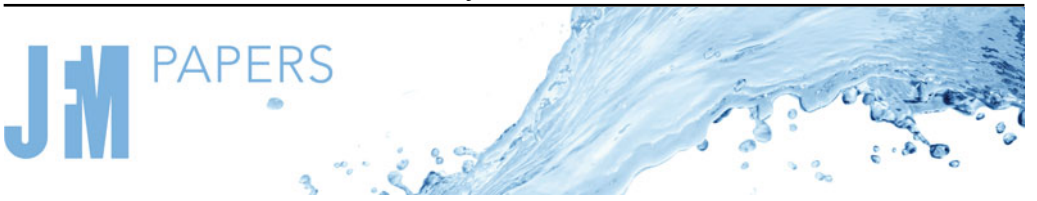
J. Fluid Mech. (2026), *vol.* 1046, A10, doi:10.1017/jfm.2026.12075



The balance between compactness and forecast accuracy of data-driven latent-space reduced-order models in controlled wake flows

Alberto Solera-Rico[1,2], **Patricia García-Caspueñas**[3], **Carlos Sanmiguel Vila**[1,2] and **Stefano Discetti**[2]

[1]Sub-Directorate General of Aeronautical Systems, Spanish National Institute for Aerospace Technology (INTA), ctra. M-301, km. 10.5, San Martín de la Vega, Madrid 28330, Spain

[2]Departamento de Ingeniería Aeroespacial, Universidad Carlos III de Madrid, Avenida de la Universidad 30, Madrid, Leganés 28911, Spain

[3]Department of Mechanical Engineering, University of Washington, Seattle, WA 98195, USA

Corresponding authors: Stefano Discetti, sdiscett@ing.uc3m.es; Alberto Solera-Rico, alberto.solera@alumnos.uc3m.es

(Received 9 January 2026 UTC; revised 24 July 2026 UTC; accepted 15 August 2026 UTC)

Model-based active flow control requires predictive models that are accurate, stable and fast enough for real-time optimisation. In controlled wake flows, this is often achieved through reduced-order models (ROMs) that first compress high-dimensional velocity snapshots into a latent space and then learn a time-stepping predictor for the dynamics in the latent space. Here, we study how the choice of the spatial encoder affects the predictability of the resulting latent coordinates for wake flows under control inputs. Using two actuated two-dimensional wake configurations, a simplified truck wake and the fluidic pinball, we compare proper orthogonal decomposition (POD) against nonlinear convolutional autoencoders (CAEs) and two types of variational autoencoders for compression, and evaluate several temporal predictors based on long short-term memory networks. The CAEs achieve higher compression efficiency and sharper short-term reconstructions, but they produce a latent dynamics that is more irregular and with broadband spectral content. As a consequence, long-horizon forecasts degrade faster and show a higher probability of catastrophic divergence than POD-based models. The POD yields smoother latent trajectories that are easier to learn and extrapolate, leading to more reliable predictions beyond the short-term regime. These results reveal a clear trade-off between compactness and forecast accuracy, and suggest that the stability of

© The Author(s), 2026. Published by Cambridge University Press. This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (https://creativecommons.org/licenses/by/4.0/), which permits unrestricted re-use, distribution and reproduction, provided the original article is properly cited.

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

the latent dynamics prediction can outweigh maximal compression. This is particularly relevant for control strategies rooted in forecasts of the dynamics, such as model predictive control and reinforcement learning. The findings provide practical guidance for designing actuation-aware, hardware-feasible predictive ROMs for real-time flow control.

Key words: flow control, machine learning, low-dimensional models

1. Introduction

Active flow control (AFC) has emerged as a relevant strategy to enhance the aerodynamic performance of ground and air vehicles (Brunton & Noack 2015). By manipulating flow separation and wake structures using actuators such as synthetic jets, plasma devices or moving surfaces, significant performance gains can be achieved (Greenblatt & Wygnanski 2000; Glezer & Amitay 2002; Cattafesta & Sheplak 2011). For ground vehicles, the efficacy of open-loop actuation has been well demonstrated in wind tunnel studies on simplified bluff bodies, where steady and pulsed blowing can substantially recover base pressure and reduce drag (Littlewood & Passmore 2012; McNally *et al.* 2015; Cerutti *et al.* 2020; Amico *et al.* 2022, 2024; Robledo *et al.* 2026).

Feedback control promises further significant improvements (Brunton & Noack 2015). Real-world conditions are inherently unsteady, and an effective AFC system must be able to sense the current flow state, predict its evolution and apply an optimal control action that is adapted in real time. To meet these demands, data-driven methods and machine learning are now opening new research avenues (Brunton *et al.* 2020). Broadly, these approaches fall into two categories: model-free controllers that learn a policy directly from data, and model-based controllers that exploit an explicit dynamical model.

Model-free strategies can be attractive when little is known *a priori* about the system. Such approaches have shown promising results in low-Reynolds-number or numerically simulated flows where large amount of training data are generated in a cheap and safe manner (Garcia *et al.* 2025; Suárez *et al.* 2025). However, model-free methods are often sample inefficient, provide less transparent guarantees on stability and safety and struggle to incorporate hard constraints on actuators or states. These limitations are particularly restrictive in experimental AFC, where data are expensive and hardware must operate within strict bounds. This motivates a focus on model-based approaches, where control decisions are derived from an explicit predictive model.

A typical example of model-based control is model predictive control (MPC) (Camacho & Bordons 2013; Rawlings *et al.* 2017). At each time step, MPC leverages a model of the dynamics to forecast future flow states over a finite horizon and solves an optimisation problem to obtain the optimal control sequence. Compared with purely model-free strategies, this framework is more sample efficient, offers clearer routes to stability and safety guarantees and handles constraints naturally. The primary challenge is therefore the development of a predictive model that is both sufficiently accurate to capture the controlled dynamics and computationally efficient for real-time execution (Kaiser *et al.* 2018; Bieker *et al.* 2020; Marra *et al.* 2024*b*; Liu *et al.* 2025; Solera-Rico *et al.* 2026).

The effectiveness of MPC ultimately hinges on the reliability of the plant model. In turbulent flows, the difficulty is not only to approximate the unforced dynamics, but also to represent how actuation reshapes the accessible state space. Recent studies show that controlled wakes often evolve on low-dimensional, actuation-dependent manifolds (Marra *et al.* 2024*a*). Reduced coordinates learned without accounting for control inputs may thus

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

yield a latent dynamics that is unnecessarily complex or even inconsistent with feasible actuation commands. This perspective motivates the need to develop actuation-aware reduced-order predictors that remain simple enough for receding-horizon optimisation while capturing the key controlled dynamics.

For this reason, this work focuses on the central challenge of building predictive models suitable for model-based control. While being a gold standard and highly informative in predictive-control implementations (Bewley *et al.* 2001), full-order simulations are computationally intractable in real-time loops. We thus focus on a framework based on reduced-order models (ROMs), with two stages: (i) compress the high-dimensional velocity field into a low-dimensional latent space (the encoding stage), and (ii) learn a time-stepping model that predicts the latent evolution under actuation (the prediction step). This approach has been recently explored by several studies with data-driven methods (Maulik *et al.* 2021; Bukka *et al.* 2021; Solera-Rico *et al.* 2024). This separation is consistent with many classical ROM pipelines, where the compression step is based on proper orthogonal decomposition (POD): since POD is a purely energy-based decomposition, the resulting latent basis is not explicitly optimised jointly with the subsequent dynamical estimator. While the merits of such techniques are already apparent, we identify two research gaps. On the one hand, the generalisation capabilities of these methods under different exogenous inputs are not clear, and a recipe for training and predicting under environments with active control has not been consolidated yet. On the other hand, there is a growing tendency towards using powerful encoders, while it is not clear if an Occam's razor solution could be more effective under some configurations. Wake flows, for instance, often evolve on low-dimensional attractors, with a rather predictable dynamics. In flows dominated by vortex shedding, the leading POD coefficients often exhibit a narrow-band, nearly oscillatory dynamics because POD isolates the most energetic coherent structures. In special cases, such as nearly periodic flows, these structures may align with Koopman-related spectral components (Rowley *et al.* 2009; Mezić 2013; Taira *et al.* 2017). However, it must be remarked that, even though POD is a linear decomposition, it does not imply that a linearised dynamics is achievable in its reduced space. Therefore, any analogy between the dynamics of the leading POD coefficients and Koopman modes should be understood as problem-dependent rather than intrinsic to POD. On the other hand, convolutional autoencoders (CAEs) can be driven to approximate Koopman operators (Lusch *et al.* 2018), although, generally speaking, training driven by reconstruction compactness does not entail robustness guarantees on the dynamics. Existing comparisons between POD and CAE have been targeted already for systems with autonomous dynamics (Fresca *et al.* 2021; Fresca & Manzoni 2022); however, the interaction between the encoder choice and multi-input actuation, on actuation-dependent manifolds, remains less explored.

To address the gaps above, we compare linear and nonlinear options for encoding, considering wake flow systems under control actions. For encoding, we contrast the classical energy-optimal POD (Sirovich 1987) with nonlinear CAEs (Solera-Rico *et al.* 2024), which typically yield more compact latent representations but may induce a more complex controlled latent dynamics. For temporal prediction, we employ long short-term memory (LSTM) networks (Hochreiter & Schmidhuber 1997), which are well suited for sequential latent dynamics. Using two two-dimensional (2-D) wake flows, a simplified truck model (Solera-Rico *et al.* 2026) and the fluidic pinball (Deng *et al.* 2020), we quantify the trade-off between compactness and forecast accuracy of the latent-space dynamics, and show how the preferred reduction strategy depends on the prediction horizon required by the controller.

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

Parameter	Simplified truck	Fluidic pinball
Reynolds number	500	150
Control type	Single-input (synthetic jets)	Multi-input (rotating cylinders)
Control signal	Filtered random	Quasi-stationary step ramps
Control dimension	1	3
Total snapshots	50 000	70 000
Test snapshots	5000	20 000
Time step (Δt)	$0.2t_c$	$0.1t_c$
Spatial resolution ($N_y \times N_x$)	128×256	96×192

Table 1. Summary of the datasets for the two flow configurations.

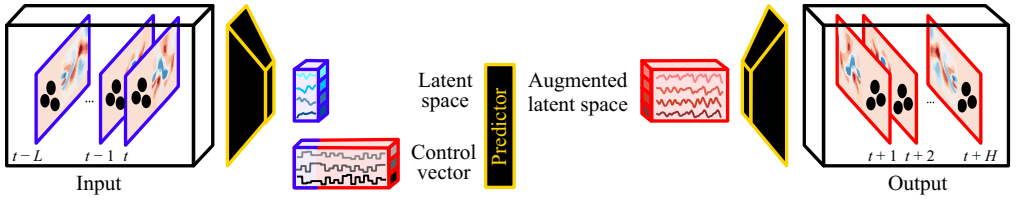


Figure 1. Schematic of the data-driven reduced-order modelling framework. High-dimensional input velocity fields are compressed into a low-dimensional latent-space time series using an encoder (POD or CAE). This latent state, along with the corresponding control sequence, is fed into a temporal predictor to infer the future evolution of the latent space. Finally, a decoder reconstructs the predicted latent states back into the high-dimensional, full-field velocity state space.

The paper is organised as follows. Section 2 provides a description of the methodology, including the generation of data for the chosen test cases. The analysis of the model results is provided in § 3. Finally, the conclusions are discussed in § 4.

2. Methodology

This section details the methodology to develop and evaluate predictive ROMs. First, we describe the two flow configurations and the generation of their respective high-fidelity direct numerical simulation (DNS) datasets. Next, we present the two methods used to encode the high-dimensional spatial fields into a low-dimensional latent space. Finally, in § 2.3, we detail three approaches based on LSTM network architectures for predicting the temporal evolution of this latent space in response to control inputs. The proposed ROM framework is illustrated schematically in figure 1.

2.1. *Flow configurations and data generation*

Two different 2-D wake flow configurations are used to assess the control architectures. These cases are chosen to represent different control challenges: the first is a single-input system, while the second is a more complex multi-input problem. Both datasets are created using DNSs with open-loop control signals designed to excite the underlying flow dynamics. In low-Reynolds-number flows, DNS provides a simulation framework with minimal numerical approximations to precisely represent the flow dynamics and capture all scale features. The following sections describe the set-up for each case in detail, and a summary of the datasets can be found in table 1.

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

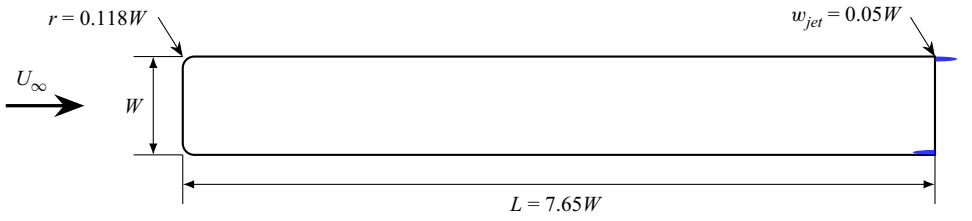


Figure 2. Schematic of the simplified truck flow configuration. Zero-net-mass flow jets are illustrated in blue.

2.1.1. *Simplified 2-D truck model*

This first case is a simplified model of a wake flow around a road vehicle. Figure 2 depicts the flow configuration based on the horizontal mid-plane geometry of the ground transportation system (Storms *et al.* 2001), a standard truck model also studied in recent work (McArthur *et al.* 2016). The model under study is a rectangular bluff body with a reference non-dimensional width $W = 1$, length $L = 7.647W$ and rounded leading edges with radius $r = 0.118W$. The inflow velocity is uniform with a magnitude of U_∞, and the Reynolds number, defined as $Re = U_\infty W/\nu$, where ν represents the fluid kinematic viscosity, is set to 500. The rectangular computational domain extends across the range $(-7W, 23W)$ in the streamwise direction (x) and $(-7.5W, 7.5W)$ in the transverse direction (y), with the front of the bluff body placed at $x = 0$. The simulation domain employs a hybrid mesh, structured near the wall and unstructured elsewhere, encompassing around 168 000 cells. The time is scaled using the convective time $t_c = W/U_\infty$. The DNS is executed in OpenFOAM, using the Gym-preCICE (Shams & Elsheikh 2023) wrapper for the coupling library (Chourdakis *et al.* 2022) and the OpenFOAM adapter (Chourdakis *et al.* 2023) to integrate the simulation with the specified control sequence. This Reynolds number is chosen as a representative bluff-body wake configuration to develop and validate control-oriented ROMs before scaling to higher-Re turbulent regimes. At $Re = 500$ the wake already exhibits the features relevant to this study, namely vortex shedding and sensitivity to base actuation, while the 2-D flow assumption remains valid, as verified in advance with a 3-D simulation. The compactness–predictability trade-off identified in this work, rooted in the spectral organisation of POD versus the broadband character of the CAE latent dynamics, is not expected to be specific to this Reynolds-number regime; an explicit assessment at higher Re and in 3-D configurations is left for future work.

The control is implemented using two opposing flow jets at the sides of the vehicle base with zero-net-mass flow, similar to previous experimental set-ups (Barros *et al.* 2016; Cerutti *et al.* 2020). In this case, the overall mass flow rate of the two jets combined is forced to be zero at each instant of the simulation. The jets, each with a width of $w_{jet} = 0.05W$, exhibit parabolic velocity profiles with a maximum mean velocity of $1.5U_\infty$. The dataset comprises a time series of velocity fields, with a time increment of $\Delta t = t_c/5 = W/5U_\infty$. Figure 3 illustrates the global mesh and a flow sample around the jet region.

