CAMBRIDGE
UNIVERSITY PRESS

RESEARCH ARTICLE

Decoder decomposition for the analysis of the latent space of nonlinear autoencoders with wind-tunnel experimental data

Yaxin Mo[1], Tullio Traverso[1,2] and Luca Magri[1,2,3]

[1]Department of Aeronautics, Imperial College London, London, UK
[2]The Alan Turing Institute, London, UK
[3]Politecnico di Torino, DIMEAS, Torino, Italy
Corresponding author: Luca Magri; Email: l.magri@imperial.ac.uk

Abstract

Turbulent flows are chaotic and multi-scale dynamical systems, which have large numbers of degrees of freedom. Turbulent flows, however, can be modeled with a smaller number of degrees of freedom when using an appropriate coordinate system, which is the goal of dimensionality reduction via nonlinear autoencoders. Autoencoders are expressive tools, but they are difficult to interpret. This article aims to propose a method to aid the interpretability of autoencoders. First, we introduce the *decoder decomposition*, a post-processing method to connect the latent variables to the coherent structures of flows. Second, we apply the decoder decomposition to analyze the latent space of synthetic data of a two-dimensional unsteady wake past a cylinder. We find that the dimension of latent space has a significant impact on the interpretability of autoencoders. We identify the physical and spurious latent variables. Third, we apply the decoder decomposition to the latent space of wind-tunnel experimental data of a three-dimensional turbulent wake past a bluff body. We show that the reconstruction error is a function of both the latent space dimension and the decoder size, which are correlated. Finally, we apply the decoder decomposition to rank and select latent variables based on the coherent structures that they represent. This is useful to filter unwanted or spurious latent variables or to pinpoint specific coherent structures of interest. The ability to rank and select latent variables will help users design and interpret nonlinear autoencoders.

Impact Statement

Nonlinear dimensionality reduction by autoencoders can efficiently compress high-dimensional data into a low-dimensional latent space, but the results may be difficult to interpret. We propose the *decoder decomposition* to select and rank the latent variables based on the coherent structures that they represent. This opens opportunities for building interpretable models with nonlinear autoencoding.

1. Introduction

Turbulent flows are nonlinear and multi-scale systems, which have large numbers of degrees of freedom. High-fidelity simulations of turbulent flows can be performed by solving the governing equations on fine spatiotemporal grids, but the computational cost can be prohibitively high (Rowley and Dawson, 2017).

When computationally cheaper modeling of turbulent flows is needed, reduced-order models are applied to approximate the flows with fewer degrees of freedom (Noack et al., 2011; Rowley and

Dawson, 2017). Commonly, reduced-order models are constructed via projection-based methods, such as proper orthogonal decomposition (POD) (Noack et al., 2011; Rowley and Dawson, 2017). POD is a decomposition method dating back to 1970 (Lumley, 1970), which enables the computation of an optimal linear subspace based on the energy norm (Taira et al., 2017). Each POD mode is associated with an energy, which ranks the importance of the mode, and POD modes can be interpreted as coherent structures of flows (Alfonsi and Primavera, 2007; Kevlahan et al., 1994; Rigas et al., 2014). There are other methods of linear decomposition, such as the spectral POD (Lumley, 1970; Schmidt and Colonius, 2020), dynamic mode decomposition (Schmid, 2010; Tu et al., 2014), and wavelet analysis (Albukrek et al., 2002), which have been employed for dimensionality reduction and discovery of coherent structures. These linear methods are relatively straightforward to implement, but may require large numbers of modes to reduce the approximation error for accurate modeling of nonlinear flows (Alfonsi and Primavera, 2007; Murata et al., 2020). On the other hand, nonlinear reduced-order modeling seeks nonlinear manifolds onto which dimensionality reduction can be performed to approximate the dynamics (Magri and Doan, 2022; Racca et al., 2023).

Machine learning has been increasingly applied for dimensionality reduction of fluids, in particular via nonlinear autoencoders (AEs) (Csala et al., 2022; Doan et al., 2023; Eivazi et al., 2022; Fukami et al., 2021; Fukami et al., 2024; Fukami and Taira, 2023; Magri and Doan, 2022). AEs consist of an encoding and a decoding part: the encoding part maps the input (the physical flow field) into a lower-dimensional latent space, whereas the decoding part maps the latent space back to the physical space. The purpose of the AEs is to approximate the identity mapping. The turbulent system's dynamics can be predicted on the low-dimensional latent space with sequential methods such as reservoir computers (Doan et al., 2021; Racca et al., 2023) and long short-term memory networks (Nakamura et al., 2021) for flow forecasting. The latent space of AE, which is a nonlinear manifold (Magri and Doan, 2022), may be difficult to interpret. This is because the latent variables are entangled and their coordinate bases may not be locally orthogonal (Magri and Doan, 2022).

The mode-decomposing AE (MD-AE) was developed by Murata et al. (Murata et al., 2020) to improve the interpretability of nonlinear decomposed fields. MD-AEs assign a decoder to each variable of the latent space vector (latent variable), and then superpose the single-decoder outputs to generate the MD-AE output. MD-AEs improve the visualization of flow decomposition because the effect of each latent variable is isolated in the decoding part, unlike standard AEs in which one decoder contains the effect of all latent variables. However, an AE outperforms an MD-AE in terms of reconstruction error with the same latent space dimension (Eivazi et al., 2022; Murata et al., 2020). This is because the design of MD-AEs does not capture the nonlinear coupling of the decoded latent variables. For highly nonlinear flows, the difference between the reconstruction errors may be significant (Csala et al., 2022). Other AE architectures, such as the hierarchical AE (Fukami et al., 2020) and the β-variational AE (Eivazi et al., 2022; Solera-Rico et al., 2024), have been proposed to disentangle the latent variables to improve the interpretability, but with a larger reconstruction error as compared to an AE. The decomposed fields, nonetheless, may be still influenced by more than one latent variable, so the exact effect of each latent variable on the output is difficult to isolate.

A comparison between mode decomposition methods, including MD-AE and POD, can be found in the work by Csala et al. (Csala et al., 2022). Effort have been made to obtain interpretable AE latent space for specific problems (Fukami and Taira, 2023; Fukami et al., 2024), but general interpretability of AE latent space is an open problem (Magri and Doan, 2022; Vinuesa and Sirmacek, 2021).

The overarching objective of this article is to aid the interpretability of the latent space of common AEs. For this purpose, we propose the *decoder decomposition*, which is a post-processing method for raking and selecting the latent space of nonlinear AEs. The decoder decomposition is based on POD modes and the decoders' sensitivities computed as gradients. Specifically, the objectives are to (i) propose the decoder decomposition, which disentangles the contribution of latent variables in the decoded field; (ii) analyze and verify the decoder decomposition of two commonly used AEs (the standard AE and the MD-AE) with synthetic data from numerical simulation of the unsteady two-dimensional wake past a cylinder; and

(iii) apply the decoder decomposition to gain physical insight into a realistic flow (three-dimensional turbulent flow from wind-tunnel experiments) and isolate the latent variables of physical interest with a filtering strategy. The three-dimensional turbulent flow is the high-speed wake past a bluff body.

This article is organized as follows. First, we provide an overview of POD in Section 2 and detail our datasets in Section 3. Next, we introduce the AE architectures in Section 4 and propose the decoder decomposition in Section 5. We apply the decoder decomposition to decompose the unsteady laminar cylinder wake and investigate the impact of the dimension of the latent space on the interpretability of AEs in Section 6. We decompose the wind-tunnel wake in Section 7 and demonstrate how to rank latent variables to filter for coherent structures in the output of the AEs. We present our conclusions in Section 8.

2. Proper orthogonal decomposition

Let $Q \in \mathbb{R}^{N \times N_t}$ be a generic dataset of some fluctuating quantites from a flow field such that each snapshot is a column, where N is the product of the number of grid points, the number of variables, and N_t is the number of snapshots. The ith row, $Q_{i,:}$, is the time series of the measured quantity at grid point i. The tth column, $Q_{:,t}$, is the snapshot of the measured quantities at the discrete time step t. (In this article, either the fluctuating velocities or the pressure is measured.) The covariance matrix $C \in \mathbb{R}^{N \times N}$ is

$$C = \frac{1}{N_t - 1} W_p Q Q^T W_p^T = \Phi \Lambda \Phi^T, \tag{2.1}$$

where W_p is the POD weight matrix given to each element of Q, Λ is the diagonal matrix of the eigenvalues, and $\Phi \in \mathbb{R}^{N \times N}$ is the matrix of the eigenvectors. The weight matrix for each dataset is given in Section 3. The ith column of the matrix of eigenvalues, $\Phi_{:,i}$, is the POD mode i, which represents the ith principal axis of the space occupied by the observed data ranked by the eigenvalues, that is, the energy of the mode (Taira et al., 2017; Weiss, 2019). The matrix of temporal coefficients A is obtained by projecting the data matrix onto the POD modes

$$A = Q^T \Phi, \tag{2.2}$$

which contains the temporal coordinates of the data snapshots along the principal axes (Weiss, 2019). The ith row of A, denoted $A_{i,:}$, contains the time series of the time coefficient of the ith POD mode. The time coefficient for the mode i at time step t is $A_{i,t}$. By setting $N_m < N$ as the truncated number of modes for reconstructing the flow, we approximate the flow field in the subspace spanned by the first N_m POD modes

$$\tilde{Q} = \sum_{i=1}^{N_m} \Phi_{:,i} A_{i,:}^T. \tag{2.3}$$

If $N_m = N$, then $\tilde{Q} = Q$, that is, no approximation is made, and only a linear change of coordinate system is performed.

3. Datasets and preprocessing

Two datasets are considered in this article. First, an unsteady laminar wake behind a cylinder, which is the benchmark case whose dynamics are well-known (Section 3.1). Second, a wind-tunnel dataset of a three-dimensional turbulent bluff body wake (Brackston, 2017) (Section 3.2). Both datasets are characterized by POD analysis.

3.1. Unsteady laminar wake of a two-dimensional cylinder

The unsteady laminar wake behind a 2D circular cylinder at Reynolds number $Re = 100$ is generated by solving the dimensionless Navier–Stokes equations
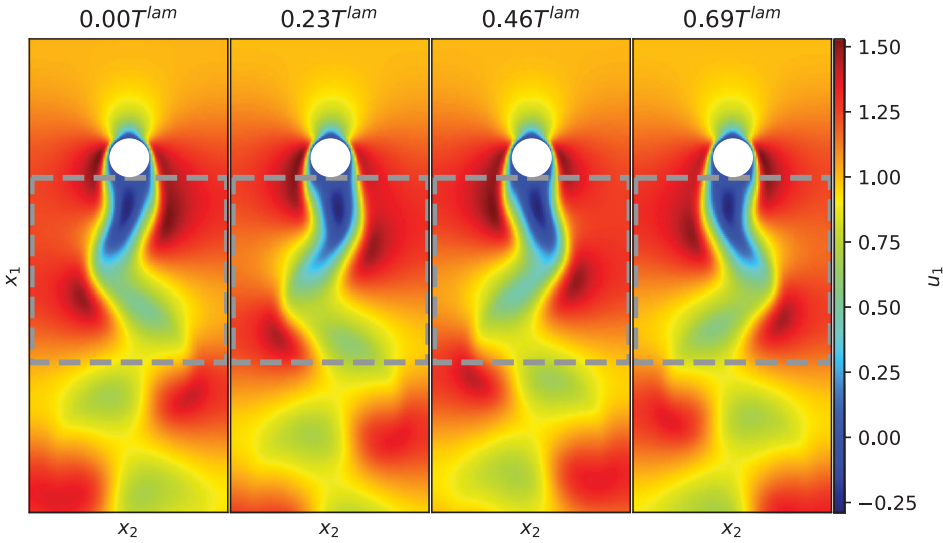
Figure 1. *Snapshots of the streamwise velocity of the laminar wake dataset at different times within the same period. A vortex shedding period is denoted with T^{lam}. The area bounded by the gray box is used for training.*

$$\begin{cases} \nabla \cdot \boldsymbol{u} = 0 \\ \dfrac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} = -\nabla p + \dfrac{1}{Re}\varDelta \boldsymbol{u} \end{cases}, \tag{3.1}$$

where the vector $\boldsymbol{u} \in \mathbb{R}^{N_u}$ with $N_u = 2$ is the velocity, p is the pressure, and t is the time. The velocity and length are nondimensionalized by the inlet velocity U_∞ and the diameter of the cylinder D. The computation domain has size $L_1 = 12$, $L_2 = 5$, and $L_3 = 1$ (Figure 1), divided uniformly into 513, 129, and 8 nodes in streamwise, wall-normal and spanwise directions[1], respectively. The boundary conditions are Dirichlet boundary condition at $x_1 = 12$; slip walls at $x_2 = 0$ and $x_2 = 5$; and the periodic at $x_3 = 0$ and $x_3 = 1$. The center of the cylinder is at $(3,2.5)$. The dataset is simulated with direct numerical simulation using Xcompact3D (Bartholomew et al., 2022). A time step $\Delta t = 0.0002$ is chosen to satisfy the Courant–Friedrichs–Lewy condition. The numerical schemes are the sixth-order compact scheme in space (Laizet and Lamballais, 2009) and the third-order Adams–Bashforth in time. The simulation matches the results of Ganga Prasath et al. (Ganga Prasath et al., 2014).

The transient period of the first 100 time units at the beginning of the simulation is discarded to ensure the dataset contains only the periodic flow (Ganga Prasath et al., 2014). Snapshots of the streamwise velocity, u_1, are saved every 0.125 time units, giving over 40 snapshots per vortex shedding period, T^{lam}. The final dataset contains 720 snapshots. We consider a 200-by-129 grid behind the cylinder (gray box in Figure 1). The domain includes areas from 0 to $4.5D$ downstream of the body, capturing the near wake and the vortex shedding (Zdravkovich, 1997). The dataset used by the AEs, $\boldsymbol{U} \in \mathbb{R}^{N_1 \times N_2 \times N_t}$, consists of the fluctuating streamwise velocity, where $N_t = 720$ is the number of snapshots, $N_1 = 200$ and $N_1 = 129$ are the number of grid points in the streamwise and wall-normal directions, respectively.

