1 Analysis, Modeling, and Control of the Cylinder Wake

B. R. Noack, A. Ehlert, C. N. Nayeri, and M. Morzynski

We give a tour de force through select powerful methods of machine learning for data analysis, dynamic modeling, model-based control, and model-free control. Focus is placed on a few Swiss army knife methods that have proven capable of solving a large variety of flow problems. Examples are proximity maps, manifold learning, proper orthogonal decomposition, clustering, dynamic modeling, and control theory methods as contrasted with machine learning control (MLC). In Chapters 14 and 17 of this book, the mentioned machine learning approaches are detailed for reduced-order modeling and for turbulence control. All methods are applied to a classical, innocent looking benchmark: the oscillatory two-dimensional incompressible wake behind a circular cylinder at $Re = 100$ without and with actuation. This example has the advantage of being visually accessible to interpretation and foreshadows already key challenges and opportunities with machine learning.

1.1 Introduction

Machine learning, and more generally, artificial intelligence, is increasingly transforming fluid mechanics (Brunton et al. 2020). This change is based on several trends. First, the efforts from first principles to new theoretical insights have diminishing returns after hundreds of years of theoretical research. Second, multiphysics multiscale problems of engineering interest significantly increase in complexity. Third, fluid mechanics creates increasing amounts of high-quality data from experiments, for example, participle image velocimetry, to simulations. Finally, the methods of computer science become increasingly powerful with increasing performance of computers and with continual discoveries of new algorithms and their improvements.

In this chapter, we focus on machine learning methods for three classical fields of fluid mechanics: analysis of snapshot data, dynamic reduced-order modeling, and the control of a given configuration. Following the literature, we take the transient oscillatory cylinder wake as the most simple, yet nontrivial benchmark. The flow physics is phenomenologically easy to visualize and can be conceptualized as a nonlinear oscillator. Yet, an accurate description poses already challenges. Section 1.2 describes the employed configuration and data.

https://doi.org/10.1017/9781108896214.004 Published online by Cambridge University Press

Analysis may start with the search of a few features helping to categorize or explain fluid mechanics. In the case of the cylinder wake, the amplitude and phase are two such features helping to parameterize drag, lift, and even the flow with good accuracy. Section 1.3 presents two tools for this purpose. First, the *proximity map* cartographs all snapshot data in a two-dimensional plane with *classical multidimensional scaling* (CMDS) as the prominent approach. Second, an automated manifold extraction, *local linear embedding* (LLE), is presented. Both methods allow us to extract the amplitude and the phase of vortex shedding. CMDS can be applied to arbitrary even turbulent data. LLE comes with an estimate of the embedding dimension, if the dynamics is simple enough.

The analysis may continue with the search of a low-dimensional representation of the flow data – as described in Section 1.4. Two different approaches are presented. First, the flow data is represented by a data-driven Galerkin expansion with *proper orthogonal decomposition (POD)*. POD minimizes the averaged error of the expansion residual. Second, the flow data may also be represented by a small number of representative state, called *centroids*. K-means++ clustering achieves this goal by minimizing the averaged representation error between the snapshots and their closest centroids.

The dynamics may be understood by *reduced-order models* (ROM) building on such low-dimensional representations. The spectrum of ROM has a bewildering richness with a myriad of enabling auxiliary methods. Hence, an overview is postponed until Chapter 14. We focus on simple dynamical models of the cylinder wake, illustrating the analytical insights that may be gained (see Section 1.5).

The stabilization of the wake is discussed in Section 1.6. This discussion starts with a classical approach employing ROM for the derivation of the control law. A powerful model-free alternative is *MLC*, which learns the control laws in hundreds or thousands of test runs.

Finally, Section 1.7 summarizes some good practices of analysis, modeling, and control. The chapter cannot be an exhaustive state-of-the-art compendium of machine learning approaches. Instead, the presented methods can be seen as the Swiss army knife of machine learning. They are simple yet powerful and can already be applied in experimental projects with no or limited availability to first principle equations.

1.2 The Cylinder Wake: A Classical Benchmark

This section describes a classical, innocent looking benchmark of modeling and control, the two-dimensional oscillatory flow behind a circular cylinder. The first wake models were proposed over 100 years ago (von Kármán 1911), giving the von Kármán vortex street its name. The configuration and direct numerical simulation are described in Section 1.2.1. The transient flow behavior is outlined in Section 1.2.2. A sketch of the dynamics is previewed in Section 1.2.3.

https://doi.org/10.1017/9781108896214.004 Published online by Cambridge University Press

1.2.1 Configuration and Direct Numerical Simulation

The two-dimensional viscous, incompressible wake behind a circular cylinder is computed. This flow is characterized by the Reynolds number $Re = UD/\nu$ where D represents the cylinder diameter, U the oncoming velocity, and ν the kinematic viscosity of the fluid. The reference Reynolds number is set to $Re = 100$, which is significantly above the onset of vortex shedding at $Re = 47$ (Jackson 1987, Zebib 1987) and also far below the onset of three-dimensional instabilities around $Re = 160$ (Zhang et al. 1995, Barkley & Henderson 1996).

In the following, all quantities are assumed to be normalized with the cylinder diameter D, the oncoming velocity U, and the density of the fluid ρ. The two-dimensional cylinder wake is described by a Cartesian coordinate system (x, y) with the origin in the cylinder center, the x-axis pointing in streamwise, and the y-axis in the transverse direction. The incompressibility condition and Navier–Stokes equations read

$$\nabla \cdot \boldsymbol{u} = 0, \tag{1.1a}$$

$$\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} = -\nabla p + \frac{1}{Re} \triangle \boldsymbol{u}, \tag{1.1b}$$

where p represents the pressure, "∂_t" partial differentiation with respect to time, "∇" the Nabla operator, "$\triangle$" the Laplace operator, and "$\cdot$" an inner product or contraction in tensor algebra.

The rectangular computational domain Ω_{DNS} has a length and width of 50 and 20 diameters, respectively. The cylinder center has a distance of 10 diameter to the front and lateral sides. Summarizing,

$$\Omega_{\mathrm{DNS}} = \left\{ (x, y) \in \mathcal{R}^2 \colon x^2 + y^2 \geq 1/4 \ \cap \ -10 \leq x \leq 40 \ \cap \ |y| \leq 10 \right\}.$$

On the cylinder, the no-slip condition $\boldsymbol{u} = \boldsymbol{0}$ is enforced. At the front $x = -10$ and lateral sides of the domain $y = \pm 10$, a uniform oncoming flow $\boldsymbol{u}_\infty = (1, 0)$ is assumed. A vanishing stress condition is employed at the outflow boundary $x = 40$.