The control input for the simplified truck configuration was generated using a filtered-noise excitation strategy to ensure a rich broadband excitation. First, a discrete white Gaussian noise sequence was generated

$$\xi_n \sim \mathcal{N}(0, 1), \tag{2.1}$$

where $n \in \mathbb{N}_0$ represents the discrete time step index. This index corresponds to the physical time $t_n = n\Delta t$, with a control update interval of $\Delta t = 0.2t_c$.

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

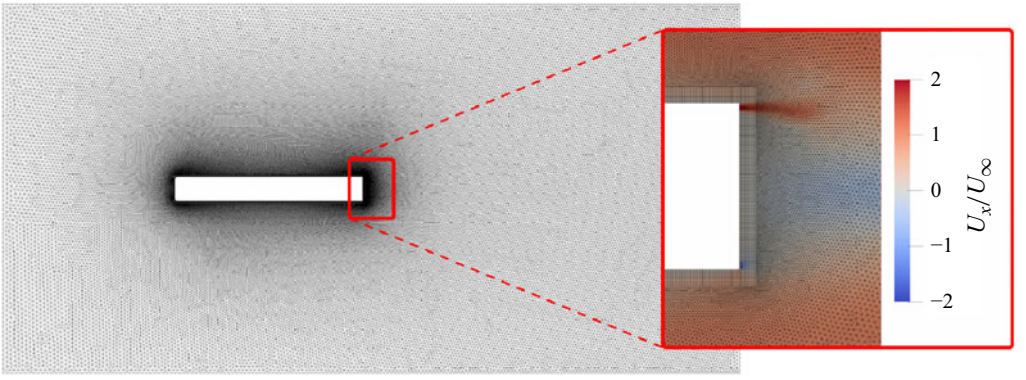


Figure 3. Direct numerical simulation mesh and flow details near the base of the truck when jets operate at peak suction/blowing. The symbol U_x in the inset indicates the streamwise velocity component. Figure adapted from Solera-Rico *et al.* (2026).

To target the relevant flow dynamics, this stochastic sequence was processed through a fourth-order Butterworth bandpass filter, denoted by the operator $\mathcal{B}_4$

$$\tilde{\xi}_n = \mathcal{B}_4\left(\xi_n;\, f_{low},\, f_{high}\right). \tag{2.2}$$

The filter cutoff frequencies were set to $f_{low} = 0.05t_c^{-1}$ and $f_{high} = 0.5t_c^{-1}$ to encompass the wake's natural shedding frequency of $f_{sh} \approx 0.2t_c^{-1}$. The filter was applied in a zero-phase (forward–backward) configuration to prevent distortion.

Finally, the filtered sequence $\tilde{\xi}_n$ was bounded to reflect the physical limits of the synthetic jets and scaled to determine the applied jet flow rate, $b(t_n)$

$$b(t_n) = \dot{m}_{max} \min\left(1, \max\left(-1, \tilde{\xi}_n\right)\right), \tag{2.3}$$

where the maximum volumetric flow rate capacity of the actuator is $\dot{m}_{max} = 0.075U_\infty W$. During the simulation, the environment linearly interpolated the commanded values between the discrete points t_n and t_{n+1} to provide a continuous forcing signal to the DNS solver.

Following the simulation, a data preprocessing pipeline was executed to convert the raw OpenFOAM output into a structured dataset. The velocity fields (with u, v being respectively the streamwise and cross-wise velocity components) were first interpolated onto a uniform Cartesian grid of 128×256 points using cubic interpolation. This grid spans the domain from $x/W = -6$ to $x/W = 22$ and $y/W = -7$ to $y/W = 7$. A mask was applied to set to zero the velocity in the grid points inside the truck body. The velocity fields $\boldsymbol{U}(\boldsymbol{x}, t)$ were then normalised by subtracting the temporal mean field, $\bar{\boldsymbol{U}}(\boldsymbol{x})$, and dividing the components u and v by their respective global scalar standard deviations (σ_u, σ_v), calculated throughout the entire dataset. In parallel, the actual jet flow rate was extracted from post-processing files and time synchronised with the velocity snapshots. The final 10 % snapshots of the long sequence were used for testing, with the remaining used for training.

2.1.2. *Fluidic pinball*

The second test case is the fluidic pinball, a canonical 2-D set-up often used for complex multi-input multi-output control problems (Deng *et al.* 2020), schematised in figure 4. The geometry consists of three cylinders of equal diameter $D = 2R$, with their centres located at the vertices of an equilateral triangle. The centre-to-centre side length of the

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

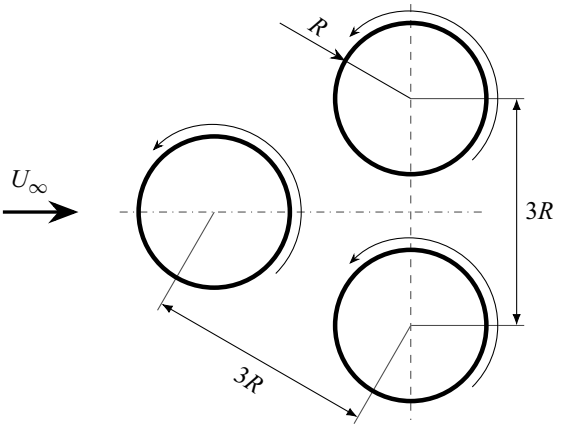


Figure 4. Schematic of the fluidic pinball flow configuration.

triangle is $3R$. The origin of the reference system is located in the midpoint between the two centres of the downstream cylinders, with the x axis pointing in the direction and the sense of the free-stream velocity U_∞ and the y axis perpendicular to it in the transverse direction. The rectangular computational domain extends from $(-5D, 15D)$ in the streamwise (x) direction and $(-5D, 5D)$ in the transverse (y) direction.

The dataset for this configuration was generated with a solver based on an implicit time integration finite element method. The Reynolds number is set to $Re = 150$, higher than the transition point $Re \simeq 115$ from a quasi-periodic asymmetric regime to chaotic flow (Deng *et al.* 2020, 2025). This regime exhibits a dominant vortex shedding with random upward and downward switches in the centre jet, remaining as a challenging dynamical system. The time is scaled using the convective time $t_c = D/U_\infty$. The resulting dataset contains the 2-D velocity fields (u, v) for each time step, normalised with the free-stream velocity U_∞. Control is applied by changing the tangential speed of the three cylinder surfaces, providing three independent control parameters. Notice that the input speeds are normalised with U_∞ and defined positive when the rotation is set to be counter-clockwise.

The control set-up for the fluidic pinball configuration was designed as a multi-input strategy using quasi-stationary step ramps to excite the dynamics of the system. The control inputs are defined by the normalised tangential speeds of the three cylinders, $\boldsymbol{b}(t) = [b_1(t), b_2(t), b_3(t)]^\top$, corresponding to the top, bottom and forward cylinders, respectively.

To cover the control spectrum, target speed combinations, denoted as $\boldsymbol{v}_k$, were generated from the 125 possible permutations of discrete rotational speeds

$$\boldsymbol{v}_k \in \mathcal{V}^3, \quad \text{where} \quad \mathcal{V} = \{-1, -0.5, 0, 0.5, 1\}. \tag{2.4}$$

For a given target step k, each cylinder $i \in \{1, 2, 3\}$ maintains its target speed $v_{k,i}$ for a constant hold duration of $T_{hold,i}$. Following the hold phase, the speed transitions linearly to the subsequent target value $v_{k+1,i}$ over a cylinder-specific ramp duration $T_{ramp,i}$.

Mathematically, the control signal for cylinder i during the transition to the $(k+1)$th step and its subsequent hold phase can be described by the piecewise function

$$b_i(t) = \begin{cases} v_{k,i} + \dfrac{v_{k+1,i} - v_{k,i}}{T_{ramp,i}}(t - t_{k,i}), & \text{if } t_{k,i} \le t < t_{k,i} + T_{ramp,i}, \\ v_{k+1,i}, & \text{if } t_{k,i} + T_{ramp,i} \le t \le t_{k+1,i}, \end{cases} \tag{2.5}$$

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

where $t_{k,i}$ marks the beginning of the kth ramp for cylinder i, and $t_{k+1,i} = t_{k,i} + T_{ramp,i} + T_{hold,i}$ is the beginning of the next cycle. The transition durations vary for each actuator, defined as $T_{ramp,1} = 25\,t_c$ for the top cylinder, $T_{ramp,2} = 40\,t_c$ for the bottom cylinder and $T_{ramp,3} = 120\,t_c$ for the forward cylinder. Similarly, the hold durations are also specific to each cylinder, i.e. $T_{hold,1} = 30\,t_c$, $T_{hold,2} = 235\,t_c$ and $T_{hold,3} = 1250\,t_c$.

This dataset contains a total of 70 000 snapshots, captured with a time spacing of 0.1 convective times between each snapshot. The final 20 000 snapshots are reserved for testing, with the first 50 000 used for training. Following the simulation, each snapshot of the velocity field was preprocessed and interpolated onto a regular grid, resulting in a final resolution of 96×192 pixels. As with the previous flow configuration, a zero velocity mask is applied to the inner body regions. Both POD and CAE operate on the fluctuating velocity components $\boldsymbol{u}(\boldsymbol{x}, t) = \boldsymbol{U}(\boldsymbol{x}, t) - \bar{\boldsymbol{U}}(\boldsymbol{x})$, normalised by their respective global standard deviations.

2.2. *Dimensionality reduction methods*

The datasets consist of long time series of full-field velocity field snapshots under different control actions. Attempting to directly predict the temporal evolution of the entire high-dimensional spatial field, $\boldsymbol{U}(\boldsymbol{x}, t)$, is computationally intractable for real-time control applications. The core strategy is therefore to first reduce the high spatial dimensionality of each snapshot into a low-dimensional latent-space representation, $\boldsymbol{z}(t) \in \mathbb{R}^d$, where $d \ll N_p$ is the latent dimension, and N_p the number of grid points. The subsequent task, detailed in § 2.3, is to build a predictive model that can forecast the temporal evolution of this much smaller state vector. In this work, we implement and systematically compare two canonical families of methods for the spatial dimensionality reduction task: the classical, energy-optimal linear POD, and nonlinear deep CAEs following previous work such as Solera-Rico *et al.* (2024) and Fukagata & Fukami (2025). Within the nonlinear family, we additionally consider two regularised variational variants, a β-variational autoencoder (β-VAE) and a decomposed Kullback–Leibler (KL) variational autoencoder (DKL-VAE), to assess whether latent-space regularisation modifies the resulting latent dynamics. The following sections detail the implementation of each model.

To compare the performance of CAE and POD methods, we define the energy percentage E metric, following Eivazi *et al.* (2022) and Solera-Rico *et al.* (2024), a metric of the energy captured by the low-order reconstruction as

$$E = \left(1 - \left\langle \frac{\sum_{j=1}^{N_p} \sum_{i=1}^{N_c} \left(u_{i,j} - \tilde{u}_{i,j}\right)^2}{\sum_{j=1}^{N_p} \sum_{i=1}^{N_c} u_{i,j}^2} \right\rangle \right) \times 100\,\%, \tag{2.6}$$

where $\langle \cdot \rangle$ indicates ensemble averaging in time, N_p is the number of grid points, N_c the number of velocity components, $u_{i,j}$ denotes the ith component of the reference fluctuating velocity at grid point j and $\tilde{u}_{i,j}$ its low-order reconstruction. The same metric is evaluated at two stages of the framework: on the decoded snapshot to assess the compression quality of POD or CAE, and on the decoded prediction at a horizon τ (in convective-time t_c units) to assess the end-to-end prediction accuracy. When the time average $\langle \cdot \rangle$ is omitted, the same error ratio evaluated for a single snapshot is referred to as the instantaneous accuracy; the pooled distributions reported in § 3 are built from these instantaneous values.

2.2.1. *Proper orthogonal decomposition*

Proper orthogonal decomposition is employed as a linear baseline to compare against the nonlinear CAE. Proper orthogonal decomposition provides an optimal linear basis

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

to represent a dataset in terms of the L_2 norm, which can be interpreted in this case as turbulent kinetic energy. We first decompose the velocity field $\boldsymbol{U}(\boldsymbol{x}, t)$ (which contains both u and v components) as

$$\boldsymbol{U}(\boldsymbol{x}, t) = \bar{\boldsymbol{U}}(\boldsymbol{x}) + \boldsymbol{u}(\boldsymbol{x}, t), \tag{2.7}$$

where $\bar{\boldsymbol{U}}(\boldsymbol{x})$ is the velocity field averaged over time and $\boldsymbol{u}(\boldsymbol{x}, t)$ is the fluctuating component. The fluctuating velocity field can be approximated as a linear combination of orthonormal spatial basis functions $\boldsymbol{\phi}_i(\boldsymbol{x})$, the POD modes

$$\boldsymbol{u}(\boldsymbol{x}, t) \approx \sum_{i=1}^{d} a_i(t)\boldsymbol{\phi}_i(\boldsymbol{x}), \tag{2.8}$$

where $a_i(t)$ are the time-dependent modal coefficients and d is the number of modes retained for reconstruction.

To find the modes, the snapshot matrix $\boldsymbol{S}$ is assembled from the training data, where each row is a flattened velocity snapshot $\boldsymbol{u}(\boldsymbol{x}, t_j)$. Let N_t be the number of training snapshots and N_p be the number of grid points. The matrix $\boldsymbol{S}$ thus has dimensions $N_t \times (N_p N_c)$, with N_c being the number of velocity components. Following the method of snapshots (Sirovich 1987), the spatial modes $\boldsymbol{\phi}_i$ are found by solving the eigenvalue problem for the spatial covariance matrix $\boldsymbol{C}$

$$\boldsymbol{C} = \frac{1}{N_t - 1}\boldsymbol{S}^{\mathrm{T}}\boldsymbol{S}, \tag{2.9}$$

where $\boldsymbol{C}$ is an $N_p N_c \times N_p N_c$ matrix. The eigenvectors of $\boldsymbol{C}$, sorted by their corresponding eigenvalues λ_i, are the POD modes $\boldsymbol{\phi}_i$. The eigenvalues represent the kinetic energy content of each mode. Given the high dimensionality of $\boldsymbol{C}$, for computational efficiency we compute only the leading d eigenvectors using a randomised SVD algorithm.

Once the spatial modes $\boldsymbol{\Phi}$ (a matrix of size $N_p N_c \times d$) are known, the temporal modes $\boldsymbol{A}$ (a matrix of size $N_t \times d$) are obtained by projecting the snapshot matrix onto the modes

$$\boldsymbol{A} = \boldsymbol{S}\boldsymbol{\Phi}. \tag{2.10}$$

This matrix $\boldsymbol{A}$ of temporal coefficients forms the latent-space representation for POD-based models, analogous to the latent vector $\boldsymbol{z}$ of the CAE.

2.2.2. *Convolutional autoencoder*

An autoencoder (AE) is a neural network architecture designed for dimensionality reduction (Hinton & Salakhutdinov 2006), commonly used for feature learning and data compression. It consists of two main components: an encoder ($\mathcal{E}$) and a decoder ($\mathcal{D}$). The encoder maps the high-dimensional input $\boldsymbol{u}$ to a low-dimensional latent-space vector $\boldsymbol{z} = \mathcal{E}(\boldsymbol{u})$. The decoder then reconstructs the input from this latent vector, $\tilde{\boldsymbol{u}} = \mathcal{D}(\boldsymbol{z})$, aiming to obtain a reconstruction $\tilde{\boldsymbol{u}}$ as close to the original $\boldsymbol{u}$ as possible.

The encoder and decoder networks are trained simultaneously by gradient descent and backpropagation using the Adam algorithm (Kingma & Ba 2014), with the hyperparameters detailed in table 2. The training process minimises a loss function that measures the discrepancy between the input $\boldsymbol{u}$ and the reconstruction $\tilde{\boldsymbol{u}}$, defined as the squared L_2 norm of the reconstruction error: $\mathcal{L}_{rec} = \|\boldsymbol{u} - \tilde{\boldsymbol{u}}\|_2^2$. Once trained, the encoder is used to generate the temporal evolution of the latent space by applying it to each time step: $\boldsymbol{z}_t = \mathcal{E}(\boldsymbol{u}_t)$.

