Gradient-free optimization of chaotic acoustics with reservoir computing

Francisco Huhn
Department of Engineering, University of Cambridge, Cambridge CB2 1PZ, United Kingdom

Luca Magri[*]
Aeronautics Department, Imperial College London, London SW7 2AZ, United Kingdom; Department of Engineering, University of Cambridge, Cambridge CB2 1PZ, United Kingdom; The Alan Turing Institute, London NW1 2DB, United Kingdom; and Institute of Advanced Study, TU Munich, 85748 Munich Garching, Germany (visiting)

(Received 17 June 2021; accepted 11 January 2022; published 31 January 2022)

Gradient-based optimization of chaotic acoustics is challenging for a threefold reason: (i) first-order perturbations grow exponentially in time; (ii) the statistics of the solution may have a slow convergence; and (iii) the time-averaged acoustic energy may physically have discontinuous variations, which means that the gradient does not exist for some design parameters. We develop a versatile optimization method, which finds the design parameters that minimize time-averaged acoustic cost functionals, and overcomes the three aforementioned challenges. The method is gradient-free, model-informed, and data-driven with reservoir computing based on echo state networks. First, we analyze the predictive capabilities of echo state networks in thermoacoustics both in the short- and long-time prediction of the dynamics. We find that both fully data-driven and model-informed architectures are able to learn the chaotic acoustic dynamics, both time-accurately and statistically. Informing the training with a physical reduced-order model with one acoustic mode markedly improves the accuracy and robustness of the echo state networks, while keeping the computational cost low. Echo state networks offer accurate predictions of the long-time dynamics, which would be otherwise expensive by integrating the governing equations to evaluate the time-averaged quantity to optimize. Second, we couple echo state networks with a Bayesian technique to explore the design thermoacoustic parameter space. The computational method is minimally intrusive because it requires only the initialization of the physical and hyperparameter optimizers. Third, we find the set of flame parameters that minimize the time-averaged acoustic energy of chaotic oscillations, which are caused by the positive feedback with a heat source, such as a flame in gas turbines or rocket motors. These oscillations are known as thermoacoustic oscillations. The optimal set of flame parameters is found with the same accuracy as brute-force grid search but with a convergence rate that is more than one order of magnitude faster. This work opens up new possibilities for nonintrusive ("hands-off") optimization of chaotic systems, in which the cost of generating data, for example, from high-fidelity simulations and experiments, is high.

DOI: 10.1103/PhysRevFluids.7.014402

[*] lm547@cam.ac.uk

I. INTRODUCTION

When the heat released by a flame is sufficiently in phase with the acoustic waves of a confined environment, such as a gas turbine, thermoacoustic oscillations can arise [1]. Physically, thermoacoustic oscillations occur when the thermal power released by the flame, which is converted into acoustic energy, exceeds fluid mechanic dissipation. In gas turbines and rocket motors, thermoacoustic oscillations are unwanted because they can cause structural vibrations, fatigue, noise, and, if uncontrolled, can shake the device apart. Therefore, the objective of manufacturers is to design and operate stable devices [2–4]. The preliminary design of thermoacoustic systems is based on linear analysis, in which the growth rates of infinitesimal oscillations is computed on top of a time-independent baseline solution. If no growth rate is positive, then the system is linearly stable [2,5–7]. If the system is linearly unstable, then sensitivity methods, which are based on gradient computation, have been introduced to answer the practitioners' question "How can we change the design parameters to reduce the growth rate of infinitesimal oscillations?" In particular, adjoint methods proved computationally cheap tools to calculate the gradients of an eigenvalue with respect to all parameters of interest [7–11] with higher-order corrections [12]. Adjoint gradients were applied to the optimization of a longitudinal combustor [13] and annular combustors [e.g., Refs. 14,15].

Although linear analysis provides valuable information on the system's stability and sensitivity, thermoacoustic oscillations are inherently nonlinear. First, the heat release varies nonlinearly with the acoustics, which perturb the flame dynamics [5,16]. Second, hydrodynamic instabilities (e.g., vortex shedding), which are promoted by the geometry of the combustor, can modulate the flame dynamics and, thus, the heat release rate [17]. Because of these nonlinearities, a thermoacoustic system may be stable to infinitesimal perturbations (i.e., linearly stable), but finite-amplitude perturbations can trigger sustained oscillations [18], i.e., in the bistable region of a subcritical bifurcation. These sustained oscillations can be periodic, quasiperiodic or chaotic, as shown in experiments [19–24] and numerical studies [25–27], to name a few. Among these nonlinear regimes, chaotic oscillations are the most intractable to optimization [17,28].

Chaotic oscillations are extremely sensitive to infinitesimal perturbations [29], which results in an exponential growth of infinitesimal perturbations. In other words, the tangent space in unstable[1]. Because of this, the calculation of gradients of ergodic averages, i.e., *time-averaged* quantities of interest, is intractable with traditional sensitivity methods. This roadblock motivated the development of alternative gradient-based methods, which can be grouped into six categories: (i) ensemble methods [30–32], which average the gradient over an ensemble of short time trajectories; (ii) probability density methods [33,34], which calculate the gradient from the change in the probability density function of the chaotic attractor; (iii) unstable periodic orbits [35], which decompose the chaotic attractor into unstable periodic orbits and compute their gradients; (iv) fluctuation-dissipation-theorem methods [36–38], which compute the mean linear response of a system to small changes in external forcing; (v) shadowing methods [39–43], which average over time the difference between a baseline trajectory and its shadowing trajectory; and (vi) recent developments on linear response theory [44,45]. In particular, shadowing methods have successfully computed first-order sensitivities of time-averaged energies in fluid mechanics [46,47]. In thermoacoustics, shadowing-based gradients were embedded into a gradient update routine for design optimization [28], in which the time-averaged acoustic energy was minimized by computing the optimal set of flame parameters. The study highlighted three challenges in gradient-based optimization of chaotic thermoacoustics. First, thermoacoustic systems physically exhibit an abundance of bifurcations, across which the time-averaged cost functional being optimized can be discontinuous [17,28]. Second, shadowing-based methods require a number of tangent (i.e., first-order perturbation) solutions equal to the number of positive Lyapunov exponents [41], which can bear a significant

[1] In other words, at least one Lyapunov exponent is positive.

computational cost. Third, the nonlinear dynamics of thermoacoustics can be nonhyperbolic, i.e., the covariant Lyapunov basis may become defective [48], for certain design parameters [28]. This means that gradients cannot be guaranteed to exist for all thermoacoustic design parameters, which can hinder, and can even prevent, the optimization process via gradient-update. In this paper, we develop a gradient-free optimization methodology to find the optimal design parameters that minimize the time-averaged acoustic energy.

In either gradient-based and gradient-free methods, the solution must be integrated sufficiently long in time (ideally, ad infinitum), such that the quantities of interest (gradient in gradient-based, and cost functional in gradient-free methods) have converged to within a desired precision[2]. The generation of such a time series, however, can carry a high cost. As an alternative to the integration of the governing equations, we propose the use of a data-driven technique to produce accurate predictions of the system's dynamics to generate the required long time series. Such a task naturally falls within the category of supervised learning for time-dependent systems. In time-dependent problems, the order by which data is sorted (i.e., time) is of paramount importance. Feed-forward neural networks are a classic architecture, which works well in regression problems [49], but it does not naturally include recurrences to accurately learn the temporal correlations. To extend feed-forward networks to sequential data, recurrent neural networks have an internal state, which is updated by taking into account both the current input and the previous state. Thus, the sequence by which data is fed affects the internal state and therefore its output. Within recurrent neural networks, three architectures are highlighted: (i) long short-term memory networks (LSTM) [50], (ii) gated recurrent unit networks (GRU) [51], and (iii) echo state networks (ESN) [52,53]. While the three have been successfully employed to learn and predict time-dependent problems, the ESN architecture offers an advantage, which is exploited in this paper. Because its output is a linear combination of the hidden state variables, its training reduces to a least-squares problem, which is more computationally robust than repeatedly calculating gradients, as in LSTM and GRU networks. In chaotic learning, ESNs have been recently explored in multiple applications in chaotic systems, from time-accurate prediction [54,55], to the reconstruction of hidden variables [56–58], or the calculation of ergodic quantities [59]. In particular, in Huhn and Magri [59], ESNs were employed in the prediction of the long-time average of a thermoacoustic dynamical system. Moreover, Hart *et al.* [60] proved analytically that, under certain conditions, ESNs can approximate the invariant measure of a dynamical system, which is key to the calculation of accurate statistical quantities. In this paper, we employ ESNs for the generation of sufficiently long time series, from which the time-averaged cost functional to be optimized is evaluated. Specifically, we apply this framework to predict the time-averaged thermoacoustic energy.

The objective of this paper is threefold. First, we propose a versatile gradient-free methodology to optimize time-averaged cost functionals. The methodology requires a minimal number of user-defined parameters, which makes it a minimally intrusive tool. We apply the methodology to a chaotic thermoacoustic system. Second, we investigate the capability of ESNs of learning thermoacoustic solutions from small data. Both short- and long-time predictions are analyzed. Third, we minimize a chaotic thermoacoustic oscillation by finding the optimal set of design parameters.

The paper is structured as follows. Section II presents the general optimization problem with a focus on the thermoacoustic system. Section III introduces the proposed gradient-free optimization method, both in general and in particular case of this paper. The method combines the tools of Sec. IV, which describes Bayesian optimization; and Sec. V, which presents both the traditional and hybrid echo state networks. Section VI A investigates the short- and long-time predictions of the ESNs in learning thermoacoustic dynamics. Section VI B applies the framework of Sec. III to the optimization of a chaotic thermoacoustic system. A final discussion and conclusions end the paper. We have also included a discussion of the potential cost benefit of the proposed optimization framework in Appendix C.

[2] Both converge with $t^{-1/2}$, where t is time.

II. PROBLEM FORMULATION AND PHYSICAL MODELS

We consider a nonlinear dynamical system,

$$\frac{d\boldsymbol{q}}{dt} = \boldsymbol{F}(\boldsymbol{q}, \boldsymbol{s}), \tag{1}$$

where $\boldsymbol{q} \in \mathbb{R}^{N_d}$ is the state vector; $\boldsymbol{F} : \mathbb{R}^{N_d} \to \mathbb{R}^{N_d}$ is a nonlinear operator; $\boldsymbol{s} \in \mathbb{R}^{N_p}$ is a vector of physical (or design) parameters; and N_d is the number of degrees of freedom of the system. Given an initial condition, $\boldsymbol{q}_0$, Eq. (1) can be solved to obtain a solution $\boldsymbol{q}(t, \boldsymbol{s})$. We wish to optimize the time-average of a cost functional,

$$\langle \mathcal{J} \rangle(\boldsymbol{s}) = \lim_{T\to\infty} \frac{1}{T} \int_0^T \mathcal{J}[\boldsymbol{q}(t, \boldsymbol{s}), \boldsymbol{s}]\, dt, \tag{2}$$

where $\mathcal{J}$ is, for example, an energy. Because we consider ergodic systems, $\langle \mathcal{J} \rangle$ does not depend on the initial condition or trajectory, but it depends only on the parameters, $\boldsymbol{s}$. The goal is to find a set of parameters, $\boldsymbol{s}^+$, that minimize the time-averaged cost functional in Eq. (2). Mathematically, $\boldsymbol{s}^+$ is the solution of

$$\min_{\boldsymbol{s}} \langle \mathcal{J} \rangle, \tag{3}$$

$$\boldsymbol{G}(\boldsymbol{s}) = 0, \tag{4}$$

$$\boldsymbol{H}(\boldsymbol{s}) \geqslant 0, \tag{5}$$

where $\boldsymbol{G}$ and $\boldsymbol{H}$ are equality and inequality constraints, respectively. The inequality constraints guarantee that the physical parameters are searched in a feasible region.

A. Thermoacoustic dynamical system

We consider an acoustic resonator that consists of a tube and a heat source in it. We assume that the cutoff frequency of the acoustic resonator is sufficiently high such that only longitudinal acoustics propagate. The mean flow is assumed to have a zero Mach number with a spatially averaged temperature. The equations that govern the acoustics are the linearized momentum and energy equations

$$\frac{\partial u}{\partial t} + \frac{\partial p}{\partial x} = 0, \tag{6}$$

$$\frac{\partial p}{\partial t} + \frac{\partial u}{\partial x} + \zeta p - \dot{q}\delta(x - x_f) = 0, \tag{7}$$

where u, p, ζ, $\dot{q}$, δ, and x_f are the acoustic velocity, pressure, damping, heat-release rate, Dirac δ, and flame position, respectively, which are nondimensionalized as in Refs. [28,61]. The axial coordinate is $x \in [0, 1]$, which is nondimensionalized by the tube length. The heat release rate is given by a modified King's law [62–65],

$$\dot{q}(t) = \beta\{[1 + u(x_f, t - \tau)]^{1/2} - 1\}, \tag{8}$$

where β and τ are the heat release intensity and time delay, respectively. The time delay models the time that the heat release takes to be perturbed by an acoustic perturbation at the base of the heat source. The solutions are decomposed in N_g acoustic eigenfunctions of the undamped acoustic system [66], which is also known as Galerkin decomposition,

$$u(x, t) = \sum_{j=1}^{N_g} \eta_j(t) \cos(j\pi x), \tag{9}$$

$$p(x, t) = -\sum_{j=1}^{N_g} \mu_j(t) \sin(j\pi x), \tag{10}$$

which results in a system of $2N_g$ oscillators that are nonlinearly coupled by the heat source

$$\frac{d\eta_j}{dt} - j\pi\mu_j = 0, \tag{11}$$

$$\frac{d\mu_j}{dt} + j\pi\eta_j + \zeta_j\mu_j + 2\dot{q}\sin(j\pi x_f) = 0, \tag{12}$$

where $\zeta_j = c_1 j + c_2 j^{1/2}$ is the modal damping, which damps out higher-frequency oscillations according to physical scaling [67]. Despite its simplicity, the thermoacoustic model in Eqs. (6)–(8) qualitatively captures complex nonlinear dynamics and bifurcations, as shown in Refs. [18,28,61,66]. Because we wish to use numerical integrators and echo state networks, which march from time step n to $n+1$, it is convenient to transform the time-delayed problem Eq. (8) into an initial value problem. To achieve this, we model the advection of a dummy variable v with velocity τ^{-1} as [28]

