MIT Libraries | DSpace@MIT

MIT Open Access Articles

Bayesian optimization for active flow control

The MIT Faculty has made this article openly available. ***Please share*** how this access benefits you. Your story matters.

Citation: Blanchard, Antoine B., Cornejo Maceda, Guy Y., Fan, Dewei, Li, Yiqing, Zhou, Yu et al. 2022. "Bayesian optimization for active flow control."

As Published: https://doi.org/10.1007/s10409-021-01149-0

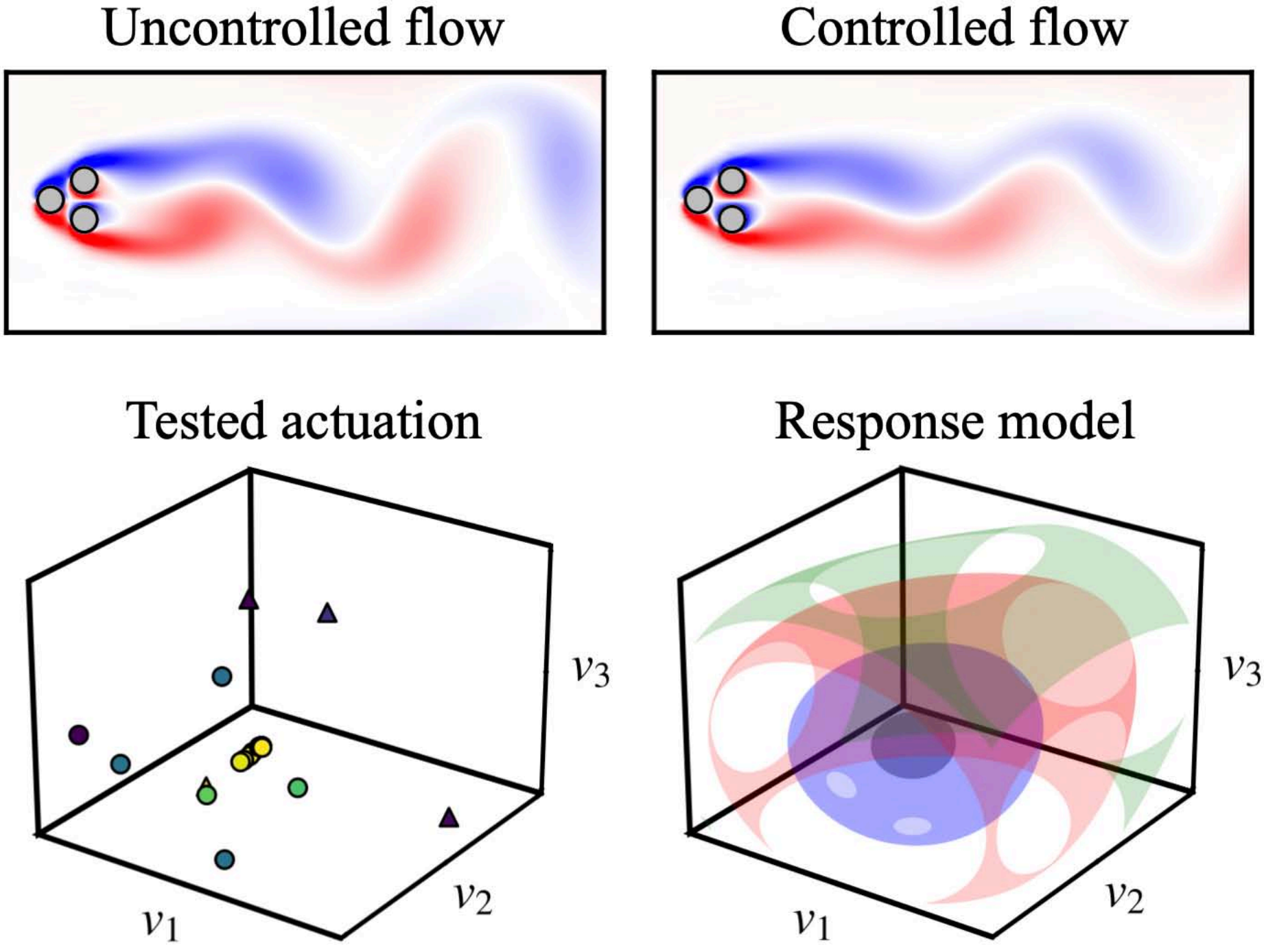
Publisher: The Chinese Society of Theoretical and Applied Mechanics; Institute of Mechanics, Chinese Academy of Sciences

Persistent URL: https://hdl.handle.net/1721.1/142072

Version: Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

Terms of use: Creative Commons Attribution-Noncommercial-Share Alike

Massachusetts Institute of Technology


Uncontrolled flow
Controlled flow
Tested actuation
Response model
v_1
v_2
v_3
v_1
v_2
v_3

Acta Mech. Sin. manuscript No.
(will be inserted by the editor)

Bayesian optimization for active flow control

Antoine B. Blanchard[1] · **Guy Y. Cornejo Maceda**[2] · **Dewei Fan**[3] · **Yiqing Li**[4] · **Yu Zhou**[3] · **Bernd R. Noack**[4] · **Themistoklis P. Sapsis**[1]

©*Acta Mechanica Sinica*, The Chinese Society of Theoretical and Applied Mechanics (CSTAM) 2020

Abstract A key question in flow control is that of the design of optimal controllers when the control space is high-dimensional and the experimental or computational budget is limited. We address this formidable challenge using a particular flavor of machine learning and present the first application of Bayesian optimization to the design of open-loop controllers for fluid flows. We consider a range of acquisition functions, including the recently introduced output-informed criteria of Blanchard and Sapsis (2021), and evaluate performance of the Bayesian algorithm in two iconic configurations for active flow control: computationally, with drag reduction in the fluidic pinball; and experimentally, with mixing enhancement in a turbulent jet. For these flows, we find that Bayesian optimization identifies optimal controllers at a fraction of the cost of other optimization strategies considered in previous studies. Bayesian optimization also provides, as a by-product of the optimization, a surrogate model for the latent cost function, which can be leveraged to paint a complete picture of the control landscape. The proposed methodology can be used to design open-loop controllers for virtually any complex flow and, therefore, has significant implications for active flow control at an industrial scale.

Keywords Bayesian optimization · Flow control · Drag reduction · Turbulence

✉ Bernd R. Noack
E-mail: bernd.noack@hit.edu.cn

✉ Themistoklis P. Sapsis
E-mail: sapsis@mit.edu

[1] Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, USA

[2] Université Paris-Saclay, CNRS, Laboratoire Interdisciplinaire des Sciences du Numérique, 91400 Orsay, France

[3] Center for Turbulence Control, Harbin Institute of Technology, Shenzhen 518058, China

[4] School of Mechanical Engineering and Automation, Harbin Institute of Technology, Shenzhen 518058, China

1 Introduction

Control of fluid flows, specifically turbulent flows, has been a long-standing challenge for the engineering community, and with the appearance of new applications in the transportation and energy industries its importance has never been higher [1,2]. The last century has seen significant progress toward a better understanding of complex fluid flows and, consequently, a number of tools have emerged that enable flow manipulation to a degree that is now nearly surgical [3,4,5]. The actuation policies thus designed have been used to serve a variety of purposes, including drag reduction for bluff-body flows [6,7], mixing enhancement for chemical and combustion processes [8,9,10], and transition delay in boundary-layer flows [11,12].

However, designing optimal controllers for flow control is no easy task, primarily because fluid flows are intrinsically high-dimensional and nonlinear. These two features conspire to render the task of designing controllers rather arduous, as many control strategies either do not scale well with the dimension of the system or are not applicable to dynamics that are strongly nonlinear [2,13]. The major bottleneck is due to the fact that in practice the amount of resources available for solving the design problem may be quite limited. Experts estimate a typical optimization budget to be roughly on the order of one

hundred water-tunnel experiments or high-fidelity flow simulations (such as direct numerical simulations or large-eddy simulations), and perhaps a few hundred to a thousand wind-tunnel experiments or Reynolds-averaged numerical simulations [14].

This conundrum has left many a practitioner scratching their head. How should one go about optimizing an actuation policy given the complexity of the system at hand and the limited budget available for experimentation? Right away these constraints severely undermine the appeal of several classical optimization algorithms. For example, grid search, random search, Latin hypercube sampling, and gradient-based methods all have tremendous advantages of their own (e.g., simple implementation, fast convergence, or optimal space coverage) but are fairly greedy when it comes to function queries [15,16]. Therefore, there is a need for more efficient optimization algorithms to address the challenge of flow control.

Part of the solution might be provided by machine learning (ML), whose incursion in the realm of flow control is relatively recent but has been quite noticeable [17,18,19]. This is due to the ability of ML algorithms to forego first-principle modeling and treat the system (i.e., the actuation, flow, and quantity of interest) as a generic input–output relationship that can be learned directly from data. Using a variety of ML tools such as support vector machines [20], neural networks [21,22,23], genetic algorithms [14,24], genetic programming [25,26] and reinforcement learning [27], pioneers in the field of ML control have been able to discover optimal control laws that have led to previously unknown physical phenomena as well as considerable improvement of the optimized quantities of interest.