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

Hyperparameter	Simplified truck	Fluidic pinball
Latent dimension (d)	8	10
β (β-VAE)	0.0025	0.005
λ_{MI} (DKL-VAE)	1.6×10^{-4}	1×10^{-4}
λ_{TC} (DKL-VAE)	5×10^{-4}	5×10^{-3}
λ_{Dim} (DKL-VAE)	1.6×10^{-4}	1×10^{-4}
Batch size	1024	256
Learning rate	1×10^{-3}	1×10^{-3}
Epochs	200	300
Test snapshots	5000	20 000

Table 2. The CAE, β-VAE and DKL-VAE training hyperparameters for each case. Here, β applies to the β-VAE, and the loss weights λ_{MI}, λ_{TC}, λ_{Dim} to the DKL-VAE.

The spatial nature of flow field data guides the choice of a convolutional neural network to build the encoder and decoder networks, since spatial patterns are encoded more naturally by convolutional layers (Lecun *et al.* 1998). In the encoder, we use five or six convolutional layers (for the pinball and truck case, respectively) with a stride of two so that each layer halves the spatial dimension. This spatial reduction allows the subsequent layers to capture information at larger scales in the input flow data. As the spatial dimensions decrease, the count of filters in each layer increases to maintain information of the flow. After the last convolutional layer, the spatial information is flattened, and a fully connected layer is added to combine the information. Finally, a single linear layer with d units outputs the latent vector $\boldsymbol{z}$.

The decoder model is configured as a nearly symmetric network relative to the encoder. The latent vector $\boldsymbol{z}$ is input into a fully connected layer, with its output reshaped to match the last convolutional layer of the encoder. Subsequently, five to six transposed convolution layers are applied to incrementally expand the spatial dimension while reducing the number of filters. The last transposed convolution layer, using two filters, generates the output channels for the u and v velocity components. The activation function is the exponential linear unit (Clevert *et al.* 2015) for all layers except the last, where the activation is linear.

2.2.3. *The β-variational autoencoder*

In addition to the deterministic CAE, we also include a β-VAE (Solera-Rico *et al.* 2024) as a regularised nonlinear baseline. The β-VAE shares the same convolutional encoder-decoder backbone as the CAE described above, but uses a probabilistic latent representation (Higgins *et al.* 2017). The encoder outputs the parameters of a Gaussian distribution in latent space, $\boldsymbol{\mu}(\boldsymbol{u})$ and $\boldsymbol{\sigma}(\boldsymbol{u})$, from which the latent vector is sampled via the reparametrisation trick, $\boldsymbol{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}$, with $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. The decoder reconstructs $\tilde{\boldsymbol{u}}$ from $\boldsymbol{z}$. This probabilistic structure allows us to assess whether constraining the geometry of the nonlinear latent manifold can mitigate the loss of long-horizon predictability observed for the CAE.

The training loss combines the reconstruction error with the KL divergence D_{KL}, which measures the discrepancy between the encoded latent distribution and a unit Gaussian prior

$$\mathcal{L}_{\beta\text{-VAE}} = \|\boldsymbol{u} - \tilde{\boldsymbol{u}}\|_2^2 + \beta\, D_{KL}, \tag{2.11}$$

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

where β controls the strength of the regularisation. Larger values of β promote more disentangled and statistically independent latent coordinates at the cost of reconstruction fidelity (Solera-Rico *et al.* 2024). The parameter β is tuned to increase in statistical independence relative to the deterministic CAE without significantly increasing reconstruction error. The β-VAE is trained with the same latent dimension and remaining hyperparameters as the CAE, reported in table 2.

2.2.4. *DKL-VAE*

A limitation of the β-VAE is that the single coefficient β weights the whole KL term, which couples several effects: promoting independence between latent coordinates also forces each marginal towards the prior, and an excessive penalty can reduce the information capacity of the latent space. To assess whether a more selective regularisation modifies the latent dynamics, we consider a DKL-VAE, in which the divergence is split into three independently weighted contributions following the decomposition of Chen *et al.* (2018), recently applied to manifold learning of fluid flows by Wang *et al.* (2026)

$$\mathcal{L}_{DKL\text{-}VAE} = \|\boldsymbol{u} - \tilde{\boldsymbol{u}}\|_2^2 + \lambda_{MI}\,\mathcal{I} + \lambda_{TC}\,\mathcal{T} + \lambda_{Dim}\,\mathcal{K}, \tag{2.12}$$

where $\mathcal{I}$ is the index-code mutual information between the snapshot index and the latent code, $\mathcal{T}$ is the total correlation that measures the statistical dependence among latent coordinates and $\mathcal{K}$ is the dimension-wise divergence between each marginal latent distribution and the prior. These three terms sum to the standard KL divergence, so the β-VAE is recovered when $\lambda_{MI} = \lambda_{TC} = \lambda_{Dim} = \beta$. Increasing λ_{TC} alone encourages independent latent coordinates without tightening the matching of each coordinate to the prior, thereby decoupling disentanglement from prior matching. The aggregated posterior required by these terms is estimated through minibatch stratified sampling (Chen *et al.* 2018). We evaluate the DKL-VAE on the simplified truck and fluidic pinball cases as an alternative regularisation strategy, using $\lambda_{MI} = \lambda_{Dim} = 1.6 \times 10^{-4}$ and $\lambda_{TC} = 5 \times 10^{-4}$ for the truck, and $\lambda_{MI} = \lambda_{Dim} = 1 \times 10^{-4}$ and $\lambda_{TC} = 5 \times 10^{-3}$ for the pinball (table 2), keeping the same convolutional backbone, latent dimension and remaining hyperparameters as the CAE and β-VAE.

2.3. *Time-series prediction models*

Once high-dimensional spatial data $\boldsymbol{U}(\boldsymbol{x}, t)$ are compressed into a low-dimensional latent-space time series $\boldsymbol{z}(t)$, the next task is to build a predictive model that can predict the evolution of this latent state. The goal is to learn a function f that predicts a future sequence of H latent states (the horizon), given a sequence of L past latent states and their corresponding control inputs (the lookback). Critically, the control action $\boldsymbol{b}_t$ applied at time t influences the state at $t+1$. Therefore, a causal model must predict the discrete state sequence $[\hat{\boldsymbol{z}}_{t+1}, \ldots, \hat{\boldsymbol{z}}_{t+H}]$ using the control sequence $[\boldsymbol{b}_{t-L}, \ldots, \boldsymbol{b}_t, \ldots, \boldsymbol{b}_{t+H-1}]$

$$[\hat{\boldsymbol{z}}_{t+1}, \ldots, \hat{\boldsymbol{z}}_{t+H}] = f([\boldsymbol{z}_{t-L+1}, \ldots, \boldsymbol{z}_t], [\boldsymbol{b}_{t-L}, \ldots, \boldsymbol{b}_t, \ldots, \boldsymbol{b}_{t+H-1}]), \tag{2.13}$$

where $\hat{\boldsymbol{z}}$ denotes the predicted state.

From a closure-modelling standpoint, projection-based ROMs are not in general closed (Noack *et al.* 2003). An in-principle infinite Mori–Zwanzig memory kernel encodes the action of unresolved scales on the resolved coordinates (Menier *et al.* 2023; Gupta *et al.* 2025). We adopt as a working hypothesis a finite-memory approximation in which this kernel is truncated to a window of length L and represented by the learned predictor, which absorbs both the resolved-mode contribution and a deterministic, data-driven surrogate of

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

the orthogonal-dynamics term. Under this hypothesis, latent representations that better separate resolved and unresolved scales are expected to render the truncated closure a closer approximation; the empirical consequences are revisited in § 3.3.

We restrict the temporal modelling stage to LSTM-based predictors (Hochreiter & Schmidhuber 1997) for two reasons. First, LSTMs are a well-established baseline for nonlinear time-series modelling in fluids and ROM settings (Solera-Rico *et al.* 2026), and they provide sufficient expressive power to test our main question, namely how the choice of spatial encoder affects predictability in the latent space. Second, LSTMs offer a favourable accuracy–complexity balance: they are comparatively lightweight, stable to train on moderate datasets, and amenable to real-time deployment on embedded hardware. By keeping the predictor family fixed, we ensure a controlled comparison in which differences in forecasting performance can be attributed primarily to the latent representations produced by POD versus CAEs.

The input to the LSTM at each step in the lookback sequence is the concatenation of the latent state and the control signal at that time. In this study, we implement and compare three distinct predictive LSTM architectures, illustrated in figure 5.

(i) Single step: this is a standard autoregressive model, corresponding to figure 5(*a*). The LSTM takes the lookback sequence (up to $\boldsymbol{z}_t$ and $\boldsymbol{b}_t$) and predicts only the single next latent state, $\hat{\boldsymbol{z}}_{t+1}$. To generate long-term predictions, this output is combined with the next control input $\boldsymbol{b}_{t+1}$ and fed back into the model to predict $\hat{\boldsymbol{z}}_{t+2}$. This process is repeated recursively, thus potentially suffering from error propagation.

(ii) Direct sequence: this model, shown in figure 5(*b*), is a sequence-to-sequence architecture. It takes the lookback sequence as input to the LSTM core. The final hidden state of the LSTM (which encodes information up to time t) is then concatenated with the future control sequence that will modify the future states, $[\boldsymbol{b}_t, \ldots, \boldsymbol{b}_{t+H-1}]$, and passes through a MLP. This network directly predicts the entire future sequence of latent states $[\hat{\boldsymbol{z}}_{t+1}, \ldots, \hat{\boldsymbol{z}}_{t+H}]$ in a single forward pass, avoiding the recursive error accumulation of the single-step model. While this requirement for the complete future control sequence makes it unsuitable for standard reactive controllers, it is the ideal structure for MPC, where this exact future control sequence is the variable being optimised.

(iii) Derivative based: this hybrid architecture, shown in figure 5(*c*), embeds a numerical time integrator within the prediction loop as described in Yang *et al.* (2025). Instead of directly predicting the future state $\hat{\boldsymbol{z}}_{t+1}$, the LSTM is trained to predict the time derivative of the latent state $\dot{\boldsymbol{z}}$, based on the inputs at time t, $\dot{\boldsymbol{z}}_t = g(\boldsymbol{z}_t, \boldsymbol{b}_t)$. The future state is then obtained using an explicit time-stepping scheme. The model uses the second-order Adams–Bashforth method

$$\hat{\boldsymbol{z}}_{t+1} = \hat{\boldsymbol{z}}_t + \Delta t \left(\frac{3}{2}\dot{\boldsymbol{z}}_t - \frac{1}{2}\dot{\boldsymbol{z}}_{t-1} \right), \tag{2.14}$$

where the LSTM provides the derivative $\dot{\boldsymbol{z}}_t$ at each step. $\dot{\boldsymbol{z}}_{t-1}$ is initialised by estimating the initial derivative using a backward finite difference of the last two latent states in the lookback window. This imposes a physical structure on the model, which can improve stability for long-horizon predictions.

All three architectures are trained by minimising a loss function, $\mathcal{L}_{pred}$, based on the L_2 norm between the predicted latent-state sequence and the ground truth sequence over the

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

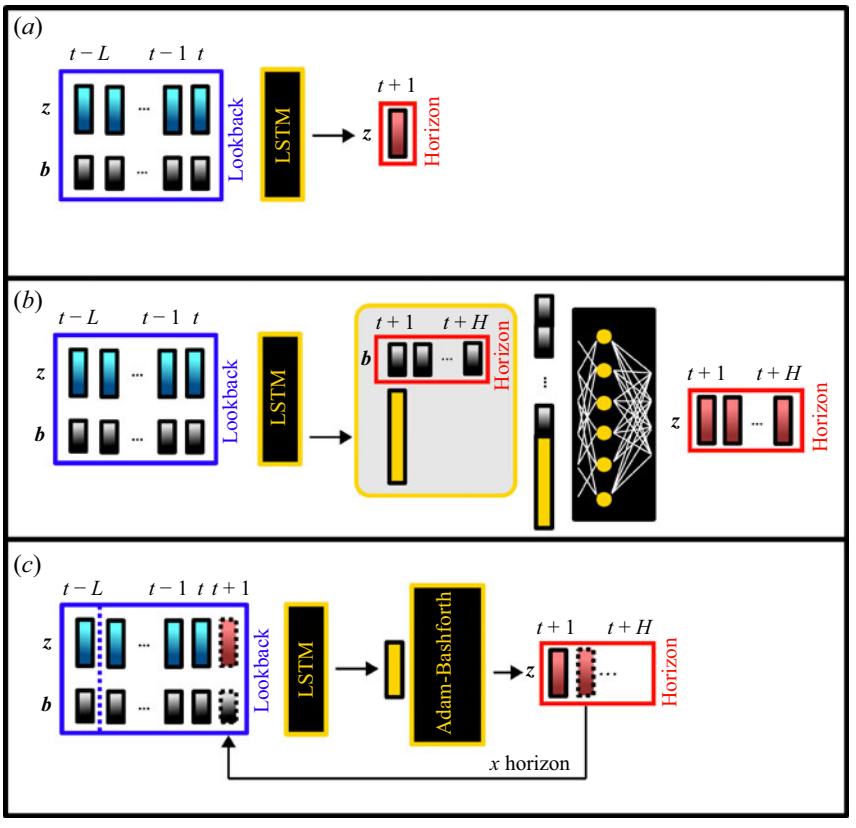


Figure 5. Schematic of the three LSTM-based prediction architectures tested. For all architectures, the latent space (in blue) and control space (in grey) are fed as input throughout the entire lookback period up to the current time step t. (*a*) Single-step model, consisting of an LSTM layer that predicts the immediate next time step of the latent space z_{t+1} (in red). (*b*) Direct sequence model. The latter predicts the entire horizon sequence of the latent space at once by combining a latent context vector (yellow bar) as the output of the LSTM layer with the horizon sequence of the control space through a multi-layer perceptron (MLP) layer. (*c*) Derivative-based model. The LSTM predicts the time derivative of the latent state $\dot{z}_t$. An Adams–Bashforth method is used to propagate the solution to z_{t+1} with derivative information. The process is repeated up to time instant $t + H$ by feeding the predictions into the advanced lookback sequence.

prediction horizon H

$$\mathcal{L}_{pred} = \frac{1}{H}\sum_{i=1}^{H} \left\| \hat{z}_{t+i} - z_{t+i} \right\|_2^2 . \tag{2.15}$$

While `Single-step` and `Direct sequence` models are trained using this standard L_2 loss, the `Derivative-based` architecture employs a specific loss, also introduced in Yang *et al.* (2025), which weights only the L_2 norm of the first and last steps in the horizon. This was found to improve the long-term stability of the derivative-based integration. A summary of the training parameters is shown in table 3. We note that the hyperparameters have been tuned via a manual hyperparameter search on the test set metrics, specifically for each predictor architecture, in order to compare each architecture in its optimal configuration.

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

Model	d	L	H	LSTM lay.	LSTM dim.	MLP dim.	Dropout
				Simplified 2-D truck case			
				POD-based models			
`Single-step`	8	16	1	2	32	32	0.0
`Direct sequence`	8	16	8	2	64	64	0.0
`Derivative-based`	8	4	8	2	64	64	0.3
				AE-based models			
`Single-step`	8	8	1	2	128	128	0.0
`Direct sequence`	8	8	2	2	128	128	0.0
`Derivative-based`	8	8	2	2	128	128	0.0
				Fluidic pinball case			
				POD-based models			
`Single-step`	22	64	1	1	16	16	0.3
`Direct sequence`	22	32	16	2	128	256	0.0
`Derivative-based`	22	8	16	2	64	64	0.3
				AE-based models			
`Single-step`	10	64	1	1	16	16	0.3
`Direct sequence`	10	32	16	2	128	256	0.0
`Derivative-based`	10	8	16	2	64	64	0.3

Table 3. The LSTM training hyperparameters.

