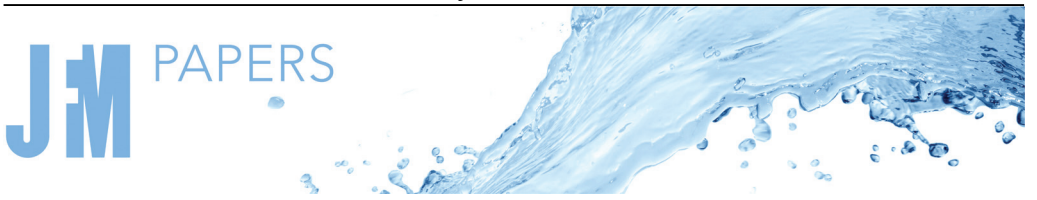
*J. Fluid Mech.* (2024), *vol*. 999, A86, doi:10.1017/jfm.2024.904



# Data-driven stabilization of an oscillating flow with linear time-invariant controllers

**William Jussiau**[1],†, **Colin Leclercq**[2], **Fabrice Demourant**[1] **and Pierre Apkarian**[1]

[1]ONERA/DTIS, Université de Toulouse, 31000 Toulouse, France

[2]ONERA/DAAA, Institut Polytechnique de Paris, 92190 Meudon, France

(Received 15 April 2024; revised 5 July 2024; accepted 9 September 2024)

This paper presents advances towards the data-based control of periodic oscillator flows, from their fully developed regime to their equilibrium stabilized in closed loop, with linear time-invariant (LTI) controllers. The proposed approach directly builds upon the iterative method of Leclercq *et al.* (*J. Fluid Mech.*, vol. 868, 2019, pp. 26–65) and provides several improvements for an efficient online implementation, aimed at being applicable in experiments. First, we use input–output data to construct an LTI mean transfer functions of the flow. The model is subsequently used for the design of an LTI controller with linear quadratic Gaussian synthesis, which is practical to automate online. Then, using the controller in a feedback loop, the flow shifts in phase space and oscillations are damped. The procedure is repeated until equilibrium is reached, by stacking controllers and performing balanced truncation to deal with the increasing order of the compound controller. In this article, we illustrate the method for the classic flow past a cylinder at Reynolds number $Re = 100$. Care has been taken such that the method may be fully automated and hopefully used as a valuable tool in a forthcoming experiment.

**Key words:** control theory, instability control

## 1. Introduction

For decades, understanding and controlling fluid flows have proven to be a considerable challenge due to their inherent complexity and potential impact, attracting attention from technological and academic research (Brunton & Noack 2015). In engineering, applications span several domains, from drag reduction in transport, to lift increase or

† Email address for correspondence: william.jussiau@gmail.com

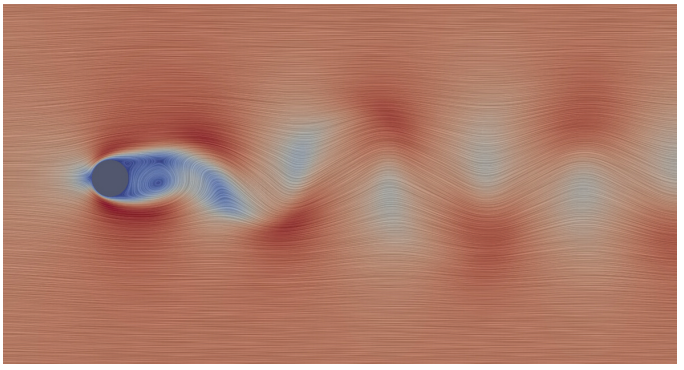


Figure 1. Streamlines of a snapshot of the incompressible flow past a two-dimensional cylinder at Reynolds number $Re = 100$. Coloured by velocity magnitude.

acoustic noise reduction for aircraft, or mixing enhancement in chemical processes. Earlier approaches mainly used passive control (i.e. without exogenous input), while active control of flows is today an ongoing and fruitful research topic (Schmid & Sipp 2016; Garnier *et al.* 2021). Active flow control can be itself separated into two categories: open-loop control and closed-loop control. While open-loop control proves to be effective (e.g. with harmonic signals in Bergmann, Cordier & Brancher (2005)), it usually requires large and sustained control inputs to drive the system to a more beneficial regime. On the other hand, closed-loop control aims at modifying the intrinsic dynamics of the system, therefore usually requiring less energy, at the cost of an increased complexity in the design and implementation phases.

In this article, we are interested in the feedback control of laminar oscillator flows (Schmid & Sipp 2016). They form a particular class of flows that are linearly unstable, dominated by a nonlinear regime of self-sustained oscillations and mostly insensitive to upstream perturbations. The most well-known example of oscillator flow is probably the flow past a cylinder in two dimensions (Barkley 2006), displaying a wake of alternating vortices known as the von Kármán vortex street (see figure 1). This category of flows generally exhibits two equilibria or more: an unstable fixed point (referred to as the base flow) and an unsteady attractor. In this application, the objective is to drive the flow from an initial state lying on the attractor, to the base flow stabilized in closed loop. It is notable that open-loop control strategies cannot stabilize the equilibrium, while closed-loop strategies can be designed as such.

From a control perspective, closed-loop strategies developed in the literature may be categorized based on the flow regime they aim to address. The first category of approaches focuses on preventing the growth of linear instabilities from a neighbourhood of the stationary equilibrium, while the second category of approaches tackles the reduction of oscillations in the fully developed nonlinear regime of self-sustained oscillations. For the control law design, the main difficulty is due to the Navier–Stokes equations that are both nonlinear and infinite-dimensional. Often, the choice of the regime of interest naturally induces a structure for the control policy (e.g. linear or nonlinear, static or dynamic, etc.) and/or a controller design method. In the following, we propose to sort control approaches depending on the flow regime addressed, and we showcase some of the methods used to circumvent the difficulty posed by the high dimensionality and nonlinearity of the controlled system.

### 1.1. *Control in a neighbourhood of the equilibrium*

The historical approaches to oscillator flow control aim at efficiently counteracting the linear instabilities, whose development from equilibrium is eventually responsible for self-sustained oscillations. To do so, one may linearize the equations about the unstable equilibrium and work in the linear regime, in a neighbourhood of the equilibrium. For applying linear synthesis techniques to the inherently high-dimensional system, a common workaround is the use of a linear reduced-order model (ROM) that captures the most essential features of the flow around a set point. Linear ROMs may be established with a wide range of techniques, such as Galerkin projection of the governing equations (Barbagallo, Sipp & Schmid 2009, 2011; Weller, Camarri & Iollo 2009), system identification from time data (Illingworth, Morgans & Rowley 2011; Flinois & Morgans 2016) or frequency data (Jin, Illingworth & Sandberg 2020), $\mathcal{H}_\infty$ balanced truncation (Benner, Heiland & Werner 2022) or low-order conceptual physical modelling (Illingworth, Morgans & Rowley 2012). Using linear ROMs, linear control techniques of various complexity may be applied: the control techniques range from proportional control (Weller *et al.* 2009) and linear quadratic Gaussian (LQG) (Barbagallo *et al.* 2009, 2011; Illingworth *et al.* 2011, 2012) to $\mathcal{H}_\infty$ synthesis or loop-shaping (Flinois & Morgans 2016; Jin *et al.* 2020; Benner *et al.* 2022). In an attempt to address some shortcomings of the linearized approaches, a linear parameter varying approach was proposed in Heiland & Werner (2023) for both low-order modelling and control; it shows an expanded basin of attraction of control strategies around the equilibrium for a weakly supercritical flow.

These approaches are mainly restricted by the region of validity of the low-dimensional linearized model of the flow, and the assumed knowledge of the equations. In particular, these approaches are only satisfactory in the vicinity of the equilibrium (or for weakly supercritical flows), but rapidly fail for strong nonlinearity when linearization about the equilibrium becomes irrelevant (Schmid & Sipp 2016).

### 1.2. *Control of the fully developed regime*

The second category of approaches tackles the reduction of oscillations in the fully developed nonlinear regime of self-sustained oscillations. To address this regime, both data-based and model-based approaches may be suitable.

#### 1.2.1. *Nonlinear reduced-order modelling and control*

In order to address the shortcomings of approaches using linear models, approaches were developed to handle the nonlinearity in different ways, especially with low-dimension nonlinear approximations of the flow. Linear ROMs may be extended with nonlinear terms in Galerkin projection, in order to reproduce the oscillating behaviour of the flow (e.g. Rowley & Juttijudata 2005; King *et al.* 2005; Lasagna *et al.* 2016). They may be consequently used for the design of various linear and nonlinear control methods such as linear parameter varying pole placement (Aleksić-Roeßner *et al.* 2014), model predictive control (MPC) (Aleksić-Roeßner *et al.* 2014), backstepping (King *et al.* 2005), sliding mode control (Aleksić *et al.* 2010), the sum-of-squares formulation (Lasagna *et al.* 2016; Huang *et al.* 2017)) or more physically based solutions (Gerhard *et al.* 2003; Rowley & Juttijudata 2005). These studies show that nonlinear ROMs may be used for the design of control methods and provide satisfactory performance in a high-dimensional nonlinear system. They, however, might require strong model assumptions for building a

nonlinear ROM (e.g. mode computation, full-field information, etc.), which may hinder their applicability in experiments with localized measurements, noise or model mismatch.

#### 1.2.2. *Approaches using the Koopman operator*

More recent approaches use input–output data to directly build a surrogate nonlinear model of the flow, and use it for control with the MPC framework, leveraging the prediction power of cheap surrogate models. Rooted in Koopman operator theory, Korda & Mezić (2018*a*) and Arbabi, Korda & Mezić (2018) developed a framework for computing a linear representation of a dynamical system, in a user-chosen lifted coordinate space. Using a moderate-dimension flow, they show the possibility of replacing full-state measurement by sparse measurements with delay embedding. A similar approach is used in Morton *et al.* (2018) with full-state measurement but the space of observables is learned with an encoder neural network, which is illustrated for a weakly supercritical cylinder wake flow. With the same idea, using Koopman operator theory, Peitz & Klus (2020) summarize two findings introduced in two previous papers: in Peitz & Klus (2019), the flow under any actuation signal is modelled using autonomous systems with constant input (derived from data), and the control problem is turned into a switching problem; in Peitz, Otto & Rowley (2020), a bilinear model is interpolated from said autonomous, constant-input flow models and the control problem is solved on the bilinear model. In a related work, Otto, Peitz & Rowley (2022) directly built a bilinear model using delayed sparse observables and applied said model to the fluidic pinball, even providing decent performance at off-design Reynolds number. In Page & Kerswell (2019), it is shown that a single Koopman expansion might not be able to reproduce the behaviour of the flow around two distinct invariant solutions (namely the stationary equilibrium and the attractor), which may underline the necessity of building several such models for efficient model-based control. Not explicitly linked to the Koopman operator, Bieker *et al.* (2019) modelled the actuated flow as a black box, in a latent space with a recurrent neural network that may be updated online, and demonstrated the possibility of efficiently addressing complex flows such as the chaotic fluidic pinball, by only using a small number of localized measurements, provided large amounts of training data are available.

#### 1.2.3. *Interacting with high-dimensional nonlinear systems*

On the other side of the spectrum, some techniques try to address the control of high-dimensional nonlinear systems with direct interaction with the system itself. Some of these approaches use tools and controller structures from linear control theory, such as with proportional integral derivative control (Park, Ladd & Hendricks 1994; Son & Choi 2018; Yun & Lee 2022), or structured $\mathcal{H}_\infty$ control (Jussiau *et al.* 2022), in order to design controllers by direct interaction with full-size nonlinearity, either heuristically (Park *et al.* 1994; Yun & Lee 2022) or with the help of optimization (Son & Choi 2018; Jussiau *et al.* 2022). Furthermore, a significant body of literature uses nonlinear model-free control laws, using information conveyed by the full-order nonlinear system. Examples of such approaches are found in Cornejo Maceda *et al.* (2021) and Castellanos *et al.* (2022), where nonlinear control laws consisting of nested operations ($+$, $\times$, cos, etc.) are built with optimization, for applications to the fluidic pinball. Naturally, the reinforcement learning framework has also proved its efficiency at discovering control policies (see Garnier *et al.* (2021) and Viquerat *et al.* (2022) for reviews of approaches, or Paris, Beneddine & Dandois (2021), Rabault *et al.* (2019) and Ghraieb *et al.* (2021) for illustrations). A recent study in reinforcement learning (Xia *et al.* 2023) demonstrates the effectiveness of including

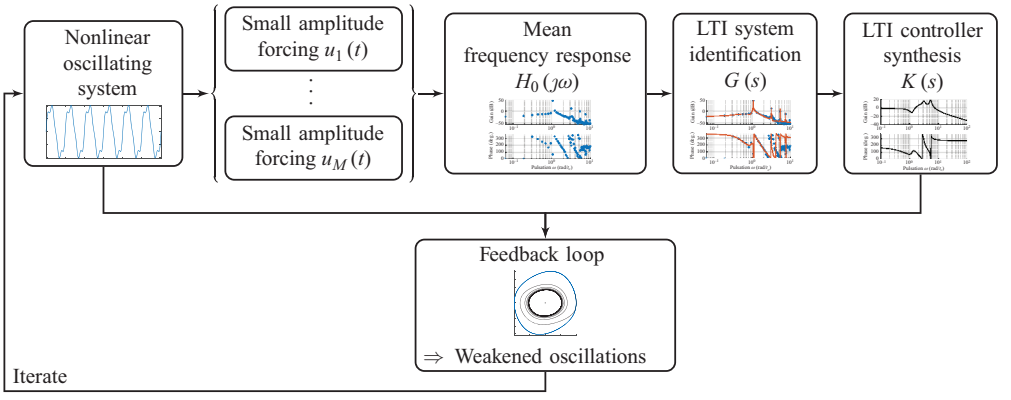



Figure 2. Graphical summary of the method: data-based stabilization of an oscillating flow with LTI controllers, using the mean resolvent framework (Leclercq & Sipp 2023).

delayed measurements and past control inputs within a nonlinear control policy in discrete time. This approach essentially transforms the control law from static to dynamic, utilizing the concept of dynamic output feedback from control theory (Syrmos *et al.* 1997).

While the use of data makes these approaches more easily applicable in experiments, they still require large amounts of data and their training can be made more challenging by various external factors, such as convective time delays stemming from convective phenomena or partial observability (i.e. localized sensing, often requiring numerous sensors to reconstruct worthwhile information).

### 1.3. *Proposed approach*

In this paper, we aim at driving the system from its natural limit cycle to its equilibrium, stabilized in closed loop, by handling the nonlinearity iteratively, with the same idea as Leclercq *et al.* (2019) that completely suppresses oscillations on top of a cavity, by solving the nonlinear control problem using a sequence of low-order linear approximations. Contrary to Leclercq *et al.* (2019), where the model is obtained from linearizing the Navier–Stokes equations about the mean flow, our proposed method is fully data-based in the sense that no knowledge about the governing equations is necessary. In addition, it aims at being easy to design and implement, and does not require multiple sensors, extensive training or tricky parameter tuning. It aims at handling the nonlinearity iteratively, and tackles the large dimension of the system with system identification solely from input–output data. Using the mean resolvent framework from Leclercq & Sipp (2023), we can establish a linear time-invariant (LTI) model of the oscillating flow, uniquely from input–output data, which is used to design a dynamic output feedback LTI controller. While the constructed controller successfully reduces oscillations in the flow, it cannot stabilize the flow completely due to the local validity (in phase space) of the LTI model. Consequently, the flow reaches a new dynamical equilibrium characterized by a lower perturbation kinetic energy. The procedure is then iterated from this new dynamical equilibrium, until the flow is fully stabilized – the procedure is illustrated in figure 2.

The paper is structured as follows. In § 2, we present and justify the method and its associated tools in detail. In § 3, we demonstrate the applicability of the method to the canonical two-dimensional flow past a cylinder at $Re = 100$, essentially reaching equilibrium with data-driven LTI controllers, and analyse the solution found. We discuss

| Step | Leclercq *et al.* (2019) | Our proposition |
|---|---|---|
| Modelling | Linearization of the dynamics around the time-averaged flow | Identification of mean transfer function from input–output data (§ 2.4) |
| Controller synthesis | Hand-tuned structured $\mathcal{H}_\infty$ | Low-gain LQG (§ 2.5) |
| Controller update | Full-order stacking | Stacking, balanced truncation and state initialization (§ 2.6) |

Table 1. Converting the method of Leclercq *et al.* (2019) into a data-based and automated method.

several points of the method in § 4. Finally, the main results are recalled and perspectives are drawn in § 5.

## 2. Method and tools

### 2.1. *Objective and overview of the method*

We consider the flow past a cylinder in two dimensions, at Reynolds number $Re = 100$ (presented in more detail in § 2.3). This flow is an oscillator flow with an unstable equilibrium, referred to as the base flow, and a periodic attractor, which is the regime naturally observed: the von Kármán vortex street (figure 1). Our objective is to drive the system from its attractor back to its equilibrium stabilized in closed loop, using a single local sensor and a single actuator, in a fully data driven way (i.e. without the need for knowledge of the equations, the base flow or sensor/actuator models).

The procedure is based on the same idea as that described in Leclercq *et al.* (2019). In that previous study, the oscillating flow is modelled, from an input–output viewpoint, as an LTI system enabling LTI controller design. However, as the controller is unable to completely stabilize the flow, the feedback system converges to a new attractor with lower perturbation energy, and the procedure is reiterated until the base flow is reached. More specifically, in Leclercq *et al.* (2019), the authors used a linearized model around the mean flow (i.e. equations linearized around the temporal average $\bar{\boldsymbol{q}} = \langle \boldsymbol{q}(t) \rangle_t$ of a statistically steady flow), which is shown to represent features of the flow important for control (Liu *et al.* 2018). Although the mean flow may be estimated in experiments, it remains quite impractical for applications, and the linearization performed requires significant assumptions (e.g. models for both the sensor and actuator, neglecting three-dimensional effects, working with expensive three-dimensional equations, etc.). Also, they used multi-objective structured $\mathcal{H}_\infty$ synthesis (Apkarian & Noll 2006; Apkarian, Gahinet & Buhr 2014) for the design of low-order controllers. While this technique is very powerful and well suited to this problem (as it can enforce, for example, controller structure, roll-off or location of poles in the closed loop), it is not easy to automate and often requires the control engineer perspective to be used at its maximum potential. Finally, as the controllers are being stacked onto each other during the iterations, the controller effectively operating in closed loop has its order increasing linearly with the number of iterations. In an experiment where the procedure would likely never really converge to a steady equilibrium (due, at least, to residual incoming turbulence), this ever-increasing order could be a problem for runtime and numerical conditioning.

In this study, we tackle these three shortcomings preventing the use of the method in an automated data-based manner, summarized in table 1. First, the modelling part is done solely with input–output data, using the mean resolvent framework introduced in Leclercq

& Sipp (2023), identified with multisine excitations (Schoukens, Guillaume & Pintelon 1991). Not only is the mean transfer function easier to derive and implement, but it is also better founded than the resolvent around the mean flow (Leclercq & Sipp 2023). Second, the controller is designed with LQG synthesis, which is easy to automate. While this method is arguably less powerful and flexible than structured $\mathcal{H}_\infty$ synthesis, it permits easy automation while maintaining some desirable properties for the controller. Third, the increasing order of stacked controllers is managed with online controller reduction with balanced truncation (denoted $\mathcal{B}_T$; see Moore 1981; Zhou, Salomon & Wu 1999) and transient regimes are handled with a two-step initialization of the new controller. Overall, these advantages aim at making the method applicable in a real-life set-up. The overview of the method is as follows, with $i$ the iteration index:

$\mathcal{S}.1$ Simulate flow with feedback controller $\tilde{K}_i(s)$. After transient regime, reach new statistically steady regime (dynamical equilibrium). (§ 2.3)
↪ At iteration 0, controller is null: $\tilde{K}_0(s) = 0$.