Simulations are performed with a finite-element method on an unstructured grid with implicit time integration. This solver is third-order accurate in time and second-order accurate in space. Details about the Navier–Stokes and stability solvers are described in Morzyński et al. (1999) and Noack et al. (2003). The triangular mesh consists of 59 112 elements.

The employed initial conditions are based on the unstable steady solution $\boldsymbol{u}_s$ and a small disturbance with the most unstable eigenmode $\boldsymbol{f}_1$. The steady solution is computed with a Newton gradient solver. The eigenmode computation is described in our earlier work (Noack et al. 2003). The disturbance is the real part of the product of the eigenmode and unit phase factor $e^{\mathrm{j}\phi}$. Here, "j" denotes the imaginary unit and ϕ the phase. The amplitude ε is chosen to create a perturbation with a fluctuation energy of 10^{-4}. The resulting initial condition reads

$$\boldsymbol{u}(\boldsymbol{x}, t = 0) = \boldsymbol{u}_s(\boldsymbol{x}) + \varepsilon \mathbb{R} \left\{ \boldsymbol{f}_1(\boldsymbol{x})\, e^{\mathrm{j}\phi} \right\}. \tag{1.2}$$

Sixteen initial conditions are considered. These correspond to equidistantly sampled phases $\phi \in [22.5°, 45°, \ldots, 337.5°, 360°]$. Integration is performed from $t = 0$

https://doi.org/10.1017/9781108896214.004 Published online by Cambridge University Press

to $t = 200$, capturing the complete transient and post-transient state. The time step is $\Delta t = 0.1$, corresponding to roughly one 50th of the period.

1.2.2 Unforced Transients

In this section, the transients from the steady solution to periodic vortex shedding are investigated. The flow is analyzed in the observation domain

$$\Omega := \{(x, y) \in \Omega_{\text{DNS}} : 5 \leq x \leq 15 \wedge 5 \leq y \leq 5\}\,. \tag{1.3}$$

This domain is about twice as long as the vortex bubble of the steady solution. The streamwise extent is large enough to resolve over one wavelength of the initial vortex shedding as characterized by the stability eigenmode. A larger domain is not desirable, because a small increase in wavenumber during the transient will give rise to large phase differences in the outflow region, complicating the comparison between flow states. The domain is consistent with earlier investigations by the authors (Gerhard et al. 2003, Noack et al. 2003) and similar to the domains of other studies (Deane et al. 1991).

The analysis is based on the inner product of the Hilbert space of square-integrable functions over the observation domain Ω. This inner product between two velocity fields $\boldsymbol{v}$ and $\boldsymbol{w}$ is defined by

$$(\boldsymbol{v}, \boldsymbol{w})_\Omega = \int_\Omega d\boldsymbol{x}\; \boldsymbol{v} \cdot \boldsymbol{w}, \tag{1.4}$$

where "$\cdot$" denotes the Euclidean inner product. The corresponding norm of the velocity field $\boldsymbol{v}$ reads

$$\|\boldsymbol{v}\|_\Omega = \sqrt{(\boldsymbol{v}, \boldsymbol{v})_\Omega}. \tag{1.5}$$

The flow $\boldsymbol{u}$ is decomposed into a slowly varying base flow $\boldsymbol{u}^D$ and an oscillatory fluctuation $\boldsymbol{u}'$,

$$\boldsymbol{u} = \boldsymbol{u}^D + \boldsymbol{u}'. \tag{1.6}$$

The short-term averaged flow is approximated as the projection of the flow on the one-dimensional affine space containing the steady solution $\boldsymbol{u}_s$ and the post-transient mean flow $\boldsymbol{u}_0$. The superscript "D" comes from the term *distorted mean flow* of mean-field theory (Stuart 1958). In other words,

$$\boldsymbol{u}^D(\boldsymbol{x}, t) = \boldsymbol{u}_s(\boldsymbol{x}) + a_\Delta(t)\, \boldsymbol{u}_\Delta(\boldsymbol{x}), \tag{1.7}$$

with the shift-mode $\boldsymbol{u}_\Delta = (\boldsymbol{u}_0 - \boldsymbol{u}_s)\,/\|\boldsymbol{u}_0 - \boldsymbol{u}_s\|_\Omega$ and amplitude $a_\Delta = (\boldsymbol{u} - \boldsymbol{u}_s, \boldsymbol{u}_\Delta)_\Omega$. This definition approximates a short-term averaged flow and generalizes the notion in the stability literature where the steady solution is identified with the base flow.

The shift-mode amplitude a_Δ characterizes the mean-flow distortion while the fluctuation energy

$$\mathcal{K} := \|\boldsymbol{u}'\|_\Omega^2\,/2 \tag{1.8}$$

https://doi.org/10.1017/9781108896214.004 Published online by Cambridge University Press

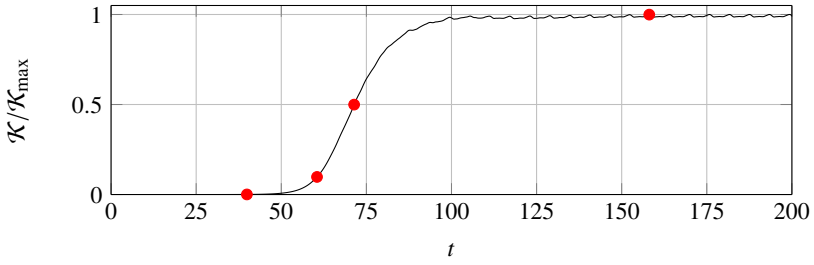

Figure 1.1 Evolution of the turbulent kinetic energy $\mathcal{K}$ with time t associated with an initial condition for $\phi = 22.5°$. The values are normalized with the maximum value $\mathcal{K}_{\max}$. Red points indicate normalized fluctuation levels of 0%, 10%, 50%, and 100%. For details, see Ehlert et al. (2020).

parameterizes the fluctuation level. We also refer to $\mathcal{K}$ as *turbulent kinetic energy* (TKE) following the mathematical definition of statistical fluid mechanics, realizing that the flow is laminar, not turbulent.