3. Results

This section presents a systematic comparison of the performance of the eight main model combinations: POD and CAE encoding, each paired with the three LSTM predictors, plus the β-VAE and DKL-VAE encodings, each paired with the best-performing predictor. We first evaluate the baseline compression efficiency of encoders, followed by an analysis of the long-term predictive accuracy, and conclude with an investigation into the latent-space dynamics to explain the observed performance.

3.1. *Compression methods: POD, CAE, β-VAE and DKL-VAE*

We first establish the reconstruction accuracy of the four dimensionality reduction methods, independent of the temporal prediction. Figure 6 provides a qualitative comparison of the reconstructed velocity fields from POD, CAE, β-VAE and DKL-VAE models against the reference (top row of figure 6). The 'Decoded' columns show the instantaneous reconstruction of a test snapshot from its latent representation.

In all cases, the latent-dimension selection criterion is such that the minimum number of coordinates leads to at least 95 % of the original variance of the data for the truck and 90 % for the fluidic pinball configuration. These thresholds refer to the mean reconstruction accuracy E averaged over the entire test set (or the cumulative eigenvalue ratio for POD). For the simplified truck case (figure 6*b*), both models use a latent dimension of $d = 8$. A more significant difference is seen in the fluidic pinball case (figure 6*a*). Here, the CAE, β-VAE and DKL-VAE are able to encode, across the entire test set, 90 % of the variance with a latent space of dimension 10, compared with POD which needs 22.

The nonlinear AEs are more efficient and powerful compressors, capable of capturing the features of the flow under control actions in a more compact latent representation than the linear POD, as observed in previous works (Murata *et al.* 2020; Fukami *et al.* 2020; Eivazi *et al.* 2022; Zhu *et al.* 2024). Notably, the AE reconstructions in figure 6 retain less diffuse and higher-wavenumber features than POD, while β-VAE prediction exhibits a

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

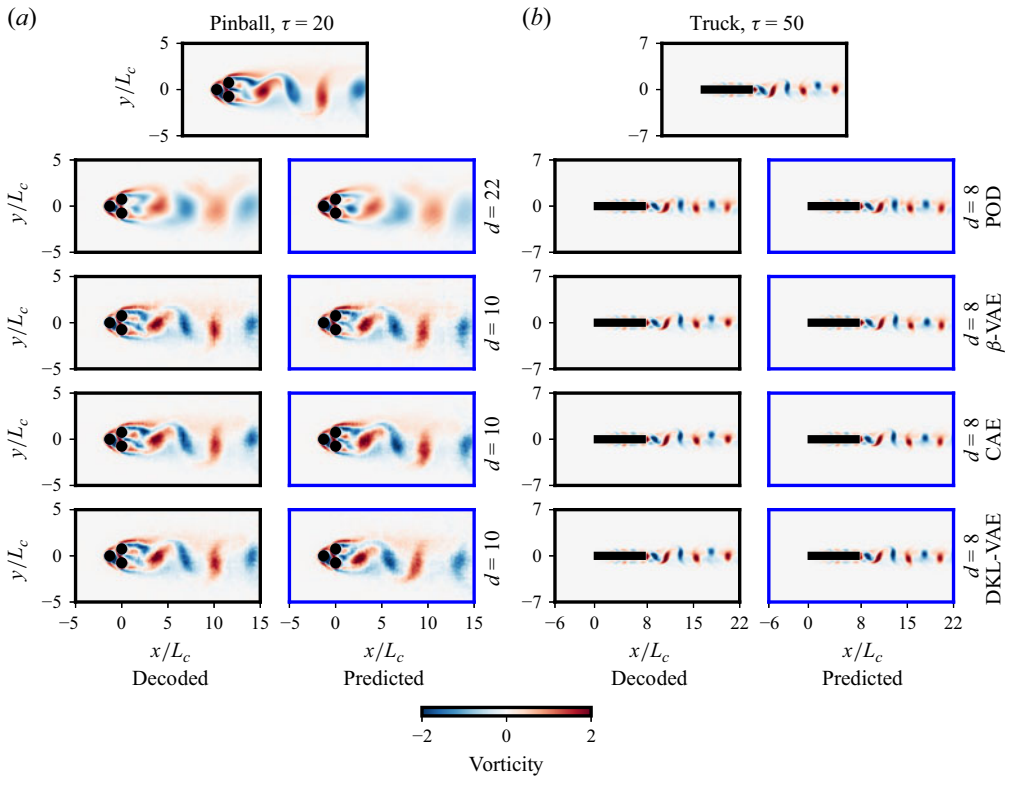


Figure 6. Qualitative comparison of flow field reconstruction and long-term prediction for the fluidic pinball ((*a*), at $\tau = 20$) and the simplified truck ((*b*), at $\tau = 50$). The top row shows the reference non-dimensional vorticity; all rows for a given case are extracted from the same test snapshot. The 'Decoded' column shows the instantaneous reconstruction from the latent space for the POD, β-VAE, CAE and DKL-VAE models, without prediction. The 'Predicted' column shows the final predicted field decoded from the best LSTM models' latent-space predictions after the specified time horizon.

visible phase shift in the wake, discussed in § 3.3. As shown next, preserving these details can increase the complexity of the latent dynamics and reduce long-horizon predictability.

3.2. *Prediction accuracy*

Having established the compression baseline, we now assess the core task: the long-term prediction of the latent-space dynamics. Figures 7(*a*) (fluidic pinball) and 8(*a*) (simplified truck) show the accuracy of the prediction, E, as a function of the prediction time horizon, τ. This accuracy metric E is computed on the reconstructed flow fields, thus measuring the performance of the whole framework from end to end. To quantify the sensitivity of the comparison to training stochasticity, the complete framework is retrained for an ensemble of 20 independent seeds per case: for each seed, the CAE, β-VAE and DKL-VAE encoders are retrained from scratch and the LSTM predictors are trained on the latent space of the same-seed encoder, while the POD basis is deterministic and only its predictors are re-seeded. The curves report, at each horizon, the median of E over all prediction windows in the test set and all seeds, and the violin plots in figures 7(*b*) and 8(*b*) pool the corresponding instantaneous values.

A consistent trend emerges across both flow cases. The `Single-step` models (in red) exhibit a rapid decay in accuracy, as the recursive error accumulation quickly destabilises

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

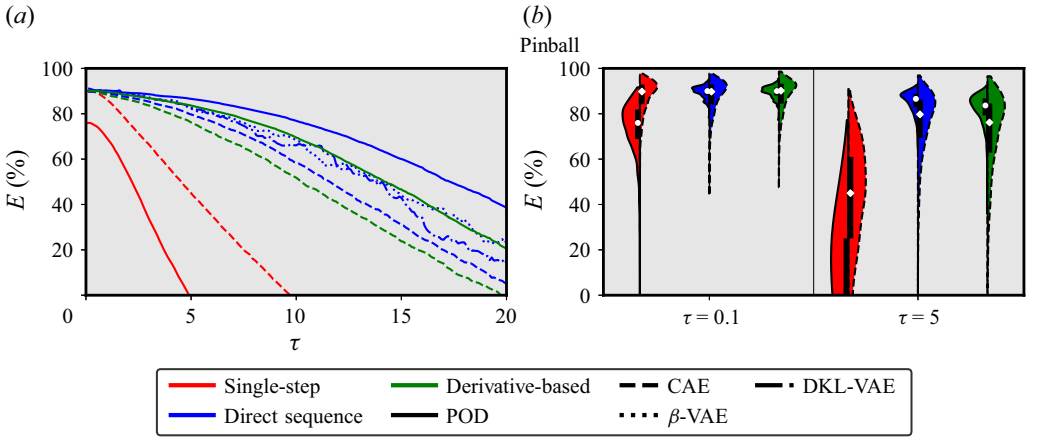


Figure 7. Prediction accuracy E for the fluidic pinball over an ensemble of 20 training seeds. (a) median of E (2.6) as a function of the prediction horizon τ (convective-time t_c units), taken over all prediction windows in the test set and all seeds. Colours denote the LSTM architectures (`Single-step`, `Direct sequence`, `Derivative-based`); line styles denote the encoder: solid POD, dashed CAE, dotted β-VAE, dash-dotted DKL-VAE. (b) distributions of the instantaneous E (evaluated snapshot by snapshot) for the POD- and CAE-based models, pooling all prediction windows and seeds, at $\tau = 0.1$ and $\tau = 5$. White markers indicate the median ($\circ$ for POD, $\diamondsuit$ for CAE); black boxes span the interquartile range (25th–75th percentiles). The first reported horizon is $\tau = 0.1$.

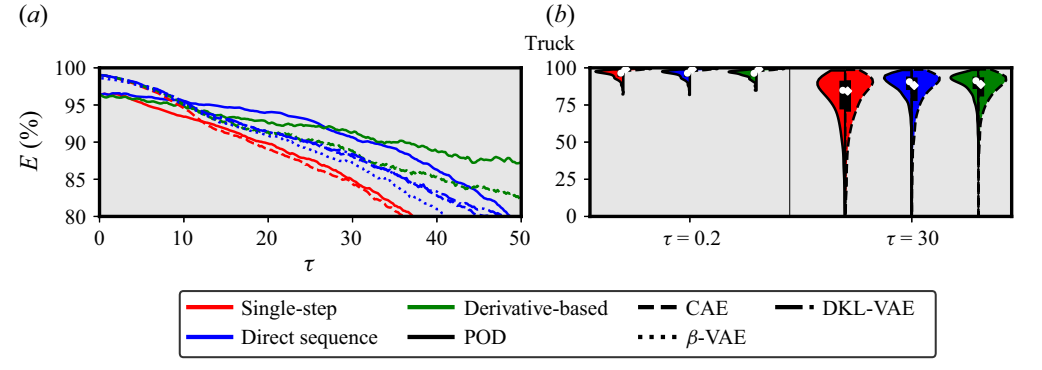


Figure 8. Prediction accuracy E for the simplified truck over an ensemble of 20 training seeds, with the same protocol, statistics and line conventions as figure 7. (a) median of E over all prediction windows and seeds as a function of τ, reported up to $\tau = 50$ (see text). (b) distributions of the instantaneous E for the POD- and CAE-based models at $\tau = 0.2$ and $\tau = 30$. The first reported horizon is $\tau = 0.2$.

the prediction. The `Direct sequence` (blue) and `Derivative-based` (green) models are more stable, generally retaining high accuracy for longer prediction horizons.

However, the most critical finding is the comparison between the compression methods. For the simplified truck case (figure 8), the AE-based models (dashed lines) initially show a slight advantage due to the higher AEs compression power, but for horizons longer than $\tau \approx 10$, the POD-based models (solid lines) are more accurate and stable. The separation between encoders nevertheless remains within a few percentage points of E over the reported truck horizons, indicating that this case lacks the dynamical complexity to discriminate strongly between compression methods; the fluidic pinball provides the more demanding comparison. This trend is even more pronounced in the pinball case (figure 7). Despite the superior reconstruction performance of the AEs, the POD-based models (solid

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

blue and green lines) consistently outperform their AE-based counterparts (dashed blue and green lines) for nearly the entire prediction horizon. This identifies a practical cross-over: for short horizons, the AEs can be competitive due to their compactness, whereas for horizons beyond roughly $\tau \approx 10$ (truck) and essentially across all tested horizons (pinball), POD yields more reliable forecasts.

The DKL-VAE, in dash-dotted lines in figures 7 and 8, follows the same pattern as the other nonlinear encoders. For the truck ($d = 8$) and the `Direct sequence` model, its accuracy at the first reported horizon, a median $E \approx 99\,\%$, is indistinguishable from the CAE; at intermediate horizons it remains at CAE level, with a median $E = 91.3\,\%$ at $\tau = 20$ against 91.4 % for the CAE, 90.9 % for the β-VAE and 94.0 % for POD (`Direct sequence` predictors), and at $\tau = 50$ it stays below the best POD-based model (78.9 % against 87.4 %). For the pinball ($d = 10$), the DKL-VAE accuracy at the first reported horizon is again in line with the other nonlinear encoders (median $E \approx 91.1\,\%$, between the CAE and β-VAE values); its forecast accuracy is comparable to the CAE and β-VAE at both the short and long ($\tau = 5$, median $E \approx 82.6\,\%$) horizons used elsewhere in the manuscript, while POD remains more accurate throughout. In both cases the more selective regularisation does not recover the long-horizon prediction accuracy of POD. The trade-off between compactness and predictability therefore persists when the latent regularisation is decomposed into independently weighted information-theoretic terms.

For the truck, the horizon reported in figure 8 is limited to $\tau = 50$. Beyond this horizon the run-to-run variability across training seeds grows rapidly: at $\tau = 100$ a substantial fraction of the 20 seeds produces diverged predictions (median $E < 0$ over the prediction windows) for every encoder–predictor combination except the POD-based `Derivative-based` model, so conclusions drawn at longer horizons would reflect training stochasticity rather than the choice of encoder. The seed analysis also identifies the `Derivative-based` predictor as the most robust option at long horizons in this case: combined with POD, it shows no diverged seeds at $\tau = 50$ and retains a median $E = 80.5\,\%$ at $\tau = 100$, with a single mild failure across the ensemble.

This result is further clarified by the violin plots in figures 7(*b*) and 8(*b*), which show the statistical distribution of the prediction accuracy E of an ensemble of test trajectories and training seeds in both a short- and long-term prediction horizon. At the single-step, short prediction horizon ($\tau = 0.1$ for pinball and $\tau = 0.2$ for truck), all models perform well, with high median accuracies (white dots) and compact distributions. However, at the long horizon (e.g. $\tau = 5$ or $\tau = 30$), the distributions diverge significantly. The POD-based models (solid-line violins) maintain a higher median accuracy and generally a narrower distribution, indicating a more reliable performance. In contrast, CAE-based models (dashed-line violins) exhibit a lower median accuracy and a very long tail towards $E = 0$. This long tail reveals that a large fraction of the CAE-based predictions fail significantly, diverging from the true solution. This implies greater sensitivity of the nonlinear latent dynamics to minor prediction errors, so some trajectories drift into qualitatively different regimes and abruptly fail. From a MPC standpoint, this means that a slightly less compact but more predictable latent model can enlarge the feasible prediction horizon and, in principle, improve closed-loop robustness. Pooling the training seeds does not alter this picture: the long tail of the CAE-based distributions is a property of the models rather than of a particular training realisation.

