Probabilistic learning on manifolds constrained by nonlinear partial differential equations for small datasets

C. Soize[a,*], R. Ghanem[b]

[a] *Université Gustave Eiffel, MSME UMR 8208 CNRS, 5 bd Descartes, 77454 Marne-la-Vallée, France*
[b] *University of Southern California, 210 KAP Hall, Los Angeles, CA 90089, United States*

Abstract

A novel extension of the Probabilistic Learning on Manifolds (PLoM) is presented. It makes it possible to synthesize solutions to a wide range of nonlinear stochastic boundary value problems described by partial differential equations (PDEs) for which a stochastic computational model (SCM) is available and which depend on a vector-valued random control parameter. The cost of a single numerical evaluation of this SCM is assumed to be such that only a limited number of points can be computed for constructing the training dataset (small data). Each point of the training dataset is made up of realizations from a vector-valued stochastic process (the stochastic solution) and the associated random control parameter on which it depends. The presented PLoM constrained by PDE allows for generating a large number of learned realizations of the stochastic process and its corresponding random control parameter. These learned realizations are generated so as to minimize the vector-valued random residual of the PDE in the mean-square sense. Appropriate novel methods are developed to solve this challenging problem. Three applications are presented. The first one is a simple uncertain nonlinear dynamical system with a nonstationary stochastic excitation. The second one concerns the 2D nonlinear unsteady Navier-Stokes equations for incompressible flows in which the Reynolds number is the random control parameter. The last one deals with the nonlinear dynamics of a 3D elastic structure with uncertainties. The results obtained make it possible to validate the PLoM constrained by stochastic PDE but also provide further validation of the PLoM without constraint.

Keywords: probabilistic learning, PLoM,, partial differential equations, small dataset, non-Gaussian, statistical constraints, stochastic computational models, Navier-Stokes equations, nonlinear solid dynamics, machine learning, data driven, data science, Kullback-Leibler, minimum cross-entropy, uncertainty quantification

1. Introduction

(i) Context. Physics-informed machine learning (eg. neural networks and deep learning) has seen a growing interest in recent years. Numerous works in this area of machine learning have been carried out for Schrödinger equation [1], for reconstructing Reynolds stress model in turbulent flow [2] and for augmenting turbulence model [3, 4], for predicting large eddy simulation [5], for projection-based model reduction [6]. Physics-informed machine learning has also been used in the context of linear [7], quasilinear [8], and nonlinear partial differential equations [9, 10]. All these approaches are based on the use of Neural Networks, which, in general, require big data for training. In machine learning, probabilistic or statistical learning deals with the problem of inferring a function based on data. In particular for supervised learning, it consists in training from a set of data consisting of input-output pairs. The statistical and probabilistic learning approaches have been extensively developed (see for instance, [11, 12, 13, 14, 15, 16, 17, 18, 19, 20]) and play an increasingly important role in computational physics and engineering science, in particular for problems using large scale computational models. In this framework, statistical methods for supervised learning have been developed in the form of surrogate models from which approximations of the expensive functions can be easily evaluated [21, 22, 23, 24, 25, 26, 27, 28, 29]. Among the many methodologies available on this topic, we are interested

*Corresponding author: C. Soize, christian.soize@univ-eiffel.fr
Email addresses: christian.soize@univ-eiffel.fr (C. Soize), ghanem@usc.edu (R. Ghanem)

here in *Probabilistic Learning on Manifolds* (PLoM), which was initially introduced in [30] and specially developed for solving unsupervised and supervised problems in the case of small data, as opposed to big data. The PLoM is not based on the use of neural networks. The first initial work published in 2016 was completed from a theoretical point of view in [31, 32], extended to the case of the polynomial chaos representation [33], to the case of the learning in presence of physics constraints [34], and to the sampling of Bayesian posteriors [35]. The PLoM has been used with success for probabilistic nonconvex optimization under constraints and uncertainties [36] that has allowed for analyzing very complex engineering systems such as hypersonic combustion flows [37, 38], optimal placement of wells [39], the design optimization under uncertainties of mesoscale implants [40], the quantification of model uncertainties in nonlinear computational dynamics [41, 42], the fracture paths in random composites [43], the digital twin for composite material systems [44], the computation of Sobol indices for complex stochastic system [45], and the detection of structural changes from dynamic monitoring [46].

(ii) Objective of the paper. As already mentioned, the PLoM is not based on the use of neural networks and was developed for the unsupervised or supervised cases of small data. For the supervised case, these small data constitute the training dataset that is constructed as N_d input-output pairs. The PLoM then allows, using only the training dataset, and by a relying on purely probabilistic arguments, to generate a large number $n_{\text{MC}} \gg N_d$ of additional realizations (the learned dataset) while preserving the concentration of the probability measure on an intrinsic manifold delineated using diffusion maps [47]. We are then entitled to ask the following fundamental question: do the n_{MC} additional realizations generated by the PLoM satisfy, in a "reasonable" sense, the stochastic Partial Differential Equation (PDE) that was used to build the training dataset. As previously explained, the PLoM has been validated for many situations and the convergence of the learning with respect to N_d has theoretically and numerically been studied. This paper will give further confirmation of this point. However, for certain problems, we want to be able to control that the additional realizations satisfy, with a quantifiable metric, the stochastic PDE underlying the stochastic computational model used to generate N_d points of the training dataset. This is the objective of this paper: to develop and validate the PLoM method constrained by the stochastic PDE that has been used to build the training dataset. It should be noted that the method requires evaluations of the residual of the PDE. As the proposed method is placed in a general framework, these evaluations cannot be obtained explicitly by an algebraic calculation. They can therefore only be calculated using the computational model associated with the stochastic boundary value problem. Under these conditions, the intrusive character of the numerical evaluations of the residue can arise. However, in the framework of the algorithm proposed in this paper, the commercial and in-house software allow, in general, such residual evaluations to be made in a nonintrusive manner with an almost negligible numerical cost (this is a possibility offered by the software used in computational sciences and engineering).

(iii) Novelty of the paper. Two novel extensions of the PLoM are presented, which allow to give a constructive answer to our objective. The first is to extend the initial PLoM for nonlinear stochastic dynamical systems (that is to say, physics system depending on time) such as those of the unsteady computational fluid dynamics (CFD) or the nonlinear solid dynamics (SD) in presence of a vector-valued random control parameter. Indeed the PLoM was developed for random vectors and it is therefore necessary to extend it to the case of vector-valued stochastic processes for analyzing these types of stochastic dynamical systems. The second is to extend the PLoM method to take into account the constraint that the generated additional realizations verify the stochastic PDE in a certain sense. These two novel extensions constitute a real challenge and correspond to a difficult problem, which will necessitate the introduction of appropriate methods.

(iv) Organization of the paper. In order to facilitate the reading of this work, we have chosen to present an abstract formulation in the first part in order to better introduce the concepts and methods. These developments are presented for a generic nonlinear stochastic computational dynamical model, which results from the spatial discretization of a nonlinear stochastic boundary value problem. We delay the introduction of the complexity in the stochastic boundary value problem until the sections where the respective applications are introduced. We can thus focus our attention initially on setting up the PLoM with PDE constraints. Later in the paper, we consider nonlinear PDEs with random coefficients. We intersperse in that presentation additional new developments required to efficiently implement our proposed procedure. We specifically introduce two applications that are representative of broad classes of problems, namely a first order (CFD case) and a second order (SD case) problems, which involve one field (SD case) or several

fields (CFD case) whose partial time derivatives are different according to the field in question (eg. velocity field and pressure field).

(v) Brief discussion on the hypotheses on which the PLoM method has been built [30, 32]. The PLoM approach starts from a training dataset made up of a relatively small number N_d of points (initial realizations). For the supervised case, it is assumed that the training dataset is related to an underlying stochastic manifold related to a $\mathbb{R}^{n_x}$-valued random variable $\mathbf{X} = (\mathbf{Q}, \mathbf{W})$ with $\mathbf{Q} = \mathbf{f}(\mathbf{U}, \mathbf{W}) = \mathbf{F}(\mathbf{W})$. The measurable mapping $\mathbf{f}$ is unknown and $\mathbf{F}$ is also an unknown stochastic mapping. The probability distributions of the vector-valued random variables $\mathbf{W}$ (control parameter) and $\mathbf{U}$ (non-controlled parameter) are given. The stochastic manifold is defined by the unknown graph $\{\mathbf{w}, \mathbf{F}(\mathbf{w})\}$ for $\mathbf{w}$ belonging to an admissible set C_w. In the PLoM construction, its is not assumed that this stochastic manifold can directly be described; for instance, it is not assumed that there exist properties of local differentiability (moreover, the manifold is stochastic). Under these conditions, the probability measure of $\mathbf{X}$ is concentrated in a region of $\mathbb{R}^{n_x}$ for which the only available information is the cloud of the points of the training dataset. The PLoM method makes it possible to generate the learned dataset whose $n_{\text{MC}} \gg N_d$ points (additional realizations) are generated by the probability measure that is estimated from the training dataset. The concentration of the probability measure is preserved thanks to the use of the diffusion-maps basis that allows to enrich the available information from the training dataset. It should also be noted that the estimate of this unknown probability measure cannot be performed from the training dataset by using an arbitrarily estimator. It must be parameterized in a manner that permits convergence to any probability measure as its number of points goes towards infinity. The PLoM method therefore does not only consist in generating points that belong to the region in which the measure is concentrated, but also allows these additional points to be realizations of the estimate probability measure with the convergence properties evoked above. The choice of the kernel estimation method for estimating the probability measure from the training dataset guarantees that this required fundamental property is satisfied [32]. Considering all of these assumptions in the construction and in the methodology of the PLoM makes PLoM as a general approach for small datasets for arbitrary non-Gaussian measures.

Notations

Lower-case letters such as q or η are deterministic real variables.
Boldface lower-case letters such as $\mathbf{q}$ or $\boldsymbol{\eta}$ are deterministic vectors.
Upper-case letters such as X or H are real-valued random variables.
Boldface upper-case letters such as $\mathbf{X}$ or $\mathbf{H}$ are vector-valued random variables.
Lower-case letters between brackets such as $[x]$ or $[\eta]$ are deterministic matrices.
Boldface upper-case letters between brackets such as $[\mathbf{X}]$ or $[\mathbf{H}]$ are matrix-valued random variables.

$[I_n]$: identity matrix in $\mathbb{M}_n$.
$\mathbb{M}_{n,N}$: set of all the $(n \times N)$ real matrices.
$\mathbb{M}_N$: set of all the square $(n \times n)$ real matrices.
$\mathbb{M}_n^{+0}$: set of all the positive symmetric $(n \times n)$ real matrices.
$\mathbb{M}_n^{+}$: set of all the positive-definite symmetric $(n \times n)$ real matrices.
$\mathbb{R}$: set of all the real numbers.
$\mathbb{R}^n$: Euclidean vector space on $\mathbb{R}$ of dimension n.
$[x]_{kj}$: entry of matrix $[x]$.
$[x]^T$: transpose of matrix $[x]$.
$\|\cdot\|$: usual Euclidean norm on $\mathbb{R}^n$.
$<\mathbf{x}, \mathbf{y}>$: usual Euclidean inner product in $\mathbb{R}^n$.
$\delta_{\alpha\beta}$: Kronecker's symbol.

In this paper, any Euclidean space $\mathcal{E}$ (such as $\mathbb{R}^N$) is equipped with its Borel field $\mathcal{B}_{\mathcal{E}}$, which means that $(\mathcal{E}, \mathcal{B}_{\mathcal{E}})$ is a measurable space on which a probability measure can be defined. The mathematical expectation is denoted by E.