Figure 1.1 displays the TKE evolution with time. The maximum TKE value $\mathcal{K}_{\max}$ is used for normalization. Three dynamic phases can be distinguished. Within the first 30 convective time units D/U, the flow exhibits *linear dynamics* or exponential growth in the neighborhood of the steady solution. This exponential growth can clearly be seen in a logarithmic plot (Noack et al. 2003). In the second, *nonlinear transient phase* for $50 < t < 100$ the flow transitions from the steady solution to the limit cycle with decreasing growth rate. In the *post-transient phase* for $t > 150$, a periodic vortex shedding or, equivalently, limit-cycle dynamics is observed. The figure marks four times for TKE levels near 0%, 10% , 50%, and 100%, corresponding to the linear dynamics phase, the beginning and middle of the nonlinear transient phase and the limit cycle.

In Figure 1.2, the vorticity for the four selected time instants is shown. Positive (negative) values of vorticity are shown in red (blue) bounded by solid (dashed) lines. The three dynamic phases can be distinguished based on the closeness of vortex shedding to the cylinder and on the formation of pronounced individual vortices.

This discussion provides a basis for the time interval $[t_{\min}, t_{\max}]$ for snapshot selection. A lower bound $t_{\min} = 40$ is chosen. This bound guarantees a TKE below 0.01% or, equivalently 10^{-4} of the asymptotic maximum value. The upper bound $t_{\max} = 110$ includes few periods on the limit cycle.

1.2.3 Wake Dynamics

Figure 1.3 foreshadows a state-space picture of the transient dynamics. Near the steady solution (bottom of the paraboloid), the flow spirals outward on a plane spanned by the real and imaginary part of the unstable stability mode (bottom right). With increasing fluctuation amplitude, the Reynolds stress deforms the short-term averaged flow in the direction of the shift mode depicted middle left. This deformation leads

https://doi.org/10.1017/9781108896214.004 Published online by Cambridge University Press

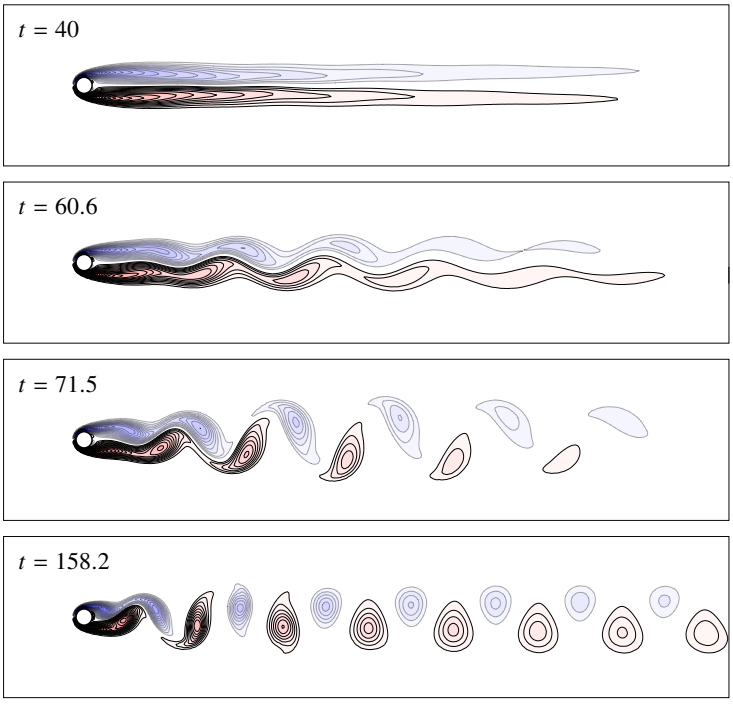

Figure 1.2 Vorticity snapshots corresponding to 0%, 10%, 50%, and 100% fluctuation level for the simulation displayed in Figure 1.1. The flow is visualized by iso-contours of vorticity with positive (negative) values marked by solid (dashed) lines and red (blue) background. The iso-contour levels and color scales are the same for all snapshots. For details, see Ehlert et al. (2020).

to a shorting of the vortex bubble and an upward motion of the fluctuation energy. The state moves outward and upward on a paraboloid until it converges against a limit cycle. The center of this limit cycle is characterized by the mean flow while the fluctuations are well approximated by the first two POD modes. In Chapters 14 and 17, this dynamics will be distilled from the data, dynamically modeled, and reversed by stabilizing control.

1.3 Cartographing the Data with Features

In this section, feature extraction is discussed. For the sake of concreteness, features are considered for an ensemble of M velocity snapshots $\boldsymbol{u}^m(\boldsymbol{x}) = \boldsymbol{u}(\boldsymbol{x}, t^m)$, $m = 1, \ldots, M$. For an oscillatory flow, amplitude and frequencies are important features that can completely or, in case of slow drifts, partially characterize the state. For a turbulent flow, features are far more challenging to design. In case of skin friction reduction of a turbulent boundary layer, features might be the position and amplitude of sweeps and ejections. In the following, two feature extraction methods are presented. First (Section 1.3.1), proximity maps via *classical multidimensional*

https://doi.org/10.1017/9781108896214.004 Published online by Cambridge University Press

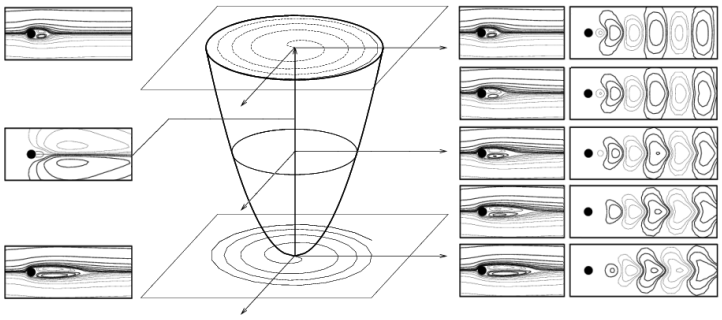

Figure 1.3 Principal sketch of the wake dynamics. The left side displays the mean flow (top), shift-mode (middle) and steady solution (bottom). The right side illustrates interpolated vortex streets on the mean-field paraboloid (middle column). The short-term averaged flows are depicted also as the streamline plots. Adapted from Morzyński et al. (2007).

scaling (CMDS) can be employed for any data. Second (Section 1.3.2), manifold extraction with *local linear embedding* (LLE) is particularly powerful for low-dimensional dynamics.