Among the zoo of existing ML algorithms is one that has remained inconspicuous to the flow-control community despite the fact that it had been designed specifically to address the problem of optimizing expensive-to-evaluate black-box functions. This algorithm, known as Bayesian optimization (BO) [28,29,30], is a sequential strategy in which the decision of where to query the objective function next (i.e., what actuation command to try next) is determined by examining a surrogate model that has been trained on data from previous function queries. Bayesian optimization has earned its stripes because of its ability to capture prior beliefs about the behavior of the function and to compromise between exploration and exploitation before each function evaluation [28,29,30].

In this work, we present the first application of BO to flow control. We consider the case of open-loop actuation, which is the first step toward understanding the capabilities of the algorithm and its effect on the flow. We investigate several criteria for determining the algorithm's next move, including the output-informed criteria of Blanchard and Sapsis [31] which turn out to be key enablers of optimization efficiency. We also show how the surrogate model, a by-product of the algorithm, can be used to paint a complete picture of the control landscape, facilitating interpretability of the control laws. The proposed methodology has significant implications for industrial applications as it can be extended to any open-loop actuation and any complex flow.

We demonstrate the power of BO for flow control computationally by reducing drag in the fluidic pinball, and experimentally by enhancing mixing in a turbulent jet. The fluidic pinball, that is, flow past three circular cylinders arranged in an equilateral triangle, has recently established itself as a reference benchmark for flow control, owing to rich and surprising dynamics as well as countless actuation opportunities [32,33,34]. The turbulent jet, on the other hand, is a widely studied shear flow critical to many civil and military applications and for which the key challenge is to properly manipulate coherent structures in the flow in order to control entrainment and enhance mixing [35,26]. Any gain made for these two flows might have colossal implications for the industrial sector at large [2].

The remainder to the paper is structured as follows. We describe the optimization problem and Bayesian algorithm in Sect. 2, apply the algorithm to the fluidic pinball in Sect. 3 and the turbulent jet experiment in Sect. 4, and provide some conclusions in Sect. 5.

2 Bayesian optimization for flow control

2.1 Formulation of the problem

We consider the problem of minimizing a cost function $J : \mathbb{R}^d \longrightarrow \mathbb{R}$ over a compact set $\mathscr{B} \subset \mathbb{R}^d$ of actuation parameters:

$$\min_{\mathbf{b}\in\mathscr{B}} J(\mathbf{b}). \tag{1}$$

For example, J can be the average drag over a bluff body with $\mathbf{b}$ parametrizing blowing-and-suction actuators. In practical applications, evaluating the cost function J for a given actuation vector $\mathbf{b}$ often requires running a large-scale computer simulation or conducting a field experiment, which can be quite expensive. A high evaluation cost adds to the issue of opacity since in general the dependence of the cost function on the actuation parameters has no simple closed form, precluding the use of gradient-based optimization algorithms. In a field experiment, these difficulties are exacerbated by the presence of measurement noise, typically modeled as additive Gaussian noise:

$$q = J(\mathbf{b}) + \varepsilon, \quad \varepsilon \sim \mathscr{N}(0, \sigma_n^2). \tag{2}$$

These three constraints—expensiveness, opacity, noisiness—makes the problem in (1) a suitable candidate for BO [28,29]. A sketch of the algorithm is shown in Fig. 1. As is commonplace in optimization, BO first needs to be bootstrapped by evaluating the cost function for a small number (n_{init}) of randomly selected actuation vectors. This requires the practitioner to perform n_{init} experiments or computations and record the observed cost for each. The resulting actuation–cost pairs $\{\mathbf{b}_i, q_i\}_{i=1}^{n_{init}}$ make up the initial dataset $\mathscr{D}_0$ (red dots in Fig. 1). A surrogate model $\bar{J}$ (solid blue line in Fig. 1), whose mathematical form is at the discretion of the user, is then trained on the available data, providing a proxy for the unknown latent cost function J (dotted black line in Fig. 1).

With the initialization step complete, BO begins to explore the control space sequentially, one step at a time. At each iteration, BO determines the best controller to test out by minimizing an acquisition function $a : \mathbb{R}^d \longrightarrow \mathbb{R}$, shown in solid green in Fig. 1. The acquisition function leverages the surrogate model $\bar{J}$ and available data $\mathscr{D}_{n-1}$ to guide the algorithm as it combs through the control space. The controller recommended by the acquisition function (orange diamond in Fig. 1) can then be implemented in the laboratory by the practitioner. The newly collected actuation–cost pair $\{\mathbf{b}_n, q_n\}$ is then appended to the existing dataset, which is used to refine the surrogate model $\bar{J}$.

This process is repeated until the allocated budget for function queries (whether experimental or computational) is exhausted, at which point the BO algorithm is terminated and the surrogate model is used to make a final recommendation for the optimal controller. The final recommendation may be the global minimizer of the surrogate model itself,

$$\mathbf{b}^* = \underset{\mathbf{b}\in\mathscr{B}}{\arg\min}\, \bar{J}(\mathbf{b}), \tag{3}$$

or the controller that produced the smallest cost recorded during optimization,

$$\mathbf{b}^* = \underset{q_i\in\mathscr{D}_n}{\arg\min}\, q_i. \tag{4}$$

It is thus clear that the two critical ingredients in BO are the surrogate model on the one hand, and the acquisition function on the other. We review each in the next two sections. The surrogate model is important because it is used as a proxy for the latent cost function during the optimization and in subsequent analyses. The acquisition function is important because it is responsible for deciding which points or regions of the input space should be explored.

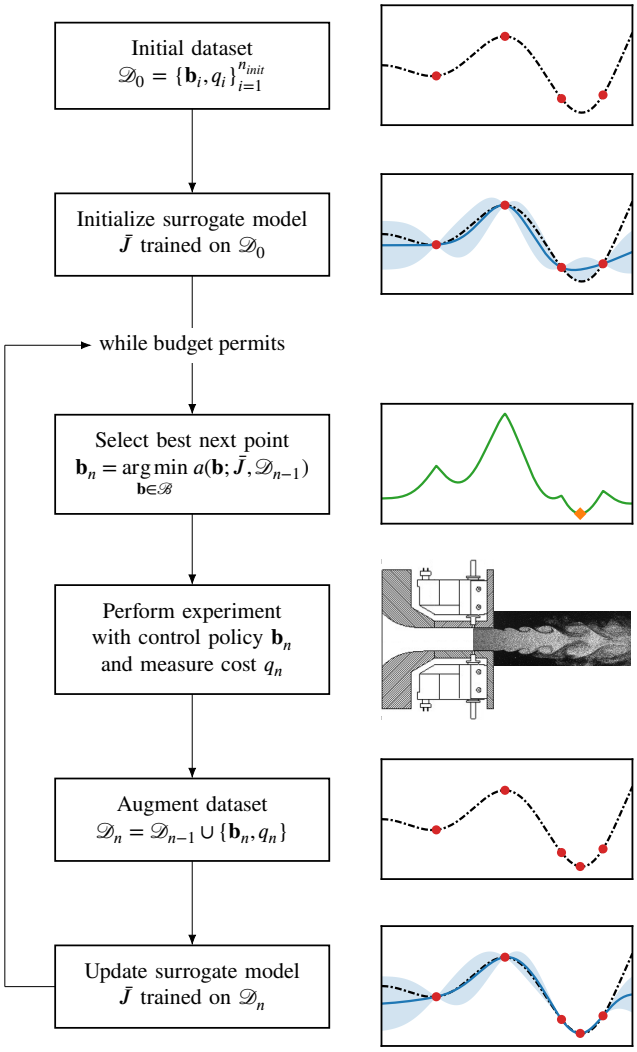


Fig. 1: Sketch of the Bayesian optimization algorithm for design of open-loop controllers for fluid flows.

2.2 Model selection

For the surrogate model, a number of approaches have been proposed in the past, with various levels of success [29]. The most popular among them is arguably Gaussian process (GP) regression as it offers a flexible non-parametric framework for modeling nonlinear black-box functions [36]. GP regression is particularly appropriate for BO because Gaussian processes are agnostic to the internal workings of the latent function, and they allow for quantification of uncertainty due to measurement noise [36,37,38].