2. Abstract nonlinear stochastic computational model, training dataset, and time sampling

2.1. Nonlinear stochastic equation with its initial condition

We consider the abstract nonlinear stochastic computational model (SCM) on $\mathbb{R}^N$ written as

$$\boldsymbol{\mathcal{N}}_t\big(\mathbf{Y}(t), t, \mathbf{W}\big) = \mathbf{0} \quad a.s. \quad , \quad t \in]t_0, T]\,, \tag{1}$$

$$\boldsymbol{\mathcal{L}}^{\mathrm{IC}}_{t_0}\big(\mathbf{Y}\big) = \mathbf{y}_0 \quad a.s. \quad , \quad t = t_0\,, \tag{2}$$

in which $[t_0, T]$ is a given compact interval of $\mathbb{R}$ and $\mathbf{Y} = \big\{\mathbf{Y}(t), t \in [t_0, T]\big\}$ is the unknown $\mathbb{R}^N$-valued stochastic process. The vector $\mathbf{W}$ is the $\mathbb{R}^{n_w}$-valued random control parameter of the system, which can correspond either to a set of system parameters, or to the set of generalized coordinates of a reduced-order representations based on the Karhunen-Loève or chaos expansions, (1) of random fields that are the coefficients of partial differential operators of the problem and/or (2) of stochastic processes modeling the time-dependent excitations of the system. In Eq. (1), $\boldsymbol{\mathcal{N}}_t$ is a given nonlinear differential operator in time t of order $m_d \geq 1$ with values in $\mathbb{R}^N$. Equation (2) is the deterministic initial condition at $t = t_0$. The operator $\boldsymbol{\mathcal{L}}^{\mathrm{IC}}_{t_0}\big(\mathbf{Y}\big)$ is a linear differential operator with values in $\mathbb{R}^{N_0}$ in which $N_0 \geq N$ depends on m_d. The deterministic vector $\mathbf{y}_0$ is given in $\mathbb{R}^{N_0}$. For instance, $\boldsymbol{\mathcal{L}}^{\mathrm{IC}}_{t_0}\big(\mathbf{Y}\big) = \mathbf{Y}(t_0)$ and $N_0 = N$ for $m_d = 1$ (first-order nonlinear differential operator $\boldsymbol{\mathcal{N}}_t$) or $\boldsymbol{\mathcal{L}}^{\mathrm{IC}}_{t_0}\big(\mathbf{Y}\big) = \big(\mathbf{Y}(t_0), \dot{\mathbf{Y}}(t_0)\big)$ with $N_0 = 2N$ for $m_d = 2$ (second-order nonlinear differential operator). In addition, in order to simplify the presentation, we will assume that the first m_d derivatives of $\mathbf{Y}$ with respect to t at initial time $t = t_0$ take deterministic values (this simplified hypothesis can easily be removed without difficulty and does not call into question the presented developments). These operators will be detailed for each of the applications presented in the sequel.

2.2. Vector-valued random control parameter

The random vector $\mathbf{W} = (W_1, \ldots, W_{n_w})$ is defined on a probability space $(\Theta, \mathcal{T}, \mathcal{P})$. Its prior probability distribution $P_{\mathbf{W}}(d\mathbf{w}) = p_{\mathbf{W}}(\mathbf{w})\, d\mathbf{w}$ is defined by a given probability density function $\mathbf{w} \mapsto p_{\mathbf{W}}(\mathbf{w})$ on $\mathbb{R}^{n_w}$ with respect to the Lebesgue measure $d\mathbf{w}$ on $\mathbb{R}^{n_w}$. Its support, $C_w \subset \mathbb{R}^{n_w}$, is the admissible set of $\mathbf{W}$, which will be specified for each of the applications.

2.3. Uniqueness of the stochastic solution of the SCM

Let m_d and $\mathbf{y}_0$ be as defined in Section 2.1. Let $C^{m_d}_{\mathrm{ad}} = \big\{\mathbf{y} \in \mathrm{C}^{m_d}([t_0, T], \mathbb{R}^N)\,;\, \boldsymbol{\mathcal{L}}^{\mathrm{IC}}_{t_0}(\mathbf{y}) = \mathbf{y}_0\big\}$ in which $\mathrm{C}^{m_d}([t_0, T], \mathbb{R}^N)$ is the set of all the m_d-times continuously differentiable functions $t \mapsto \mathbf{y}(t) = (y_1(t), \ldots, y_N(t))$ defined on $[t_0, T]$ with values in $\mathbb{R}^N$. For $m_d = 1$, the first time derivative is denoted by $\dot{\mathbf{y}}(t) = d\mathbf{y}(t)/dt$ and for $m_d = 2$, the second time derivative is denoted by $\ddot{\mathbf{y}}(t) = d^2\mathbf{y}(t)/dt^2$, which are continuous functions. Let $\mathbf{w}^\ell = \mathbf{W}(\theta_\ell) \in C_w \subset \mathbb{R}^{n_w}$ be any realization of $\mathbf{W}$ for $\theta_\ell \in \Theta$. It is assumed that the problem defined by

$$\boldsymbol{\mathcal{N}}_t(\mathbf{y}^\ell(t), t, \mathbf{w}^\ell) = \mathbf{0} \quad , \quad t \in]t_0, T]\,, \tag{3}$$

$$\boldsymbol{\mathcal{L}}^{\mathrm{IC}}_{t_0}(\mathbf{y}^\ell) = \mathbf{y}_0 \quad , \quad t = t_0\,. \tag{4}$$

admits a unique solution $\mathbf{y}^\ell = \big\{t \mapsto \mathbf{y}^\ell(t)\big\}$ from $[t_0, T]$ into $\mathbb{R}^N$, which is m_d-times differentiable in time, that is to say, $(\mathbf{y}^\ell, \mathbf{w}^\ell) \in C^{m_d}_{\mathrm{ad}} \times C_w$. Consistent with this hypothesis, it is assumed that Eqs. (1)-(2) admit a unique stochastic solution $\mathbf{Y}^{\mathrm{sol}} = \big\{\mathbf{Y}^{\mathrm{sol}}(t),\, t \in [t_0, T]\big\}$ that depends on $\mathbf{W}$ and $\mathbf{y}_0$, defined on $(\Theta, \mathcal{T}, \mathcal{P})$, indexed by $[t_0, T]$, with values in $\mathbb{R}^N$, which is assumed to be a second-order stochastic process whose trajectories (sample paths) are thus m_d-times differentiable in time. Consequently, for all θ_ℓ in Θ, $\big(\mathbf{Y}^{\mathrm{sol}}(\cdot\,; \theta_\ell), \mathbf{W}(\theta_\ell)\big) \in C^{m_d}_{\mathrm{ad}} \times C_w$. It should be noted that the probability measure of $(\mathbf{Y}^{\mathrm{sol}}, \mathbf{W})$ (defined for instance by its system of marginal (or cylindrical) distributions) cannot be exactly constructed in the considered general case. We will limit ourselves to building an approximation of this measure using the training dataset and the PLoM without or under constraints.

2.4. Training dataset $\mathcal{Y}_{N_d}^d$

We consider N_d independent realizations $\mathbf{w}^\ell \in C_w$ of $\mathbf{W}$ and the corresponding N_d independent realizations $\mathbf{y}^\ell \in C_{\text{ad}}^{m_d}$ of $\mathbf{Y}^{\text{sol}}$ (see Section 2.3). The training dataset $\mathcal{Y}_{N_d}^d$ is then defined by

$$\mathcal{Y}_{N_d}^d = \left\{(\mathbf{y}^1, \mathbf{w}^1), \ldots, (\mathbf{y}^{N_d}, \mathbf{w}^{N_d})\right\} \subset (C_{\text{ad}}^{m_d} \times C_w)^{N_d} . \tag{5}$$

Thus the construction of the training dataset $\mathcal{Y}_{N_d}^d$ requires solving N_d samples of the SCM defined by Eqs. (3)-(4). As we have explained in Section 1, it is assumed that the numerical cost for one computation is very high and consequently, only a small value of N_d can be considered (small data case).

2.5. Time sampling and time integration scheme

The construction of the training dataset requires the numerical solution of Eqs. (3)-(4). Therefore, we have to introduce a time sampling and a time integration scheme. For all $\ell \in \left\{1, \ldots, N_d\right\}$, the time sampling is performed (thus, is independent from ℓ) with a constant time step Δt and n_{time} sampling times $t_1, \ldots, t_{n_{\text{time}}}$ constituting a finite part $\mathcal{J}$ of $[t_0, T]$, such that

$$t_n = t_0 + n\,\Delta t \ , \ n = 1, \ldots n_{\text{time}} \quad , \quad \mathcal{J} = \left\{t_1, \ldots, t_{n_{\text{time}}}\right\} \subset [t_0, T] , \tag{6}$$

in which Δt and n_{time} verify $T - t_0 = n_{\text{time}}\,\Delta t$. For all ℓ in $\left\{1, \ldots, N_d\right\}$, Eqs. (3)-(4) are solved using a time integration scheme (such as central difference scheme or Newmark scheme, etc), which must be adapted to the type of nonlinear operator $\boldsymbol{\mathcal{N}}_t$, that is to say, which depends on each application (see Sections 10, 11, and 12). We then obtain the time sampling of function $\left\{t \mapsto \mathbf{y}^\ell(t)\right\} \in C_{\text{ad}}^{m_d}$,

$$\left\{\mathbf{y}^\ell(t), t \in \mathcal{J}\right\} = \left\{\mathbf{y}^\ell(t_1), \ldots, \mathbf{y}^\ell(t_{n_{\text{time}}})\right\} \quad , \quad \mathbf{y}^\ell(t_n) \in \mathbb{R}^N . \tag{7}$$

Using the time integration scheme, the time derivative up to order m_d of $\mathbf{y}^\ell$ can be computed at all sampling times belonging to $\mathcal{J}$. It should be noted that for each pair $(\mathbf{y}^\ell, \mathbf{w}^\ell)$ in $\mathcal{Y}_{N_d}^d$, function $\mathbf{y}^\ell$ is only known at the sampling times of $\mathcal{J}$.

2.6. Definition of the stochastic process $\mathbf{Y}$ and its second-order moments

Let us consider the couple $(\mathbf{Y}, \mathbf{W})$ in which $\mathbf{Y} = \left\{\mathbf{Y}(t), t \in [t_0, T]\right\}$ is a mean-square continuous second-order stochastic process, defined on $(\Theta, \mathcal{T}, \mathcal{P})$, indexed by $[t_0, T]$, with values in $\mathbb{R}^N$, whose trajectories are assumed to be m_d-times differentiable in time, and where the N_d independent realizations of $(\mathbf{Y}, \mathbf{W})$ are those defined in Eq. (5). Therefore, $(\mathbf{Y}, \mathbf{W})$ is an approximation of $(\mathbf{Y}^{\text{sol}}, \mathbf{W})$ which depends on N_d. Let $t \mapsto \underline{\mathbf{y}}(t) = E\left\{\mathbf{Y}(t)\right\}$ be the mean function of $\mathbf{Y}$, which is a continuous function from $[t_0, T]$ into $\mathbb{R}^N$ and let $(t, t') \mapsto [C_{\mathbf{Y}}(t, t')] = E\left\{\left(\mathbf{Y}(t) - \underline{\mathbf{y}}(t)\right)\left(\mathbf{Y}(t') - \underline{\mathbf{y}}(t')\right)^T\right\}$ be its matrix-valued covariance function that is a continuous function from $[t_0, T] \times [t_0, T]$ into $\mathbb{M}_N$. These two functions are estimated at the sampling times as follows: for n and n' in $\left\{1, \ldots, n_{\text{time}}\right\}$,

$$\underline{\mathbf{y}}(t_n) = \frac{1}{N_d} \sum_{\ell=1}^{N_d} \mathbf{y}^\ell(t_n) \quad , \quad [C_{\mathbf{Y}}(t_n, t_{n'})] = \frac{1}{N_d - 1} \sum_{\ell=1}^{N_d} \mathbf{y}_c^\ell(t_n) \mathbf{y}_c^\ell(t_{n'})^T \tag{8}$$

in which, for $\ell \in \left\{1, \ldots, N_d\right\}$, $\mathbf{y}_c^\ell$ is the centered realization such that

$$\mathbf{y}_c^\ell(t_n) = \mathbf{y}^\ell(t_n) - \underline{\mathbf{y}}(t_n) \quad , \quad n \in \left\{1, \ldots, n_{\text{time}}\right\} . \tag{9}$$

3. Reduced-order representation of the nonstationary stochastic process Y

The construction of the reduced-order representation of stochastic process $\mathbf{Y} = \left\{\mathbf{Y}(t), t \in [t_0, T]\right\}$ is based on its classical Karhunen-Loève expansion that will effectively be calculated at sampling times $t_0, t_1, \ldots, t_{n_{\text{time}}}$.