1.3.1 Proximity Maps with Multidimensional Scaling

The goal of a proximity map is to cartograph high-dimensional snapshots in a visually accessible, often two-dimensional, feature space such that neighborhood relations are preserved as much as possible. Let $\boldsymbol{\gamma}^m = \left[\gamma_1^m, \gamma_2^m\right]^{\mathrm{T}} \in \mathcal{R}^2$ (the superscript "T" denotes the transpose) with $m = 1, \ldots, M$ the two-dimensional feature vectors corresponding to the snapshots $\boldsymbol{u}^m, m = 1, \ldots, M$. In CMDS (Cox & Cox 2000), these features minimize the accumulative error of the distances between the snapshots

$$E = \sum_{m=1}^{M} \sum_{n=1}^{M} \left[\|\boldsymbol{u}^m - \boldsymbol{u}^n\| - \|\boldsymbol{\gamma}^m - \boldsymbol{\gamma}^n\| \right]^2 . \tag{1.9}$$

The translational degree of freedom is removed by requesting centered features, $\sum_{m=1}^{M} \boldsymbol{\gamma}^m = 0$. The rotational degree of freedom is fixed by requiring the first feature coordinate to have maximum variation. In general, for an N-dimensional feature space, the sum of first I variances is maximized for all $I \in \{1, \ldots, N\}$. For the invariance of the error under mirroring, however, there is no cure, as with the sign indeterminacy of POD modes and amplitudes. In fact, the resulting proximity map yields the first two POD amplitudes a_1, a_2. The resulting metric may be tailored to specific applications, for example, identifying regions with similar cost functions (Kaiser et al. 2017b). Since proximity maps are based on preserving neighborhood information, it is strongly related to LLE.

In this formulation, the features γ_1 and γ_2 coincide exactly with the POD mode amplitudes a_1 and a_2 discussed in Section 1.4.1. Hence, the proximity map is identical with the phase portrait of these POD mode coefficients. Numerous generalizations of

https://doi.org/10.1017/9781108896214.004 Published online by Cambridge University Press

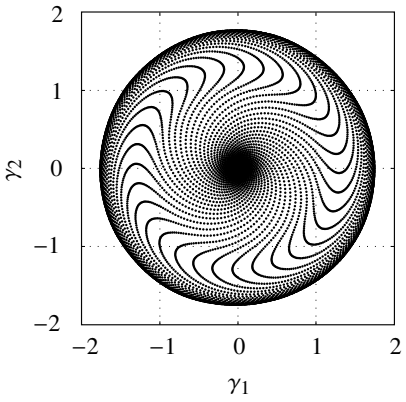

Figure 1.4 LLE of 16 cylinder wake transients. The figures displays the first two embedding coordinates $\boldsymbol{\gamma} = [\gamma_1, \gamma_2]^{\mathrm{T}}$ resulting from $K = 15$ nearest neighbors. For details, see Ehlert et al. (2020).

proximity maps have been proposed. In case of control, the error may include also the cost function to bring similarly performing states closer together (Kaiser et al. 2017).

1.3.2 Identifying the Manifold with Local Linear Embedding

LLE (Roweis & Lawrence 2000) targets to approximate M data points of a typically high-dimensional space by a low-dimensional manifold. In particular, neighboring points in the original data space remain neighbors in the low-dimensional embedding space.

The result is an optimal mapping from the snapshots $\boldsymbol{u}^m$ to N-dimensional features $\boldsymbol{\gamma}^m \in \mathcal{R}^N$. The neighborhood relation is preserved as follows. Let $i_1^m, \ldots, i_K^m$ be the indices of the K closest snapshots to the mth one. Let

$$\boldsymbol{u}^m \approx \sum_{k=1}^{K} w_{mk} \boldsymbol{u}^{i_k^m}$$

be the best approximation of the mth snapshot by its neighbors with optimized nonnegative weights w_{mk} adding up to unity. These constraints on the weights enforce a local interpolation. Then, also the feature vector can be approximated by the same expansion, $\boldsymbol{\gamma}^m \approx \sum_{k=1}^{K} w_{mk} \boldsymbol{\gamma}^{i_k^m}$. Here, K is a design parameter. It must be larger than the dimension of the manifold yet sufficiently small for the assumed locally linear behavior of the manifold. N is increased until convergence of the error is reached. Then, N denotes the dimension of the manifold. For the details, we refer to the original literature (Roweis & Lawrence 2000).

Figure 1.4 displays the LLE feature coordinates of the wake snapshot data. The origin corresponds to the steady solution $\boldsymbol{u}_s$, while the outer circle represents

https://doi.org/10.1017/9781108896214.004 Published online by Cambridge University Press

post-transient vortex shedding states. An analysis of the polar representation $\gamma_1 + \mathrm{j}\gamma_2 = r\exp(\mathrm{j}\theta)$ reveals that the distance r correlates well with the fluctuation energy, while θ can be considered the vortex shedding phase (Ehlert et al. 2020).

1.3.3 Other Features

Evidently, many other features can be constructed. An obvious design parameter of CMDS and LLE is the chosen distance between two snapshots, for example, the domain and the considered state variables.

Here, we mention one feature vector which is of large relevance to experiments: time-delay coordinates from sensor signals (Takens 1981). While reduced-order representations of fluid flows reduce the dimension of the state, time-delay coordinates increase the dimension of the measured signal to a level where the trajectories do not cross each other (no false neighbors) and a dynamical system can be identified. Loiseau et al. (2018) discuss the construction of velocity field manifold for the transient cylinder wake from the lift coefficient.