In this work, we model the cost function J as a Gaussian process with constant mean m_0 and covariance $k(\mathbf{b}, \mathbf{b}'; \boldsymbol{\Theta})$, where $\boldsymbol{\Theta}$ is a vector of hyper-parameters that can be varied. For a dataset of actuation–cost pairs $\mathscr{D} = \{\mathbf{B}, \mathbf{q}\}$ (with the columns of $\mathbf{B} \in \mathbb{R}^{d\times n}$ containing the

actuation vectors, and the entries of $\mathbf{q} \in \mathbb{R}^n$ the corresponding cost values), the hyper-parameters $\{\boldsymbol{\Theta}, \sigma_n^2\}$ are optimized by maximum likelihood estimation [36]. Conditioned on $\mathscr{D}$, the trained model $\bar{J}(\mathbf{b})$ follows a normal distribution with posterior mean and variance

$$\mu(\mathbf{b}) = m_0 + k(\mathbf{b},\mathbf{B})\mathbf{K}^{-1}(\mathbf{q} - m_0), \tag{5a}$$

$$\sigma^2(\mathbf{b}) = k(\mathbf{b},\mathbf{b}) - k(\mathbf{b},\mathbf{B})\mathbf{K}^{-1}k(\mathbf{B},\mathbf{b}), \tag{5b}$$

respectively, where $\mathbf{K} = k(\mathbf{B},\mathbf{B}) + \sigma_n^2\mathbf{I}$. Equation (5a) can be used to predict the value of the latent cost function J for any actuation vector $\mathbf{b}$, and (5b) to quantify uncertainty in prediction at that point [36].

The covariance function $k(\mathbf{b},\mathbf{b}';\boldsymbol{\Theta})$ plays a key role in GP regression as it encodes prior belief or domain expertise one may have about the optimization landscape (e.g., symmetry, invariance, or periodicity). In the absence of any domain-specific knowledge, it has become customary to assume Lipschitz-continuity of the objective function and, consequently, to rely on a generic squared-exponential kernel with automatic relevance determination,

$$k(\mathbf{b},\mathbf{b}';\sigma_f^2,\boldsymbol{\Lambda}) = \sigma_f^2 \exp[-(\mathbf{b}-\mathbf{b}')^\mathsf{T}\boldsymbol{\Lambda}^{-1}(\mathbf{b}-\mathbf{b}')/2], \tag{6}$$

where σ_f is a scaling parameter and $\boldsymbol{\Lambda}$ is a diagonal matrix containing the lengthscales for each input variable [36,28]. (Together, σ_f and $\boldsymbol{\Lambda}$ make up the hyper-parameters $\boldsymbol{\Theta}$ of the covariance function.)

2.3 Acquisition functions

The other fundamental component in BO is the acquisition function. Its role is to encourage sampling of those regions in the input space that are thought to harbor the global minimizer(s) of the latent function. The key issue is for the acquisition function to strike a balance between exploration (i.e., visiting regions that have not been explored before) and exploitation (i.e., focusing on regions where the surrogate model predicts small values).

Common choices for the acquisition function include the probability of improvement (PI) [15],

$$a_{PI}(\mathbf{b}) = \Phi(\lambda(\mathbf{b})), \quad \lambda(\mathbf{b}) = [y^* - \mu(\mathbf{b}) - \kappa]/\sigma(\mathbf{b}), \tag{7}$$

the expected improvement (EI) [15],

$$a_{EI}(\mathbf{b}) = \sigma(\mathbf{b})\left[\lambda(\mathbf{b})\Phi(\lambda(\mathbf{b})) + \phi(\lambda(\mathbf{b}))\right], \tag{8}$$

and the lower confidence bound (LCB) [39],

$$a_{LCB}(\mathbf{b}) = \mu(\mathbf{b}) - \kappa\sigma(\mathbf{b}). \tag{9}$$

In the above formulas, Φ and ϕ are the cumulative and probability density functions of the standard normal distribution, y^* denotes the current best observation, and $\kappa \geq 0$ is a user-specified parameter that balances exploration (large κ) and exploitation (small κ). These acquisition functions have become popular because their implementation is straightforward, evaluation is inexpensive, and their gradients can be computed in closed form for use in gradient-based optimizers.

In this work, we also consider the integrated variance reduction for BO (IVR-BO) [31],

$$a_{IVR\text{-}BO}(\mathbf{b}) = \mu(\mathbf{b}) - \frac{\kappa}{\sigma^2(\mathbf{b})}\int \mathrm{cov}^2(\mathbf{b},\mathbf{b}')\,\mathrm{d}\mathbf{b}', \tag{10}$$

which repurposes the purely explorative IVR criterion of Sacks et al. [40] into one that is more aggressive towards minima. The integral appearing in (10) and its gradients can be computed analytically for the RBF kernel [31], allowing IVR-BO to retain the three important features discussed earlier for PI, EI and LCB.

Recently, Blanchard and Sapsis [31] introduced a new class of acquisition functions that accelerate convergence of the BO algorithm in situations where certain regions of the input space have a much larger impact on the output of the latent function than others. Key to this approach is the *likelihood ratio*, a quantity that can be seamlessly incorporated as a sampling weight into several of the acquisition functions commonly used in practice and which encourages exploration of regions associated with extreme output values. Defined as

$$w(\mathbf{b}) = \frac{p_{\mathbf{b}}(\mathbf{b})}{p_\mu(\mu(\mathbf{b}))}, \tag{11}$$

the likelihood ratio assigns to each point in the input space a measure of relevance by weighting how likely that point is to be observed "in the wild" (through the input density $p_{\mathbf{b}}$) against its expected impact on the magnitude of the output (through the output density p_μ). As such, the likelihood ratio serves as an *attention mechanism* which steers the algorithm toward the extremes [31,41].

As discussed in Blanchard and Sapsis [31], the likelihood ratio can be integrated into LCB and IVR-BO where it acts as a sampling weight or variance regularizer. The resulting likelihood-weighted (LW) acquisition functions,

$$a_{LCB\text{-}LW}(\mathbf{b}) = \mu(\mathbf{b}) - \kappa\sigma(\mathbf{b})w(\mathbf{b}) \tag{12}$$

and

$$a_{IVR\text{-}LWBO}(\mathbf{b}) = \mu(\mathbf{b}) - \frac{\kappa}{\sigma^2(\mathbf{b})}\int \mathrm{cov}^2(\mathbf{b},\mathbf{b}')w(\mathbf{b})\,\mathrm{d}\mathbf{b}', \tag{13}$$

specifically target extreme minima of the latent function. As in previous work [31,41,42], we approximate the likelihood ratio with a Gaussian mixture model in order to make the integral in (13) analytic.

2.4 Optimization protocol

In the following sections, we apply the BO algorithm to two canonical configurations for active flow control: computationally with the fluidic pinball [32] and experimentally with a turbulent jet [26]. For each problem, the BO algorithm is initialized with $n_{init} = d + 1$ actuation–cost pairs drawn by Latin hypercube sampling (LHS). As for the acquisition functions, we use $\kappa = 0.01$ in EI and PI, and $\kappa = 1$ in LCB(-LW) and IVR-(LW)BO. The likelihood ratio is approximated with two Gaussian components.

The metric of success is the observation regret

$$r(n) = \min_{q_i \in \mathscr{D}_n} q_i, \tag{14}$$

which returns the smallest cost measured in the laboratory after n iterations. The element of randomness resulting from the initial LHS draw is averaged out by repeating the BO procedure n_e times, each time with a different choice of initial points, and reporting the mean observation regret. Variability across BO rounds is quantified with the standard deviation. Our implementation uses the `gpsearch` package, which is available on GitHub[1].

3 Drag reduction in the fluidic pinball

3.1 Physical model

The physical model for the fluidic pinball is identical to that considered by Deng et al. [32] and Cornejo Maceda et al. [34], and is shown schematically in Fig. 2. A Newtonian fluid with constant density ρ and kinematic viscosity ν flows with uniform velocity $U\mathbf{e}_x$ past a cluster of three circular cylinders with diameter D. The cylinder centers are arranged in an equilateral triangle with sidelength $3D/2$ and one of the edges perpendicular to the incoming flow. We refer to the forwardmost cylinder as the "front" cylinder, and the aftmost cylinders as the "bottom" (port) and "top" (starboard) cylinders. (We will also use the alternative numbering $i \in \{1,2,3\}$ for the front, bottom, and top cylinders, respectively.) Each cylinder is actuated independently by rigid-body rotation of the outer "shell" with constant angular velocity $2U_i/D$.