3.3. *Analysis of latent-space dynamics and flow prediction*

The contrast between encoders observed in the previous section can be anticipated from the framing of § 1 and § 2.3. The POD basis inherits a structure for shedding-dominated

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

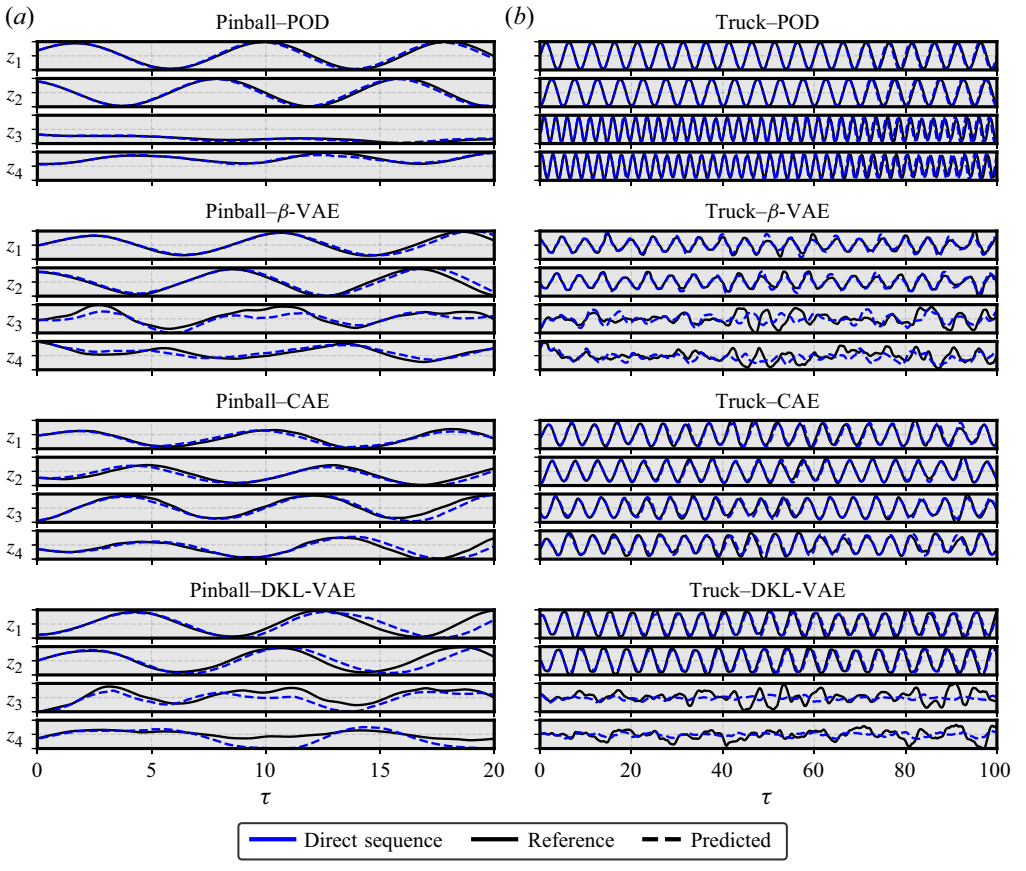


Figure 9. Temporal evolution of the first four leading latent coefficients (energy ordered for POD; ordered by peak power spectral density (PSD) of the encoded reference for CAE, β-VAE and DKL-VAE), comparing the ground truth (solid black line) with the prediction of the best-performing `Direct sequence` LSTM model (blue dashed line). (*a*) Fluidic pinball; (*b*) simplified truck. Rows show POD (top), β-VAE (second), CAE (third) and DKL-VAE (bottom) latent spaces.

flows analogous to the dynamics observed with a Koopman embedding, whereas the AEs optimise pointwise reconstruction. It is nonetheless important to distinguish between a linear decomposition and a linear dynamical representation. The POD provides the former, but not the latter. It is indeed well known that, after projection of the Navier–Stokes equations onto a POD basis, the reduced coefficients generally evolve according to nonlinear equations containing modal interaction terms (Noack *et al.* 2005).

The discrepancy in predictive performance, despite the superior encoding capabilities of the AEs, can be explained by the structure of the latent spaces. Figure 9 plots the temporal evolution of the first four latent coefficients, comparing the ground truth 'Reference' (solid black line) with the prediction from the best-performing `Direct sequence` LSTM model (blue dashed line).

The upper row shows the latent space of the POD. The dynamics is simple, smooth and quasi-periodic, dominated by low-frequency oscillations. As a result, the LSTM model can easily learn and accurately forecast this simpler dynamics, with the predicted signal tracking the reference for long durations.

The remaining rows show the β-VAE, CAE and DKL-VAE latent spaces. In contrast with the POD modes, the dynamics is more complex and non-periodic, showing

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

higher-frequency components. The AEs encode the flow features into a more compact representation whose dynamics is more irregular and broadband. The LSTM has difficulty modelling this more complex dynamics, and the predicted signal diverges from the reference at shorter prediction horizons.

This analysis supports the hypothesis that the linear basis of POD creates a latent space with a simple, low-frequency dynamics that is easier to learn and predict. Conversely, the nonlinear AEs, in their less-restricted search for optimal compression, create a more complex and chaotic latent space that is more difficult to forecast.

This conclusion is visualised and further explained by the 'Predicted' columns of figure 6. These fields show the final prediction of the flow field at a long-time horizon. They correspond to a single representative trajectory; the quantitative accuracy is reported as the median curves of figures 7(*a*) and 8(*a*), where the median of E is taken over $\sim$100 test initial conditions and 20 training seeds. For the pinball case ($\tau = 20$), the POD-based prediction retains the large-scale vortex shedding structure, while showing clear signs of smoothing of the flow fields. The AE-based predictions are sharper, retaining smaller details in the flow field but achieving lower accuracy for this snapshot; the DKL-VAE prediction shows the same sharper, lower-accuracy character as the other nonlinear encoders. For the truck case in $\tau = 50$, all models retain the wake structure, with the CAE and β-VAE predictions slightly below the reconstruction of the POD model; the DKL-VAE prediction is consistent with the other nonlinear encoders, following the statistical trend observed in figures 7(*b*) and 8(*b*). It can be noted that the higher spatial frequency content of the AE decoded fields, compared with POD reconstructed fields, produces better reconstructed energy in the short term predictions, but tends to produce faster error growth as the errors accumulate in the latent-space prediction. This faster error growth can be attributed to the accumulation of phase-shift errors in the predictions, translating into a higher error when the predicted shifted and reference wake flow fields are compared and evaluated (see figure 13 below). This behaviour can be identified in figure 10, where the averaged L_2 error of the velocity predictions is shown. The plot highlights areas where the predicted fields produce larger errors, coincident with those areas including higher spatial frequency fluctuations. This behaviour is consistent with the idea that nonlinear latent coordinates, while efficient for reconstruction, can exhibit higher sensitivity to initial-condition and phase errors. Hence, small LSTM prediction mismatches grow more quickly in the AEs latent system, effectively reducing its long-horizon predictability. This observation is consistent with the closure perspective of § 2.3: the broadband AE latent dynamics places a heavier burden on the truncated memory term that the predictor must absorb, whereas the narrow-band POD dynamics renders the finite-memory approximation closer to sufficient.

In what follows, latent-space stability refers to the predictive accuracy and reliability of the latent dynamics over the admissible control range considered in the dataset. A stable latent representation is one whose multi-time-delay predictor remains bounded, accumulates error slowly over the prediction horizon and preserves the dominant phase, amplitude and geometric structure of the encoded trajectory. This is an operational, finite-horizon definition tailored to predictive ROMs for control, not a statement of Lyapunov stability of the full controlled flow. The qualitative differences between the encoder latent spaces can be made quantitative through phase-space diagnostics applied to the best-performing `Direct sequence` predictor. Figures 11 and 12 show the phase portraits on the dominant latent pair and a Poincaré section in the next two coordinates, comparing the encoded reference with the predicted trajectory. For the truck, the POD latent trajectory exhibits a ring-like phase portrait and a compact Poincaré section, both well reproduced by the LSTM. The CAE trajectory shows a broader, irregular geometry, and the predicted

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

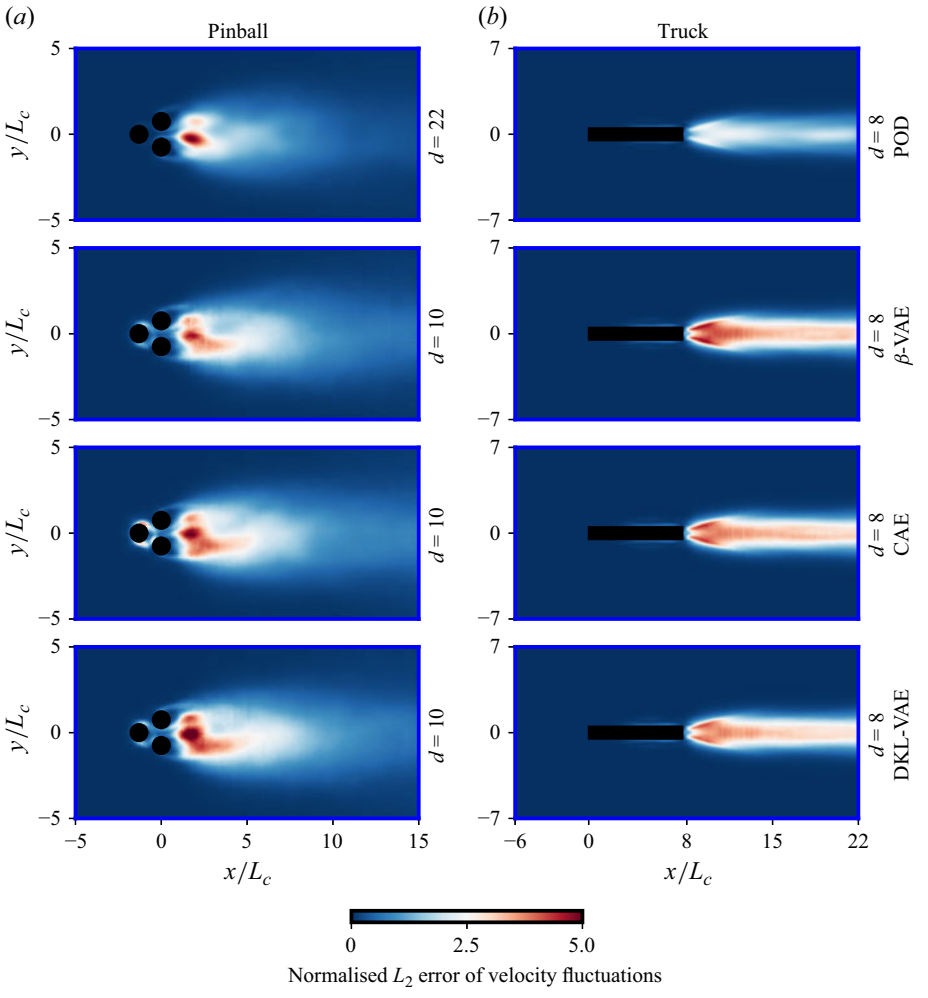


Figure 10. Pointwise normalised L_2 error magnitude of the predicted velocity fluctuations, combining the streamwise and cross-wise components as $\sqrt{e_u^2 + e_v^2}$ at each grid point, where each component $e_i = \langle(\hat{u}_i - u_i)^2\rangle/\sigma_i^2$ is the mean squared error normalised by the variance σ_i^2 of the corresponding reference velocity component. Each per-component field is averaged over the full prediction horizon, $\tau \in [0, 20]$ for the fluidic pinball (*a*) and $\tau \in [0, 100]$ for the simplified truck (*b*), and further averaged over all prediction windows in the test set. Predictions correspond to the best-performing `Direct sequence` LSTM model. Top row: POD; second row: β-VAE; third row: CAE; bottom row: DKL-VAE.

Poincaré section is markedly more concentrated than the reference, indicating that the CAE predictor collapses onto a narrower effective attractor rather than only accumulating phase errors. The β-VAE sits between the two, with a more regular phase portrait due to the latent-space KL regularisation, but a still-broad Poincaré section. The DKL-VAE shows a phase portrait and Poincaré section comparably structured to the β-VAE, consistent with the comparable degree of latent regularisation applied by both objectives. The fluidic pinball case shows analogous behaviour with the wider attractor expected from its weakly chaotic dynamics.

To separate the contribution of phase from amplitude in the prediction error, the dominant standardised latent pair is mapped to its polar form, $A(t) = \sqrt{z_1^2 + z_2^2}$ and $\phi(t) =$

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

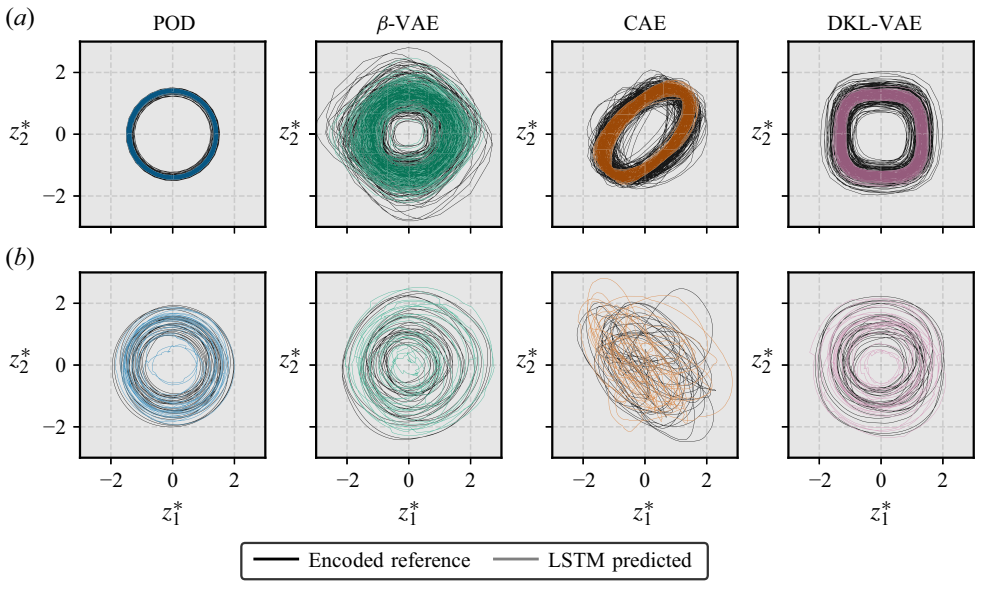


Figure 11. Phase portraits of the dominant standardised latent pair (z_1^*, z_2^*) for the encoded reference (black) and the `Direct sequence` LSTM prediction (coloured). (*a*) Truck; (*b*) fluidic pinball. Columns: POD, β-VAE, CAE and DKL-VAE. Standardisation uses the variance of the encoded reference. The POD portrait is the most regular for both flows; the CAE shows a broader, irregular geometry that the predictor fails to reproduce; the DKL-VAE portrait is comparably structured to the β-VAE.

$\arg(z_1 + iz_2)$ on the continuous branch. For POD the coordinates are energy ordered; for β-VAE, CAE and DKL-VAE the coordinates are reordered by decreasing peak intensity of the PSD of the encoded reference, so that the dominant oscillatory pair is identified consistently across encoders. The phase error is wrapped to $[-\pi, \pi]$, so $|\Delta\phi|$ is bounded in $[0, \pi]$ and $|\Delta\phi| = \pi$ indicates complete loss of phase coherence between the predicted and the encoded dominant pair. Figure 13 reports the absolute phase error $|\Delta\phi|$ and amplitude error $|\Delta A|$ for both flows. In the truck case, all three baseline encoders retain $|\Delta\phi| < \pi/2$ up to $\tau \approx 400$; β-VAE then saturates close to π around $\tau \approx 550$ and CAE around $\tau \approx 900$, while POD remains below $\pi/2$ over the entire test trajectory ($\tau \approx 1000$). The DKL-VAE loses phase coherence earlier, with $|\Delta\phi|$ exceeding $\pi/2$ from $\tau \approx 200$ onward. For the fluidic pinball $|\Delta\phi|$ grows faster and the four encoders are closer to each other, with CAE reaching $|\Delta\phi| \approx \pi$ around $\tau \approx 80$ and POD and β-VAE oscillating near π from $\tau \approx 150$ onward. The amplitude error stays in a comparable range across encoders, with the only clear separation appearing in the truck POD case where it is essentially negligible. The decomposition shows that long-horizon degradation in the latent space is driven by phase drift rather than amplitude distortion, and that for the truck CAE and β-VAE lose phase coherence earlier than POD. For the more chaotic pinball, the separation between encoders is less clear, as expected.

