


Uncertainty quantification of sensitivities of time-average quantities in chaotic systems

Kyriakos D. Kantarakias^{✉,*}, Karim Shawki,[†] and George Papadakis^{✉,‡}
Department of Aeronautics, Imperial College, London SW7 2AZ, United Kingdom

 (Received 21 November 2019; accepted 10 February 2020; published 28 February 2020)

We consider time-average quantities of chaotic systems and their sensitivity to system parameters. When the parameters are random variables with a prescribed probability density function, the sensitivities are also random. The central aim of the paper is to study and quantify the uncertainty of the sensitivities; this is useful to know in robust design applications. To this end, we couple the nonintrusive polynomial chaos expansion (PCE) with the multiple shooting shadowing (MSS) method, and apply the coupled method to two standard chaotic systems, the Lorenz system and the Kuramoto-Sivashinsky equation. The method leads to accurate results that match well with Monte Carlo simulations (even for low chaos orders, at least for the two systems examined), but it is costly. However, if we apply the concept of shadowing to the system trajectories evaluated at the quadrature integration points of PCE, then the resulting regularization can lead to significant computational savings. We call the new method shadowed PCE (sPCE).

DOI: [10.1103/PhysRevE.101.022223](https://doi.org/10.1103/PhysRevE.101.022223)

I. INTRODUCTION

All practical systems exhibit a degree of uncertainty in the values of the system parameters. For example, the angle of attack or the geometric shape of an airfoil exhibit a degree of randomness (due to free-stream turbulence or surface roughness, respectively). Uncertainty quantification (UQ) methods aim at modeling the effect of uncertainties in a computationally efficient manner. They define a quantity of interest (QoI), for example, lift or drag of an airfoil, and a set of uncertain parameters (such as angle of attack, airfoil shape, etc.) that affect the QoI. The objective of UQ methods is to evaluate the statistical behavior of the QoI using the available probabilistic information of the uncertain parameters and the input-output relation of the system. In this paper, we consider systems in which the input-output relation is governed by a set of nonlinear partial differential equations (PDEs). For a review of the available UQ methods with a particular focus on Computational fluid dynamics applications (where the governing PDEs are the Navier-Stokes equations), the reader is referred to Refs. [1,2].

The most commonly used UQ method is the polynomial chaos expansion (PCE), due to its computational efficiency for a small number of uncertain parameters. The mathematical framework was developed by Wiener [3] for Hermite polynomials and was later generalized in Ref. [4] for every polynomial in the Wiener-Askey scheme. The method uses a spectral representation of the uncertain quantities in an orthonormal stochastic space and computes the spectral coefficients with Galerkin projection.

There are two approaches for applying PCE to dynamical systems, the intrusive and nonintrusive; below they will be

denoted as iPC and niPC, respectively. In niPC, the QoI is written in spectral form and the unknown expansion coefficients are computed by evaluating the Galerkin integral at specific integration points (or nodes). Each evaluation requires the integration of the PDE system for a particular set of values of the input random variables. In that sense, niPC is a black-box approach, i.e., all that is required is a code that provides the solution of the PDE system, which is called repeatedly for a set of specific inputs determined by the PCE algorithm. However, in the iPC [5], the system degrees of freedom are written in spectral form and a new set of equations, the iPC equations, that describe the evolution of the expansion coefficients, are extracted using Galerkin projection. This results in a coupled system of equations that must be integrated, usually with methods similar to the ones used for the original dynamical system.

In steady systems, i.e., ones that do not exhibit time-dependent behavior, iPC is a well-established method with numerous applications in compressible and incompressible flows, reacting flows, flows in media, etc. (refer to Refs. [6–8] for a small sample of applications).

Application of UQ to unsteady systems is much more involved. First, the iPC method was found to exhibit significant limitations. More specifically, as the unsteady system evolves from an initial state, the number of expansion polynomials must increase to maintain the accuracy of the expansion [9–14]. The systems do not necessarily have to exhibit chaotic behavior for this problem to appear; the root of the problem lies on the nonlinear interactions that are present in either chaotic or nonchaotic systems. This is aptly demonstrated in Ref. [9] using a linear decay equation with a uniformly distributed random decay coefficient (the limits of the distribution are such that the latter always remain positive). Since both the coefficient and the solution are random, a quadratic nonlinearity appears, even though the deterministic equation is linear. As the system evolves, the solution starts to develop its own random characteristics, that deviate from those close

*k.kantarakias@imperial.ac.uk

†karim.shawki14@imperial.ac.uk

‡Corresponding author: g.papadakis@imperial.ac.uk

to the initial state. In other words, the probability density function (PDF) of the solution evolves in time and orthogonal polynomials that were initially optimal, are losing their optimality at later time instants. As a result, more and more terms are required in the expansion, but this only postpones the problem to a later time instant. For a given number of terms, the error will become unacceptable after some time. A second issue for unsteady problems, related to the importance of rare events, was highlighted by Branicki and Majda [10] using the same decay equation. If the decay coefficient assumes Gaussian distribution instead of uniform, then negative values are always possible. However, the presence of even a single event with negative decay coefficient (no matter how rare) will, after some time, dominate the statistics of the solution, leading eventually to unbounded results. This example demonstrates that very rare events can dominate the mean and standard deviation of the QoI; UQ of time dependent systems requires extreme caution.

The previous investigations have focused on quantifying the statistical behavior of the solution at specific points in time. Very few studies have investigated the uncertainty of time-averaged QoI's. Meldi *et al.* [15] applied niPC to quantify the error in LES of a spatially evolving mixing layer and its sensitivity to three simulation parameters (grid stretching in the streamwise and lateral directions and the Smagorinsky model constant). The QoI's were the mean streamwise velocity, the momentum thickness, and the shear stress. Note that although the QoI's were mean values (i.e., independent of time), they were obtained from time-averaging of the solution of an unsteady model, and therefore the problems with iPC mentioned in the previous paragraph were bound to occur; for this reason the niPC was applied. Lucor *et al.* [16] also applied niPC to study the sensitivity of Large Eddy Simulations to parametric uncertainty of the subgrid scale model in homogeneous isotropic turbulence.

In this paper, we are also concerned with the application of PCE to time-average quantities, obtained from long time integration of unsteady models. In particular, we are interested in the UQ of the sensitivities of these quantities with respect to the random system parameters (randomness affects not only the time-averages, but also their sensitivities). The latter are useful to know when performing for example gradient-based optimization under uncertainty [17–21].

At this point, the chaotic or nonchaotic behavior of the unsteady system plays the most crucial role. In systems that do not display chaotic behavior, there are well-established methods that can compute the sensitivity of a QoI with respect to system parameters, such as the adjoint method (refer to the review paper in Ref. [22]). Performing UQ to the sensitivities obtained with the adjoint technique is feasible with the niPC, in exactly the same way as for the time-average quantities discussed above.