1.4 Low-Dimensional Representations

Low-dimensional state representations are key enablers for understanding, state estimation, dynamic modeling, and control. The starting point of this section is the ensemble of M flow snapshots $\boldsymbol{u}^m(\boldsymbol{x})$, $m = 1, \ldots, M$ (Section 1.2). A low-dimensional data-driven representation is synonymous with an *autoencoder* in machine learning. An autoencoder targets a low-dimensional parameterization of the snapshot data, say in $\mathbb{R}^N$. More precisely, an autoencoder comprises an *encoder* $\boldsymbol{G}$ from the high- or infinite-dimensional state-space to a low-dimensional feature space, for example,

$$\boldsymbol{u}^m \mapsto \boldsymbol{a}^m := \boldsymbol{G}(\boldsymbol{u}^m) \in \mathbb{R}^N, \quad m = 1, \ldots, M \tag{1.10}$$

and a *decoder* or state estimator $\boldsymbol{H}$, for example,

$$\boldsymbol{a}^m \mapsto \hat{\boldsymbol{u}}^m := \boldsymbol{H}(\boldsymbol{a}^m), \quad m = 1, \ldots, M \tag{1.11}$$

for the reconstruction of the state. Ideally, the autoencoder identifies the best possible pair of encoder $\boldsymbol{G}$ and decoder $\boldsymbol{H}$ that minimizes the in-sample error of the estimator/decoder

$$E_{in} := \frac{1}{M}\sum_{m=1}^{M} \|\hat{\boldsymbol{u}}^m - \boldsymbol{u}^m\|_\Omega^2 . \tag{1.12}$$

1.4.1 Optimal Subspaces with Proper Orthogonal Decomposition

POD can be considered as an optimal linear autoencoder onto an N-dimensional affine subspace. Let $\boldsymbol{u}_0$ be the average of the snapshot ensemble, $\boldsymbol{u}_i$, $i = 1, \ldots, N$ be the N

https://doi.org/10.1017/9781108896214.004 Published online by Cambridge University Press

POD modes, and a_i be the corresponding mode coefficients. Then the encoder $\boldsymbol{G}$ of a velocity field $\boldsymbol{u}$ to the mode amplitudes $\boldsymbol{a} = [a_1, \ldots, a_N]^{\mathrm{T}}$ is defined by

$$a_i := (\boldsymbol{u} - \boldsymbol{u}_0, \boldsymbol{u}_i)_\Omega, \quad i = 1, \ldots, N, \tag{1.13}$$

while the decoder $\boldsymbol{H}$ reads

$$\hat{\boldsymbol{u}}(\boldsymbol{x}) = \boldsymbol{u}_0(\boldsymbol{x}) + \sum_{i=1}^{M} a_i \boldsymbol{u}_i(\boldsymbol{x}). \tag{1.14}$$

POD modes are an orthonormal set and sorted by energy content. The optimality condition (e.g., Holmes et al. 2012) implies a minimal in-sample representation error from (1.12). We cannot find another Nth order Galerkin expansion (more precisely, a different N-dimensional subspace) yielding a smaller error.

For the transient wake data, the most energetic POD modes $\boldsymbol{u}_i$ and the amplitude evolution a_i are displayed in Fig. 3 and 4 of Noack et al. (2016). The first two modes correspond to von Kármán vortex shedding. The third mode resolves the change of the mean field. The following modes mix different frequencies and wavelengths.

1.4.2 Coarse-Graining the Data into Bins with Clustering

The key idea of clustering is representing the snapshots by a small number, say K, of centroids $\boldsymbol{c}_k$ with $k = 1, \ldots, K$. Every snapshot $\boldsymbol{u}^m$ can be associated with its closest centroid $\boldsymbol{c}_k$. Thus, the encoder $\boldsymbol{G}$ maps the velocity field $\boldsymbol{u}$ to $k \in \{1, \ldots, K\}$, the index of the closest centroid. In other words, the encoder creates "bins" of similar snapshots. The decoder $\boldsymbol{H}$ approximates the velocity field by the closest centroid $\hat{\boldsymbol{u}} = \boldsymbol{c}_k$. The k-means algorithm aims to minimize the in-sample representation error (Arthur & Vassilvitskii 2007). For generic data, the centroids can be expected to be unique modulo the index numbering. Clustering with K bins cannot yield a lower in-sample error than a POD representation with K modes $\boldsymbol{u}_i$, $i = 0, \ldots, K-1$. Both clustering and POD span a $K-1$-dimensional subspace, but a POD expansion can interpolate states while the centroids are fixed representations of the corresponding snapshot bin.

Figure 1.5 displays the results for 10 centroids visualized with LLE features. The centroids attempt to fill the circular region. One centroid is the steady solution; seven centroids resolve the limit cycle; and a phase opposite pair of centroids represent low-amplitude oscillations.

1.4.3 Comparison and Discussion

POD and clustering are quite different autoencoders. While POD is based on a superposition of modes, centroids are representative states that cannot be superimposed. LLE can be generalized to an autoencoder. The feature vector $\boldsymbol{\gamma}$ is obtained in the encoder step. A decoder can easily be constructed, for instance, with K-nearest neighbor interpolation (Ehlert et al. 2020).

Figure 1.6 displays the reconstruction error of the three methods for the simulation data. All three methods have the largest reconstructing error in the transient phase

https://doi.org/10.1017/9781108896214.004 Published online by Cambridge University Press

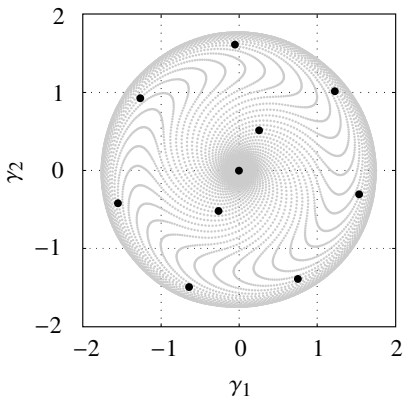

Figure 1.5 Cluster centroids localized in the LLE-based feature space. One centroid represents the steady-state solution; two resolve the opposite transient phases; and the remaining eight centroids are close to the limit cycle. For details, see Ehlert et al. (2020).