We perform POD on the dataset $\boldsymbol{U}$ to obtain the POD modes Φ^{lam} and the matrix of time coefficients A^{lam}, referred to as the "data modes" and "data time coefficients," respectively, where *lam* stands for "laminar." Examination of the eigenvalues shows that the data modes are out-of-phase pairs, the two modes in a pair contain a similar amount of flow energy (Figure 2, left panel) and their time coefficients are

[1] XCompact3D only accepts 3D domains. Therefore, the 2D flow is simulated as one x_1–x_2 plane of the wake of an infinitely long cylinder.
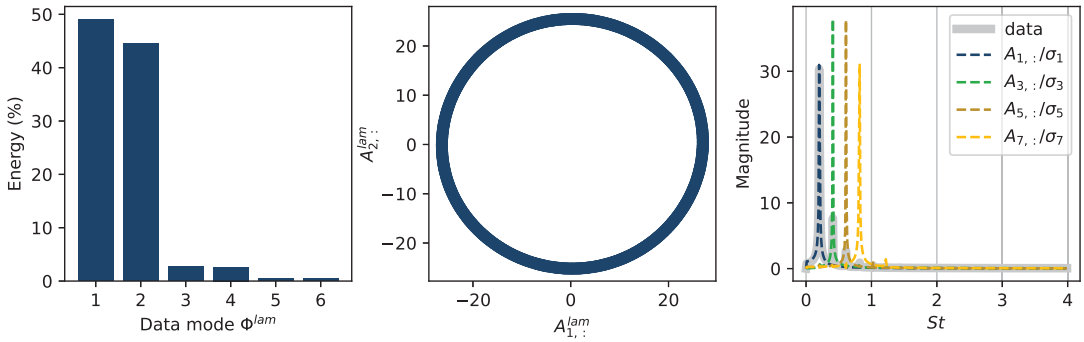
Figure 2. *POD of the laminar wake* **U**. *Left: the percentage energy contained in the first six POD modes of the unsteady wake Φ^{lam} (data mode). Data modes 1 and 2, 3 and 4, and 5 and 6 contain similar flow energy and oscillate at the same frequency but out of phase. Center: phase plot of the first two data time coefficients. Right: the frequency spectrum of the data and the data time coefficients 1, 3, 5, and 7, normalized by their standard deviations. The data contain the vortex shedding frequency and its harmonics. (Since each pair has the same frequency spectrum, only the odd data modes are shown here.)*
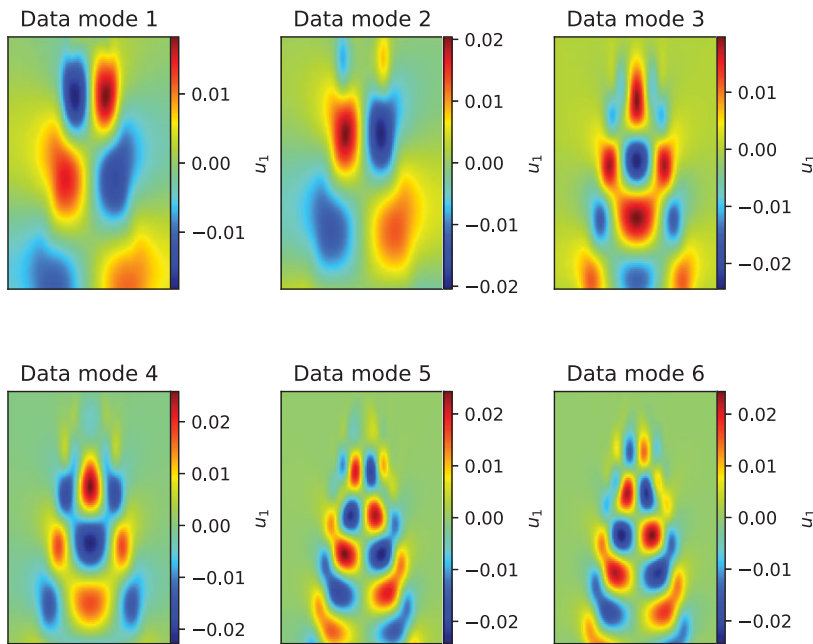

Figure 3. *The first six POD modes (data modes) of the unsteady wake behind a cylinder dataset $\Phi^{lam}_{:,1}, \ldots, \Phi^{lam}_{:,6}$.*

90° out-of-phase (Figure 2, center panel). Figure 3 shows the first six data modes of the laminar dataset. The first two data modes represent the vortex shedding, at the vortex shedding frequency, and the higher modes oscillate at the harmonic frequencies (Loiseau et al., 2020). The magnitude of the fast Fourier transform in time, summed over the grid points, is used to measure the overall frequency content of the dataset, shown in Figure 2 (right panel). The frequency content of the flow is described in terms of the Strouhal number, $St = fD/U_\infty$, where f is the frequency measured in Hz. The dominant frequency is the vortex shedding frequency at $St = 0.23$, and the dataset also contains higher harmonic frequencies. Each pair of data modes oscillates at a single, distinct frequency (Figure 2, right panel). The first pair, comprised of data modes 1 and 2, oscillates at the vortex shedding frequency, while the subsequent pairs,

composed of data modes 3 and 4, oscillate at the first harmonic, and so forth. The concentration of the flow energy in the leading data modes and the distinct frequencies of the data time coefficients make the unsteady wake behind a cylinder suitable for dimensionality reduction and interpretation with POD.

3.2. Turbulent wake of a bluff body from wind-tunnel experiments

The turbulent wake of an axisymmetric body from wind-tunnel experiments (Rigas et al., 2014; Brackston, 2017) is employed in the second part of this article. Figure 4 shows the experimental setup. The axisymmetric body has a diameter of $D = 196.5$ mm and a length-to-diameter ratio $L/D = 6.5$. Pressure measurements at the base of the body are collected via 64 static pressure sensors placed on a regular polar grid, with eight grid points in the radial direction and eight grid points in the azimuthal direction (Figure 4). We provide here an overview of the dataset. For a more in-depth analysis of the dataset and the experimental setup, the readers are referred to (Rigas et al., 2014).

The mean and root-mean-squared (RMS) pressures are both axisymmetric (Figure 5). The power spectral density (PSD) of the pressure data is shown in Figure 5 (right), premultiplied by the Strouhal number to improve the visualization (Rigas et al., 2014). The peaks at $St \approx 0.002$, 0.06, and 0.2 are associated with the 3D rotation of the symmetry plane, pulsation of the vortex core, and vortex shedding, respectively.

The weighted POD (each data point is weighted by the area of the element of the polar grid) is applied to the dataset (Rigas et al., 2014; Brackston, 2017). The resulting POD modes Φ^{exp} and time coefficients A^{exp} are the data modes and data time coefficients, respectively, where *exp* stands for "experimental." Figure 6 shows the energy contained in each data mode of the wind-tunnel dataset and the cumulative energy of the modes. The flow energy is spread over more modes than the laminar case, and 21 data modes are needed to reconstruct the dataset to recover 95% of the energy. The modes and the premultiplied PSD of their time coefficients are shown in Figure 7. Data modes 1 and 2 are antisymmetric and have frequency peaks at $St \approx 0.002, 0.1$ and 0.2. As a pair, data modes 1 and 2 represent vortex shedding and the "very-low-frequency" 3D rotation of the symmetry plane (Rigas et al., 2014). Data mode 3 has a frequency peak at $St \approx 0.06$ and is associated with the pulsation of the vortex core (Rigas et al., 2014; Brackston, 2017). The
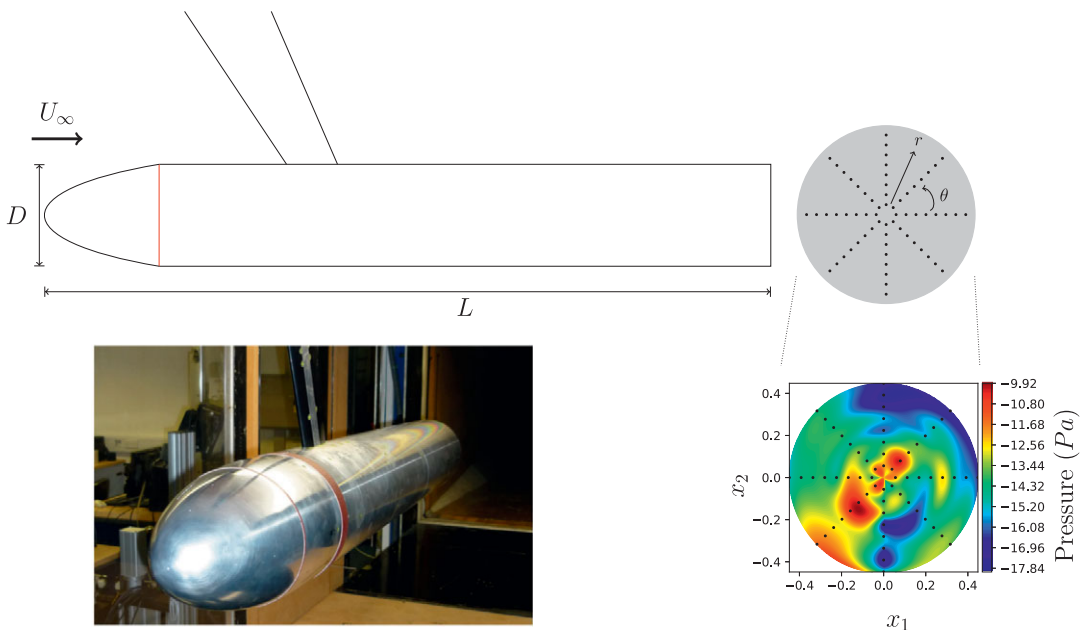

Figure 4. *Experimental setup, reproduced from (Rigas, 2021). The dimensions x_1 and x_2 are the measured location nondimensionalized by the diameter D. The black dots mark the location of the pressure sensors.*
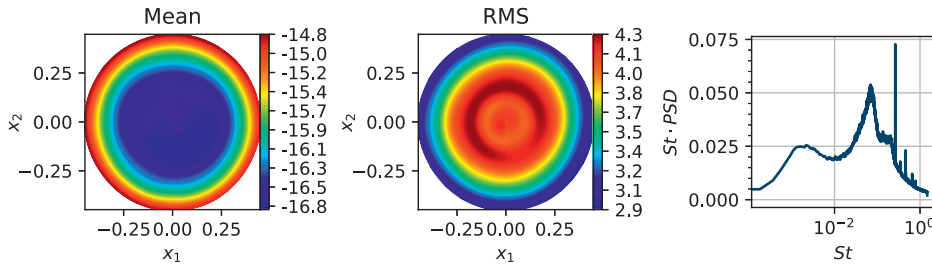
Figure 5. *The wind-tunnel pressure dataset* **P**. *Left: Mean pressure. Center: RMS pressure. Right: The premultiplied PSD (St· PSD) of the wind-tunnel dataset, with peaks at St $\approx 0.002, 0.06$ and St ≈ 0.2 and its harmonics. The peaks correspond to the three-dimensional rotation of the wake, the pulsation of the vortex core, and the vortex shedding and its harmonics, respectively.*
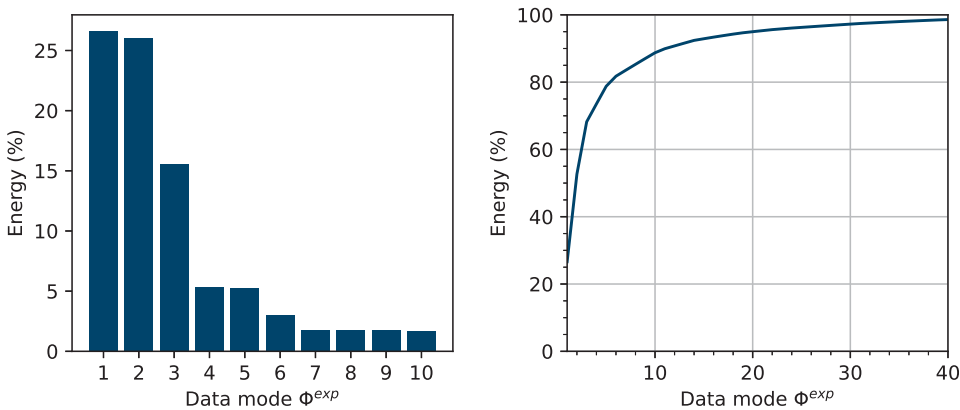

Figure 6. *POD of the wind-tunnel dataset* **P**. *Left: Percentage energy of the first 10 data modes. Right: Cumulative percentage energy of POD modes. The reconstruction of the pressure dataset to 95% energy needs 21 data modes.*

data modes 4 and 5 contain the subharmonic frequencies. Data modes 1–5 contain the coherent structures of the dataset. We base our analysis and latent variable selection on the interpretation of the data modes Φ^{exp} provided here. We consider the fluctuating field, following the same logic as for the laminar cylinder wake dataset. The final experimental pressure dataset employed for training is $\boldsymbol{P} \in \mathbb{R}^{N \times N_t}$, where $N_t = 420000$ is the number of available snapshots in time, sampled at a sampling frequency of 225 Hz and $N = 64$ is the number of pressure sensors. The dataset $\boldsymbol{P}$ contains data collected for over 1800 s.