$\mathcal{S}.2$ Compute LTI ROM of the oscillating closed loop: $G_i(s)$. (§ 2.4)
↪ With input–output data.

$\mathcal{S}.3$ Synthesize new controller $K_i^+(s)$ for the identified ROM. (§ 2.5)
↪ Automated synthesis.

$\mathcal{S}.4$ Stack controllers and reduce the order with the balanced truncation $\mathcal{B}_T$: define $\tilde{K}_{i+1}(s) = \mathcal{B}_T(\tilde{K}_i(s) + K_i^+(s))$ to use in closed loop. (§ 2.6)
↪ Reduce controller order and control input transient.

$\mathcal{S}.5$ Back to $\mathcal{S}.5$ and iterate until condition is met.
↪ For example, low norm of sensing signal.

## 2.2. *Notations*

### 2.2.1. *Notations for the iterative procedure*

One repetition of the identification-control process is referred to as an iteration, and they are repeated until the equilibrium is reached. The following paragraph is described graphically in figure 3. At the start of the process, numbered iteration $i = 0$, the flow is simulated from its perturbed unstable equilibrium, with the feedback controller $\tilde{K}_0 = 0$. Therefore, the flow evolves towards its natural limit cycle ($\mathcal{S}.5$). When the limit cycle is attained, at time $t_0^I$, an exogenous signal $u_\Phi(t)$ is injected for the identification of a model $G_0(s)$ of the oscillating flow with data from $t \in [t_0^I, t_0^K]$ ($\mathcal{S}.2$). Then, a controller $K_0^+(s)$ is synthesized ($\mathcal{S}.3$) to control $G_0(s)$. At time $t < t_0^K$, only the controller $K_0 = 0$ is in the loop. At time $t \geq t_0^K$, the new full-order controller is $K_1 = \tilde{K}_0 + K_0^+$. After a short duration $T_{sw}$ (explained in § 2.6.3), $K_1$ is reduced with balanced truncation $\mathcal{B}_T$ and its low-order counterpart $\tilde{K}_1 = \mathcal{B}_T(K_0 + K_0^+)$ is used in its place ($\mathcal{S}.4$). The flow reaches a new dynamical equilibrium with lower perturbation kinetic energy. Iteration $i = 1$ starts at $t = t_0^K$, and the notations are alike for the rest of the procedure.

### 2.2.2. *Control theory notations*

The order of an LTI plant $G$ is denoted $\partial^\circ G \in \mathbb{N}$. The state-space representation of a transfer $G(s)$ with associated matrices ($\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{D}$) is denoted as

$$G = \left[\begin{array}{c|c} \boldsymbol{A} & \boldsymbol{B} \\ \hline \boldsymbol{C} & \boldsymbol{D} \end{array}\right], \tag{2.1}$$

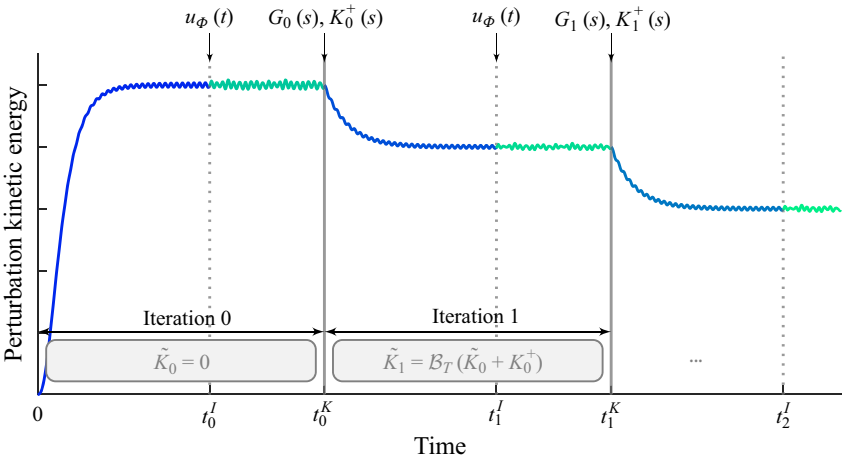



Figure 3. At each iteration $i$, a time simulation is performed in closed loop with the controller $\tilde{K}_i(s)$; then, an exogenous signal $u_\Phi(t)$ is injected for the identification of an LTI model $G_i(s)$, for which an LTI controller $K_i^+(s)$ is synthesized. This corresponds to the start of iteration $i+1$, where the controller in the loop is $\tilde{K}_{i+1} = \mathcal{B}_T(\tilde{K}_i + K_i^+)$ that should drive the flow to a new dynamical equilibrium with lower perturbation kinetic energy. The process is then repeated.

such that the state, input and output $\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{y}$ associated with this state-space realization of $G$ are as follows:

$$\begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u}, \\ \boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}\boldsymbol{u}. \end{cases} \tag{2.2}$$

Note that most of the plants used in this study are single-input, single-output (SISO), i.e. with scalar $u, y$. Also, for two plants $G_1, G_2$ with common input $u$ and respective outputs $y_1, y_2$, the plant sum $\Sigma = G_1 + G_2$ is defined as the plant with input $u$ and output $y_\Sigma = y_1 + y_2$. Its state-space representation is derived easily.

### 2.3. *Use case: flow past a cylinder at low Reynolds number*

#### 2.3.1. *Configuration*

In this paper, the use case is an incompressible bidimensional flow past a cylinder, used in numerous past studies with slightly different set-ups and various control methods (e.g. Illingworth 2016; Paris *et al.* 2021). Here, the configuration is taken from Jussiau *et al.* (2022) and some details are recalled below. A cylinder of diameter $D$ is placed at the origin of a rectangular domain $\Omega$, equipped with the Cartesian coordinate system $(x_1, x_2)$. All quantities are rendered non-dimensional with respect to the cylinder diameter $D$ and the uniform upstream velocity magnitude $v_{1\infty}$. The convective time unit is defined as $t_c = D/v_{1\infty}$ and the Reynolds number is defined as $Re = v_{1\infty}D/\nu$, balancing convective and viscous terms. The domain extends as $-15 \leq x_1 \leq 20$, $|x_2| \leq 10$. The geometry is depicted in figure 4.

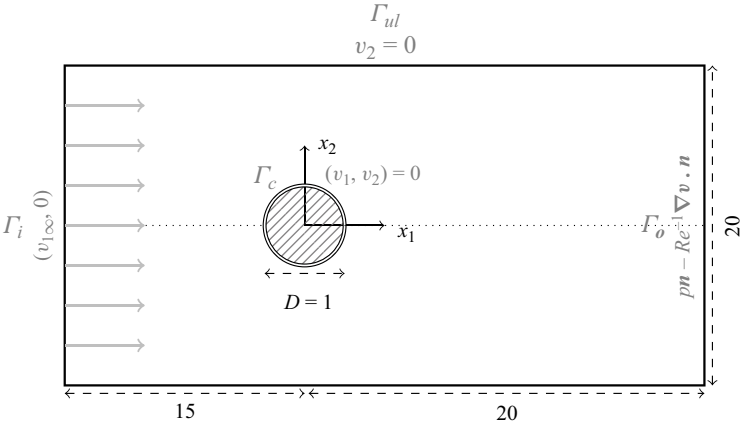



Figure 4. Domain geometry for the flow past a cylinder. Dimensions are in black, while boundary conditions are in light grey. Drawing is not to scale.

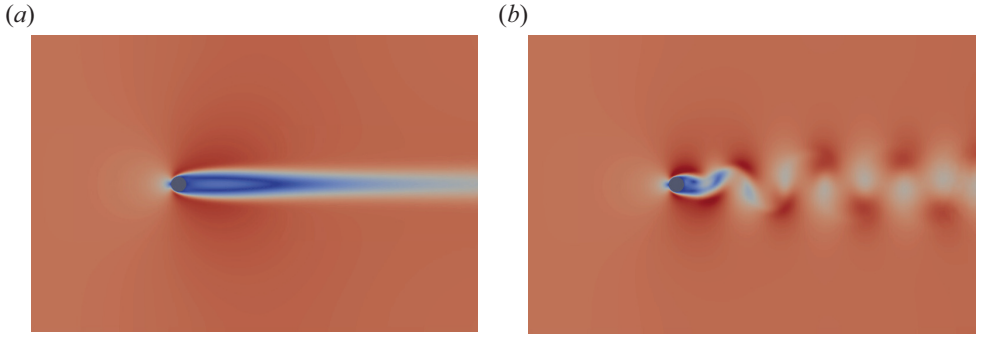



Figure 5. Cylinder flow regimes (velocity magnitude). Unstable base flow (*a*) and snapshot of the attractor (*b*). Domain is cut for clarity.

The flow is described by its velocity $\boldsymbol{v} = [v_1, v_2]^T$ and pressure $p$ in the domain $\Omega$, and satisfies the incompressible Navier–Stokes equations:

$$\begin{cases} \dfrac{\partial \boldsymbol{v}}{\partial t} + (\boldsymbol{v} \cdot \boldsymbol{\nabla})\boldsymbol{v} = -\boldsymbol{\nabla} p + \dfrac{1}{Re}\nabla^2 \boldsymbol{v}, \\ \boldsymbol{\nabla} \cdot \boldsymbol{v} = 0. \end{cases} \tag{2.3}$$

This dynamical system is known to undergo a supercritical Hopf bifurcation at the critical Reynolds number $Re_c \approx 47$ (Barkley 2006). Above the threshold, it displays an unstable equilibrium (here referred to as the base flow; see figure 5*a*) and a periodic attractor (i.e. a stable limit cycle; see figure 5*b*). In this study, we set $Re = 100$.

### 2.3.2. *Boundary conditions, control and simulation set-ups*

*Unactuated flow.* A parallel flow enters from the left of the domain, directed to the right of the domain. The boundary conditions for the unforced flow, represented in figure 4, are detailed below.

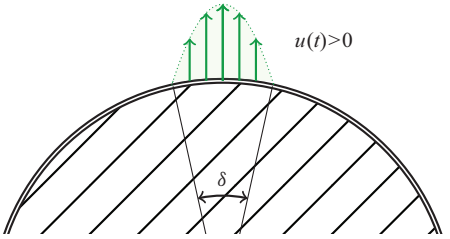



Figure 6. Zoom on actuation set-up.

(i) On the inlet $\Gamma_i$, the fluid has parallel horizontal velocity $\boldsymbol{v}^i = (v_{1\infty}, 0)$, uniform along the vertical axis $x_2$.
(ii) On the outlet $\Gamma_o$, we impose standard outflow conditions with $p^o\boldsymbol{n} - Re^{-1}\boldsymbol{\nabla}\boldsymbol{v}^o \cdot \boldsymbol{n} = 0$ where $\boldsymbol{n}$ is the outward-pointing vector.
(iii) On the upper and lower boundaries $\Gamma_{ul}$, that were set far from the cylinder to mitigate end effects, we impose an impermeability condition $v_2 = 0$.
(iv) On the surface of the cylinder $\Gamma_c$ where actuation is not active, we impose a no-slip condition with $\boldsymbol{v}^c = (0, 0)$.

*Actuation.* In this configuration, the actuation is injection/suction of fluid at the upper and lower poles of the cylinder, in the vertical direction. Both actuators are 10° wide and impose a parabolic profile $v_{2p}(x_1)$ to the normal velocity of the fluid, modulated by the control amplitude $u(t)$ (negative or positive). On the controlled boundaries, the boundary condition is $\boldsymbol{v}^{act}(x_1, t) = (0, v_{2p}(x_1)u(t))$ (see figure 6). Both actuators are functioning antisymmetrically, in order to maintain an instantaneous zero net mass flux. In other words, a positive actuation amplitude $u(t) > 0$ corresponds to blowing from the top and suction from the bottom, and conversely with $u(t) < 0$.

*Sensing.* It was shown in several past studies that a SISO (i.e. one actuator and one sensor) set-up can be adequate for controlling the cylinder configuration (Flinois & Morgans 2016; Jin *et al.* 2020; Jussiau *et al.* 2022) if the sensor is positioned in order to reconstruct sufficient information, and to not suffer too much from convective time delays. Following this trade-off, the sensor is chosen to provide the cross-stream velocity in the wake: $y(t) = v_2(x_1 = 3, x_2 = 0, t)$. The sensor is assumed perfect and not corrupted by noise. Note that including a linear sensor model (e.g. limited bandwidth with a low-pass transfer function) would be seamless, as the approach is entirely based on data and only assumes linearity.

Selecting the sensor location on the horizontal symmetry axis $x_2 = 0$ of the base flow yields an immediate benefit: the measurement value on the base flow can be deduced by symmetry arguments as $y_b = 0$. This has a direct advantage, allowing the controller to operate directly on the measurement value $y(t)$ while maintaining the natural base flow as an equilibrium point of the closed-loop system. In the case where the sensor were placed at a location with $y_b \neq 0$, two alternatives are suggested. The first option is the controller operating on the error signal $e(t) = y(t) - y_b$, requiring the computation of $y_b$ and $\boldsymbol{q}_b$, which is excluded in this data-driven approach. The second option is using a controller with zero static gain (i.e. $K(0) = 0$). In this case, it could operate directly with $y(t)$, while ensuring the base flow remains an equilibrium point, and no other equilibrium with zero input may exist, as proven in Leclercq *et al.* (2019).

2.3.3. *Numerical methods*

The incompressible Navier–Stokes equations in the two-dimensional domain (2.3) are solved in finite dimension with the finite element method using the toolbox FEniCS (Logg, Mardal & Wells 2012) in Python. The FEniCS toolbox is widely used for solving partial differential equations because of its very general framework and its native parallel computing capacity.

The unstructured mesh consists of approximately 25 000 triangles, more densely populated in the vicinity and in the wake of the cylinder. Finite elements are chosen as Taylor–Hood $(P_2, P_2, P_1)$ elements, leading to a discretized descriptor system of 113 000 states. For time stepping, a linear multistep method of second order is used. The nonlinear term is extrapolated with a second-order Adams–Bashforth scheme, while the viscous term is treated implicitly, making the temporal scheme semi-implicit. The time step is chosen as $\Delta t = 5 \times 10^{-3}$ for stability and precision. Each simulation is run in parallel on 24 CPU cores with distributed memory (MPI). All large-dimensional linear systems are solved with the software package MUMPS (Amestoy *et al.* 2000), a sparse direct solver well suited to distributed-memory architectures and natively accessible from within FEniCS.

2.3.4. *Additional notations for characterizing the flow*

For characterizing the flow globally, we define several operations and quantities that are reused in the following. First, the system (2.3) can be written as

$$\boldsymbol{E}\frac{\partial \boldsymbol{q}}{\partial t} = \boldsymbol{F}(\boldsymbol{q}), \tag{2.4}$$

where the state variable is defined as $\boldsymbol{q} = \left[\begin{smallmatrix} \boldsymbol{v} \\ p \end{smallmatrix}\right]$, $\boldsymbol{E} = \left[\begin{smallmatrix} \boldsymbol{I} & 0 \\ 0 & 0 \end{smallmatrix}\right]$ and the nonlinear operator $\boldsymbol{F}$ is expressed easily. The base flow denoted $\boldsymbol{q}_b$ refers to the unique steady equilibrium of (2.4) (Barkley 2006); we recall that it is linearly unstable. The model of the flow linearized around the base flow $\boldsymbol{q}_b$ (or indifferently a reduced-order approximation of said model) is denoted $G_b(s)$, and is used in the analysis of results in § 3. We also define the semi-inner product between two velocity–pressure fields $\boldsymbol{q} = \left[\begin{smallmatrix} \boldsymbol{v} \\ p \end{smallmatrix}\right]$ and $\widetilde{\boldsymbol{q}} = \left[\begin{smallmatrix} \widetilde{\boldsymbol{v}} \\ \widetilde{p} \end{smallmatrix}\right]$ as

$$\langle \boldsymbol{q}, \tilde{\boldsymbol{q}} \rangle_E = \int_\Omega \boldsymbol{q}^T \cdot \boldsymbol{E}\widetilde{\boldsymbol{q}}\, \mathrm{d}\Omega = \int_\Omega \boldsymbol{v}^T \cdot \widetilde{\boldsymbol{v}}\, \mathrm{d}\Omega. \tag{2.5}$$

In order to quantify the distance from a field $\boldsymbol{q}$ to the base flow $\boldsymbol{q}_b$, we define the field $\boldsymbol{\epsilon}(\boldsymbol{q}) = (\boldsymbol{q} - \boldsymbol{q}_b)^T \cdot \boldsymbol{E}(\boldsymbol{q} - \boldsymbol{q}_b)$, providing information on a local level. In turn, the scalar perturbation kinetic energy (PKE) associated with a velocity–pressure field $\boldsymbol{q}$ relative to the base flow $\boldsymbol{q}_b$ is defined as follows:

$$\mathcal{E}(\boldsymbol{q}) = \frac{1}{2}\int_\Omega \boldsymbol{\epsilon}(\boldsymbol{q})\, \mathrm{d}\Omega = \frac{1}{2}\|\boldsymbol{q} - \boldsymbol{q}_b\|_E^2. \tag{2.6}$$

While these quantities are only available in simulation, they are used *a posteriori* to quantify the distance to the base flow, i.e. the distance to convergence.

Finally, when the flow is in a periodic regime (e.g. in feedback with a given controller), we define the mean flow as the temporal average of the flow variables $\bar{\boldsymbol{q}} = \langle \boldsymbol{q}(t) \rangle_T$ over a period. It is the same quantity used in Leclercq *et al.* (2019) for iterative identification, control and analysis of the flow, and is used in §§ 3 and 4.

## 2.4. *Identification of an input–output model from data leveraging the mean resolvent*

### 2.4.1. *Introduction to the mean transfer function*

It is at first not obvious that the oscillating flow may be well approximated with an LTI model, moreover suitable for control purposes. Earlier justifications were given with variations of dynamic mode decomposition (Williams, Kevrekidis & Rowley 2015; Proctor, Brunton & Kutz 2016) whose theoretical link to the (linear) Koopman operator was established in Korda & Mezić (2018*b*), and the models were used for control in, for example, Korda & Mezić (2018*a*), Korda & Mezić (2018*b*) and Arbabi *et al.* (2018). In Leclercq & Sipp (2023), a new relevant model around the limit cycle is introduced, based upon observations from Dahan, Morgans & Lardeau (2012), Dalla Longa, Morgans & Dahan (2017) and Evstafyeva, Morgans & Dalla Longa (2017). This model, the so-called mean resolvent, is rooted in Floquet analysis and aims at providing the average linear response of the flow to a control input. It is also shown to be linked to the Koopman operator (Leclercq & Sipp 2023), and is extended to more complex attractors. This framework is briefly described in the rest of this section.

First, the linear response of the flow refers to the response of the flow to a given control input $\boldsymbol{f}(t)$ of small amplitude, i.e. small enough to allow linearization around the periodic deterministic unforced trajectory. Using a perturbation formulation, if the periodic unforced trajectory is denoted $\boldsymbol{Q}(t)$, the linear response to the control $\boldsymbol{f}(t)$ is $\delta\boldsymbol{q}(t)$, such that $\boldsymbol{q}(t) = \boldsymbol{Q}(t) + \delta\boldsymbol{q}(t)$.