$$\frac{\partial v}{\partial t} + \frac{1}{\tau}\frac{\partial v}{\partial X} = 0, \quad 0 \leqslant X \leqslant 1, \tag{13}$$

$$v(X=0,t) = u_f(t). \tag{14}$$

The time-delayed velocity is provided by the value of v at the right boundary, i.e., $u_f(t-\tau) = v(X=1,t)$. Equation (13) is discretized using $N_c + 1$ points with a Chebyshev spectral method [68], which adds N_c degrees of freedom. Thus, these equations define a dynamical system with a state vector $\boldsymbol{q} = [\eta_1, \cdots, \eta_{N_g}, \mu_1, \cdots, \mu_{N_g}, v_1, \cdots, v_{N_c}]$. This model, which is used as a proof of concept, qualitatively captures the key physics of nonlinear thermoacoustics [28]. The cost functionals that we wish to obtain and minimize are the time averages of the acoustic energy and the Rayleigh index [17]

$$E_{\text{ac}}(t) = \int_0^1 \frac{1}{2}\big(u^2(t) + p^2(t)\big)\,dx = \frac{1}{4}\sum_{j=1}^{N_g}\big[\eta_j^2(t) + \mu_j^2(t)\big], \tag{15}$$

$$I_{\text{Ra}}(t) = p_f(t)\dot{q}(t). \tag{16}$$

The first measures the total energy of the acoustic oscillations, while the second corresponds to the rate of input energy in the system, which is balanced out over time by the damping. (As shown in Ref. [28], the two time-averaged cost functionals are one-to-one related to each other, therefore, we focus on the optimization of the acoustic energy only.) In this work, the following parameters are fixed: $x_f = 0.2$, $c_1 = 0.1$, and $c_2 = 0.06$ [18]. Unless otherwise specified, we use 10 Galerkin modes (i.e., $N_g = 10$) and 11 Chebyshev points (i.e., $N_c = 10$), which provide a good compromise between accuracy and computational cost [28]. We solve the equations numerically in time with a three-stage Runge-Kutta solver [69], with a time step of 0.01 time units.

III. GRADIENT-FREE DESIGN OPTIMIZATION WITH ECHO STATE NETWORKS

We introduce the proposed methodology to optimize a chaotic system with a nonintrusive approach, using echo state networks. The flowchart in Fig. 1(a) illustrates the method. There are two optimizers, one for the physical parameters and another for the hyperparameters. The physical optimizer chooses the next point in the physical space to be evaluated. By integrating the ordinary differential equations that govern the thermoacoustic dynamics (ODEs), a short amount of data is generated, which is used to train the network. This mimics data from high-fidelity simulation or experiments, which is sparse and costly, the objective being to gain as much information as possible with a minimal number of samples. Then, the hyperparameter optimizer selects the optimal hyperparameters. With the hyperparameters tuned, the data-driven model (echo state networks in this case) runs in predictive mode for a user-defined sufficiently long time to obtain the long-time average of the physical cost functional, which is returned to the physical parameter optimizer. The

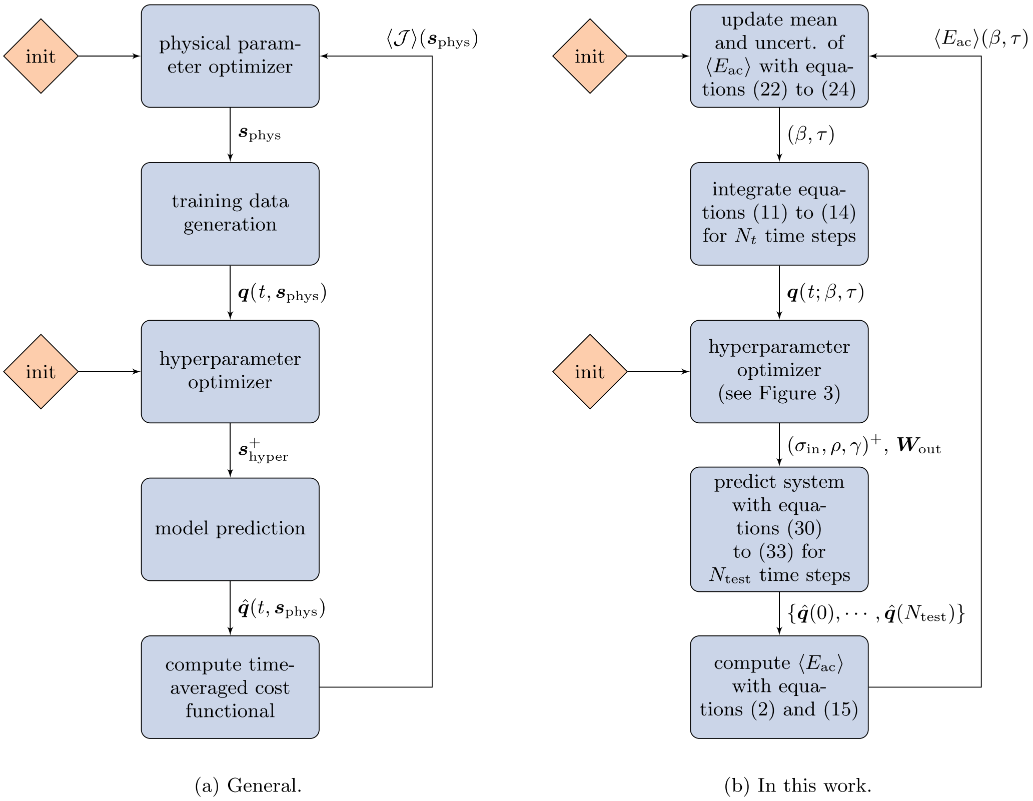
init
physical parameter optimizer
$\langle \mathcal{J} \rangle(\boldsymbol{s}_{\text{phys}})$
$\boldsymbol{s}_{\text{phys}}$
training data generation
$\boldsymbol{q}(t, \boldsymbol{s}_{\text{phys}})$
init
hyperparameter optimizer
$\boldsymbol{s}^{+}_{\text{hyper}}$
model prediction
$\hat{\boldsymbol{q}}(t, \boldsymbol{s}_{\text{phys}})$
compute time-averaged cost functional
(a) General.
init
update mean and uncert. of $\langle E_{\text{ac}} \rangle$ with equations (22) to (24)
$\langle E_{\text{ac}} \rangle(\beta, \tau)$
(β, τ)
integrate equations (11) to (14) for N_t time steps
$\boldsymbol{q}(t; \beta, \tau)$
init
hyperparameter optimizer (see Figure 3)
$(\sigma_{\text{in}}, \rho, \gamma)^{+}$, $\boldsymbol{W}_{\text{out}}$
predict system with equations (30) to (33) for N_{test} time steps
$\{\hat{\boldsymbol{q}}(0), \cdots, \hat{\boldsymbol{q}}(N_{\text{test}})\}$
compute $\langle E_{\text{ac}} \rangle$ with equations (2) and (15)
(b) In this work.

FIG. 1. Optimization chain flowcharts. Human interaction is only required in the initialization (`init` steps). The initialization defines the search space, maximum number of evaluations, optimiser parameters, kernel functions, etc.

only human intervention is at the start of the chain for the initialization of the optimizers. After initialization (e.g., defining search space, maximum number of evaluations, etc.), the optimization chain runs on its own. Figure 1(b) depicts the chain of Fig. 1(a) in the present work. In particular, in this work, the physical optimizer is a Bayesian optimizer using Gaussian process regression (GPR), with a Matérn 3/2 kernel. Because of the nature of a Gaussian process (Sec. IV A), the dependent variable (in this case, the acoustic energy, E_{ac}) can be negative. Because a negative acoustic energy is unphysical, we apply the GPR to the logarithm of the acoustic energy, which means that values of the acoustic energy are modelled by a log-normal distribution. Hence, the estimates are positive and the standard deviation is additive in the exponent only. Finally, because of interpretability, we use the Lowest Confidence Bound (see Sec. IV B) as the acquisition function, with $\kappa = 1.960$, which corresponds to a 95% confidence interval. The ordinary differential equations are integrated using a three-stage Runge-Kutta scheme [69]. The hyperparameters are also selected via Bayesian optimization with GPR, but with an RBF kernel and GP-hedge acquisition function. The system is predicted with a hybrid echo state network model and the cost functional is the time-averaged acoustic energy, calculated with the prediction from the network. All these concepts are introduced in the following two sections.

IV. BAYESIAN OPTIMIZATION WITH GAUSSIAN PROCESS REGRESSION

Gaussian process regression offers an estimate of both the mean and standard deviation of the cost functional. This allows for a more informed choice to be made and for better control of the balance between exploration and exploitation (see Sec. IV B). Moreover, it is a global optimization method, which is advantageous when the cost functional is multimodal. We summarize Gaussian process regression in Sec. IV A and Bayesian optimization in Sec. IV B.

A. Gaussian process regression

A Gaussian process (GP) is a collection of random variables, any finite number of which have a joint Gaussian distribution [70]. Here, the random variables are the values of a function on its domain. In fact, the function is deterministic, but in the context of GPs, its (unknown) values are modelled as random variables. A GP is specified by its mean function, $m(\boldsymbol{x})$, usually set to 0, and the covariance function, often called kernel function, $k(\boldsymbol{x}, \boldsymbol{x}')$, which are defined as

$$m(\boldsymbol{x}) = \mathbb{E}[f(\boldsymbol{x})], \tag{17}$$

$$k(\boldsymbol{x}, \boldsymbol{x}') = \mathbb{E}\{[f(\boldsymbol{x}) - m(\boldsymbol{x})][f(\boldsymbol{x}') - m(\boldsymbol{x}')]\}, \tag{18}$$

where $f(\boldsymbol{x})$ is the real process and $\mathbb{E}$ is the expectation. The Gaussian process is written as

$$f(\boldsymbol{x}) \sim \mathcal{GP}[m(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}')], \tag{19}$$

where $\{(\boldsymbol{x}_i, f_i)|i = 1, \dots, n\}$ is a collection of n data points, from which we construct the output vector $\boldsymbol{f}$ and the matrix $\boldsymbol{X}$, whose columns are the vectors $\boldsymbol{x}_i$. Similarly, we can define $\boldsymbol{f}_*$ and $\boldsymbol{X}_*$ for the n_* test inputs, i.e., the inputs that we wish to predict. According to the definition of a GP, prior to any observations, the joint distribution of $\boldsymbol{f}$ and $\boldsymbol{f}_*$ is a Gaussian distribution,

$$\begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{f}_* \end{bmatrix} \sim \mathcal{N}\left(\boldsymbol{0}, \begin{bmatrix} \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}) & \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}_*) \\ \boldsymbol{K}(\boldsymbol{X}_*, \boldsymbol{X}) & \boldsymbol{K}(\boldsymbol{X}_*, \boldsymbol{X}_*) \end{bmatrix}\right), \tag{20}$$

where, for any $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$, $\boldsymbol{K}(\boldsymbol{X}_1, \boldsymbol{X}_2)$ is an $n_1 \times n_2$ matrix of covariances evaluated for all pairs of columns (each column being one point) of $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$. To include the information from the observations f_i, this distribution is conditioned on the observations [70]

$$\boldsymbol{f}_*|\boldsymbol{X}_*, \boldsymbol{X}, \boldsymbol{f} \sim \mathcal{N}(\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*), \tag{21}$$

$$\boldsymbol{\mu}_* = \boldsymbol{K}(\boldsymbol{X}_*, \boldsymbol{X})\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X})^{-1}\boldsymbol{f}, \tag{22}$$

$$\boldsymbol{\Sigma}_* = \boldsymbol{K}(\boldsymbol{X}_*, \boldsymbol{X}_*) - \boldsymbol{K}(\boldsymbol{X}_*, \boldsymbol{X})\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X})^{-1}\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}_*). \tag{23}$$

If the observations are noisy, with variance σ_n^2, then $\boldsymbol{K}$ is replaced by $\boldsymbol{K} + \sigma_n^2 \boldsymbol{I}$, where $\boldsymbol{I}$ is the identity matrix.

In this work, we use two kernel functions, the radial basis function,

$$k(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{||\boldsymbol{x} - \boldsymbol{x}'||^2}{2l^2}\right), \tag{24}$$

and the Matérn 3/2 kernel function,

$$k(\boldsymbol{x}, \boldsymbol{x}') = \left(1 + \frac{\sqrt{3}}{l}||\boldsymbol{x} - \boldsymbol{x}'||\right) \exp\left(-\frac{\sqrt{3}}{l}||\boldsymbol{x} - \boldsymbol{x}'||\right), \tag{25}$$

where $||\cdot||$ is the Euclidean distance and l are tunable length scales, which control the smoothness of the function being regressed. A large l means that the covariance will be high even for relatively distant points, $\boldsymbol{x}$ and $\boldsymbol{x}'$, resulting in a smoother function than that for smaller l. Because we expect the acoustic energy to be smoother in the physical space [28] than the mean-squared error in the hyperparameter space [71], we use the RBF kernel for the former and the Matérn kernel for the latter.

B. Bayesian optimization

Bayesian optimization is used in this work to select the hyperparameters, and to optimize the physical parameters such that the acoustic energy of the system is minimal.

Bayesian optimization consists of a loop of three steps:

(1) Observe a point from the optimization domain;

(2) Update the mean, μ_*, and uncertainty, Σ_*;

(3) Select the next point to observe by finding the minimum of the acquisition function.

The first step simply evaluates the function, f, at the given point $\boldsymbol{x}$. The second step calculates the mean and variance of the distribution of Eqs. (22) and (23). Finally, in the third step, the new point to observe corresponds to the optimum of an acquisition function.