The flow is governed by the incompressible Navier–Stokes equations, written in dimensionless form as

$$\frac{\partial \mathbf{v}}{\partial \tau} + \mathbf{v}\cdot\nabla\mathbf{v} = -\nabla p + \frac{1}{Re}\nabla^2\mathbf{v}, \tag{15a}$$

$$\nabla\cdot\mathbf{v} = 0, \tag{15b}$$

[1] https://github.com/ablancha/gpsearch

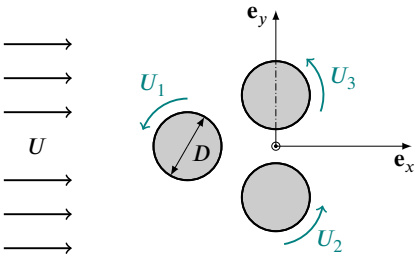


Fig. 2: Fluidic pinball in cross-flow.

subject to the boundary conditions

$$\lim_{x,y\to\infty} = \mathbf{e}_x \tag{16a}$$

in the far field, and

$$\mathbf{v}|_{\Gamma_i} = v_i\,\mathbf{e}_\theta \tag{16b}$$

on the surface of the ith cylinder Γ_i. Time, length, and velocity (including cylinder circumferential velocity $U_i = v_i U$) have been scaled with cylinder diameter D and free-stream velocity U. The Reynolds number is based on cylinder diameter and defined as $Re = UD/\nu$. The lift, drag, and torque coefficients for the ith cylinder are given by

$$C_{L,i} = \int_{\Gamma_i} (\boldsymbol{\sigma}\cdot\mathbf{n})\cdot\mathbf{e}_y \,\mathrm{d}\Gamma, \tag{17a}$$

$$C_{D,i} = \int_{\Gamma_i} (\boldsymbol{\sigma}\cdot\mathbf{n})\cdot\mathbf{e}_x \,\mathrm{d}\Gamma, \tag{17b}$$

$$C_{T,i} = \int_{\Gamma_i} (\boldsymbol{\sigma}\cdot\mathbf{n})\times(\mathbf{e}_r/2) \,\mathrm{d}\Gamma, \tag{17c}$$

respectively, where $\boldsymbol{\sigma}$ is the dimensionless stress tensor and $\mathbf{n}$ is the outward unit normal to the cylinder surface. We also define a dimensionless actuation vector,

$$\mathbf{b} = \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix}^\mathsf{T}. \tag{18}$$

The uncontrolled flow (i.e., no cylinder rotation) corresponds to $\mathbf{b} = \mathbf{0}$. The instantaneous actuation power imparted to the cylinders can be expressed in dimensionless form as

$$J_T = -2\mathbf{b}\cdot\mathbf{T}, \tag{19}$$

with $\mathbf{T} = \begin{bmatrix} C_{T,1} & C_{T,2} & C_{T,3} \end{bmatrix}^\mathsf{T}$ a vector of torque coefficients.

Absent any actuation, the flow past the fluidic pinball is known to undergo two bifurcations in the range $0 < Re \leq 100$ [32]. The steady symmetric solution becomes linearly unstable at $Re \approx 18$ through a Hopf bifurcation, giving way to a globally attracting limit cycle. At $Re \approx 68$, a pitchfork bifurcation occurs and two mirrored limit cycles emerge. They co-exist until $Re \approx 104$, at which point a Neimark–Säcker bifurcation ushers in a regime of quasi-periodic oscillations and puts the system on a route to chaos [32]. In what follows, we consider the cases $Re = 30$ and $Re = 100$.

3.2 Computational approach

The flow is computed using the open-source Navier–Stokes solver Nek5000 [43]. The computational domain is identical to that used by Deng et al. [32], extending $12D$ is the cross-stream direction and $26D$ in the stream-wise direction, with the centers of the aft cylinders lying on the $x = 0$ axis and equidistantly from the sidewalls. As in Deng et al. [32], we specify a Dirichlet boundary condition ($\mathbf{v} = \mathbf{e}_x$) at the inlet, top, and bottom boundaries of the domain. A stress-free boundary condition ($\boldsymbol{\sigma} \cdot \mathbf{n} = 0$) is specified at the outlet.

Our production runs use a mesh with 440 spectral elements (Fig. 3), polynomial order $N = 7$, and time-step size $\Delta\tau = 0.01$. As shown in Appendix A, this choice of parameters ensures adequate convergence of the computed solutions. Our computational approach is further validated against the finite-element solver used by Deng et al. [32] and Cornejo Maceda et al. [34]. The results in Appendix A show excellent agreement between the two approaches.

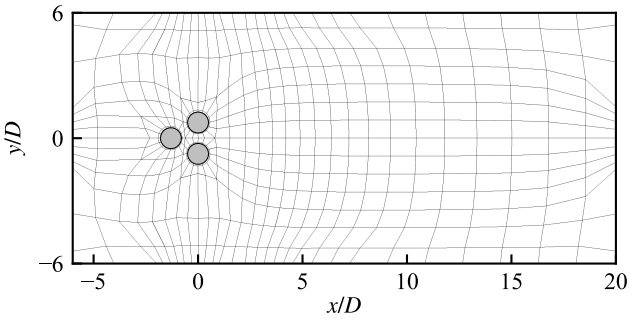


Fig. 3: Spectral-element discretization for the fluidic pinball. Each macro-element is further discretized using $N + 1$ Gauss–Lobatto–Legendre quadrature points, with N the polynomial order.

3.3 Optimization results

We use BO to uncover the optimal controller that yields maximum drag reduction in the fluid pinball. To this effect, the objective function comprises the total drag coefficient $C_D = \sum_i C_{D,i}$ (quantifying the parasitic towing power) and the dimensionless actuation power J_T, and is specified as

$$J(\mathbf{b}) = \frac{1}{t_h} \int_0^{t_h} (C_D + J_T)\, \mathrm{d}t, \tag{20}$$

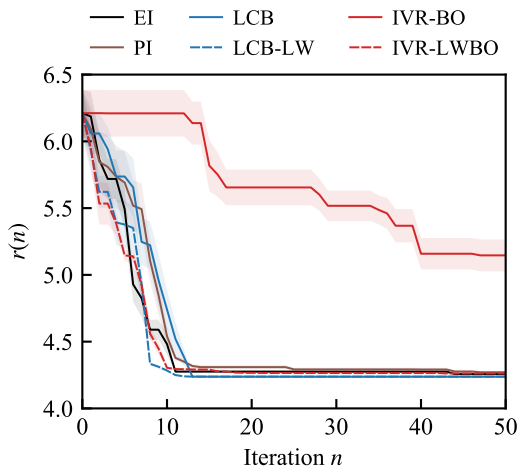
where t_h denotes the time horizon of interest. Minimizing (20) has the effect of maximally reducing drag while keeping the overall actuation cost as small as possible. In what follows, we use $t_h = 200$ to allow sufficient time for the system to settle down. The flow is initialized on the stable attractor of the uncontrolled configuration (specifically, the globally attracting limit cycle for $Re = 30$ and the downward-deflected limit cycle for $Re = 100$). The search space is specified as $\mathscr{B} = [-2, 2]^3$, and results are collected across $n_e = 5$ BO rounds.

For the two Re values and the six acquisition functions considered, Fig. 4 shows the evolution of the mean regret as the BO algorithm samples the input space. Several remarks are in order. First, we note that all the acquisition functions deliver comparable performance, except for IVR-BO which performs noticeably worse. (This is consistent with the results in Ref. [31].) Second, the likelihood-weighted acquisition functions provide a visible advantage over their unweighted counterparts, evidencing the utility of the likelihood ratio in BO. Third, Fig. 4 shows that no more than a dozen function queries are needed for EI, PI, LCB(-LW) and IVR-LWBO to identify a nearly optimal controller. This is far fewer than what has been previously reported for the explorative gradient method (EGM) for the same flow, which required at least 30 functions evaluations to achieve convergence [14].

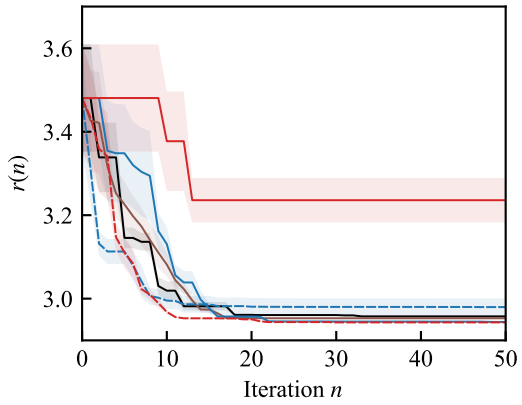
Table 1 shows the best controller (4) identified by each acquisition functions across all five BO rounds. (The optimal controllers in Table 1 do not necessarily emanate from the same set of initial LHS samples). The optimal actuation policy, found by all criteria but IVR-BO, corresponds to nearly symmetric Coanda forcing at $Re = 30$ and asymmetric Coanda forcing at $Re = 100$. Consistent with previous work [14,44,34], the symmetry (or lack thereof) of the optimal actuation policy is not a coincidence, and is indeed related to the symmetry (or lack thereof) of the mean flow on either side of the pitchfork bifurcation. We note in passing that due to the symmetry of the geometry and objective function, the mirror image of any optimal controller $\mathbf{b}^*$ is also optimal provided the initial condition for the flow is mirrored as well.