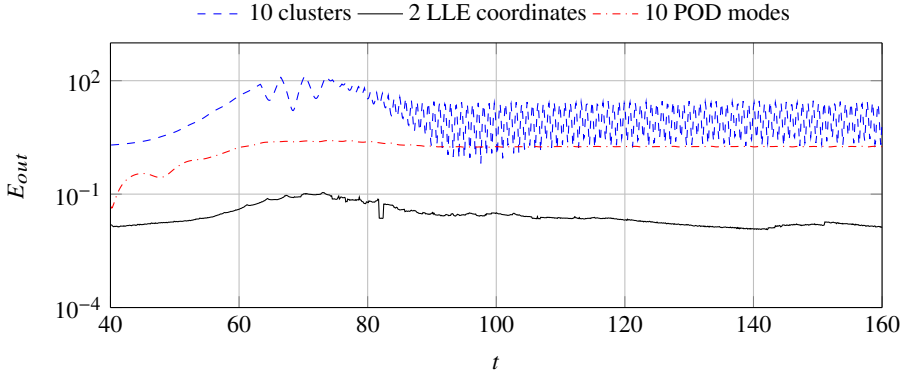

Figure 1.6 Out-of-sample error E_{out} for a new simulation trajectory at $Re = 100$. The solid line corresponds to LLE representations. The red dash-dotted curve and blue dashed curve refer to approximations with 10 centroids and 10 POD modes, respectively. For details, see Ehlert et al. (2020).

between $t = 60$ and $t = 80$. LLE significantly outperforms both POD and clustering by up to three orders of magnitudes, highlighting the two-dimensional manifold of the Navier–Stokes dynamics and a niche application of LLE. As expected, clustering performs worst lacking any intrinsic interpolation. The low error of the LLE-based autoencoder demonstrates that the dynamics is effectively two-dimensional. Yet, about 50 POD modes are necessary for a similar resolution as corroborated by Loiseau et al. (2018) for a similar manifold approximation. Evidently, POD-based representations are not efficient for slowly changing oscillatory coherent structures.

Data-driven Galerkin expansions have been optimized for a myriad of purposes. *Dynamic mode decomposition* (DMD) (Rowley et al. 2009, Schmid 2010) can extract

https://doi.org/10.1017/9781108896214.004 Published online by Cambridge University Press

stability modes from initial transients and Fourier modes from converged post-transient data. However, the performance for transient wakes is disappointing while a recursive DMD can keep some advantages of POD and DMD (Noack et al. 2016). Flexible state-dependent modes may significantly improve the accuracy of a low-order representation (Siegel et al. 2008, Tadmor et al. 2011, Babaee & Sapsis 2016).

1.5 Dynamic Models: From Proper Orthogonal Decomposition to Manifolds

In this section, a path to a least-order model for the transient cylinder wake is pursued. The starting point is the POD Galerkin method (Section 1.5.1). Then (Section 1.5.2), mean-field theory is employed to significantly simplify the dynamics. The simplification culminates in a manifold model with the Landau equation as dynamics (Section 1.5.3).

1.5.1 Proper Orthogonal Decomposition Galerkin Method

The traditional Galerkin method (Fletcher 1984) derives the dynamics of the Galerkin approximation (1.14) with orthonormal modes from the Navier–Stokes equations (1.1). Under weak conditions, a constant-linear-quadratic system of ordinary differential equations is obtained

$$\frac{da_i}{dt} = c_i + \sum_{j=1,\dots,N} l_{ij} a_j + \sum_{j,k=1,\dots,N} q_{ijk} a_j a_k . \tag{1.15}$$

These conditions may include the incompressibility of the flow, a stationary domain, stationary Dirichlet, periodic, or von Neumann boundary conditions, and smoothness of the flow. The coefficients c_i, l_{ij}, and q_{ijk} are functions of the modes and of the Reynolds number. The coefficients may also be identified from numerical solutions or experimental data (Galletti et al. 2004, Cordier et al. 2013), for instance, if the flow domain is too small or if the turbulent fluctuations are not resolved in (1.14). We refer to exquisite textbooks (Fletcher 1984, Holmes et al. 2012) for details. Schlegel and Noack (2015) have derived necessary and sufficient conditions for bounded solutions, which can be considered a requirement for a physical sound model.

1.5.2 Mean-Field Model

The Galerkin system (1.15) can be significantly simplified exploiting the manifold dynamics depicted in Figure 1.3 and derived in Section 1.5.1. The flow is dominated by a zeroth and first harmonics component (see (1.6)).

The zeroth component is on the line from the steady solution $\boldsymbol{u}_s$ to the mean flow $\overline{\boldsymbol{u}}$. This line approximates the one-period averaged flow $\boldsymbol{u}^D$ and is parameterized by the shift mode $\boldsymbol{u}_3$ and its amplitude a_3 (see (1.7)). The shift mode is effectively a backflow in the wake. The streamlines look like a fly and are depicted in Figure 1.3 (left middle subfigure).

https://doi.org/10.1017/9781108896214.004 Published online by Cambridge University Press

The first harmonics represents von Kármán vortex shedding, which may be resolved by a cosine $\boldsymbol{u}_1$ and sine $\boldsymbol{u}_2$ mode, ignoring the shape deformation for a moment:

$$\boldsymbol{u}'(\boldsymbol{x},t) = a_1(t)\,\boldsymbol{u}_1(\boldsymbol{x}) + a_2(t)\,\boldsymbol{u}_2(\boldsymbol{x}). \tag{1.16}$$

The modes may be inferred from Figure 1.3 in the rightmost column. In the following, $\boldsymbol{u}_1$, $\boldsymbol{u}_2$, and $\boldsymbol{u}_3$ are assumed to be orthonormalized. The first three POD modes of the transient yield are already a good approximation of these modes.

Following mean-field arguments (Noack et al. 2003), the Galerkin system (1.15) simplifies to a self-amplified, amplitude limited oscillator,

$$da_1/dt = \sigma a_1 - \omega a_2, \qquad \sigma = \sigma_1 - \beta a_3, \tag{1.17a}$$

$$da_2/dt = \sigma a_2 + \omega a_1, \qquad \omega = \omega_1 + \gamma a_3, \tag{1.17b}$$

$$da_3/dt = \sigma_3 a_3 + \alpha\left(a_1^2 + a_2^2\right). \tag{1.17c}$$

The oscillator has three parameters σ_1, ω_1, σ_3 for linear dynamics and three parameters for α, β, γ for the weakly nonlinear effects of Reynolds stress. Intriguingly, *sparse identification of nonlinear dynamics* (SINDy) extracts precisely this sparse dynamical system from transient simulation data (Brunton, Proctor & Kutz 2016a).