Second, the average response of the flow is considered with respect to the phase on the limit cycle when the input $\boldsymbol{f}(t)$ is injected. On a periodic attractor of pulsation $\omega$, any time instant $t$ is parametrized by a phase $\phi = \omega t \mod 2\pi \in [0, 2\pi)$, so that every point is described by its associated $\phi$. The mean resolvent $\boldsymbol{R}_0(s)$ is the operator predicting, in the frequency domain, the average linear response (over $\phi$) from a given input: $\langle \delta\boldsymbol{q}(s;\phi)\rangle_\phi = \boldsymbol{R}_0(s)\boldsymbol{f}(s)$.

Here, we focus on a SISO transfer in the flow, i.e. the transfer between a single localized actuator and a single sensor signal. The actuation signal is such that $\boldsymbol{f}(t) = \boldsymbol{B}u(t)$ and we define the measurement deviation from the limit cycle as $\delta y(t) = \boldsymbol{C}\delta\boldsymbol{q}(t)$, where $\boldsymbol{B}$, $\boldsymbol{C}$ are actuation and measurement fields, depending on the configuration. In this case, we study the SISO mean transfer function $H_0(s) : u(s) \mapsto \langle \delta y(s;\phi)\rangle_\phi$, equal to $H_0(s) = \boldsymbol{C}\boldsymbol{R}_0(s)\boldsymbol{B}$. It is shown in the following that it is possible to identify $H_0(s)$ from data, with the full measurement $y(t) = \boldsymbol{C}\boldsymbol{q}(t)$ (since $\delta y(t)$ is not measurable in practice).

### 2.4.2. *Mean frequency response*

The identification of $H_0(s)$ is done in two steps. First, the frequency response $H_0(\jmath\omega)$ is extracted from input–output data on a discrete grid of frequencies. Second, a low-order state-space model is identified from these data. The transfer function of the low-order model is denoted $G(s)$.

*Multisine excitations.* In order to extract the frequency response $H_0(\jmath\omega)$, we use here a particular class of input signals known as multisine excitations (Schoukens *et al.* 1991). As shown in Leclercq & Sipp (2023), any class of input signals could be used for this task (e.g. white noise, chirp, etc.), with various efficiency and a potential need for ensemble averaging. A multisine realization is a superposition of sines, depending on a random vector of independent identically distributed uniform variables $\boldsymbol{\Phi} = [\Phi_1, \ldots, \Phi_N] \sim \mathcal{U}([0, 2\pi]^N)$:

$$u_{\boldsymbol{\Phi}}(t) = \frac{2}{\sqrt{N}} \sum_{k=1}^{N} A_k \sin(k\omega_u t + \Phi_k). \tag{2.7}$$

The fundamental frequency of the multisine is $\omega_u$ and only harmonics $k\omega_u$, $1 \leq k \leq N$, are included, each with chosen amplitude $A_k$ (normalized by $\frac{1}{2}\sqrt{N}$) and random phase $\Phi_k$. The numerical values of the parameters $\omega_u, N, A_k$ are given in § 3.1.2. Multisines have been chosen for their deterministic amplitude spectrum and sparse representation in the frequency domain (Schoukens *et al.* 2008; Schoukens, Vaes & Pintelon 2016), but any other input signal could be used for the identification in this context.

*Average of frequency responses and convergence to the mean frequency response.* For a given input $u_{\boldsymbol{\Phi}}(t)$, we denote $y_{\boldsymbol{\Phi}}(t) = y(t) + \delta y_{\boldsymbol{\Phi}}(t)$ the measured output, which is by definition the sum of the measurement signal of the unforced flow $y(t)$, and the forced contribution $\delta y_{\boldsymbol{\Phi}}(t)$ linear with respect to $u_{\boldsymbol{\Phi}}(t)$. Following Leclercq & Sipp (2023), the Fourier coefficients of the input and output at the forcing frequencies $k\omega_u$ may be identified with harmonic averages (denoting the imaginary unit $\jmath$):

$$\begin{cases} \hat{u}_{\boldsymbol{\Phi}}(k\omega_u) = \lim\limits_{T'\to\infty} \dfrac{1}{T'} \displaystyle\int_0^{T'} u_{\boldsymbol{\Phi}}(t)\, \mathrm{e}^{-\jmath k\omega_u t}\, \mathrm{d}t, \\ \hat{y}_{\boldsymbol{\Phi}}(k\omega_u) = \lim\limits_{T'\to\infty} \dfrac{1}{T'} \displaystyle\int_0^{T'} y_{\boldsymbol{\Phi}}(t)\, \mathrm{e}^{-\jmath k\omega_u t}\, \mathrm{d}t, \end{cases} \tag{2.8}$$

which are approximated in practice with discrete Fourier transforms (DFTs) (see § 3.1.2 and Appendix A). Also, as the forcing frequency $k\omega_u$ is chosen outside the frequency support of $y(t)$ we have

$$\widehat{\delta y}_{\boldsymbol{\Phi}}(k\omega_u) = \hat{y}_{\boldsymbol{\Phi}}(k\omega_u). \tag{2.9}$$

This is particularly important since we wish to identify the transfer between the input and the linear part of the output, which is not easily measurable in practice. Then, a frequency response depending on the phase $\boldsymbol{\Phi}$ of the input may be computed as a ratio of Fourier coefficients at forced frequencies:

$$H_{\boldsymbol{\Phi}}(\jmath k\omega_u) = \frac{\widehat{\delta y}_{\boldsymbol{\Phi}}(k\omega_u)}{\hat{u}_{\boldsymbol{\Phi}}(k\omega_u)}. \tag{2.10}$$

Now, using the expression of $y_{\boldsymbol{\Phi}}(t)$ deduced from Leclercq & Sipp (2023), it can be shown that

$$\mathbb{E}(H_{\boldsymbol{\Phi}}(\jmath k\omega_u)) = H_0(\jmath k\omega_u). \tag{2.11}$$

Therefore, if the experiment is repeated over $M$ realizations of $u_{\boldsymbol{\Phi}}(t)$ (i.e. by sampling $\boldsymbol{\Phi}$), then the sample mean $\bar{H}_{\boldsymbol{\Phi}}$ of $H_{\boldsymbol{\Phi}}$ approximates $H_0$ with variance $\mathrm{Var}(\bar{H}_{\boldsymbol{\Phi}}(\jmath k\omega_u)) = (1/M)\mathrm{Var}(H_{\boldsymbol{\Phi}}(\jmath k\omega_u))$. It is notable that the ensemble average is done here on the multisine phase $\boldsymbol{\Phi}$, and not the phase $\phi$ on the limit cycle where the signal $u_{\boldsymbol{\Phi}}(t)$ is injected (which was done in Leclercq & Sipp (2023)). Here, the phase on the limit cycle $\phi$ is assumed constant. In an experiment where $\phi$ cannot be chosen, the ensemble average would rather be performed on $\phi$ only, maintaining $\boldsymbol{\Phi}$ constant (i.e. injecting the same input signal at different instants on the limit cycle). As such, it would be possible to obtain a sample average of $\mathbb{E}(H_{\phi}(\jmath k\omega_u)) = H_0(\jmath k\omega_u)$.

### 2.4.3. *System identification*

Now that $H_0(\jmath\omega)$ has been sampled on a grid of $\omega$, we wish to find a low-order state-space representation of the transfer function $G(s)$ approximating the unknown $H_0(s)$, accessible to control techniques. This task is performed with a subspace-based frequency

identification method, but could be performed with any other frequency identification method, e.g. the eigensystem realization algorithm in frequency domain (Juang & Suzuki 1988) or vector fitting (Gustavsen & Semlyen 1999; Ozdemir & Gumussoy 2017). Subspace methods form a class of blackbox linear identification methods that do not rely on nonlinear optimization as most iterative model-fitting methods do. Here, the frequency observability range space extraction (FORSE) (Liu, Jacques & Miller 1994) estimates a discrete-time state-space model with order fixed *a priori*, in two distinct steps. Matrices $\boldsymbol{A}$ and $\boldsymbol{C}$ are built directly from a singular value decomposition of a Hankel matrix built from the frequency response. Matrices $\boldsymbol{B}$ and $\boldsymbol{D}$ are then found by solving a linear least squares problem. More details can be found in Liu *et al.* (1994) or in Appendix B for a SISO version. Additional stability constraints may be enforced with linear matrix inequalities (Demourant & Poussot-Vassal 2017), as the transfers $H_0(s)$, $G(s)$ are expected to exhibit some marginally stable poles in this context.

For the sake of rendering the procedure as automatic as possible, the order of the model identified at each iteration, denoted $n_G = \partial^\circ G$, is fixed. The choice of $n_G$ is discussed in § 3.1.2.

### 2.5. *Control of the flow using the mean transfer function*

#### 2.5.1. *Rationale*

After we have determined an LTI model $G(s)$ of the fluid flow around its attractor, we wish to control it in order to reduce the self-sustained oscillations. Among the classic control methods such as pole placement, LQG, $\mathcal{H}_\infty$ techniques (e.g. mixed-sensitivity, loop-shaping, structured $\mathcal{H}_\infty$, etc.) and MPC, we choose the LQG framework for synthesis. It combines several advantages, such as being easy to automate, having predictable controller gain to some extent and producing a controller with relatively low complexity.

#### 2.5.2. *Principle of observed-state feedback*

Linear quadratic Gaussian control is very popular in flow control (see e.g. Schmid & Sipp (2016), Barbagallo *et al.* (2009), Kim & Bewley (2007), Carini, Pralits & Luchini (2015) and Brunton & Noack (2015) and references therein) due to its ease of use. The principle and main equations are recalled in Appendix C. Here, we simply recall that the dynamic LQG controller for a SISO plant $G = \left[\begin{array}{c|c} \boldsymbol{A} & \boldsymbol{B} \\ \hline \boldsymbol{C} & \boldsymbol{0} \end{array}\right]$ can be calculated from the observer gain $\boldsymbol{L}$ (depending on state noise and output noise covariances $\boldsymbol{W}$, $V$) and the state-feedback gain $\boldsymbol{K}$ (depending on state and input costs $\boldsymbol{Q}$, $R$) and expressed as follows:

$$K_{LQG} = \left[\begin{array}{c|c} \boldsymbol{A} + \boldsymbol{B}\boldsymbol{K} + \boldsymbol{L}^{\boldsymbol{T}}\boldsymbol{C} & -\boldsymbol{L}^{\boldsymbol{T}} \\ \hline \boldsymbol{K} & \boldsymbol{0} \end{array}\right]. \tag{2.12}$$

We provide important remarks on the LQG controller below.

(i) The state-feedback gain $\boldsymbol{K}$ and the observer gain $\boldsymbol{L}$ are computed independently. Additionally, each problem may be normalized. For the state feedback, the state weighting is chosen as $\boldsymbol{Q} = \boldsymbol{I}_n$ and the problem is only parametrized by the value $R > 0$ penalizing the control input. Symmetrically, for the estimation problem, we choose $\boldsymbol{W} = \boldsymbol{I}_n$ and the problem is parametrized by $V > 0$. This approach is more conservative because states are weighted equally, but allows for parametrizing the LQG problem easily with only two positive scalars $R, V > 0$. In the following,

a controller stemming from a LQG synthesis with weightings $R, V$ is denoted $K_{LQG}(R, V)$.

(ii) The matrix weights may be tuned to enforce specific behaviours for the solution controller (which was already underlined in Sipp & Schmid 2016). For the state-feedback problem, one can prioritize small control inputs ($R \to \infty$, low gain $\boldsymbol{K}$) or reactive control ($R \to 0$, large gain $\boldsymbol{K}$). Symmetrically, for the estimation problem, the choice is made between slow estimation ($V \to \infty$, low gain $\boldsymbol{L}$) or fast estimation ($V \to 0$, large gain $\boldsymbol{L}$).

(iii) While linear quadratic regulator controllers exhibit inherent robustness (in gain and phase margins) given diagonal $\boldsymbol{Q}, \boldsymbol{R}$ (Lehtomaki, Sandell & Athans 1981), these guarantees generally do not hold for LQG and stability margins may be arbitrarily small (Doyle 1978) but can be checked *a posteriori*.

### 2.6. *Controller stacking, balanced truncation and state initialization*

#### 2.6.1. *Rationale*

At the beginning of iteration $i + 1$ of the procedure, a new controller $K_i^+$ is synthesized and coupled to the flow, which is already in closed loop with the control law $K_i$. The new controller operating in the loop would be

$$K_{i+1} = K_i + K_i^+. \tag{2.13}$$

In the general case, while the newly designed controller $K_i^+$ has manageable order ($\partial^\circ K_i^+ = \partial^\circ G_i = n_G$), the total controller $K_{i+1}$ has order $\partial^\circ K_{i+1} = \partial^\circ K_i + \partial^\circ K_i^+ = (i - 1)n_G + n_G$ that is increasing linearly with iterations.

In order to reduce the order of the controller, we resort, at each iteration, to balanced truncation of the controller operating in the loop. In other words, instead of using the full controller $K_{i+1} = K_i + K_i^+$ in the loop, we use a reduced-order version $\tilde{K}_{i+1}$. Repeating the operation at each iteration leads to

$$\tilde{K}_{i+1} = \mathcal{B}_T(\tilde{K}_i + K_i^+), \tag{2.14}$$

where the operation $\mathcal{B}_T$ refers to the balanced truncation described below, enabling order reduction: $\partial^\circ[\mathcal{B}_T(K)] \leq \partial^\circ K$. This operation makes state initialization of the new controller more challenging, which is tackled in the following as well.

#### 2.6.2. *Balanced truncation*

Balanced truncation was already introduced in previous flow control articles (Rowley 2005; Kim & Bewley 2007) with the intent to reduce the order of flow models. As the traditional balanced truncation algorithm introduced in Moore (1981) is not applicable to high-dimensional models of dimension $O(10^5)$ and higher, it has led to the development of various approximate techniques. However, the objective is different here: order reduction is performed on the controller which initially has moderate dimension $O(10)$, enabling direct balanced truncation methods (see Gugercin & Antoulas (2004) for an in-depth survey).

More specifically, given a controller $K$ of order $\partial^\circ K = n_K$, we wish to find a reduced-order controller $\tilde{K} = \mathcal{B}_T(K)$ of order $\partial^\circ \tilde{K} = \tilde{n}_K < n_K$ such that the controllers $K, \tilde{K}$ have similar behaviour, quantified as the $\mathcal{H}_\infty$ norm of the difference. This is done by first computing a balanced realization of $K$ (Moore 1981; Laub *et al.* 1987), then truncating the balanced modes of $K$ with the lowest controllability and observability, quantified by its

Hankel singular values (HSVs), denoted $\sigma_j, j \in \{1, \ldots, n_K\}$ (Pernebo & Silverman 1982). For a truncation to order $\tilde{n}_K$, the reduction error is bounded explicitly (Enns 1984):

$$\|K(s) - \tilde{K}(s)\|_{\mathcal{H}_\infty} \leq 2 \sum_{j=\tilde{n}_K+1}^{n_K} \sigma_j. \tag{2.15}$$

If $K$ has unstable modes (which cannot be prevented in the LQG framework), a possibility is to separate $K$ into unstable and stable contributions $K = K_u + K_s$, and perform the reduction on $K_s$ only (Zhou *et al.* 1999).

### 2.6.3. *Controller switching and state initialization*

When adding a new controller to the flow, careful state initialization is needed in order to reduce transients in the control input (e.g. so as to not saturate the actuator in a real-life set-up or to not trigger nonlinear effects). Many techniques from the control literature address smooth switching, either with bumpless methods (usually requiring a precise full plant model; Zaccarian & Teel 2005), by focusing on fast controller transients (Cheong & Safonov 2008; see also references therein for a brief overview), or by using past data for initializing the controller (Paxman 2004). Although these techniques are very attractive, they did not seem to provide consistent performance on our study case. The explanation might come from the fact that the flow model is very crude by definition: nonlinearity is neglected, and the time-dependent variations of the input–output transfers are averaged in the mean transfer function. We present below a basic two-step method that proved to be consistent and efficient in this study.

The controller initialization and switching are done in two steps: first, basic state initialization for the full-order controller in order to reach a new dynamical equilibrium with moderate transient; then, state initialization of the reduced controller based on the past transient signal, ensuring seamless transition between the high-order and low-order controllers. For simplicity of notations, we redefine time with respect to the current iteration: $t = 0$ is the time instant where the full-order controller $K_{i+1}$ is inserted, and $t = T_{sw}$ is the instant where $K_{i+1}$ and its reduced-order counterpart $\tilde{K}_{i+1} = \mathcal{B}_T(K_{i+1})$ are exchanged.

*First step: full-order controller state initialization.* Without controller order reduction, the new controller $K_i^+$ is added to the flow with internal state $\boldsymbol{x}^0_{K_i^+} = 0$. The current internal state $\boldsymbol{x}_{\tilde{K}_i}$ of the closed-loop controller $\tilde{K}_i$ is untouched, so that the stacked controller $K_{i+1}$ has initial internal state

$$\boldsymbol{x}^0_{K_{i+1}} = \begin{bmatrix} \boldsymbol{x}_{\tilde{K}_i} \\ 0 \end{bmatrix}, \tag{2.16}$$

as in Leclercq *et al.* (2019). Initialized as such, the controller in closed loop generally produces a control input $u(t)$ and a corresponding measurement signal $y(t)$ with moderate transient amplitude, because of the choice of LQG weightings detailed in § 3.1.3. Once the transient term related to controller state initialization has decayed, after a duration $T_{sw}$, we perform a seamless switch to the reduced-order controller.