For the two Reynolds numbers investigated, the smallest value of the objective function recorded by the algorithm across all BO rounds is obtained with LCB-LW (Table 1). For the optimal actuation policy thus identified, Fig. 5 shows that the trajectory settles into a limit cycle which is markedly different from that for the uncontrolled flow. In addition to substantial drag reduction (about 28% at $Re = 30$ and 63% at $Re = 100$), the optimally actuated flow is characterized by a slight increase in Strouhal frequency (from 0.086 to about 0.090 for $Re = 30$, and from 0.116 to about 0.120 for $Re = 100$), accompanied by visible changes in the wake (Fig. 5).

Figure 6 shows the outcome of the sampling process after 50 iterations for two of the best-performing acquisition functions considered in this work. (The same set of initial LHS points for both is used to allow meaningful comparisons.) While EI and LCB-LW



(a) $Re = 30$



(b) $Re = 100$

Fig. 4: For the fluidic pinball, evolution of the mean regret as more samples are being collected.

ultimately converge to the same optimal controller, they each have their own way of scouting the parameter space. In particular, Fig. 6 shows that EI has a stronger propensity for exploration, whereas LCB-LW "zeroes in" much more aggressively on regions of the space that are thought to harbor the global minimum. (This is precisely the purpose of the likelihood ratio.)

Compared to other classical optimization algorithms, BO has the significant advantage that it provides, as a by-product of the optimization, a surrogate model for the latent cost function. With enough data to be trained on, the surrogate model can be leveraged to paint a complete picture of the control landscape. The bottom panel in Fig. 6 shows that the control landscape at $Re = 30$ is topologically simple, with the GP model producing isosurfaces that seem to be contained one inside the other. At $Re = 100$, the control landscape obtained with EI is also topologically simple, but the one obtained with LCB-LW is not. This is because EI is more explorative than LCB-LW, with the

	v_1^*	v_2^*	v_3^*	$J(\mathbf{b}^*)$
EI	0.0063	0.6002	-0.5813	4.2383
PI	0.0040	0.5979	-0.5931	4.2381
LCB	0.0014	0.6156	-0.6003	4.2376
LCB-LW	0.0024	0.6172	-0.6006	4.2376
IVR-BO	0.1421	0.1299	-0.1118	4.7703
IVR-LWBO	0.0154	0.6019	-0.5994	4.2380
Uncontrolled	-	-	-	5.0313

(a) $Re = 30$

	v_1^*	v_2^*	v_3^*	$J(\mathbf{b}^*)$
EI	0.0334	1.3732	-1.1008	2.9466
PI	0.0521	1.3237	-1.1775	2.9463
LCB	0.0594	1.3501	-1.1261	2.9417
LCB-LW	-0.0026	1.5238	-1.3478	2.9328
IVR-BO	0.1863	1.4192	-0.9077	3.0038
IVR-LWBO	0.0383	1.4093	-1.1881	2.9396
Uncontrolled	-	-	-	3.8423

(b) $Re = 100$

Table 1: For the fluidic pinball, optimal controller $\mathbf{b}^* = [v_1^* \ v_2^* \ v_3^*]^\mathsf{T}$ found across all five BO rounds and associated cost $J(\mathbf{b}^*)$.

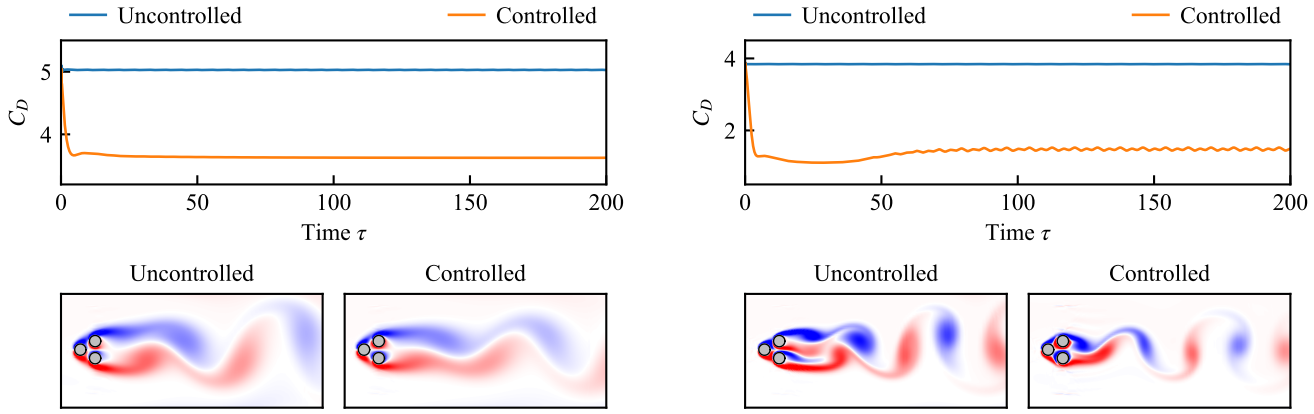
latter spending more resources in the vicinity of the global minimizer than the former. As the result, the GP model for LCB-LW is more accurate than that for EI close to the minimizer (leading to identification of a better actuation policy), but less accurate away from it. For a detailed discussion of the trade-off between exploration and exploitation in BO, we refer the reader to Refs. [28] and [42].

4 Mixing enhancement in a turbulent jet

4.1 Experimental setup

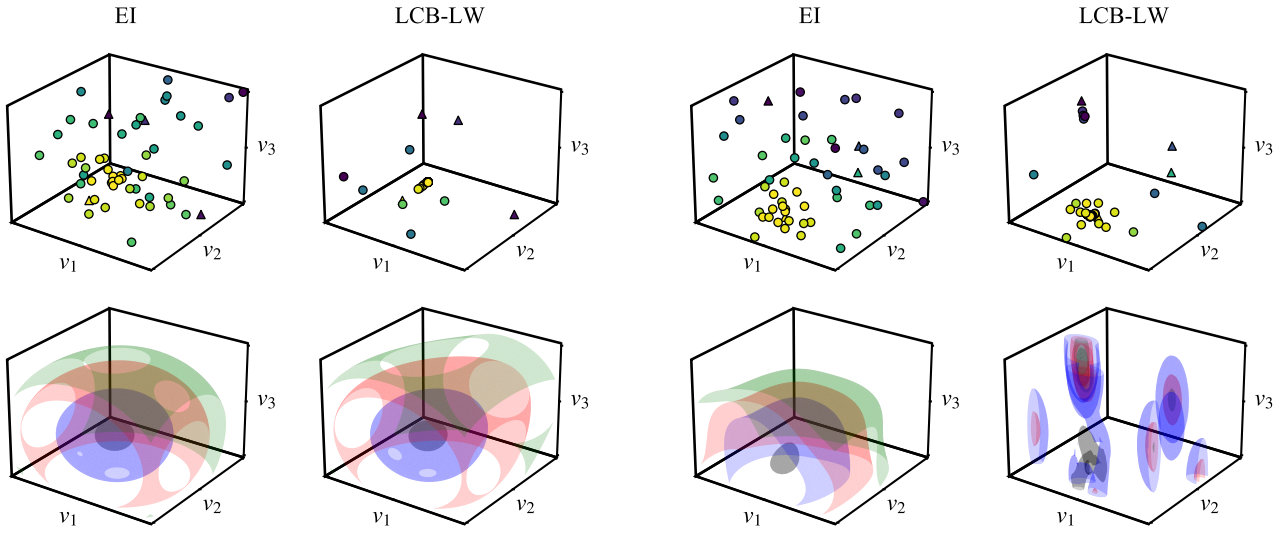
The experimental setup is identical to that in Zhou et al. [26], and is shown schematically in Fig. 7. The jet facility includes an air supply system, the turbulent jet itself, and a set of six jet actuators which are described below. The Reynolds number is defined as $Re = \bar{U}_j D/\nu$, where $\bar{U}_j$ is the time-averaged centerline velocity at the nozzle exit, D is the nozzle diameter, and ν is the kinematic viscosity of air. Our experiments use $Re = 8000$ and $D = 20$ mm. As shown in Fig. 7(a,c), we define a Cartesian coordinate system (x, y, z) with origin at the center of the jet exit and the x-axis pointing in the direction of flow.

Actuation is performed with six independent minijets with orifice diameter $d = 1$ mm located 17 mm upstream of the nozzle exit. The position of the ith minijet is given by $x_i = -0.85D$, $y_i = (D/2)\cos\theta_i$, and $z_i = (D/2)\sin\theta_i$, where $\theta_i = (i-1)\pi/3$ and $i \in \{1, \ldots, 6\}$ (see Fig. 7b). Each minijet is connected to a channel consisting of an electromagnetic valve, a mass flow meter, and a



(a) $Re = 30$ (b) $Re = 100$

Fig. 5: For the fluidic pinball, time series of the drag coefficient and vorticity distribution at $\tau = 200$ for the optimal controller obtained with LCB-LW (cf. Table 1) and the uncontrolled flow.