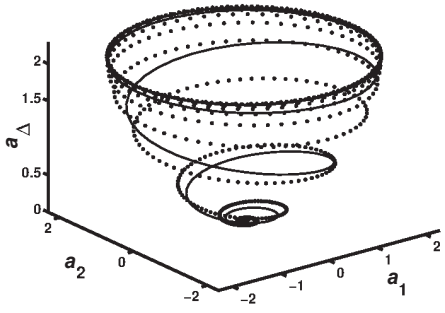

Figure 1.7 Transient dynamics of the cylinder wake from the DNS (solid line) and the mean-field Galerkin system (1.17) (dashed line). Here, $a_\Delta = a_3$. Phase portrait from Tadmor and Noack (2004).

Figure 1.7 shows that the mean-field Galerkin system (1.17) and the direct numerical simulation agree well. A detailed analysis (Tadmor & Noack 2004) quantitatively corroborates this agreement for the manifold and the temporal evolution.

1.5.3 Manifold Model

The mean-field Galerkin system can be further simplified exploiting the slaving of the shift mode to the fluctuation level. In fact, the calibrated $\sigma_3 \approx -6\sigma_1$ shows a much quicker convergence to manifold than the growth of the fluctuation. Hence, (1.17c) is well approximated by the algebraic equation

$$a_3 = \frac{\alpha_3}{|\sigma_3|}\left(a_1^2 + a_2^2\right), \tag{1.18}$$

https://doi.org/10.1017/9781108896214.004 Published online by Cambridge University Press

explaining the mean-field parabola shape in the figure. Equation (1.18) in (1.17a), (1.17b) and the introduction to polar coordinates $a_1 = r\cos\theta$, $a_2 = r\sin\theta$ lead to the famous Landau equation (Landau & Lifshitz 1987) for a supercritical instability,

$$dr/dt = \sigma_1 r - \beta' r^3, \quad d\theta/dt = \omega_1 + \gamma' r^2. \tag{1.19}$$

These equations show an exponential growth by linear instability and a cubic damping by Reynolds stress and mean-field deformation. The frequency changes as well. The nonlinearity parameters β' and γ' can easily be derived from (1.17). Intriguingly, this equation is found to remain accurate even far from the onset of vortex shedding.

The resulting Landau model does not include higher harmonics. Even worse, we have assumed fixed modes $\boldsymbol{u}_1$, $\boldsymbol{u}_2$. Yet, the prominent stability mode near the steady solution is distinctly different from the POD modes characterizing the limit cycle. a_3-dependent modes can cure this shortcoming (Morzyński et al. 2006), but the model is still blind to higher harmonics. A more accurate model is based on the LLE feature coordinates $\gamma_1 = r\cos\theta$ and $\gamma_2 = r\sin\theta$ and an identified Landau equation for the dynamics of r and θ. The LLE autoencoder incorporates the mode deformation and higher harmonics (Ehlert et al. 2020). Loiseau et al. (2018) has derived such a model based on similar premises. The mean-field Galerkin model can be generalized for two (and more) frequencies (Luchtenburg et al. 2009).

1.6 Control: Model-Based and Model-Free Approaches

This section previews two flow control approaches. Section 1.6.1 follows the classical paradigm of model-based control design, while Section 1.6.2 outlines a model-free machine-learned control optimization.

1.6.1 Model-Based Control

As a control benchmark (Hinze & Kunisch 2000), we aim to stabilize the cylinder wake with a transversal volume force in the near wake (see Figure 1.8). This admittedly academic volume force actuator is surprisingly often used in the computational flow control literature and significantly simplifies the discussion. As added complexity, experimental conditions are emulated by the hot-wire measurement of the streamwise velocity component s. The goal is stabilizing sensor-based control law (see, again, Figure 1.8).

The volume force can be shown to lead to an extra term gb in the mean-field system (Gerhard et al. 2003),

$$da_1/dt = \sigma a_1 - \omega a_2 + gb, \qquad \sigma = \sigma_1 - \beta a_3, \tag{1.20a}$$

$$da_2/dt = \sigma a_2 + \omega a_1, \qquad \omega = \omega_1 + \gamma a_3, \tag{1.20b}$$

$$a_3 = \alpha'\left(a_1^2 + a_2^2\right), \qquad \alpha' = \alpha/|\sigma_3|. \tag{1.20c}$$

https://doi.org/10.1017/9781108896214.004 Published online by Cambridge University Press

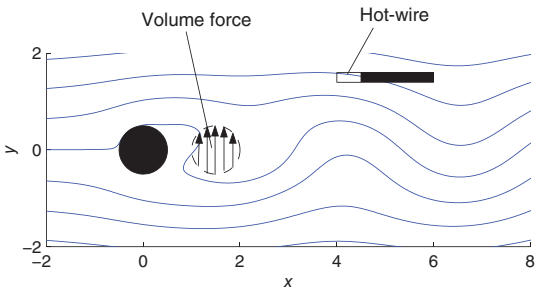

Figure 1.8 Cylinder wake configuration with volume force actuation and hot-wire sensor.

In this differential algebraic system, a_3 is slaved to the fluctuation energy (1.18). Here, b is the actuation command, for example, the induced acceleration in the circular region, while the positive gain g quantifies the effect on the dynamics and depends, for instance, on the size and location of the volume force support. Without loss of generality, the actuation term only effects a_1 by suitable rotation of the modes $\boldsymbol{u}_1$, $\boldsymbol{u}_2$.

The forced growth rate of the fluctuation energy $K = r^2/2 = (a_1^2 + a_2^2)/2$ reads

$$\frac{dK}{dt} = a_1 \frac{da_1}{dt} + a_2 \frac{da_2}{dt} = \sigma r^2 + g\, b\, a_1. \tag{1.21}$$

The fluctuation energy can be mitigated with negative actuation power $g\, b\, a_1$, that is, b has to have the opposite sign of a_1. For simplicity, a linear control law is assumed,

$$b = -k a_1, \qquad k > 0. \tag{1.22}$$

The control gain k shall ensure a forced exponential decay rate $\sigma_c < 0$ of the amplitude r. This implies with (1.22) and (1.21),

$$\frac{dK}{dt} = \sigma_c r^2 = \sigma r^2 + g\, k\, a_1^2. \tag{1.23}$$

Averaging over one period and exploiting $\overline{a_1^2} = r^2/2$ yields the control gain k and thus the control law

$$b = 2\frac{\sigma_c - \sigma}{g} a_1. \tag{1.24}$$

The implications of this law are plausible: the higher the unforced growth rate and the higher the desired damping, the larger the volume force amplitude must become. Contrary, the larger the gain g of actuation on the dynamics, the smaller the volume force needs to be. It should be born in mind that σ is r dependent.