*Second step: reduced-order controller state initialization and switch.* Just as the previous step, this step is still a controller initialization problem: How should the state of the reduced-order controller $\boldsymbol{x}_{\tilde{K}_{i+1}}$ be set in order to produce small control transient when exchanging controllers in the loop at time $t = T_{sw}$? To solve this, we exploit the fact

that both controllers produce close outputs $u(t)$, $\tilde{u}(t)$ under the same history of input signal $y(\tau)$, $0 \leq \tau \leq t$. Indeed, by definition of the $\mathcal{H}_\infty$ norm, for $\|y(t)\|_2 > 0$,

$$\frac{\|u(t) - \tilde{u}(t)\|_2}{\|y(t)\|_2} \leq \|K_{i+1}(s) - \tilde{K}_{i+1}(s)\|_{\mathcal{H}_\infty}, \tag{2.17}$$

where $\|K_{i+1}(s) - \tilde{K}_{i+1}(s)\|_{\mathcal{H}_\infty}$ is bounded from (2.15) and expected to be small due to the truncation of balanced modes with small HSVs. As a result, $u$ and $\tilde{u}$ may be expected to be close under the same input $y$. Then, if we denote the full-order controller $K_{i+1} = \left[\begin{array}{c|c} \boldsymbol{A_K} & \boldsymbol{B_K} \\ \hline \boldsymbol{C_K} & \boldsymbol{0} \end{array}\right]$, we may write its output as

$$u(t) = \boldsymbol{C_K}\, \mathrm{e}^{\boldsymbol{A_K} t} \boldsymbol{x}^0_{K_{i+1}} + \int_0^t \boldsymbol{C_K}\, \mathrm{e}^{\boldsymbol{A_K}(t-\tau)} \boldsymbol{B_K} y(\tau)\, \mathrm{d}\tau, \tag{2.18}$$

where $y$, arising from the closed loop, can be considered an external signal to the controller. If the reduced-order controller $\tilde{K}_{i+1} = \left[\begin{array}{c|c} \tilde{\boldsymbol{A}}_{\boldsymbol{K}} & \tilde{\boldsymbol{B}}_{\boldsymbol{K}} \\ \hline \tilde{\boldsymbol{C}}_{\boldsymbol{K}} & \boldsymbol{0} \end{array}\right]$ were fed the same signal $y$, but only $u$ were fed back to the flow (i.e. not $\tilde{u}$), then its output signal would be equivalently

$$\tilde{u}(t) = \tilde{\boldsymbol{C}}_{\boldsymbol{K}}\, \mathrm{e}^{\tilde{\boldsymbol{A}}_{\boldsymbol{K}} t} \boldsymbol{x}^0_{\tilde{K}_{i+1}} + \int_0^t \tilde{\boldsymbol{C}}_{\boldsymbol{K}}\, \mathrm{e}^{\tilde{\boldsymbol{A}}_{\boldsymbol{K}}(t-\tau)} \tilde{\boldsymbol{B}}_{\boldsymbol{K}} y(\tau)\, \mathrm{d}\tau. \tag{2.19}$$

Assuming $T_{sw}$ is chosen such that the initial condition contribution becomes negligible, then the state of the reduced-order controller producing the output (2.19), defined as $\tilde{u}(T_{sw}) = \tilde{\boldsymbol{C}}_{\boldsymbol{K}} \boldsymbol{x}_{\tilde{K}_{i+1}}(T_{sw})$, is

$$\boldsymbol{x}_{\tilde{K}_{i+1}}(T_{sw}) = \int_0^{T_{sw}} \mathrm{e}^{\tilde{\boldsymbol{A}}_{\boldsymbol{K}}(T_{sw}-\tau)} \tilde{\boldsymbol{B}}_{\boldsymbol{K}} y(\tau)\, \mathrm{d}\tau. \tag{2.20}$$

Therefore, the two controllers are switched at time $t = T_{sw}$, by setting the state of the reduced-order controller as per (2.20). More importantly, the closed-loop behaviour of the measurement signal $y$ for $\tau \geq T_{sw}$ is expected to remain similar since the two controllers are designed to show similar input-output behaviour. In this study, we have set $T_{sw} = 50$ which is usually larger than the characteristic time scales of the controllers (computed from the eigenvalues of $\boldsymbol{A_K}$, $\tilde{\boldsymbol{A}}_{\boldsymbol{K}}$).

*Illustration.* The two-step switching process is illustrated in figure 7, with data from § 3.2.4. For the first step, the signal $u(t)$ (solid blue) generated by the full-order controller $K_{i+1}$ is used in closed loop for $t < T_{sw}$, with initial state $\boldsymbol{x}^0_{K_{i+1}}$ as per (2.16). For the second step, for $t \geq T_{sw}$ the signal $\tilde{u}(t)$ (solid red) generated by the reduced-order controller $\tilde{K}_{i+1}$ is used in place of $u(t)$, by setting its internal state as per (2.20). For representation purposes, we compute and represent the signals $\tilde{u}(t)$, $t < T_{sw}$ and $u(t)$, $t \geq T_{sw}$ as dashed red and blue lines, respectively. In practice, they need not be computed. Figure 7 confirms the benefit of this switching procedure: the transient regime is moderate and the transition from the full-order to the reduced-order controllers is almost seamless.

## 3. Results: driving the flow from the limit cycle to the base flow

### 3.1. *Unforced flow: a test-bed for choosing parameters*

The first iteration for the identification-control procedure starts at time $t_1 = 500$, when the unforced flow is fully developed, as no controller is in the loop yet. The unforced fully

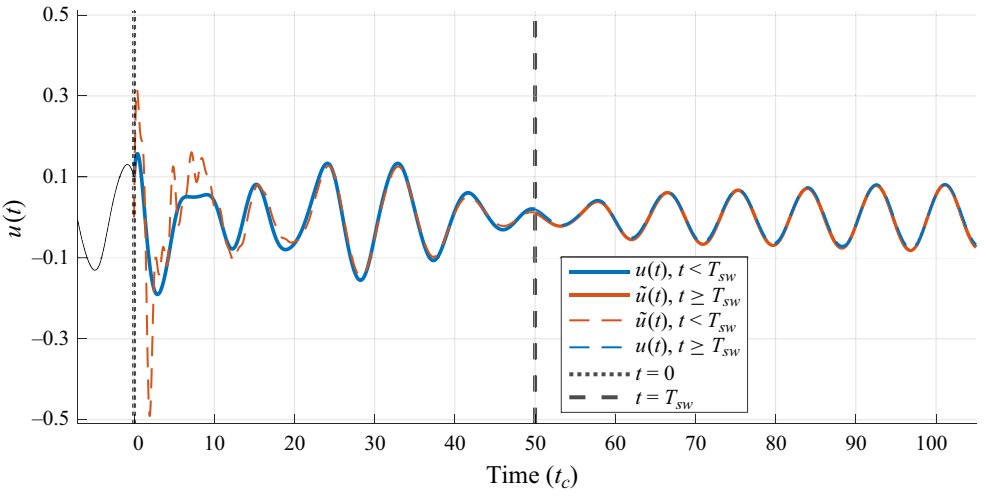



Figure 7. Controller initialization and switching procedure. The control signal used is first $u(t), t < T_{sw}$ (solid blue) generated by the full-order controller, initialized as (2.16); then, it is switched to $\tilde{u}(t), t \geq T_{sw}$ (solid red) generated by the reduced-order controller, initialized as (2.20). The dashed signals need not be computed in practice, but are represented nonetheless.

developed flow at $t_1 = 500$ is exploited in order to evaluate the sensitivity of the method to its parameters.

### 3.1.1. *Unstable equilibrium and unforced flow*

To compute the fully developed regime of self-sustained oscillations, the flow initially on its unstable equilibrium is perturbed infinitesimally to leave the equilibrium. The fundamental oscillation frequency of the flow continuously shifts from the frequency of the unstable pole of the flow linearized around the equilibrium $G_b(s) : \omega_b = 0.779\,\mathrm{rad}/t_c$ to the frequency of the limit cycle $\omega_0 = 1.062\,\mathrm{rad}/t_c$ on the attractor (see spectrogram in figure 8). Also, it is notable that almost no high-order harmonics are visible at small time instants near the equilibrium, which is a sign of the complex exponential divergence; while several higher-order harmonics appear due to the nonlinearity at greater time instants. This phenomenon is also more visible in the far wake. Note that by half-wave symmetry of the unforced signal $y(t)$ (i.e. $y(t) = -y(t + T/2)$), only odd harmonics appear in its frequency representation. This symmetry of the cross-stream velocity component on the symmetry axis of the geometry is justified in Barkley, Tuckerman & Golubitsky (2000) by the alternation of vortex shedding from the upper and lower mixing layers over a period.

### 3.1.2. *Identification: multisine design and ROM*

*Design of input signal.* The first step of the iterative process is identifying the mean transfer function of the flow thanks to multisine excitations, as per § 2.4. The fundamental pulsation $\omega_u$, that enables gathering the frequency response at pulsations $k\omega_u$, is chosen as $\omega_u = 2\pi \times 10^{-2}$, providing fine enough sampling of the frequency response, especially near resonant modes. Consequently, the input signal is periodic with period $T_u = 100$. The $N = 5000$ frequencies included in the input signal are such that the highest frequency is $N\omega_u = 2\pi \times \frac{1}{4} f_s$ (where $f_s = 200$ is the sampling frequency), in order to

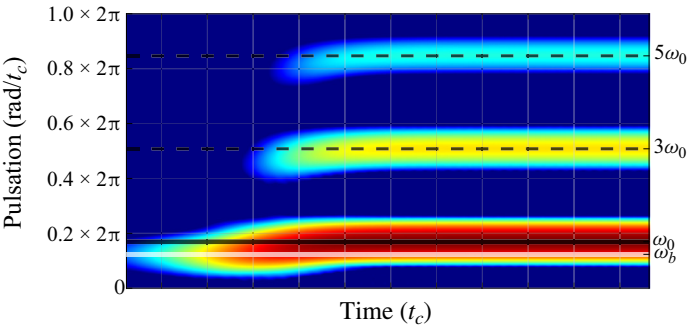



Figure 8. Spectrogram of cross-stream velocity probe at $x_1 = 3, x_2 = 0$ (used for feedback), flow trajectory from unstable equilibrium to natural stable limit cycle. The pulsation continuously shifts from $\omega_b$ to $\omega_0$, and higher-order odd harmonics gradually appear.

cut high-frequency harmonics ($f > \frac{1}{4} f_s$) that are weakly amplified by the flow, therefore not useful for identification or control.

For the first iteration, we choose the amplitude of the input $u_\Phi(t)$ as $A = 10^{-3}$ to guarantee small perturbation, i.e. $\|\delta y_\Phi(t)\|_\infty \ll 1$. However, the closer the system is to the equilibrium (i.e. at higher iterations of the procedure), the more $A$ is reduced because the relative impact of forcing increases. Note that this would need to be taken into account in an experiment, where the output noise may be important, so there is a balance to strike between small input $u_\Phi(t)$ and convenient signal-to-noise ratio.

The mean frequency response $\bar{H}_\Phi(\jmath k\omega_u)$ is estimated by averaging $M = 4$ realizations of $H_\Phi(\jmath k\omega_u)$. The choice of the value for $M$ is justified in Appendix A, where we also provide some details on the estimation in practice. For each realization, the forced system is simulated on a duration $(P_{tr} + P)T_u$, with $P_{tr} = P = 4$. The first portion of the input and output signals of duration $P_{tr}T_u$ is discarded for containing the contribution of damped Floquet modes in the flow response (Leclercq & Sipp 2023). The second portion of duration $PT_u$ is utilized for estimating $H_\Phi(\jmath k\omega_u) = \hat{\delta y}_\Phi(k\omega_u)/\hat{u}_\Phi(k\omega_u)$ (2.10).

*Identified mean transfer function.* The mean frequency response and the associated identified low-order model $G_0$ at the first iteration are represented in figure 9 (as blue circles for the frequency response, and as a solid red line for $G_0$). The ROM shows good fit with respect to the frequency data at low frequency, and manages to recover the resonant poles as well as the linear slope of the phase in the pulsation range $\omega \in [1.1, 3]\,\mathrm{rad}/t_c$. For $\omega \geq 3$, the ROM does not match the frequency response as closely, which is especially visible on the phase plot. This is explained in Appendix A, suggesting the need for more realizations of $H_\Phi(\jmath k\omega_u)$ to converge the sample mean at higher frequencies.

*Order of the ROM.* In order to avoid overfitting the frequency response (which is not always fully converged due to a lack of data), the order is chosen as low as $n_G = 8$ at each iteration. Identifying greater orders tends to introduce spurious modes into the ROM, and keeping a low order proves to still capture the main features of the flow response (resonant modes and unstable zeros). The mean transfer function is expected to have poles on the imaginary axis (Leclercq & Sipp 2023), so the identification algorithm is tuned to enforce $\Re(p_0) \leq 0$ for all poles $p_0 \in \mathbb{C}$ thanks to linear matrix inequality constraints (Demourant & Poussot-Vassal 2017). Numerically, the poles are usually found in the strictly stable half-plane ($\Re(p_0) < 0$), but marginally unstable poles would not pose an issue for control in this case.

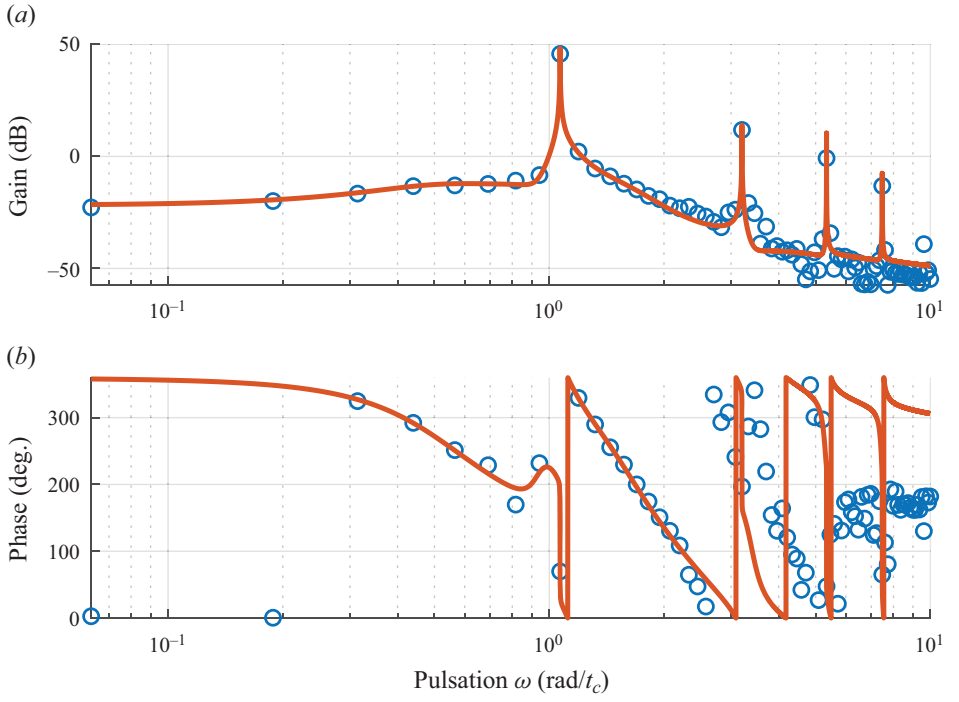



Figure 9. Mean frequency response (blue circles) and identified ROM $G_0$ (solid red line) at the first iteration.

### 3.1.3. *Control design: choice of LQG weights*

By construction, the mean transfer function ROM $G$ is valid for small control input (and weakly unsteady flow in the sense that the time-varying nature of the flow may be safely neglected from an input–output viewpoint), which is an incentive to design a controller with small gain. With LQG synthesis, it corresponds to the limit $R \to \infty$ (high penalization of control input, slow controller) and $V \to \infty$ (noisy measurement, slow observer). However, for infinitesimal control input, while the LTI approximation is valid, it is likely that the control produces imperceptible change in the flow. On the contrary, as the gain increases, modification of the flow should be more and more discernible, until the nonlinear phenomena become more present and the assumption of linearity fails.

In order to choose the more appropriate weights for the LQG controller, we coarsely mesh $R, V$, generate the associated controllers $K_{LQG}(R, V)$ and plug them in the fully developed flow. The flow in feedback with the control law $K_{LQG}(R, V)$ reaches a new dynamical equilibrium after some transient regime. A controller is deemed satisfactory if it achieves both of the following:

(i) Average PKE (defined in § 2.3.4) on the new dynamical equilibrium in closed loop $\mathcal{E}_1(R, V)$ lower than the natural limit cycle PKE $\mathcal{E}_0$. This criterion is conveyed by $\delta\mathcal{E}_1(R, V) = \mathcal{E}_1(R, V)/\mathcal{E}_0 - 1$. A map of $\delta\mathcal{E}_1(R, V)$ is shown in figure 10(*a*). Desirable control laws should yield $\delta\mathcal{E}_1(R, V) < 0$.

(ii) Moderate control input $u(t)$ in order to avoid potential actuator saturation, conveyed by the quantity $\max_t |u(t)|$, in figure 10(*b*).

The low-gain control region, with expensive control $R \to \infty$ and slow estimation $V \to \infty$, corresponds to the upper-right corners in figures 10(*a*) and 10(*b*). On the

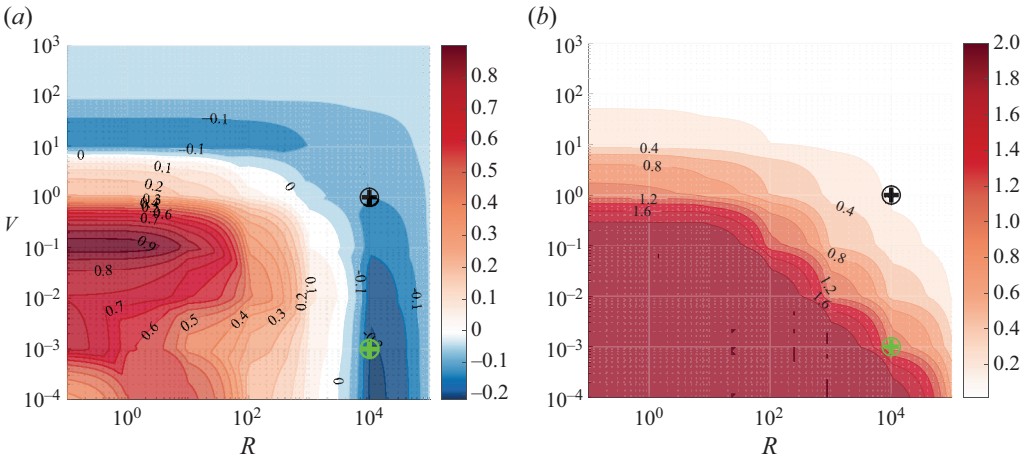



Figure 10. Meshing of parameters $R, V$ for LQG synthesis at the first iteration. A controller is deemed satisfactory when it achieves PKE reduction ($a$) and moderate control input transient ($b$). In this study, the chosen parameters are indicated as crosses (green for the first iteration, black for subsequent iterations). ($a$) The PKE criterion $\delta\mathcal{E}_1$ and ($b$) input signal peak $\max_t |u(t)|$.

contrary, the high-gain control region, with cheap control $R \to 0$ and fast estimation $V \to 0$, corresponds to the lower-left corners of the same figures. As expected, for low-gain controllers, almost no modification in the PKE is observed (upper-right corner in figure 10$a$). When increasing the gain (either decreasing $R$, or $V$, or both), energy reduction becomes increasingly more perceptible, until the performance ultimately degrades, most probably due to nonlinearity, corresponding to the limit of validity of the mean transfer function (lower-left corner in figure 10). It appears clearly that the low-gain limit (small PKE reduction) is achieved with either $R \to \infty$ (right-side boundary of figure 10$a$) corresponding to low control gains $\boldsymbol{K}$, or $V \to \infty$ (upper boundary of figure 10$a$) corresponding to low estimation gains $\boldsymbol{L}$. This limit may be thought of as a safe zone for choosing parameters (since a controller would likely not disturb the flow perceptibly) but not guaranteeing sufficient PKE reduction.

For the first iteration, assuming that the maximum value of the control input that can be implemented in simulation (corresponding to an actuator saturation in an experiment) is around $\max_t |u(t)| \approx 2$, values of $R$, $V$ are chosen as $R = 10^4$, $V = 10^{-3}$ (green cross) that offer PKE reduction of $\delta\mathcal{E}_1 \approx -20\,\%$ and moderate control input, without exceeding the arbitrary saturation. For subsequent iterations, parameters are chosen more conservatively because energy maps can be expected to vary slightly: $R = 10^4$, $V = 10^0$ (black cross).

### 3.1.4. *Controller reduction: choice of the balanced truncation threshold*

With $K$ in balanced form with HSVs sorted by decreasing magnitude, states with HSV $\sigma_j$ such that $\sigma_j/\sigma_1 < g_\sigma$ (with chosen threshold $g_\sigma$) are truncated. In our study case, $g_\sigma = 10^{-3}$ provides a good trade-off between maintaining performance and sufficient order reduction, usually such that $\partial^\circ K \leq 25$. Note that a fixed maximum number of states could be forced, with the risk of truncating important dynamics in the reduced-order controller.