Beyond the phase and amplitude decomposition, we further characterise the latent dynamics through its short-horizon sensitivity to perturbations. We estimate the effective divergence rate from the latent trajectories (Rosenstein *et al.* 1993). For each reference point $\boldsymbol{z}_i$ in the standardised d-dimensional latent state, the nearest neighbour $\boldsymbol{z}_j$ is identified subject to a temporal exclusion window of at least one reference shedding period (estimated from the dominant Welch frequency of the encoded z_1), which prevents false neighbours arising from temporal correlations. The control sequence is included in

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

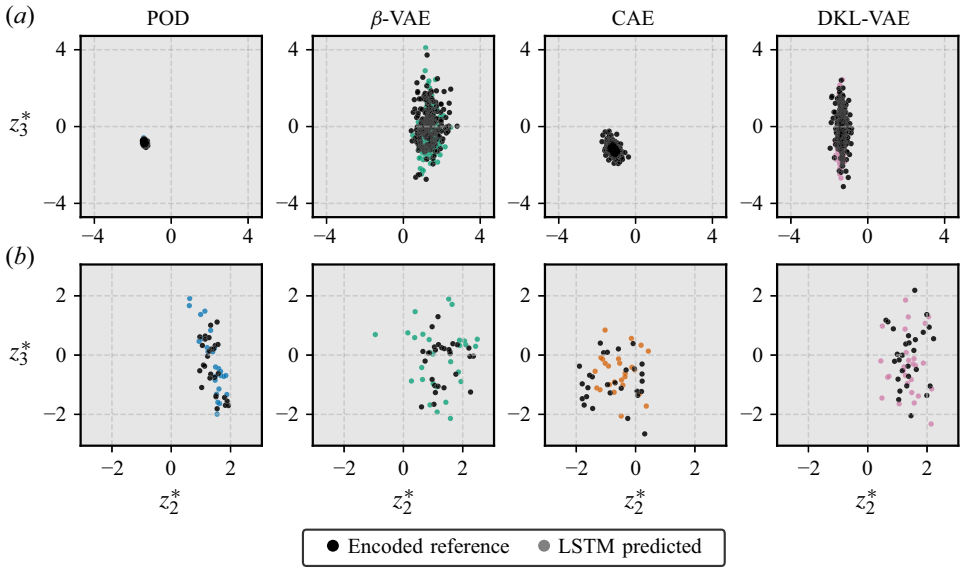


Figure 12. Poincaré sections in (z_2^*, z_3^*) taken at positive crossings of $z_1^* = 0$. Encoded reference (black) versus LSTM-predicted (coloured), with the same row/column layout as figure 11. Compact structures indicate quasi-periodic or weakly chaotic behaviour; wider clouds indicate broader spectral content.

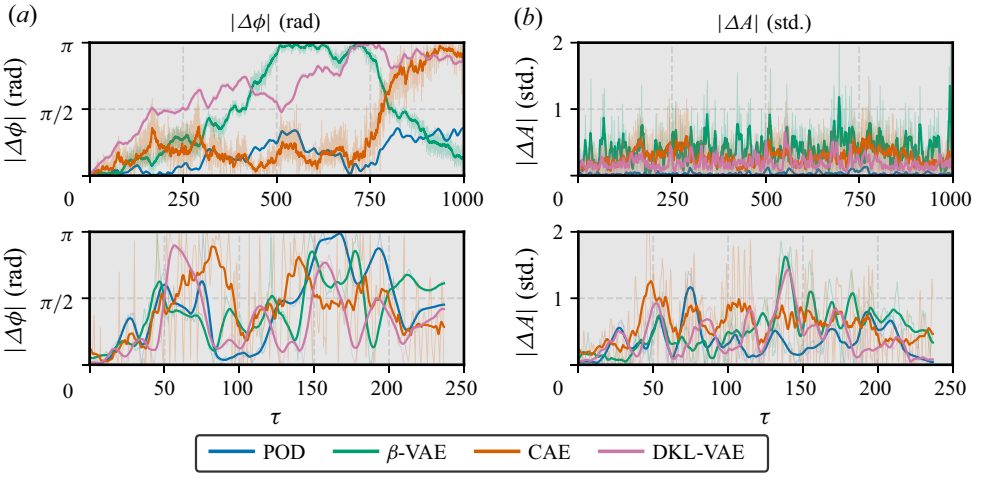


Figure 13. Absolute phase error $|\Delta\phi|$ (*a*) and absolute amplitude error $|\Delta A|$ (*b*) of the dominant latent pair, for the truck (upper row) and fluidic pinball (lower row). Lines: POD (blue), β-VAE (green), CAE (orange) and DKL-VAE (pink). Thick lines are a centred moving average over one reference shedding period; thin lines are the raw signals. The errors are similar across encoders except for the truck POD case, where the phase error grows more slowly and the amplitude error remains close to zero, indicating that it is dominated by phase error.

the neighbour-search space so that neighbours share a similar actuation history, while future control inputs are not explicitly considered and its effect is therefore included in the divergence rates. The mean logarithmic separation $\langle \ln d(k\Delta t)\rangle$ is then fitted linearly on a fixed convective-time window, with $[0.2, 10]\, t_c$ for the pinball and $[3, 10]\, t_c$ for the truck. The truck window starts at $3\, t_c$ to skip the initial transient observed in the curves of figure 14.

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

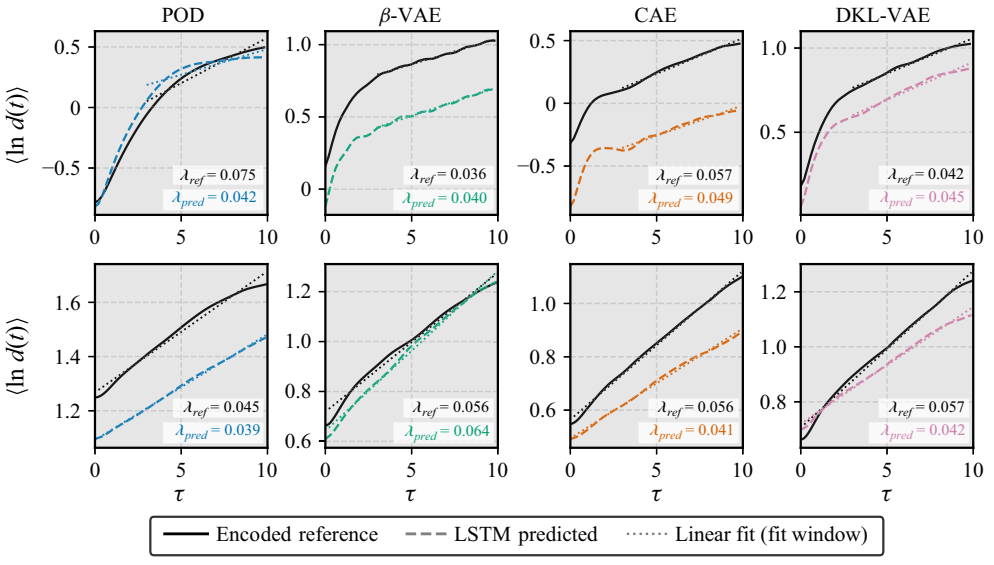


Figure 14. Rosenstein-style logarithmic separation of initially close latent-state pairs, $\langle \ln d(t) \rangle$, for the encoded reference (solid) and the LSTM-predicted trajectory (dashed). Rows: truck and fluidic pinball cases. Columns: POD, β-VAE, CAE and DKL-VAE. The dotted black line in each panel is the linear fit used to extract the effective short-horizon divergence rate, drawn only within its fit window ($[3, 10]\, t_c$ for the truck, $[0.2, 10]\, t_c$ for the pinball). Fitted slopes are annotated in the bottom-right corner of each panel as λ_{ref} and λ_{pred}. Neither case exhibits a clean asymptotic log-linear regime, reflecting the quasi-periodic or weakly chaotic nature of the actuated wakes rather than a defect of the estimator; λ_{eff} is therefore reported as an integrated short-horizon divergence rate over the fit window, not as a Lyapunov exponent.

We report the resulting slope as an effective short-horizon divergence rate λ_{eff} rather than a Lyapunov exponent. The two flows studied here are not classically chaotic in the Rosenstein sense: the truck wake at $Re = 500$ is quasi-periodic, and the fluidic pinball at $Re = 150$ is only weakly chaotic, just past the transition to chaotic dynamics (Deng *et al.* 2020). As a consequence, an asymptotic log-linear divergence regime is not observed over the test-trajectory horizon; the mean separation saturates relatively quickly as it approaches a fraction of the attractor size. This is a physical feature of the actuated wakes, not a defect of the estimator. We therefore interpret λ_{eff} as an integrated divergence rate over the fit window rather than as an asymptotic dynamical invariant, and we rely on the relative ordering of the values across methods and the comparison between encoded-reference and predicted trajectories, rather than on the absolute magnitude. Figure 14 makes this regime visible: the linear fits are drawn only within the fit window, so one can judge the local quality of the linear approximation for each curve. Under control the latent trajectory is non-autonomous, so the separation of initially close states reflects both the intrinsic sensitivity of the flow and differences in the forcing histories, a contribution that classical Lyapunov exponents do not isolate and that is instead the object of conditional Lyapunov exponents in driven systems (Pecora & Carroll 1990); the inclusion of the control sequence in the nearest-neighbour search described above ensures that compared trajectories share similar actuation histories. Indicators of this kind have recently been shown to be recoverable directly in a latent space, although in the autonomous setting (Özalp & Magri 2025).

The fluidic pinball results show encoded-reference divergence rates of comparable magnitude across all four encoders ($\lambda_{ref} \approx 0.045$–$0.057$ per convective time), consistent with the separation rate of the chaotic wake being largely independent of the choice of encoder. The LSTM predictors reproduce the same order of magnitude; the POD

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

predictor slightly underestimates the reference, the β-VAE predictor slightly overestimates it and the CAE and DKL-VAE predictors moderately underestimate it. For the truck, the encoded POD does not show a proper log-linear region, consistent with the quasi-periodic trajectory in the POD latent space. The value of λ_{ref} annotated for this case should therefore not be compared quantitatively with the other encoders, as it mainly reflects the curvature of the separation curve within the fit window. The DKL-VAE shows a divergence rate comparable to the CAE and β-VAE in both cases, consistent with its similar long-horizon forecast accuracy.

From a practical standpoint, these diagnostics suggest valuable indications to construct a stable latent space for controlled flows. Encoders producing a smooth, low-dimensional latent dynamics associated with the dominant coherent oscillatory structures, here POD, yield more predictable and therefore more stable latent trajectories than purely reconstruction-driven nonlinear encoders. In addition, standardisation of the latent coordinates, inclusion of the control history in the predictor and in the stability diagnostic, and selection of the latent dimension based on long-horizon predictability rather than reconstruction energy alone improve the conditioning of the latent predictor. For flows undergoing strong control-induced bifurcations, a single global latent space may not be sufficient; in such cases, control-conditioned encoders, local latent models for different control regimes, or regime-aware mixture models may be more appropriate. Imposing stability by construction, for instance through Lyapunov-based ROMs for predictive control (Zhao *et al.* 2022), is a promising direction for future work.

Overall, the results indicate that for reliable, long-term forecasting, the stability and simplicity of the latent dynamics can be more relevant than the compression efficiency of the encoder, as long as the large-scale structures provide enough information for the application.

4. Conclusions

We have discussed the effectiveness of data-driven frameworks based on latent-space compression and multi-time-delay embedding for prediction of wake flows under control actions. We compared the classical linear POD encoder against a nonlinear CAE and two regularised variational variants, a β-VAE and a DKL-VAE, and assessed three LSTM-based predictors, including an autoregressive single-step model, a direct sequence-to-sequence model and a derivative-based model with explicit time integration.

As expected, AEs provided clearly higher compression efficiency than POD, achieving comparable reconstruction accuracy with fewer latent variables. Surprisingly, this advantage did not translate into superior long-horizon prediction. For short horizons, AE-based models can match or slightly outperform POD due to their stronger reconstruction of small-scale features. Yet, as the horizon increases, POD-based models become consistently more accurate and markedly more stable. This cross-over occurs around $\tau \approx 10$ in the truck case and is essentially present throughout the tested horizons in the more complex pinball dynamics. The violin-plot statistics confirmed that CAE-based predictors are more prone to catastrophic divergence at long horizons, whereas POD-based predictors retain higher median accuracy and a narrower spread, indicating greater reliability. These statistics are computed over ensembles of 20 independently retrained encoder–predictor pairs per case, so the reported comparison is robust to training stochasticity; for the truck, the same analysis shows that horizons beyond $\tau = 50$ are dominated by run-to-run variability, and the reported horizon is clipped accordingly.

We explain this counterintuitive result with the different complexity of the latent space. The POD yields smooth, quasi-periodic latent trajectories that are easy for LSTMs to

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

learn and extrapolate. In contrast, AE latent variables exhibit a more irregular, broadband dynamics. Even when the decoded fields retain sharper spatial detail, the increased nonlinearity and sensitivity of the latent evolution lead to faster accumulation of phase errors, shortening the reliable prediction window. Therefore, in the context of long-horizon forecasting for receding-horizon control, the simplicity and stability of the latent dynamics can be more valuable than maximal compression.

Within the linearisation viewpoint, the long-horizon predictability of POD is due to a narrow-band dynamics resembling coordinates associated with dominant Koopman spectral components of shedding-dominated wake flows. Such alignment does not generally occur in a reconstruction-trained AE. The β-VAE results do not recover the long-horizon accuracy of POD either; for the truck it in fact trails the CAE at every horizon beyond $\tau \approx 10$ (figure 8), indicating that latent disentanglement alone is not sufficient to recover this property. The same holds for the DKL-VAE evaluated on the truck and pinball cases: decomposing the latent regularisation into independently weighted information-theoretic terms matches the reconstruction and has a comparable degree of linear independence between latent coordinates of the β-VAE but does not restore the long-horizon prediction accuracy of POD, suggesting that the trade-off is not an artefact of the particular β-VAE penalty. The trade-off is similarly robust to encoder architecture choices: supplementary experiments on the fluidic pinball, varying the filter count and activation function of the CAE over ensembles of training seeds (Appendix A.3), show no notable effect on long-term predictability, with the baseline CAE remaining a well-balanced operating point.

Regarding temporal predictors, we assessed architectures with multi-time-delay embedding of the control vector within the latent space. Even under equal lookback horizons, we observed a significant sensitivity in the prediction accuracy on the choice of the LSTM model. Autoregressive single-step models degraded rapidly due to recursive error accumulation. The direct sequence and derivative-based architectures improved stability and yielded the best long-term performance, with the direct sequence model being especially aligned with MPC practice because it maps a candidate future control sequence directly into a predicted state trajectory.

Overall, the results highlight a practical principle for data-driven model-based AFC: a slightly higher-dimensional but dynamically simpler latent representation may enable more reliable prediction over longer horizons. An essential practical caveat is that the relative performance of AE- versus POD-based ROMs also depends on the capacity of the temporal predictor. In principle, more expressive sequence models could better accommodate the broadband AE latent dynamics, shifting the trade-off in favour of nonlinear encoders. However, such gains must be weighed against real-time compute, latency and power constraints for deployment on embedded platforms, motivating co-design of the encoder and predictor with hardware feasibility in mind. Furthermore, the validation in this work relied on simulation data at relatively low Reynolds numbers, characterised by the absence of noise and compact representability using POD. It can be hypothesised that, for flows exhibiting a broader bandwidth of energetically relevant frequencies in their velocity fluctuation spectra, linear techniques would eventually be superseded by nonlinear compression methods based on AEs. Future work will be targeted to assess this aspect at higher Reynolds numbers and in different flow configurations.

The results presented in this work should be interpreted within the limitations of the chosen flow configurations. The 2-D, low-Reynolds-number regime does not capture the full 3-D complexity, broadband spectral content or the chaotic dynamics of fully developed turbulence. Future developments should also focus on testing actuation-aware or regularised nonlinear encoders that explicitly trade spatial fidelity for latent predictability,

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

assessing these ideas in 3-D and higher-Reynolds-number settings and closing the loop within real-time MPC experiments to quantify the control benefit of longer stable horizons.