An acquisition function takes into account both the mean and uncertainty and, for any point $\boldsymbol{x}$, outputs a value that relates to either or a combination of the two. This provides the probability, or amount, by which $\boldsymbol{x}$ can improve the current optimum. There are four common acquisition functions: Probability of improvement (PI), expected improvement (EI), lowest confidence bound (LCB), and GP-hedge [72]. The first, PI, computes the probability that a candidate $\boldsymbol{x}$ can improve with respect to the current optimum, $\text{prob}[f(\boldsymbol{x}) < f^+]$. The second, EI, is similar to PI, but weighs the probability by the potential gain, i.e., it is the expected value of the improvement, $\mathbb{E}\{\max[f^+ - f(\boldsymbol{x}), 0]\}$. Finally, LCB is based on intervals of confidence centered around the mean, $[\mu - \kappa\sigma, \mu + \kappa\sigma]$. For a given value of κ, the interval will cover a certain percentage of the outcomes. For example, with $\kappa \approx 1.960$, 95% of the outcomes will be contained in the interval, i.e., a 95% confidence interval[3]. When the LCB acquisition function is used, one seeks to find $\boldsymbol{x}$ for which the lower bound of the confidence interval is a minimum, i.e., the $\boldsymbol{x}$ with the smallest lower confidence bound. This means that κ controls the balance between *exploration* (exploring unobserved regions of the optimization space) and *exploitation* (improving an existing observation by searching close to it), with exploration being preferred when κ is large (κ multiplies the uncertainty σ) and exploitation when κ is small. Finally, GP-hedge [72] overcomes the difficulty of knowing which acquisition function will perform best by taking the previous three acquisition functions[4] and probabilistically picking one of the three suggestions to sample next. For each acquisition function, the better the means, μ, of the points suggested in the past, the larger the probability of being chosen. In this paper, we use the LCB because, although it does not necessarily offer the best performance among the four acquisition functions, it is simple to compute and, more importantly, it is simple to physically interpret.

V. ECHO STATE NETWORKS

Echo state networks (ESN) [52,53] are recurrent neural networks, which are composed of a set of nodes that constitute the reservoir. The ESN receives an input signal, $\boldsymbol{l}(n) \in \mathbb{R}^{N_l}$, and produces an output signal, $\hat{\boldsymbol{y}}(n) \in \mathbb{R}^{N_y}$, where n is the discrete time variable, i.e., $t = n\,\Delta t$. Usually $N_l = N_y = N_d$, where N_d is the dimension of the system being predicted, such that the network can evolve on its own. The state of the reservoir is a vector, $\boldsymbol{r}$, of the states of all units, $r_j,\ \ j \in \{1, \dots, N_r\}$. The reservoir state evolves according to the nonlinear law

$$\boldsymbol{r}(n) = \tanh\{\boldsymbol{W}_{\text{in}}[\boldsymbol{l}(n); b_{\text{in}}] + \boldsymbol{W}\boldsymbol{r}(n-1)\}, \tag{26}$$

where $\boldsymbol{W}_{\text{in}}$ is the input matrix (i.e., $\boldsymbol{W}_{\text{in}}^{i,j}$ is the weight from the jth component of the input to the ith node) and b_{in} is the input bias, with the semicolon denoting row concatenation. Similarly, in the recurrency matrix $\boldsymbol{W}$, the component $\boldsymbol{W}^{i,j}$ is the weight from the jth node to the ith node. Therefore, $\boldsymbol{W}_{\text{in}}$ and $\boldsymbol{W}$ are $N_r \times N_d$ and $N_r \times N_r$ matrices. The hyperbolic tangent in Eq. (26) is

[3] For an $\alpha\%$ confidence interval, one finds κ such that $\Phi(\mu + \kappa\sigma) - \Phi(\mu - \kappa\sigma) = \alpha$, where Φ is the cumulative distribution function of the Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$.

[4] GP-hedge can be applied to any combination of acquisition functions, not only PI, EI and LCB.

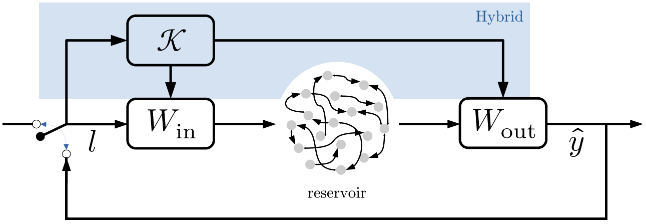

FIG. 2. Schematic of the conventional (without highlighted region) and hybrid echo state networks. The hybrid echo state network corresponds to a conventional ESN with an additional knowledge-based model ($\mathcal{K}$ box) that feeds the reservoir and the output via the augmented $\boldsymbol{W}_{\text{in}}$ and $\boldsymbol{W}_{\text{out}}$ matrices. In training, the switch is horizontal, whereas in prediction, it is vertical.

applied entrywise. Finally, the output is calculated by linear combination of the states of the reservoir units,

$$\hat{\boldsymbol{y}}(n) = \boldsymbol{W}_{\text{out}}[\boldsymbol{r}(n); b_{\text{out}}], \tag{27}$$

where $\boldsymbol{W}_{\text{out}}$ is the output matrix, of size $N_d \times N_r$, and b_{out} is the (scalar) output bias.

The network is trained to produce an output $\hat{\boldsymbol{y}}$ that matches the target $\boldsymbol{y}$ by minimizing the mean-squared error (MSE) [52,53]

$$\text{MSE} = \frac{1}{N_t}\sum_{n=1}^{N_t}\frac{||\hat{\boldsymbol{y}}(n) - \boldsymbol{y}(n)||^2}{N_d}, \tag{28}$$

where N_t is the number of (discrete) time steps, and the norm is Euclidean in $\mathbb{R}^d$. In ESNs, both $\boldsymbol{W}_{\text{in}}$ and $\boldsymbol{W}$ are generated once and fixed. In this work, each reservoir node is connected to one input, which results in every row of $\boldsymbol{W}_{\text{in}}$ having only one nonzero entry. The weight of the connections is sampled from a uniform distribution in the range $[-\sigma_{\text{in}}, \sigma_{\text{in}}]$, where σ_{in} is a scaling parameter. Hence, $\boldsymbol{W}_{\text{in}}$ can be generated by sampling uniformly from the range $[-1, 1]$ and scaling by σ_{in} directly in Eq. (26). Similarly, $\boldsymbol{W}$ is generated by sampling from the uniform distribution in the range $[-1, 1]$, with each node being on average connected to $(1 - \text{sp})N_r$ other nodes, where sp is the desired sparseness. The matrix is then scaled to have a desired spectral radius, ρ, which is typically smaller than unity to satisfy the echo state property [52,53]. A network that satisfies the echo state property "forgets" an old input after a certain time, which means that, even if starting from two different states, a network with the echo state property will converge to the same trajectory after a certain time (provided it is fed by the same input).

Because $\boldsymbol{W}_{\text{in}}$ and $\boldsymbol{W}$ are fixed, only the output weights (i.e., the entries of $\boldsymbol{W}_{\text{out}}$) are trained to solve the minimization problem Eq. (28) with ridge regression

$$\boldsymbol{W}_{\text{out}} = (\boldsymbol{R}^T\boldsymbol{R} + \gamma\boldsymbol{I})^{-1}\boldsymbol{R}^T\boldsymbol{Y}, \tag{29}$$

where γ is the user-defined Tikhonov factor, which regularises the training. The $\boldsymbol{R}$ and $\boldsymbol{Y}$ matrices are obtained by row-concatenating the reservoir states and output targets, i.e., the nth row corresponds to the discrete time n. During training mode, the network is operated in open-loop, whereas in prediction mode, the output of the network is fed to its input (closed-loop), i.e.,

$$\boldsymbol{l}(n+1) = \hat{\boldsymbol{y}}(n), \tag{30}$$

for the network to evolve autonomously. This corresponds to the schematic of Fig. 2 with the blue highlighted region removed.

A. Hybrid echo state network

The hybrid echo state network (hESN) is a variant of the conventional ESN [54]. In the hESN, the capability of the conventional ESN is complemented by (possibly imperfect) physical knowledge from a dynamical system, which may be a reduced-order model. The combination of data and model knowledge achieves higher accuracy, not only in the short time prediction [54,71] but also in the long time [59]. Figure 2 shows the architecture of an hESN. The network's input is fed to the reservoir, as in the conventional ESN, and to the physical knowledge based system (marked $\mathcal{K}$). The output of the physical system, in turn, is passed to both the reservoir, via $\boldsymbol{W}_{\text{in}}$, and the output, via $\boldsymbol{W}_{\text{out}}$. Mathematically, Eqs. (26) and (27) are augmented by $\mathcal{K}$,

$$\boldsymbol{r}(n) = \tanh\{\boldsymbol{W}_{\text{in}}[\boldsymbol{l}(n); b_{\text{in}}; \mathcal{K}(n)] + \boldsymbol{W}\boldsymbol{r}(n-1)\}, \tag{31}$$

$$\hat{\boldsymbol{y}}(n) = \boldsymbol{W}_{\text{out}}[\boldsymbol{r}(n); b_{\text{out}}; \mathcal{K}(n)], \tag{32}$$

where the dependence on n is implicit via the input, $\boldsymbol{l}(n)$,

$$\mathcal{K}(n) = \mathcal{K}[\boldsymbol{l}(n)]. \tag{33}$$

Although the hESN can perform better than a conventional ESN of equal size, as shown in Sec. VI A, it can result in an unstable behavior. In prediction mode, the feedback of the output of $\mathcal{K}$, via $\boldsymbol{W}_{\text{out}}$, into its own input can create a self-sustaining amplification that diverges to infinity (see Appendix A). In such cases, validation and test errors are undefined, violating regularity assumptions (e.g., continuity in the hyperparameter space), which are essential to many optimization algorithms. We propose three ways of overcoming this issue. The first is error saturation. If the prediction error becomes greater than a threshold, then the error is set to the threshold. The second is saturation, where a saturation function is applied to the output of the physical model itself, e.g., $\mathcal{K} \rightarrow \tanh(\mathcal{K})$, with the tanh taken entrywise (as in the conventional reservoir update equation). This can be seen as effectively increasing the reservoir by a number of units equal to the dimension of $\mathcal{K}$, where each of these units is connected to one entry of $\mathcal{K}$ only and without repetition. The drawback is that, due to the saturation, the sensitivity to changes in the output of $\mathcal{K}$ is reduced, which can impact the performance. The third is to eliminate the connection between $\mathcal{K}$ and $\boldsymbol{W}_{\text{out}}$, with the output of $\mathcal{K}$ feeding the reservoir only, effectively preventing unbounded growth. We tested the three suggested methods (result not shown). We found that the first option performed best for the case under investigation, which is why it is adopted in the remainder of the paper.

B. Hyperparameter selection

The traditional technique for selecting hyperparameters is manual selection, which is dependent on prior (human) knowledge and experience. However, that does not suit a nonintrusive approach, which is central to the objective of this work, as explained in Sec. VI B. The simplest nonintrusive technique is grid search, but it can carry high computational cost [71,73,74]. Furthermore, the discretization of the hyperparameter space is a delicate matter because, if it is too coarse, the optimum can be missed; whereas, if it is too fine, the computational cost becomes prohibitive. Bayesian optimization with Gaussian process regression has been documented to achieve good performance in hyperparameter tuning of echo state networks [71]. For example, Reinier Maat *et al.* [74] found that this technique systematically achieves similar, or lower, values of test error compared to grid search but with fewer evaluations. In an in-depth examination of training techniques [71] (e.g., single-shot, cross validation, etc.), it was found that grid search and Bayesian optimization have similar values of validation error, with Bayesian optimization being more robust and efficient. In this paper, the hyperparameters are selected by minimizing the validation mean-squared error (validation MSE), using Bayesian optimization with Gaussian process regression (GPR). We use the implementation in the `scikit-optimize` library. The GPR uses a 3/2 Matérn kernel [70] (see Sec. IV A) and the acquisition function is the GP-hedge (see Sec. IV B). The initial seed points are generated using a Latin hypercube sampling method. To make it more amenable to optimization,

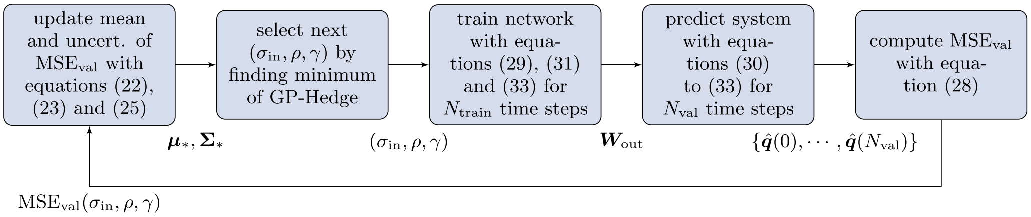
update mean and uncert. of $\mathrm{MSE}_{\mathrm{val}}$ with equations (22), (23) and (25)
select next $(\sigma_{\mathrm{in}}, \rho, \gamma)$ by finding minimum of GP-Hedge
train network with equations (29), (31) and (33) for N_{train} time steps
predict system with equations (30) to (33) for N_{val} time steps
compute $\mathrm{MSE}_{\mathrm{val}}$ with equation (28)
$\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*$
$(\sigma_{\mathrm{in}}, \rho, \gamma)$
$\boldsymbol{W}_{\mathrm{out}}$
$\{\hat{\boldsymbol{q}}(0), \cdots, \hat{\boldsymbol{q}}(N_{\mathrm{val}})\}$
$\mathrm{MSE}_{\mathrm{val}}(\sigma_{\mathrm{in}}, \rho, \gamma)$

FIG. 3. Hyperparameter selection with Bayesian optimization.

we smooth the cost functional with two modifications. First, because the values of the validation MSE cover multiple orders of magnitude (e.g., 10^{-6}–10^{3}), we minimize the logarithm of the MSE. Second, we cap the error when it is larger than the threshold of 10^3, as explained in Sec. V A. The saturation smooths the MSE at these points.