However, the conventional adjoint approach fails when the system under investigation exhibits chaotic behavior, with one or more positive Lyapunov exponents [23]. Indeed, for such systems, the values of the adjoint variables grow exponentially in time and the computed sensitivities become meaningless. This is an important restriction because, in the area of fluid mechanics for example, most engineering flows become chaotic beyond a certain Reynolds number. In a series

of papers, Wang and co-workers have introduced the idea of least-squares shadowing (LSS) to address this problem [24–28]. Other approaches have also been proposed [29,30]. The LSS method is based on the shadowing lemma which is applicable for ergodic and uniformly hyperbolic systems [31–33]. The method locates a trajectory that shadows the reference one (i.e., remains always close to it), thereby regularizing the problem (for details refer to the above references). Recently, a pre-conditioner has been developed for a variant of LSS, called multiple shooting shadowing (MSS), that improves the condition number of the resulting linear system by orders of magnitude and makes the convergence rate almost independent from the number of degrees of freedom and the length of the trajectory in phase space [34]. The method relies on matrix-vector products only and is applicable to very large systems.

In this work, we couple the preconditioned MSS algorithm with niPC to compute the statistical behavior of the sensitivities of time-average quantities in chaotic systems. We reduce the computational cost of the coupled method by taking advantage of the shadowing lemma and reusing the preconditioner. We call the new method shadowed PCE (sPCE).

The paper is organized as follows: in Sec. II, we discuss the basic concepts of PCE and then present the MSS method. In Sec. III, we couple the two methods to create the new algorithm, which is applied to two standard chaotic systems, the Lorenz and the Kuramoto–Sivashinsky equation in Sec. IV. In Sec. V, we present sPCE, an approximate method with reduced computational cost, and we assess its accuracy. Finally, we conclude in Sec. VI.

II. BRIEF DESCRIPTION OF THE TWO CONSTITUENT METHODS: PCE AND MSS

A. The PCE method

We present below the basic concepts of PCE; for a more detailed exposition refer to Ref. [2]. We consider a general dynamical system governed by the evolution equation

$$\frac{d\mathbf{u}}{dt} = f(\mathbf{u}, s), \quad (1)$$

where the vector \mathbf{u} represents the state variables of the system and s is a set of design variables specifying the dynamics of Eq. (1). Uncertainty is introduced to variables s and it is modelled through a random vector $\boldsymbol{\xi} = (\xi_1, \dots, \xi_m)$ of m independent stochastic variables defined on an abstract probability space $\mathcal{P} = (\Omega, \Sigma, dP)$, where Ω is the set of random events, Σ is the σ -algebra of the events and P is the probability measure. We define a quantity of interest (QoI), $J = J(\boldsymbol{\xi})$, and the objective is to estimate the effect of the random variables s on the statistical moments of J .

Each component ξ_i is assumed to follow a known probability density function (PDF), $w_i(\xi_i)$, which is defined in the domain \mathcal{E}_i . Since the variables are independent, the vector $\boldsymbol{\xi}$ follows the PDF $W = \prod_{i=1}^m w_i(\xi_i)$, defined in the domain $\mathcal{E} = \prod_{i=1}^m \mathcal{E}_i$. The stochastic components ξ_i define a polynomial basis $\boldsymbol{\psi}^{(i)} = \{\psi_0^{(i)}, \psi_1^{(i)}, \dots\}$ with a well-defined tensor product, $\boldsymbol{\Psi} := \otimes_{i=1}^m \boldsymbol{\psi}^{(i)} = \{\Psi_0, \Psi_1, \dots\}$ that is orthogonal with

respect to the PDF W , i.e.,

$$\langle \Psi_j, \Psi_k \rangle = \int_{\mathcal{E}} \Psi_j \Psi_k W d\xi = \delta_{jk} \langle \Psi_j, \Psi_j \rangle, \quad (2)$$

where δ_{jk} is Kronecker's symbol (repeated superscripts do not imply summation).

In niPC, the QoI $J(\xi)$ is written in terms of the basis polynomials Ψ_i , i.e., in spectral form as

$$J(\xi) = \sum_{i=0}^{\infty} J^i \Psi_i(\xi). \quad (3)$$

In practice, the PCE expansion Eq. (3) is truncated after $q+1$ terms, and only polynomials up to degree C are kept (C is known as the chaos order). The number of independent variables, m , and the order of chaos, C , determine the number of terms $q+1 = \frac{(m+C)!}{m!C!}$. Using the orthogonality condition Eq. (2), the spectral coefficients J^i can be computed by Galerkin projection as

$$J^i = \int_{\mathcal{E}} J \Psi_i W d\xi \approx \sum_{j=1}^{\alpha} \omega_j J(\xi_j) \Psi_i(\xi_j). \quad (4)$$

The integral over the space \mathcal{E} is approximated numerically using Gauss Quadrature. In Eq. (4), α is the number of terms, while the weights ω_j and the quadrature points ξ_j at which $J(\xi_j)$ must be evaluated, depend on the PDFs of the random variables; their values for several standard PDFs can be found in Ref. [2]. Once the spectral coefficients J^i are known, the mean value and standard deviation of $J(\xi)$ can be computed from

$$\mu_J = J^0, \quad \sigma_J = \sqrt{\sum_{i=1}^q (J^i)^2}. \quad (5)$$

In this paper, we consider long time-averaged quantities defined as

$$\bar{G}_{\infty}(s) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T G(u, s) dt, \quad (6)$$

and our QoI are the sensitivities $J = \frac{d\bar{G}_{\infty}(s)}{ds}$, which is a vector with the same dimension as s . To compute the mean and standard deviation, the sensitivities must be evaluated at the quadrature points ξ_j ; see Eq. (4). We compute these using the multiple shooting shadowing method (MSS) as explained in the next subsection.

Before we proceed, however, we need to recall a fundamental relation between the PDFs of input and output. If the mapping of the input s to the QoI $J = g(s)$ is one-to-one and has an inverse, then the PDF of J , $f_J(J)$, is related to the PDF of s , $f_s(s)$, by the formula

$$f_J(J) = f_s(g^{-1}(J)) \left| \frac{d}{dJ} g^{-1}(J) \right|, \quad (7)$$

where $s = g^{-1}(J)$ is the inverse mapping. If $g(s)$ is a continuous and monotonic increasing or decreasing function, then the mapping is one-to-one. If the function $g(s)$ is not monotonic, then the contributions of all roots of the equation $J = g(s)$ for a given value of J must be included and the above formula becomes slightly more complicated. Equation (7) can be easily

derived using the principle of probability conservation (refer to Refs. [2,35]). In the particular case of a linear mapping between input and output, for example, $J = \alpha_1 s + \alpha_0$, the above formula simplifies to

$$f_J(J) = f_s \left(\frac{J - \alpha_0}{\alpha_1} \right) \left| \frac{1}{\alpha_1} \right|, \quad (8)$$

which indicates that the shape of two PDFs are the same, and the output PDF is obtained from the input PDF by scaling and a linear variable transformation. In this case, the mean and standard deviation of the QoI can be easily obtained analytically from the corresponding values of $f_s(s)$. PCE expansions with order $C = 1$ do not alter the shape of the input PDF; i.e., they are exact for this case. Relation Eq. (7) can be extended to include more than one random inputs and outputs, in which case the inverse of the Jacobian of the transformation appears on the right-hand side.