4. Autoencoder architectures

An AE approximates the identity function by means of an encoder and a decoder. We define a generic $\boldsymbol{Y} \in \mathbb{R}^{N^* \times N_t}$, where N^* is the spatial dimension. For all time steps, the encoder $F_{en} : \mathbb{R}^{N^*} \to \mathbb{R}^{N_z}$ maps the snapshot of the data at time t, $\boldsymbol{Y}_{:,t}$ into a latent space, represented by a latent vector $\boldsymbol{Z}_{:,t}$, where $N_z \ll N^*$ is the dimension of the latent space. The ith latent variable is $\boldsymbol{Z}_{i,:}$. The encoder, F_{en}, is a composition of layers $f_{en}^{(1)}, f_{en}^{(2)}, \ldots, f_{en}^{(n)}$ and activation functions $\theta_{en}^{(1)}, \theta_{en}^{(2)}, \ldots, \theta_{en}^{(n)}$ applied to each layer. The decoder, F_{de}, is a composition of layers $f_{de}^{(1)}, f_{de}^{(2)}, \ldots, f_{de}^{(n)}$ and the activation functions $\theta_{de}^{(1)}, \theta_{de}^{(2)}, \ldots, \theta_{de}^{(n)}$ applied to each layer

$$
\begin{aligned}
F_{en} &= \theta_{en}^{(n)} \circ f_{en}^{(n)} \circ \theta_{en}^{(n-1)} \circ f_{en}^{(n-1)} \ldots \circ \theta_{en}^{(1)} \circ f_{en}^{(1)}, \\
F_{de} &= \theta_{de}^{(n)} \circ f_{de}^{(n)} \circ \theta_{de}^{(n-1)} \circ f_{de}^{(n-1)} \ldots \circ \theta_{de}^{(1)} \circ f_{de}^{(1)},
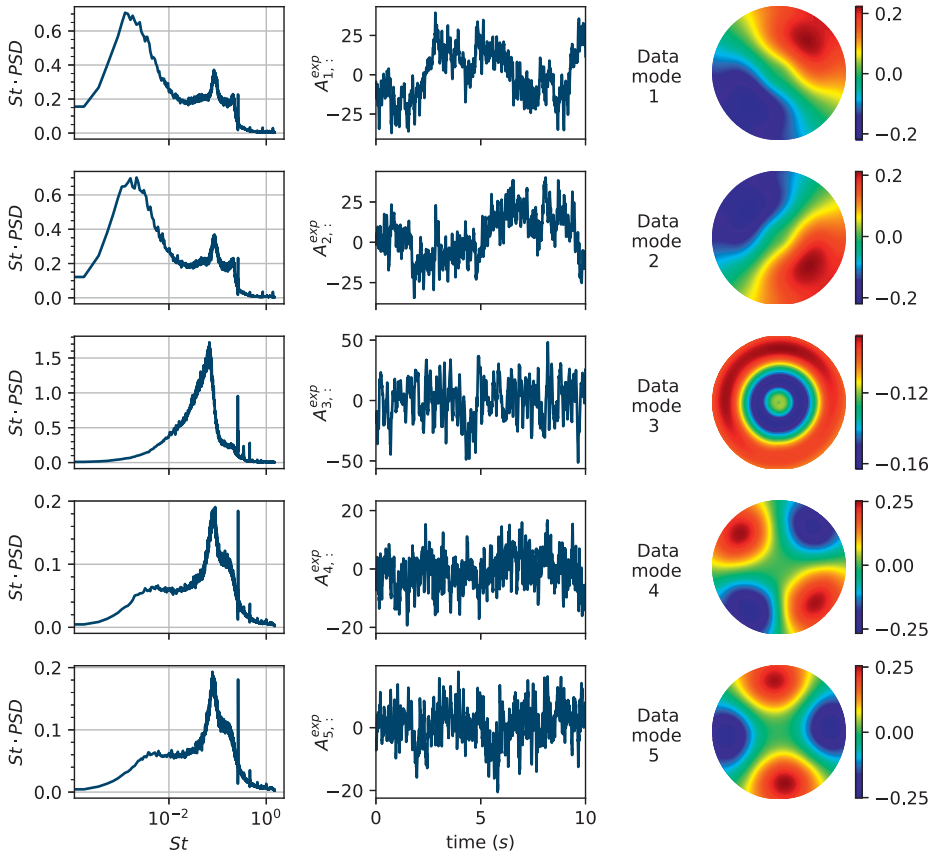\end{aligned}
\tag{4.1}
$$

Figure 7. *Left: The premultiplied PSD (St PSD) of their associated time coefficients $A_{1,:}^{exp}$ to $A_{5,:}^{exp}$. Middle: The temporal evolution of the data time coefficients, $A_{1,:}^{exp}$ to $A_{5,:}^{exp}$, for the first 10 seconds of the experiment. Right: The first five data modes of the wind-tunnel dataset $\Phi_{:,1}^{exp}$ to $\Phi_{:,5}^{exp}$.*

where n is the number of layers. Each layer ($f_{en}^{(*)}$ or $f_{de}^{(*)}$) maps the output of the previous layer to the input of the next layer, and is a function of the trainable parameters ($\omega_{en}^{(*)}$ or $\omega_{de}^{(*)}$). Each layer is either a convolution or a linear mapping depending on the test case and the datasets. Details of the layers and activation functions are explained in Sections 4.1 and 4.2.

The decoding part maps the latent vector at t back to the original space, with the output $\widehat{Y}_{:,t}$ approximating the input $Y_{:,t}$. The AE is trained with snapshots at all t to output $\widehat{Y}$, which approximates the input Y. The error between $\widehat{Y}$ and Y is measured with the mean squared error (MSE)

$$MSE(Y, \widehat{Y}) = \frac{1}{N^* \times N_t} \sum_{i=1}^{N^*} \sum_{t=1}^{N_t} (Y_{i,t} - \widehat{Y}_{i,t})^2. \qquad (4.2)$$

The parameters of the AE, $\omega = \left\{ \omega_{en}^{(1)}, \ldots, \omega_{en}^{(n)}, \omega_{de}^{(1)}, \ldots, \omega_{de}^{(n)}, \right\}$, are obtained by minimizing the MSE

$$\omega^* = \arg\min_{\omega} MSE\left(Y, \widehat{Y}\right). \qquad (4.3)$$

This analysis includes two different AE architectures—the standard AE (Section 4.1) and the MD-AE (Section 4.2). Because we use networks with no bias and train on the fluctuating fields, the trivial latent vector **0** maps to the trivial output **0**. Two types of intermediate layers are employed—convolutional layers for the laminar dataset, and feedforward layers for the wind-tunnel dataset to handle the polar grid.

Table 1. *Different autoencoder architectures and the training datasets*

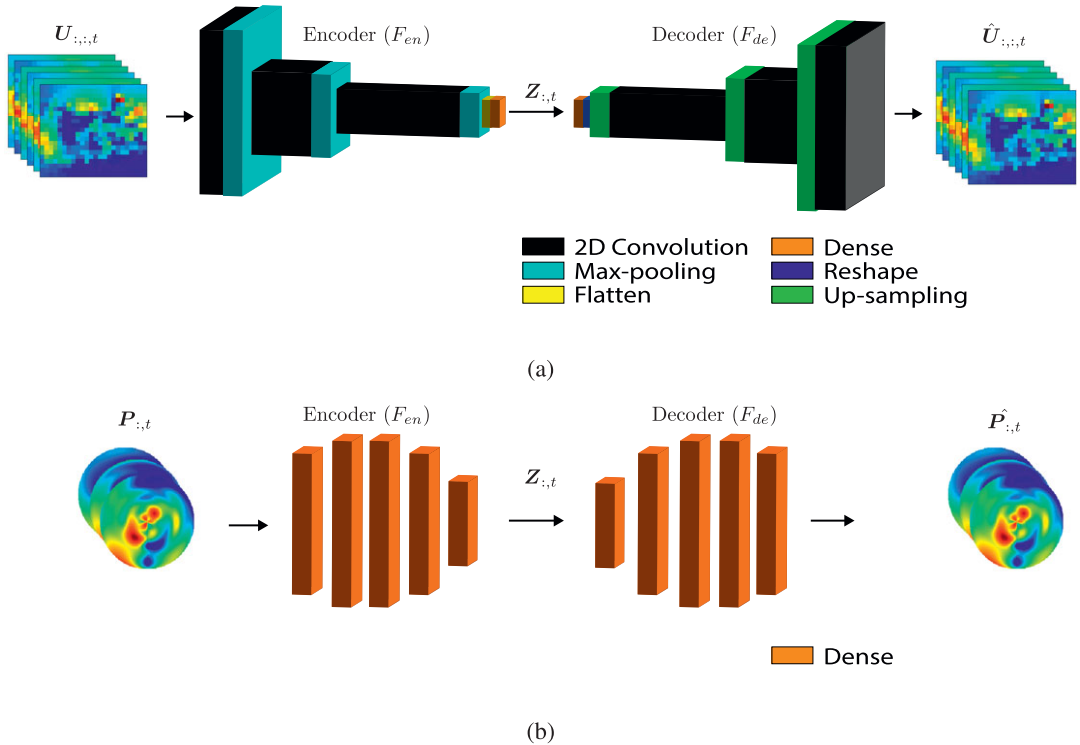| Results in section	Autoencoder architecture	Inputs Y	Spatial dimension N^*
6.1	MD-AE (CNN)	$\mathbf{U}$	200×129
6.2	AE (CNN)	$\mathbf{U}$	200×129
7	AE (feedforward)	$\mathbf{P}$	64

(a)

(b)

Figure 8. *The schematics of the standard autoencoder (AE). Different AEs are employed for different datasets and tests. The AE architecture and dataset for each test are listed in Table 1a. (a) AE for decomposing the laminar cylinder wake* $\mathbf{U}$ *with three convolution layers in both the encoder and the decoder. The hyperparameters are in Tables A.1 and A.2 in the Appendix. (b) AE for decomposing the wind-tunnel pressure data* $\mathbf{P}$ *with five feedforward layers. The input is a flattened vector of measurement taken from all sensors at time t. The hyperparameters are given in Tables A.3 and A.4 in the Appendix.*

In this section, we define the AEs for a generic input dataset Y, where the spatial dimension N^* depends on the dataset under investigation (numerical and wind-tunnel datasets) and the AE architectures. The inputs and AE architectures are summarized in Table 1.

4.1. Standard autoencoders

We refer to a standard AE as an AE with the structure shown in Figure 8, which consists of one encoder F_{en} and one decoder F_{de}. In compact notation, an AE is

$$\begin{aligned} \boldsymbol{Z}_{:,t} &= F_{en}(\boldsymbol{Y}_{:,t};\omega), \\ \widehat{\boldsymbol{Y}}_{:,t} &= F_{de}(\boldsymbol{Z}_{:,t};\omega). \end{aligned} \tag{4.4}$$

Details of the layers are given in the Appendix A (Tables A.1–A.4). All neural networks that have the convolutional encoder and decoder share the same layers and filter size (Tables A.1 and A.2). The feedforward AEs have layers given in Tables A.3 and A.4 unless stated otherwise. For all the AEs, we use the activation function "tanh" because the outputs contain both positive and negative values. (We performed further tests in which we found that the AEs do not train well with an activation function that does not contain 0, such as the "sigmoid.") All hyperparameters are given in Table A.5.

4.2. Mode-decomposing autoencoders

An MD-AE associates one decoder to each latent variable (Figure 9), so we can visualize the effect of each latent variable separately, which makes it easier to interpret (Murata et al., 2020). The MD-AE is

$$\boldsymbol{Z}_{:,t} = F_{en}(\boldsymbol{Y}_{:,t}; \omega),$$

$$\widehat{\boldsymbol{Y}}_{:,t} = \sum_{i=1}^{N_z} \underbrace{F_{de}^i(\boldsymbol{Z}_{:,t}^i; \omega)}_{\equiv \boldsymbol{M}_{:,:,t}^i} \tag{4.5}$$

where $\boldsymbol{M}^i$ denotes the ith decomposed field of the MD-AE and $\boldsymbol{M}_{:,:,t}^i$ is the ith decomposed field at time step t, corresponding to the output of the ith decoder. The MD-AE is only applied to the laminar cylinder wake case (the reasoning behind this decision is explained in Section 6).

5. The decoder decomposition

The dynamics of POD modes are encapsulated in their time coefficient. To gain physical insight into the decomposed fields, we propose the decoder decomposition to obtain a relationship between the data time coefficients (Eq. (2.2)) and the latent variables trained with the same data. The decoder decomposition is a post-processing method that applies to a trained network. We define the decoder decomposition for AEs in Section 5.1 and for MD-AEs in Section 5.2.

5.1. The decoder decomposition for standard autoencoders

The data modes Φ^Y (the matrix of POD modes of $\boldsymbol{Y}$, see Table 1) form a basis of the subspace in which the output $\widehat{\boldsymbol{Y}}_{:,t}$ is represented as
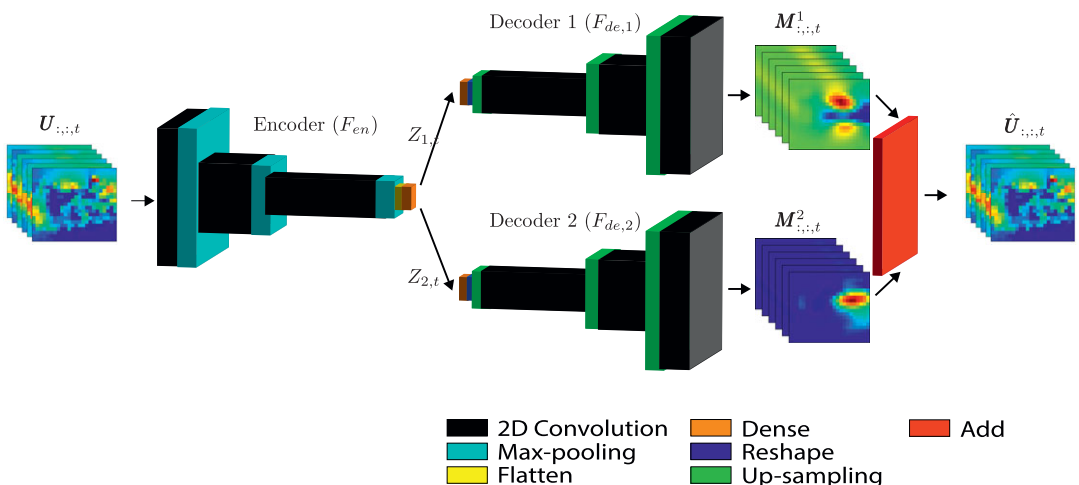

Figure 9. *Schematic of the MD-AE (Murata et al., 2020) with two latent variables as an example. Each latent variable is decoded by a decoder to produce a decomposed field. The sum of the decomposed fields* $\mathbf{M}^1$ *and* $\mathbf{M}^2$ *is the output of the MD-AE.*

$$\widehat{\boldsymbol{Y}}_{:,t} = \sum_{k=1}^{N^*} B_{k,t}\Phi^Y_{:,k} = \Phi^Y \boldsymbol{B}_{:,t}, \tag{5.1}$$

with $\boldsymbol{B} \in \mathbb{R}^{N^* \times N_t}$ being the temporal coefficients (decoder coefficients). The decoder coefficient for the data mode k at time step t is $B_{k,t}$. The matrix form of Equation (5.1) is

$$\widehat{\boldsymbol{Y}} = \Phi^Y \boldsymbol{B}^T. \tag{5.2}$$

In other words, the reconstructed flow $\widehat{\boldsymbol{Y}}$ is expressed as a linear combination of the POD modes of the original training data $\boldsymbol{Y}$ with $\boldsymbol{B}$ being the matrix of coefficients. In a trained network, $\boldsymbol{B}$ depends only $\widehat{\boldsymbol{Y}}$, which depends only on the latent variables $\boldsymbol{Z}$. Thus, the gradient of the output of the decoder with respect to the latent variables is

$$\frac{d\widehat{\boldsymbol{Y}}}{d\boldsymbol{Z}} = \Phi^Y \frac{d\boldsymbol{B}}{d\boldsymbol{Z}}. \tag{5.3}$$

The gradient of the decoder coefficients is the sensitivity to changes in the latent variables. Each element of the gradient matrix $\left(\frac{d\widehat{Y}}{d\boldsymbol{Z}}\right)_{j,i,t}$ is the rate of change of the output at grid point j and time step t $\left(\widehat{Y}_{j,t}\right)$ with respect to the latent variable $Z_{i,t}$. Because a single decoder coefficient corresponds to a single data mode, which represents a coherent structure; thus, the gradient of the decoder coefficients also reflects the sensitivity of the coherent structure to changes in the latent variables. Physically, the sensitivity quantifies the relative importance of the different latent variables for the decoder coefficients and the data modes they represent. We quantify the relative importance of the decoder coefficient $\boldsymbol{B}_{j,:}$ due to the latent variable $\boldsymbol{Z}_{i,:}$ by defining the average rate of change

$$\epsilon_{i,j} = \frac{\int \left|\frac{\partial \boldsymbol{B}_{j,:}}{\partial \boldsymbol{Z}_{i,:}}\right| d\boldsymbol{Z}_{1,:}\dots d\boldsymbol{Z}_{N_z}}{\int d\boldsymbol{Z}_{1,:}\dots d\boldsymbol{Z}_{N_z}}. \tag{5.4}$$

The average rate of change of a decoder coefficient due to a latent variable i quantifies its contribution to the data mode j in the output associated with that decoder coefficient (Eq. 5.1). The larger the $\epsilon_{i,j}$, the more important the latent variable i is in representing the data mode j in the output. This can be used to rank and select latent variables, as demonstrated in Section 7.2. Algorithm 1 shows how the average rate of change is computed.