Acknowledgements. The Department of Aerospace Engineering at Universidad Carlos III de Madrid has been accredited as a Unit of Excellence 'María de Maeztu', a distinction awarded by the Spanish State Research Agency for the period 01/01/2026-31/12/2031.

Funding. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 949085, NEXTFLOW ERC StG). Views and opinions expressed are, however, those of the authors only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them. Funding for APC: Universidad Carlos III de Madrid (Agreement CRUE-Madroño 2026).

Declaration of generative AI in scientific writing. During the preparation of this work, the authors used Gemini, Claude and ChatGPT to improve the readability and language of the manuscript. After using these tools, the authors reviewed and edited the content as needed and assume full responsibility for the content of the published article.

Declaration of interests. The authors report no conflict of interest.

Data availability statement. The datasets, trained models and result files used in this work are openly available at Zenodo, https://doi.org/10.5281/zenodo.23508477, and the code at https://github.com/eaplab/The-balance-between-compactness-and-forecast-accuracy-of-data-driven-latent-space-ROMs.

Author contributions. A. Solera-Rico: conceptualisation, methodology, software, validation, formal analysis, investigation, data curation, writing – original draft, visualisation. P. García-Caspueñas: conceptualisation, methodology, software, validation, formal analysis, investigation, data curation, writing – original draft, visualisation. C. Sanmiguel Vila: conceptualisation, methodology, resources, writing – original draft, writing – review and editing, supervision. S. Discetti: conceptualisation, methodology, resources, writing – review and editing, supervision, project administration, funding acquisition.

Appendix A. Sensitivity analyses

This appendix reports the sensitivity studies that support the conclusions of § 3, covering the sampling frequency of the latent-space time series, the capacity and architecture of the encoders and an alternative latent-space predictor based on sparse identification of nonlinear dynamics (SINDy) (Brunton *et al.* 2016*a*). All studies use the fluidic pinball case unless stated otherwise, with the `Direct sequence` LSTM predictor as the reference dynamical model.

A.1. *Sampling frequency*

The original sampling interval was selected to resolve the dominant frequency content of the latent variables while keeping memory and computational cost within practical bounds. To assess the sensitivity of the prediction performance to this choice, we repeated the analysis with two coarser sampling rates, $\Delta t = 0.2$ and $\Delta t = 0.4$ ($2\times$ and $4\times$ the original step). The results in figure 15 at $2\Delta t$ are very close to those at the original Δt, confirming that the original sampling rate is sufficient to capture the relevant latent dynamics. The CAE is slightly more affected by undersampling than POD, consistent with the broader spectral content of its latent trajectories, but the gap is not large enough to explain alone the performance difference observed at the original Δt. The relative ranking of POD and CAE is preserved across all tested rates. The PSD of the latent coordinates, reported in figure 16, further confirms that the dominant frequencies are well resolved for both encoders, with several orders of magnitude separating the spectral peaks from the values at the Nyquist frequency of each sampling rate.

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

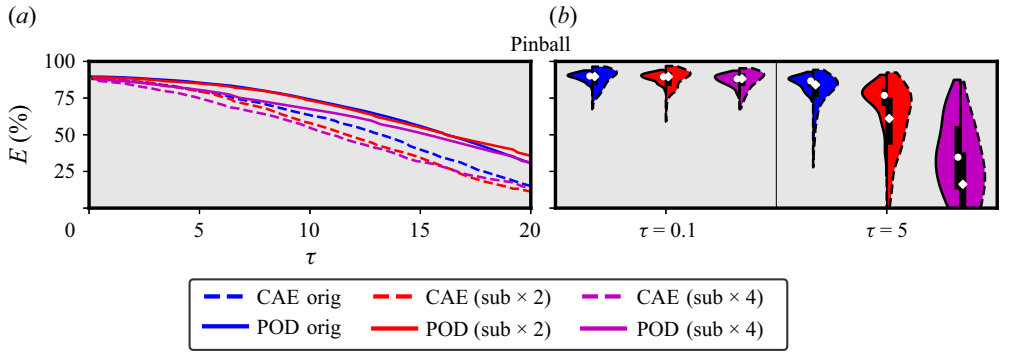


Figure 15. Effect of the sampling rate of the latent-space time series on the prediction performance for the fluidic pinball with $E = 90\,\%$ and the `Direct sequence` LSTM predictor. Three rates are compared: Δt, $2\Delta t$ and $4\Delta t$. (*a*) Prediction accuracy E as a function of the prediction horizon τ. (*b*) Violin plots of E at a short-term and a long-term horizon. White markers indicate the median (○ for POD, ◊ for CAE); black boxes span the interquartile range (25th–75th percentiles).

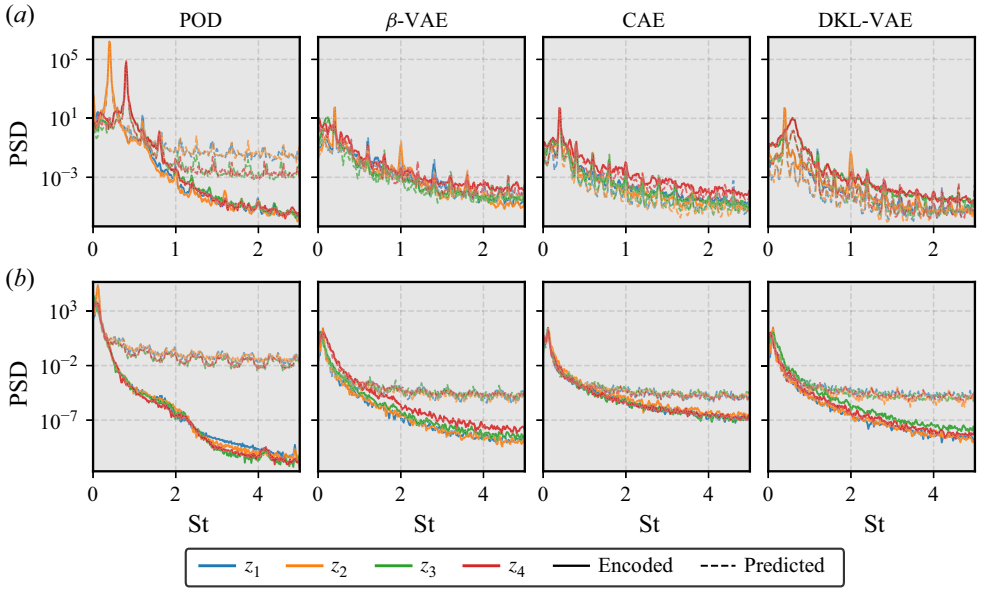


Figure 16. Power spectral density against Strouhal number $\mathrm{St} = f L_{ref}/U_\infty$ of the four leading latent coordinates (z_1–z_4) for the encoded reference (solid) and the `Direct sequence` LSTM prediction (dashed), with L_{ref} the truck width or cylinder diameter and U_∞ the reference velocity. The upper limit of each panel corresponds to the Nyquist frequency of the original sampling rate. (*a*) Truck; (*b*) fluidic pinball. Columns: POD, β-VAE, CAE and DKL-VAE. The dominant spectral peaks are well resolved at the original sampling rate for all encoders, with several orders of magnitude separating the peaks from the values at the Nyquist frequency.

A.2. *Encoder capacity*

To rule out underfitting as the source of the lower forecast accuracy of the CAE, we performed a latent-dimension sweep for both encoders, selecting d values corresponding to $E = 80\,\%$, $90\,\%$ (baseline), and $95\,\%$ of the reconstructed variance. This yields CAE models with $d = 7$, 10, 15 and POD models with $d = 13$, 22, 36, see figure 17. Increasing the reconstructed energy reduces the prediction variance for short-to-medium horizons

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

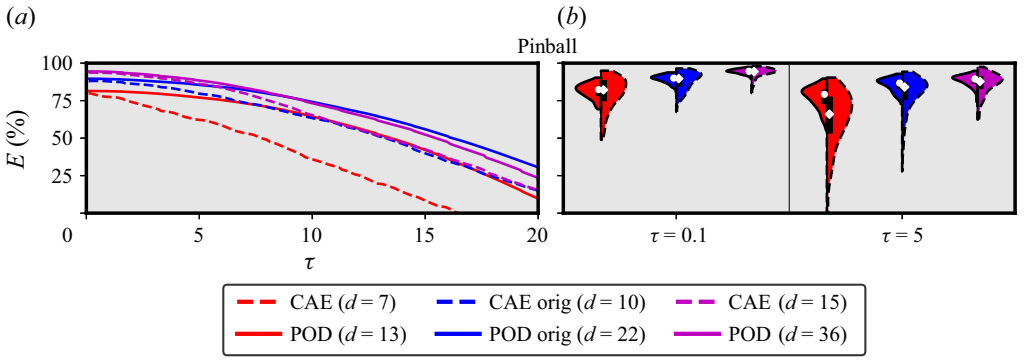


Figure 17. Effect of the encoder capacity on the prediction performance for the fluidic pinball trained with the `Direct sequence` LSTM predictor, for three reconstructed energy targets: $E = 80\,\%$, 90 % (baseline) and 95 %. (*a*) Prediction accuracy E as a function of the prediction horizon τ. (*b*) Violin plots of E at a short-term and a long-term horizon. White markers indicate the median ($\circ$ for POD, $\diamondsuit$ for CAE); black boxes span the interquartile range (25th–75th percentiles).

(up to approximately $\tau = 5$). At long horizons, however, the baseline $E = 90\,\%$ remains the most reliable configuration for both encoders. Larger latent dimensions improve short-term accuracy but degrade long-term stability, confirming that the compactness–predictability trade-off is a fundamental property of the framework rather than a consequence of underfitting at the baseline configuration. The relative ranking of POD and CAE is preserved across all tested dimensions. We also note that the capacity of the baseline CAE model was previously optimised through a hyperparameter study.

A.3. *Encoder architecture*

Beyond the latent dimension, the encoder architecture itself is a potential source of bias in the compactness–predictability comparison. To assess its influence, we varied the number of convolutional filters and the activation function of the CAE encoder–decoder ($d = 10$, all other hyperparameters fixed). The notation CAE_N denotes a variant whose convolutional layers use N times the number of filters of the baseline CAE_1; CAE_{tanh} uses a hyperbolic tangent activation in place of the baseline ELU, with the same filter count as CAE_1. For each encoder the same `Direct sequence` LSTM predictor is trained with identical settings, and each variant is independently retrained for 20 seeds, following the protocol of § 3.

The results are reported in figure 18. Across the tested range, the architecture has only a marginal effect on long-term predictability: CAE_2 performs marginally better than the baseline (median $E = 81.2\,\%$ against 79.7 % for CAE_1 at $\tau = 5$), while $\mathrm{CAE}_{1/2}$, $\mathrm{CAE}_{1/4}$ and CAE_{tanh} remain within 3.5 percentage points of the baseline, and no variant approaches the long-horizon accuracy of POD (median $E = 86.6\,\%$ at $\tau = 5$). The relative ranking of POD and the CAE variants is preserved, and the baseline CAE_1 remains a well-balanced operating point between capacity and accuracy. These results indicate that the compactness–predictability trade-off is a robust property of the encoder family and is not an artefact of the specific architecture chosen for the baseline comparison.

Appendix B. SINDy as latent-space predictor

We finally tested SINDy with control inputs (Brunton *et al.* 2016*b*) as an alternative to the LSTM predictor on both the POD and CAE latent spaces of the two flow configurations. The regularisation variant used was SINDy-ALASSO, reported by Fukami *et al.* (2021)

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

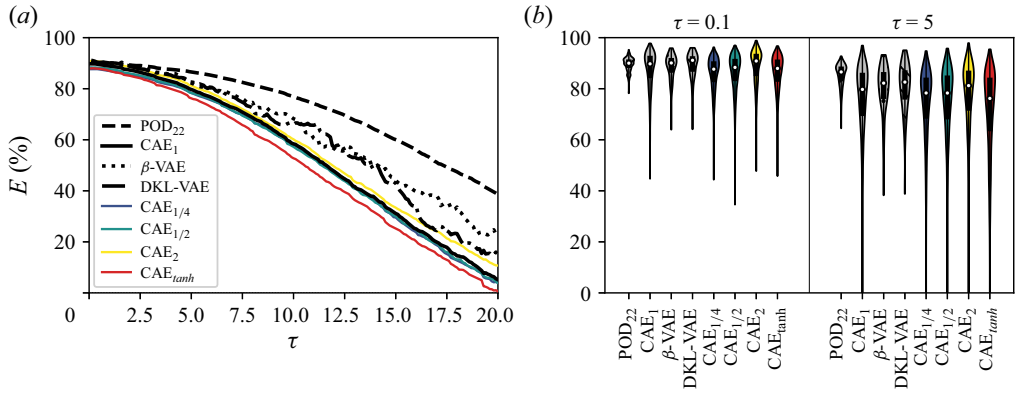


Figure 18. Median prediction accuracy $E(\tau)$ over the training-seed ensemble (a) and pooled distributions of the instantaneous E at $\tau = 0.1$ and $\tau = 5$ (b) for the paper baselines (POD_{22}, CAE_1, β-VAE, DKL-VAE) and the architecture variants, evaluated on the fluidic pinball case with $d = 10$ and the `Direct sequence` LSTM. Numerical subscripts denote the filter-count multiplier relative to CAE_1; CAE_{tanh} has the same filter count as CAE_1 but uses a hyperbolic tangent activation.

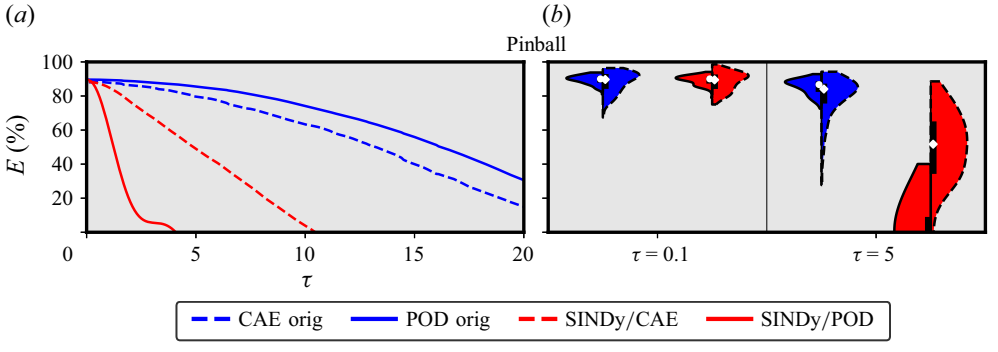


Figure 19. Comparison between SINDy and the `Direct sequence` LSTM predictor on the fluidic pinball with $E = 90\,\%$. (a) Prediction accuracy E as a function of the prediction horizon τ. (b) Violin plots of E at a short-term and a long-term horizon. White markers indicate the median ($\circ$ for POD, $\diamondsuit$ for CAE); black boxes span the interquartile range (25th–75th percentiles).

as a robust sparse identification method for complex dynamics of higher dimensionality. Forward integration of the identified sparse polynomial models without an explicit closure causes the predicted trajectories to diverge from the reference within a few convective times. For the chaotic fluidic pinball, the absence of a closure term makes the identified equations unstable for long-horizon forecasting (figure 19). The POD–SINDy model, with 22 coupled ordinary differential equations, was particularly expensive to identify and still diverged within a few convective times; the CAE–SINDy model showed a comparable instability. The truck case, with a smaller latent dimension $d = 8$, should in principle be more favourable for SINDy; nevertheless, the resulting predictors decay faster than the LSTM alternatives, figure 20. The approach completely fails to capture the dynamics of the CAE latent space and lacks long-term stability for the POD modes. Rather than acting as an effective regulariser, the structural sparsity of SINDy proves insufficient to compensate for the complexity of the controlled-wake latent dynamics under the conditions tested here. The LSTM, with its recurrent memory, provides a more effective implicit closure for this class of problems.