3.1. Eigenvalue problem for the Karhunen-Loève (KL) expansion of stochastic process $\{\mathbf{Y}(t), t \in [t_0, T]\}$

The KL expansion of stochastic process $\left\{\mathbf{Y}(t), t \in [t_0, T]\right\}$ is based on the use of the spectral analysis of its covariance operator. Let $\mathbb{H} = L^2_{\mu_0}([t_0, T], \mathbb{R}^N)$ be the Hilbert space of all the square integrable functions $t \mapsto \mathbf{y}(t)$ from $[t_0, T]$ into $\mathbb{R}^N$, equipped with the inner product $< \mathbf{y}, \mathbf{y}' >_{\mathbb{H}}$ and its associated norm $\| \mathbf{y} \|_{\mathbb{H}} =< \mathbf{y}, \mathbf{y} >^{1/2}_{\mathbb{H}}$ such that $< \mathbf{y}, \mathbf{y}' >_{\mathbb{H}} = \int_{t_0}^{T} < \mathbf{y}(t), \mathbf{y}'(t) > \mu_0(dt)$, in which the measure $\mu_0(dt)$ on $\mathbb{R}$ is defined by $\mu_0(dt) = (T - t_0)^{-1} dt$ with dt the Lebesgue measure on $\mathbb{R}$. Since the covariance function $(t, t') \mapsto [C_{\mathbf{Y}}(t, t')]$ is continuous on the compact set $[t_0, T] \times [t_0, T]$, it can be seen that $\int_{t_0}^{T} \int_{t_0}^{T} \| [C_{\mathbf{Y}}(t, t')] \|^2_F \, \mu_0(dt) \, \mu_0(dt') < +\infty$ in which $\| \cdot \|_F$ is the Froebenius norm. Therefore, the covariance linear operator $\mathbf{C_Y}$ such that $(\mathbf{C_Y y})(t) = \int_{t_0}^{T} [C_{\mathbf{Y}}(t, t')] \, \mathbf{y}(t') \, \mu_0(dt')$ belongs to $\mathcal{L}_{\mathrm{HS}}(\mathbb{H})$ (space of the Hilbert-Schmidt operators in $\mathbb{H}$) [48]. Consequently, the eigenvalue problem $\mathbf{C_Y} \widetilde{\boldsymbol{\varphi}} = \widetilde{\Lambda} \, \widetilde{\boldsymbol{\varphi}}$ has a countable spectrum $\widetilde{\Lambda}_1 \geq \widetilde{\Lambda}_2 \geq \ldots$ such that $\sum_\alpha \widetilde{\Lambda}^2_\alpha < +\infty$.

3.2. Reduced-order representation of discrete stochastic process $\{\mathbf{Y}(t), t \in \mathcal{J}\}$

Instead of discretizing the continuous eigenvalue problem $\mathbf{C_Y} \widetilde{\boldsymbol{\varphi}} = \widetilde{\Lambda} \, \widetilde{\boldsymbol{\varphi}}$, we choose to approximate it by the eigenvalue problem related to the covariance matrix of the random vector $\mathbf{Y}_{\mathcal{J}} = \big(\mathbf{Y}(t_1), \ldots, \mathbf{Y}(t_{n_{\mathrm{time}}})\big)$ with values in $\mathbb{R}^{n_{\mathrm{time}} N}$, which corresponds to the time sampling of stochastic process $\left\{\mathbf{Y}(t), t \in [t_0, T]\right\}$. This choice is justified by Section 3.1 (the spectrum of $\mathbf{C_Y}$ is countable) and by the fact that for applications devoted to nonlinear dynamical system, number n_{time} of sampling times is generally very high, and consequently, the constructed approximation is excellent. Consequently, the KL expansion of stochastic process $\mathbf{Y}$ is replaced by the Principal Component Analysis (PCA) of random vector $\mathbf{Y}_{\mathcal{J}}$.

(i) Eigenvalue problem for the PCA expansion. The eigenvalue problem related to the covariance matrix of random vector $\mathbf{Y}_{\mathcal{J}}$ is written, using Eq. (8), as

$$\frac{1}{n_{\mathrm{time}}} \sum_{n'=1}^{n_{\mathrm{time}}} [C_{\mathbf{Y}}(t_n, t_{n'})] \, \boldsymbol{\varphi}^\alpha(t_{n'}) = \Lambda_\alpha \, \boldsymbol{\varphi}^\alpha(t_n) \quad , \quad n \in \left\{1, \ldots, n_{\mathrm{time}}\right\}. \tag{10}$$

The eigenvalues are ordered as $\Lambda_1 \geq \Lambda_2 \geq \ldots \geq 0$ and the family of eigenvectors $\left\{\boldsymbol{\varphi}^\alpha(t_1), \ldots, \boldsymbol{\varphi}^\alpha(t_{n_{\mathrm{time}}})\right\}_\alpha$ in $\mathbb{R}^{n_{\mathrm{time}} N}$ satisfies the orthogonality property, $(1/n_{\mathrm{time}}) \sum_{n=1}^{n_{\mathrm{time}}} <\boldsymbol{\varphi}^\alpha(t_n), \boldsymbol{\varphi}^\beta(t_n)>= \delta_{\alpha\beta}$.

(ii) Reduced-order representation of discrete stochastic process $\left\{\mathbf{Y}(t), t \in \mathcal{J}\right\}$. The reduced representation $\left\{\mathbf{Y}^{(n_q)}(t), t \in \mathcal{J}\right\}$ of $\left\{\mathbf{Y}(t), t \in \mathcal{J}\right\}$ of order n_q is constructed using the eigenvectors associated with the first $n_q \ll n_{\mathrm{time}} N$ strictly positive eigenvalues, and is written as

$$\mathbf{Y}^{(n_q)}(t_n) = \underline{\mathbf{y}}(t_n) + [V(t_n)] \, \mathbf{Q} \quad , \quad n \in \left\{1, \ldots, n_{\mathrm{time}}\right\}, \tag{11}$$

in which $\underline{\mathbf{y}}(t_n) \in \mathbb{R}^N$ is defined by Eq. (8) and where $[V(t_n)]$ is the matrix defined by

$$[V(t_n)] = [\Lambda_1^{1/2} \, \boldsymbol{\varphi}^1(t_n) \ldots \Lambda_{n_q}^{1/2} \, \boldsymbol{\varphi}^{n_q}(t_n)] \in \mathbb{M}_{N, n_q}. \tag{12}$$

In Eq. (11), $\mathbf{Q} = (Q_1, \ldots, Q_{n_q})$ is a second-order centered and orthogonal random variable defined on probability space $(\Theta, \mathcal{T}, \mathcal{P})$ with values in $\mathbb{R}^{n_q}$, which is written as

$$\mathbf{Q} = [\Lambda]^{-1} \frac{1}{n_{\mathrm{time}}} \sum_{n=1}^{n_{\mathrm{time}}} [V(t_n)]^T \left(\mathbf{Y}(t_n) - \underline{\mathbf{y}}(t_n)\right), \tag{13}$$

in which $[\Lambda]$ is the positive-definite diagonal $(n_q \times n_q)$ matrix, $[\Lambda]_{\alpha\beta} = \Lambda_\alpha \, \delta_{\alpha\beta}$. The N_d independent realizations $\left\{\mathbf{q}^\ell\right\}_\ell$

in $\mathbb{R}^{n_q}$ of $\mathbf{Q}$, associated with the N_d points of the training dataset $\mathcal{Y}_{N_d}^d$ defined by Eq. (5), are computed by

$$\mathbf{q}^\ell = [\Lambda]^{-1} \frac{1}{n_{\text{time}}} \sum_{n=1}^{n_{\text{time}}} [V(t_n)]^T \left(\mathbf{y}^\ell(t_n) - \underline{\mathbf{y}}(t_n) \right) . \tag{14}$$

The empirical estimates of the mean vector $\underline{\mathbf{q}} \in \mathbb{R}^{n_q}$ and the covariance matrix $[C_{\mathbf{Q}}] \in \mathbb{M}_{n_q}^+$ of random vector $\mathbf{Q}$ are such that $\underline{\mathbf{q}} = (1/N_d) \sum_{\ell=1}^{N_d} \mathbf{q}^\ell = \mathbf{0}_{n_q}$ and $[C_{\mathbf{Q}}] = \big(1/(N_d - 1)\big) \sum_{\ell=1}^{N_d} \mathbf{q}^\ell (\mathbf{q}^\ell)^T = [I_{n_q}]$. Using a time integration scheme and the initial condition at t_0 yields the m_d time derivatives of $\underline{\mathbf{y}}(t)$ and $[V(t)]$ at $t \in \{t_0, t_1, \ldots, t_{n_{\text{time}}}\}$ and consequently, allows for calculating the m_d time derivatives of $\mathbf{Y}^{(n_q)}(t_n)$ for $n \in \{0, 1, \ldots, n_{\text{time}}\}$. For this type of calculation, an adapted methodology must be developed for each application because N and n_{time} can be very large, inducing possible difficulties due to Random Access Memory (RAM) limitations.

(iii) Calculation of the reduction order n_q. Let $n_q \mapsto \text{err}_{\text{KL}}(n_q)$ be the error function related to the relative mean-square error between $\{\mathbf{Y}(t), t \in \mathcal{J}\}$ and $\{\mathbf{Y}^{(n_q)}(t), t \in \mathcal{J}\}$, defined by

$$\text{err}_{\text{KL}}(n_q) = 1 - \frac{\sum_{\alpha=1}^{n_q} \Lambda_\alpha}{(1/n_{\text{time}}) \sum_{n=1}^{n_{\text{time}}} \text{tr}\, [C_{\mathbf{Y}}(t_n, t_n)]} . \tag{15}$$

The value of $n_q \ll n_{\text{time}} N$ is calculated such that $\Lambda_{n_q} > 0$ and $\text{err}_{\text{KL}}(n_q) \leq \varepsilon_{\text{KL}}$ where ε_{KL} is a given small positive real number used for error control.