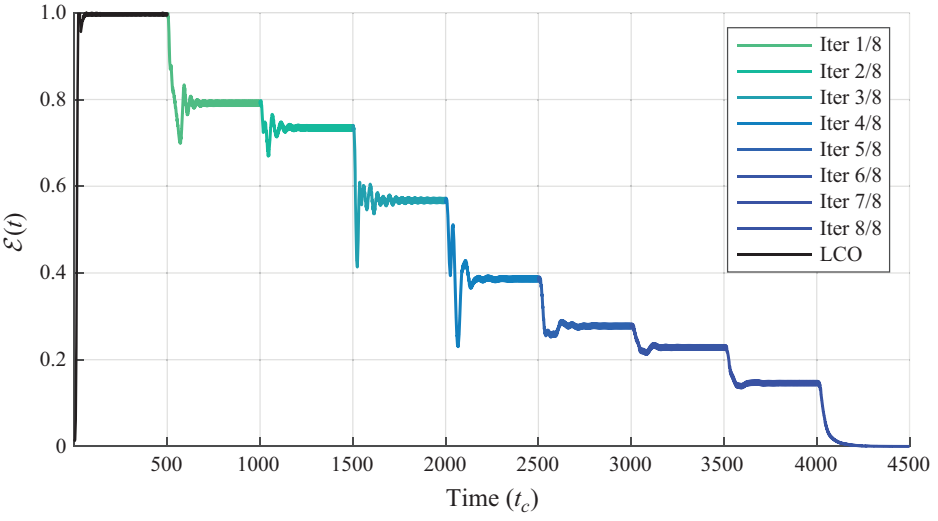



Figure 11. Plot of PKE $\mathcal{E}(t)$ throughout iterations.

### 3.2. *Main result: data-based convergence to equilibrium with piecewise LTI controller*

The main result of this study is the stabilization of the cylinder from its natural limit cycle to its unstable equilibrium, using only input–output data and LTI controllers, in a way that could be implemented in an experiment. In the results presented here, the stabilization took 8 iterations from the limit cycle in order to attain the equilibrium, stabilized by a closed-loop LTI controller.

#### 3.2.1. *Iterative convergence to the base flow*

The convergence to the base flow is shown through the PKE, defined in § 2.3.4. Although this quantity is not available outside the simulation realm, it is used as a posterior criterion of convergence to the base flow. At each iteration, it is observed in figure 11 that the PKE $\mathcal{E}$ decreases to reach a new dynamical equilibrium with time-averaged value $\langle \mathcal{E}(t) \rangle_t$ (normalized by the PKE on the attractor $\mathcal{E}_0$). At the last iteration starting at $t = 4000$, the flow is attracted to the base flow. We have $\lim_{t\to\infty} \mathcal{E}(t) = 0$, and the convergence is exponential: the system is stabilized around the base flow in closed loop in a linear sense. It is notable that the approach is based solely on input–output data, so the PKE would not be ensured to decrease at each iteration (and other solutions to the problem showed that convergence can be achieved without monotonic decrease of the average PKE at each iteration).

#### 3.2.2. *Probing the wake*

The full stabilization of the flow is confirmed by inserting probes in the flow (not used for feedback), on the symmetry axis $x_2 = 0$, at $x_1 = 1, 2$ (upstream of the feedback sensor) and $x_1 = 5, 7, 10$ (downstream of the feedback sensor). Additionally, it enables tracing profiles of convergence of the cylinder wake: in figure 12, we compute the RMS value of the signal at each probe location, normalized by the RMS of the same probe in the unforced flow. Closer sensor signals ($x_1 \leq 3$) have normalized RMS decay very rapidly and reach 10 % as soon as iteration 4, while for downstream probes, the signal reduces very mildly at first. It corresponds to the incremental lengthening of the recirculation bubble behind

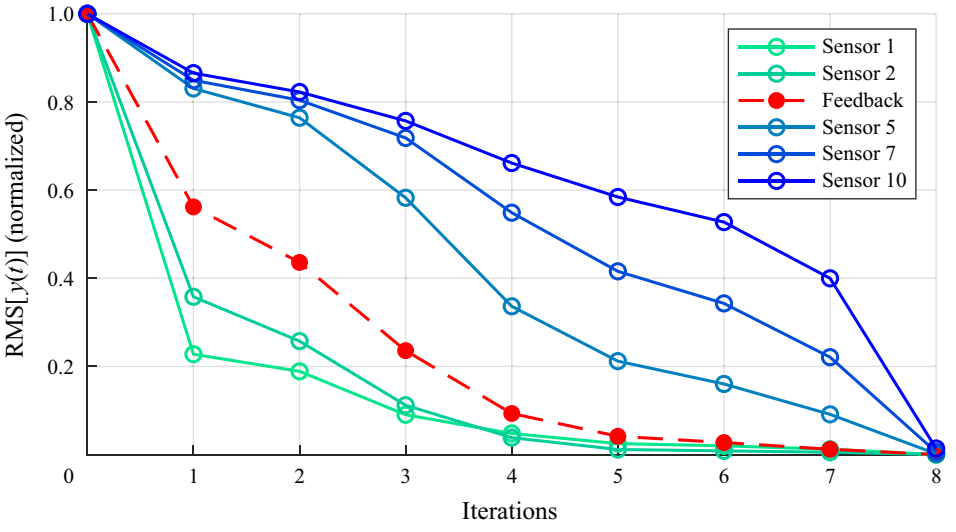



Figure 12. Normalized RMS of measured signals (performance sensors) depending on their position in the wake, throughout iterations.

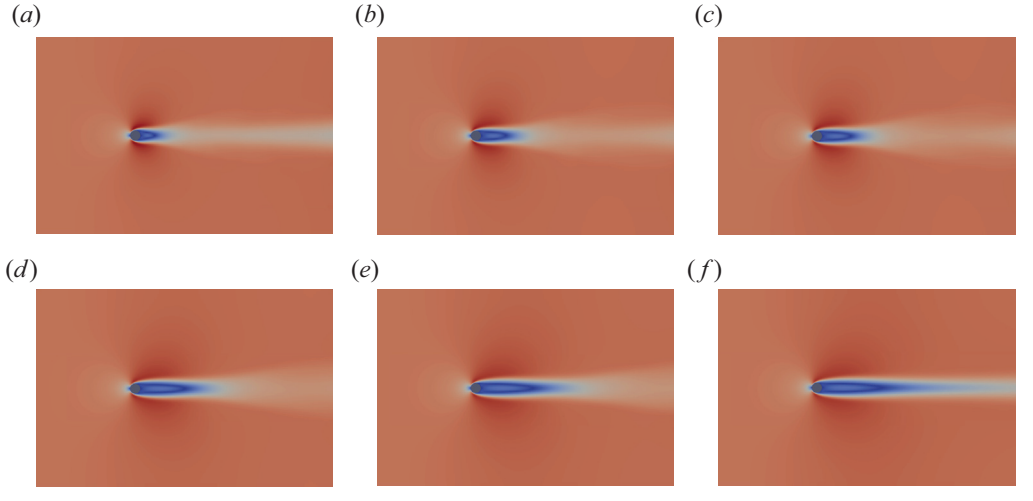



Figure 13. Mean flow (velocity magnitude) $\bar{\boldsymbol{q}} = \langle \boldsymbol{q}(t) \rangle_t$ at the end of iterations 0 (uncontrolled, *a*), 1, 2, 4, 6, 8 (last iteration, $\bar{\boldsymbol{q}} = \boldsymbol{q}_b$, *f*). Colour scaling is constant.

the cylinder, so that closer sensors quickly lie in the mean recirculation bubble, while downstream sensors still probe a strongly oscillating wake. It would be interesting to apply the procedure with various feedback sensor positions, in order to identify the zones where the feedback sensor is the most effective in a nonlinear setting.

The recirculation bubble of the mean flow is defined as the zone behind the cylinder, inside the contour delimited by $\boldsymbol{v}(x_1, x_2) = 0$ on the mean flow $\bar{\boldsymbol{q}} = \langle \boldsymbol{q}(t) \rangle_t$ (figure 13). The recirculation bubble lengthens almost linearly throughout iterations, as the flow is being stabilized, until it finally reaches the base flow.

Throughout iterations, the spatial distribution of PKE evolves in a manner similar to the mean recirculation bubble. It is quantified with the mean flow deviation from the base

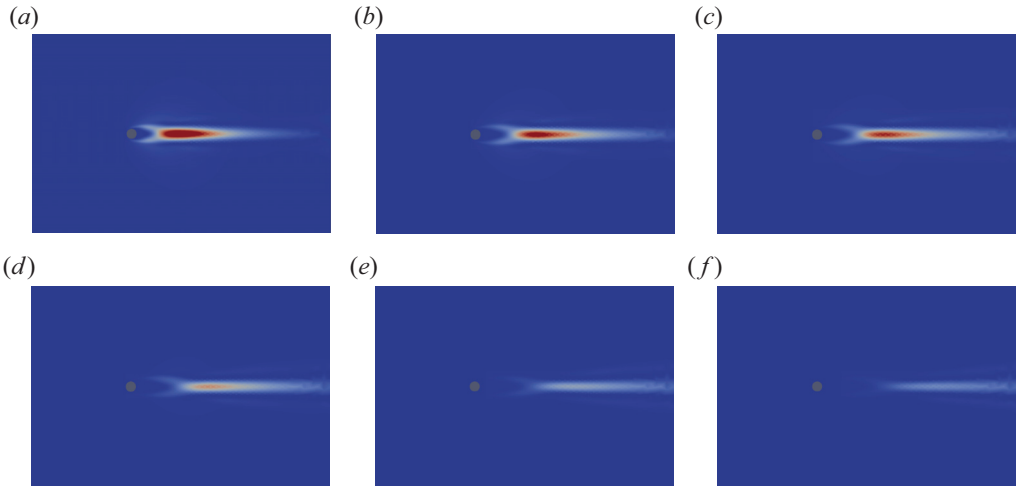



Figure 14. Field $\epsilon(\bar{\boldsymbol{q}})$ from iteration 0 ($a$) to iteration 5 ($f$). Colour scaling is constant.

flow, i.e. with the steady field $\boldsymbol{\epsilon}(\bar{\boldsymbol{q}}) = (\bar{\boldsymbol{q}} - \boldsymbol{q}_b)^T \cdot \boldsymbol{E}(\bar{\boldsymbol{q}} - \boldsymbol{q}_b)$ (figure 14). It is notable from figure 14 that the PKE peak is located far downstream in the wake, near the downstream limit of the mean recirculation bubble. The mean flow remains symmetric with respect to the $x_2$ axis, and the PKE is pushed downstream during iterations, with its peak value decreasing exponentially in amplitude.

### 3.2.3. *Frequency and nonlinearity analysis: spectrogram*

*Frequency of the flow.* The attractor of the flow (unforced or controlled) stays periodic throughout iterations, which facilitates the analysis to be done with spectrograms: only one fundamental frequency and its harmonics are visible. The flow frequency throughout iterations can be extracted from the feedback probe. As mentioned in § 3.1, the flow leaves the equilibrium with a pulsation $\omega_b = 0.779\,\mathrm{rad}/t_c$ (associated to the most unstable eigenvalue of the linearized operator) and settles on the natural attractor with a free pulsation $\omega_0 = 1.062\,\mathrm{rad}/t_c$ after a continuous transition from one to the other. The evolution of the instantaneous fundamental frequency of the flow throughout the iterative process is displayed in figure 15: the blue curve is the divergence from the base flow (from $\omega_b$ (green line) to $\omega_0$ (red line)) and the red curve is the frequency throughout the iterative process (from $\omega_0$ to $\omega_{cl}$ (black line), which is defined in the paragraph below).

In a very small number of iterations ($\approx$3), the frequency of the flow almost matches the frequency of the base flow, but the flow is not stabilized yet. Indeed, the frequency remains almost constant in subsequent oscillations, while the amplitude of all signals decreases. At the very last iterations, the frequency of the flow does not match $\omega_b$, which is expected. Indeed, the system at this point is equivalent to the closed loop between the base flow $G_b = \left[\begin{array}{c|c} \boldsymbol{A_b} & \boldsymbol{B_b} \\ \hline \boldsymbol{C_b} & \boldsymbol{0} \end{array}\right]$ and the last controller $K = \left[\begin{array}{c|c} \boldsymbol{A_K} & \boldsymbol{B_K} \\ \hline \boldsymbol{C_K} & \boldsymbol{0} \end{array}\right]$. The dynamic matrix of the closed loop is $\bar{\boldsymbol{A}} = \left[\begin{array}{cc} \boldsymbol{A_b} & \boldsymbol{B_b C_K} \\ \boldsymbol{B_K C_b} & \boldsymbol{A_K} \end{array}\right]$ and its singular mass matrix is $\bar{\boldsymbol{E}} = \left[\begin{array}{cc} \boldsymbol{E} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} \end{array}\right]$. The frequency $\omega_{cl}$ of the flow at the last iteration matches the least damped pole of this system, i.e. it is the imaginary part of the eigenvalue with largest real part, from the following generalized eigenvalue problem (singular, sparse, high-dimensional): $\lambda \in \mathbb{C}$ s.t. $\exists \boldsymbol{x} \neq \boldsymbol{0} : \bar{\boldsymbol{A}}\boldsymbol{x} = \lambda \bar{\boldsymbol{E}}\boldsymbol{x}$.

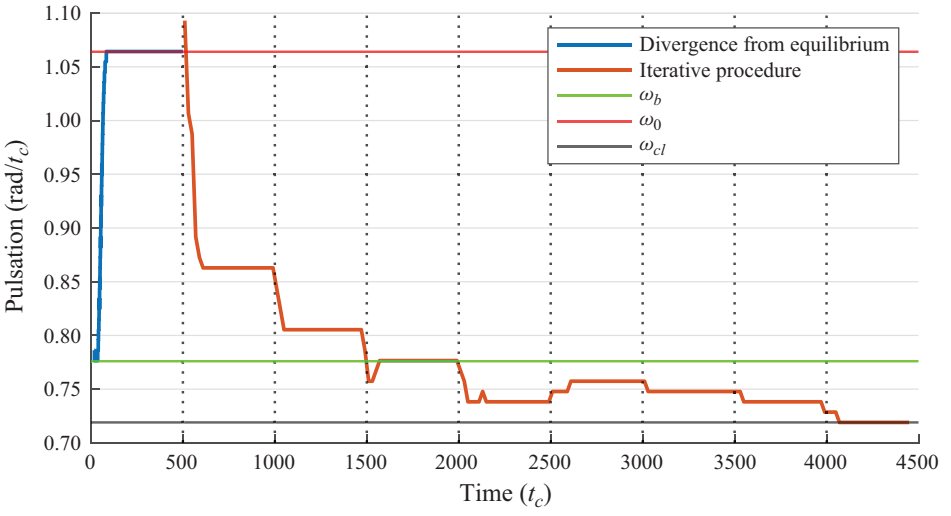



Figure 15. Spectrogram of the feedback sensor signal (dominant frequency versus time) throughout the iterative procedure. The blue curve corresponds to the trajectory from the equilibrium to the natural attractor, while the red curve corresponds to the iterative procedure itself. Notable pulsations are marked by horizontal lines: $\omega_b$ in green, $\omega_0$ in red and $\omega_{cl}$ in black. Note that the pulsation of the flow leaving the equilibrium without control ($\omega_b$) does not match the pulsation of the flow stabilized to the equilibrium by the control law ($\omega_{cl}$) – see main text for an explanation.

*Nonlinearity weakening throughout iterations.* Another interesting observation is that nonlinearities are less and less active during iterations, symmetrically to the observations on the divergence from the base flow. Previously in § 3.1 (figure 8), it was observed that nonlinearities appeared after a few time instants after perturbing the flow from its equilibrium, as higher-order harmonics in the frequency content of signals. When converging to the equilibrium with the iterative procedure, the observation is reversed (but not shown in figure 15 for space reasons): the higher-order harmonics are very present in the first 3 iterations, and they almost disappear in subsequent iterations. However, even with weaker nonlinearity, the stabilization of the flow still takes a moderate amount of additional iterations.

### 3.2.4. *Control law*

*A piecewise LTI control law.* The control law produced by the iterative procedure is in essence piecewise LTI: for a fixed iteration index, the total controller is a sum of LTI controllers, and the iteration index is a piecewise-constant function of time. The control law at any given time instant $t$ can be expressed as $K(t, s) = K_{i(t)}(s)$, where $i(t)$ is a piecewise-constant function of time. The control law can also be considered adaptive, in that the trajectory in phase space is not defined *a priori*; instead, at each iteration, controllers are synthesized to control a specific regime of the flow and reach a new, previously unknown regime.

It is notable that implementing the final controller directly from the limit cycle does not lead to stabilization of the flow, while the same controller implemented at the last iteration stabilizes the flow. First, it confirms that the control law produces a finite basin of attraction around the equilibrium, which does not encompass the natural limit cycle. Second, it indicates that the control law found here uses the variation in time of the flow to ultimately reach equilibrium, which is confirmed in the following. For the cylinder, several

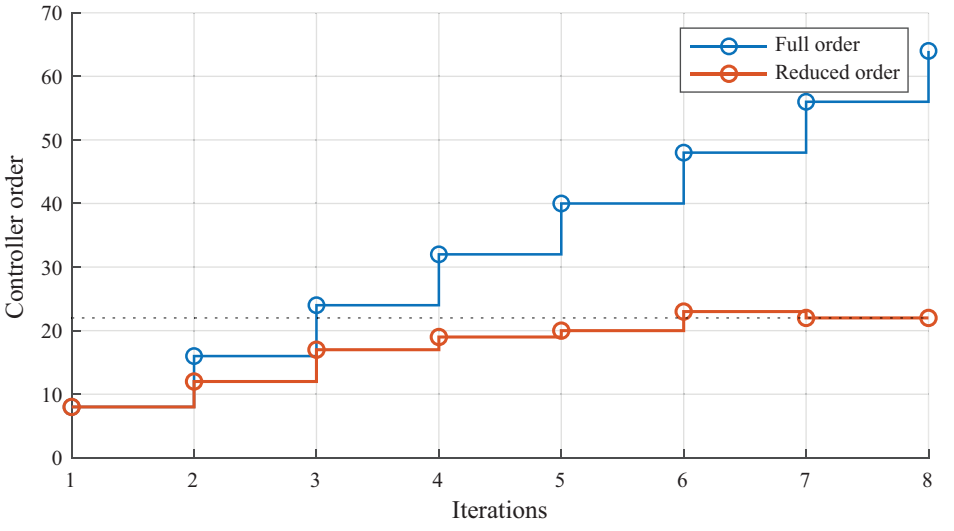



Figure 16. Controller order throughout iterations, without reduction method (blue) and with balanced truncation (red).

studies have reported full stabilization of the flow with single LTI controllers (finite basin of attraction encompassing the natural attractor, and no time variation of the control law) at $Re = 100$, but they usually use strong model hypotheses such as ROMs from linearization or Galerkin projection (see e.g. Camarri & Iollo 2010; Illingworth 2016; Jussiau *et al.* 2022).

*Controller order reduction.* Without controller order reduction, the order of the controller at iteration $i$ would be $i \times n_G$, while balanced truncation permits a controller order remaining almost constant throughout iterations. It only peaks at $n_K = 23$, then settles down to $n_K = 22$ (to be compared with a full order of 64 at the last iteration without balanced truncation in the process), as per figure 16. If needed, the reduction can also be performed with a maximum order constraint instead of a fixed HSV threshold $g_\sigma$, but there is a risk of neglecting critical dynamics for efficient control.

