port. Struct Multidiscip Optim. 2010;41(4):555–74.
11. Sun X, Choi J-I. Non-intrusive reduced-order modeling for uncertainty quantification of space-time-dependent parameterized problems. Comput Math Appl. 2021;87:50–64.
12. El Moçayd N, Mohamed MS, Ouazar D, Seaid M. Stochastic model reduction for polynomial chaos expansion of acoustic waves using proper orthogonal decomposition. Reliabil Eng Syst Saf. 2020;195: 106733.
13. Abdedou A, Soulaïmani A. A non-intrusive reduced-order modeling for uncertainty propagation of time-dependent problems using a b-splines bézier elements-based method and proper orthogonal decomposition: Application to dam-break flows. Comput Math Appl. 2021;102:187–205.
14. Jacquier P, Abdedou A, Delmas V, Soulaïmani A. Non-intrusive reduced-order modeling using uncertainty-aware deep neural networks and proper orthogonal decomposition: Application to flood modeling. J Comput Phys. 2021;424: 109854.
15. Hesthaven JS, Ubbiali S. Non-intrusive reduced order modeling of nonlinear problems using neural networks. J Comput Phys. 2018;363:55–78.
16. Wang Q, Hesthaven JS, Ray D. Non-intrusive reduced order modeling of unsteady flows using artificial neural networks with application to a combustion problem. J Comput Phys. 2019;384:289–307.
17. Taddei T, Perotto S, Quarteroni ALFIO. Reduced basis techniques for nonlinear conservation laws. ESAIM Math Modell Numer Anal. 2015;49(3):787–814.
18. Ohlberger M, Rave S. Reduced basis methods: success, limitations and future challenges. arXiv preprint arXiv:1511.02021, 2015.
19. N S, Kalogeris I, P V. Non-intrusive surrogate modeling for parametrized time-dependent partial differential equations using convolutional autoencoders. Eng Appl Artif Intell. 2022;109: 104652.
20. LeCun Y, Bengio Y, Hinton G. Deep learning. Nature. 2015;521(7553):436–44.
21. Zhu J, Shi H, Song B, Tao Y, Tan S. Information concentrated variational auto-encoder for quality-related nonlinear process monitoring. J Process Control. 2020;94:12–25.
22. Zhu J, Shi H, Song B, Tan S, Tao Y. Deep neural network based recursive feature learning for nonlinear dynamic process monitoring. Can J Chem Eng. 2020;98(4):919–33.
23. Fresca S, Dede' L, Manzoni A. A comprehensive deep learning-based approach to reduced order modeling of nonlinear time-dependent parametrized pdes. J Sci Comput. 2021;87:1–36.
24. Kim Y, Choi Y, Widemann D, Zohdi T. A fast and accurate physics-informed neural network reduced order model with shallow masked autoencoder. J Comput Phys. 2022;451: 110841.
25. Otto SE, Rowley CW. Linearly recurrent autoencoder networks for learning dynamics. SIAM J Appl Dynam Syst. 2019;18(1):558–93.
26. Halder R, Fidkowski K, Maki K. Non-intrusive reduced-order modeling using convolutional autoencoders. arXiv preprint arXiv:2204.08280, 2022.
27. He Y, Shi H, Tan S, Song B, Zhu J. Multiblock temporal convolution network-based temporal-correlated feature learning for fault diagnosis of multivariate processes. J Taiwan Inst Chem Eng. 2021;122:78–84.
28. Gonzalez Francisco J, Balajewicz M. Deep convolutional recurrent autoencoders for learning low-dimensional feature dynamics of fluid systems. arXiv preprint arXiv:1808.01346, 2018.
29. Eivazi H, LeClainche S, Hoyas S, Vinuesa R. Towards extraction of orthogonal and parsimonious non-linear modes from turbulent flows. Expert Syst Appl. 2022;202: 117038.
30. Theodoridis S. Machine learning: a Bayesian and optimization perspective. USA: Academic Press; 2015.
31. Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput. 1997;9(8):1735–80.
32. Jiayang X, Duraisamy K. Multi-level convolutional autoencoder networks for parametric prediction of spatio-temporal dynamics. Comput Methods Appl Mech Eng. 2020;372: 113379.
33. Dutta S, Rivera-Casillas P, Styles B, Farthing MW. Reduced order modeling using advection-aware autoencoders. Math Comput Appl. 2022;27(3):34.
34. Bhatt P, Kumar Y, Soulaimani A. Deep convolutional architectures for extrapolative forecast in time-dependent flow problems. arXiv preprint arXiv:2209.09651, 2022.
35. Elhassouny A, Smarandache F. Trends in deep convolutional neural networks architectures: a review. In 2019 International Conference of Computer Science and Renewable Energies (ICCSRE), pages 1–8. IEEE, 2019.

36. Maulik R, Lusch B, Balaprakash P. Reduced-order modeling of advection-dominated systems with recurrent neural networks and convolutional autoencoders. Phys Fluids. 2021;33(3): 037106.
37. Westermann P, Welzel M, Evins R. Using a deep temporal convolutional network as a building energy surrogate model that spans multiple climate zones. Appl Energy. 2020;278: 115563.
38. Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Devin M, Ghemawat S, Irving G, Isard M et al. Tensorflow: a system for large-scale machine learning. In: Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI). Savannah, Georgia, USA, 2016.
39. Fu R, Xiao D, Navon IM, Wang C. A data driven reduced order model of fluid flow by auto-encoder and self-attention deep learning methods. arXiv preprint arXiv:2109.02126, 2021.
40. Burgers JM. A mathematical model illustrating the theory of turbulence. In: Advances in applied mechanics, vol. 1. Amsterdam: Elsevier; 1948. p. 171–99.
41. Guo M, Hesthaven JS. Data-driven reduced order modeling for time-dependent problems. Comput Methods Appl Mech Eng. 2019;345:75–99.
42. Stoker JJ. Water waves: the mathematical theory with applications. Interscience, New York. 1957;2:5.
43. Delestre O, Lucas C, Ksinant P-A, Darboux F, Laguerre C, Vo TNT, James F, Cordier S. Swashes: a compilation of shallow water analytic solutions for hydraulic and environmental studies. Int J Numer Methods Fluids. 2013;72(3):269–300.
44. Seyedashraf O, Akhtari AA. Two-dimensional numerical modeling of dam-break flow using a new tvd finite-element scheme. J Brazil Soc Mech Sci Eng. 2017;39(11):4393–401.
45. Delmas V, Soulaïmani A. Multi-gpu implementation of a time-explicit finite volume solver using cuda and a cuda-aware version of openmpi with application to shallow water flows. Comput Phys Commun. 2022;271: 108190.

Publisher’s Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.