There is one more observation that we need to make that will prove useful later. In this paper we quantify the uncertainties of sensitivities, $J = \frac{d\bar{G}_{\infty}(s)}{ds}$. If the function $J = g(s)$ is a polynomial, then its order will be smaller (by one) compared to the order of $\bar{G}_{\infty}(s)$. If, for example, $\bar{G}_{\infty}(s)$ is second order, then the PDF $f_{G_{\infty}}(G_{\infty})$ will be distorted compared to $f_s(s)$ [according to Eq. (7)], but the PDF of $J = \frac{d\bar{G}_{\infty}(s)}{ds}$ will remain intact [according to Eq. (8)]. This indicates that it is easier (i.e., requires smaller chaos orders C) to quantify the uncertainty of $\frac{d\bar{G}_{\infty}(s)}{ds}$ compared to that of $\bar{G}_{\infty}(s)$.

B. Computation of sensitivities using MSS

Conventional methods for the computation of sensitivity, $J = \frac{d\bar{G}_{\infty}(s)}{ds}$, include the linearization of Eq. (1) around the trajectory in phase space evaluated at s . If the underlying system has one or more positive Lyapunov exponents, however, then standard methods fail because the trajectories evaluated at s and $s + ds$ diverge exponentially [23]. Wang *et al.* [25] introduced the least-squares shadowing (LSS) algorithm to regularize the problem. The idea is to find a trajectory at $s + ds$ that stays in close proximity (i.e., shadows) the reference trajectory at s . The two trajectories have different initial conditions, but for ergodic systems this does not affect the time-average (or its sensitivity). The shadowing trajectory is guaranteed to exist for uniformly hyperbolic systems [32]. Blonigan and Wang [28] introduced the MSS algorithm, which is a variant of LSS. For the two trajectories to remain close to each other, the reference trajectory is split into K segments and the following minimization problem is solved:

$$\min_{\mathbf{v}(t_i^+)} \sum_{i=0}^{K-1} \|\mathbf{v}(t_i^+)\|_2^2, \quad (9a)$$

$$\text{subject to } \mathbf{v}(t_i^+) = \mathbf{v}(t_i^-) \quad (i = 1, \dots, K-1), \quad (9b)$$

$$\frac{d\mathbf{v}}{dt} - \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \mathbf{v} - \frac{\partial \mathbf{f}}{\partial s} - \eta \mathbf{f} = 0 \quad t_i < t < t_{i+1} \quad (i = 0, \dots, K-1), \quad (9c)$$

$$\langle \mathbf{f}(\mathbf{u}(t), s), \mathbf{v}(t) \rangle = 0 \quad t_i < t < t_{i+1} \quad (i = 0, \dots, K-1), \quad (9d)$$

where

$$\mathbf{v}(t) = \lim_{\delta s \rightarrow 0} \frac{\mathbf{u}(\tau(t); s + \delta s) - \mathbf{u}(t; s)}{\delta s}$$

and
$$\eta(t) = \frac{d}{ds} \left(\frac{d\tau(t)}{dt} - 1 \right). \quad (10)$$

Equation (9a) defines the norm to be minimized; it is the sum of the distances (squared) between the two trajectories in phase plane evaluated at the $K + 1$ points, t_0, \dots, t_K , that define the K segments. The superscripts $-$ and $+$, for example, in t_i^- and t_i^+ , denote time instants immediately before and after t_i , respectively. Equation (9b) imposes the continuity of $\mathbf{v}(t)$ between two consecutive segments, and Eq. (9c) is the evolution equation for $\mathbf{v}(t)$ at each segment. The latter is derived by linearizing Eq. (1) around $\mathbf{u}(t; s)$ and using definitions Eq. (10). Finally, Eq. (9d) imposes orthogonality between trajectories $\mathbf{v}(t)$ and $\mathbf{f}(\mathbf{u}(t), s)$.

This is a standard minimization problem that results in the solution of a linear system of equations,

$$S\mathbf{w} = \mathbf{b}, \quad (11)$$

where S is a block symmetric, positive definite matrix, and $\mathbf{w} = [\mathbf{w}(t_1) \ \mathbf{w}(t_2) \ \dots \ \mathbf{w}(t_K)]^T$ are the discrete adjoint variables related to the minimization problem Eq. (9). All the steps for the derivation of Eq. (11) from Eq. (9) can be found in Ref. [28]. To reduce the condition number of system Eq. (11), a block-diagonal preconditioner, M_{BD} , was proposed in Ref. [34] that annihilates the growth of the faster modes of S . Together with a regularization parameter γ that deals with the very small eigenvalues of S , the above system is written as

$$(\gamma I + M_{BD}S)\mathbf{w} = M_{BD}\mathbf{b}. \quad (12)$$

The introduction of γ and M_{BD} result in several orders of magnitude reduction in the condition number of S , and makes the convergence of standard algorithms, like Conjugate Gradient, almost independent of the number of degrees of freedom of the system as well the number of segments K . After solving Eq. (12) for \mathbf{w} , the QoI $J = \frac{d\bar{G}_\infty(s)}{ds}$ can be easily computed (refer to Refs. [28,34]). We are now ready to couple the two algorithms to predict the statistical behavior of J . This is the subject of the next section.

III. COUPLING OF MSS AND PCE

The sensitivities are treated as any other QoI and are written in terms of the truncated polynomial basis Ψ_i [refer to Eq. (3)] as

$$J = \frac{d\bar{G}_\infty}{ds} \approx \sum_{i=0}^q \frac{d\bar{G}_\infty}{ds} \Psi_i(\xi), \quad (13)$$

and the spectral coefficients are computed from

$$J^i = \frac{d\bar{G}_\infty}{ds} \Big|_s = \int_{\mathcal{E}} \frac{d\bar{G}_\infty}{ds} \Psi_i W d\xi \approx \sum_{j=1}^{\alpha} \omega_j \frac{d\bar{G}_\infty}{ds}(\xi_j) \Psi_i(\xi_j). \quad (14)$$

In the equation above, only the sensitivities $\frac{d\bar{G}_\infty}{ds}(\xi_j)$ evaluated at the quadrature points ξ_j are unknown, and can be computed using the MSS algorithm described in the previous

TABLE I. Mean (μ_J) and standard deviation (σ_J) of sensitivity $J = \frac{d\bar{z}}{d\beta}$ for the Lorenz system for $\xi \sim N(0, 1)$. Comparison between niPC and Monte Carlo (5000 samples and random initial conditions). The Error(%) column refers to the discrepancy between niPC with $C = 2$ and Monte Carlo.