For the sensor-based control, the state $\boldsymbol{a} = [a_1, a_2]^{\mathrm{T}}$ is estimated from the sensor signal with a dynamic observer. The resulting stabilization effect is shown in Figure 1.9. While the model allows complete stabilization, the fluctuation energy of the DNS has been reduced by 30% in the observation region $x < 6$. The model is only partially accurate as it ignores shedding mode changes due to actuation and convection

https://doi.org/10.1017/9781108896214.004 Published online by Cambridge University Press

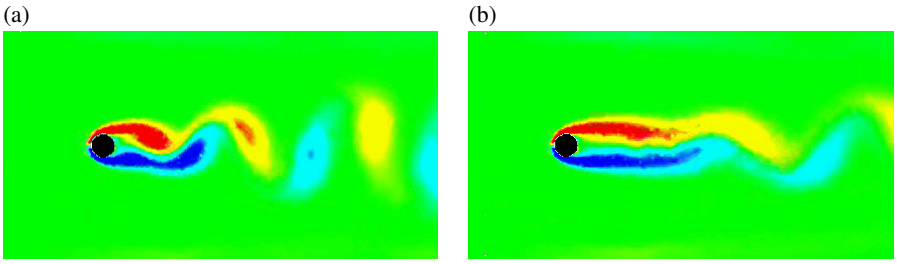

Figure 1.9 Model-based cylinder wake stabilization. (a) unforced and (b) controlled flow. Here, the vorticity field is shown: red and blue mark negative and positive vorticity, while green corresponds to potential flow.

effects. The maximum reduction of the fluctuation level is 60%, leading to a complete stabilization of the near wake and a residual fluctuation in the far wake.

1.6.2 Model-Free Machine Learning Control

In Section 1.6.1, the inaccuracy of the model has led to a reduced control performance. In model-free control, such model-based errors are avoided. Instead, a regression problem is solved: find a control law $b = K(s)$ that minimizes the cost function J, for example, the fluctuation energy. Any regression solver requires repeated tests of control laws in the full plant for the optimization problem. A linear ansatz for the control law allows for a gradient-based approach, for example, the downhill simplex method, for parameter optimization. If no structure of the control law shall be assumed, for example, in case of strongly nonlinear dynamics, MLC has proven to be very powerful. MLC is based on genetic programming as regression solver and has optimized the nonlinear control law in dozens of experiments and simulations (Noack 2019). The observed testing requires hundreds to one thousand control laws from simple single-input single-output (SISO) to complex nonlinear multiple-input multiple-output (MIMO) control.

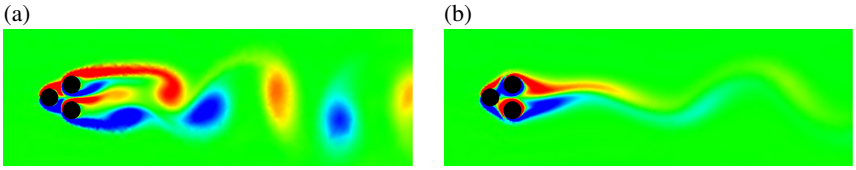

Figure 1.10 Machine learning control of the fluidic pinball with 3 cylinder rotations responding to 15 undisplayed downstream sensors. Contour levels of vorticity for the unforced flow (a) and controlled flow (b) by the control law. Solid lines and dashed lines represent respectively positive and negative vorticity. For details, see Cornejo Maceda et al. (2019).

Figure 1.10 provides a synopsis for MLC applied to the fluidic pinball configuration (Cornejo Maceda et al., (2019), (2021)). Forty-two percent reduction of the effective drag power is achieved accounting for actuation energy. The enabling MIMO feedback

https://doi.org/10.1017/9781108896214.004 Published online by Cambridge University Press

control has 15 downstream sensors (not displayed) and commands the rotation of the three cylinders. The mechanism is a combination of open-loop Coanda forcing and closed-loop phasor control (as in the cylinder wake example).

1.7 Conclusions

This chapter provides examples of machine learning applications in fluid mechanics. The examples are from analysis, modeling, and control of the oscillatory cylinder wake. All applications are based on regression problems, that is, finding a function that minimizes a cost. Feature extraction leads to a mapping from the flow to few coordinates optimizing some neighborhood/distance criteria. An autoencoder comprises an encoder and decoder for a low-dimensional flow representation that minimizes the representation error for the snapshot data. Dynamic modeling relies on a mapping from the state to the change of state. State estimation is a function from the measured sensor signals to the flow field. Control design leads to a function from the state or sensor signals to the actuation command minimizing a given cost. Closures can be seen as control terms that minimize the tracking error between computed and observed states. More examples can be added (Brunton & Noack 2015).

Machine learning provides powerful tools for regression solvers based on existing data ("curve fitting") or based on in situ optimization ("variational problems"). The discussed data-based regressions – classical multidimensional scaling, local linear embedding, proper orthogonal decomposition, clustering – can be used for a wide range of problems. Genetic programming is a powerful tool for solving variational problem, control design or closures, without required data but with in situ testing.

The oscillatory cylinder wake and its stabilization looks like an innocently simple benchmark. Yet, it shows already that machine learning methods need to be carefully chosen. For instance, local linear embedding does a remarkable job of compressing the data to a two-dimensional manifold. In contrast, 50 POD modes are required for a similar representation error making understanding, state estimation, dynamic modeling, and model-based control next to impossible. We will elaborate on machine learning methods and more advanced applications for reduced-order modeling (Chapter 14) and for control (Chapter 17). We highly recommend three introductory books for machine learning: Abu-Mostafa et al. (2012) as a deep introduction to how one can learn from data, Wahde (2008) for optimization solvers, and Burkov (2019) for an inspiring overview of machine learning methods.

https://doi.org/10.1017/9781108896214.004 Published online by Cambridge University Press