The minimization runs until the MSE is below a target threshold of 3×10^{-2}, which was chosen by trial and error; or the maximum number of calls, 20, has been reached. This value for the maximum number of calls is sufficiently large for the hyperparameter space to be explored, but not too large for the computation to become exceedingly expensive. We optimally tune ρ, σ_{in}, which are two hyperparameters that markedly affect the training [71]. The hyperparameter space is log-uniform, i.e., the optimization tunes the exponents of the hyperparameters,

$$\log_{10}(\sigma_{\mathrm{in}}) \in [-2, 2], \tag{34}$$

$$\log_{10}(\rho) \in [-3, 0]. \tag{35}$$

A log-uniform allows a more efficient exploration of different scales than a linear space. In particular, ρ is related to the time scale of the dynamics, which can be of different orders of magnitude for different systems or attractors. In the optimization of Sec. VI B, because the attractor varies, we also include the Tikhonov factor, γ, as a hyperparameter with $\log_{10}(\gamma) \in [-11, -4]$. The process of hyperparameter selection via Bayesian optimization is schematized in Fig. 3.

C. Data normalization

Data normalization is crucial to obtaining good performance [53]. Because different components of the data vector can have vastly different ranges, a single input scaling factor, σ_{in}, can be insufficient. If the same scaling is applied to variables of different orders of magnitude, then the tanh might "ignore" one of the variables because of the saturation of values away from 0. In that case, the information from that variable would be lost.

While various normalizations exist, here we choose the "min-max" normalization, which divides the data variable by the difference between its maximum and minimum in the time period. For an unnormalized variable, $\breve{l}$, whose time series is $\{\breve{l}(0), \breve{l}(1), \dots\}$, the normalized variable, l, is given by

$$l(n) = \frac{\breve{l}(j)}{\max_j[\breve{l}(j)] - \min_j[\breve{l}(j)]}, \quad n = 0, 1, \dots. \tag{36}$$

This normalization forcibly makes $l \in [-1, 1]$, which means that all the variables have the same range. With all the (normalized) variables in the same range, the use of the single scaling factor σ_{in} is justified.

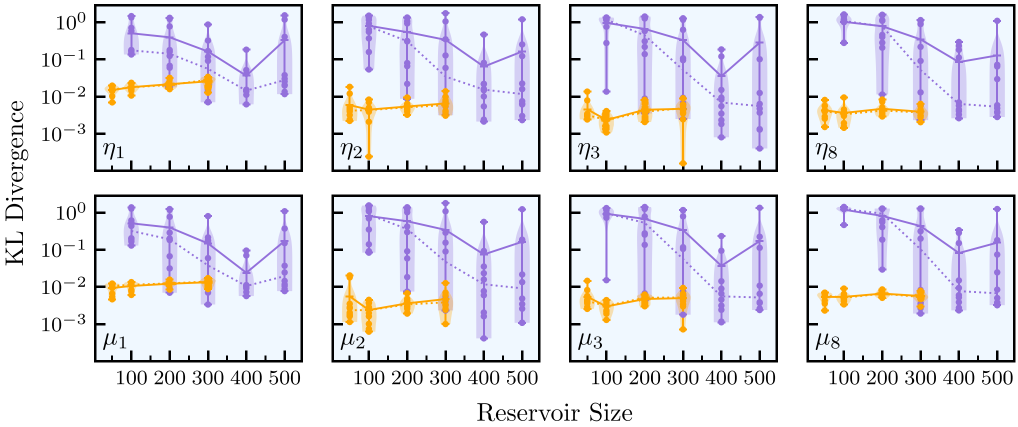

FIG. 4. Kullback-Leibler divergence, D_{KL}, versus reservoir size. The first and second rows correspond to the velocity and pressure modes, η_j and μ_j. The shaded regions correspond to the distributions of D_{KL} arising from the reservoir realizations. Each dot is a reservoir realization. The solid and dashed lines correspond to the mean and median. ESN (_), hESN (_).

VI. RESULTS

The training, validation and test lengths are $N_{\mathrm{train}} = 5000$, $N_{\mathrm{val}} = 2000$ and $N_{\mathrm{test}} = 10000$, respectively. For the chaotic attractor of Sec. VI A 1, this corresponds to approximately 6, 2.4 and 12 Lyapunov times[5]. To initialize the network, the reservoir state is initialized to 0 and the first 100 iterations are discarded. This ensures that the training of the network is not affected by the initialization. The physical model of the hESN is the same dynamical system that generates the data [Eqs. (11) and (12)], but with one Galerkin mode only instead of 10 (i.e., $N_g = 1$). This mimics a situation in which data is available from an experiment for which a simple physical model exists. Alternatively, data can come from a high-fidelity simulation, while $\mathcal{K}$ is a reduced-order model obtained from first principles and approximations.

The reservoir is composed of 400 and 100 nodes in the conventional and hybrid architectures, respectively. Although the reservoir size is usually chosen by heuristics, such as choosing the largest one can afford [53], or by human experience, here, for completeness, we show that these values are optimal for their respective architectures. To analyze the quality of a prediction, we use the Kullback-Leibler divergence [75],

$$D_{\mathrm{KL}}(P||Q) = \int p(\boldsymbol{x}) \log\left[\frac{p(\boldsymbol{x})}{q(\boldsymbol{x})}\right] dV, \tag{37}$$

where P and Q are continuous distributions and p and q are the respective probability density functions. The integral is taken in the phase space of the system. In the Kullback-Leibler divergence, P refers to a "truth," against which a "model" Q is being compared, which is suitable for the present situation, in which the truth is the data generated by the ODEs and the models are the echo state networks. If Q perfectly matches P, i.e., $p(\boldsymbol{x}) = q(\boldsymbol{x})\ \forall \boldsymbol{x}$, then $D_{\mathrm{KL}} = 0$, indicating that the larger D_{KL}, the worse the match. The numerical calculation of Eq. (37) is performed with the empirical distributions, i.e., via the histograms of P and Q.

Figure 4 shows the variation of D_{KL} with respect to the reservoir size for both the conventional and hybrid echo state network architectures. For each reservoir size, an ensemble of 10 network

[5] The leading Lyapunov exponent is approximately 0.12 [59], which means that the Lyapunov time is approximately 0.12^{-1}.

TABLE I. Characteristics of the ESN and hESN. sp is the sparseness (i.e., fraction of 0 entries) of $\boldsymbol{W}$. The Tikhonov factor is $\gamma = 10^{-9}$.

	N_r	$\boldsymbol{b}_{\text{in}}$	$\boldsymbol{b}_{\text{out}}$	sp	ρ	σ_{in}
ESN	400	1	1	99%	0.00176	12.2
hESN	100	0	0	97%	0.25760	0.02825

realizations is run, which allows the estimation of a distribution with its mean and spread. This is shown for the Galerkin modes 1, 2, 3, and 8, with the first three corresponding to the three most energetic modes and the last being representative of higher-order modes. The hESN performs better on average than the ESN. In fact, there are only a few realizations of the ESN that outperform realizations of the hESN. This is not unexpected and is further explored in the next section. Furthermore, compared to the hESN, the ESN exhibits much larger variability within realizations of the same size. Analyzing separately, the hESN exhibits low variability in network realizations. The results also indicate that reservoir size has little, and possibly detrimental, impact on performance, with the optimal size being 100. This is due to a combination of: (i) the physical model, which offers estimates of a good quality leaving reduced work to the network; (ii) too many reservoir nodes, thus training parameters, for the small amount of data used in the training. On the one hand, at very low numbers of nodes, the network will have high error because it does not have a sufficient number of parameters to train. On the other hand, at very large numbers of nodes, there are too many parameters and overfitting becomes a problem. Therefore, a U-shaped curve can be expected. This shape is barely visible for the hESN because there is only one point, 50, to the left of the optimum. Notwithstanding, this effect is more visible in the ESN curves, where there is an improvement as the reservoir size increases. Although both mean and median decrease up to 400, the mean increases from 400 to 500, while the median flattens or slightly decreases. This is explained by the large variability of the reservoirs of size 500. Therefore, given the similar median and comparable performances between ESN realizations of sizes 400 and 500, we select 400 nodes to keep the computational cost minimal. The realizations chosen for the following section are those closest to the median of selected sizes.

A. Short- and long-time predictions

In this section, we compare the predictive capabilities of both the conventional (ESN) and hybrid echo state networks (hESN). We fix the physical parameters $\beta = 7.0$ and $\tau = 0.2$, which correspond to a chaotic solution [59]. The physical system [Eqs. (11) and (12)] will be referred to as the "Truth." Information about each network, including the optimal hyperparameters, is given in Table I.

On the one hand, in short-time prediction, the objective is to time-accurately reproduce the dynamics of the system, i.e., starting from some initial condition, the objective is for the difference between the prediction and the true signals to be minimal for the largest possible time. This task is covered in Sec. VI A 1. On the other hand, in long-time prediction, the objective is to accurately reproduce the ergodic properties (statistics) of the system, i.e., the objective is for the difference between the true attractor (the stationary measure) and the attractor of the echo state networks to be minimal. Good performance in either task does not necessarily imply good performance in the other, as can be seen in Sec. VI A 2 (good long-time, but poor short-time performance with the conventional ESN) and Appendix B (good short-time, but poor long-time performance).

1. Short-time prediction

Figure 5 shows the time series of the first three (velocity) Galerkin modes, for the truth (data from ODE integration) and closed-loop predictions of the ESN and hESN. These modes are significantly more energetic than those of higher order because the flame is located at $x_f = 0.2$, which markedly

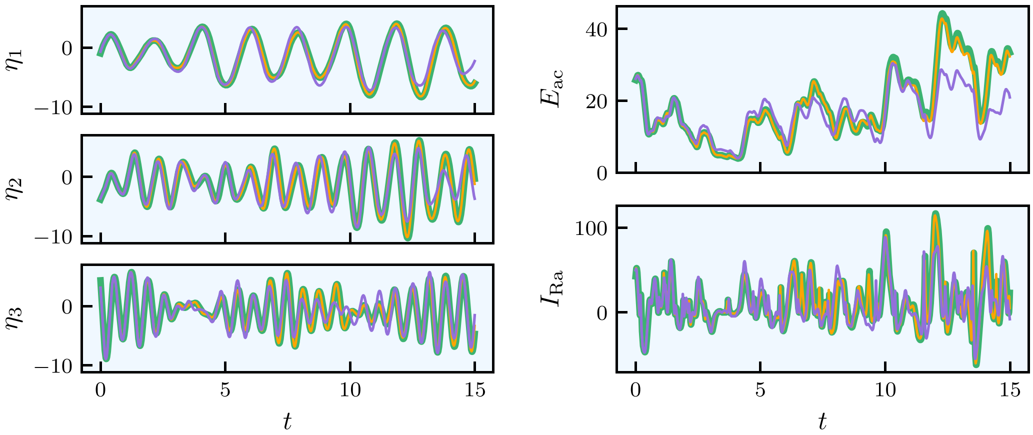

FIG. 5. Short-time prediction. Time series of the first three (velocity) Galerkin modes, η_1, η_2, and η_3; acoustic energy, E_{ac}; Rayleigh index, I_{Ra}. Truth (—), ESN (—), and hESN (—).

excites the first modes [28]. All three modes oscillate in a nonperiodic manner, with the peak frequency increasing with the mode number. Figure 5 also shows the time series of the two cost functionals, acoustic energy and Rayleigh index. The Rayleigh index oscillates substantially more than the acoustic energy because the time derivative of the acoustic energy is equal to the sum of the Rayleigh index and the dissipation from damping [17]. Timewise, the hESN is able to time-accurately predict these modes for the whole time span shown, whereas the ESN deviates from the truth signal at $t \approx 10$.

As a global metric, we compute the normalized root mean-squared error,

$$\mathrm{NRMSE}(n) = \left(\frac{||\hat{\mathbf{y}}(n) - \mathbf{y}(n)||^2}{N^{-1} \sum_{j=1}^{N} ||\mathbf{y}(j)||^2} \right)^{1/2}, \tag{38}$$

which is shown in Fig. 6. The hESN performs better than the ESN. Given a threshold, the predictability horizon is defined as the time at which the first crossing of the threshold occurs [54,55]. With a threshold of 0.5, the hESN achieves a predictability horizon of 42.1 time units [5.1 Lyapunov times, compared to 7.5 time units (0.9 Lyapunov times) of the ESN]. The NRMSE is, however, sensitive to normalization, and cannot discriminate between a time-inaccurate prediction that has similar dynamics and another that has completely different dynamics. An alternative visualization

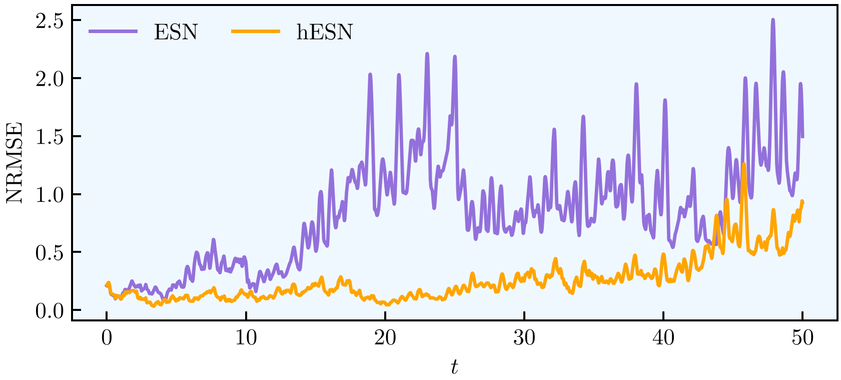

FIG. 6. Short-time prediction. Normalized root mean-squared error of the acoustic modes.