Algorithm 1: Computing the average rate of change.

Input: F_{de}–trained decoder.
        Δz–constant step size.
Output: ϵ–average rate of change.

for $i \leftarrow 1,\dots,N_z$ **do.**

$\quad \left| \quad \frac{d\widehat{Y}}{d\boldsymbol{Z}_{i,:}} \leftarrow \frac{1}{(\boldsymbol{Z}_{i,:})} \frac{F_{de}(\boldsymbol{Z}_{i,:}+\Delta z) - F_{de}(\boldsymbol{Z}_{i,:}-\Delta z)}{2\Delta z}; \right.$ // approximate derivatives & normalize

end

Collect $\frac{d\widehat{Y}}{d\boldsymbol{Z}}$;

$\frac{d\boldsymbol{B}}{d\boldsymbol{Z}} \leftarrow \Phi^{YT} \frac{d\widehat{Y}}{d\boldsymbol{Z}}$;

$\epsilon^T \leftarrow \frac{1}{N_t} \sum_{t=1}^{N_t} \left|\frac{d\boldsymbol{B}}{d\boldsymbol{Z}}\right|_{:,:,t}$;   // average rate of change

return ϵ.

5.2. Decoder decomposition for MD-AEs

In an MD-AE, the decomposed fields M^i is decomposed into

$$M^i = \Phi^Y B^{iT}, \tag{5.5}$$

where B^i is the matrix of the ith decoder coefficients of the ith decomposed field. Combined with Equation (4.5), the output of the MD-AE becomes

$$\widehat{Y} = \Phi^Y \sum_{i=1}^{N_z} B^{iT}. \tag{5.6}$$

Equations (5.3) and (5.4) still apply as they are.

6. Decomposition of the unsteady laminar wake

In this section, we analyze the latent space of AEs trained with the laminar cylinder wake dataset (Section 3.1). We first decompose the dataset with an MD-AE and apply the decoder decomposition in Section 6.1, which serves as a benchmark to compare against the literature. We then apply the decoder decomposition to interpret the latent variables of AEs in Section 6.2.

6.1. The latent space of a mode-decomposing autoencoder

We decompose the unsteady wake dataset, described in Section 3.1, with an MD-AE with $N_z = 2$ (Figure 10) and compare our results with (Murata et al., 2020) to verify the decoder decomposition. The reconstruction loss, measured with MSE (Eq. (4.2)), is 2.6×10^{-5}, showing that the output $\widehat{U}$ is an accurate approximation of U. Figure 10a shows that the latent variables have the same periodic behavior as the data time coefficients, matching the results of Murata et al. (2020). By applying POD to the decomposed fields and by inspection, the decomposed field 1 contains the data modes 1, 3, and 6, and the decomposed field 2 contains the data modes 2, 4, and 5 (Figure 10c), similarly to what was observed by Murata et al. (Murata et al., 2020). This observation is based on visual inspection of the POD modes of the decomposed fields (Figure 10c) and the data modes (Figure 3), which is a qualitative comparison. Therefore, the POD modes of the decomposed fields may have different physical interpretations to the data modes they resemble. A quantitative comparison between the decomposed fields and the data modes is performed later in this section by applying the decoder decomposition. Figure 10b shows that the two latent variables have the same frequency spectrum, with peaks at $St = 0.23$ and 0.69, corresponding to the vortex shedding frequency and its second harmonic. Because the decomposed field 1 contains a POD mode similar to data mode 1 and the decomposed field 2 contains a similar mode to data mode 2, and the two latent variables have an identical frequency spectrum, are periodic and out of phase, we conclude that the two latent variables form a pair similar to the data time coefficients. However, the latent variables have two frequency peaks instead of a singular peak of the first two data time coefficients, showing that the latent variables contain nonlinear temporal information.

We apply the decoder decomposition to the MD-AE and plot the first four decoder coefficients for the two decomposed fields in Figure 11. None of the first four decoder coefficients are constant in Figure 11, meaning that the first four data modes all contribute toward both decomposed fields. The relative frequencies of the latent variables and the decoder coefficients can also be deduced from Figure 11. As the latent variables change from -1 to 1, decoder coefficients 1 and 2 in both decomposed fields complete a quarter of a sine wave, but the decoder coefficients 3 and 4 complete half a sine wave, meaning that decoder coefficients 3 and 4 oscillate at double the frequency of decoder coefficients 1 and 2.

For a more quantitative measure of how the data modes are split into the decomposed fields of the MD-AE, we calculate the equivalent energy (Kneer et al., 2021) for each decomposed field separately. The matrix of equivalent energy for the ith decomposed field and the data modes of the laminar flow, $\widehat{\Lambda}^i$, is given by
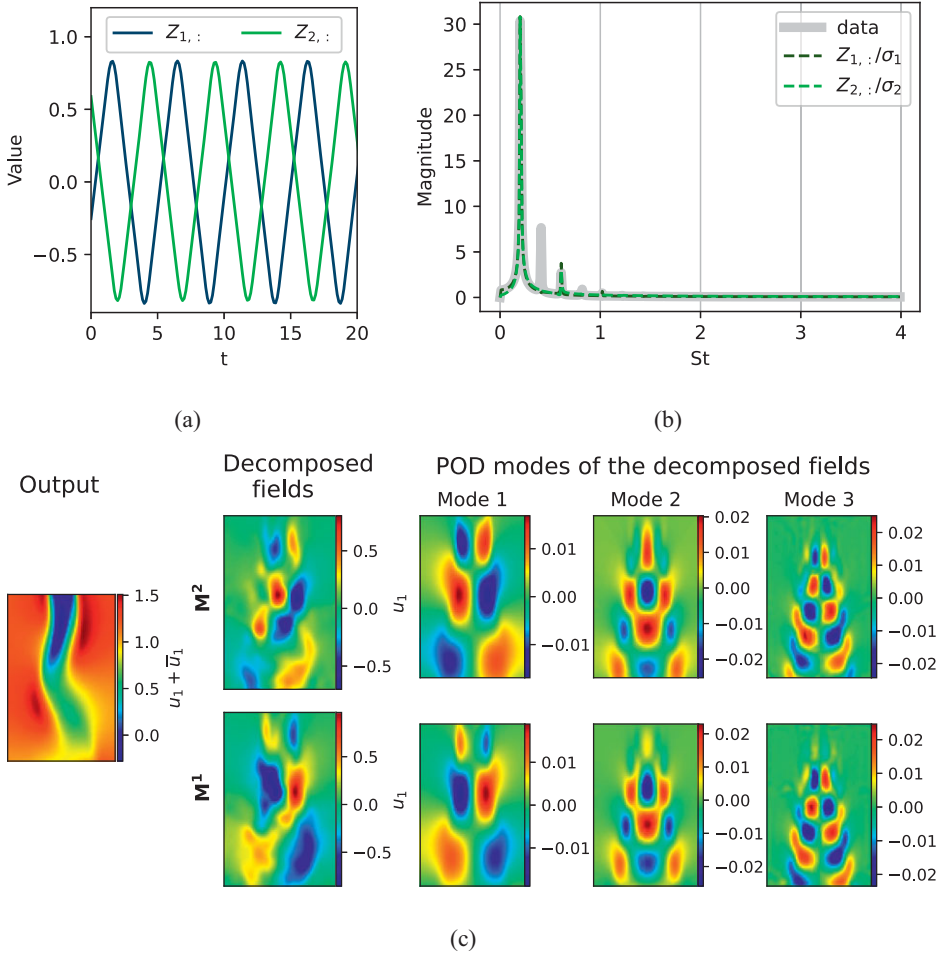
Figure 10. *Results of training the MD-AE on the laminar wake dataset with a latent dimension of 2. (a) Both latent variables are periodic in time. (b) Frequency spectrum of the latent variables, normalized by their standard deviation, compared with the frequency of the data. The latent variables both contain the vortex shedding frequency and the second harmonic of the vortex shedding frequencies. (c) A snapshot output of the MD-AE (mean flow added) and the POD modes of the decomposed fields. The POD modes of decomposed field 1 are similar to data modes 1, 3, and 6 and the POD modes of decomposed field 2 are similar to data modes 2, 4, and 5.*

$$\widehat{\Lambda}^i = \frac{1}{N_t - 1} B^i B^{iT}, \tag{6.1}$$

which is interpreted as the variance of the ith decomposed field, M^i, projected onto the data modes. The equivalent energy of the ith decomposed field and the kth data mode is $\widehat{\Lambda}^i_{kk}$. As M^i approaches the first data POD decomposed field ($\Phi^{lam}_{:,1} A^T_{i,:}$), $\widehat{\Lambda}^i_{11}$ approaches Λ_{11} while $\widehat{\Lambda}^i_{kk}$, where $k \neq 1$, approaches 0.

Figure 12 shows the equivalent energy of the first six data modes in the two decomposed fields and in the predicted output $\widehat{U}$. The equivalent energies of the predicted output for all data modes are approximately 100%, meaning that the output $\widehat{U}$ is an accurate approximation of U. Figure 12 also shows that all latent variables jointly contribute toward the presence of the data modes in the output, which also shows that the MD-AE does not fully separate data modes into different decomposed fields.
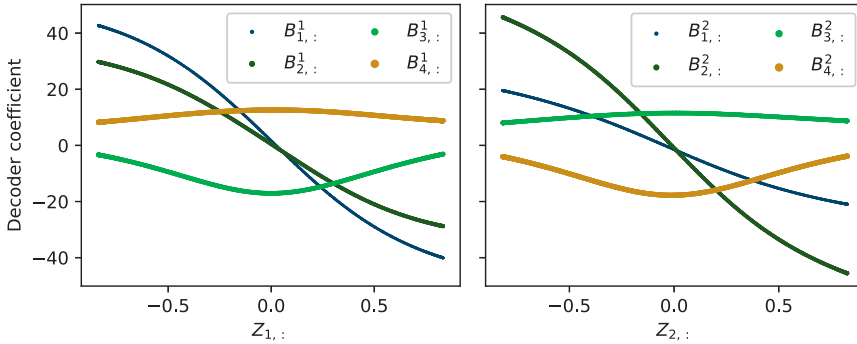
Figure 11. *Results from the MD-AE trained with the laminar wake dataset with two latent variables. The first four decoder coefficients plotted against the latent variables. Both latent variables affect the magnitude of the first four POD modes of the data in the output of the MD-AE.*
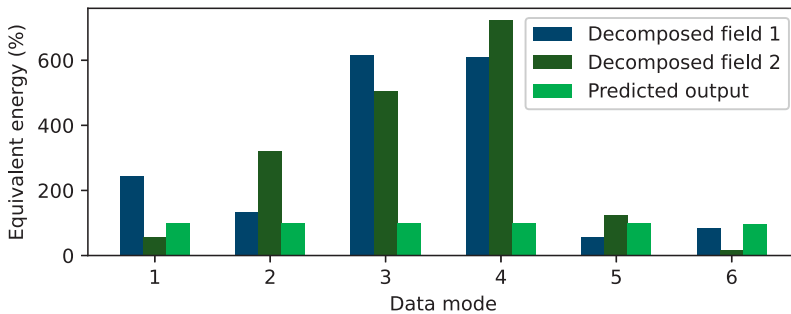

Figure 12. *Results from the MD-AE trained with the laminar wake dataset with two latent variables. The equivalent energy of the two decomposed fields of the MD-AE, showing the first four POD modes of the data. All POD modes of data are present in both decomposed fields.*

In summary, the decoder decomposition applied to an MD-AE (Murata et al., 2020) leads to a detailed interpretation of the decomposed fields. By computing the equivalent energy, we also show that, although the MD-AE works well for visualization, the decomposed fields are entangled.

6.2. *The latent space of standard autoencoders*

Unlike the MD-AE, an AE has no decomposed fields to visualize. We apply the decoder decomposition to AEs trained with the unsteady wake dataset to connect the data time coefficients to the latent variables. Since the frequency content of data modes of the laminar, periodic wake are well known (Zdravkovich, 1997), we use the interpretation of the data modes and time coefficients to provide an interpretation of the latent variables.