*Control input.* One objective of the switching method from § 2.6.3 is to induce small transients in the control input, and still manage to control the flow. The control input $u(t)$ throughout iterations is represented in figure 17. As expected from the tuning of LQG control at the first iteration, there is large overshoot in the transient of the control input at time $t_1 = 500$. In the following iterations, moderate transient in the control input is indeed observed, although the control input peaks more at some iterations (e.g. iterations 3, 8).

A last observation is that stabilizing the flow does not require increasing control input power in general. Indeed, the control input is maximal during the first iterations and keeps reducing as iterations advance, to finally reach $u(t) \to 0$ on the equilibrium as $t \to \infty$. It underlines that the first iterations are the most critical from a control input point of view, and that constraints on the control input (i.e. on the controller gain) may be relaxed as the flow comes closer to equilibrium.

*Independence of dynamical equilibrium from controller initial state.* On this configuration, the statistically steady flow regime, reached after adding a controller in the loop, seems almost independent from the initial condition of the controller. Controller initialization only has an impact on the transient before reaching said statistically steady regime, which allows using any technique for switching, such as those evoked in § 2.6.3 if

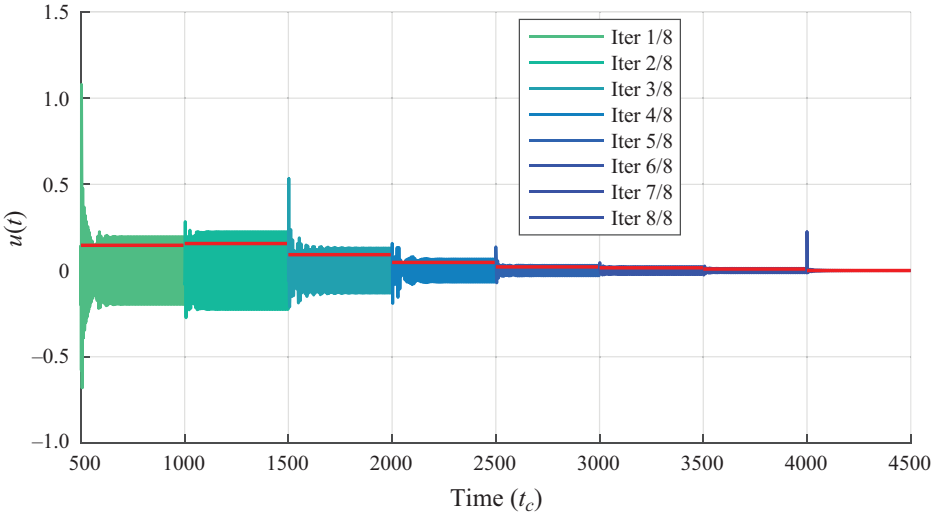



Figure 17. Control input $u(t)$ throughout iterations. The RMS$(u(t))$ is indicated by a red horizontal level at each iteration.

the transient is not deemed to be an important factor. As the two-dimensional flow past a cylinder at $Re = 100$ naturally exhibits only one attractor, this observation may be different on a more complex flow where several attractors exist simultaneously, e.g. the flow over an open cavity in Bengana *et al.* (2019) for $4410 \leq Re \leq 4600$.

## 4. Discussion

### 4.1. *Modal analysis: mean flow and implicit models*

#### 4.1.1. *Closed-loop identification and implicit flow model*

At each iteration, whenever the flow is in feedback with the control law $K$, it lies on a dynamical equilibrium and we aim at identifying the mean transfer function $G$ for the following control step. The mean transfer function $G$ corresponds to a closed-loop system: it is the flow alone, in feedback with a known controller $K$. Therefore, one may remove the influence of the controller, to retrieve an LTI model of the flow itself, denoted $G^I$, as illustrated in figure 18. More specifically, we can write

$$G = \mathcal{F}_b(G^I, K) = \frac{G^I}{1 - G^I K}, \tag{4.1}$$

and one can then deduce the implicit model of the flow alone $G^I$:

$$G^I = \frac{G}{1 + GK}. \tag{4.2}$$

From the frequency response of $G$ and $K$, we compute the frequency response, $G^I(\jmath\omega) = G(\jmath\omega)/(1 + G(\jmath\omega)K(\jmath\omega))$, and identify $G^I(s)$ as a low-order model. In practice, it proves satisfactory to choose $G^I(s)$ with the same order as $G(s)$.

#### 4.1.2. *Implicit flow model and mean flow model*

A question that might arise is the meaning of this implicit flow model. To that aim, we compare the implicit flow model with the mean flow model used in Leclercq *et al.*

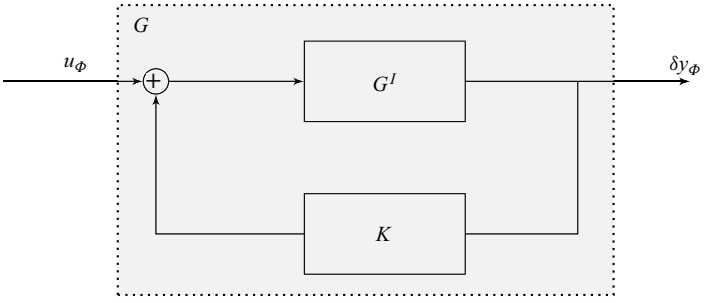



Figure 18. Block diagram of the identified closed-loop system $G$ (mean transfer function) and the model of the flow alone $G^I$ (implicit model). The feedback of the system $G^I$ with the known controller $K$ produces the identified closed-loop $G$.

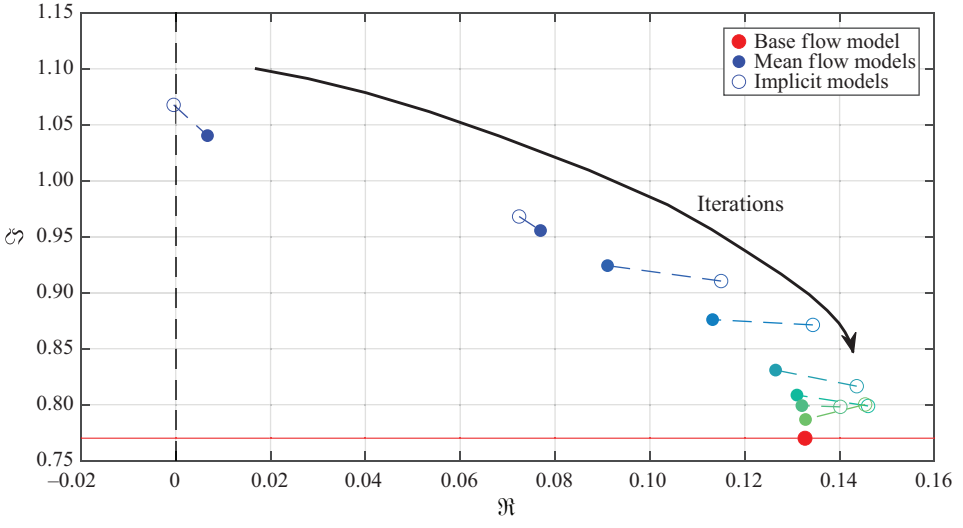



Figure 19. Unstable pole of the implicit model $G_i^I$ (open circles) and mean flow model $\bar{G}_i$ (filled circles) in complex plane, throughout iterations. The red circle is the unstable pole of the base flow model $G_b$, with its frequency represented as the red line.

(2019) for control purposes. The mean flow model corresponds to the linearization of the equations of the flow (2.4) around the mean flow $\bar{\boldsymbol{q}}$, and is denoted $\bar{G}$. It is easily deduced that the two models are not strictly identical. At the first iteration, the implicit flow model is $G_0^I = G_0$ (mean transfer function) because the controller in the loop is $K_0 = 0$, so its least stable poles are stable with very low damping (numerical artefact); and at the same time, the mean flow model $\bar{G}_0$ (figure 19) has an unstable pole.

*Displacement of the unstable pole in the complex plane.* However, it can be checked that at each iteration, both models $G_i^I$ and $\bar{G}_i$ remain close together. The location of their respective unstable pole is tracked in figure 19 (open circles for $G_i^I$, filled circles for $\bar{G}_i$, linked by a dotted line for the same iteration). At the first iteration, the pole of $G_0^I$ has very low damping, while the pole of $\bar{G}_0$ is unequivocally unstable. As iterations progress, the poles drift together towards the unstable pole of the base flow (in red), while remaining close to each other at each iteration.

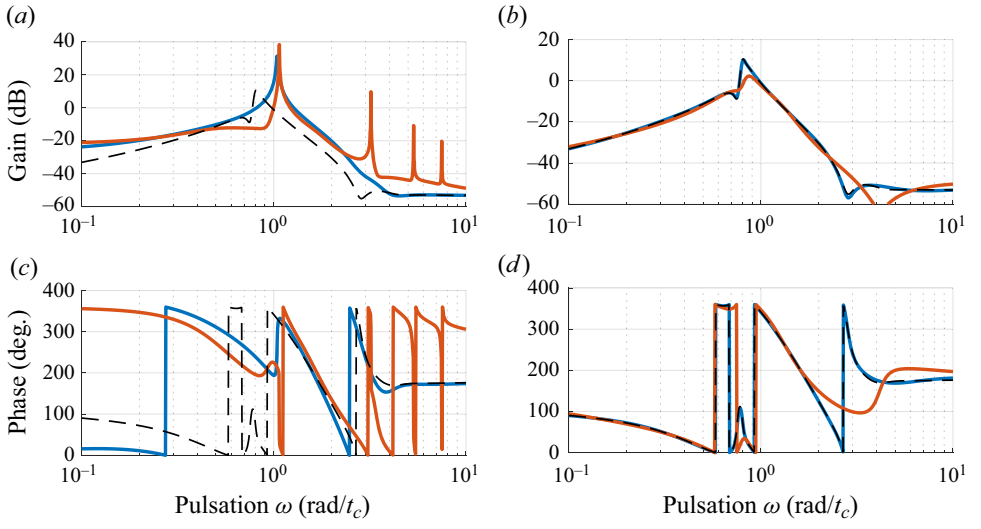



Figure 20. Bode diagrams of mean flow $\bar{G}$ (blue) and implicit $G^I$ (red) models. (*a*,*c*) At iteration 1 (unactuated flow). (*b*,*d*) At iteration 8 (last). The base flow model $G_b$ is in dashed black.

*Bode diagrams.* Finally, we also depict the Bode diagrams of the implicit flow model and the mean flow model in figure 20. While the resemblance of $\bar{G}_0$, $G_0^I$ is not obvious (slightly different frequency of the resonant/unstable pole, additional harmonics in the implicit mode explained in Leclercq & Sipp (2023)), both $\bar{G}_7$ and $G_7^I$ are very close together, and close to the base flow model $G_b$ (although the implicit model pole has larger damping).

It would be interesting to address more intricate flow regimes, such as the quasiperiodic flow over an open cavity (Leclercq *et al.* 2019) that displays several unstable poles in its mean flow model, and investigate whether the mean flow model and the implicit flow model have common features in general.

#### 4.1.3. *Nonlinear relaxation of poles*

At each iteration, the mean transfer function is supposed to exhibit poles on the imaginary axis $i\mathbb{R}$, although the identification of $G_i$ places the poles in the stable half-plane. The aim of the controllers $K_i^+$ is to damp the poles of $G_i$, farther away from the imaginary axis. After a transient regime where the flow shifts in phase space and reaches a new dynamical equilibrium, a new mean transfer function $G_{i+1}$ can be identified as well, and also has poles near the imaginary axis. Therefore, poles of $G_i$ are first damped by an LTI controller $K_i^+$, then after a transient regime, are relaxed and rejoin the vicinity of the imaginary axis again, which is illustrated in figure 21. This relaxation resembles a similar phenomenon, referred to as nonlinear relaxation, in Leclercq *et al.* (2019). Below, we display the movement of poles, from $i\mathbb{R}$ to the stable plane as they are damped by the controller, then back to $i\mathbb{R}$ due to the nonlinear transient. The same observation as in figure 15 can be made, in that the frequency of the oscillation quickly becomes constant, as the imaginary part of the pole approaches the black line at $\omega_{cl}$.

### 4.2. *Unmodelled phenomena and gaps to cover for experiments*

Oscillator flows are typically considered as being dominated by self-sustained oscillations and mostly insensitive to perturbations (in a broad sense: numerical or experimental noise,

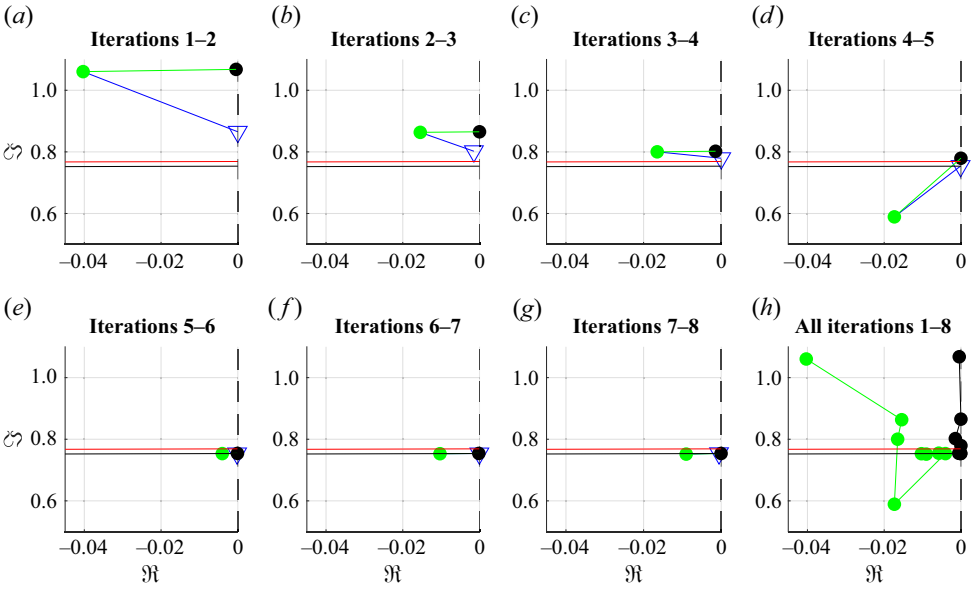



Figure 21. Nonlinear relaxation of resonant pole associated with the unique fundamental frequency, throughout iterations. The resonant pole identified from the mean transfer function $G_i$ (black circle) is displaced into the stable plane (in green) by the controller $K_i^+$. A nonlinear transient regime leading to a new dynamical equilibrium shifts the stabilized pole to the imaginary axis again (in blue). The red line is the pulsation of the base flow $\omega_b$, the black line is the pulsation of the final closed loop $\omega_{cl}$. (*h*) The route of the stabilized poles is indicated in green.

sometimes referred to as state noise), in contrast to noise amplifiers. However, in the context of periodic oscillator flows, it was shown in Bagheri (2014) that a noisy limit cycle is defined by two time scales: its natural time scale $T$, and a correlation time $t_d$ characterizing phase diffusion, which may cause oscillations to become uncorrelated after a certain duration. Depending on the quality factor $\mathcal{Q} = 2\pi(t_d/T)$, the impact of noise may be more or less important. Additionally, it is shown that eigenvalues of the Koopman operator (which are the poles of the mean transfer function; Leclercq & Sipp 2023) are expected to be damped instead of purely imaginary. For the transfer function estimation in the presence of a noisy limit cycle, more sophisticated estimation techniques should be used, such as the ones described in Van Overschee & De Moor (2012) for identification or in Bendat & Piersol (2011) for random noise processing.

Regarding actuation and sensing, we have assumed that both the actuator and sensor could be modelled in a linear fashion, although this is transparent from the data viewpoint. In the future, attention should be paid to the impact of actuator saturations or rate saturations. Also, both the actuator and sensor may be corrupted by noise, to which system identification is naturally robust, but the impact should be assessed nonetheless. For the sensor specifically, using a sensor in the wake in this study may not be limiting, as the procedure is likely to be qualitatively comparable with a wall-mounted sensor.

## 5. Conclusion and perspectives

In this article, we have proposed an automated iterative methodology for the complete stabilization of the flow past a cylinder at $Re = 100$ on its unknown natural equilibrium,

using solely input–output data and LTI controllers, to produce a piecewise LTI adaptive control law of low complexity. The objective of the approach is to drive the flow from its fully developed regime of vortex shedding, to its natural unstable equilibrium stabilized in closed loop in a linear sense. This is achieved in simulation with a sequence of 8 LTI controllers of moderate order, synthesized using models inferred from input–output data.

The methodology builds upon a previous linear iterative method for the closed-loop control of oscillator flows, proposed in Leclercq *et al.* (2019). The principle is to solve this complex nonlinear control problem iteratively: instead of addressing stabilization of a high-dimensional nonlinear system, the problem is solved as a sequence of low-order linear approximations of the problem, whose resolution iteratively drives the system closer and closer to equilibrium. Although this study demonstrated its capability to fully stabilize the quasiperiodic flow over an open cavity, it is not entirely applicable to an automated experimental set-up: it relies on the linearization of the equations about the mean state; it uses structured $\mathcal{H}_\infty$ synthesis that is arguably hard to automate reliably; and the order of the controller is increasing linearly with the iterations. In the present study, using the same principle, we target full automation of the procedure and rely only on input–output data for experimental compatibility. The modelling of the flow with an LTI plant is justified by the mean resolvent framework (Leclercq & Sipp 2023), providing keys for estimating the mean transfer with measurable input–output data only; controller synthesis is addressed with the LQG framework; and the order of the controllers is managed with online balanced truncation.

The main result is that each new controller disrupts the current attractor, leading to a new dynamical equilibrium with lower PKE, constituting an adaptive piecewise LTI control law. This iterative process ultimately drives the flow to the natural equilibrium, stabilized in closed loop. The key difference from the previous study (Leclercq *et al.* 2019) is that our approach does not assume prior knowledge of the equations or the mean flow, and may operate with minimal human supervision. In simulation, the whole method is also computationally inexpensive: it only requires forward-time simulations of the actuated flow, and low-dimensional numerical algebra.

Several investigations should be conducted in the simulation in order to study the following points. Firstly, for model-based LTI control strategies, the mean transfer function is, as an average model, the most appropriate model choice for synthesis. The optimal class of input signals to obtain the frequency response of the mean transfer function efficiently in practice (i.e. requiring the least amount of simulation or experimental time) is still an open question. Also, to enhance flow control efficiency, the focus should not be directed towards extensively polishing the identification process, but rather towards refining the synthesis method. Although the LQG framework has demonstrated effectiveness in the low-gain limit and is easily automated, alternative model-based control strategies could be explored, possibly incorporating constraints: typically, MPC (as in Arbabi *et al.* (2018)) with iterative model adaptation would naturally handle actuator saturations.

However, there is little hope that energy decrease can be predicted and optimized using solely linear criteria, as nonlinear effects are not incorporated in the mean resolvent framework. Nevertheless, model-based LTI methods provide a simple yet efficient framework for building control strategies, even based on data, and they may be more physically grounded than the model-free control methods proposed by reinforcement learning. In this sense, we would argue that model-based LTI strategies could be fine-tuned with input–output data obtained directly from the nonlinear system, aligning with the approach in Jussiau *et al.* (2022). As a final objective, the scope of this work suggests trying to implement the method in an experiment. While the approach is theoretically

fully data-based, some aspects of the method will still require being addressed in a real-life set-up, such as sensor and actuator noise, saturations and rate saturations, three-dimensionality or turbulence.