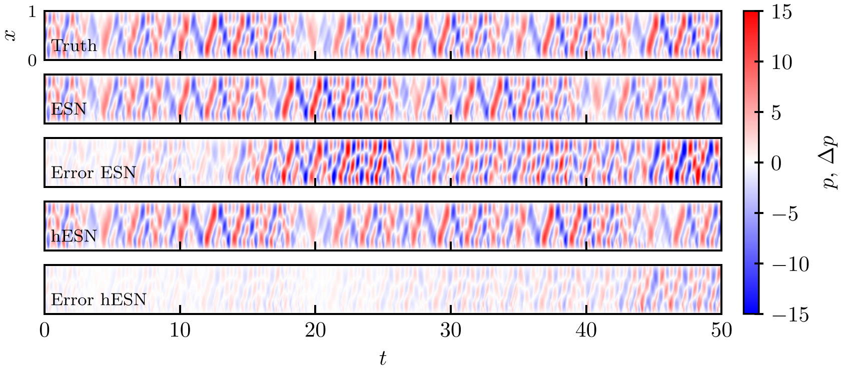

FIG. 7. Short-time prediction. Acoustic pressure. The error is defined as $\Delta p(t,x) = \hat{p}(t,x) - p(t,x)$, where $\hat{p}$ is the prediction and p is the true pressure field.

of the short-time behavior is given in Fig. 7, which shows the time evolution of the acoustic pressure in an x-t diagram. The truth panel shows that the flow is unsteady, featuring nonperiodic acoustic waves propagating inside the domain. Although nonperiodic, there appears to be a dominant frequency, with roughly five waves in each 10 time units long window, which corresponds to an approximate period of 2 time units. This is related to the first acoustic eigenfunction. The third panel, corresponding to the ESN error, shows that the predictability horizon of the ESN is relatively short, which corroborates the findings of the NRMSE of Fig. 6. However, the dynamics of the ESN are qualitatively similar to those of the truth. This can be important, because, as shown in Ref. [59], an ESN can display inaccurate short-time prediction, but can have accurate long-time dynamics. The findings from the pressure map agree with those of the NRMSE not only for the ESN, but also for the hESN. The pressure plot of the hESN indicates that it only starts to exhibit significant error at $t \approx 45$, which is similar to the predictability horizon of 42.1 time units found with the NRMSE. This result further shows that hESN is capable of time-accurate prediction. We remark that such conclusions do not apply in general to the classes of conventional and hybrid echo state networks (i.e., an ESN need not perform worse than an hESN). Increasing the reservoir size of the ESN could yield satisfactory short-time prediction as well. However, the inclusion of model knowledge significantly improves the performance of an ESN for the same reservoir size.

2. Long-time prediction

In this section, we focus on the ergodic (i.e., long-time) prediction, which is key to this paper. As previously mentioned, inaccurate short-time (i.e., time-accurate) prediction does not necessarily imply inaccurate long-time prediction [59]. (Conversely, as shown in Appendix B, accurate short-time prediction does not necessarily imply accurate long-time prediction either.) We analyze the predictive capability of the ESN and hESN in the long-time with metrics that are naturally defined in the statistically stationary regime.

First, we compute the frequency spectra (Fig. 8). The spectra are continuous, which is consistent with the underlying signal being chaotic. The spectra match satisfactorily, with the largest error appearing at higher frequencies $f \gtrapprox 2$, which have a negligible importance because the power of the signal is concentrated in the lower frequencies. For lower frequencies (inset of Fig. 8), there is a favorable agreement between the two types of networks and the truth. In particular, both ESN and hESN match the dominant acoustic frequency and the peak of the true signal, which are close to the first natural acoustic eigenmode, $f = 0.5$. This is consistent with the wave number in the x-t plot of Fig. 7. Analysis of the spectra of the other state variables suggest similar conclusions (result not shown). We can conclude that echo state networks, both conventional and hybrid, reproduce the physical system satisfactorily in the time domain.

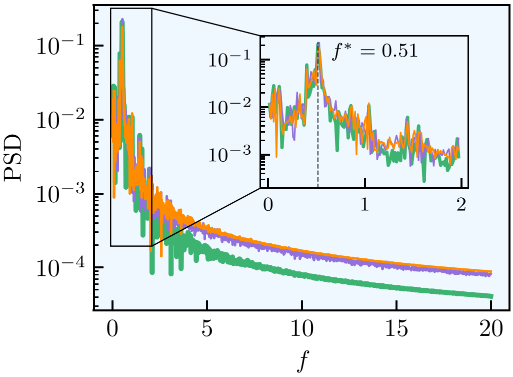

FIG. 8. Long-time prediction. Frequency spectrum of the acoustic velocity mode, $\eta_1(t)$, for Truth (—), ESN (—), and hESN (—).

Second, we compute the probability density functions (PDFs) of the chaotic attractor because the long-time behavior of a system and its statistics depend on the invariant measure of its attractor. We compute two-dimensional joint PDFs of the Galerkin modes, i.e., (η_1, μ_1), (η_2, μ_2), etc.; and the one-dimensional PDFs of the individual state variables. These are performed for modes 1, 2, 3, and 8 (Fig. 9), the first three being the most energetic and the last being representative of the higher modes. The PDFs are obtained via kernel density estimation [76,77]. Both networks perform well, with their PDFs matching those of the truth relatively well. However, although the ESN shows a good agreement with the truth, it is less accurate than the hESN. The difference in performance is more evident close to the modes of η_1 and η_2, where the ESN over- and underpredicts the values of the peaks. This indicates that the invariant measure of the attractor of the dynamical system is well captured by both the ESN and hESN, with the latter being more accurate. Therefore, both ESN and hESN predict the long-time statistics of the physical system, with training from relatively small data.

B. Design optimization

For the physical optimizer, we define the cost functional, $\langle E_{\mathrm{ac}} \rangle$; the optimization space, $\beta \in [7.5, 10.0]$, $\tau \in [0.1, 0.3]$; the acquisition function, LCB with $\kappa = 1.960$; the covariance function, RBF; the number of initial seed points, 4, which should be sufficient to properly initialize the GP; and the maximum number of evaluations (12, including the seed points), which we find to be a good compromise between efficacy and efficiency[6]. Similarly, for the hyperparameter optimiser, we define the cost functional, validation MSE; optimization space, $\log_{10}(\sigma_{\mathrm{in}}) \in [-2, 2]$, $\log_{10}(\rho) \in [-3, 0]$, $\log_{10}(\gamma) \in [-11, -4]$; the acquisition function, GP-hedge; the covariance function, Matérn 3/2; the number of initial seed points, 5; and the maximum number of evaluations, 25, which include the seed points. Here, we include the Tikhonov factor as a hyperparameter. The reason is that the physical optimization evaluates attractors that can vary widely. In this case, adjusting the Tikhonov factor becomes important because it controls the relative importance of the MSE and the norm of $\boldsymbol{W}_{\mathrm{out}}$ in the training problem, two terms that can vary substantially depending on the attractor. This is in contrast with Sec. VI A, where only one attractor was learned and predicted. To save on computational cost, the hyperparameter optimization stops when the error is below the threshold of 3×10^{-2} (Sec. V B). The chain then runs on its own.

[6] Efficacy here relates to whether the goal is achieved or not, whereas efficiency relates to how costly achieving the goal was.

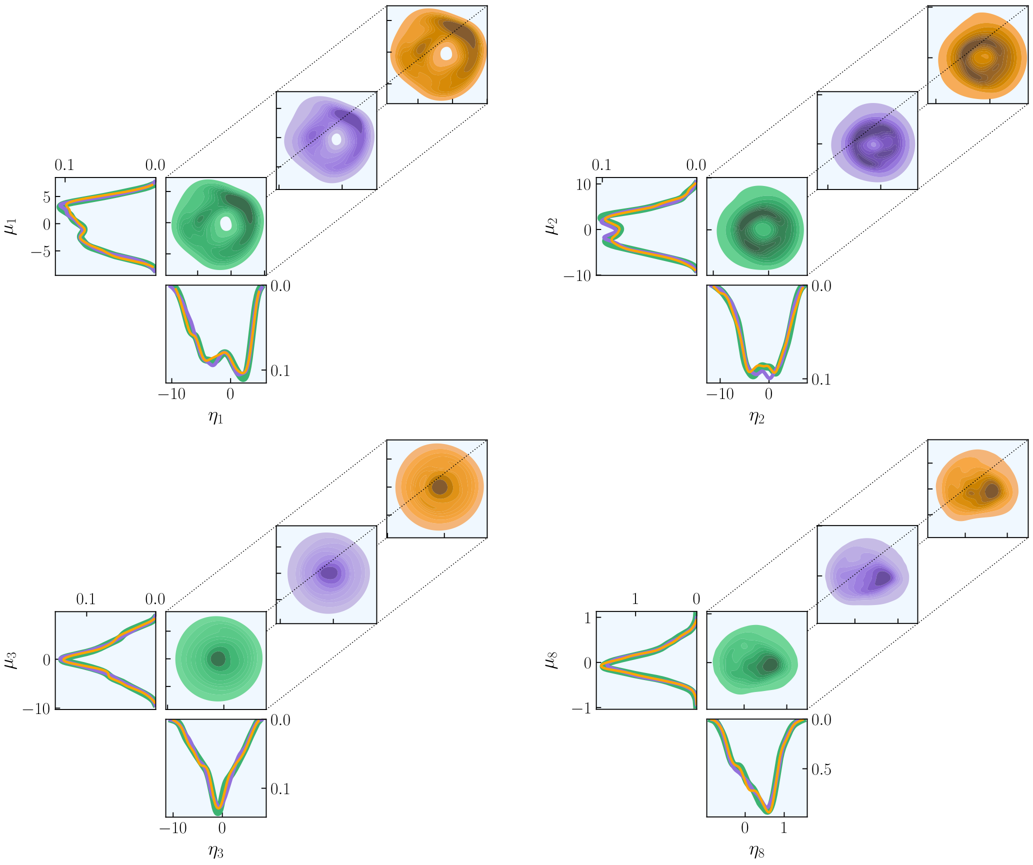

FIG. 9. Long-time prediction. Probability density functions of the acoustic modes 1, 2, 3, and 8. The two-dimensional joint PDFs correspond to the velocity and pressure variables of the same acoustic mode pair (η_j, μ_j). The one-dimensional PDFs are the marginalizations of the two-dimensional joint PDFs. Truth (—), ESN (—), hESN (—).

First, the physical optimizer randomly generates seed points. For each of these points, $\boldsymbol{s}_{\text{phys}} = (\beta, \tau)$, data, $\{\boldsymbol{q}(0), \boldsymbol{q}(1), \ldots\}$[7], is generated by integrating Eqs. (8), (11), and (12). Then, the hyperparameter optimizer selects the optimal hyperparameters, $\boldsymbol{s}^{+}_{\text{hyper}} = (\sigma^{+}_{\text{in}}, \rho^{+}, \gamma^{+})$, using Bayesian optimization. With the optimal hyperparameters (and the corresponding optimal $\boldsymbol{W}_{\text{out}}$), the network is run in closed-loop, generating a long time series, $\{\hat{\boldsymbol{y}}(0), \hat{\boldsymbol{y}}(1), \ldots\}$, from which the time-averaged acoustic energy, $\langle E_{\text{ac}}\rangle$, is computed and returned to the physical optimizer. After the seed points have been evaluated, the physical optimizer selects the next point by finding the optimum of the acquisition function, $\boldsymbol{s}_{\text{phys}} = (\beta, \tau)$, which is then evaluated in the same manner as the seed points. The optimization stops when 12 points (including seed points) have been evaluated.

For comparison, the true cost functional is physically shown in Fig. 10. This chart is generated by integrating the ODEs on a grid of 11 values of β and 21 of τ. There is a large region of high acoustic energy, which can be divided into two subregions, each centered around a local maximum,

[7] Whereas $\boldsymbol{q}$ is defined in equation (1) as continuous in time, here, $\boldsymbol{q}$ is the numerical solution, which is only defined at discrete times $0, 1, \ldots$. Thus, slightly abusing notation, we write $\boldsymbol{q}(t = n\,\Delta t)$ as $\boldsymbol{q}(n)$, where n is a discrete time.

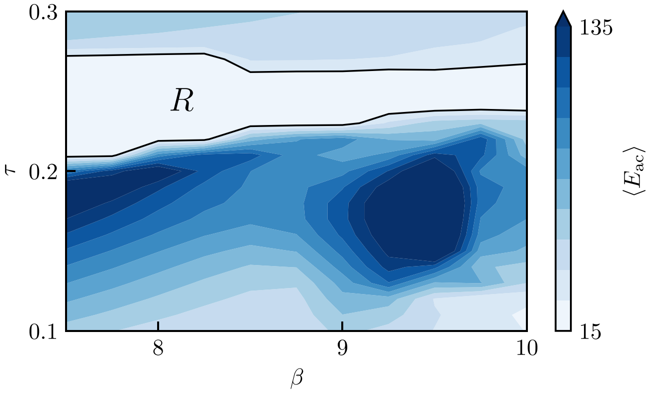

FIG. 10. Time-averaged acoustic energy, $\langle E_{\mathrm{ac}} \rangle$, versus the flame parameters, β and τ, obtained with a brute-force grid search. Benchmark solution.

one of which is on the boundary of the domain ($\beta = 7.5$ and $\tau \approx 0.18$). Above this, there exists a nearly horizontal strip spanning the whole range of β, marked R in Fig. 10, which is where the optimum from the optimization is likely to be found. Physically, as noted in Huhn and Magri [28], the time-averaged acoustic energy may be discontinuous at certain flame parameters because the attractor is structurally unstable. The global minimum found with the grid of Fig. 10 is $\langle E_{\mathrm{ac}} \rangle(\beta \approx 8.25, \tau \approx 0.27) = 15.04$.

Figure 11 shows the results of the physical optimization. The three columns correspond to the mean of the GPR, standard deviation of the GPR, and the acquisition function LCB. The ith row (starting at 0) corresponds to the state of the optimization after $n_{\mathrm{seed}} + i$ evaluations, where n_{seed} is the number of initial seed points (4 here) used to seed the optimization. It shows the previously evaluated points, with the most recent being encircled. The minimum of the acquisition function, and therefore the next point to be evaluated, is marked with a cross in the third column. Each row in Fig. 11 corresponds to a row of Fig. 12, which contains the time series of the acoustic energy, the phase plot μ_1 versus η_1 and the frequency spectrum of η_1 of the newly evaluated point. For the purpose of comparison, we include the true signals.

Initially (row $i = 0$), with 4 seed points, the GPR indicates a two-dimensional dependence on both β and τ, with clear regions of similar acoustic energy around each of the seed points, especially the two at the extremes of β. This is because these two points correspond to low and high values of acoustic energy. The combination of a larger distance in β than in τ between these two points, and the fact that the other two points, which have low and high values of τ, have similar $\langle E_{\mathrm{ac}} \rangle$, leads the fit of the GPR to place a larger weight on β than τ. In terms of dynamics, the optimum of the seed points is a chaotic attractor, as shown in the first row of Fig. 12.