Figure 13 shows the loss of AEs trained with the unsteady wake dataset using 1, 2, 3, and 4 latent variables. The first plateau of loss occurs at $N_z = 2$ before the loss decreases again at $N_z = 4$. Similar behavior is also observed by Csala et al. (2022). The AEs with two latent variables achieve a significantly smaller loss than AEs with a single latent variable but the difference in loss between using two and three latent variables is small. The laminar wake U has periodic underlying dynamics and thus can be described by a single variable in polar coordinates (the angle) or two variables in Cartesian coordinates. However, the AEs do not find the best representation when forced to use only one latent variable due to their inability to perform Cartesian to polar transformation. Polar to Cartesian relies on the sin function, which is a many-to-one function and can be learned by neural networks either through approximation with any activation function or by using *sin* as the activation function (Wong et al., 2024). However, the Cartesian
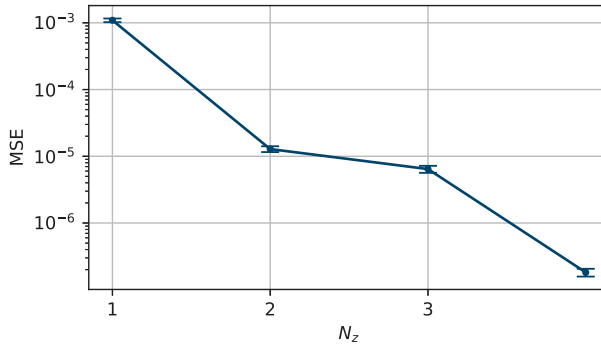
Figure 13. *The MSE of AEs trained with the unsteady wake dataset with different numbers of latent variables averaged over five repeats each. The error bars represent the standard deviation of the repeats. At $N_z = 2$, the MSE is approximately 1.3×10^5.*

to polar transformation relies on arcsin, which is not defined for all inputs. The Cartesian to polar transformation is a known difficulty in machine learning, with benchmark problems such as the two-spiral problem built around it (Fahlman and Lebiere, 1989; Liu et al., 2018), this is consistent with our observations (Figure 13). Therefore, we analyze an AE with $N_z = 2$ with decoder decomposition to obtain the physical interpretation in the current section, even though $N_z = 1$ should be the theoretical minimum number of latent variables to represent the laminar wake.

The phase plot of the two latent variables is a unit circle (Figure 14a), which indicates that the latent variables of an AE are periodic; the same phase plot also describes the first two data time coefficients. The same periodic behavior also shows in the time trace of the latent variables (Figure 14b). Figure 14c compares the frequency spectrum of the latent variables and the data. We find that both latent variables oscillate at the vortex shedding frequency, which is the lowest frequency in the dataset (Figure 2, right panel). The higher harmonic frequencies are not included in the latent variables because they are functions of the vortex shedding frequency, and the decoder can represent the spatial patterns of multiple data modes. The AE has produced latent variables that have a singular peak in the frequency spectrum, less than the frequency spectrum of the latent variables of the MD-AE, without sacrificing accuracy, showing that AEs are more expressive given the same number of latent variables than MD-AEs.

Figure 15 shows the sensitivity of the decoder coefficients, $\boldsymbol{B}_{1,:}, \ldots, \boldsymbol{B}_{4,:}$ to the two latent variables. The contours are the values of the decoder coefficients, shown for all values of the latent variables allowed by the nonlinear activation function. The dynamics of the dataset are shown as the gray circle, which shows the values of the latent variables that are observed during training.

The decoder coefficients $\boldsymbol{B}_{3,:}$ and $\boldsymbol{B}_{4,:}$ have double the frequency of $\boldsymbol{B}_{1,:}$ and $\boldsymbol{B}_{2,:}$, consistent with the characteristics of the first four data modes. At $\boldsymbol{Z} = \boldsymbol{0}$, all decoder coefficients have the value 0 because the network has no bias. Figure 15 provides a qualitative overview of the sensitivity of the data modes to the latent variables.

Because the decoder output $\widehat{\boldsymbol{U}}$ depends on multiple latent variables, the equivalent energy (Eq. (6.1)) cannot be used to identify the contribution of each latent variable. We use the average rate of change (Eq. (5.4)) to quantify the contribution of individual latent variables of the AE (Figure 16). Among the first four data modes, latent variable $\boldsymbol{Z}_{1,:}$ contributes most to the presence of the second data mode $\Phi_{:,2}^{lam}$ in the output $\widehat{\boldsymbol{U}}$, while $\boldsymbol{Z}_{2,:}$ mainly contributes toward the presence of the other three data modes. The first six data modes depend on both latent variables, so no latent variable can be removed without affecting the dynamics of the output. However, the average rate of change suggests that the smaller-scale energy-containing structures between the vortex streets (Zdravkovich, 1997), namely the pair comprised of data modes 3 and 4, are mainly represented by $\boldsymbol{Z}_{2,:}$. Therefore, $\boldsymbol{Z}_{2,:}$ should be used if the representation of data modes 3 and 4 in the output is required.
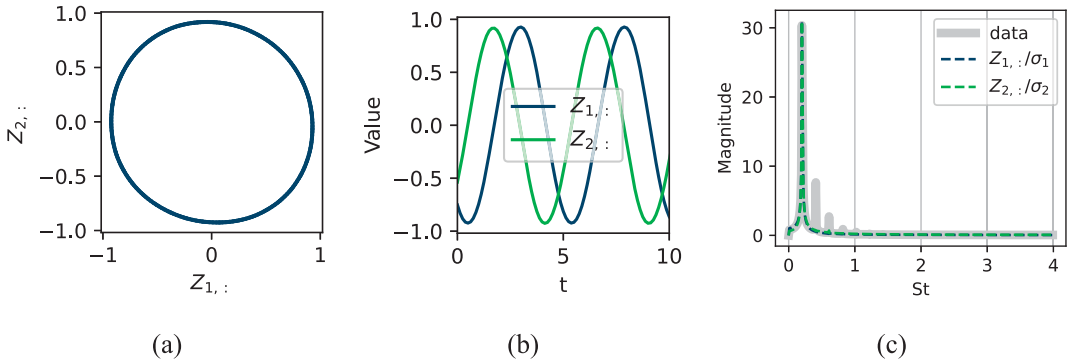
Figure 14. *Results from the AE trained with the laminar wake dataset with two latent variables. (a) Phase plot of two latent variables. The unit circle indicates harmonic oscillations. (b) Time trace of the two latent variables. The latent variables are periodic and $90°$ out of phase, behaving the same as the first two data time coefficients. (c) The frequency spectrum of the latent variables, compared to the frequency in the dataset. The variables are normalized by their standard deviation, σ, before applying the Fourier transform for visualization.*
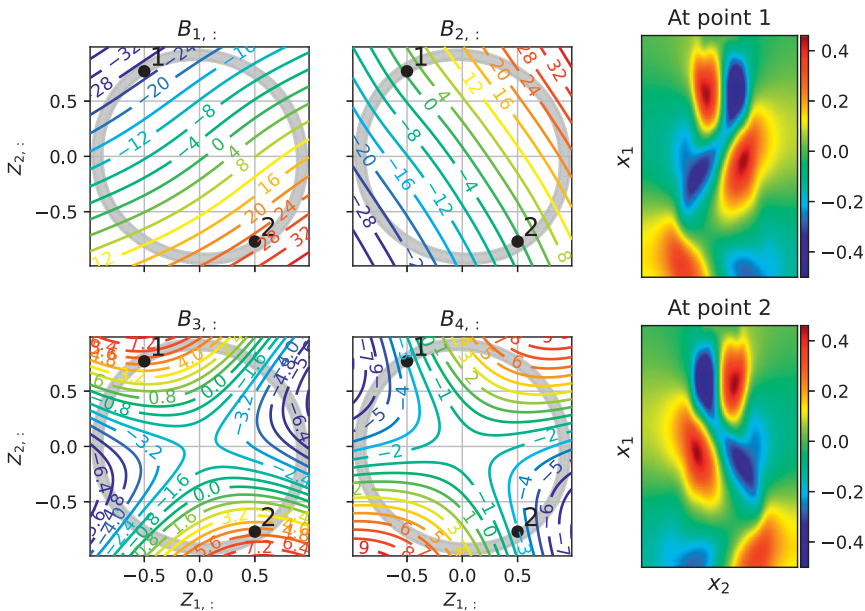

Figure 15. *Contour plots of the first four decoder coefficients of an AE with two latent variables, trained with the laminar wake dataset. The contours show the values of the decoder coefficients. The gray circle labels the values of the latent variables observed during training, which shows the dynamics of the dataset. The images in the last column show the output of the AE at points 1 and 2 labeled on the contour plots.*

By comparing the frequency spectrum of the latent variables of the AE and the MD-AE (Section 6.1), both with $N_z = 2$, we find that the AE has learned a more accurate representation of the data than the MD-AE because the latent variables have the minimum number of frequency peaks needed to represent the data. The frequency of the two latent variables coincides with the lowest frequency in the frequency spectrum of the data, meaning that the data modes with higher frequencies are added by the decoder. By
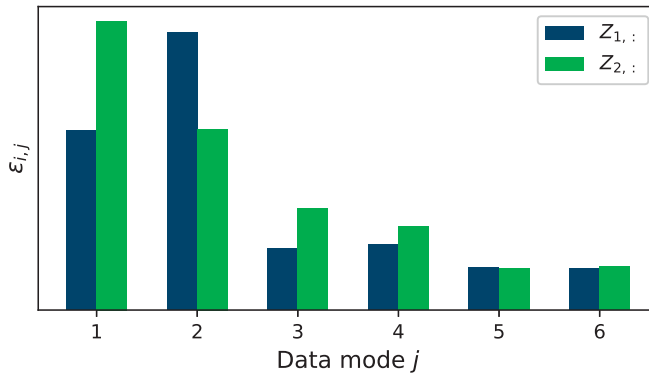
Figure 16. *Results from the AE trained with the laminar wake dataset with two latent variables. The figure shows the average rate of change of the decoder coefficients due to the latent variable i.*

introducing the average rate of change, we show that both latent variables are equally important in representing the vortex shedding, but the first harmonic is contained mainly in one of the latent variables in this case. Because the training of an AE depends on random seeds, the physical meaning of the latent variables may change for different cases. During training, we noticed that the dimension of latent space has a significant impact on the physical interpretation of the latent variables.[2] In-depth comments with example AEs trained with $N_z = 1$ and $N_z = 3$ are found in Appendix C.

7. Autoencoder decomposition of the turbulent wake from wind-tunnel experiments

We analyze the AEs trained to decompose the wind-tunnel wake dataset P (Section 3.2). We show that, for a turbulent dataset, the size of the network becomes an additional limiting factor for the interpretability of AEs in Section 7.1. We demonstrate how to select latent variables to single out a particular flow structure in Section 7.2.

7.1. The limiting factor is network size for turbulence

At least two latent variables are needed to represent flows with periodicity (Section 6.2). Because the turbulent wake of the axisymmetric body contains periodic behaviors such as vortex shedding (Brackston, 2017), we first train an AE with two latent variables on the wind-tunnel dataset (feedforward, see Table 1). Figure 17 (top & bottom left panels) shows the RMS pressure and the frequency content of the predicted pressure. The frequency peaks of the reconstructed pressure field match the data, but the RMS pressure and the magnitude of the PSD differ from the reference pressure field. We investigate the effect of increasing the latent variables from two to three. The prediction made by the AE with $N_z = 3$ is more accurate in both the RMS pressure and frequency content (Figure 18, top & bottom left panels) compared to the prediction by the AE with $N_z = 2$. By increasing the number of the latent variables from two to three, the prominent frequencies of the prediction remain unchanged, but the magnitude is closer to the reference data. By increasing the dimension of the latent space from $N_z = 2$ to $N_z = 3$, the prediction has improved without changing any frequency-related behavior. This suggests that the benefit of increasing the latent dimension comes from increasing the size of the decoder, thus improving the decoder's ability to express spatial patterns.

To further understand the cause of the improved prediction, we train a large decoder-only network (Figure 19) with the latent variables obtained with the previous $N_z = 2$ AE. (The details of the large

---

[2] The results presented in Section 6.2 are from AEs trained with the 'tanh', which are consistent. Training an AE with $N_z = 2$ using "ReLU" occasionally lead to results similar to those shown in Appendix C.
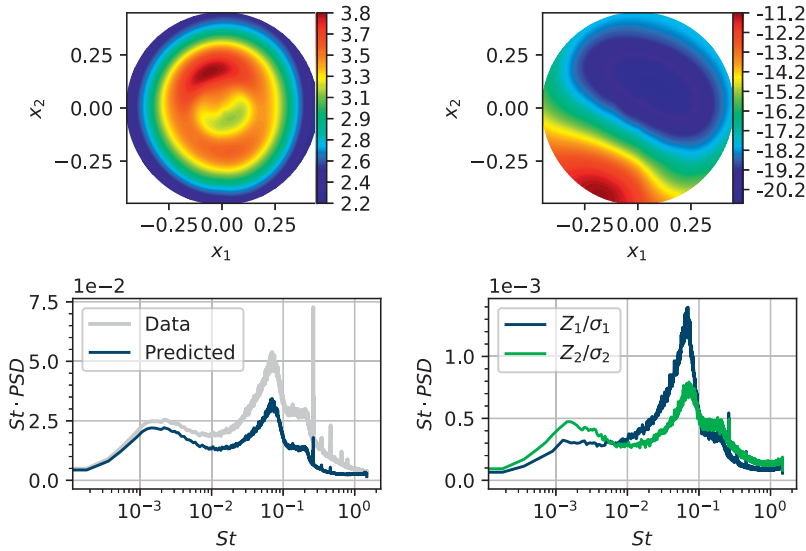
Figure 17. *The AE trained with the wind-tunnel dataset with two latent variables. Top left: Predicted RMS pressure. Top right: Predicted instantaneous pressure. Bottom left: The premultiplied overall PSD of the prediction. Bottom right: The premultiplied PSD of the latent variables of the AE trained with two latent variables, normalized by their standard deviation.*
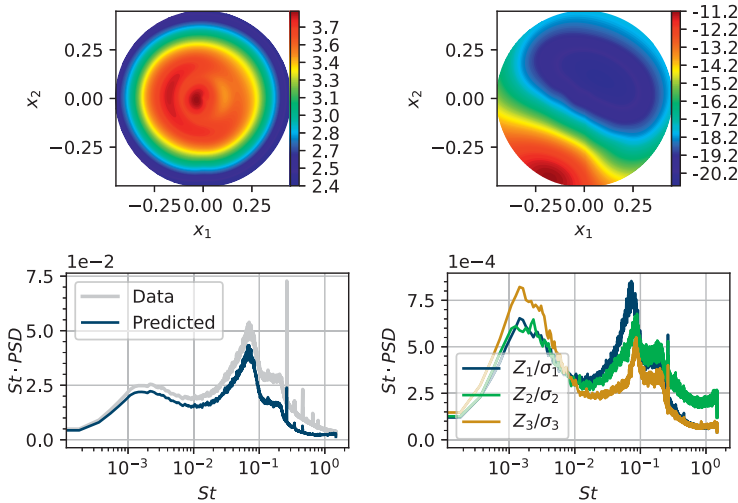

Figure 18. *Same as Figure 17 with three latent variables.*

decoder are given in Table A.6.) The large decoder has approximately twice the number of trainable parameters than the decoder in the previous $N_z = 2$ AE. The input to the large decoder is the latent variables obtained from the trained $N_z = 2$ AE, the frequency contents of which are shown in Figure 17 (bottom right panel); the output of the large decoder approximates the reference fluctuating pressure P.