(iv) Algorithm for solving the eigenvalue problem. For $\ell \in \{1, \ldots, N_d\}$ and for $\alpha \in \{1, \ldots, n_q\}$, we introduce the vectors $\mathbf{y}_{\mathcal{J}}^\ell = \big(\mathbf{y}_c^\ell(t_1), \ldots, \mathbf{y}_c^\ell(t_{n_{\text{time}}})\big)$ and $\boldsymbol{\varphi}_{\mathcal{J}}^\alpha = \big(\boldsymbol{\varphi}^\alpha(t_1), \ldots, \boldsymbol{\varphi}^\alpha(t_{n_{\text{time}}})\big)$ that belong to $\mathbb{R}^{n_{\text{time}} N}$, in which $\mathbf{y}_c^\ell(t_n)$ is defined by Eq. (9) and $\boldsymbol{\varphi}^\alpha(t_n)$ by Eq. (10). To these vectors, we associate the matrices $[y_{\mathcal{J}}] \in \mathbb{M}_{n_{\text{time}} N, N_d}$ and $[\varphi_{\mathcal{J}}] \in \mathbb{M}_{n_{\text{time}} N, n_q}$ such that $[y_{\mathcal{J}}] = [\mathbf{y}_{\mathcal{J}}^1 \ldots \mathbf{y}_{\mathcal{J}}^{N_d}]$ and $[\varphi_{\mathcal{J}}] = [\boldsymbol{\varphi}_{\mathcal{J}}^1 \ldots \boldsymbol{\varphi}_{\mathcal{J}}^{n_q}]$. Using these notations, the eigenvalue problem defined by Eq. (10) can be rewritten as $[C_{\mathcal{J}}]\, \boldsymbol{\varphi}_{\mathcal{J}}^\alpha = \Lambda_\alpha \boldsymbol{\varphi}_{\mathcal{J}}^\alpha$ in which $[C_{\mathcal{J}}]$ is a symmetric real $(n_{\text{time}} N \times n_{\text{time}} N)$ matrix whose rank is $n_q \leq N_d$, and which is written as $[C_{\mathcal{J}}] = \big(1/(N_d - 1)\big) (n_{\text{time}}^{-1/2} [y_{\mathcal{J}}]) (n_{\text{time}}^{-1/2} [y_{\mathcal{J}}])^T$. In general, for the considered nonlinear SCM, N is large and the number n_{time} of sampling times is also large. Consequently, the dimension $(n_{\text{time}} N \times n_{\text{time}} N)$ of matrix $[C_{\mathcal{J}}]$ of this eigenvalue problem is a full matrix that cannot be assembled (the number of entries is, for a medium problem, of order 10^{18}). In the framework of the present work, N_d is small (small size of the training dataset, for instance, of order 100). Therefore, $n_{\text{time}} N \gg N_d$, which allows for solving the eigenvalue problem, without assembling $[C_{\mathcal{J}}]$, by using a "thin SVD" [49] applied to rectangular matrix $n_{\text{time}}^{-1/2} [y_{\mathcal{J}}]$. Keeping only the n_q largest singular values $s_\alpha = [s]_{\alpha\alpha}$ ordered as $s_1 \geq s_2 \geq \ldots \geq s_{n_q} > 0$, one obtains the following factorization, $n_{\text{time}}^{-1/2} [y_{\mathcal{J}}] = [b_{\mathcal{J}}]\, [s]\, [v]$ in which $[v] \in \mathbb{M}_{n_q}$ verifies $[v]\, [v]^T = [I_{n_q}]$ and where $[b_{\mathcal{J}}] \in \mathbb{M}_{n_{\text{time}} N, n_q}$ is such that $[b_{\mathcal{J}}]^T [b_{\mathcal{J}}] = [I_{n_q}]$. Finally, eigenvalues $\Lambda_1, \ldots, \Lambda_{n_q}$ and matrix $[\varphi_{\mathcal{J}}] \in \mathbb{M}_{n_{\text{time}} N, n_q}$ are given by $\Lambda_\alpha = s_\alpha^2/(N_d - 1)$ and $[\varphi_{\mathcal{J}}] = n_{\text{time}}^{1/2} [b_{\mathcal{J}}]$.

4. Random residual associated with the nonlinear SCM and time subsampling

In this section, we consider any mean-square continuous second-order stochastic process $\{\widehat{\mathbf{Y}}(t), t \in [t_0, T]\}$ with values in $\mathbb{R}^N$, defined on $(\Theta, \mathcal{T}, \mathcal{P})$, whose trajectories belong to $C_{\text{ad}}^{m_d}$, and which is statistically dependent on a random vector $\widehat{\mathbf{W}}$ defined on $(\Theta, \mathcal{T}, \mathcal{P})$, with values in $\mathcal{C}_w \subset \mathbb{R}^{n_w}$. It is assumed that n_r independent realizations $(\widehat{\mathbf{y}}^\ell, \widehat{\mathbf{w}}^\ell)$ of $(\widehat{\mathbf{Y}}, \widehat{\mathbf{W}})$ are known, that is to say, for all $\ell \in \{1, \ldots, n_r\}$ and $\theta_\ell \in \Theta$,

$$(\widehat{\mathbf{y}}^\ell, \widehat{\mathbf{w}}^\ell) \in C_{\text{ad}}^{m_d} \times \mathcal{C}_w \ , \ \widehat{\mathbf{y}}^\ell = \widehat{\mathbf{Y}}(\cdot; \theta_\ell) \ , \ \widehat{\mathbf{w}}^\ell = \widehat{\mathbf{W}}(\theta_\ell) . \tag{16}$$

As previously, $\{\widehat{\mathbf{Y}}(t), t \in \mathcal{J}\}$ and $\{\widehat{\mathbf{y}}^\ell(t), t \in \mathcal{J}\}$ will denote the time sampling of $\widehat{\mathbf{Y}}$ and $\widehat{\mathbf{y}}^\ell$. In this section, we will formulate the residual calculation for the couple $(\widehat{\mathbf{Y}}, \widehat{\mathbf{W}})$. We will use it for computing the random residual when $(\widehat{\mathbf{Y}}, \widehat{\mathbf{W}})$ will represent either $(\mathbf{Y}, \mathbf{W})$ with $n_r = N_d$, $\mathbf{W}$ defined in Section 2.2, and $\mathbf{Y}$ defined in Section 2.6 (the reference) or will represent $(\mathbf{Y}^{(n_q)}, \mathbf{W})$ with $n_r = n_{\text{MC}}$, $\mathbf{Y}^{(n_q)}$ defined by Eq. (11), and for which the probability measure of random vector $\mathbf{X} = (\mathbf{Q}, \mathbf{W})$ will be estimated either with the PLoM without constraint or with the PLoM under constraints.

4.1. Random residual

(i) Definition of the time dependent vector-valued random residual $\boldsymbol{\mathcal{R}}(t)$. For all t fixed in $[t_0, T]$, the time dependent $\mathbb{R}^N$-valued residual $\boldsymbol{\mathcal{R}}(t) = (\mathcal{R}_1(t), \ldots, \mathcal{R}_N(t))$, associated with the couple $(\widehat{\mathbf{Y}}, \widehat{\mathbf{W}})$, is defined by

$$\boldsymbol{\mathcal{R}}(t) = \boldsymbol{\mathcal{N}}_t\big(\widehat{\mathbf{Y}}(t), t, \widehat{\mathbf{W}}\big) \quad , \quad t \in [t_0, T]\,. \tag{17}$$

It is assumed that $\big\{\boldsymbol{\mathcal{R}}(t), t \in [t_0, T]\big\}$ is a mean-square continuous second-order $\mathbb{R}^N$-valued stochastic process defined on $(\Theta, \mathcal{T}, \mathcal{P})$, whose trajectories are almost surely continuous. For $\ell \in \big\{1, \ldots, n_r\big\}$, and using Eqs. (16) and (17), the realization $\mathbf{r}^\ell(t)$ of $\boldsymbol{\mathcal{R}}(t)$ is written as

$$\mathbf{r}^\ell(t) = \boldsymbol{\mathcal{N}}_t(\widehat{\mathbf{y}}^\ell(t), t, \widehat{\mathbf{w}}^\ell)\,. \tag{18}$$

The $\mathbb{R}^N$-valued function $t \mapsto \mathbf{r}^\ell(t)$ is continuous on $[t_0, T]$ and is evaluated at the sampling times of $\mathcal{J}$. It should be noted that the approximations of the time derivatives involved in the differential operator for the calculation of the residual are exactly the same as those used for the calculation of each realization of the random solution of the stochastic boundary problem, using the stochastic computational model (see Sections 11 and 12 devoted to Applications 2 and 3).

(ii) Definition of the time dependent real-valued random residual $\mathcal{R}_{\text{norm}}(t)$. For all t fixed in $[t_0, T]$, the time dependent real-valued random residual $\mathcal{R}_{\text{norm}}(t)$ is the positive-valued random variable is defined by

$$\mathcal{R}_{\text{norm}}(t) = \frac{1}{\sqrt{N}} \| \boldsymbol{\mathcal{R}}(t) \|\,, \tag{19}$$

in which $\| \cdot \|$ is the usual Euclidean norm.

(iii) Definition of the random residual $\widehat{\rho}$. The *random residual* associated with the couple $(\widehat{\mathbf{Y}}, \widehat{\mathbf{W}})$ is then defined as follows,

$$\widehat{\rho} = \| \mathcal{R}_{\text{norm}} \|_\mu \quad , \quad \| \mathcal{R}_{\text{norm}} \|_\mu^2 = \frac{1}{T - t_0} \int_{t_0}^T \mathcal{R}_{\text{norm}}(t)^2 \, d\mu(t)\,, \tag{20}$$

in which $d\mu$ is a given bounded positive measure on $\mathbb{R}$, which will be defined in Section 4.2 when introducing the time subsampling for computing the residual. From Eqs. (19) and (20), it can be seen that

$$\widehat{\rho}^2 = \frac{1}{N(T - t_0)} \int_{t_0}^T \| \boldsymbol{\mathcal{R}}(t) \|^2 \, d\mu(t)\,. \tag{21}$$

It should be noted that, if $d\mu(t)$ was chosen as dt on bounded interval $[t_0, T]$, then for $\ell \in \big\{1, \ldots, n_r\big\}$, the realization $\widehat{\rho}^\ell$ of random variable $\widehat{\rho}$ could only be computed using the time sampling $\mathcal{J}$, that is to say, by the formula $(\widehat{\rho}^\ell)^2 = (N\, n_{\text{time}})^{-1} \sum_{n=1}^{n_{\text{time}}} \| \mathbf{r}^\ell(t_n) \|^2$.

(iv) Definition of the L^2*-norm of random residual* $\widehat{\rho}$. Since $\boldsymbol{\mathcal{R}}$ is mean-square continuous on compact set $[t_0, T]$, it can be deduced [50, 51] that $\widehat{\rho}$ is a real-valued (in fact a positive-valued) second-order random variable. Consequently, $\widehat{\rho}$ belongs to $L^2(\Theta, \mathbb{R})$ and its norm is $\|\widehat{\rho}\|_{L^2} = \sqrt{E\big\{\widehat{\rho}^2\big\}}$. If $(\widehat{\mathbf{Y}}, \widehat{\mathbf{W}})$ is the couple $(\mathbf{Y}^{\text{sol}}, \mathbf{W})$ defined in Section 2.3, then $\|\widehat{\rho}\|_{L^2} = 0$. Using the n_r independent realizations $\widehat{\rho}^\ell$ of $\widehat{\rho}$, the empirical estimate of $\|\widehat{\rho}\|_{L^2}$ is computed as $\Big((1/n_r) \sum_{\ell=1}^{n_r} (\widehat{\rho}^\ell)^2\Big)^{1/2}$.

(v) Definition of the normalized random residual ρ. Let $\widehat{\rho}_{\text{ref}}$ be the random residual $\widehat{\rho}$ calculated for a fixed value $N_{d,\text{ref}}$ of the number N_d of points in the training dataset (thus $n_r = N_{d,\text{ref}}$ and $(\widehat{\mathbf{Y}}, \widehat{\mathbf{W}})$ is the couple $(\mathbf{Y}, \mathbf{W})$ defined in Section 2.6 with $N_d = N_{d,\text{ref}}$). Let $\underline{\widehat{\rho}}_{\text{ref}}$ be the empirical mean value of $\widehat{\rho}_{\text{ref}}$ estimated with the $N_{d,\text{ref}}$ realizations of $\widehat{\rho}_{\text{ref}}$, computed with the KL expansion $\big\{\mathbf{Y}^{(n_q)}(t), t \in [t_0, T]\big\}$ of $\big\{\mathbf{Y}(t), t \in [t_0, T]\big\}$, that is to say, $(\widehat{\mathbf{Y}}, \widehat{\mathbf{W}})$ is the couple

$(\mathbf{Y}^{(n_q)}, \mathbf{W})$. The positive-valued normalized random residual ρ is then defined by

$$\rho = \frac{\widehat{\rho}}{\widehat{\rho}_{\text{ref}}} , \tag{22}$$

in which $\widehat{\rho}$ is the random residual defined by Eq. (20). Consequently, if $n_r = N_{d,\text{ref}}$ and $(\widehat{\mathbf{Y}}, \widehat{\mathbf{W}})$ is the couple $(\mathbf{Y}, \mathbf{W})$ defined in Section 2.6 with $N_d = N_{d,\text{ref}}$, then the empirical mean value of ρ is equal to 1 (because, in this case $\widehat{\rho}$ is $\widehat{\rho}_{\text{ref}}$).