The agreement between truth and hESN is favorable. Furthermore, the time series remains below the dashed line, which corresponds to the previous optimum, showing that this design is the optimum of the seed points. As expected, the uncertainty is lower in regions centered around evaluated points. The dependence on τ is substantially reduced after the evaluation of the first selected point ($i = 1$), a design that is close to the current optimum, not only in distance in the design space, but also in time-averaged acoustic energy (Fig. 12). The perceived small dependence on τ and relatively strong dependence on β, in conjunction with low estimated uncertainty, makes the acquisition function discard approximately three quarters of the optimization domain, corresponding approximately to the upper three quarters of the range of β (Fig. 11, $i = 1$, third column). With uncertainty low almost everywhere, its highest values are found close to the corners of the design space, where the distance from the sampled points is maximal. Moreover, given that the GPR indicates positive dependence of the cost functional on β and slightly negative dependence on τ, the minimum of the acquisition function is naturally found at the upper left corner, where β is minimal and τ maximal.

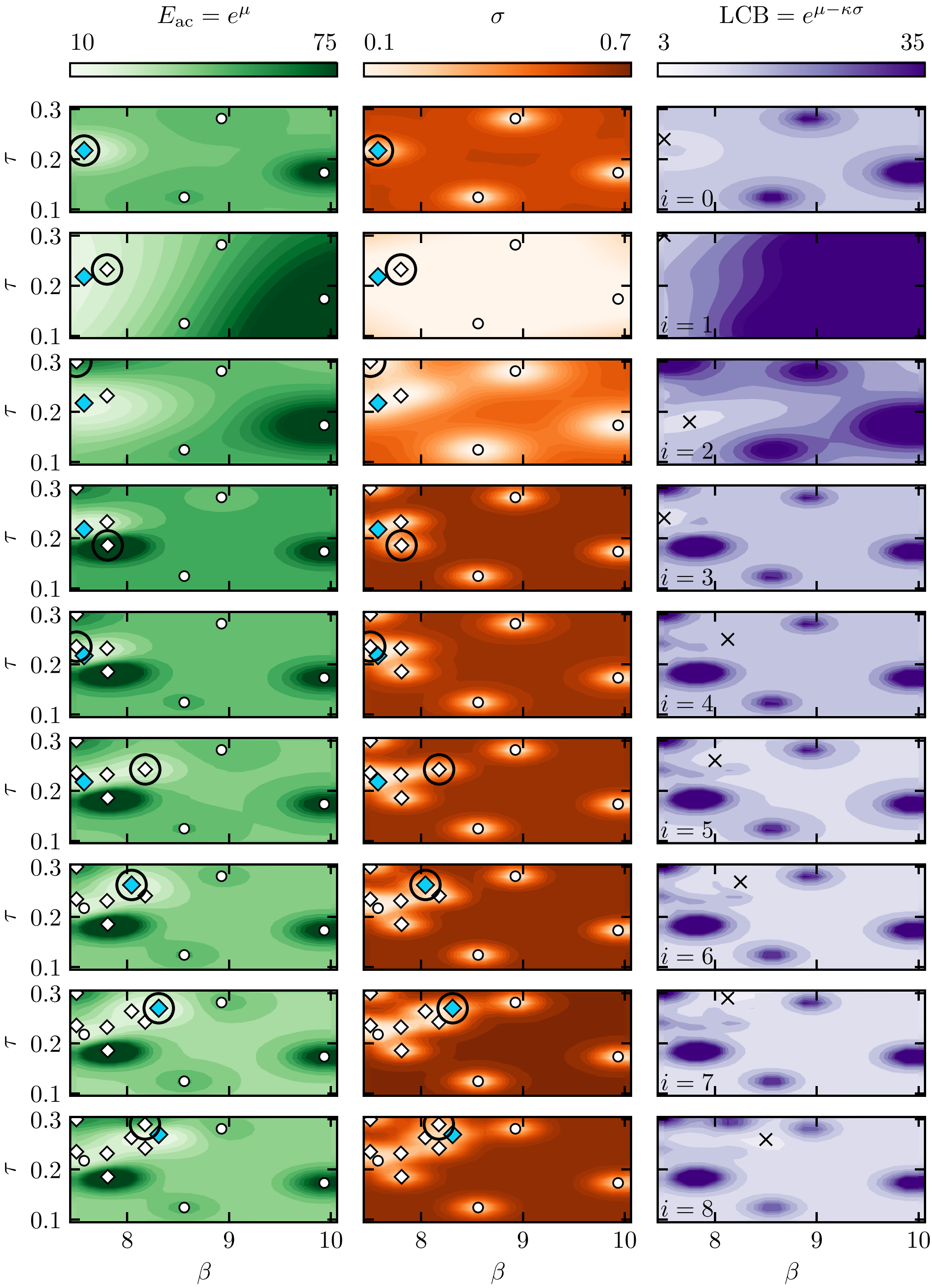

FIG. 11. Optimization history. The first and second columns show the mean and standard deviation of the GPR. Seed points and previously evaluated points are marked with white circles and diamonds, respectively. The current optimum is blue and the last evaluated point is encircled. The third column shows the acquisition function, LCB, the minimum of which is the next point to be evaluated (cross).

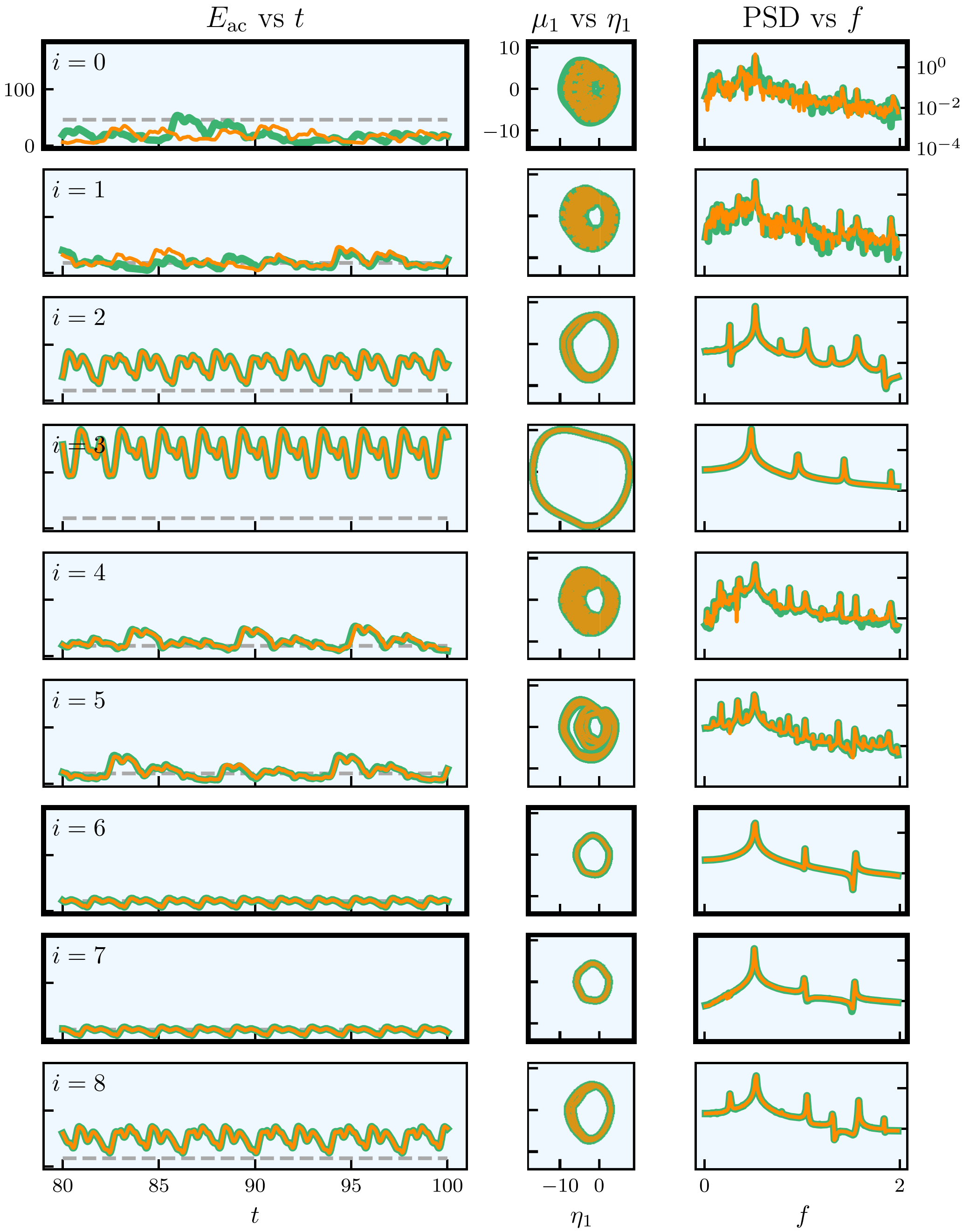

FIG. 12. Evaluated points during optimization. The columns correspond to the last fifth of the time series of the acoustic energy; the phase plot of the first acoustic mode (μ_1 vs η_1); and the frequency spectrum of the first acoustic velocity mode, η_1. Rows with thick spines correspond to a new optimum. The time-averaged acoustic energy of the current optimum is shown for reference as a horizontal dashed gray line.

This point ($i = 2$), in contrast with the estimate prior to its evaluation, is in a region of moderately high acoustic energy (Fig. 10). This can be verified in Fig. 12, which shows that this combination of β and τ corresponds to a limit cycle whose instantaneous acoustic energy never drops below the time-averaged acoustic energy of the current optimal design. The new data updates the GPR, which now exhibits moderate uncertainty throughout the domain, except relatively close to previously evaluated points ($i = 2$ row, second column). While the dependence on β remains, the dependence on τ increases and is no longer monotonic. With moderately high uncertainty away from the points, and with low mean in a small region only around the current optimum, the acquisition function selects a point where the two regions (high uncertainty, low mean) "meet". The new design point, however, has very high acoustic energy (row $i = 3$ of Fig. 12), as it belongs to the area surrounding the leftmost of the two peaks of high acoustic energy (Fig. 10). This newly acquired information strongly contrasts with the prior estimate of the GPR. Naturally, this is because of the sharp variation of acoustic energy at the lower edge of the region R. A small variation of parameters from the current optimum resulted in high variation of the cost functional. As such, uncertainty is now high everywhere, except close to previously sampled points. At $i = 4$, the acquisition function chooses a point close to the current optimum, which is a clear evidence of exploitation. This design results in chaotic dynamics, similarly to the current optimum, but it does not produce lower time-averaged acoustic energy. Thus, it is not a new optimum. With three points (and the left boundary) enclosing the current optimum, there is little advantage in continuing exploiting. Switching to exploration, the new point ($i = 5$) is relatively far from the current optimum. Once again, it is at the intersection of low mean and high uncertainty, since that combination minimises the acquisition function. While the new design does not improve the optimum, it does provide crucial new information. Because its acoustic energy is low, the region of low mean expands with the updated GPR. This new expansion provides space to exploit. Thus, a new design ($i = 6$) relatively close to the previous is selected. This new design offers lower time-averaged acoustic energy than the current optimum, i.e., the optimization found a new optimum. The new optimal design represents an improvement of 8.4% with respect to the previous optimum. In the GPR, not only has a new optimum been found, but the region of low mean is now larger. Thus, at $i = 7$, the most recently selected design further expands the spread of points into higher β and higher τ. Similarly to the last design, a new optimum is found, this one offering a further 11.4% reduction of acoustic energy. Finally, the last design ($i = 8$), despite being close to the previous two optima, does not further improve the cost functional. It is likely that this design is above the upper boundary of the region R in Fig. 10. Had the optimization continued to run, it is possible the optimum could have been slightly improved. However, the maximum number of evaluations was reached. Furthermore, the optimum of the optimization, found with 12 design evaluations (4 seed points plus 8 selected points), $\langle E_{\mathrm{ac}}\rangle(\beta \approx 8.31, \tau \approx 0.27) = 14.68$, is slightly better than that found with a brute-force grid search (Fig. 10), $\langle E_{\mathrm{ac}}\rangle(\beta \approx 8.25, \tau \approx 0.27) = 15.04$, which needed 231 evaluations. A larger number of evaluations would likely have been a relatively poor trade-off between design improvement (i.e., decrease in cost functional) and computational cost.

Figure 13 shows the convergence of the optimization procedure, i.e., the current optimum versus the number of points evaluated for three values of κ.

The largest value of κ, 2.576, favors exploration the most. This results in quickly, in the second optimization step, finding a design that improves on the best seed point. However, because of its tendency to explore, it does not try to exploit and locally improve its current optimum as much. Hence why there is a large spread of points for $\kappa = 2.576$ in Fig. 14, which shows the last state of the optimization for the same three values of κ of Fig. 13. In contrast, $\kappa = 0.967$, the lowest of the three, will seek to mostly exploit. The various designs concentrated in a small region are evidence of this. Unsurprisingly, this does not produce a better design than the optimal seed point. Finally, $\kappa = 1.960$, used in the optimization of Figs. 11 and 12, navigates between these two lines, exploration and exploitation, unsuccessfully trying to exploit initially, finding a new optimum with exploration and subsequently improving the recent optimum by exploiting its surrounding region. In conclusion, the larger κ is, the larger the spread of points, which shows the influence that this

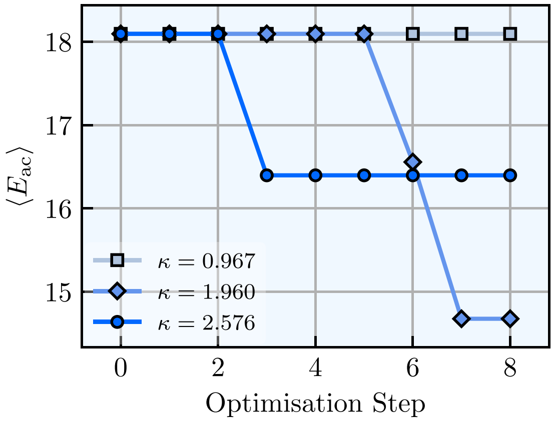

FIG. 13. Time-averaged acoustic energy, $\langle E_{\mathrm{ac}} \rangle$, versus number of points evaluated, for three values κ: 0.967, 1.960, and 2.576; corresponding to 67, 95, and 99% confidence intervals.

parameter has on the balance between exploration and exploitation. In this optimization problem, the initially chosen value of $\kappa = 1.960$ seems to offer the best performance among the three.