The PSD of the large decoder-only network overlaps with the PSD of the $N_z = 3$ smaller-sized AE (Figure 20 right panel), both are closer to the PSD of the reference data than the smaller $N_z = 2$ AE. When the latent variables remain unchanged, the large decoder-only $N_z = 2$ network can make predictions that match the predictions made by a smaller dimension $N_z = 3$ AE. By comparing the results of the smaller size $N_z = 2$ and $N_z = 3$ AEs to the large decoder-only network with identical latent variable to the $N_z = 2$
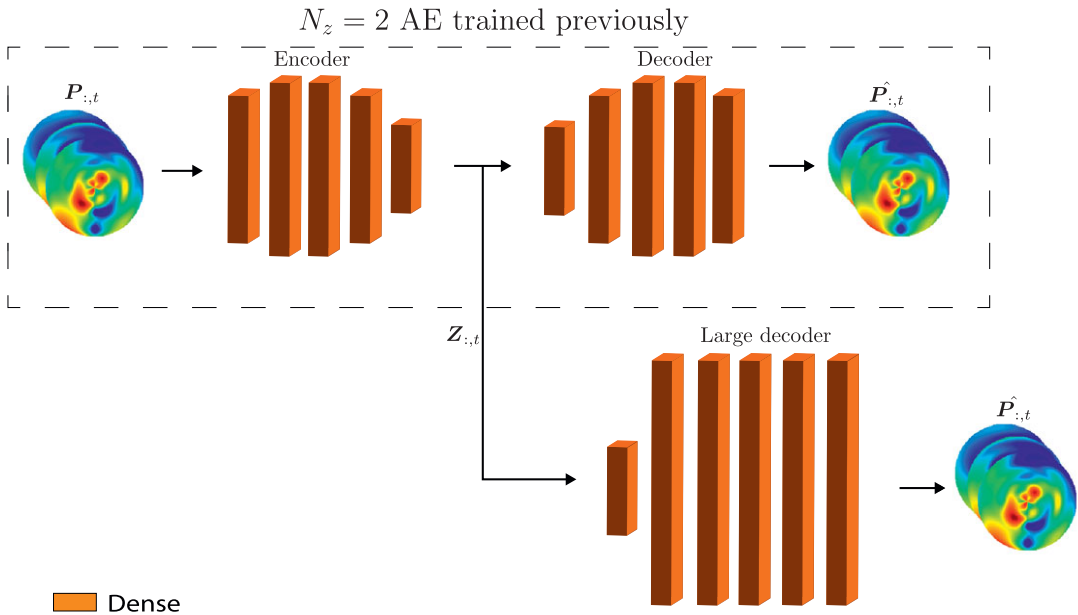
Figure 19. *The schematics and training process of the decoder-only network. The decoder has 99% more trainable parameters than the decoder in the $N_z = 2$ AE discussed in Section 7.1.*
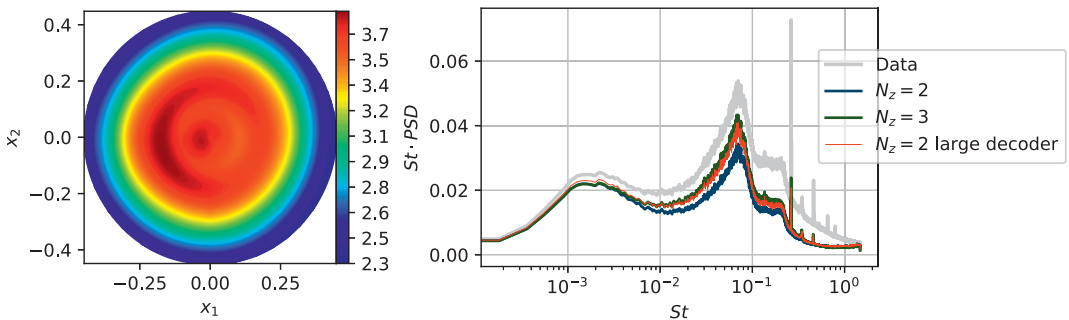

Figure 20. *Left: Predicted RMS pressure of the decoder-only network. Right: The premultiplied PSD of the predicted pressure from the AE with two and three latent variables, and from the large decoder-only network, trained with the wind-tunnel dataset.*

AE, we find that using a larger decoder has the same effect as increasing the number of latent variables. More importantly, the frequency peaks in the latent variables and predictions remain the same for all three networks. Therefore, we conclude that the critical factor for an accurate prediction using the wind-tunnel dataset, which contains multiple spatiotemporal scales, is the size of the decoder. In an AE, the weights are responsible for representing the spatial patterns in the flow, and the latent variables represent the time-dependent behaviors. We found that the optimal dimension of the latent space is a function of both of the underlying manifold's dimension (Magri and Doan, 2022; Doan et al., 2023; Racca et al., 2023) and the size of the decoder.

7.2. Filtering unwanted latent variables

Section 7.1 shows that changing both the dimension of the latent space, N_z, and the size of the AEs have a similar effect for the turbulent datasets. In this section, we apply the proposed decoder decomposition to
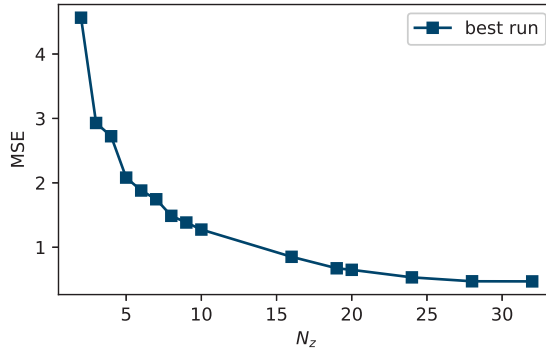
Figure 21. *AE with different numbers of latent variables and the same hyperparameters, trained with the wind-tunnel dataset. The loss stops decreasing at $N_z \approx 28$. Figure 22 shows the results from the AE with 28 latent variables. The RMS predicted pressure is in agreement with the reference RMS pressure, and the prediction has PSD that approximates the data's PSD in terms of both magnitude and frequency content. To understand which data modes are present in the prediction, we define the equivalent energy for an AE with a POD weight matrix, $\mathbf{W}_p$ (Eq. 2.1), as:*
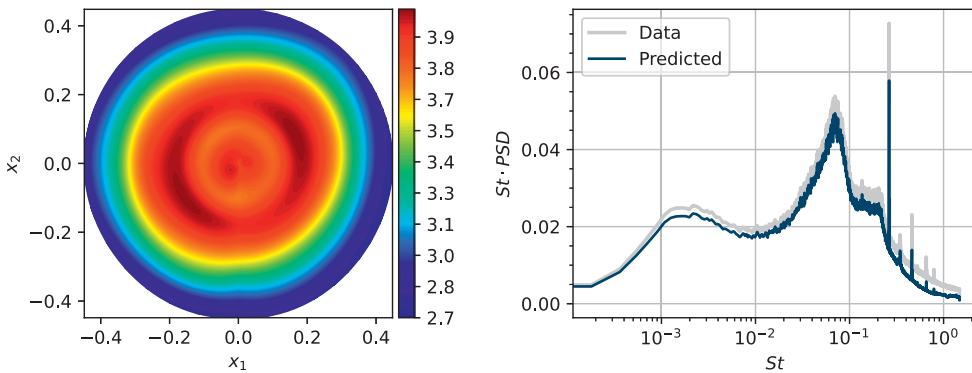

Figure 22. *RMS predicted pressure and premultiplied PSD of the prediction of the AE trained with the wind-tunnel dataset with 28 latent variables. The network attains an MSE equivalent to the reconstruction with 30 POD modes.*

filter out unwanted latent variables. When training on the turbulent pressure dataset, $\boldsymbol{P}$, we find that the MSE is sufficiently small at $N_z \approx 28$ (Figure 21). Therefore, we employ the AE with 28 latent variables. We filter the latent variables to enhance or reduce the importance of certain data modes among other data modes in the output of the AEs, similar to selecting data modes for reconstructing only a particular coherent structure. Figure 22 shows the results from the AE with 28 latent variables. The RMS predicted pressure is in agreement with the reference RMS pressure, and the prediction has PSD that approximates the data's PSD in terms of both magnitude and frequency content. To understand which data modes are present in the prediction, we define the equivalent energy for an AE with a POD weight matrix, W_p (Eq. 2.1), as

$$\widehat{\Lambda} = \frac{1}{N_t - 1} \Phi^{YT} \left(\left(\boldsymbol{W}_p \Phi^Y \boldsymbol{B}^T \right) \left(\boldsymbol{B} \Phi^{YT} \boldsymbol{W}_p^T \right) \right) \Phi^Y \tag{7.1}$$

from Equation (6.1) (Kneer et al., 2021). The matrix $\Phi^Y = \Phi^{exp}$ contains the data modes. Figure 23a compares the equivalent energy of the predicted pressure from the AE with $N_z = 28$ with the POD eigenvalues of the reference data. The prediction contains a similar amount of energy to the reference data for the more energetic coherent structures identified by POD. We focus our analysis on data modes 1 and
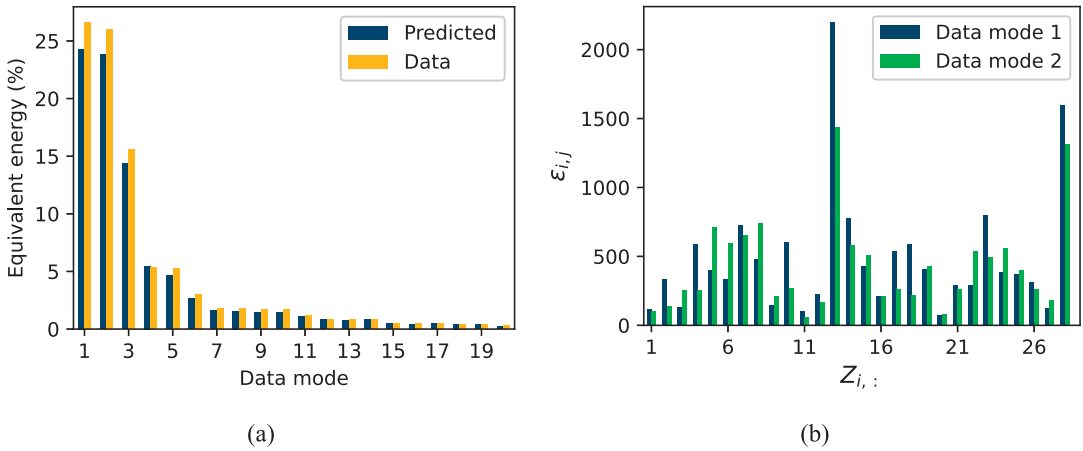
Figure 23. *The AE trained with the wind-tunnel dataset with 28 latent variables. (a) Equivalent energy of the predicted pressure compared to the POD eigenvalues of the reference data. (b) The average rate of change of decoder coefficients 1 and 2 due to the latent variables, normalized by the standard deviation of the latent variables. The decoder coefficients 1 and 2 are direct analogies of the data POD time coefficients 1 and 2, which are associated with vortex shedding.*

2, which represent vortex shedding. Figure 23b shows how each latent variable affects the decoder coefficients 1 and 2 in the prediction, which are associated with data modes 1 and 2 in the prediction. We compute the average rate of change $\epsilon_{i,j}$ (Equation (5.4)) and focus the analysis on data modes 1 and 2, which represent vortex shedding. Figure 23b shows how each latent variable affects the decoder coefficients 1 and 2 in the prediction, which are associated with POD modes 1 and 2 in the prediction. For both data modes, the largest contribution comes from the latent variable 13, followed by the latent variable 28. The numbering changes with the random seeds and the gradient update algorithms, but the conclusions do not. For the output of the AE to represent mainly the data modes 1 and 2, we set all latent variables (except for $\boldsymbol{Z}_{13,:}$ and $\boldsymbol{Z}_{28,:}$) to $\boldsymbol{0}$. This yields the filtered latent space $\boldsymbol{Z}_f$ such that

$$\widehat{\boldsymbol{P}}_f = F_{de}(\boldsymbol{Z}_f), \tag{7.2}$$

which contains mostly data modes 1 and 2. By filtering, we identify the contribution that the latent variables make to the data modes in the output, and manipulate some latent variables based on that contribution. Technically, by setting the latent variables to $\boldsymbol{0}$, we eliminate the fluctuation of the data modes in the output of the AE. Since the AE is nonlinear, setting all but latent variables $\boldsymbol{Z}_{13,:}$ and $\boldsymbol{Z}_{28,:}$ to $\boldsymbol{0}$ is not equivalent to truncating the data modes from mode 2. Thus, the filtering process aims to minimize the fluctuation of the data modes $\Phi_{:,3}^{exp}$ to $\Phi_{:,N_z}^{exp}$ in the output. The results of decoding the filtered latent variables are shown in Figure 24. The RMS of the filtered prediction shows strong fluctuations only in the outer region. The PSD of the filtered prediction (Figure 25) shows that $\widehat{\boldsymbol{P}}_f$ has a large amount of energy at $St \approx 0.2$ and $St \approx 0.002$. The frequency $St \approx 0.06$ does not appear in the PSD of the filtered prediction, meaning that by keeping only the two most contributing latent variables for data modes 1 and 2, the AE no longer models the pulsation of the vortex core. This means that the filtering has been successful.