(a) $Re = 30$ (b) $Re = 100$

Fig. 6: For the fluidic pinball, outcome of the BO algorithm after 50 iterations using EI and LCB-LW. Top panel shows the sampling strategy in the control space with initial LHS samples shown with triangles and optimized samples with circles; each marker is colored according to the value of the objective function, with lighter color indicating smaller cost. Bottom panel shows isosurfaces produced by the GP model; level sets are shown at $\mu = 4.5$ (black), 6.5 (blue), 10 (red) and 12.5 (green) for $Re = 30$, and at $\mu = 3$ (black), 4 (blue), 5 (red) and 6 (green) for $Re = 100$.

flow-limiting valve. The flow-limiting valve controls the mass flow rate of the minijet, which is measured by the mass flow meter. The frequency and duty cycle of each minijet is controlled by the electromagnetic valve operated in on/off mode. The valves ensure that the frequency of the minijet does not exceed 500 Hz, more than three times the preferred-mode frequency of the unforced jet at $Re = 8000$ ($f_0 = 135$ Hz), and more than seven times the actuation frequency used in the present study ($f_a = 67$ Hz).

The jet exit velocity at $(0, -D/4, 0)$ is measured with a tungsten wire with diameter 5 μm operated on a constant temperature circuit (Dantec Streamline) at an overheat ratio of 1.8. The centerline jet velocity at $x/D = 5$ is monitored with a second hot-wire. Both hot-wires are calibrated at the jet exit using a Pitot tube connected to a micromanometer

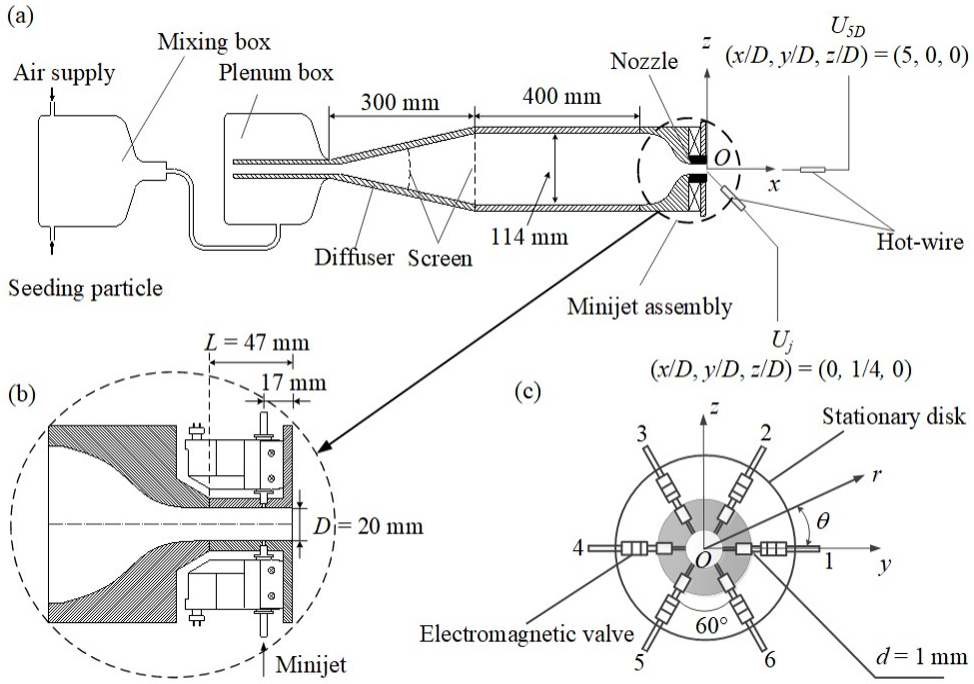


Fig. 7: Schematic of the experimental apparatus: (a) main jet facility; (b) minijet assembly; (c) minijet arrangement.

(Furness Controls FCO510). The experimental uncertainty of the hot-wire measurements is estimated to be less than 2%.

Flow visualization is performed using a planar high-speed particle imaging velocimetry (PIV) system, which includes a high-speed camera (PhantomV641, double frames, with a resolution of 2560 × 1600 pixels) and a pulsed laser source (Litron LDY304- PIV, Nd:YLF, 120 mJ/pulse). An oil droplet generator (TSI MCM-30) is used to generate a fog for flow seeding. The seeding particles are fed into a mixing chamber (Fig. 7a) where they mix with air. Flow illumination is done by a one-millimeter-thick laser sheet generated by the pulsed laser via a cylindrical lens. Measurements are made in the x-z and x-y planes of the main jet. Particle images are captured at a sampling rate of 405 Hz, corresponding to $3f_0$ and about $6f_a$.

4.2 Optimization results

We apply the BO algorithm to find a set of actuation parameters that maximizes mixing of the turbulent jet. The objective function is defined as

$$J(\mathbf{b}) = \bar{U}_{5D}/\bar{U}_j, \tag{21}$$

where $\bar{U}_{5D}$ denotes the time-averaged centerline velocity of the jet at $x/D = 5$. As discussed in Perumal and Zhou [45] and Zhou et al. [26], minimizing the cost function (21) is equivalent to maximizing the decay rate $K = 1 - J$ of the mean centerline velocity, which is a good indicator of the jet's mixing efficacy. Time averages are computed over 15 seconds. The open-loop control law for the ith minijet is specified as $a_i + \sin(2\pi f_a t + \phi_i)$, with values above and below 0.1 corresponding to the minijet being on and off, respectively. Therefore, the actuation vector

$$\mathbf{b} = \left[\{a_i, \phi_i\}_{i=1}^{6}\right]^{\mathsf{T}} \tag{22}$$

contains no fewer than 12 actuation parameters to be optimized. The search space $\mathscr{B}$ is such that $a_i \in [-1, 1]$ and $\phi_i \in [0, 2\pi]$. Results are collected across $n_e = 3$ BO rounds.

For three of the six acquisition functions considered earlier (EI, IVR-BO, IVR-LWBO), Figure 8 shows the evolution of the mean regret as the BO algorithm explores the control space. We note that IVR-LWBO is able to identify an optimal controller more rapidly than EI, which appears to reach a plateau after about 250 iterations, and IVR-BO, illustrating the benefits provided by the likelihood ratio in high-dimensional spaces. With IVR-LWBO, one can expect the optimal controller to yield a cost value of about 0.28, some 8% less than the best controller found by genetic programming in Ref. [26], notwithstanding the higher level of sophistication of the control laws in that work. We also note that BO requires fewer function evaluations to converge (about 200 for IVR-LWBO), as opposed to about one thousand for genetic programming in Ref. [26]).

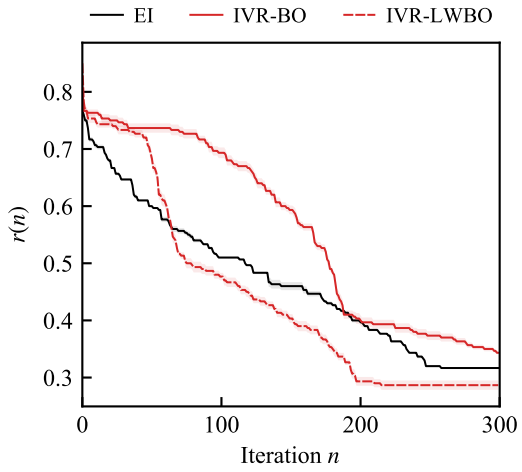


Fig. 8: For the turbulent jet, evolution of the mean regret as more samples are being collected.

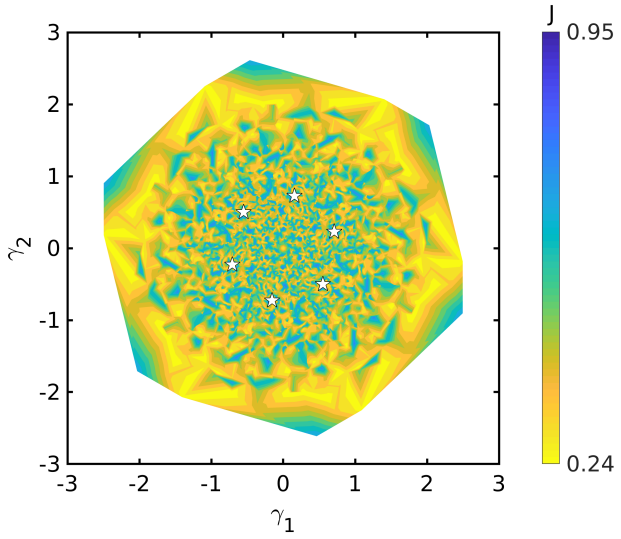


Fig. 9: For the turbulent jet, proximity maps generated by EI, IVR-BO, and IVR-LWBO, with white stars denoting the optimal actuation policy.