VII. CONCLUSIONS

Gradient-based design optimization of chaotic acoustics is notably challenging for a threefold reason. First, first-order perturbations grow exponentially in time, which makes the computation of the gradients with respect to the design parameters ill-posed. Second, the statistics of the solution may have a slow convergence, which makes the time integration of the equations computationally expensive. Third, chaotic acoustic systems may have discontinuous variations of the time-averaged energy [28], which means that the gradient may not exist for all design parameters. In this paper, we develop an optimization method to find the design parameters that minimize time-averaged acoustic cost functionals. The method is gradient-free, with Bayesian sampling; model-informed, with a reduced-order acoustic model; and data-driven, with reservoir computing.

First, we analyze the predictive capabilities of reservoir computing based on echo state networks. Both fully data-driven and model-informed architectures are considered. In the short-time prediction, model-informed networks can time-accurately predict the chaotic pressure oscillations beyond the predictability time (the Lyapunov time). For the same reservoir size, informing the training with a cheap model extends the prediction from ~1 Lyapunov time to ~5 Lyapunov times. In the long-time prediction, we show that echo state networks accurately reproduce the statistics of chaotic acoustic attractors. The hyperparameters are automatically tuned by using Bayesian optimization, which provides a consistently good performance across different architectures, reservoir sizes and data. With accurate predictions at a lower computational cost, the long-time series are generated

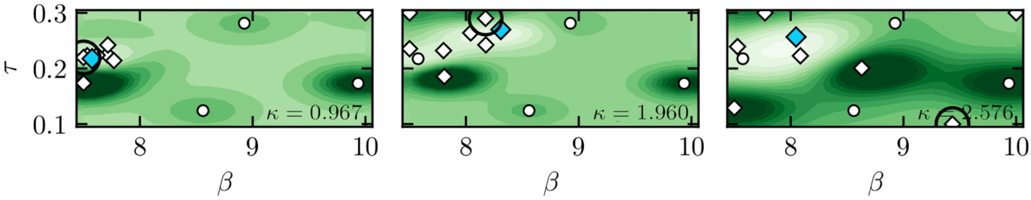

FIG. 14. Final state of optimization with three values of κ: 0.967, 1.960, and 2.576; corresponding to 67, 95, and 99% confidence intervals. This shows the effect of κ in the balance between exploration vs exploitation.

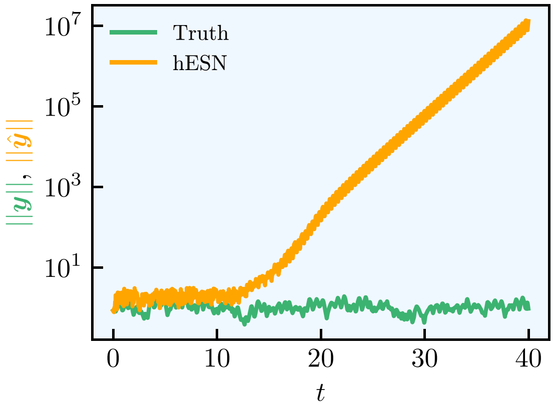

FIG. 15. Divergence of the prediction of a hybrid echo state network. This is obtained with the configuration detailed in Sec. V B and $\sigma_{\text{in}} = 4.23$ and $\rho = 0.9$. Other combinations of hyperparameters and physical parameters may result in similar behavior.

to obtain the time-averaged acoustic energy that is being optimized. Second, we couple echo state networks with a Bayesian technique based on Gaussian processes to explore the design parameter space. The computational method is minimally intrusive because it requires only the initialization of the physical and hyperparameter optimizers; e.g., a factor to balance the exploration versus exploitation of the sampling; and the reservoir size. Third, we apply the computational method to the minimization of the time-averaged acoustic energy of chaotic oscillations. We focus on the acoustics that is excited by a heat source, which is relevant to thermoacoustic oscillations in propulsion and power generation. The design parameters that are changed during the optimization are the flame intensity and time delay. Nonetheless, the method can tackle other physical parameters. Starting from five random designs with energetic chaotic oscillations, we find an optimal set of parameters in eight iterations. This optimum is practically equal to the optimum found by brute-force grid search, which needs 231 evaluations. The thermoacoustic system shows a variety of solutions and bifurcations during the optimization update (e.g., limit cycles, strange attractors), which are accurately learnt by the echo state networks. This is because the echo state network learns the physical temporal correlations of the acoustic modes through the sparse recurrent dynamics of the reservoir.

This work opens up new possibilities for the optimization of chaotic systems, in which the cost of generating data, for example, from high-fidelity simulations and experiments, is high.

ACKNOWLEDGMENTS

F. Huhn is supported by Fundação para a Ciência e Tecnologia under the Research Studentship No. SFRH/BD/134617/2017. L. Magri gratefully acknowledges financial support from the ERC Starting Grant No. PhyCo 949388, and TUM Institute for Advanced Study (German Excellence Initiative and the EU 7th Framework Programme No. 291763). The authors thank A. Racca for insightful discussions.

The authors report no conflict of interest.

APPENDIX A: DIVERGENCE OF HYBRID ECHO STATE NETWORK

Unlike conventional echo state networks, whose output is bounded (though the bound can be very far from the attractor), due to the feedback from the output of $\mathcal{K}$ to its own input via $\boldsymbol{W}_{\text{out}}$, hybrid echo state networks may diverge to infinity. This can be seen in Fig. 15. However, it should be noted that this behavior is not necessarily a function of the physical parameters or hyperparameters only.

A certain fixed combination of hyperparameters may result in both divergence and nondivergence depending on the physical parameters. Similarly, for fixed physical parameters, changing the hyperparameters may result in divergence or not. This is not an issue with the training method (ridge regression), but the complex combination of training and validation time series, realizations of $\boldsymbol{W}_{\text{in}}$ and $\boldsymbol{W}$, values of σ_{in} and ρ, Tikhonov factor γ, model $\boldsymbol{\mathcal{K}}$ and its numerical scheme, etc. Furthermore, $\boldsymbol{\mathcal{K}}$ can be stable (i.e., its solution is stable), but the hybrid echo state network using it can be unstable. In fact, that is the case of Fig. 15, where, if $\boldsymbol{\mathcal{K}}$ evolved on its own, then a limit cycle arises. It is the (linear) transformation due to the output weights $\boldsymbol{W}_{\text{out}}$ that changes the output of $\boldsymbol{\mathcal{K}}$, which is then fed back to $\boldsymbol{\mathcal{K}}$ itself, that can make the whole system unstable.

A very succinct example, where we omit the reservoir nodes for simplicity, is

$$\mathcal{K}(y) = (1 - \lambda)y, \tag{A1}$$

where $0 < \lambda < 1$ is a physical parameter. If left to evolve on its own, i.e.,

$$y(n + 1) = \mathcal{K}[y(n)] = (1 - \lambda)y(n), \tag{A2}$$

then y will converge to 0. However, in the hybrid echo state network framework, we have instead

$$y(n + 1) = \mathcal{K}[W_{\text{out}}y(n)] = W_{\text{out}}(1 - \lambda)y(n). \tag{A3}$$

For y to converge to 0, we must have

$$W_{\text{out}} < \frac{1}{1 - \lambda}. \tag{A4}$$

If W_{out} does not verify this condition, then the system will diverge, despite $\mathcal{K}$ being stable.

APPENDIX B: ACCURATE SHORT-TIME AND INACCURATE LONG-TIME PREDICTION

Here, we use a short example based on the Lorenz system [29] to show that accurate short-time prediction does not necessarily mean accurate long-time prediction. The Lorenz system is a three-dimensional system,

$$\frac{dx_L}{dt} = \sigma_L(y_L - x_L), \tag{B1}$$

$$\frac{dy_L}{dt} = x_L(\rho_L - z_L) - y_L, \tag{B2}$$

$$\frac{dz_L}{dt} = x_L y_L - \beta_L z_L, \tag{B3}$$

where σ_L, ρ_L, and β_L are parameters, often equal to 10, 28, 8/3, which is a combination that produces chaotic motion. This system is numerically integrated with time step 0.01 to generate training, validation, and test data. For this example, we use an echo state network with 100 nodes with no biases. The network is trained and validated on datasets of length 500, using Bayesian optimization. Figure 16 shows the NRMSE and (long-time) phase plot for the Lorenz system. The NRMSE remains below the threshold of 0.2 until $t \approx 2.87$, which corresponds to approximately 2.6 Lyapunov times (leading Lyapunov exponent of approximately 0.9). Thus, the ESN predicts the system relatively well in the short time. However, the phase plot shows a completely different behavior between prediction and data in the long time. In this case in particular, the network has no biases (i.e., $b_{\text{in}} = b_{\text{out}} = 0$), in which case the reservoir evolves according to

$$\boldsymbol{r}(n) = \tanh[\widetilde{\boldsymbol{W}}\boldsymbol{r}(n - 1)], \tag{B4}$$

where $\widetilde{\boldsymbol{W}} = \boldsymbol{W} + \boldsymbol{W}_{\text{in}}\boldsymbol{W}_{\text{out}}$ [59]. This means that taking some reservoir state $\boldsymbol{r}(n - 1)$, and flipping its sign, i.e., $\boldsymbol{r}'(n - 1) = -\boldsymbol{r}(n - 1)$, one gets $\boldsymbol{r}'(n) = -\boldsymbol{r}(n)$. Thus, either the ESN admits two attractors symmetric of each other, or admits one symmetric attractor, which is the case here. In conclusion, accurate short time prediction does not necessarily imply accurate long-time prediction.

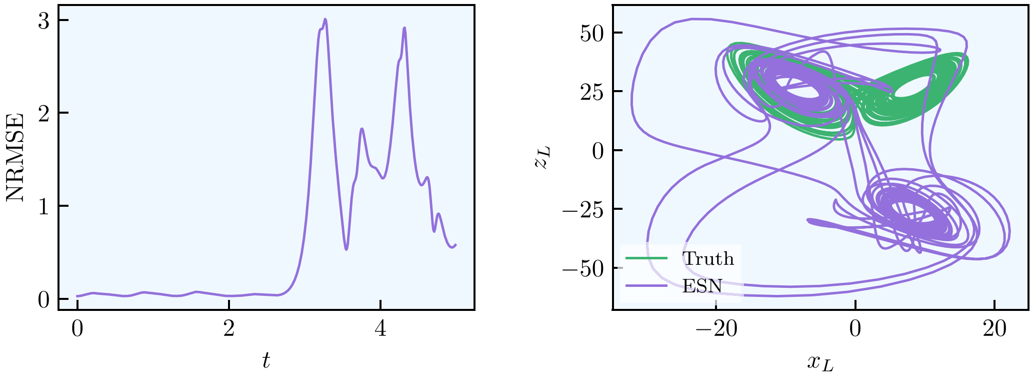

FIG. 16. Lorenz system. Time series of NRMSE and phase plot z_L vs x_L.

APPENDIX C: COMPUTATIONAL COST OF THE METHOD

The optimization framework in this paper (i.e., the “chain”) was demonstrated on a relatively low-dimensional system. In this particular case, the echo state network (and everything it involves) could have been foregone and Bayesian optimization applied directly to the result of the (longer run) ODEs. However, the application in this work is a proof of concept and not an example of computational gains. These are meant to be achieved in larger-scale systems, such as high-fidelity simulations. The cost of the method is

$$N_{\text{opt}}(N_{\text{train}}C_{\text{ODE}} + C_{\text{train}} + N_{\text{test}}C_{\text{ESN}}), \tag{C1}$$

where N_{opt} is the number of optimization steps in the physical domain, N_{train} is the number of training timesteps, C_{ODE} is the cost per timestep of solving the ODEs, C_{train} is the cost of training the network (including the hyperparameters), N_{test} is the number of test timesteps, and C_{ESN} is the cost of per timestep of the closed-loop ESN. However, applying Bayesian optimization directly to the ODEs instead would have a cost of

$$N_{\text{opt}}N_{\text{test}}C_{\text{ODE}}. \tag{C2}$$

For there to be a cost benefit

$$N_{\text{opt}}(N_{\text{train}}C_{\text{ODE}} + C_{\text{train}} + N_{\text{test}}C_{\text{ESN}}) < N_{\text{opt}}N_{\text{test}}C_{\text{ODE}} \tag{C3}$$

must be verified. Thus, each term must be analyzed. The most expensive operation is training the network, C_{train}, which scales with $O(N_x^3)$ due to the matrix inversion. If the number of nodes, N_x, is proportional to the dimension of the dynamical system, N_d, then this cost becomes $O(N_d^3)$. This operation only happens once (per hyperparameter combination), though. Once it is performed, the largest cost is the closed-loop ESN simulation at $O(N_xN_d) \sim O(N_d^2)$ per timestep. However, the cost of a timestep in a numerical simulation (ODE) can scale with $O(N_d^2)$ or $O(N_d^3)$, depending on the numerical scheme. This would put it at a similar scaling to the networks. Therefore, it would seem hard for Eq. (C3) to be verified. However, the goal is not to apply the technique directly to a high-fidelity simulation, but to a lower resolution of its results. While a certain number of grid points may be needed for an accurate simulation, after obtaining the results, only a subset of these points is required for the accurate computation of the cost functional. In other words, not all points are needed. Thus, the full state vector need not be computed/predicted. For example, if there is a downsample of 10 to 1 in every direction of a 3D high-fidelity simulation, then there is a 1000-fold reduction in N_d, 1000^2 in output cost and 1000^3 in training cost; compared to predicting the full state vector of a high-fidelity simulation. In such a case, the training data generation via a high-fidelity simulation would be the most expensive step, as we assume in the paper, which would also mean that the RHS of Eq. (C3) would be much larger than the LHS. Additionally, as remarked before, this approach also suits an experimental framework where running the experiment for a sufficiently long time might be expensive.