The equivalent energy in Figure 26a further shows that the filtered prediction represents only the data modes 1 and 2, which is the objective that we set. Figure 26b shows the two leading POD modes of the filtered prediction. The modes are approximately antisymmetric with frequency peaks at $St \approx 0.2$ and $St \approx 0.002$, which represent vortex shedding. The structures in Figure 26b are not perfectly antisymmetric due to information being lost by removing 26 out of 28 latent variables. The two leading modes contain over 99% of the energy in the filtered prediction, showing that the filtering has significantly
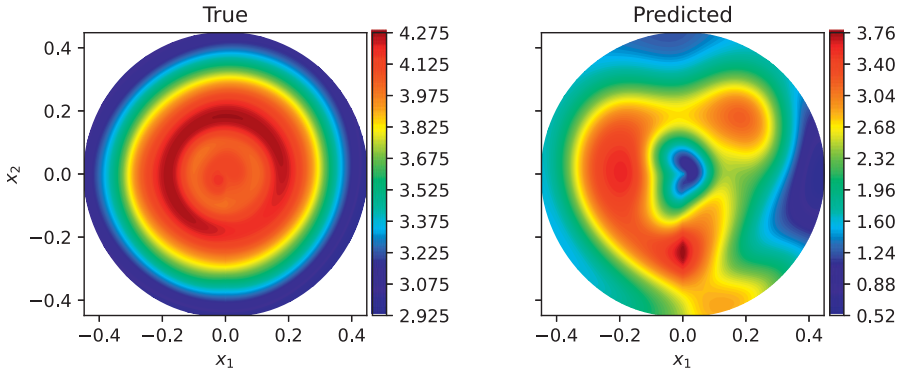
Figure 24. *Prediction obtained with the AE with 28 latent variables from the filtered latent variables* $\mathbf{Z}_f$ *and the reference data.*
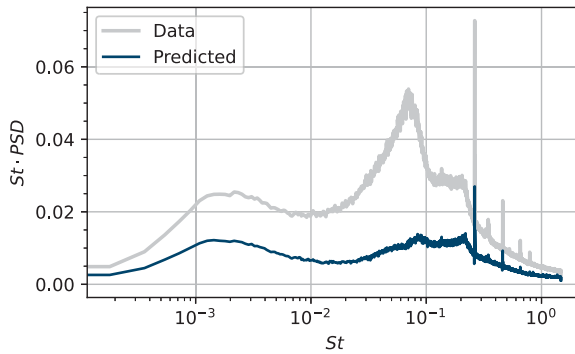
Figure 25. *Premultiplied PSD of the prediction obtained with the AE with 28 latent variables from the filtered latent variables. All peaks (except for* $St \approx 0.2$*) are filtered.*
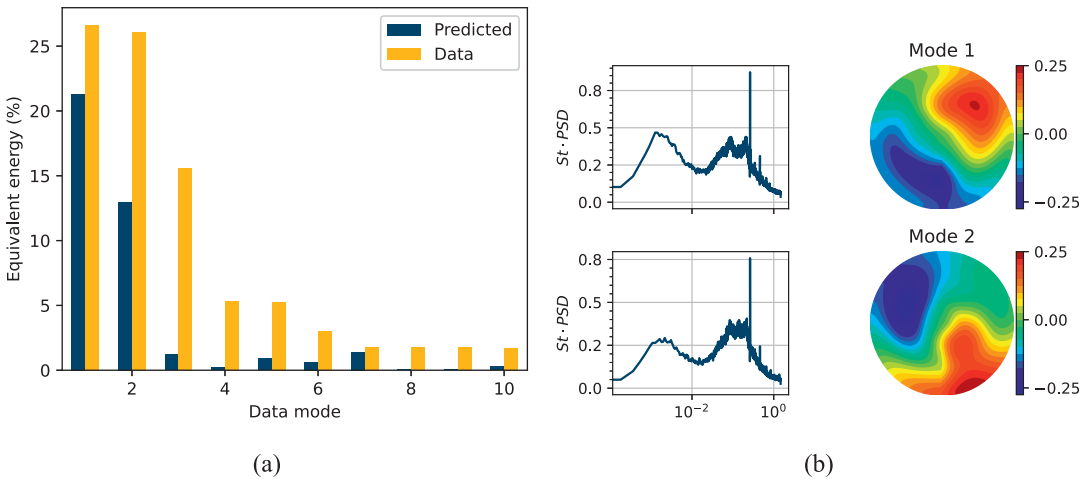
Figure 26. *POD on the prediction obtained with the AE with 28 latent variables from the filtered latent variables. (a) Equivalent energy of the filtered prediction* $\widehat{\mathbf{P}}_f$ *compared to the POD eigenvalues of the reference data. Among the five most energetic POD modes, the filtered prediction shows a large amount of energy only in data modes 1 and 2, which is the goal we set. (b) POD modes 1 and 2 of* $\widehat{\mathbf{P}}_f$*, which contain over 99% of the flow energy of* $\widehat{\mathbf{P}}_f$*. These two modes show the same frequency peak at* $St \approx 0.2$ *and have antisymmetric spatial structures. These two modes represent vortex shedding.*

compressed the spatial information. Combining the energy, the structures, and frequency content, we conclude that the only coherent structures in the filtered prediction are modes associated with vortex shedding. By selecting latent variables based on contribution to the decoder coefficients, we have successfully filtered the coherent structures of the output of the AE. Thus, we have shown decoder decomposition as a viable method for selecting latent variables based on coherent structures of the flow for an AE with a large latent dimension. This is useful for filtering spurious or unwanted latent variables, as well as singling out specific modes of interest.

8. Conclusions

We propose the decoder decomposition to help practitioners design and interpret the latent space of nonlinear AEs. The decoder decomposition is a post-processing method which ranks the latent variables of an AE. Two types of AEs are considered: the standard AEs with one decoder and one encoder, and the MD-AEs with one decoder per latent variable. First, we apply the decoder decomposition to analyze the latent space of the two-dimensional unsteady laminar cylinder wake. Both AEs and MD-AEs are applied to the laminar wake and their results are compared. By analyzing AEs with different latent dimensions, we show that the dimension of the latent space significantly impacts the physical meanings of the latent variables even if the reconstruction errors remain unaffected. Second, we apply the decoder decomposition to AEs trained with the wind-tunnel experimental data of a three-dimensional turbulent wake past a bluff body. We show that increasing the size of the decoder has a similar effect to increasing the number of latent variables when the number of latent variables is small. Third, we apply the decoder decomposition to rank and select the latent variables that are associated with the vortex shedding structures. We apply the average rate of change to rank the latent variables based on their contribution to the data modes 1 and 2, which correspond to vortex shedding. Finally, we filter the latent space to minimize the effect of the unwanted data modes in the AE output. The output contains mainly the two coherent structures associated with vortex shedding, which verifies the method. The decoder decomposition is a simple yet robust post-processing method for ranking and selecting latent variables based on the flow physics and coherent structures of interest. In the future, the decoder decomposition will be employed to select the number of latent variables to model the flow physics of interest.

References

Abadi M, **Agarwal A**, **Barham P**, **Brevdo E**, **Chen Z**, **Citro C**, **Corrado GS**, **Davis A**, **Dean J**, **Devin M**, **Ghemawat S**, **Goodfellow I**, **Harp A**, **Irving G**, **Isard M**, **Jia Y**, **Jozefowicz R**, **Kaiser L**, **Kudlur M**, **Levenberg J**, **Mané D**, **Monga R**, **Moore S**, **Murray D**, **Olah C**, **Schuster M**, **Shlens J**, **Steiner B**, **Sutskever I**, **Talwar K**, **Tucker P**, **Vanhoucke V**, **Vasudevan V**, **Viégas F**, **Vinyals O**, **Warden P**, **Wattenberg M**, **Wicke M**, **Yu Y**, **Zheng X** (2015) TensorFlow: large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Albukrek CM, **Urban K**, **Rempferr D**, **Lumley JL** (2002) Divergence-free wavelet analysis of turbulent flows. *Journal of Scientific Computing* 17(1–4), 49–66.

Alfonsi G., **Primavera L.** (2007). The structure of turbulent boundary layers in the wall region of plane channel flow. *Proceedings of the Royal Society A: Mathematical, Physical, and Engineering Sciences*, *463*(2078), 593–612.

Baldi P, **Hornik K** (1989) Neural networks and principal component analysis: Learning from examples without local minima. *Neural Networks 2*, 53–58.

Bansal N, **Chen X**, **Wang Z** (2018) Can we gain more from orthogonality regularizations in training deep networks? In Bengio S, Wallach H, Larochelle H, Grauman K, Cesa-Bianchi N and Garnett R (eds.), *Advances in Neural Information Processing Systems*, Vol. *31*, pp. 4261–4271. New York: Curran Associates, Inc.

Bartholomew P, **Laizet S**, **Schuch FN**, **Frantz RAS**, **Rolfo S**, **Hamzehloo A**, **CFLAG, Deskos GY**, **Schaefer K**, **Jing H and Monteiro LR** (2022) Xcompact3d/Incompact3d:. Zenodo.

Csala H, **Dawson STM**, **Arzani A** (2022) Comparing different nonlinear dimensionality reduction techniques for data-driven unsteady fluid flow modeling. *Physics of Fluids 34*(11), 117119.

Doan NAK, **Polifke W**, **Magri L** (2021) Auto-encoded reservoir computing for turbulence learning. In *Computational Science – ICCS 2021*, pp. 344–351, Cham: Springer International Publishing.

Doan NAK, **Racca A**, **Magri L** (2023) Convolutional autoencoder for the spatiotemporal latent representation of turbulence. In Mikyška J, de Mulatier C, Paszynski M, Krzhizhanovskaya VV, Dongarra JJ, Sloot PM, eds., *Computational Science – ICCS 2023*, Lecture Notes in Computer Science, pp. 328–335, Cham: Springer Nature Switzerland.

Eivazi H, **Le Clainche S**, **Hoyas S**, **Vinuesa R** (2022) Towards extraction of orthogonal and parsimonious non-linear modes from turbulent flows. *Expert Systems with Applications 202*, 117038.

Fahlman S, **Lebiere C** (1989) The cascade-correlation learning architecture. In *Advances in Neural Information Processing Systems*, vol. *2*. Massachusetts, US: Morgan-Kaufmann.

Fukami K, **Hasegawa K**, **Nakamura T**, **Morimoto M**, **Fukagata K** (2021) Model order reduction with neural networks: application to laminar and turbulent flows. *SN Computer Science 2*(6), 467.

Fukami K, **Nakamura T**, **Fukagata K** (2020) Convolutional neural network based hierarchical autoencoder for nonlinear mode decomposition of fluid field data. *Physics of Fluids 32*(9), 095110.

Fukami K, **Nakao H**, **Taira K** (2024). Data-driven transient lift attenuation for extreme vortex gust-airfoil interactions. arXiv: 2403.00263. https://doi.org/10.48550/arXiv.2403.00263

Fukami K, **Taira K** (2023) Grasping extreme aerodynamics on a low-dimensional manifold. *Nature Communications 14*(1), 6480.

Ganga Prasath S, **Sudharsan M**, **Vinodh Kumar V**, **Diw akar S**, **Sundararajan T**, **Tiwari S** (2014) Effects of aspect ratio and orientation on the wake characteristi cs of low Reynolds number flow over a triangular prism. *Journal of Fluids and Structures 46*, 59–76.

Kevlahan NKR, **Hunt JCR**, **Vassilicos JC** (1994) A comparison of different analytical techniques for identifying structures in turbulence. *Applied Scientific Research 53*(3–4), 339–355.

Kneer S, **Sayadi T**, **Sipp D**, **Schmid P**, **Rigas G** (2021) Symmetry-aware autoencoders: s-pca and s-nlpca. 10.48550/arXiv.2111.02893.

Laizet S, **Lamballais E** (2009) High-order compact schemes for incompressible flows: A simple and efficient method with quasi-spectral accuracy. *Journal of Computational Physics 228*(16), 5989–6015.

Liu R, **Lehman J**, **Molino P**, **Petroski Such F**, **Frank E**, **Sergeev A**, **Yosinski J** (2018) An intriguing failing of convolutional neural networks and the CoordConv solution. In *Advances in Neural Information Processing Systems*, vol. *31*. New York, US: Curran Associates, Inc.

Loiseau J-C, **Brunton SL**, **Noack BR** (2020) *From the POD-Galerkin method to sparse manifold models*, pp. 279–320. *Model Order Reduction*: Volume *3* Applications. Berlin, Boston: De Gruyter.

Loshchilov I, **Hutter F** (2017) Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations*. https://openreview.net/forum?id=Skq89Scxx.

Lumley JL (1970) Stochastic tools in turbulence. In *Stochastic Tools in Turbulence, Applied Mathematics and Mechanics*, vol. 12. Academic Press, New York.

Zdravkovich MM (1997) *Flow Around Circular Cylinders Vol. 1: Fundamentals*, vol. 1. Oxford Univerisity Press, Oxford, 1 edition.

Magri L, **Doan NAK** (2022) On interpretability and proper latent decomposition of autoencoders. *Center for Turbulence Research Proceedings of the Summer Program 2022.*

Murata T, **Fukami K**, **Fukagata K** (2020) Nonlinear mode decomposition with convolutional neural networks for fluid dynamics. *Journal of Fluid Mechanics 882*, A13.

Nakamura T, **Fukami K**, **Hasegawa K**, **Nabae Y**, **Fukagata K** (2021) Convolutional neural network and long short-term memory based reduced order surrogate for minimal turbulent channel flow. *Physics of Fluids 33*(2), 025116.

Noack BR, **Morzynski M**, **Tadmor G**, **Tadmor G** (2011) *Reduced-Order Modelling for Flow Control* vol. *528*. Springer Wien, Vienna, Austria.

Racca A, **Doan NAK**, **Magri L** (2023) Predicting turbulent dynamics with the convolutional autoencoder echo state network. *Journal of Fluid Mechanics 975*, A2.

Rigas G (2021) Turbulent experimental database1. Internal Document.

Rigas G, **Oxlade AR**, **Morgans AS**, **Morrison JF** (2014) Low-dimensional dynamics of a turbulent axisymmetric wake. *Journal of Fluid Mechanics 755*, R5.

RD Brackston (2017). Feedback control of three-dimensional bluff body wakes for efficient drag reduction. PhD thesis, Imperial College London, London.

Rowley CW, **Dawson STM** (2017) Model reduction for flow analysis and control. *Annual Review of Fluid Mechanics 49*, 387–417.

Schmid PJ (2010) Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics 656*(August), 5–28.

Schmidt OT Colonius T (2020) Guide to spectral proper orthogonal decomposition. *AIAA Journal 58*(3), 1023–1033.

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

Taira K, **Brunton SL**, **Dawson STM**, **Rowley CW**, **Colonius T**, **McKeon BJ**, **Schmidt OT**, **Gordeyev S**, **Theofilis V**, **Ukeiley LS** (2017) Modal analysis of fluid flows: an overview. *AIAA Journal 55*(12), 4013–4041.

Tu JH, **Rowley CW**, **Luchtenburg DM**, **Brunton SL**, **Kutz JN** (2014) On dynamic mode decomposition: Theory and applications. *Journal of Computational Dynamics 1*(2), 391–421.

Vinuesa R, **Sirmacek B** (2021) Interpretable deep-learning models to help achieve the sustainable development goals. *Nature Machine Intelligence 3*(11), 926.

Weiss J (2019) A tutorial on the proper orthogonal decomposition. In *2019 AIAA Aviation Forum*, Dallas, TX: American Institute of Aeronautics and Astronautics.

Wong JC, **Ooi C**, **Gupta A**, **Ong Y-S** (2024) Learning in sinusoidal spaces with physics-informed neural networks. *IEEE Transactions on Artificial Intelligence 5*(3), 985–1000. https://doi.org/10.1109/TAI.2022.3192362.