4.2. Time subsampling for computing the realizations of the random residual

The role played by measure $d\mu$ introduced in Eq. (20) is the following. The computation of the n_r realizations $\{\mathbf{r}^\ell(t), t \in \mathcal{J}\}_\ell$ of $\{\mathcal{R}(t), t \in \mathcal{J}\}$ using Eq. (18) requires $n_r \times n_{\text{time}}$ numerical evaluations of the nonlinear operator $\mathcal{N}_t$ and requires a RAM of $n_r \times n_{\text{time}} \times N$ 64-bit words. The values of N, n_{time}, and n_r can be large. In general the CPU time is not a limitation because a parallel computation can be done for the loop on the number n_r of realizations, but difficulties can occur with possible RAM limitations. On the other hand, such a large value of n_{time} is not necessary for constructing the constraint on ρ and a reasonable approximation of the integral on $[t_0, T]$ in the Eq. (21) can be used to compute the n_r realizations $\{\widehat{\rho}^\ell\}_\ell$ of random residual $\widehat{\rho}$. For such a computation, we introduce the following subsampling times,

$$\{\tau_{n_s}, n_s = 1, \ldots, n_{\text{sp}}\} \subset \{t_n, n = 1, \ldots, n_{\text{time}}\} \quad , \quad n_{\text{sp}} \ll n_{\text{time}} , \tag{23}$$

in which $\tau_1, \ldots, \tau_{n_{\text{sp}}}$ are defined as the subsampling times. The measure $d\mu$ is then defined by $d\mu(t) = \big((T - t_0)/n_{\text{sp}}\big) \sum_{n_s=1}^{n_{\text{sp}}} \delta_0(t - \tau_{n_s})$, in which δ_0 is the Dirac measure on $\mathbb{R}$ at 0. With this approximation, there is a big gain with respect to the RAM limitations and also a gain for the CPU time. Using this measure, Eq. (21) yields

$$\widehat{\rho}^2 = \frac{1}{N n_{\text{sp}}} \sum_{n_s=1}^{n_{\text{sp}}} \| \mathcal{R}(\tau_{n_s}) \|^2 . \tag{24}$$

The n_r independent realizations $\{\widehat{\rho}^\ell\}_\ell$ of random residual $\widehat{\rho}$ are such that

$$(\widehat{\rho}^\ell)^2 = \frac{1}{N n_{\text{sp}}} \sum_{n_s=1}^{n_{\text{sp}}} \| \mathbf{r}^\ell(\tau_{n_s}) \|^2 \quad , \quad \ell \in \{1, \ldots, n_r\} , \tag{25}$$

in which $\mathbf{r}^\ell(\tau_{n_s}) \in \mathbb{R}^N$ is computed with Eq. (18) for $t = \tau_{n_s}$. The time subsampling has to be defined for each application.

5. Definition of random vectors X and H

The formulation of PLoM under constraints for stochastic processes requires the definition of the initial dataset $\mathcal{D}^d_{N_d}$ of the random vector $\mathbf{X} = (\mathbf{Q}, \mathbf{W})$ and its PCA constructed using $\mathcal{D}^d_{N_d}$.

(i) Definition of the initial dataset $\mathcal{D}^d_{N_d}$. For the probabilistic learning, we introduce the initial dataset $\mathcal{D}^d_{N_d}$ that is associated with the training dataset $\mathcal{Y}^d_{N_d} = \{(\mathbf{y}^1, \mathbf{w}^1), \ldots, (\mathbf{y}^{N_d}, \mathbf{w}^{N_d})\}$ defined by Eq. (5) and with the reduced-order representation defined in Section 3.2-(ii). Each realization $\mathbf{y}^\ell$ of $\mathbf{Y}$ is associated with the realization $\mathbf{w}^\ell$ of $\mathbf{W}$ (see Section 2.3). The reduced-order representation $\mathbf{Y}^{(n_q)}$, defined by Eq. (11), allows the N_d independent realizations $\{\mathbf{q}^\ell, \ell = 1, \ldots, N_d\}$ of the $\mathbb{R}^{n_q}$-valued random variable $\mathbf{Q}$ to be computed by using Eq. (14). Let $\mathbf{X} = (\mathbf{Q}, \mathbf{W})$ be the random variable defined on $(\Theta, \mathcal{T}, \mathcal{P})$ with values in $\mathbb{R}^{n_x} = \mathbb{R}^{n_q} \times \mathbb{R}^{n_w}$. Each realization $\mathbf{q}^\ell$ depends on realization $\mathbf{w}^\ell$, and $\mathbf{x}^\ell = (\mathbf{q}^\ell, \mathbf{w}^\ell)$ is a realization of $\mathbf{X}$. The initial dataset $\mathcal{D}^d_{N_d}$ is made up of the N_d independent realizations $\mathbf{x}^\ell$,

$$\mathcal{D}^d_{N_d} = \{\mathbf{x}^\ell = (\mathbf{q}^\ell, \mathbf{w}^\ell) \in \mathbb{R}^{n_x} = \mathbb{R}^{n_q} \times \mathbb{R}^{n_w} \ , \ \ell = 1, \ldots, N_d\} , \tag{26}$$

(ii) PCA of random vector $\mathbf{X}$. The first step consists of performing a PCA of $\mathbf{X}$. This step is mandatory to formulate the constraints and to extend the PLoM under constraints. It should be noted that the reduced-order representation of

stochastic process $\mathbf{Y}$ has already introduced the random vector $\mathbf{Q}$, which is centered and whose covariance matrix is the identity matrix. However, random vector $\mathbf{X}$ is not centered and its covariance matrix is not the identity matrix. The purpose of the PCA is to improve the statistical computation of $\mathbf{X}$ through de-correlation and normalization. Let $\underline{\mathbf{x}} \in \mathbb{R}^{n_x}$ and $[C_{\mathbf{X}}] \in \mathbb{M}_{n_x}^{+0}$ be the empirical estimates of the mean vector and the covariance matrix of $\mathbf{X}$ such that $\underline{\mathbf{x}} = (1/N_d) \sum_{\ell=1}^{N_d} \mathbf{x}^\ell$ and $[C_{\mathbf{X}}] = \big(1/(N_d-1)\big) \sum_{\ell=1}^{N_d} (\mathbf{x}^\ell - \underline{\mathbf{x}}) (\mathbf{x}^\ell - \underline{\mathbf{x}})^T$. Let $\xi_1 \geq \xi_2 \geq \ldots \geq \xi_{n_x} \geq 0$ be the eigenvalues and let $\boldsymbol{\psi}^1, \ldots, \boldsymbol{\psi}^{n_x}$ be the orthonormal eigenvectors of the eigenvalue problem $[C_{\mathbf{X}}] \boldsymbol{\psi}^\alpha = \xi_\alpha \boldsymbol{\psi}^\alpha$. For $\varepsilon_{\text{PCA}} > 0$ fixed, let ν be the integer such that $n_q + 1 \leq \nu \leq n_x$ and $xi_\nu > 0$, verifying,

$$\text{err}_{\text{PCA}}(\nu) = 1 - \frac{\sum_{\alpha=1}^{\nu} \xi_\alpha}{\text{tr}\,[C_{\mathbf{X}}]} \leq \varepsilon_{\text{PCA}} \,. \tag{27}$$

The PCA of $\mathbf{X}$ yields its representation $\mathbf{X}^{(\nu)}$ that is written as

$$\mathbf{X}^{(\nu)} = \underline{\mathbf{x}} + [\psi]\,[\xi]^{1/2}\,\mathbf{H} \quad , \quad E\big\{\|\,\mathbf{X} - \mathbf{X}^{(\nu)}\,\|^2\big\} \leq \varepsilon_{\text{PCA}}\, E\big\{\|\,\mathbf{X}\,\|^2\big\}\,, \tag{28}$$

in which $[\psi] = [\boldsymbol{\psi}^1 \ldots \boldsymbol{\psi}^\nu] \in \mathbb{M}_{n_x,\nu}$ such that $[\psi]^T [\psi] = [I_\nu]$ and where $[\xi]$ is the positive-definite diagonal $(\nu \times \nu)$ matrix such that $[\xi]_{\alpha\beta} = \xi_\alpha \delta_{\alpha\beta}$. The $\mathbb{R}^\nu$-valued random variable $\mathbf{H}$ is obtained by projection,

$$\mathbf{H} = [\xi]^{-1/2}\,[\psi]^T\,(\mathbf{X} - \underline{\mathbf{x}})\,, \tag{29}$$

and its N_d independent realizations are such that

$$\boldsymbol{\eta}_d^\ell = [\xi]^{-1/2}\,[\psi]^T\,(\mathbf{x}^\ell - \underline{\mathbf{x}}) \in \mathbb{R}^\nu \quad , \quad \ell \in \big\{1, \ldots, N_d\big\}\,. \tag{30}$$

Using $\big\{\boldsymbol{\eta}_d^1, \ldots, \boldsymbol{\eta}_d^{N_d}\big\}$, the estimates of the mean vector $\underline{\boldsymbol{\eta}}$ and the covariance matrix $[C_{\mathbf{H}}]$ of random vector $\mathbf{H}$ are such that $\underline{\boldsymbol{\eta}} = \mathbf{0}_\nu$ and $[C_{\mathbf{H}}] = [I_\nu]$.

(iii) Probability density function of $\mathbf{H}$ *estimated with the initial dataset.* Let $P_{\mathbf{H}}(d\boldsymbol{\eta}) = p_{\mathbf{H}}(\boldsymbol{\eta})\,d\boldsymbol{\eta}$ be the probability distribution of the $\mathbb{R}^\nu$-valued random variable $\mathbf{H}$ defined by Eq. (29). The pdf $\boldsymbol{\eta} \mapsto p_{\mathbf{H}}(\boldsymbol{\eta})$ on $\mathbb{R}^\nu$ with respect to $d\boldsymbol{\eta}$ is estimated by using the Gaussian kernel-density estimation method (KDE) with the N_d independent realizations $\big\{\boldsymbol{\eta}_d^\ell, \ell = 1, \ldots, N_d\big\}$ defined by Eq. (30). Using the modification proposed in [52] of the classical formulation [53], $p_{\mathbf{H}}(\boldsymbol{\eta})$ is estimated, for all $\boldsymbol{\eta}$ in $\mathbb{R}^\nu$, as

$$p_{\mathbf{H}}(\boldsymbol{\eta}) \simeq c_\nu\, \zeta(\boldsymbol{\eta}) \quad , \quad c_\nu = \frac{1}{(\sqrt{2\pi}\,\widehat{s}_\nu)^\nu}\,, \tag{31}$$

$$\zeta(\boldsymbol{\eta}) = \frac{1}{N_d} \sum_{\ell=1}^{N_d} exp\big\{ -\frac{1}{2\widehat{s}_\nu^2} \|\, \frac{\widehat{s}_\nu}{s_\nu} \boldsymbol{\eta}_d^\ell - \boldsymbol{\eta}\,\|^2 \big\}\,, \tag{32}$$

in which $s_\nu = \big\{4/\big(N_d(2+\nu)\big)\big\}^{1/(\nu+4)}$ is the usual multidimensional optimal Silverman bandwidth (taking into account that the covariance matrix of $\mathbf{H}$ is $[I_\nu]$), and where $\widehat{s}_\nu = s_\nu \big(s_\nu^2 + (N_d-1)/N_d\big)^{-1/2}$ has been introduced to ensure that $E\big\{\mathbf{H}\big\} = (\widehat{s}_\nu/s_\nu)\,\underline{\boldsymbol{\eta}} = \mathbf{0}_\nu$ and $E\big\{\mathbf{H}\mathbf{H}^T\big\} = \widehat{s}_\nu^2\,[I_\nu] + \widehat{s}_\nu^2 s_\nu^{-2}(N_d-1)N_d^{-1}\,[C_{\mathbf{H}}] = [I_\nu]$. The pdf of $\mathbf{H}$, defined by Eqs. (31) and (32), is rewritten as

$$p_{\mathbf{H}}(\boldsymbol{\eta}) = c_\nu\, e^{-\phi(\boldsymbol{\eta})} \quad , \quad \phi(\boldsymbol{\eta}) = -\log \zeta(\boldsymbol{\eta})\,. \tag{33}$$

(iv) Remark. By construction of the PCA of $\mathbf{X}$, we have $\nu \leq N_d$. For the applications, ν is such that $\nu < N_d$ and even can be such that $\nu \ll N_d$ when the dataset is small. Consequently, there is no overfitting of the KDE with the data.