Uncertain input		$C = 1$	$C = 2$	Monte Carlo	Error (%)
$\beta = 2.5 + 0.1\xi$	μ_J	-1.5834	-1.5793	-1.5801	0.14
	σ_J	0.0579	0.0573	0.0601	4.66
$\beta = 2.5 + 0.05\xi$	μ_J	-1.5889	-1.5799	-1.5804	0.10
	σ_J	0.0318	0.0321	0.0324	1.01

section. Each quadrature point is treated separately and the computations can be fully parallelized.

This implementation is accurate but computationally expensive. Each nonintrusive evaluation requires the integration of Eq. (1) at the quadrature point ξ_j , the construction of the j th preconditioner M_{BDj} , and the solution of the system Eq. (12). Since the system is chaotic, the integration of Eq. (1) at the different points ξ_j results in entirely different trajectories and each matrix M_{BDj} must be constructed afresh. In the following section, we apply this method to two standard chaotic systems, and in Sec. V, we propose an approximate method to partially mitigate the computational cost.

IV. APPLICATION TO CHAOTIC SYSTEMS

Two common systems with (one or more) positive Lyapunov exponents are the Lorenz attractor and the Kuramoto-Sivashinsky equation. Both systems display chaotic behavior and are excellent test cases to apply the PCE/MSS coupling.

A. Application to the Lorenz system

The Lorenz system is

$$\frac{dx}{dt} = \sigma(y - x), \quad \frac{dy}{dt} = x(\rho - z) - y, \quad \frac{dz}{dt} = xy - \beta z. \quad (15)$$

We define the time-average quantity

$$\bar{G}_\infty = \bar{z} = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T z dt, \quad (16)$$

and the QoI is its sensitivity $J = \frac{d\bar{G}_\infty}{d\beta}$ with respect to the random variable $\beta = \beta_0 + \beta_1\xi$, with ξ following standard normal distribution, i.e., $\xi \sim N(0, 1)$. The Lorenz system was integrated in the domain $t \in [-1000, 200]$ and the first 1000 time units were discarded. The statistical moments of J for two values of β_1 obtained with niPC (with $C = 1$, $C = 2$) and Monte Carlo (MC) with $N = 5000$ samples are compared in Table I. Note that the sample size required by the Monte Carlo for well-converged statistics was found to be about half ($N \approx 2250$). To compute each Monte Carlo sample, we applied the MSS method for a random value of ξ that follows the aforementioned distribution and random initial conditions. As can be seen, there is very good agreement niPC and Monte Carlo simulations.

We repeated the simulations for three different probability density functions for ξ , Gaussian, uniform, and exponential.

TABLE II. Mean and standard deviation of sensitivity $J = \frac{d\bar{z}}{d\beta}$ for $\beta = 2.5 + 0.1\xi$, with $\xi \sim N(0, 1)$ (Gaussian distribution), $\xi \sim U(0, 1)$ (uniform distribution) and $\xi \sim E(0, 1)$ (exponential distribution), for $T = 200$ and $C = 1$. Results are compared with Monte Carlo ($N = 5000$ samples and random initial conditions).

		Gaussian	Uniform	Exponential
niPC ($C = 1$)	μ_J	-1.5834	-1.5667	-1.7259
	σ_J	0.0579	0.0623	0.0083
Monte Carlo	μ_J	-1.5801	-1.5696	-1.7386
	σ_J	0.0601	0.0641	0.0081
Error (%)	μ_J	0.57	0.51	1.72
	σ_J	3.67	2.80	2.47

The results are shown in Table II, again demonstrating good agreement.

To explain the good matching between niPC for small chaos orders C and Monte Carlo, in Fig. 1 we plot the variation of the deterministic value of the QoI with β in the region 2–3 (the results were obtained using integration time $T = 500$). The variation has finite but small curvature and is close to linear; therefore, good results are expected with small values of C . Note the better matching between niPC and MC for $\beta_1 = 0.05$ in Table I (the difference is around 1%). This is because for smaller β_1 , β falls within a region of the plot where the sensitivity variation is very close to linear. This is exactly the case we referred to at the end of Sec. II A, where the sensitivity varies linearly with the random variable, thus making the UQ easier.

B. Application to the Kuramoto-Sivashinsky equation

The extended Kuramoto-Sivashinsky (KS) system [24] can be expressed in mathematical form as

$$\begin{aligned}
 u_t &= -(u+c)u_x + u_{xx} + u_{xxx} \text{ with } x \in [0, L], \\
 u(0, t) &= u(L, t) = 0, \\
 u_x(0, t) &= u_x(L, t) = 0.
 \end{aligned} \tag{17}$$

Partial differentiation is denoted by a subscript (t or x) and the number of subscripts is equal to the derivative order. The u_{xx} term destabilizes the system playing the role of energy source, while u_{xxx} provides the necessary dissipation. The existence

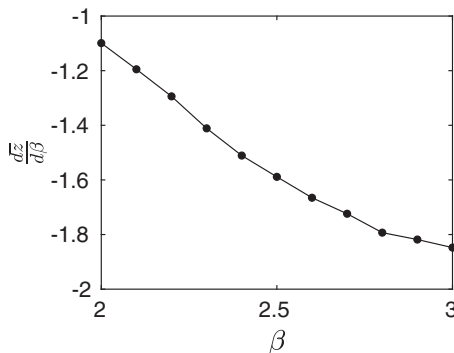


FIG. 1. Variation of the deterministic value of QoI $J = \frac{d\bar{z}}{d\beta}$ with β for the Lorenz system.

TABLE III. Mean and standard deviation of $J = \frac{d(\bar{u})}{dc}$ for $c = 0.5 + 0.1\xi$, with $\xi \sim N(0, 1)$. Comparison between niPC for $C = 1$, $C = 2$ and Monte Carlo with $N = 5000$ samples and random initial conditions (Tikhonov regularization parameter $\gamma = 0.1$, number of time segments $K = 10$, $T = 100$). The error is between niPC (with $C = 2$) and the MC (the latter is considered as reference).

	$C = 1$	$C = 2$	MC MSS	Error (%)
μ_J	-0.9031	-0.9068	-0.9095	0.30
σ_J	0.0080	0.0085	0.0090	5.56

of chaotic solutions is guaranteed by setting L to a large value, in the present case $L = 128$ [36], while the Dirichlet and Neumann boundary conditions guarantee ergodicity of the solutions [24]. The spatial derivatives are discretized with a second order finite difference scheme. A variable step fourth-order Runge-Kutta is used for time integration of the discrete equations.