The control landscape is shown in Fig. 9 following the approach of Duriez et al. [17]. The dataset used to construct the control landscape consists of all actuation–cost pairs $\{\mathbf{b}_n, q_n\}$ collected by EI, IVR-BO, and IVR-LWBO across all three BO rounds, and enriched six-fold by rotational symmetry of the actuation vectors. The original 12-dimensional control space is displayed as a two-dimensional proximity map using classical multi-dimensional scaling (CMDS). Every actuation vector $\mathbf{b}_n$ is associated with a feature coordinate γ_n such that the Euclidean distance between actuation vectors is optimally preserved in the two-dimensional feature plane (i.e., $\|\mathbf{b}_n - \mathbf{b}_m\| \approx \|\gamma_n - \gamma_m\|$). The cost function is interpolated in the feature space, with lighter color indicating enhanced mixing and darker color corresponding to the unforced flow. Figure 10 shows the learning curve of the algorithm in the control landscape. The EI criterion is seen to quickly converge to the minimum while IVR-BO and IVR-LWBO explore a larger portion of the control landscape. Figure 10 also shows that the optimization leads to all six sectors of rotation-symmetric control laws.

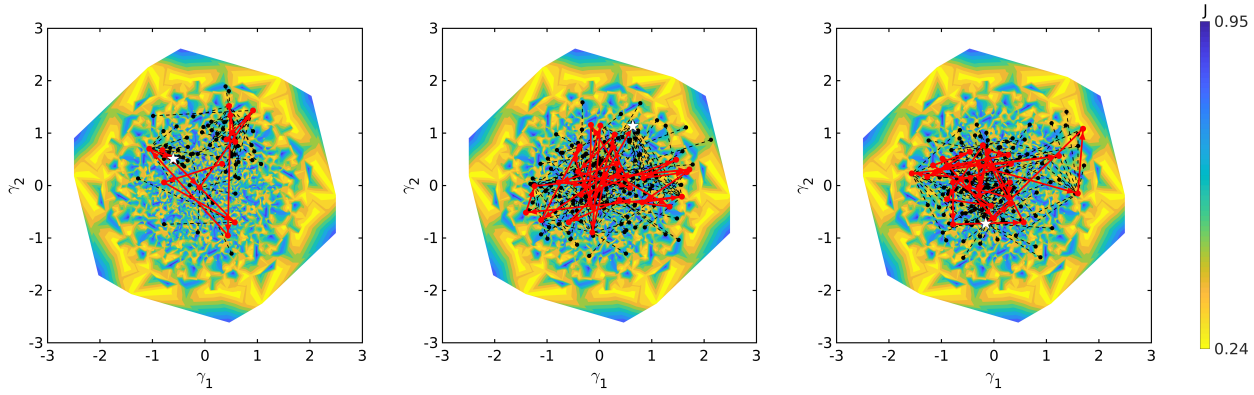
Interestingly, PIV measurements reveal that the optimal controller found by IVR-LWBO gives rise to a flow that resembles the "flapping" motion found by Zhou et al. [26] with a different control law (see Fig. 11). The optimal actuation policy deliberately breaks the symmetry of the uncontrolled flow, thus causing column oscillations of the jet and, consequently, enhancing mixing. A thorough description of these effects is given in Ref. [26].

5 Conclusions

The purpose of this work was to pioneer the use of BO for the design of optimal open-loop controllers in fluid flows. Our investigation reveals that, despite having received little to no attention from the flow-control community, BO proves to be an ideal candidate for solving the design problem. This is because BO is agnostic to the details of the interplay between the actuation, the flow, and the quantity of interest and treats the control plant as a black box which is expensive to evaluate and whose internal structure is unknown. This framework mirrors the sort of conditions to which the experimentalist and the numericist are routinely confronted.

We have examined how the two pillars on which BO rests (i.e., the surrogate model and the acquisition function) play an important role in the context of flow control. On the one hand, the surrogate model provides the complete cartography of the control landscape, which can be used for a posteriori design and analysis tasks, and offers valuable insight into the physics of the controlled flow. On the other hand, the acquisition function is responsible for guiding the algorithm on its journey through the control space and, as such, has a considerable impact on the rate of convergence of the algorithm. Our results provide further evidence in favor of the output-informed approach of Blanchard and Sapsis [31] which was found to outperform the competition in the examples considered.

The power of BO for flow control was first demonstrated computationally with the fluidic pinball. A geometrically simple but dynamically complex flow, the fluidic pinball has been the subject of several attempts aimed at reducing the total drag imparted to the array of cylinders. In this work, we showed that BO identifies the same optimal actuation policies as those reported



(a) EI (b) IVR-BO (c) IVR-LWBO

Fig. 10: For the turbulent jet, optimization routes of EI, IVR-BO, and IVR-LWBO, with red dots denoting newly found minima, and black dots connecting the next suboptimal samples.

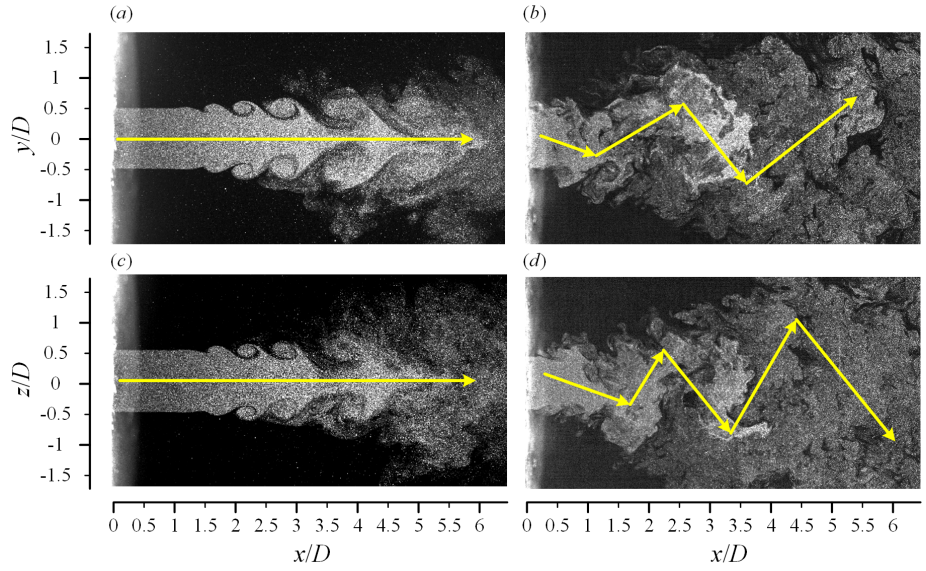


Fig. 11: Flow visualization for the turbulent jet, with arrows indicating major directions of flow oscillation: (a) uncontrolled flow and (b) controlled flow in the $z = 0$ plane; (c) uncontrolled flow and (d) controlled flow in the $y = 0$ plane.

previously, though at a fraction of the cost. We then ventured in the laboratory and leveraged BO to enhance mixing in a turbulent jet using a set of six minijet actuators. Despite the added complexity of the apparatus and increased dimensionality of the control space, BO was able to identify a simple control law just as effective as that reported in Ref. [26].

This work has laid down the groundwork for using BO with more sophisticated control laws and more complex flow configurations. One possible application is the discovery of nonlinear actuation mechanisms for multi-input, multi-output (MIMO) experiments, for which a simple data-driven realization was proposed by Nair et al. [46] who employed a low-dimensional cluster-based parametrization to optimize MIMO control laws. Another avenue is the incorporation of multi-fidelity capabilities in which computation and experiment would operate in tandem to achieve faster identification of optimal controllers. For these reasons, we expect BO—and more generally ML—to have a significant influence on the field of flow control in the foreseeable future.

Acknowledgements ABB and TPS gratefully acknowledge support from a MathWorks Faculty Research Innovation Fellowship at MIT. GYCM and BRN acknowledge funding by the French National Research Agency (ANR) via the grant 'FlowCon' (ANR-17-ASTR-0022). The thesis of GYCM was supported by

LIMSI/CNRS and Paris-Sud University. BRN acknowledges support from the National Science Foundation of China (NSFC) through grant 12172109. YZ wishes to acknowledge support given to him from NSFC through grants 11632006, 91752109 and 91952204.

A Validation of the computational approach for the fluidic pinball

To determine appropriate values for the time-step size $\Delta\tau$ and polynomial order N, we consider the uncontrolled case ($\mathbf{b} = \mathbf{0}$) at $Re = 100$ with initial condition $\mathbf{v}(x,y) = [1 + 10^{-3}\sin(y)]\mathbf{e}_x$. The slight asymmetry in the initial condition allows vortex shedding to develop more rapidly than if one were to rely on small asymmetries in the numerics. The flow is evolved for 2000 time units and only the last 500 time units of that interval are retained to compute the statistics. We report the temporal mean, standard deviation (std), and peak-to-peak amplitude (amp) of C_D and C_L, as well as the Strouhal frequency St (i.e., the dominant frequency of C_L). Table 2 shows that the polynomial order has a larger effect on the statistics than the time-step size. None of the computational parameters has a significant effect on the Strouhal frequency. The results show that, for the present purposes, adequate convergence is achieved by specifying $\Delta\tau = 0.02$ and $N = 7$. In our production runs, however, we use a smaller time-step size ($\Delta\tau = 0.01$) to alleviate the possibility of solution blow-up when the actuation vector is non-zero.