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

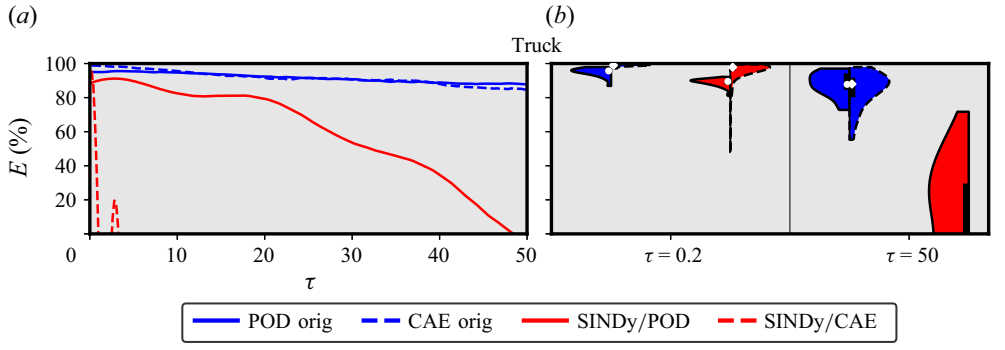


Figure 20. Comparison between SINDy and the `Direct sequence` LSTM predictor on the truck case. (*a*) Prediction accuracy E as a function of the prediction horizon τ. (*b*) Violin plots of E at a short-term and a long-term horizon. White markers indicate the median (○ for POD, ◊ for CAE); black boxes span the interquartile range (25th–75th percentiles).

References

Amico, E., Cafiero, G. & Iuso, G. 2022 Deep reinforcement learning for active control of a three-dimensional bluff body wake. *Phys. Fluids* **34** (10), 105126.

Amico, E., Serpieri, J., Iuso, G. & Cafiero, G. 2024 Flow topology of deep reinforcement learning drag-reduced bluff body wakes. *Phys. Fluids* **36** (8), 087122.

Barros, D., Borée, J., Noack, B.R., Spohn, A. & Ruiz, T. 2016 Bluff body drag manipulation using pulsed jets and Coanda effect. *J. Fluid Mech.* **805**, 422–459.

Bewley, T.R., Moin, P. & Temam, R. 2001 DNS-based predictive control of turbulence: an optimal benchmark for feedback algorithms. *J. Fluid Mech.* **447**, 179–225.

Bieker, K., Peitz, S., Brunton, S.L., Kutz, J.N. & Dellnitz, M. 2020 Deep model predictive flow control with limited sensor data and online learning. *Theor. Comput. Fluid Dyn.* **34** (4), 577–591.

Brunton, S.L. & Noack, B.R. 2015 Closed-loop turbulence control: progress and challenges. *Appl. Mech. Rev.* **67** (5), 050801.

Brunton, S.L., Noack, B.R. & Koumoutsakos, P. 2020 Machine learning for fluid mechanics. *Annu. Rev. Fluid Mech.* **52**, 477–508.

Brunton, S.L., Proctor, J.L. & Kutz, J.N. 2016*a* Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proc. Natl Acad. Sci.* **113** (15), 3932–3937.

Brunton, S.L., Proctor, J.L. & Kutz, J.N. 2016*b* Sparse identification of nonlinear dynamics with control (SINDYc). *IFAC-PapersOnLine* **49** (18), 710–715.

Bukka, S.R., Gupta, R., Magee, A.R. & Jaiman, R.K. 2021 Assessment of unsteady flow predictions using hybrid deep learning based reduced-order models. *Phys. Fluids* **33** (1), 013601.

Camacho, E.F. & Bordons, C. 2013 *Model Predictive Control*, Advanced Textbooks in Control and Signal Processing, 2nd edn. Springer.

Cattafesta, L.N. & Sheplak, M. 2011 Actuators for active flow control. *Annu. Rev. Fluid Mech.* **43**, 247–272.

Cerutti, J.J., Sardu, C., Cafiero, G. & Iuso, G. 2020 Active flow control on a square-back road vehicle. *Fluids* **5** (2), 55.

Chen, R.T.Q., Li, X., Grosse, R. & Duvenaud, D. 2018 Isolating sources of disentanglement in variational autoencoders. In *Advances in Neural Information Processing Systems* (ed. S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi & R. Garnett), vol. 31. Curran Associates, Inc.

Chourdakis, G., *et al.* 2022 preCICE v2: a sustainable and user-friendly coupling library [version 2; peer review: 2 approved]. *Open Res. Europe* **2**, 51.

Chourdakis, G., Schneider, D. & Uekermann, B. 2023 OpenFOAM-preCICE: coupling OpenFOAM with external solvers for multi-physics simulations. *OpenFOAM® J.* **3**, 1–25.

Clevert, D.-A., Unterthiner, T. & Hochreiter, S. 2015 Fast and accurate deep network learning by exponential linear units (ELUs). arXiv: 1511.07289.

Deng, N., Noack, B.R., Cornejo Maceda, G.Y. & Pastur, L.R. 2025 Nonlinear dynamics of the actuated fluidic pinball – steady, periodic, and chaotic regimes. *Chaos, Solitons Fractals* **193**, 116075.

Deng, N., Noack, B.R., Morzyński, M. & Pastur, L.R. 2020 Low-order model for successive bifurcations of the fluidic pinball. *J. Fluid Mech.* **884**, A37.

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

EIVAZI, H., LE CLAINCHE, S., HOYAS, S. & VINUESA, R. 2022 Towards extraction of orthogonal and parsimonious non-linear modes from turbulent flows. *Expert Syst. Appl.* **202**, 117038.

FRESCA, S., DEDE', L. & MANZONI, A. 2021 A comprehensive deep learning-based approach to reduced order modeling of nonlinear time-dependent parametrized PDEs. *J. Sci. Comput.* **87** (2), 61.

FRESCA, S. & MANZONI, A. 2022 POD-DL-ROM: enhancing deep learning-based reduced order models for nonlinear parametrized PDEs by proper orthogonal decomposition. *Comput. Meth. Appl. Mech. Engng* **388**, 114181.

FUKAGATA, K. & FUKAMI, K. 2025 Compressing fluid flows with nonlinear machine learning: mode decomposition, latent modeling, and flow control. *Fluid Dyn. Res.* **57** (4), 041401.

FUKAMI, K., MURATA, T., ZHANG, K. & FUKAGATA, K. 2021 Sparse identification of nonlinear dynamics with low-dimensionalized flow representations. *J. Fluid Mech.* **926**, A10.

FUKAMI, K., NAKAMURA, T. & FUKAGATA, K. 2020 Convolutional neural network based hierarchical autoencoder for nonlinear mode decomposition of fluid field data. *Phys. Fluids* **32** (9), 095110.

GARCIA, X., MIRÓ, A., SUÁREZ, P., ALCÁNTARA-ÁVILA, F., RABAULT, J., FONT, B., LEHMKUHL, O. & VINUESA, R. 2025 Deep-reinforcement-learning-based separation control in a two-dimensional airfoil. *Intl J. Heat Fluid Flow* **116**, 109913.

GLEZER, A. & AMITAY, M. 2002 Synthetic jets. *Annu. Rev. Fluid Mech.* **34**, 503–529.

GREENBLATT, D. & WYGNANSKI, I.J. 2000 The control of flow separation by periodic excitation. *Prog. Aerosp. Sci.* **36** (7), 487–545.

GUPTA, P., SCHMID, P., SIPP, D., SAYADI, T. & RIGAS, G. 2025 Mori–Zwanzig latent space Koopman closure for nonlinear autoencoder. *Proc. R. Soc. A: Math., Phys. Engng Sci.* **481** (2313), 20240259.

HIGGINS, I., MATTHEY, L., PAL, A., BURGESS, C., GLOROT, X., BOTVINICK, M., MOHAMED, S. & LERCHNER, A. 2017 beta-VAE: learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations (ICLR 2017)*.

HINTON, G.E. & SALAKHUTDINOV, R.R. 2006 Reducing the dimensionality of data with neural networks. *Science* **313** (5786), 504–507.

HOCHREITER, S. & SCHMIDHUBER, J. 1997 Long short-term memory. *Neural Comput.* **9** (8), 1735–1780.

KAISER, E., KUTZ, J.N. & BRUNTON, S.L. 2018 Sparse identification of nonlinear dynamics for model predictive control in the low-data limit. *Proc. R. Soc. A* **474** (2219), 20180335.

KINGMA, D.P. & BA, J. 2014 Adam: a method for stochastic optimization. arXiv: 1412.6980.

LECUN, Y., BOTTOU, L., BENGIO, Y. & HAFFNER, P. 1998 Gradient-based learning applied to document recognition. *Proc. IEEE* **86** (11), 2278–2324.

LITTLEWOOD, R.P. & PASSMORE, M.A. 2012 Aerodynamic drag reduction of a simplified squareback vehicle using steady blowing. *Exp. Fluids* **53** (2), 519–529.

LIU, Z., BECKERS, D. & ELDREDGE, J.D. 2025 Model-based reinforcement learning for control of strongly disturbed unsteady aerodynamic flows. *AIAA J.* **63** (10), 4105–4125.

LUSCH, B., KUTZ, J.N. & BRUNTON, S.L. 2018 Deep learning for universal linear embeddings of nonlinear dynamics. *Nat. Commun.* **9** (1), 4950.

MARRA, L., CORNEJO MACEDA, G.Y., MEILÁN-VILA, A., GUERRERO, V., RASHWAN, S., NOACK, B.R., DISCETTI, S. & IANIRO, A. 2024*a* Actuation manifold from snapshot data. *J. Fluid Mech.* **996**, A26.

MARRA, L., MEILÁN-VILA, A. & DISCETTI, S. 2024*b* Self-tuning model predictive control for wake flows. *J. Fluid Mech.* **983**, A26.

MAULIK, R., LUSCH, B. & BALAPRAKASH, P. 2021 Reduced-order modeling of advection-dominated systems with recurrent neural networks and convolutional autoencoders. *Phys. Fluids* **33** (3), 037106.

MCARTHUR, D., BURTON, D., THOMPSON, M. & SHERIDAN, J. 2016 On the near wake of a simplified heavy vehicle. *J. Fluids Struct.* **66**, 293–314.

MCNALLY, J., FERNANDEZ, E., ROBERTSON, G., KUMAR, R., TAIRA, K., ALVI, F., YAMAGUCHI, Y. & MURAYAMA, K. 2015 Drag reduction on a flat-back ground vehicle with active flow control. *J. Wind Engng Ind. Aerodyn.* **145**, 292–303.

MENIER, E., BUCCI, M.A., YAGOUBI, M., MATHELIN, L. & SCHOENAUER, M. 2023 CD-ROM: complemented deep-reduced order model. *Comput. Meth. Appl. Mech.* **410**, 115985.

MEZIĆ, I. 2013 Analysis of fluid flows via spectral properties of the Koopman operator. *Annu. Rev. Fluid Mech.* **45** (1), 357–378.

MURATA, T., FUKAMI, K. & FUKAGATA, K. 2020 Nonlinear mode decomposition with convolutional neural networks for fluid dynamics. *J. Fluid Mech.* **882**, A13.

NOACK, B.R., AFANASIEV, K., MORZYŃSKI, M., TADMOR, G. & THIELE, F. 2003 A hierarchy of low-dimensional models for the transient and post-transient cylinder wake. *J. Fluid Mech.* **497**, 335–363.

NOACK, B.R., PAPAS, P. & MONKEWITZ, P.A. 2005 The need for a pressure-term representation in empirical Galerkin models of incompressible shear flows. *J. Fluid Mech.* **523**, 339–365.

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press

Özalp, E. & Magri, L. 2025 Stability analysis of chaotic systems in latent spaces. *Nonlinear Dyn.* **113**, 13791–13806.

Pecora, L.M. & Carroll, T.L. 1990 Synchronization in chaotic systems. *Phys. Rev. Lett.* **64**, 821–824.

Rawlings, J.B., Mayne, D.Q. & Diehl, M.M. 2017 *Model Predictive Control: Theory, Computation, and Design*, 2nd edn. Nob Hill Publishing.

Robledo, I., Alfaro, J., Duro, V., Solera-Rico, A., Castellanos, R. & Sanmiguel Vila, C. 2026 Reducing base drag on road vehicles using pulsed jets optimized by hybrid genetic algorithms. *Phys. Fluids* **38** (4), 047109.

Rosenstein, M.T., Collins, J.J. & De Luca, C.J. 1993 A practical method for calculating largest Lyapunov exponents from small data sets. *Phys. D: Nonlinear Phenom.* **65** (1), 117–134.

Rowley, C.W., Mezić, I., Bagheri, S., Schlatter, P. & Henningson, D.S. 2009 Spectral analysis of nonlinear flows. *J. Fluid Mech.* **641**, 115–127.

Shams, M. & Elsheikh, A.H. 2023 Gym-preCICE: reinforcement learning environments for active flow control. *SoftwareX* **23**, 101446.

Sirovich, L. 1987 Turbulence and the dynamics of coherent structures. Part I. Coherent structures. *Q. Appl. Maths* **45** (3), 561–571.

Solera-Rico, A., Sanmiguel Vila, C. & Discetti, S. 2026 A framework for realisable data-driven active flow control using model predictive control applied to a simplified truck wake. *Engng Appl. Artif. Intell.* **182**, 115580.

Solera-Rico, A., Sanmiguel Vila, C., Gómez-López, M., Wang, Y., Almashjary, A., Dawson, S.T.M. & Vinuesa, R. 2024 β-Variational autoencoders and transformers for reduced-order modelling of fluid flows. *Nat. Commun.* **15** (1), 1361.

Storms, B.L., Ross, J.C., Heineck, J.T., Walker, S.M., Driver, D.M., Zilliac, G.G. & Bencze, D.P. 2001 An experimental study of the ground transportation system (GTS) model in the NASA Ames 7 by 10 ft wind tunnel. *Tech. Rep.* NASA/TM-2001-209621. NASA Ames Research Center.

Suárez, P., Alcántara-Ávila, F., Rabault, J., Miró, A., Font, B., Lehmkuhl, O. & Vinuesa, R. 2025 Flow control of three-dimensional cylinders transitioning to turbulence via multi-agent reinforcement learning. *Commun. Engng* **4** (1), 113.

Taira, K., Brunton, S.L., Dawson, S.T.M., Rowley, C.W., Colonius, T., McKeon, B.J., Schmidt, O.T., Gordeyev, S., Theofilis, V. & Ukeiley, L.S. 2017 Modal analysis of fluid flows: an overview. *AIAA J.* **55** (12), 4013–4041.

Wang, Z., Tirelli, I., Discetti, S. & Ianiro, A. 2026 Information decomposition for disentangled and interpretable manifold learning of fluid flows via variational autoencoders. *Chin. J. Aeronaut.* **39** (11), 104342.

Yang, S., Vinuesa, R. & Kang, N. 2025 Model-agnostic AI framework with explicit time integration for long-term fluid dynamics prediction. *J. Comput. Des. Engng* **12** (10), 133–153.

Zhao, T., Zheng, Y., Gong, J. & Wu, Z. 2022 Machine learning-based reduced-order modeling and predictive control of nonlinear processes. *Chem. Engng Res. Des.* **179**, 435–451.

Zhu, Y., Sun, Q., Xiao, D., Yao, J. & Mao, X. 2024 Compressed neural networks for reduced order modeling. *Phys. Fluids* **36** (5), 057121.

https://doi.org/10.1017/jfm.2026.12075 Published online by Cambridge University Press