6. Probabilistic learning on manifolds (PLoM) without constraint

(i) Generation of the learned datasets $\mathcal{D}_{n_{\text{MC}}}^{\text{ar}}$ *and* $\mathcal{Y}_{n_{\text{MC}}}^{ar}$. Using only the realizations $\big\{\boldsymbol{\eta}_d^1, \ldots, \boldsymbol{\eta}_d^{N_d}\big\}$ given by Eq. (30), the PLoM method [30, 32] allows for generating $n_{\text{MC}} \gg N_d$ additional realizations $\big\{\boldsymbol{\eta}_{\text{ar}}^1, \ldots, \boldsymbol{\eta}_{\text{ar}}^{n_{\text{MC}}}\big\}$ of $\mathbf{H}$. Using Eq. (28)

allows for computing $n_{\rm MC}$ additional realizations $\left\{\mathbf{x}_{\rm ar}^1, \ldots, \mathbf{x}_{\rm ar}^{n_{\rm MC}}\right\}$ of $\mathbf{X}^{(\nu)}$ such that $\mathbf{x}_{\rm ar}^\ell = \underline{\mathbf{x}} + [\psi]\,[\xi]^{1/2}\,\boldsymbol{\eta}_{\rm ar}^\ell$ for all ℓ in $\left\{1, \ldots, n_{\rm MC}\right\}$. For random vector $\mathbf{X}^{(\nu)}$, the learned dataset is defined by $\mathcal{D}_{n_{\rm MC}}^{\rm ar} = \left\{\mathbf{x}_{\rm ar}^\ell \in \mathbb{R}^{n_x}\ ,\ \ell = 1, \ldots, n_{\rm MC}\right\}$. From the knowledge of the $n_{\rm MC}$ realizations $\left\{\mathbf{x}_{\rm ar}^\ell\right\}_\ell$, we can deduce the $n_{\rm MC}$ additional realizations $\left\{(\mathbf{q}_{\rm ar}^\ell, \mathbf{w}_{\rm ar}^\ell)\right\}_\ell$ in $\mathbb{R}^{n_q} \times \mathbb{R}^{n_w}$ of random variable $\left(\mathbf{Q}^{(\nu)}, \mathbf{W}^{(\nu)}\right) = \mathbf{X}^{(\nu)}$ such that

$$(\mathbf{q}_{\rm ar}^\ell, \mathbf{w}_{\rm ar}^\ell) = \mathbf{x}_{\rm ar}^\ell \quad , \quad \ell \in \left\{1, \ldots, n_{\rm MC}\right\}. \tag{34}$$

For each additional realization $\mathbf{q}_{\rm ar}^\ell$ given by Eq. (34), and using Eq. (11), the corresponding additional realization $\mathbf{y}_{\rm ar}^\ell = \left\{\mathbf{y}_{\rm ar}^\ell(t), t \in \mathcal{J}\right\}$ of stochastic process $\left\{\mathbf{Y}^{(n_q)}(t), t \in \mathcal{J}\right\}$ is computed by $\mathbf{y}_{\rm ar}^\ell(t) = \underline{\mathbf{y}}(t) + [V(t)]\,\mathbf{q}_{\rm ar}^\ell$. The learned dataset for the couple $(\mathbf{Y}^{(n_q)}, \mathbf{W})$ is then defined by $\mathcal{Y}_{n_{\rm MC}}^{\rm ar} = \left\{(\mathbf{y}_{\rm ar}^1, \mathbf{w}_{\rm ar}^1), \ldots, (\mathbf{y}_{\rm ar}^{n_{\rm MC}}, \mathbf{w}_{\rm ar}^{n_{\rm MC}})\right\}$, in which $\mathbf{y}_{\rm ar}^\ell = \left\{\mathbf{y}_{\rm ar}^\ell(t)\, t \in \mathcal{J}\right\}$ and where $\mathbf{w}_{\rm ar}^\ell \in \mathcal{C}_w \subset \mathbb{R}^{n_w}$. As explained in Section 2.5, the time derivatives up to order m_d of $\mathbf{y}_{\rm ar}^\ell$ are computed at all the sampling times belonging to $\mathcal{J}$ by using the time integration scheme. It should be noted that the PLoM method introduces a $\mathbb{M}_{\nu,N_d}$-valued random variable $[\mathbf{H}]$ related to random vector $\mathbf{H}$ whose probability distribution is defined by Eq. (33). The method allows for generating $n_{\rm MC}$ additional realizations $\left\{[\eta_{\rm ar}^1], \ldots, [\eta_{\rm ar}^{n_{\rm MC}}]\right\}$ in $\mathbb{M}_{\nu,N_d}$ of random matrix $[\mathbf{H}]$. For each ℓ, the additional realization $\boldsymbol{\eta}_{\rm ar}^\ell$ is obtained by a random extraction of one column of matrix $[\eta_{\rm ar}^\ell]$, such that, for $k \in \left\{1, \ldots, \nu\right\}$ and for $\ell \in \left\{1, \ldots, n_{\rm MC}\right\}$, we have $\left\{\boldsymbol{\eta}_{\rm ar}^\ell\right\}_k = [\eta_{\rm ar}^\ell]_{k\, j_0^\ell}$ in which $j_0^1, \ldots, j_0^{n_{\rm MC}}$ are $n_{\rm MC}$ independent realizations of a uniform discrete random variable J_0 with values in $\left\{1, \ldots, N_d\right\}$, which are constructed at the beginning of the algorithms and are never modified.

(ii) Computing the random residual of the stochastic equations for the learned realizations. The $n_{\rm MC}$ realizations $\left\{\widehat{\rho}_{\rm ar}^\ell\right\}_\ell$ of random residual $\widehat{\rho}$, corresponding to the learned dataset $\mathcal{Y}_{n_{\rm MC}}^{ar}$, are computed (see Eq. (25)), for $\ell \in \left\{1, \ldots, n_{\rm MC}\right\}$, by $(\widehat{\rho}_{\rm ar}^\ell)^2 = \left(1/(N\, n_{\rm sp})\right) \sum_{n_s=1}^{n_{\rm sp}} \| \mathbf{r}_{\rm ar}^\ell(\tau_{n_s}) \|^2$, in which $\mathbf{r}_{\rm ar}^\ell(\tau_{n_s}) \in \mathbb{R}^N$ is computed (see Eq. (18)), for $t = \tau_{n_s}$, by $\mathbf{r}_{\rm ar}^\ell(\tau_{n_s}) = \boldsymbol{\mathcal{N}}_t\left(\mathbf{y}_{\rm ar}^\ell(\tau_{n_s}), \tau_{n_s}, \mathbf{w}_{\rm ar}^\ell\right)$. The corresponding additional realizations $\left\{\rho_{\rm ar}^1, \ldots, \rho_{\rm ar}^{n_{\rm MC}}\right\}$ of the normalized random residual ρ can then be computed using Eq. (22) and written as,

$$\rho_{\rm ar}^\ell = \widehat{\rho}_{\rm ar}^\ell \,/\, \widehat{\underline{\rho}}_{\rm ref} \quad , \quad \ell = 1, \ldots, n_{\rm MC}. \tag{35}$$

7. Definition of the constraints for the PLoM under constraints

In a first formulation of the proposed theory, only the constraint on the normalized random residual ρ of the stochastic equations was introduced in the PLoM under constraints (it will be *algo* 1 defined in Section 8.3-(i)). Some applications revealed a small difficulty when the constraint on ρ alone was taken into account. As the learning is done on the random vector $\mathbf{X} = (\mathbf{Q}, \mathbf{W})$, the calculation of the probability measure of $\mathbf{X}$, which is based on the minimization of the Kullback-Leibler cross-entropy does not guarantee the preservation of the prior probability distribution of $\mathbf{W}$. Indeed, we obtained good results, but the prior probability distribution of $\mathbf{W}$ is altered during the minimization process. Since the objective is to find the optimal probability measure of the couple $(\mathbf{Y}^{(n_q)}, \mathbf{W})$, that is to say, the optimal probability measure of $\mathbf{X}$, which preserves the prior probability distribution of $\mathbf{W}$ while minimizing the L^2-norm of ρ, we have introduced constraints on $\mathbf{W}$ in order to best preserve its prior probability distribution (it will be *algo* 2 or *algo* 3 defined in Section 8.3-(ii-iii)). In this section, we define the constraints on ρ and $\mathbf{W}$. The PLoM under constraints is formulated using the Kullback-Leibler principle in order to satisfy the constraints on ρ and $\mathbf{W}$. However, as this principle is formulated for the random vector $\mathbf{H}$, the constraints must be expressed in terms of a mathematical expectation of a function of $\mathbf{H}$. For this, we must therefore use the representations constructed previously to express these constraints as a function of $\mathbf{H}$. We introduce the block decomposition of $\underline{\mathbf{x}} \in \mathbb{R}^{n_x}$ and $[\psi] \in \mathbb{M}_{n_x,\nu}$ such that

$$\underline{\mathbf{x}} = \begin{bmatrix} \underline{\mathbf{x}}_q \\ \underline{\mathbf{x}}_w \end{bmatrix} \quad , \quad [\psi] = \begin{bmatrix} [\psi_q] \\ [\psi_w] \end{bmatrix}, \tag{36}$$

in which $\underline{\mathbf{x}}_q \in \mathbb{R}^{n_q}$, $\underline{\mathbf{x}}_w \in \mathbb{R}^{n_w}$, $[\psi_q] \in \mathbb{M}_{n_q,\nu}$, and $[\psi_w] \in \mathbb{M}_{n_w,\nu}$. Therefore, $\mathbf{X}^{(\nu)} = (\mathbf{Q}^{(\nu)}, \mathbf{W}^{(\nu)})$ defined by Eq. (28) can be rewritten as

$$\mathbf{Q}^{(\nu)} = \underline{\mathbf{x}}_q + [\psi_q]\,[\xi]^{1/2}\,\mathbf{H} \quad , \quad \mathbf{W}^{(\nu)} = \underline{\mathbf{x}}_w + [\psi_w]\,[\xi]^{1/2}\,\mathbf{H}. \tag{37}$$

Equation (11) is $\mathbf{Y}^{(n_q)}(t_n) = \underline{\mathbf{y}}(t_n) + [V(t_n)]\,\mathbf{Q}^{(\nu)}$. Substituting in it the representation of $\mathbf{Q}^{(\nu)}$ given by Eq. (37) yields

$$\mathbf{Y}^{(n_q)}(t_n) = \underline{\mathbf{y}}_H(t_n) + [V_H(t_n)]\,\mathbf{H}\,, \tag{38}$$

in which $\underline{\mathbf{y}}_H(t_n) = \underline{\mathbf{y}}(t_n) + [V(t_n)]\,\underline{\mathbf{x}}_q$ and $[V_H(t_n)] = [V(t_n)]\,[\psi_q]\,[\xi]^{1/2}$.