Two time-averaged quantities are considered,

$$\begin{aligned}
 \langle \bar{u} \rangle &= \lim_{T \rightarrow \infty} \frac{1}{TL} \int_0^T \int_0^L u \, dx \, dt \text{ and} \\
 \langle \bar{u}^2 \rangle &= \lim_{T \rightarrow \infty} \frac{1}{TL} \int_0^T \int_0^L u^2 \, dx \, dt,
 \end{aligned} \tag{18}$$

and their sensitivities with respect to the parameter c , $J = \frac{d(\bar{u})}{dc}$ and $J = \frac{d(\bar{u}^2)}{dc}$, are the QoI's. Uncertainty is introduced through $c = c_0 + c_1\xi$, where ξ is a random variable. For the deterministic system, the effect of the fixed value of c on the above sensitivities was studied in Ref. [24].

1. Quantity of interest $J = \frac{d(\bar{u})}{dc}$

We consider the case $c = 0.5 + 0.1\xi$, with ξ following standard normal distribution, i.e., $\xi \sim N(0, 1)$. Comparison of the mean and standard deviation obtained with niPC using 2 chaos orders ($C = 1$ and 2) and the Monte Carlo method (MC) with $N = 5000$ samples is shown in Table III (note that we use capital C for the chaos order and lower case c for the system parameter). It can be seen that the niPC and MC results match very closely. Very small values of standard deviation are obtained (two orders of magnitude smaller than the mean), because as demonstrated in Ref. [24], the value of $J = \frac{d(\bar{u})}{dc}$ is constant in the range of c from 0 to approximately 1.2. The parameters $c_0 = 0.5$ and $c_1 = 0.1$ are such that all Monte Carlo samples are in this region. The standard deviation takes a nonzero value because of the finite averaging time (here $T = 100$). We have performed additional simulations with $T = 200$ and 1000 and confirmed that the standard deviation tends to 0, as expected.

2. Quantity of interest $J = \frac{d(\bar{u}^2)}{dc}$

Results for the mean value and the standard deviation for $c_0 = 0.5$ and for two values of $c_1 = 0.1$ and 0.05 are summarized in Table IV. Comparisons with MC demonstrate very good agreement.

We also performed simulations with chaos order $C = 3$. In Fig. 2, we plot the four spectral coefficients J^i [refer

TABLE IV. Mean and standard deviation of $J = \frac{d(\overline{u^2})}{dc}$ for $\xi \sim N(0, 1)$. Comparison between niPC with Monte Carlo with $N = 5000$ samples (Tikhonov regularization parameter $\gamma = 0.1$, number of time segments $K = 20$, $T = 200$). The error is between niPC (with $C = 2$) and the MC (the latter is considered as reference).

Uncertain input		$C = 1$	$C = 2$	MC MSS	Error (%)
$c = 0.5 + 0.1\xi$	μ_J	0.7667	0.7654	0.7642	0.16
	σ_J	0.1431	0.1479	0.1459	1.30
$c = 0.5 + 0.05\xi$	μ_J	0.7617	0.7601	0.7609	0.11
	σ_J	0.0741	0.0779	0.0759	2.75

to Eq. (14)] and we compare with the results of MC. The coefficients were computed as functions of time by applying the niPC to the QoI at each time step. The results from the Monte Carlo simulation were computed at the last time instant, $T = 200$. It is clear that the results from the niPC converge to those from the MC for time larger than about 20, which corresponds to about 2 Lyapunov timescales (see Fig. 9 in Ref. [24]). Note also that the magnitude of the coefficients decreases rapidly (the fourth coefficient, J^3 , is two orders of magnitude smaller than the first, J^0) indicating that the series

expansion Eq. (13) converges very fast. This explains why in Table IV good results are also obtained with the smaller chaos order, $C = 2$.

In the two chaotic systems examined, good results for mean and standard deviation of sensitivities were obtained with relatively small chaos orders. This result is of course case dependent, but a plausible explanation is the reduced order of the underlying polynomial mapping the random input variable to the sensitivity, as explained above, at the end of Sec. II A. In the literature, there are very few applications of niPC to time-averaged quantities of chaotic systems (and to the best of our knowledge none refers to sensitivities). There is some evidence that good results can be obtained with small values of C . For LES of a spatially evolving mixing layer, values of C up to 3 were used [15]. In Fig. 19 of Ref. [15], the authors show the effect of C and comment that 40–70% of the total variance of momentum thickness at one particular location is due to polynomials of order $C = 1, 2$ while third-order polynomials affect the solution by 1–5%. In Ref. [16] the error for the mean value of kinetic energy in a decaying homogenous isotropic turbulence predicted using niPC with $C = 2$ is less than 10% (for the large interval of the Smagorinsky constant examined) and drops to 1% for the small interval (refer to Fig. 3(b) in Ref. [16]). Even for enstrophy, which depends on