For $\Delta\tau = 0.01$ and $N = 7$, we compare our computational approach with the finite-element solver used by Deng et al. [32]. We consider the actuation configuration described in Cornejo Maceda et al. [34] in which the front cylinder is not allowed to rotate ($v_1 = 0$) and the aft cylinders rotate with equal and opposite angular velocities ($v_2 = -v_3$). As in Cornejo Maceda et al. [34], the Reynolds number is 100 and the flow is effected for 1000 time units starting from an initial state on the limit cycle of the uncontrolled configuration. We report the temporal mean for C_D and J_T. Figure 12 shows good agreement between our approach and that of Cornejo Maceda et al. [34,47].

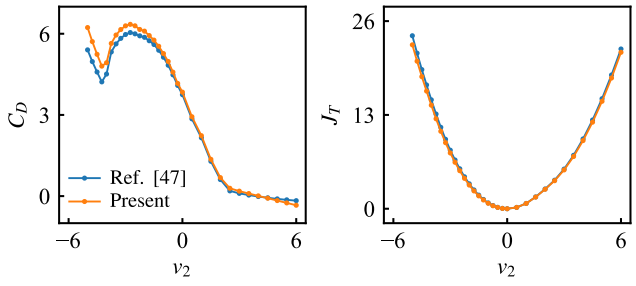


Fig. 12: Long-time averages of C_D and J_T for the symmetric actuation configuration of Cornejo Maceda et al. [34] at $Re = 100$.

References

1. M. Gad-el Hak, *Flow Control: Passive, Active, and Reactive Flow Management* (Cambridge University Press, 2007)
2. S.L. Brunton, B.R. Noack, Applied Mechanics Reviews **67** (2015)
3. M. Gad-el Hak, Applied Mechanics Reviews **49**, 365 (1996)
4. T.R. Bewley, Progress in Aerospace Sciences **37**, 21 (2001)
5. L.N. Cattafesta III, M. Sheplak, Annual Review of Fluid Mechanics **43**, 247 (2011)
6. H. Choi, W.P. Jeon, J. Kim, Annual Review of Fluid Mechanics **40**, 113 (2008)
7. M. Pastoor, L. Henning, B.R. Noack, R. King, G. Tadmor, Journal of Fluid Mechanics **608**, 161 (2008)
8. O.M. Aamo, M. Krstic, *Flow control by feedback: stabilization and mixing* (Springer Verlag, London, UK, 2003)
9. P.E. Dimotakis, Annual Review of Fluid Mechanics **37**, 329 (2005)
10. A.P. Dowling, A.S. Morgans, Annual Review of Fluid Mechanics **37**, 151 (2005)
11. S. Bagheri, D.S. Henningson, Philosophical Transactions of the Royal Society A **369**, 1365 (2011)
12. N. Fabbiane, O. Semeraro, S. Bagheri, D.S. Henningson, Applied Mechanics Reviews **66** (2014)
13. C.W. Rowley, S.T.M. Dawson, Annual Review of Fluid Mechanics **49**, 387 (2017)
14. Y. Li, W. Cui, Q. Jia, Q. Li, Z. Yang, M. Morzyński, B.R. Noack, arXiv preprint arXiv:1905.12036 (2020)
15. D.R. Jones, M. Schonlau, W.J. Welch, Journal of Global Optimization **13**, 455 (1998)
16. P. Hennig, C.J. Schuler, Journal of Machine Learning Research **13**, 1809 (2012)
17. T. Duriez, S.L. Brunton, B.R. Noack, *Machine Learning Control — Taming Nonlinear Dynamics and Turbulence* (Springer Verlag, Cham, Switzerland, 2017)
18. J.N. Kutz, Journal of Fluid Mechanics **814**, 1 (2017)
19. S.L. Brunton, B.R. Noack, P. Koumoutsakos, Annual Review of Fluid Mechanics **52**, 477 (2020)
20. D. Fernex, R. Semaan, M. Albers, P.S. Meysonnat, W. Schröder, B.R. Noack, Physical Review Fluids **5**, 073901 (2020)
21. C. Lee, J. Kim, D. Babcock, R. Goodman, Phys. Fluids **9**(6), 1740 (1997)
22. M. Efe, M. Debiasi, P. Yan, H. Ozbay, M. Samimy, in *43rd AIAA Aerospace Sciences Meeting and Exhibit* (2005), p. 294
23. J. Park, H. Choi, Journal of Fluid Mechanics **904** (2020)
24. N. Benard, J. Pons-Prats, J. Periaux, G. Bugeda, P. Braud, J.P. Bonnet, E. Moreau, Experiments in Fluids **57**, 22 (2016)
25. F. Ren, H.B. Hu, H. Tang, Journal of Hydrodynamics **32**, 247 (2020)
26. Y. Zhou, F. D., B. Zhang, , R. Li, B.R. Noack, Journal of Fluid Mechanics **897**, A27:1 (2020)
27. J. Rabault, M. Kuchta, A. Jensen, U. Réglade, N. Cerardi, Journal of Fluid Mechanics **865**, 281 (2019)
28. E. Brochu, V.M. Cora, N. De Freitas, arXiv preprint arXiv:1012.2599 (2010)
29. B. Shahriari, K. Swersky, Z. Wang, R.P. Adams, N. De Freitas, Proceedings of the IEEE **104**, 148 (2015)
30. P.I. Frazier, in *Recent Advances in Optimization and Modeling of Contemporary Problems* (INFORMS, 2018), pp. 255–278
31. A. Blanchard, T. Sapsis, Journal of Computational Physics **425**, 109901 (2021)
32. N. Deng, B.R. Noack, M. Morzyński, L.R. Pastur, Journal of Fluid Mechanics **884**, A37 (2020)
33. W. Chen, C. Ji, M.M. Alam, J. Williams, D. Xu, Journal of Fluid Mechanics **891**, A14 (2020)
34. G.Y. Cornejo Maceda, Y. Li, F. Lusseyran, M. Morzyński, B.R. Noack, Journal of Fluid Mechanics **917** (2021)
35. P. Koumoutsakos, J. Freund, D. Parekh, AIAA Journal **39**, 967 (2001)
36. C.E. Rasmussen, C.K.I. Williams, *Gaussian processes for machine learning* (MIT Press, Cambridge, MA, 2006)
37. M. Raissi, P. Perdikaris, G.E. Karniadakis, Journal of Computational Physics **348**, 683 (2017)
38. G. Pang, P. Perdikaris, W. Cai, G.E. Karniadakis, Journal of Computational Physics **348**, 694 (2017)

		mean(C_D)	std(C_D)	amp(C_D)	mean(C_L)	std(C_L)	amp(C_L)	St
$\Delta\tau = 0.02$	$N = 5$	3.851	0.003015	0.008832	-0.05300	0.03111	0.08757	0.116
	$N = 7$	3.842	0.002959	0.008539	-0.05383	0.02852	0.08043	0.116
	$N = 9$	3.842	0.002944	0.008493	-0.05388	0.02846	0.08028	0.116
$\Delta\tau = 0.01$	$N = 5$	3.852	0.003017	0.008839	-0.05301	0.03108	0.08750	0.116
	$N = 7$	3.842	0.002961	0.008544	-0.05382	0.02852	0.08045	0.116
	$N = 9$	3.842	0.002946	0.008498	-0.05388	0.02847	0.08030	0.116

Table 2: Convergence results for the uncontrolled fluidic pinball ($\mathbf{b} = \mathbf{0}$) at $Re = 100$.

39. N. Srinivas, A. Krause, S.M. Kakade, M. Seeger, arXiv preprint arXiv:0912.3995 (2009)
40. J. Sacks, W.J. Welch, T.J. Mitchell, H.P. Wynn, Statistical Science pp. 409–423 (1989)
41. Y. Yang, A. Blanchard, T. Sapsis, P. Perdikaris, arXiv preprint arXiv:2102.10085 (2021)
42. A. Blanchard, T. Sapsis, SIAM/ASA Journal on Uncertainty Quantification **9**, 564 (2021)
43. P.F. Fischer, J.W. Lottes, S.G. Kerkemeier. Nek5000 web page (2008). http://nek5000.mcs.anl.gov
44. C. Raibaudo, P. Zhong, B.R. Noack, R.J. Martinuzzi, Physics of Fluids **32**, 015108 (2020)
45. A.K. Perumal, Y. Zhou, Journal of Fluid Mechanics **848**, 592 (2018)
46. A.G. Nair, C.A. Yeh, E. Kaiser, B.R. Noack, S.L. Brunton, K. Taira, Journal of Fluid Mechanics **875**, 345 (2019)
47. G.Y. Cornejo Maceda, Gradient-enriched machine learning control exemplified for shear flows in simulations and experiments. Ph.D. thesis, Université Paris-Saclay (2021)