(i) Constraint associated with the normalized random residual of the stochastic equation. Equation (17) is written as $\mathcal{R}(t_n) = \mathcal{N}_t(\mathbf{Y}^{(n_q)}(t_n), t_n, \mathbf{W}^{(\nu)})$. Substituting in its right-hand side the representations $\mathbf{Y}^{(n_q)}(t_n)$ given by Eq. (38) and $\mathbf{W}^{(\nu)}$ given by Eq. (37) show that $\mathcal{R}(t_n)$ depends only on t_n and $\mathbf{H}$. Consequently, using Eqs. (22) with (21), ρ^2 is a nonlinear mapping $\boldsymbol{\eta} \mapsto h^c_\rho(\boldsymbol{\eta})$ of $\mathbf{H}$ such that $\rho^2 = h^c_\rho(\mathbf{H})$. The constraint consists of imposing a reference value to $\|\rho\|^2_{L^2} = E\{\rho^2\}$ that is chosen as $b^c_\rho = E\{\rho^2_{\text{ref}}\}$ with $\rho_{\text{ref}} = \widehat{\rho}_{\text{ref}}/\widehat{\underline{\rho}}_{\text{ref}}$, for which the mathematical expectation is computed using $N_d = N_{d,\text{ref}}$ realizations of the reduced-order representation $\mathbf{Y}^{(n_q)}$ and where $N_{d,\text{ref}}$ is the given reference value of N_d introduced in Section 4.1-(v). The constraint on ρ is then written as

$$E\{h^c_\rho(\mathbf{H})\} = b^c_\rho\,. \tag{39}$$

Note that the nonlinear application h^c_ρ cannot be constructed explicitly by an algebraic calculation. Only, a numerical evaluation of $h^c_\rho(\boldsymbol{\eta})$ for each given $\boldsymbol{\eta}$ in $\mathbb{R}^\nu$ can be performed using the computational model. In addition we will need to calculate the gradient of $h^c_\rho(\boldsymbol{\eta})$ with respect to $\boldsymbol{\eta}$. This important difficulty will be resolved by statistical conditioning (see Section 9).

(ii) Constraints associated with the prior probability model of $\mathbf{W}^{(\nu)}$. In order to avoid a significant modification of the prior probability distribution of $\mathbf{W}^{(\nu)}$, we impose the mean value and the second-order moment of each component $W^{(\nu)}_j$ of $\mathbf{W}^{(\nu)}$. For all $j \in \{1, \ldots, n_w\}$, we thus write $E\{W^{(\nu)}_j\} = b^c_{w,j}$ and $E\{(W^{(\nu)}_j)^2\} = b^c_{w,j+n_w}$ in which $b^c_{w,j}$ and $b^c_{w,j+n_w}$ are the mean value and the second-order moment of W_j (and not of $W^{(\nu)}_j$), which are computed with the training dataset for $N_d = N_{d,\text{ref}}$. The use of the representation $\mathbf{W}^{(\nu)} = \underline{\mathbf{x}}_w + [\psi_w]\,[\xi]^{1/2}\,\mathbf{H}$ allows for defining a nonlinear mapping $\boldsymbol{\eta} \mapsto \mathbf{h}^c_w(\boldsymbol{\eta})$ from $\mathbb{R}^\nu$ into $\mathbb{R}^{2n_w}$ such that $E\{h^c_{w,j}(\mathbf{H})\} = b^c_{w,j}$ and $E\{h^c_{w,j+n_w}(\mathbf{H})\} = b^c_{w,j+n_w}$. Introducing the vector $\mathbf{b}^c_w = (b^c_{w,1}, \ldots, b^c_{w,n_w}, b^c_{w,1+n_w}, \ldots, b^c_{w,2n_w}) \in \mathbb{R}^{2n_w}$, the constraint on $\mathbf{W}^{(\nu)}$ is written as

$$E\{\mathbf{h}^c_w(\mathbf{H})\} = \mathbf{b}^c_w\,. \tag{40}$$

(iii) Global constraint imposed to random vector $\mathbf{H}$. Defining the vector $\mathbf{b}^c = (b^c_\rho, \mathbf{b}^c_w) \in \mathbb{R}^{1+2n_w}$ and the mapping $\mathbf{h}^c = (h^c_\rho, \mathbf{h}^c_w)$ with values in $\mathbb{R}^{1+2n_w}$, Eqs. (39) and (40) can be rewritten as

$$E\{\mathbf{h}^c(\mathbf{H})\} = \mathbf{b}^c\,. \tag{41}$$

8. Probabilistic learning on manifolds under constraints for stochastic process $\mathbf{Y}^c$

For the PLoM under constraints, stochastic process $\mathbf{Y}^{(n_q)}$ is denoted by $\mathbf{Y}^c$.

8.1. Methodology

For the different random variables of the problem, when they will be relative to the PLoM under constraints, we will use an exponent c. Thus $\mathbf{H}$ relates to the training dataset and to the PLoM without constraint, while $\mathbf{H}$ will be noted $\mathbf{H}^c$ for the PLoM under constraints. In these conditions, the random variable $\mathbf{X}^{(\nu)} = (\mathbf{Q}^{(\nu)}, \mathbf{W}^{(\nu)})$ defined by Eq. (37) is denoted by $\mathbf{X}^c = (\mathbf{Q}^c, \mathbf{W}^c)$ when the constraints are applied. We have to find among all the probability density functions that satisfy the constraint $E\{\mathbf{h}^c(\mathbf{H}^c)\} = \mathbf{b}^c$, the pdf $p_{\mathbf{H}^c}$ that is closest to the pdf $p_{\mathbf{H}}$. The use of the Kullback-Leibler minimum cross-entropy principle [54, 55, 56] yields

$$p_{\mathbf{H}^c} = \arg \min_{p \in C_{\text{ad},\mathbf{H}^c}} \int_{\mathbb{R}^\nu} p(\boldsymbol{\eta})\, \log \frac{p(\boldsymbol{\eta})}{p_{\mathbf{H}}(\boldsymbol{\eta})}\, d\boldsymbol{\eta}\,, \tag{42}$$

in which the admissible set is defined by $C_{\text{ad},\mathbf{H}^c} = \{p : \mathbb{R}^\nu \to \mathbb{R}^+, \int_{\mathbb{R}^\nu} p(\boldsymbol{\eta})\, d\boldsymbol{\eta} = 1, \int_{\mathbb{R}^\nu} \mathbf{h}^c(\boldsymbol{\eta})\, p(\boldsymbol{\eta})\, d\boldsymbol{\eta} = \mathbf{b}^c\}$.

8.2. Expressing $p_{\mathbf{H}^c}$ as the solution of an optimization problem

The optimization problem defined by Eq. (42) can only be solved by using an adapted numerical method. A major difficulty is due to the normalization constant of $p_{\mathbf{H}^c}$ that induces considerable difficulties when ν is large, which is bypassed by using the following method. Let $\lambda_0 > 0$ be the real Lagrange multiplier that allows for imposing the normalization constraint $\int_{\mathbb{R}^\nu} p(\boldsymbol{\eta})\, d\boldsymbol{\eta} = 1$. Let $\boldsymbol{\lambda} \in C_{\text{ad},\boldsymbol{\lambda}} \subset \mathbb{R}^{1+2n_w}$ be the one introduced for imposing the constraint $\int_{\mathbb{R}^\nu} \mathbf{h}^c(\boldsymbol{\eta})\, p(\boldsymbol{\eta})\, d\boldsymbol{\eta} = \mathbf{b}^c$ with $C_{\text{ad},\boldsymbol{\lambda}} = \left\{ \boldsymbol{\lambda} \in \mathbb{R}^{1+2n_w},\ \int_{\mathbb{R}^\nu} e^{-\phi(\boldsymbol{\eta}) - <\boldsymbol{\lambda}, \mathbf{h}^c(\boldsymbol{\eta})>}\, d\boldsymbol{\eta} < +\infty \right\}$, in which ϕ is defined by Eq. (33). For $\boldsymbol{\lambda}$ given in $C_{\text{ad},\boldsymbol{\lambda}}$, let $\mathbf{H}_{\boldsymbol{\lambda}}$ be the $\mathbb{R}^\nu$-valued random variable whose pdf with respect to $d\boldsymbol{\eta}$ on $\mathbb{R}^\nu$ is written as

$$p_{\mathbf{H}_{\boldsymbol{\lambda}}}(\boldsymbol{\eta}) = c_0(\boldsymbol{\lambda})\, \exp\left\{ -\phi(\boldsymbol{\eta}) - <\boldsymbol{\lambda}, \mathbf{h}^c(\boldsymbol{\eta})> \right\}, \tag{43}$$

in which $c_0(\boldsymbol{\lambda})$ is the normalization constant that depends on $\boldsymbol{\lambda}$. Using the normalization condition of $p_{\mathbf{H}_{\boldsymbol{\lambda}}}$ yields $c_0(\boldsymbol{\lambda}) = (\int_{\mathbb{R}^\nu} \exp\{-\phi(\boldsymbol{\eta}) - <\boldsymbol{\lambda}, \mathbf{h}^c(\boldsymbol{\eta})>\}\, d\boldsymbol{\eta})^{-1}$. Consequently, $(\boldsymbol{\nabla} c_0(\boldsymbol{\lambda}))/c_0(\boldsymbol{\lambda}) = -E\{\mathbf{h}^c(\mathbf{H}_{\boldsymbol{\lambda}})\}$. Let $\boldsymbol{\lambda} \mapsto \Gamma(\boldsymbol{\lambda})$ be the real function on $C_{\text{ad},\boldsymbol{\lambda}}$ such that $\Gamma(\boldsymbol{\lambda}) = <\boldsymbol{\lambda}, \mathbf{b}^c> - \log c_0(\boldsymbol{\lambda})$ (see [57]). The gradient of Γ at point $\boldsymbol{\lambda}$ is

$$\boldsymbol{\nabla}\Gamma(\boldsymbol{\lambda}) = \mathbf{b}^c - E\left\{\mathbf{h}^c(\mathbf{H}_{\boldsymbol{\lambda}})\right\}, \tag{44}$$

while its Hessian matrix $[\Gamma''(\boldsymbol{\lambda})] = [\text{cov}\left\{\mathbf{h}^c(\mathbf{H}_{\boldsymbol{\lambda}})\right\}]$ is the covariance matrix of random vector $\mathbf{h}^c(\mathbf{H}_{\boldsymbol{\lambda}})$. It will be assume that $[\Gamma''(\boldsymbol{\lambda})]$ belongs to $\mathbb{M}^+_{1+2n_w}$ for all $\boldsymbol{\lambda} \in C_{\text{ad},\boldsymbol{\lambda}}$. This hypothesis is related to the algebraic independence of the constraints and will be checked in the algorithm for each application. Consequently, $\boldsymbol{\lambda} \mapsto \Gamma(\boldsymbol{\lambda})$ is a strictly convex function in $C_{\text{ad},\boldsymbol{\lambda}}$. Nevertheless, it cannot be proven that admissible set $C_{\text{ad},\boldsymbol{\lambda}}$ is convex for the general case because the mapping h^c_ρ depends on the application that is performed. So, we will assume that the following optimization problem

$$\boldsymbol{\lambda}^{\text{sol}} = \arg \min_{\boldsymbol{\lambda} \in C_{\text{ad},\boldsymbol{\lambda}}} \Gamma(\boldsymbol{\lambda}), \tag{45}$$

has a unique solution $\boldsymbol{\lambda}^{\text{sol}}$ that is such that $\boldsymbol{\nabla}\Gamma(\boldsymbol{\lambda}^{\text{sol}}) = \mathbf{0}_{1+2n_w}$ and that

$$p_{\mathbf{H}^c}(\boldsymbol{\eta}) = \left\{ p_{\mathbf{H}_{\boldsymbol{\lambda}}}(\boldsymbol{\eta}) \right\}_{\boldsymbol{\lambda}=\boldsymbol{\lambda}^{\text{sol}}} \quad , \quad p_{\mathbf{H}}(\boldsymbol{\eta}) = \left\{ p_{\mathbf{H}_{\boldsymbol{\lambda}}}(\boldsymbol{\eta}) \right\}_{\boldsymbol{\lambda}=\mathbf{0}_{1+2n_w}}. \tag{46}$$

It should be noted that $\Gamma(\boldsymbol{\lambda})$ cannot be evaluated for any given $\boldsymbol{\lambda}$ because function Γ depends on normalization constant $c_0(\boldsymbol{\lambda})$ that involves an integral on $\mathbb{R}^\nu$, which cannot be calculated when ν is large.