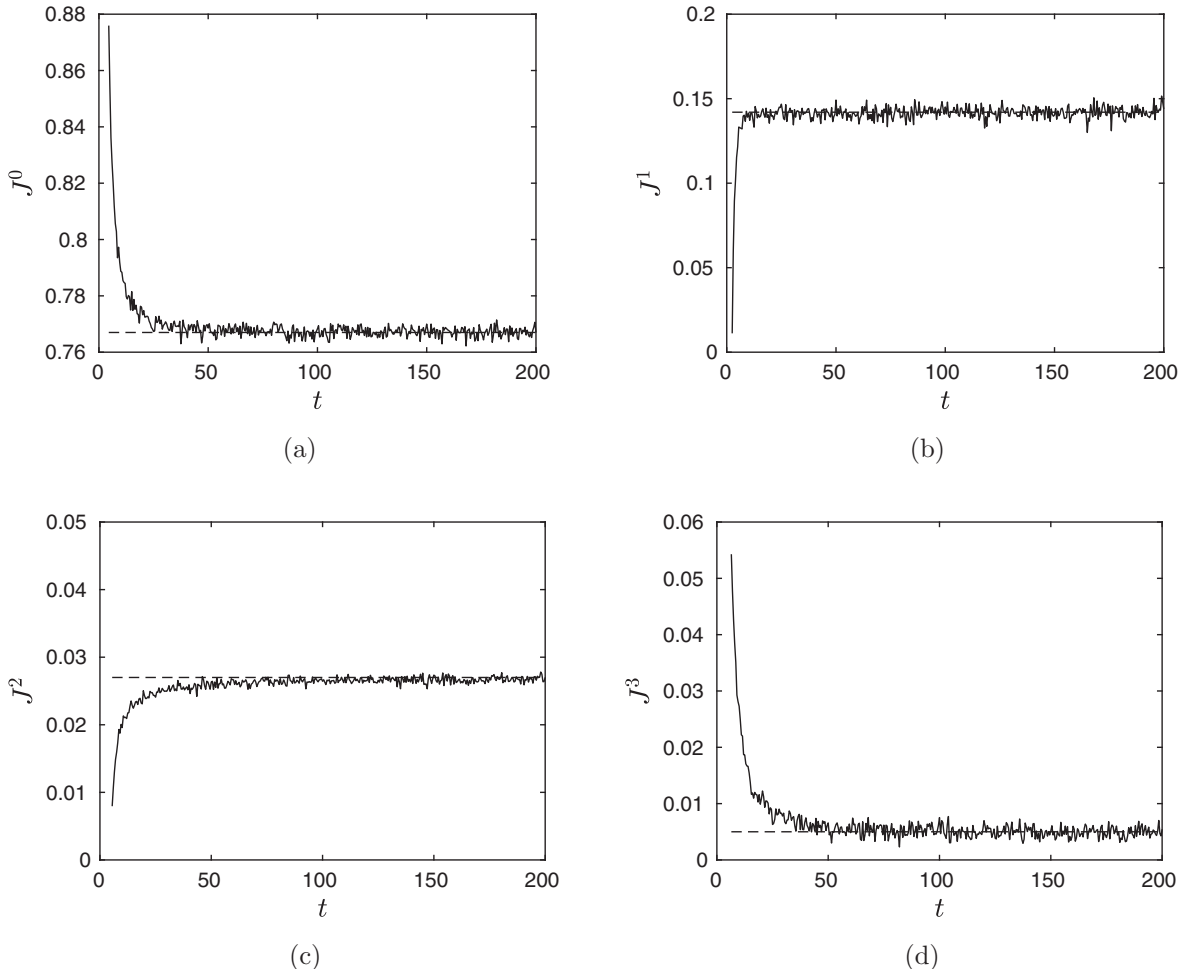


FIG. 2. Convergence of the first four spectral coefficients of $J = \frac{d(\overline{u^2})}{dc}$ with time and comparison with MC (dashed line). Initial conditions are random.

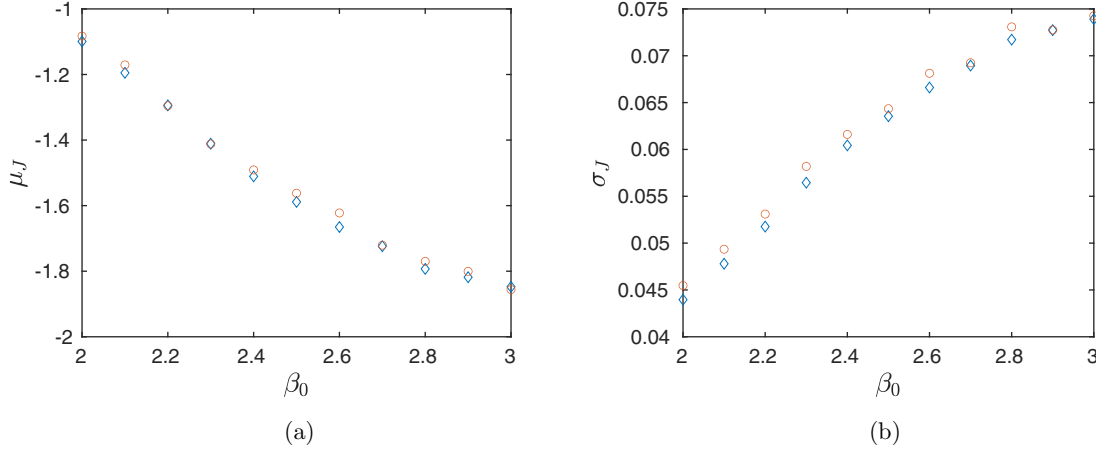


FIG. 3. Comparison of statistical moments of $J = \frac{d\bar{z}}{d\beta}$ obtained from using niPC (blue) and sPCE (red) for different values of mean β_0 . (a) mean value; (b) standard deviation.

the very small scales of motion, the error is of the order of 3% for $C = 2$ and the small Smagorinsky interval.

It is clear that the accuracy of low chaos orders depends on the QoI, the PDF and the standard deviation of the input variables and of course the dynamical system itself that maps input to the output. If, however, the input variables have small standard deviation (as in the case of small Smagorinsky interval above) and we are concerned with the statistical moments of sensitivities, then a good first approximation may be obtained even for $C = 1$. This observation provides the motivation for the development of a more efficient, but approximate, method which is detailed in the following section.

V. SHADOWING PCE (SPCE)

The approach presented in the previous section is very costly, as it requires one MSS solution for each integration point. To mitigate the computational cost, we propose here an approximate method. The idea is to find trajectories for the different parameters ξ_j that shadow each other. We then reuse the preconditioner M_{BD} and alleviate the corresponding cost (which can be considerable if the number of positive

Lyapunov exponents is large). We call the new approach shadowing PCE (sPCE).

More specifically, we consider the trajectory at one integration node, say ξ_1 , as reference and we compute the trajectories at the other nodes so that they shadow the reference one. We store variables $\mathbf{u}(t; \xi_1)$ and $\mathbf{v}(t; \xi_1)$ and compute the trajectories at the integration nodes from [refer to Eq. (10)]

$$\mathbf{u}(\tau(t); \xi_j) \approx \mathbf{u}(t; \xi_1) + \mathbf{v}_0(t; \xi_1) ds_j, \quad (19)$$

where $ds_j = \xi_j - \xi_1$ is the distance between integration nodes ξ_j and ξ_1 , which depends on the standard deviation of the input variables. Equation (19) is an approximation that is accurate for small standard deviations.

The computation of the exact trajectory that shadows a reference one is a nonlinear problem (the iterative solution algorithm can be found in Ref. [26]). In the present paper, we do not iterate (that would have increased the computational cost further) but apply instead the approximation Eq. (19). Computational savings arise from two reasons: first, we integrate the dynamical system only at the node ξ_1 , and

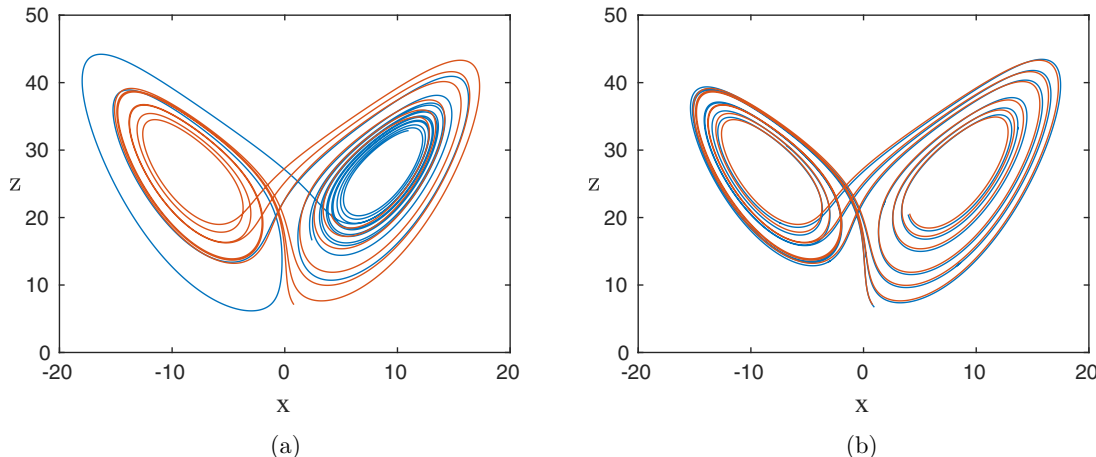


FIG. 4. Comparison of trajectories (projected in the x - z plane) at the two integration nodes $\xi_{1,2} = \beta_0 \pm \beta_1$ computed by niPC (a) and sPCE (b) for $m = 1$ uncertain variable and $C = 1$.