[1] L. Rayleigh, The explanation of certain acoustical phenomena, Nature (London) **18**, 319 (1878).

[2] T. C. Lieuwen and V. Yang, *Combustion Instabilities in Gas Turbine Engines: Operational Experience, Fundamental Mechanisms, and Modeling* (American Institute of Aeronautics and Astronautics, Reston, VA, 2005).

[3] F. E. C. Culick, *Unsteady Motions in Combustion Chambers for Propulsion Systems* (RTO AGARDograph AG-AVT-039, North Atlantic Treaty Organization, 2006).

[4] A. P. Dowling and Y. Mahmoudi, Combustion noise, Proc. Combustion Inst. **35**, 65 (2015).

[5] A. P. Dowling, Nonlinear self-excited oscillations of a ducted flame, J. Fluid Mech. **346**, 271 (1997).

[6] M. P. Juniper and R. Sujith, Sensitivity and nonlinearity of thermoacoustic oscillations, Annu. Rev. Fluid Mech. **50**, 661 (2018).

[7] L. Magri, Adjoint methods as design tools in thermoacoustics, Appl. Mech. Rev. **71**, 020801 (2018).

[8] L. Magri and M. P. Juniper, Sensitivity analysis of a time-delayed thermo-acoustic system via an adjoint-based approach, J. Fluid Mech. **719**, 183 (2013).

[9] L. Magri and M. Juniper, Global modes, receptivity, and sensitivity analysis of diffusion flames coupled with duct acoustics, J. Fluid Mech. **752**, 237 (2014).

[10] A. Orchini and M. P. Juniper, Linear stability and adjoint sensitivity analysis of thermoacoustic networks with premixed flames, Combust. Flame **165**, 97 (2015).

[11] G. A. Mensah and J. P. Moeck, Acoustic damper placement and tuning for annular combustors: An adjoint-based optimization study, J. Eng. Gas Turbines Power **139**, 061501 (2017).

[12] G. A. Mensah, L. Magri, A. Orchini, and J. P. Moeck, Effects of asymmetry on thermoacoustic modes in annular combustors: A higher-order perturbation study, J. Eng. Gas Turbines Power **141**, 041030 (2018).

[13] J. G. Aguilar and M. P. Juniper, Thermoacoustic stabilization of a longitudinal combustor using adjoint methods, Phys. Rev. Fluids **5**, 083902 (2020).

[14] J. G. Aguilar and M. P. Juniper, Adjoint methods for elimination of thermoacoustic oscillations in a model annular combustor via small geometry modifications, in *Turbo Expo: Power for Land, Sea, and Air*, Vol. 51050 (American Society of Mechanical Engineers, New York, NY, 2018), p. V04AT04A054.

[15] F. Schaefer, L. Magri, and W. Polifke, A hybrid adjoint network model for thermoacoustic optimization, in *Turbo Expo: Power for Land, Sea, and Air* (American Society of Mechanical Engineers, New York, NY, 2021).

[16] A. P. Dowling, A kinematic model of a ducted flame, J. Fluid Mech. **394**, 51 (1999).

[17] F. Huhn and L. Magri, Optimisation of chaotically perturbed acoustic limit cycles, Nonlin. Dynam. **100**, 1641 (2020).

[18] P. Subramanian, S. Mariappan, R. I. Sujith, and P. Wahi, Bifurcation analysis of thermoacoustic instability in a horizontal Rijke tube, Inte. J. Spray Combust. Dynam. **2**, 325 (2011).

[19] L. Kabiraj, R. I. Sujith, and P. Wahi, Bifurcations of self-excited ducted laminar premixed flames, J. Eng. Gas Turbines Power **134**, 31502 (2011).

[20] H. Gotoda, H. Nikimoto, T. Miyano, and S. Tachibana, Dynamic properties of combustion instability in a lean premixed gas-turbine combustor, Chaos **21**, 013124 (2011).

[21] H. Gotoda, T. Ikawa, K. Maki, and T. Miyano, Short-term prediction of dynamical behavior of flame front instability induced by radiative heat loss, Chaos **22**, 033106 (2012).

[22] L. Kabiraj, A. Saurabh, P. Wahi, and R. I. Sujith, Route to chaos for combustion instability in ducted laminar premixed flames, Chaos **22**, 023129 (2012).

[23] V. Nair, G. Thampi, and R. I. Sujith, Intermittency route to thermoacoustic instability in turbulent combustors, J. Fluid Mech. **756**, 470 (2014).

[24] V. Nair and R. I. Sujith, A reduced-order model for the onset of combustion instability: Physical mechanisms for intermittency and precursors, Proc. Combustion Inst. press **35**, 3193 (2015).

[25] K. Kashinath, I. C. Waugh, and M. P. Juniper, Nonlinear self-excited thermoacoustic oscillations of a ducted premixed flame: Bifurcations and routes to chaos, J. Fluid Mech. **761**, 399 (2014).

[26] I. C. Waugh, K. Kashinath, and M. P. Juniper, Matrix-free continuation of limit cycles and their bifurcations for a ducted premixed flame, J. Fluid Mech. **759**, 1 (2014).

[27] A. Orchini, S. J. Illingworth, and M. P. Juniper, Frequency domain and time domain analysis of thermoacoustic oscillations with wave-based acoustics, J. Fluid Mech. **775**, 387 (2015).

[28] F. Huhn and L. Magri, Stability, sensitivity and optimisation of chaotic acoustic oscillations, J. Fluid Mech. **882**, A24 (2020).

[29] E. N. Lorenz, Deterministic nonperiodic flow, J. Atmos. Sci. **20**, 130 (1963).

[30] D. J. Lea, M. R. Allen, and T. W. Haine, Sensitivity analysis of the climate of a chaotic system, Tellus A: Dynam. Meteorol. Oceanogr. **52**, 523 (2000).

[31] D. J. Lea, T. W. N. Haine, M. R. Allen, and J. A. Hansen, Sensitivity analysis of the climate of a chaotic ocean circulation model, Quart. J. R. Meteorol. Soc. **128**, 2587 (2002).

[32] G. L. Eyink, T. W. Haine, and D. J. Lea, Ruelle's linear response formula, ensemble adjoint schemes and Lévy flights, Nonlinearity **17**, 1867 (2004).

[33] J. Thuburn, Climate sensitivities via a Fokker-Planck adjoint approach, Quarterly J. R. Meteorological Soc. **131**, 73 (2005).

[34] P. J. Blonigan and Q. Wang, Probability density adjoint for sensitivity analysis of the mean of chaos, J. Comput. Phys. **270**, 660 (2014).

[35] D. Lasagna, Sensitivity analysis of chaotic systems using unstable periodic orbits, SIAM J. Appl. Dynamical Systems **17**, 547 (2018).

[36] C. E. Leith, Climate response and fluctuation dissipation, J. Atmospheric Sci. **32**, 2022 (1975).

[37] R. V. Abramov and A. J. Majda, Blended response algorithms for linear fluctuation-dissipation for complex nonlinear dynamical systems, Nonlinearity **20**, 2793 (2007).

[38] R. V. Abramov and A. J. Majda, New approximations and tests of linear fluctuation-response for chaotic nonlinear forced-dissipative dynamical systems, J. Nonlin. Sci. **18**, 303 (2008).

[39] Q. Wang, Forward and adjoint sensitivity computation of chaotic dynamical systems, J. Comput. Phys. **235**, 1 (2013).

[40] Q. Wang, R. Hu, and P. Blonigan, Least squares shadowing sensitivity analysis of chaotic limit cycle oscillations, J. Comput. Phys. **267**, 210 (2014).

[41] A. Ni and Q. Wang, Sensitivity analysis on chaotic dynamical systems by nonintrusive least squares shadowing (NILSS), J. Comput. Phys. **347**, 56 (2017).

[42] P. J. Blonigan and Q. Wang, Multiple shooting shadowing for sensitivity analysis of chaotic dynamical systems, J. Comput. Phys. **354**, 447 (2018).

[43] A. Ni and C. Talnikar, Adjoint sensitivity analysis on chaotic dynamical systems by nonintrusive least squares adjoint shadowing (NILSAS), J. Comput. Phys. **395**, 690 (2019).

[44] N. Chandramoorthy and Q. Wang, A computable realization of Ruelle's formula for linear response of statistics in chaotic systems, arXiv:2002.04117.

[45] A. Ni, Fast linear response algorithm for differentiating stationary measures of chaos, arXiv:2009.00595.

[46] P. J. Blonigan, Adjoint sensitivity analysis of chaotic dynamical systems with nonintrusive least squares shadowing, J. Comput. Phys. **348**, 803 (2017).

[47] A. Ni, Hyperbolicity, shadowing directions and sensitivity analysis of a turbulent three-dimensional flow, J. Fluid Mech. **863**, 644 (2019).

[48] D. Ruelle, A review of linear response theory for general differentiable dynamical systems, Nonlinearity **22**, 855 (2009).

[49] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, Cambridge, MA, 2016), http://www.deeplearningbook.org.

[50] S. Hochreiter and J. Schmidhuber, Long short-term memory, Neural Comput. **9**, 1735 (1997).

[51] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, Learning phrase representations using RNN encoder-decoder for statistical machine translation, arXiv:1406.1078.

[52] H. Jaeger and H. Haas, Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication, Science **304**, 78 (2004).

[53] M. Lukoševičius, *A Practical Guide to Applying Echo State Networks* (Springer, Berlin, 2012), pp. 659–686.

[54] J. Pathak, A. Wikner, R. Fussell, S. Chandra, B. R. Hunt, M. Girvan, and E. Ott, Hybrid forecasting of chaotic processes: Using machine learning in conjunction with a knowledge-based model, Chaos: An Inte. J. Nonlin. Sci. **28**, 041101 (2018).

[55] N. A. K. Doan, W. Polifke, and L. Magri, Physics-informed echo state networks for chaotic systems forecasting, in *Proceedings of the International Conference on Computational Science (ICCS)* (Springer International Publishing, Cham, 2019), pp. 192–198.

[56] Z. Lu, J. Pathak, B. Hunt, M. Girvan, R. Brockett, and E. Ott, Reservoir observers: Model-free inference of unmeasured variables in chaotic systems, Chaos **27**, 041102 (2017).

[57] N. A. K. Doan, W. Polifke, and L. Magri, Learning hidden states in a chaotic system: A physics-informed echo state network approach, in *Proceedings of the International Conference on Computational Science (ICCS)* (Springer International Publishing, Cham, 2020), pp. 117–123.

[58] A. Racca and L. Magri, Automatic-differentiated physics-informed echo state network (API-ESN), *Computational Science ICCS* (Springer International Publishing, Cham, 2021).

[59] F. Huhn and L. Magri, Learning ergodic averages in chaotic systems, in *Proceedings of the International Conference on Computational Science (ICCS)* (Springer International Publishing, Cham, 2020), pp. 124–132.

[60] A. G. Hart, J. L. Hook, and J. H. P. Dawes, Echo state networks trained by Tikhonov least squares are l2(μ) approximators of ergodic dynamical systems, Physica D **421**, 132882 (2021).

[61] M. P. Juniper, Triggering in the horizontal Rijke tube: Nonnormality, transient growth, and bypass transition, J. Fluid Mech. **667**, 272 (2011).

[62] L. V. King, On the convection of heat from small cyclinders in a stream of fluid: Determination of the convection constants of small platinum wires with applications to hot-wire anemometry, Proc. R. Soc. **214**, 373 (1914).

[63] M. A. Heckl, Active control of the noise from a Rijke tube, J. Sound Vib. **124**, 117 (1988).

[64] W. Polifke, A. Poncet, C. O. Paschereit, and K. Döbbeling, Reconstruction of acoustic transfer matrices by instationary computational fluid dynamics, J. Sound Vib. **245**, 483 (2001).

[65] A. Orchini, G. Rigas, and M. P. Juniper, Weakly nonlinear analysis of thermoacoustic bifurcations in the Rijke tube, J. Fluid Mech. **805**, 523 (2016).

[66] K. Balasubramanian and R. I. Sujith, Nonnormality and nonlinearity in combustion-acoustic interaction in diffusion flames, J. Fluid Mech. **594**, 29 (2008).

[67] L. D. Landau and E. M. Lifshitz, *Fluid Mechanics*, 2nd ed. (Pergamon Press, Oxford, UK, 1987).

[68] L. N. Trefethen, *Spectral Methods in MATLAB*, Vol. 10 (SIAM, Philadelphia, PA, 2000).

[69] C. A. Kennedy, M. H. Carpenter, and R. Lewis, Low-storage, explicit Runge-Kutta schemes for the compressible Navier-Stokes equations, Appl. Numer. Math. **35**, 177 (2000).

[70] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning* (MIT Press, Cambridge, MA, 2006).

[71] A. Racca and L. Magri, Robust optimization and validation of echo state networks for learning chaotic dynamics, Neural Networks **142**, 252 (2021).

[72] M. Hoffman, E. Brochu, and N. de Freitas, Portfolio allocation for bayesian optimization, in *Proceedings of the 27th Conference on Uncertainty in Artificial Intelligence (UAI)* (AUAI Press, Arlington, VA, 2011), pp. 327–336.

[73] J. Yperman and T. Becker, Bayesian optimization of hyper-parameters in reservoir computing, arXiv:1611.05193.

[74] J. Reinier Maat, N. Gianniotis, and P. Protopapas, Efficient optimization of echo state networks for time series datasets, *Proceedings of the International Joint Conference on Neural Networks (IJCNN)* (IEEE, 2018), pp. 1–7.

[75] S. Kullback and R. A. Leibler, On information and sufficiency, Ann. Math. Stat. **22**, 79 (1951).

[76] D. Scott, *Multivariate Density Estimation: Theory, Practice, and Visualization* (John Wiley & Sons, New York, NY, 1992).

[77] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey *et al.*, SciPy 1.0: Fundamental algorithms for scientific computing in Python, Nat. Methods **17**, 261 (2020).