Xie D, **Xiong J**, **Pu S** (2017) All you need is beyond a good init: exploring better solution for training extremely deep convolutional neural networks with orthonormality and modulation. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, pp. 5075–5084, doi: 10.1109/CVPR.2017.539.

Appendices

A. Autoencoder layers

The convolutional autoencoders are trained with the laminar wake dataset. All convolutional AEs and MD-AEs are built with encoders described in Table A.1 and decoders described in Table A.2. The encoders are identical except for the latent dimension N_z. For all convolutional AEs, the decoders are identical except for N_z. For all convolutional MD-AEs, the input to the decoder has shape (1), specified in Table A.2. When training on the wind-tunnel dataset, AEs with feedforward layers are used (Tables A.3 and A.4). The large decoder-only network (Table A.6) is only used in Section 7.1. The hyperparameters used by all networks can be found in Table A.5.

Table A.1. *The encoder used in all convolutional AEs and MD-AEs*

Layers	Output shape	Notes
input	(200, 129, 1)	
conv2d	(200, 129, 8)	3-by-3 filter
maxpooling	(100, 65, 8)	
conv2d	(100, 65, 16)	3-by–3 filter
maxpooling	(50, 33, 16)	
conv2d	(50, 33, 32)	3-by-3 filter
maxpooling	(25, 17, 32)	
flatten	(13600)	
dense	(N_z)	Output shape is (1) if MD-AE

Table A.2. *The decoder used in all convolutional AEs and MD-AEs*

Layers	Output shape	Notes
input	(N_z)	Shape is (1) if MD-AE
dense	(13600)	
reshape	(25, 17, 32)	
upsampling	(50, 33, 32)	Bilinear interpolation
conv2d	(50, 33, 16)	3-by–3 filter
upsampling	(100, 65, 16)	Bilinear interpolation
conv2d	(100, 65, 8)	3-by–3 filter
upsampling	(200, 129, 8)	Bilinear interpolation
conv2d	(200, 129, 1)	3-by–3 filter

Table A.3. *The encoder used in feedforward AEs*

Layers	Output shape	Notes
input	(64)	
dense	(128)	
batch_normalisation	(128)	
dropout	(128)	
dense	(256)	
batch_normalisation	(256)	
dropout	(256)	
dense	(256)	
batch_normalisation	(256)	
dropout	(256)	
dense	(128)	
batch_normalisation	(128)	
dropout	(128)	
dense	(64)	
batch_normalisation	(64)	
dropout	(64)	
dense	(N_z)	

Table A.4. *The decoder used in feedforward AEs*

Layers	Output shape	Notes
input	(N_z)	
dense	(64)	
batch_normalisation	(64)	
dropout	(64)	
dense	(128)	
batch_normalisation	(128)	
dropout	(128)	
dense	(256)	
batch_normalisation	(256)	

Continued

Table A.4. *Continued*

Layers	Output shape	Notes
dropout	(256)	
dense	(256)	
batch_normalisation	(256)	
dropout	(256)	
dense	(128)	
batch_normalisation	(128)	
dropout	(128)	
dense	(64)	

Table A.5. *The hyperparameters used in all networks*

Which network (table/figure number(s) of the network(s) using these hyperparameters)	MD-AEs and AEs for the unsteady laminar wake (Table A.1, A.2)	AEs used to search for suitable latent dimension (Figure 21)	AEs for the wind-tunnel turbulent wake (Tables A.3, A.4, and A.6)
Learning rate	0.001	0.004	0.0022
Learning rate schedule	n/a	CosineDecayRestarts	CosineDecayRestarts
Activation function	tanh	tanh	tanh
Regularization	0.0	0.0	0.00003
Dropout rate	0.0%	0.0%	1.4%

CosineDecayRestarts is a built-in tensorflow learning rate schedule (Abadi et al., 2015; Loshchilov and Hutter, 2017)

Table A.6. *Large decoder used in Section 7.1*

Layers	Output shape	Notes
input	(2)	The large decoder is only used with $N_z = 2$
dense	(64)	
batch_normalisation	(64)	
dropout	(64)	
dense	(256)	
batch_normalisation	(256)	
dropout	(256)	
dense	(256)	
batch_normalisation	(256)	
dropout	(256)	
dense	(256)	
batch_normalisation	(256)	
dropout	(256)	
dense	(256)	
batch_normalisation	(256)	
dropout	(256)	
dense	(64)	

B. POD and linear AE

Performing POD is equivalent to solving the quadratic optimization problem (Fukami et al., 2021)

$$\Phi = \arg\min_{\Phi^*} \left\| \boldsymbol{Q} - \Phi^* \Phi^{*T} \boldsymbol{Q} \right\|_2^2. \tag{B.1}$$

Let us consider a linear AE in which the number of latent variables is equal to the number of grid points in the input, that is, $N_z = N$. A linear AE is an AE with linear activation functions. If $N_z < N$, the linear AE recovers only the N_z most energetic POD modes in an L_2-norm sense. For a linear AE without biases, we express the training as

$$\omega^* = \arg\min_{\omega} \left\| \boldsymbol{Q} - \widehat{W}\mathbf{W}\mathbf{Q} \right\|_2^2, \tag{B.2}$$

which has the solution

$$\widehat{W}W = \mathbf{I}, \tag{B.3}$$

where W and $\widehat{W}$ are the weights of the encoder and the decoder of the linear AE. These matrices are not necessarily orthogonal. To obtain the POD solution, we seek orthogonal $\widehat{W}$ and W. Baldi and Hornik (1989) showed that for a linear AE without biases, the only solution is the POD solution, whereas all the remaining critical points are saddle points. However, there are saddle points in the optimization of (B.2), which may affect convergence. We apply L_2-regularization to the weights for the linear AE to converge to the POD solution, which is equivalent to minimizing the Frobenius norm of weights, denoted $\|W\|_F$. When the autoencoder is linear, applying L_2-regularization of weights is known as applying a "soft constraint of orthogonality" (Bansal et al., 2018; Xie et al., 2017). Training the linear AE with L_2-regularization is solving

$$\omega^* = \arg\min_{\omega} \left\| \boldsymbol{Q} - \widehat{\boldsymbol{Q}} \right\|_2^2 + \gamma \left(\|W\|_F^2 + \|\widehat{W}\|_F^2 \right), \tag{B.4}$$

where $\gamma \geq 0$ is the regularization factor. The only solution to the optimization problem (B.2) which minimizes (B.4) is

$$WW^T = \boldsymbol{I} \tag{B.5}$$

where $\mathbf{I}$ is the identity matrix. Therefore, $\widehat{W} = W^T$, that is, the decoder and encoder matrices are orthogonal. A similar analysis can be applied to a linear MD-AE to show that each decomposed field is a POD mode.

C. Physical interpretation is more difficult when unsuitable latent dimension is used

In Section 6.2, we show that the MSE of the AEs converges for $N_z = 2$, and that at least two latent variables are needed to describe the periodic behavior. However, it is difficult to know the theoretical number of latent variables to use in common machine learning tasks. The common approach in machine learning is to treat the latent dimension as a hyperparameter of the networks. By using the laminar wake as an example, we discuss how the choice of the latent variables affects the physical interpretation of the latent variables. First, we show the results from an AE with $N_z = 1$ in Figure 27. The latent variable is periodic in time, but the time trace of
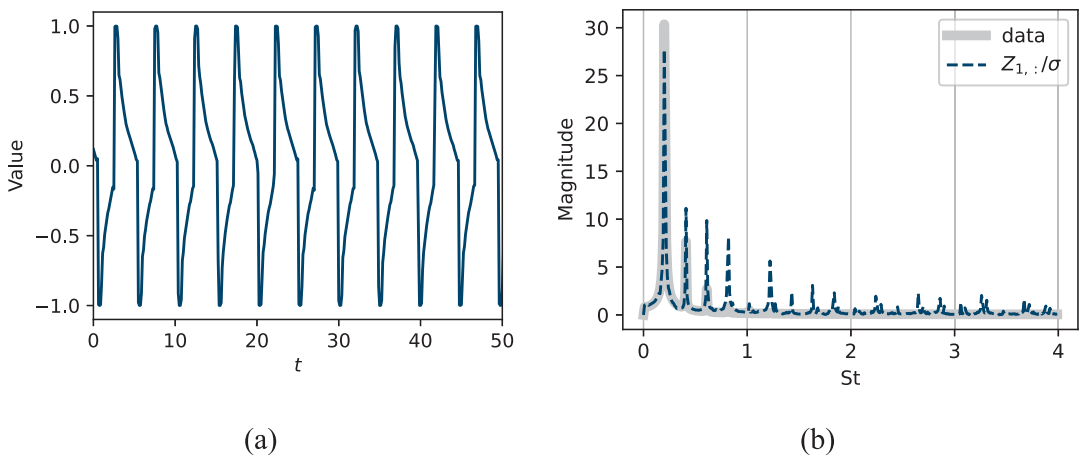

(a)   (b)

Figure 27. *Results from the AE trained with the laminar wake dataset with one latent variable. (a) The latent variable approximates the discontinuity caused by the angle moving from 2π to 0. (b) The frequency spectrum of the latent variable, normalized by its standard deviation, compared with the data frequencies. The latent variables contain frequencies that do not exist in the dataset.*
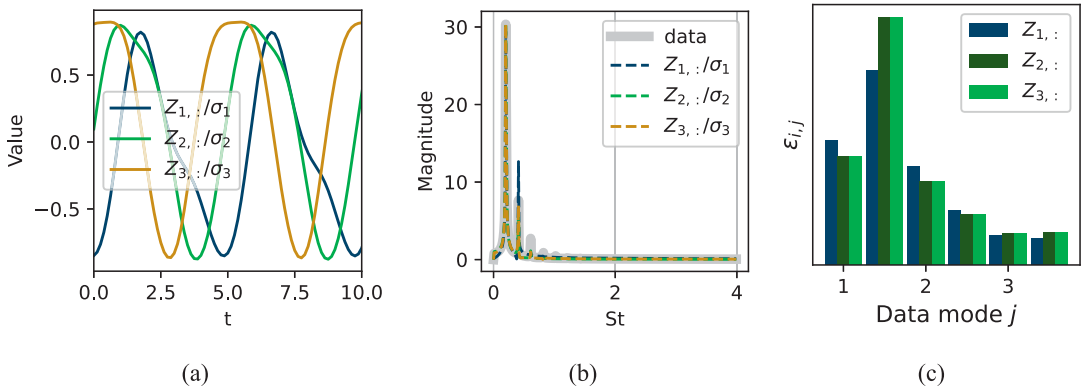
Figure 28. *Results from the AE trained with the laminar wake dataset with three latent variables. (a) The time trace of the latent variables. (b) The frequency spectrum of the latent variables, normalized by their standard deviations. The latent variables contain the vortex shedding and the first harmonic frequency. (c) The average rate of change of decoder coefficients due to the latent variables of an AE with three latent variables. Latent variables $\mathbf{Z}_{2,:}$ and $\mathbf{Z}_{3,:}$ have the same contributions toward the first six data modes, meaning that both latent variables carry the same information.*

the latent variable is not smooth (Figure 27a) and have frequency peaks at high frequencies not present in the data (Figure 27b). When a single variable (the angle) is used to represent the periodic behavior of the dataset U, there is a discontinuity when the angle moves from 2π to 0. The latent variable is forced to use higher frequencies to try to approximate the discontinuity. These higher frequencies are the results of numerical approximation and are unphysical. In an AE with $N_z = 3$, the latent variables contain the first harmonic in addition to the vortex shedding frequency (Figure 28b), which is present in the data but not in the first two data time coefficients (Figure 2, right panel) or the latent variables from an AE with two latent variables (Figure 14b). We discussed (Section 6.2) that the additional frequency peak is not necessary for the accurate reconstruction of the dataset.

The average rate of change shows that latent variables $\mathbf{Z}_{2,:}$ and $\mathbf{Z}_{3,:}$ contribute equally to the first six data modes (Figure 28c). Equal contribution to all data modes indicates that the two latent variables have the same role in representing the data, which means that the information carried by one of the latent variables is duplicate information contained in the other latent variables. The three latent variables are not suitable because they lead to redundant information. All three latent variables contribute similarly to the first six data modes, making the selection of latent variables more difficult. In the present case, the three latent variables form a group to represent the counter-rotating vortices, even though two latent variables should be, in principle, sufficient.

Both the AE with $N_z = 3$ and the MD-AE with $N_z = 2$ (Section 6.1) have latent variables with an additional frequency peak compared to the latent variables of the AE with $N_z = 2$ (Section 6.2), even though there is no substantial difference in the reconstruction error. Additionally, unphysical frequencies also arise from numerical approximations when using AE with $N_z = 1$. Therefore, a design consideration of AEs for nonlinear reduced-order modeling is to use a latent dimension that results in the least number of frequency peaks in the latent variables while maintaining the reconstruction accuracy of the AEs.

D. Nomenclature

Acronyms

AE	a standard autoencoder
exp	experimental wake
lam	laminar wake
MD-AE	a mode-decomposing autoencoder
MSE	mean square error
POD	proper orthogonal decomposition
PSD	power spectrum density

Matrix/tensors

$\widehat{\Lambda}$	the equivalent energy of the output of an AE.
$\widehat{\Lambda}^i$	the equivalent energy of the ith decomposed field of an MD-AE.

Λ	diagonal matrix of eigenvalues, sorted from largest to smallest
Φ	POD modes
A	time coefficients for POD modes
B	matrix of decoder coefficients
C	correlation matrix
I	identity matrix
M^i	the ith decomposed field of an MD-AE.
P	the experimental wake dataset consists of the pressure measurements of the experimental bluff body wake on a two-dimensional polar grid.
Q	snapshot matrix
U	the laminar wake dataset consists of the streamwise velocity of the simulated laminar wake on a two-dimensional grid.
W_P	weighting factor for the snapshot matrix for POD
Y	an example data matrix used for defining the autoencoders and related methods.
Z	matrix of latent variables
$\tilde{Q}$	reconstructed snapshot matrix $\mathbf{Q}$ using N_m POD modes.

Nondimensional group

Re	Reynolds number
St	Strouhal number

Symbols

$\widehat{\ast}$	dataset * reconstructed by an autoencoder
σ	standard deviation
ω	parameters of a network
u	velocities
F_{de}	function composition that represents the decoder
F_{en}	function composition that represents the encoder
N	the number of measured variables in the dataset
N_m	number of truncated POD modes to keep for reconstruction
N_t	number of time steps
N_u	number of velocity components
p	pressure
U_∞	inlet velocity
D	diameter of the cylinder/axisymmetric body
f	frequency
L	domain length
T	period of vortex shedding