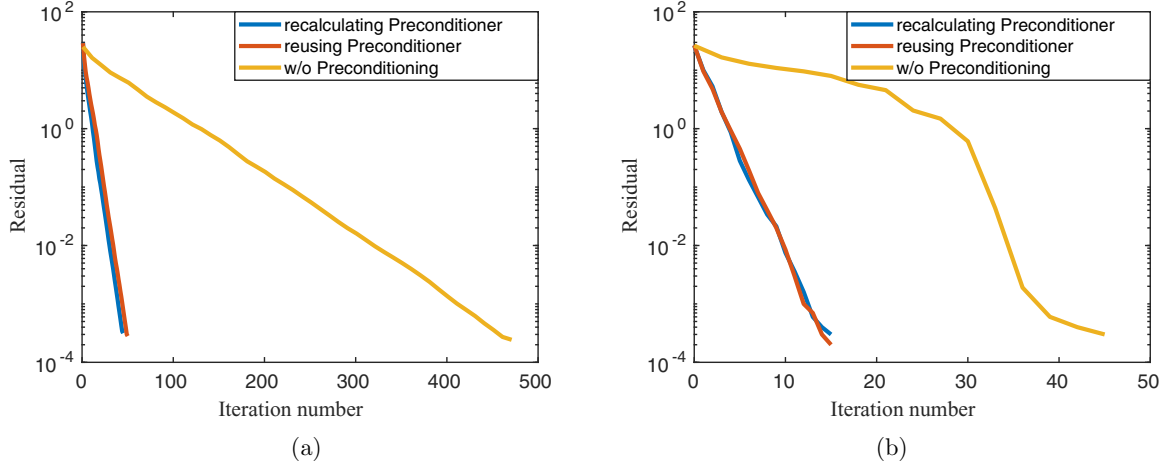


FIG. 5. Convergence rate of the KS (a) and the Lorenz (b) systems (for $c = 0.5 + 0.1\xi$ and $\beta = 2.5 + 0.1\xi$, respectively). Comparison of three techniques for system solution: computation of the preconditioner at each integration node separately, usage of the same preconditioner and solution without any preconditioner at all.

second, we can reuse the preconditioner M_{BD} computed at ξ_1 to solve the MSS system Eq. (12) for the rest of the ξ_j .

A few comments are warranted here on the nature of the exact PCE and the approximate one just presented. The exact PCE approach is global in nature, that is the dynamical system is integrated at the quadrature points ξ_j , exact information is collected at these points, and the spectral coefficients are evaluated from Eq. (14). The approximate approach is local, that is the behavior of the system at the far points ξ_j ($j \neq 1$) is obtained from the information available at ξ_1 only. This approximation becomes globally accurate only if the underlying function has linear behavior.

We proceed to compare the results obtained using the accurate implementation described in Sec. III and the approximate sPCE in terms of accuracy and cost for the Lorenz system with QoI $J = \frac{d\bar{G}_\infty}{d\beta}$, where \bar{G}_∞ is given in Eq. (16). In Fig. 3 we compare the mean and standard deviation of the QoI produced by the two methods for random $\beta = \beta_0 + \beta_1\xi$, $\xi \sim N(0, 1)$ with different mean values $\beta_0 = 2-3$ and constant $\beta_1 = 0.1$. The difference between the two approaches is very small, of the order of 2–3%. This is due to the (almost) linear underlying function variation shown in Fig. 1.

In Fig. 4, we plot the trajectories evaluated at the two integration nodes $\xi_{1,2} = \beta_0 \pm \beta_1$ (projected in the x/z plane of the phase space). The two trajectories deviate from each other for the niPC method (left panel), but remain close together for the sPCE (right panel), as expected.

We applied the same idea to the KS system for $J = \frac{d\langle \bar{u} \rangle}{dc}$ and we examined the effect of reusing the preconditioner on the number of iterations. Figure 5 shows the convergence rate for the two systems. In both cases, calculating M_{BD} afresh at each integration point (for niPC) or computing only at one point and reusing at the second (for sPCE) resulted in approximately the same convergence rate. For sPCE, the same matrix M_{BD} works well for both points because the corresponding trajectories shadow each other. Without a preconditioner, convergence deteriorates as expected.

In Table V we compare the computing time required by the two approaches for the Lorenz system (results are averaged over 200 random initial conditions). The run times are normalized by the total cost of node ξ_1 [which comprises the integration of the dynamical system Eq. (15), the construction of the preconditioner, and the solution of the linear system Eq. (12)]. As can be seen, the total cost is reduced by 27% for node ξ_2 in relation to node ξ_1 . This is due to the elimination of the cost of the construction of M_{BD} (which accounts for 21% of the total time, as shown in the 3rd column) and the reduced time for the computation of the shadowing trajectory [which is obtained directly from Eq. (19)]. If more uncertain variables are present, which necessitate the computation at other integration nodes, then we expect that this saving will be carried over to all the other nodes, ξ_j ($j \neq 1$), as well. In the last column we present the average error in the standard deviation of the sensitivity $J = \frac{d\bar{z}}{d\beta}$ as computed by the two approaches in the range $\beta_0 \in [2, 3]$; the error is quite small, less than 3%.

TABLE V. Comparison of computational time required for niPC and sPCE for the Lorenz system. The average error between the approaches is also given in the last column. One preconditioned MSS evaluation required on average 22 s of CPU time for $T = 50$ and $dt = 0.001$.

	Total cost per integration node (node ξ_1 + node ξ_2)	Cost of preconditioner (node ξ_1 + node ξ_2)	Average (node ξ_1 + node ξ_2)	Error
niPC	1.00 + 1.00	0.21 + 0.21	2.86%	
sPCE	1.00 + 0.73	0.21 + 0.00		

TABLE VI. Comparison of computational time required for niPC and sPCE for the KS system. The average error between the approaches is also given in the last column. One preconditioned MSS evaluation required on average 31 s of CPU time for $T = 50$ and $dt = 0.1$.