**Funding.** This research was funded by ONERA – the French Aerospace Lab.

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

**Author ORCIDs.**
William Jussiau https://orcid.org/0000-0002-7433-5722;
Colin Leclercq https://orcid.org/0000-0002-8262-0697.

## Appendix A. Mean transfer function estimation in practice

We recall that the objective from § 2.4.2 is to estimate the mean frequency response $H_0(\jmath\omega)$ from input-output data, for which some equations are reproduced below. For a multisine input $u_\Phi(t)$ at frequencies $k\omega_u$ and a measured output $y_\Phi(t)$, the Fourier coefficients can be found as harmonic averages:

$$\begin{cases} \hat{u}_\Phi(k\omega_u) = \lim\limits_{T'\to\infty} \dfrac{1}{T'} \displaystyle\int_0^{T'} u_\Phi(t)\, \mathrm{e}^{-\jmath k\omega_u t}\, \mathrm{d}t, \\ \hat{y}_\Phi(k\omega_u) = \lim\limits_{T'\to\infty} \dfrac{1}{T'} \displaystyle\int_0^{T'} y_\Phi(t)\, \mathrm{e}^{-\jmath k\omega_u t}\, \mathrm{d}t = \hat{\delta y}_\Phi(k\omega_u). \end{cases} \tag{A1}$$

Then, the frequency response depending on the phase $\boldsymbol{\Phi}$ of the input may be computed as a ratio of Fourier coefficients:

$$H_\Phi(\jmath k\omega_u) = \frac{\hat{\delta y}_\Phi(k\omega_u)}{\hat{u}_\Phi(k\omega_u)}, \tag{A2}$$

which is such that $\mathbb{E}_\Phi(H_\Phi(\jmath k\omega_u)) = H_0(\jmath k\omega_u)$. Below, we describe how the input signals $u_\Phi(t)$ are designed, how harmonic averages from (A1) (initially in (2.8)) are approximated with DFTs in practice and how the sample mean of $H_\Phi(\jmath k\omega_u)$ is computed.

### A.1. *Design of numerical experiments and DFTs*

In practice, time series are extracted with time step $\Delta t$ and finite duration $T$ to be defined, and the computations are done in sampled time. For the harmonic averages in (2.8) to be approximated with finite horizon $T$, the transient is discarded and a DFT is computed with several adjustments. Indeed, when injecting the input $u_\Phi$, the measurement $y_\Phi$ undergoes a transient regime as per Leclercq & Sipp (2023), containing the contribution of damped Floquet modes, before settling in a statistically steady regime, represented in figure 22. The contribution of the transient to the harmonic average (2.8) vanishes for $T \to \infty$; in practice, it is better to suppress it from the dataset for an estimation based on a finite time window, using the DFT. Therefore, the beginning of the experiment, corresponding to the first $P_{tr}$ periods of the signal $u_\Phi(t)$, is discarded. Also, as the DFT assumes periodicity of the signals, which is not the case for $y_\Phi(t)$, a Hann window is used. Accounting for the $P_{tr}$ discarded periods, the total duration of the experiment is $(P_{tr}+P)(2\pi/\omega_u)$ with $P_{tr}, P \in \mathbb{N}$, but a subset of length $T = P(2\pi/\omega_u)$ is used for the estimation. The subsequent DFT resolution is by definition $\Delta\omega = 2\pi/T$, such that $\omega_u = P\Delta\omega$, i.e. non-zero contributions in the input are retrieved every $P$ point in the DFT. In practice, $P_{tr} = P = 4$ are fixed, and the number of realizations $M$ for the ensemble average is chosen as described below.

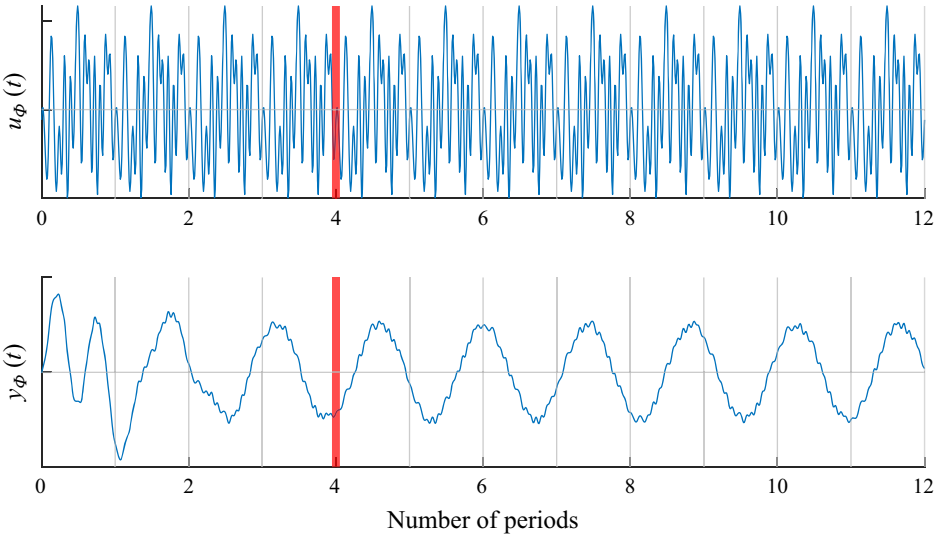



Figure 22. Design of multisines – synthetic data. In this illustration, 12 periods of a periodic input $u_\Phi(t)$ are represented, along with the corresponding output $y_\Phi(t)$. The first $P_{tr} = 4$ periods of both the input and the output are discarded, for containing a transient regime due to damped Floquet modes (Leclercq & Sipp 2023). To mitigate the quasiperiodicity of the remaining portion of $y_\Phi(t)$, a Hann window is used when computing both DFTs.

### A.2. *Convergence of the sample mean and the time-invariance hypothesis*

The sample mean $\bar{H}(\jmath\omega)$ of $H_\Phi(\jmath k\omega_u)$ is computed by performing $M$ independent simulations of the nonlinear system with different realizations of the input $u_\Phi(t)$, from the same limit cycle phase $\phi$. The output $y_\Phi(t)$ of each simulation is gathered for the frequency response computation. We perform a total of $\hat{M} = 16$ experiments. For each experiment $m \in [1, \hat{M}]$, we can compute the frequency response $H^m_\Phi(\jmath\omega)$. Now, for efficient estimation in practice, we wish to be able to reduce $\hat{M}$; we would like to find a lower value $M \leq \hat{M}$ such that the sample mean depending on $M$, defined below, is converged:

$$\bar{H}_\Phi(\jmath\omega; M) = \frac{1}{M}\sum_{m=1}^{M} H^m_\Phi(\jmath\omega). \tag{A3}$$

For that purpose, the reference is taken as $\bar{H}(\jmath\omega) = \bar{H}_\Phi(\jmath\omega; \hat{M})$, which is reasonable in practice since the value of $\bar{H}_\Phi(\jmath\omega; \hat{M})$ is almost constant for $M \in [12, 16]$.

First, we consider that the sample mean is converged with respect to $M$ at a given frequency when $\sqrt{\mathrm{Var}\bar{H}_\Phi(\jmath\omega; M)} \ll |\bar{H}(\jmath\omega)|$. Since $\mathrm{Var}\bar{H}_\Phi(\jmath\omega; M) = (1/M)\mathrm{Var}H_\Phi(\jmath\omega)$, the previous condition translates into

$$\zeta(\jmath\omega) = \frac{\sqrt{\mathrm{Var}H_\Phi(\jmath\omega)}}{|\bar{H}(\jmath\omega)|} \ll \sqrt{M}. \tag{A4}$$

In the above expression, the variance of the frequency response itself $\mathrm{Var}H_\Phi(\jmath\omega)$ is estimated once with the full data $M = \hat{M}$. Note that the formulation (A4) is similar to that of Leclercq & Sipp (2023) with the quantity $\eta(\jmath\omega; u)$ that quantifies the variation of the transfer with respect to the limit cycle phase $\phi$ (for a given input signal $u$). Here, $\phi$ is kept

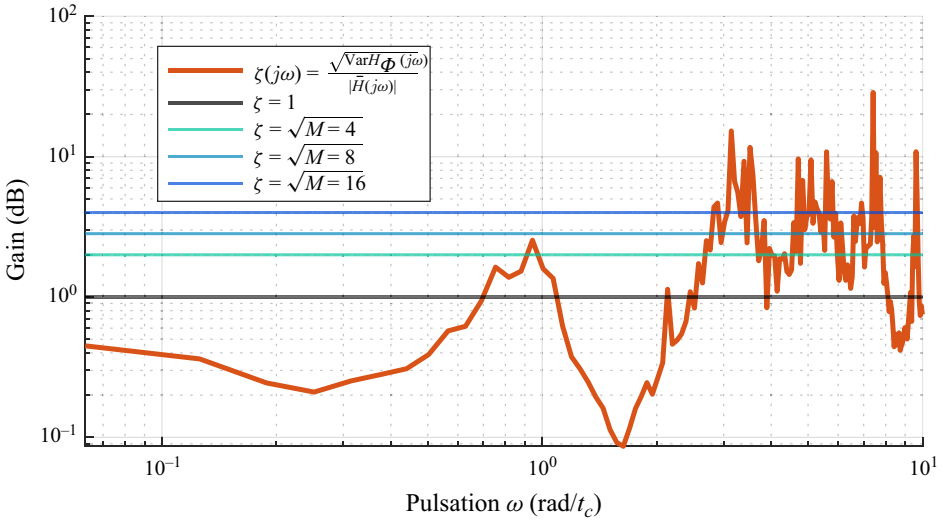



Figure 23. Comparison of $\zeta(\jmath\omega)$ with $\zeta = 1$ and $\sqrt{M}$ for $M = 4, 8, 16$. It is notable that the mean transfer at $M = 4$ cannot be considered entirely converged for $\omega \approx 0.9$ or $\omega \geq 3$. In turn, models presented in the paper have low reliability for $\omega \geq 3$.

constant (signals are injected at the same phase of the limit cycle for every realization), but the phase $\boldsymbol{\Phi}$ of the input signal itself is varied.

In figure 23, we show the estimation of $\mathrm{Var} H_{\boldsymbol{\Phi}}(\jmath\omega)$ on the whole pulsation range, and horizontal lines as $\sqrt{M}$ for $M = 4, 8, 16$. In this situation, it appears that $M = 4$ offers a fair estimation of the mean with low computational complexity. Although there seem to be missed samples for the mean to be converged at high frequency ($\omega \geq 3$), a significant part of the low-frequency content indeed lies beyond the horizontal line $\zeta = \sqrt{4} = 2$.

Besides, the horizontal line $\zeta = 1$ in figure 23 conveys important information about the relevance of the time-invariant approximation of the flow. Indeed, the samples of $H_{\boldsymbol{\Phi}}(\jmath\omega)$ are distributed around their mean $\bar{H}(\jmath\omega)$ due to the time dependence of the unforced flow (Leclercq & Sipp 2023). When approximating the frequency response with its mean, representing a time-invariant model, it is important that $\bar{H}(\jmath\omega)$ conveys sufficient information. Here, $\zeta(\jmath\omega) \ll 1$ indicates that the mean transfer function is a satisfactory time-invariant approximation of the model at said frequency and for a given input signal (i.e. the time-varying effects are low), and conversely for $\zeta(\jmath\omega) \gg 1$. The limit $\zeta = 1$ is indicated as a black horizontal line in figure 23 and it may be observed that the time-invariance hypothesis holds well in the range $\omega \in [0, 2.5]\,\mathrm{rad}/t_c$, except in the vicinity of the resonance at $\omega \approx 1\,\mathrm{rad}/t_c$. Similarly to Leclercq & Sipp (2023), the time-invariance hypothesis is poorer at higher frequency.

## Appendix B. Subspace identification method: FORSE

The FORSE (Liu *et al.* 1994) estimates a discrete-time state-space representation in two distinct steps, from the frequency response $\{H(\jmath\omega_i)\}_{i=1}^{M}$. We denote $\Delta t$ the sampling time of the discrete-time SISO system ($\hat{\boldsymbol{A}}$, $\hat{\boldsymbol{B}}$, $\hat{\boldsymbol{C}}$, $\hat{\boldsymbol{D}}$) estimated by the algorithm. We start by

constructing a matrix containing the shifted frequency response:

$$\boldsymbol{Y} = \begin{bmatrix} H(\jmath\omega_1) & \dots & H(\jmath\omega_M) \\ H(\jmath\omega_1)\,\mathrm{e}^{\jmath\omega_1\Delta t} & \dots & H(\jmath\omega_M)\,\mathrm{e}^{\jmath\omega_M\Delta t} \\ \vdots & \vdots & \vdots \\ H(\jmath\omega_1)\,\mathrm{e}^{\jmath\omega_1(q-1)\Delta t} & \dots & H(\jmath\omega_M)\,\mathrm{e}^{\jmath\omega_M(q-1)\Delta t} \end{bmatrix}, \tag{B1}$$

where $q \in \mathbb{N}$ is a parameter of the identification algorithm. When working with a frequency response directly, the influence of input and output is included in the matrix $\boldsymbol{Y} \in \mathbb{C}^{q\times M}$, and an impulse input matrix $\boldsymbol{U} \in \mathbb{C}^{q\times M}$ needs to be constructed, as follows:

$$\boldsymbol{U} = \begin{bmatrix} 1 & \dots & 1 \\ \vdots & \vdots & \vdots \\ \mathrm{e}^{\jmath\omega_1(q-1)\Delta t} & \dots & \mathrm{e}^{\jmath\omega_1(q-1)\Delta t} \end{bmatrix}. \tag{B2}$$

At this point, it is possible to weight the frequency response with a positive-definite diagonal matrix $\boldsymbol{R} \in \mathbb{R}^{M\times M}$ in order to focus on particular frequency domains (e.g. resonant modes). We denote the real part with $\Re$, the Moore–Penrose pseudo-inverse with the superscript $\dagger$ and construct the following real matrix $\boldsymbol{H}$, on which we subsequently perform a singular value decomposition:

$$\begin{aligned} \boldsymbol{H} &= \Re(\boldsymbol{YRY}^*) - \Re(\boldsymbol{YRU}^*)\Re(\boldsymbol{URU}^*)^\dagger\Re(\boldsymbol{YRU}^*)^T \\ &= \begin{bmatrix} \boldsymbol{\Psi}_q & \tilde{\boldsymbol{\Psi}}_q \end{bmatrix} \begin{bmatrix} \boldsymbol{\Sigma} & 0 \\ 0 & \tilde{\boldsymbol{\Sigma}} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Psi}_q \\ \tilde{\boldsymbol{\Psi}}_q \end{bmatrix}. \end{aligned} \tag{B3}$$

The $n_r$ largest singular values are stored in the diagonal matrix $\boldsymbol{\Sigma}$ associated with the singular vectors $\boldsymbol{\Psi}_q$, and the rest are discarded, dictating the precision of the ROM. Then, we partition $\boldsymbol{\Psi}_q$ as rows $\boldsymbol{\psi}_i \in \mathbb{C}^{1\times n_r}$, and construct submatrices $\boldsymbol{\Psi}_{q-1}$, $\hat{\boldsymbol{\Psi}}_{q-1}$ such that

$$\boldsymbol{\Psi}_q = \begin{bmatrix} \boldsymbol{\psi}_0 \\ \boldsymbol{\psi}_1 \\ \vdots \\ \boldsymbol{\psi}_{q-2} \\ \boldsymbol{\psi}_{q-1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Psi}_{q-1} \\ \boldsymbol{\psi}_{q-1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\psi}_0 \\ \hat{\boldsymbol{\Psi}}_{q-1} \end{bmatrix}. \tag{B4}$$

Finally, we estimate the discrete-time dynamics $\hat{\boldsymbol{A}}$ and the measurement matrix $\hat{\boldsymbol{C}}$ as follows:

$$\hat{\boldsymbol{A}} = \boldsymbol{\Psi}_{q-1}^\dagger \hat{\boldsymbol{\Psi}}_{q-1}, \quad \hat{\boldsymbol{C}} = \boldsymbol{\psi}_0. \tag{B5$a$,$b$}$$

If we wish to introduce stability constraints on the system to be identified, the aforementioned pseudo-inverse operation (B5$a$,$b$) is augmented with linear matrix inequalities on $\hat{\boldsymbol{A}}$ (Demourant & Poussot-Vassal 2017). After $\hat{\boldsymbol{A}}$, $\hat{\boldsymbol{C}}$ have been estimated, the computation of the frequency response from the state-space realization is linear in $\hat{\boldsymbol{B}}$

and $\hat{\boldsymbol{D}}$ at each frequency:

$$H(\jmath\omega_k) = \hat{\boldsymbol{C}}(\mathrm{e}^{-\jmath\omega_k\Delta t}\boldsymbol{I} - \hat{\boldsymbol{A}})^{-1}\hat{\boldsymbol{B}} + \hat{\boldsymbol{D}} = \boldsymbol{P}(\jmath\omega_k)\begin{bmatrix}\hat{\boldsymbol{B}}\\ \hat{\boldsymbol{D}}\end{bmatrix}. \tag{B6}$$

Finding estimates of $\hat{\boldsymbol{B}}$, $\hat{\boldsymbol{D}}$ is then reformulated as a linear least-squares problem:

$$\hat{\boldsymbol{B}}, \hat{\boldsymbol{D}} = \arg\min_{\boldsymbol{B},\boldsymbol{D}} \left\| \begin{bmatrix} H(\jmath\omega_1) \\ \vdots \\ H(\jmath\omega_M) \end{bmatrix} - \begin{bmatrix} \boldsymbol{P}(\jmath\omega_1) \\ \vdots \\ \boldsymbol{P}(\jmath\omega_M) \end{bmatrix} \begin{bmatrix} \boldsymbol{B} \\ \boldsymbol{D} \end{bmatrix} \right\|_F^2. \tag{B7}$$

## Appendix C. The LQG synthesis

The principle of a LQG controller, which is an observed-state feedback, is recalled here for SISO plants. Additional information in the context of flow control may be found in e.g. Schmid & Sipp (2016), Barbagallo *et al.* (2009), Kim & Bewley (2007), Carini *et al.* (2015) and Brunton & Noack (2015). Consider a SISO plant $G = \left[\begin{array}{c|c} \boldsymbol{A} & \boldsymbol{B} \\ \hline \boldsymbol{C} & \boldsymbol{0} \end{array}\right]$ with state vector $\boldsymbol{x}$, control input $u$ and measurement output $y$.

*Linear quadratic regulator.* We wish to solve an optimal control problem with full-state information, i.e. find the control input $u(t)$ that minimizes a performance criterion defined as $J = \int_0^\infty [\boldsymbol{x}(t)^T \boldsymbol{Q}\boldsymbol{x}(t) + Ru^2(t)]\,\mathrm{d}t$ with parameters $R > 0$, $\boldsymbol{Q} \succeq 0$ (positive semi-definite matrix). The solution is a state feedback law $u(t) = \boldsymbol{K}\boldsymbol{x}(t)$, where $\boldsymbol{K}$ is a matrix of gains computed as $\boldsymbol{K} = -R^{-1}\boldsymbol{B}^T\boldsymbol{P}$, with $\boldsymbol{P}$ the solution of the Riccati equation: $\boldsymbol{A}^T\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{P}\boldsymbol{B}R^{-1}\boldsymbol{B}^T\boldsymbol{P} + \boldsymbol{Q} = 0$.