$(\xi_1 + \xi_2)$	Total Cost per integration node ($\xi_1 + \xi_2$)	Cost of preconditioner error	Average
niPC	1.00 + 1.00	0.41 + 0.41	2.50%
sPCE	1.00 + 0.54	0.41 + 0.00	

The same results for the KS system are presented in Table VI, where the cost of computing the preconditioner on each integration node amounts to approximately 41% of the total runtime. The KS results are also averaged over 200 initial conditions and $c \in [0, 1]$. The preconditioner is only evaluated once, resulting at computational savings of approximately 46% per node and an average error of 2.5% in the value of the computed sensitivity. All the computations were conducted in parallel in the six cores of an Intel Core i7 8700 CPU.

VI. CONCLUSIONS

Nonintrusive polynomial chaos, niPC, can be used for UQ of time-averaged QoIs and its accuracy doesn't deteriorate as the time frame of the simulation grows. In this paper the QoIs considered are the sensitivities of time-averages to design parameters. Knowledge of the standard deviation of the sensitivities is useful for design-optimization under uncertainty. We couple a recently proposed algorithm that evaluates these sensitivities (known as multiple shoot-

ing shadowing) with niPC, and we applied it successfully to two systems, the Lorenz and the Kuramoto-Sivashinsky equation.

The cost of the coupled PCE-MSS approach is dominated by the computation of the sensitivities at the Galerkin integration points. This cost can be mitigated with the use of the approximate sPCE method that we introduce in this paper. The idea is to compute approximate shadowing trajectories at the integration points and then reuse the resulting preconditioner. We showed that for the parameters used in the present investigation, reusing the preconditioner does not affect the convergence rate and results in significant savings (of the order of 27% per evaluation). The accuracy is also good, with errors less than 3%.

ACKNOWLEDGMENTS

K.D.K. acknowledges the financial support of the President's Scholarship Award from Imperial College London. K.S. wishes to acknowledge the financial support of Al-Alfi Foundation in the form of a PhD scholarship.

-
- [1] H. N. Najm, *Annu. Rev. Fluid Mech.* **41**, 35 (2009).
 - [2] O. L. Maître and O. M. Knio, *Spectral Methods for Uncertainty Quantification With Applications to Computational Fluid Dynamics* (Springer-Verlag, Berlin/Heidelberg, 2010).
 - [3] N. Wiener, *Am. J. Math.* **60**, 897 (1938).
 - [4] D. Xiu and G. Karniadakis, *J. Comput. Phys.* **187**, 137 (2003).
 - [5] R. G. Ghanem and P. D. Spanos, *Stochastic Finite Elements: A Spectral Approach* (Springer-Verlag, Berlin/Heidelberg, 1991).
 - [6] O. P. L. Maître, O. M. Knio, H. N. Najm, and R. G. Ghanem, *J. Comput. Phys.* **173**, 481 (2001).
 - [7] X. Chen, B. Ng, Y. Sun, and C. Tong, *J. Comput. Phys.* **248**, 383 (2013).
 - [8] M. Chatzimanolakis, K.-D. Kantarakias, V. Asouti, and K. Giannakoglou, *Comput. Meth. App. Mech. Eng.* **348**, 207 (2019).
 - [9] M. Gerritsma, J.-B. van der Steen, P. Vos, and G. Karniadakis, *J. Comput. Phys.* **229**, 8333 (2010).
 - [10] M. Branicki and A. Majda, *Commun. Math. Sci.* **11**, 55 (2013).
 - [11] C. Y. Shen, T. E. Evans, and S. Finette, *J. Atm. Oce. Tech.* **27**, 1059 (2010).
 - [12] O. L. Maître, H. Najm, R. Ghanem, and O. Knio, *J. Comput. Phys.* **197**, 502 (2004).
 - [13] O. P. L. Maître, L. Mathelin, O. M. Knio, and M. Y. Hussaini, *Disc. Cont. Dyn. Syst.* **28**, 199 (2010).
 - [14] Y. Che and C. Cheng, *Chaos, Solitons, and Fractals* **116**, 208 (2018).
 - [15] M. Meldi, M. V. Salvetti, and P. Sagaut, *Phys. Fluids* **24**, 035101 (2012).
 - [16] D. Lucor, J. Meyers, and P. Sagaut, *J. Fluid. Mech.* **585**, 255 (2007).
 - [17] M. B. Giles and N. A. Pierce, *Flow, Turbulence and Combustion* **65**, 393 (2000).
 - [18] E. M. Papoutsis-Kiachagias and K. C. Giannakoglou, *Arch. Comp. Meth. Eng.* **23**, 255 (2016).
 - [19] A. Jameson, *J. Sci. Comp.* **3**, 233 (1988).
 - [20] J. R. Martins, J. J. Alonso, and J. J. Reuther, *Optim. Eng.* **6**, 33 (2005).
 - [21] I. Kavvadias, E. Papoutsis-Kiachagias, and K. Giannakoglou, *J. Comput. Phys.* **301**, 1 (2015).
 - [22] P. Luchini and A. Bottaro, *Annu. Rev. Fluid Mech.* **46**, 493 (2014).
 - [23] G. L. Eyink, T. W. N. Haine, and D. J. Lea, *Nonlinearity* **17**, 1867 (2004).
 - [24] P. J. Blonigan and Q. Wang, *Chaos, Solitons, and Fractals* **64**, 16 (2014).
 - [25] Q. Wang, R. Hu, and P. Blonigan, *J. Comput. Phys.* **267**, 210 (2014).
 - [26] Q. Wang, S. A. Gomez, P. J. Blonigan, A. L. Gregory, and E. Y. Qian, *Phys. Fluids* **25**, 110818 (2013).

- [27] P. Blonigan and Q. Wang, *Num. Lin. Alge. App.* **24**, e1946 (2014).
- [28] P. J. Blonigan and Q. Wang, *J. Comput. Phys.* **354**, 447 (2018).
- [29] J. Craske, *Chaos, Solitons, and Fractals* **119**, 243 (2019).
- [30] D. Lasagna, *J. Appl. Dyn. Syst.* **17**, 547 (2018).
- [31] R. Bowen, *J. Diff. Eq.* **18**, 333 (1975).
- [32] S. Y. Pilyugin, *Shadowing in Dynamical Systems* (Springer-Verlag, Berlin, 1999).
- [33] P. Holmes, J. L. Lumley, G. Berkooz, and C. W. Rowley, *Turbulence, Coherent Structures, Dynamical Systems and Symmetry* (Cambridge University Press, Cambridge, 2012).
- [34] K. Shawki and G. Papadakis, *J. Comput. Phys.* **398**, 108861 (2019).
- [35] H. Hsu, *Signals and Systems*, 2nd ed., Schaum's Outline Series (McGraw Hill, New York, 2011).
- [36] J. M. Hyman and B. Nicolaenko, *Phys. D. Non. Phen.* **18**, 113 (1986).