*Construction of an observer.* Here, the state $\boldsymbol{x}(t)$ of the reduced-order plant $G$ cannot be accessed directly, and only the output $y(t) = \boldsymbol{C}\boldsymbol{x}(t)$ is available for feedback control. We construct an estimate $\hat{\boldsymbol{x}}(t)$ of the state of the process based on past measurements and inputs, and use it in an observed-state feedback law: $u(t) = \boldsymbol{K}\hat{\boldsymbol{x}}(t)$. The dynamics for the estimated state $\hat{\boldsymbol{x}}$ reproduces the plant dynamics, with a corrective forcing $\hat{y} - y$ weighted by a gain $\boldsymbol{L}$ to be defined, accounting for measurements in real time:

$$\begin{cases} \dot{\hat{\boldsymbol{x}}} = \boldsymbol{A}\hat{\boldsymbol{x}} + \boldsymbol{B}u + \boldsymbol{L}^T(\hat{y} - y), \\ \hat{y} = \boldsymbol{C}\hat{\boldsymbol{x}}. \end{cases} \tag{C1}$$

It can be shown that solving for $\boldsymbol{L}$ is dual to the previous problem, with $\boldsymbol{A} \leftarrow \boldsymbol{A}^T$, $\boldsymbol{B} \leftarrow \boldsymbol{C}^T$ and new parameters $\boldsymbol{W} \succeq 0$, $V > 0$ that are covariances of additive white noise on the state $\boldsymbol{x}(t)$ and measurement $y(t)$.

*Linear quadratic Gaussian regulator.* The final controller is formed with the observer gain $\boldsymbol{L}$ (depending on parameters $\boldsymbol{W}, V$) and the state-feedback gain $\boldsymbol{K}$ (depending on parameters $\boldsymbol{Q}, R$) making it dynamic and expressed as follows:

$$K_{LQG} = \left[\begin{array}{c|c} \boldsymbol{A} + \boldsymbol{B}\boldsymbol{K} + \boldsymbol{L}^T\boldsymbol{C} & -\boldsymbol{L}^T \\ \hline \boldsymbol{K} & \boldsymbol{0} \end{array}\right]. \tag{C2}$$

## References


Aleksić, K., Luchtenburg, M., King, R., Noack, B.R. & Pfeifer, J. 2010 Robust nonlinear control versus linear model predictive control of a bluff body wake. In *5th Flow Control Conference*, p. 4833. AIAA.

Aleksić-Roessner, K., King, R., Lehmann, O., Tadmor, G. & Morzyński, M. 2014 On the need of nonlinear control for efficient model-based wake stabilization. *Theor. Comput. Fluid Dyn.* **28**, 23–49.

Amestoy, P.R., Duff, I.S., L'Excellent, J.-Y. & Koster, J. 2000 MUMPS: a general purpose distributed memory sparse solver. In *International Workshop on Applied Parallel Computing*, pp. 121–130. Springer.

Apkarian, P., Gahinet, P. & Buhr, C. 2014 Multi-model, multi-objective tuning of fixed-structure controllers. In *2014 European Control Conference (ECC)*, pp. 856–861. IEEE.

Apkarian, P. & Noll, D. 2006 Nonsmooth $\mathcal{H}_\infty$ synthesis. *IEEE Trans. Autom. Control* **51** (1), 71–86.

Arbabi, H., Korda, M. & Mezić, I. 2018 A data-driven Koopman model predictive control framework for nonlinear partial differential equations. In *2018 IEEE Conference on Decision and Control (CDC)*, pp. 6409–6414. IEEE.

Bagheri, S. 2014 Effects of weak noise on oscillating flows: linking quality factor, Floquet modes, and Koopman spectrum. *Phys. Fluids* **26** (9), 094104.

Barbagallo, A., Sipp, D. & Schmid, P.J. 2009 Closed-loop control of an open cavity flow using reduced-order models. *J. Fluid Mech.* **641**, 1–50.

Barbagallo, A., Sipp, D. & Schmid, P.J. 2011 Input–output measures for model reduction and closed-loop control: application to global modes. *J. Fluid Mech.* **685**, 23–53.

Barkley, D. 2006 Linear analysis of the cylinder wake mean flow. *Europhys. Lett.* **75** (5), 750.

Barkley, D., Tuckerman, L.S. & Golubitsky, M. 2000 Bifurcation theory for three-dimensional flow in the wake of a circular cylinder. *Phys. Rev.* E **61** (5), 5247.

Bendat, J.S. & Piersol, A.G. 2011 *Random Data: Analysis and Measurement Procedures*. Wiley.

Bengana, Y., Loiseau, J.-C., Robinet, J.-C. & Tuckerman, L.S. 2019 Bifurcation analysis and frequency prediction in shear-driven cavity flow. *J. Fluid Mech.* **875**, 725–757.

Benner, P., Heiland, J. & Werner, S.W.R. 2022 Robust output-feedback stabilization for incompressible flows using low-dimensional $\mathcal{H}_\infty$-controllers. *Comput. Optim. Appl.* **82** (1), 225–249.

Bergmann, M., Cordier, L. & Brancher, J.-P. 2005 Optimal rotary control of the cylinder wake using proper orthogonal decomposition reduced-order model. *Phys. Fluids* **17** (9), 097101.

Bieker, K., Peitz, S., Brunton, S.L., Kutz, J.N. & Dellnitz, M. 2019 Deep model predictive control with online learning for complex physical systems. Preprint, arXiv:1905.10094.

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

Camarri, S. & Iollo, A. 2010 Feedback control of the vortex-shedding instability based on sensitivity analysis. *Phys. Fluids* **22** (9), 094102.

Carini, M., Pralits, J.O. & Luchini, P. 2015 Feedback control of vortex shedding using a full-order optimal compensator. *J. Fluids Struct.* **53**, 15–25.

Castellanos, R., Cornejo Maceda, G.Y., De La Fuente, I., Noack, B.R., Ianiro, A. & Discetti, S. 2022 Machine-learning flow control with few sensor feedback and measurement noise. *Phys. Fluids* **34** (4), 047118.

Cheong, S.-Y. & Safonov, M.G. 2008 Bumpless transfer for adaptive switching controls. *IFAC Proc. Vol.* **41** (2), 14415–14420.

Cornejo Maceda, G.Y., Li, Y., Lusseyran, F., Morzyński, M. & Noack, B.R. 2021 Stabilization of the fluidic pinball with gradient-enriched machine learning control. *J. Fluid Mech.* **917**, A42.

Dahan, J.A., Morgans, A.S. & Lardeau, S. 2012 Feedback control for form-drag reduction on a bluff body with a blunt trailing edge. *J. Fluid Mech.* **704**, 360–387.

Dalla Longa, L., Morgans, A.S. & Dahan, J.A. 2017 Reducing the pressure drag of a D-shaped bluff body using linear feedback control. *Theor. Comput. Fluid Dyn.* **31**, 567–577.

Demourant, F. & Poussot-Vassal, C. 2017 A new frequency-domain subspace algorithm with restricted poles location through LMI regions and its application to a wind tunnel test. *Intl J. Control* **90** (4), 779–799.

Doyle, J.C. 1978 Guaranteed margins for LQG regulators. *IEEE Trans. Autom. Control* **23** (4), 756–757.

Enns, D.F. 1984 Model reduction with balanced realizations: an error bound and a frequency weighted generalization. In *The 23rd IEEE Conference on Decision and Control*, pp. 127–132. IEEE.

Evstafyeva, O., Morgans, A.S. & Dalla Longa, L. 2017 Simulation and feedback control of the Ahmed body flow exhibiting symmetry breaking behaviour. *J. Fluid Mech.* **817**, R2.

FLINOIS, T.L.B. & MORGANS, A.S. 2016 Feedback control of unstable flows: a direct modelling approach using the Eigensystem Realisation Algorithm. *J. Fluid Mech.* **793**, 41–78.

GARNIER, P., VIQUERAT, J., RABAULT, J., LARCHER, A., KUHNLE, A. & HACHEM, E. 2021 A review on deep reinforcement learning for fluid mechanics. *Comput. Fluids* **225**, 104973.

GERHARD, J., PASTOOR, M., KING, R., NOACK, B.R., DILLMANN, A., MORZYNSKI, M. & TADMOR, G. 2003 Model-based control of vortex shedding using low-dimensional Galerkin models. In *33rd AIAA Fluid Dynamics Conference and Exhibit*, p. 4262. AIAA.

GHRAIEB, H., VIQUERAT, J., LARCHER, A., MELIGA, P. & HACHEM, E. 2021 Single-step deep reinforcement learning for open-loop control of laminar and turbulent flows. *Phys. Rev. Fluids* **6** (5), 053902.

GUGERCIN, S. & ANTOULAS, A.C. 2004 A survey of model reduction by balanced truncation and some new results. *Intl J. Control* **77** (8), 748–766.

GUSTAVSEN, B. & SEMLYEN, A. 1999 Rational approximation of frequency domain responses by vector fitting. *IEEE Trans. Power Deliv.* **14** (3), 1052–1061.

HEILAND, J. & WERNER, S.W.R. 2023 Low-complexity linear parameter-varying approximations of incompressible Navier–Stokes equations for truncated state-dependent Riccati feedback. Preprint, arXiv:2303.11515.

HUANG, D., JIN, B., LASAGNA, D., CHERNYSHENKO, S. & TUTTY, O. 2017 Expensive control of long-time averages using sum of squares and its application to a laminar wake flow. *IEEE Trans. Control Syst. Technol.* **25** (6), 2073–2086.

ILLINGWORTH, S.J. 2016 Model-based control of vortex shedding at low Reynolds numbers. *Theor. Comput. Fluid Dyn.* **30** (5), 429–448.

ILLINGWORTH, S.J., MORGANS, A.S. & ROWLEY, C.W. 2011 Feedback control of flow resonances using balanced reduced-order models. *J. Sound Vib.* **330** (8), 1567–1581.

ILLINGWORTH, S.J., MORGANS, A.S. & ROWLEY, C.W. 2012 Feedback control of cavity flow oscillations using simple linear models. *J. Fluid Mech.* **709**, 223–248.

JIN, B., ILLINGWORTH, S.J. & SANDBERG, R.D. 2020 Feedback control of vortex shedding using a resolvent-based modelling approach. *J. Fluid Mech.* **897**, A26.

JUANG, J.-N. & SUZUKI, H. 1988 An eigensystem realization algorithm in frequency domain for modal parameter identification. *J. Vib. Acoust. Stress Reliab. Des.* **110** (1), 24.

JUSSIAU, W., LECLERCQ, C., DEMOURANT, F. & APKARIAN, P. 2022 Learning linear feedback controllers for suppressing the vortex-shedding flow past a cylinder. *IEEE Control Syst. Lett.* **6**, 3212–3217.

KIM, J. & BEWLEY, T.R. 2007 A linear systems approach to flow control. *Annu. Rev. Fluid Mech.* **39**, 383–417.

KING, R., SEIBOLD, M., LEHMANN, O., NOACK, B.R., MORZYŃSKI, M. & TADMOR, G. 2005 Nonlinear flow control based on a low dimensional model of fluid flow. In *Control and Observer Design for Nonlinear Finite and Infinite Dimensional Systems*, pp. 369–386. Springer.

KORDA, M. & MEZIĆ, I. 2018*a* Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control. *Automatica* **93**, 149–160.

KORDA, M. & MEZIĆ, I. 2018*b* On convergence of extended dynamic mode decomposition to the koopman operator. *J. Nonlinear Sci.* **28**, 687–710.

LASAGNA, D., HUANG, D., TUTTY, O.R. & CHERNYSHENKO, S. 2016 Sum-of-squares approach to feedback control of laminar wake flows. *J. Fluid Mech.* **809**, 628–663.

LAUB, A., HEATH, M., PAIGE, C. & WARD, R. 1987 Computation of system balancing transformations and other applications of simultaneous diagonalization algorithms. *IEEE Trans. Autom. Control* **32** (2), 115–122.

LECLERCQ, C., DEMOURANT, F., POUSSOT-VASSAL, C. & SIPP, D. 2019 Linear iterative method for closed-loop control of quasiperiodic flows. *J. Fluid Mech.* **868**, 26–65.

LECLERCQ, C. & SIPP, D. 2023 Mean resolvent operator of a statistically steady flow. *J. Fluid Mech.* **968**, A13.

LEHTOMAKI, N., SANDELL, N. & ATHANS, M. 1981 Robustness results in linear-quadratic Gaussian based multivariable control designs. *IEEE Trans. Autom. Control* **26** (1), 75–93.

LIU, K., JACQUES, R.N. & MILLER, D.W. 1994 Frequency domain structural system identification by observability range space extraction. In *Proceedings of 1994 American Control Conference-ACC'94*, vol. 1, pp. 107–111. IEEE.

LIU, Q., SUN, Y., CATTAFESTA, L.N., UKEILEY, L.S. & TAIRA, K. 2018 Resolvent analysis of compressible flow over a long rectangular cavity. In *2018 AIAA Aerospace Sciences Meeting*, p. 0588. AIAA.

LOGG, A., MARDAL, K.-A. & WELLS, G. 2012 *Automated Solution of Differential Equations by the Finite Element Method: The FEniCS Book*, vol. 84. Springer.

MOORE, B. 1981 Principal component analysis in linear systems: controllability, observability, and model reduction. *IEEE Trans. Autom. Control* **26** (1), 17–32.

MORTON, J., JAMESON, A., KOCHENDERFER, M.J. & WITHERDEN, F. 2018 Deep dynamical modeling and control of unsteady fluid flows. In *Advances in Neural Information Processing Systems 31 (NeurIPS 2018)* (ed. S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi & R. Garnett), pp. 9258–9268. Curran Associates.

OTTO, S.E., PEITZ, S. & ROWLEY, C.W. 2022 Learning bilinear models of actuated Koopman generators from partially-observed trajectories. Preprint, arXiv:2209.09977.

OZDEMIR, A.A. & GUMUSSOY, S. 2017 Transfer function estimation in system identification toolbox via vector fitting. *IFAC-PapersOnLine* **50** (1), 6232–6237.

PAGE, J. & KERSWELL, R.R. 2019 Koopman mode expansions between simple invariant solutions. *J. Fluid Mech.* **879**, 1–27.

PARIS, R., BENEDDINE, S. & DANDOIS, J. 2021 Robust flow control and optimal sensor placement using deep reinforcement learning. *J. Fluid Mech.* **913**, A25.

PARK, D.S., LADD, D.M. & HENDRICKS, E.W. 1994 Feedback control of von Kármán vortex shedding behind a circular cylinder at low Reynolds numbers. *Phys. Fluids* **6** (7), 2390–2405.

PAXMAN, J.P. 2004 Switching controllers: realization, initialization and stability. PhD thesis, University of Cambridge.

PEITZ, S. & KLUS, S. 2019 Koopman operator-based model reduction for switched-system control of PDEs. *Automatica* **106**, 184–191.

PEITZ, S. & KLUS, S. 2020 Feedback control of nonlinear PDEs using data-efficient reduced order models based on the Koopman operator. In *The Koopman Operator in Systems and Control: Concepts, Methodologies, and Applications*, pp. 257–282. Springer.

PEITZ, S., OTTO, S.E. & ROWLEY, C.W. 2020 Data-driven model predictive control using interpolated Koopman generators. *SIAM J. Appl. Dyn. Syst.* **19** (3), 2162–2193.

PERNEBO, L. & SILVERMAN, L. 1982 Model reduction via balanced state space representations. *IEEE Trans. Autom. Control* **27** (2), 382–387.

PROCTOR, J.L., BRUNTON, S.L. & KUTZ, J.N. 2016 Dynamic mode decomposition with control. *SIAM J. Appl. Dyn. Syst.* **15** (1), 142–161.

RABAULT, J., KUCHTA, M., JENSEN, A., RÉGLADE, U. & CERARDI, N. 2019 Artificial neural networks trained through deep reinforcement learning discover control strategies for active flow control. *J. Fluid Mech.* **865**, 281–302.

ROWLEY, C.W. 2005 Model reduction for fluids, using balanced proper orthogonal decomposition. *Intl J. Bifurcation Chaos* **15** (03), 997–1013.

ROWLEY, C.W. & JUTTIJUDATA, V. 2005 Model-based control and estimation of cavity flow oscillations. In *Proceedings of the 44th IEEE Conference on Decision and Control*, pp. 512–517. IEEE.

SCHMID, P.J. & SIPP, D. 2016 Linear control of oscillator and amplifier flows. *Phys. Rev. Fluids* **1** (4), 040501.

SCHOUKENS, J., GUILLAUME, P. & PINTELON, R. 1991 Design of multisine excitations. In *International Conference on Control 1991*, pp. 638–643. IET.

SCHOUKENS, J., LATAIRE, J., PINTELON, R. & VANDERSTEEN, G. 2008 Robustness issues of the equivalent linear representation of a nonlinear system. In *2008 IEEE Instrumentation and Measurement Technology Conference*, pp. 332–335. IEEE.

SCHOUKENS, J., VAES, M. & PINTELON, R. 2016 Linear system identification in a nonlinear setting: nonparametric analysis of the nonlinear distortions and their impact on the best linear approximation. *IEEE Control Syst. Mag.* **36** (3), 38–69.

SIPP, D. & SCHMID, P.J. 2016 Linear closed-loop control of fluid instabilities and noise-induced perturbations: a review of approaches and tools. *Appl. Mech. Rev.* **68** (2), 020801.

SON, D. & CHOI, H. 2018 Iterative feedback tuning of the proportional-integral-differential control of flow over a circular cylinder. *IEEE Trans. Control Syst. Technol.* **27** (4), 1385–1396.

SYRMOS, V.L., ABDALLAH, C.T., DORATO, P. & GRIGORIADIS, K. 1997 Static output feedback – a survey. *Automatica* **33** (2), 125–137.

VAN OVERSCHEE, P. & DE MOOR, B. 2012 *Subspace Identification for Linear Systems: Theory–Implementation–Applications*. Springer.

VIQUERAT, J., MELIGA, P., LARCHER, A. & HACHEM, E. 2022 A review on deep reinforcement learning for fluid mechanics: an update. *Phys. Fluids* **34** (11), 111301.

WELLER, J., CAMARRI, S. & IOLLO, A. 2009 Feedback control by low-order modelling of the laminar flow past a bluff body. *J. Fluid Mech.* **634**, 405–418.

WILLIAMS, M.O., KEVREKIDIS, I.G. & ROWLEY, C.W. 2015 A data–driven approximation of the koopman operator: extending dynamic mode decomposition. *J. Nonlinear Sci.* **25**, 1307–1346.

XIA, C., ZHANG, J., KERRIGAN, E.C. & RIGAS, G. 2023 Active flow control for bluff body drag reduction using reinforcement learning with partial measurements. Preprint, arXiv:2307.12650.

YUN, J. & LEE, J. 2022 Active proportional feedback control of turbulent flow over a circular cylinder with averaged velocity sensor. *Phys. Fluids* **34** (9), 095133.

ZACCARIAN, L. & TEEL, A.R. 2005 The L2 bumpless transfer problem for linear plants: its definition and solution. *Automatica* **41** (7), 1273–1280.

ZHOU, K., SALOMON, G. & WU, E. 1999 Balanced realization and model reduction for unstable systems. *Intl J. Robust Nonlinear Control* **9** (3), 183–198.