8.3. Iterative algorithm for computing the Lagrange multipliers $\boldsymbol{\lambda}^{\text{sol}}$

Taking into account Eqs. (41) and (44), the solution $\boldsymbol{\lambda}^{\text{sol}}$ of the optimization problem defined by Eq. (45) is computed by solving the equation $\boldsymbol{\nabla}\Gamma(\boldsymbol{\lambda}) = \mathbf{0}_{1+2n_w}$ in $\boldsymbol{\lambda}$ using a Newton iterative method, which allows for constructing a solution $\mathbf{H}_{\boldsymbol{\lambda}^{\text{sol}}}$ that satisfies the constraint defined by Eq. (41). The Lagrange multiplier $\boldsymbol{\lambda}$ is rewritten as $\boldsymbol{\lambda} = (\lambda_\rho, \boldsymbol{\lambda}_w)$ in which the real-valued Lagrange multiplier λ_ρ is associated with the constraint $E\left\{h^c_\rho(\mathbf{H})\right\} = b^c_\rho$ (see Eq. (39)) and where the vector-valued multiplier $\boldsymbol{\lambda}_w$ is associated with the constraint $E\left\{\mathbf{h}^c_w(\mathbf{H})\right\} = \mathbf{b}^c_w$ (see Eq. (40)). On this basis, three iteration algorithms, referred as *algo* 1 to *algo* 3, have been thoroughly tested through the applications presented in Sections 10, 11, and 12.

- The first algorithm, *algo* 1, does not take into account the constraints associated with the prior probability distribution of $\mathbf{W}^{(\nu)}$. As we have explained in Section 7, this algorithm can be not suitable.

- The two other algorithms, *algo* 2 and *algo* 3, take into account not only the constraint on ρ but also the constraints on $\mathbf{W}^{(\nu)}$. Algorithm *algo* 2 is the classical approach for which the constraints vector is globally taken. We have been able to verify that this *algo* 2 is not always efficient, involving convergence difficulties. This phenomenon is linked to the fact that the two constraints (one on ρ and the other on $\mathbf{W}^{(\nu)}$) play in opposite directions (which does not prevent obtaining the minimum, but with potential numerical difficulties). It is for this reason that we have introduced algorithm *algo* 3 for which the constraint on ρ and the constraints on $\mathbf{W}^{(\nu)}$ are decoupled for the computation of the Lagrange multipliers. The numerous numerical tests, which have been carried out, have shown that *algo* 3 is often the most efficient algorithm.

(i) Algorithm algo 1. The only constraint is on the normalized random residual ρ. The iteration algorithm is written, for $i \geq 1$, as $\lambda^{i+1}_\rho = \lambda^i_\rho - \Gamma''_\rho(\lambda^i_\rho)^{-1}\, \Gamma'_\rho(\lambda^i_\rho)$ with $\lambda^1_\rho = 0$, where $\Gamma'_\rho(\lambda^i_\rho) = b^c_\rho - E\left\{h^c_\rho(\mathbf{H}_{\lambda^i_\rho})\right\}$ and where $\Gamma''_\rho(\lambda^i_\rho) = \text{var}(h^c_\rho(\mathbf{H}_{\lambda^i_\rho}))$

is the variance. At each iteration i, the convergence of the iteration algorithm is controlled by calculating the error $\text{err}_R(i)$ for $R = \rho^2$ and the corresponding error $\text{err}_W(i)$ for $\mathbf{W}^{(\nu)}$, such that

$$\text{err}_R(i) = |\, b_\rho^c - E\big\{h_\rho^c(\mathbf{H}_{\lambda_\rho^i})\big\}\,| \,/\, b_\rho^c \quad , \quad \text{err}_W(i) = \|\, \mathbf{b}_w^c - E\big\{\mathbf{h}_w^c(\mathbf{H}_{\lambda_\rho^i})\big\}\,\| \,/\, \|\, \mathbf{b}_w^c \,\| \,. \tag{47}$$

(ii) Algorithm algo 2. The constraints are on the normalized random residual ρ and on the random control parameter $\mathbf{W}^{(\nu)}$. The iteration algorithm is written as

$$\boldsymbol{\lambda}^{i+1} = \boldsymbol{\lambda}^i - [\Gamma''(\boldsymbol{\lambda}^i)]^{-1}\, \boldsymbol{\Gamma}'(\boldsymbol{\lambda}^i) \quad , \quad i \geq 1 \,, \tag{48}$$

with $\boldsymbol{\lambda}^1 = \mathbf{0}_{1+2n_w}$, where $\boldsymbol{\Gamma}'(\boldsymbol{\lambda}^i) = \mathbf{b}^c - E\big\{\mathbf{h}^c(\mathbf{H}_{\boldsymbol{\lambda}^i})\big\}$, and where $[\Gamma''(\boldsymbol{\lambda}^i)] = [\text{cov}\big\{\mathbf{h}^c(\mathbf{H}_{\boldsymbol{\lambda}^i})\big\}]$ is the covariance matrix. At each iteration i, by using $\boldsymbol{\lambda}^i = (\lambda_\rho^i, \boldsymbol{\lambda}_w^i)$, the convergence of the iteration algorithm is controlled by calculating the error

$$\text{err}_{RW}(i) = \sqrt{\text{err}_R(i)^2 + \text{err}_W(i)^2} \,, \tag{49}$$

in which $\text{err}_R(i)$ and $\text{err}_W(i)$ are defined by

$$\text{err}_R(i) = |\, b_\rho^c - E\big\{h_\rho^c(\mathbf{H}_{\boldsymbol{\lambda}^i})\big\}\,| \,/\, b_\rho^c \quad , \quad \text{err}_W(i) = \|\, \mathbf{b}_w^c - E\big\{\mathbf{h}_w^c(\mathbf{H}_{\boldsymbol{\lambda}^i})\big\}\,\| \,/\, \|\, \mathbf{b}_w^c \,\| \,. \tag{50}$$

(iii) Algorithm algo 3. It is algorithm *algo* 2 for which λ_ρ and $\boldsymbol{\lambda}_w$ are decoupled in Eq. (48). The iteration algorithm is then written, for $i \geq 1$, as

$$(\lambda_\rho^i, \boldsymbol{\lambda}_w^i) = \boldsymbol{\lambda}^i \quad , \quad \lambda_\rho^{i+1} = \lambda_\rho^i - \Gamma_\rho''(\boldsymbol{\lambda}^i)^{-1}\, \Gamma_\rho'(\boldsymbol{\lambda}^i) \quad , \quad \boldsymbol{\lambda}_w^{i+1} = \boldsymbol{\lambda}_w^i - [\Gamma_w''(\boldsymbol{\lambda}^i)]^{-1}\, \boldsymbol{\Gamma}_w'(\boldsymbol{\lambda}^i) \quad , \quad \boldsymbol{\lambda}^{i+1} = (\lambda_\rho^{i+1}, \boldsymbol{\lambda}_w^{i+1}) \,, \tag{51}$$

with $\boldsymbol{\lambda}^1 = \mathbf{0}_{1+2n_w}$, where $(\Gamma_\rho'(\boldsymbol{\lambda}^i), \boldsymbol{\Gamma}_w'(\boldsymbol{\lambda}^i)) = \boldsymbol{\Gamma}'(\boldsymbol{\lambda}^i)$, and where $\Gamma_\rho''(\boldsymbol{\lambda}^i) = [\Gamma''(\boldsymbol{\lambda}^i)]_{11}$ and $[\Gamma_w''(\boldsymbol{\lambda}^i)]_{jj'} = [\Gamma''(\boldsymbol{\lambda}^i)]_{1+j,1+j'}$ for j and j' in $\{1, \ldots, n_w\}$. At each iteration i, the convergence of the iteration algorithm is controlled by calculating the errors $\text{err}_R(i)$ and $\text{err}_W(i)$ by Eq. (50), and $\text{err}_{RW}(i)$ by Eq. (49).

8.4. Additional realizations of $\boldsymbol{H}_{\boldsymbol{\lambda}^i}$ and learned dataset for $(\boldsymbol{Y}^c, \boldsymbol{W}^c)$

(i) Learned dataset $\boldsymbol{\mathcal{Y}}_{n_{\text{MC}}}^{c,\text{ar}}$ for $(\mathbf{Y}^c, \mathbf{W}^c)$. For each $\boldsymbol{\lambda}^i$ of the iteration algorithm, $n_{\text{MC}} \gg N_d$ addition realizations of $\mathbf{H}_{\boldsymbol{\lambda}^i}$ are generated for estimating $E\big\{\mathbf{h}^c(\mathbf{H}_{\boldsymbol{\lambda}^i})\big\}$ and $[\text{cov}\big\{\mathbf{h}^c(\mathbf{H}_{\boldsymbol{\lambda}^i})\big\}]$. At convergence for $i = i_{\text{opt}}$, we have $\boldsymbol{\lambda}^{\text{sol}} \simeq \boldsymbol{\lambda}^{i_{\text{opt}}}$. Taking into account Eq. (46), we have $p_{\mathbf{H}^c} \simeq p_{\mathbf{H}_\lambda}$ for $\boldsymbol{\lambda} = \boldsymbol{\lambda}^{i_{\text{opt}}}$. Consequently, at the last iteration, the generator allows for computing n_{MC} additional realizations $\big\{\boldsymbol{\eta}_{c,\text{ar}}^1, \ldots, \boldsymbol{\eta}_{c,\text{ar}}^{\text{MC}}\big\}$ of $\mathbf{H}^c$ for which the generator is described in Paragraphs (iv) to (vi) below. The learned dataset $\boldsymbol{\mathcal{Y}}_{n_{\text{MC}}}^{c,\text{ar}}$ under constraints is defined by $\boldsymbol{\mathcal{Y}}_{n_{\text{MC}}}^{c,\text{ar}} = \big\{(\mathbf{y}_{c,\text{ar}}^1, \mathbf{w}_{c,\text{ar}}^1), \ldots, (\mathbf{y}_{c,\text{ar}}^{n_{\text{MC}}}, \mathbf{w}_{c,\text{ar}}^{n_{\text{MC}}})\big\}$, in which $\mathbf{w}_{c,\text{ar}}^\ell$ and $\mathbf{y}_{c,\text{ar}}^\ell = \big\{\mathbf{y}_{c,\text{ar}}^\ell(t_n) \in \mathbb{R}^N \,,\, n = 1, \ldots, n_{\text{time}}\big\}$ are given, using Eqs. (37) and (38), by $\mathbf{w}_{c,\text{ar}}^\ell = \underline{\mathbf{x}}_w + [\psi_w]\,[\xi]^{1/2}\, \boldsymbol{\eta}_{c,\text{ar}}^\ell$ and for $n \in \big\{1, \ldots, n_{\text{time}}\big\}$, by $\mathbf{y}_{c,\text{ar}}^\ell(t_n) = \underline{\mathbf{y}}_H(t_n) + [V_H(t_n)]\, \boldsymbol{\eta}_{c,\text{ar}}^\ell$.

(ii) Computing the random residual of the stochastic equation for the learned realizations under constraints. The n_{MC} realizations $\big\{\widehat{\rho}_{c,\text{ar}}^\ell\big\}_\ell$ of random residual $\widehat{\rho}$, corresponding to the learned dataset $\boldsymbol{\mathcal{Y}}_{n_{\text{MC}}}^{c,\text{ar}}$ under constraints, are computed (see Eq. (25)), for $\ell \in \big\{1, \ldots, n_{\text{MC}}\big\}$, by $(\widehat{\rho}_{c,\text{ar}}^\ell)^2 = \big(1/(N\, n_{\text{sp}})\big) \sum_{n_s=1}^{n_{\text{sp}}} \|\, \mathbf{r}_{c,\text{ar}}^\ell(\tau_{n_s})\,\|^2$, in which $\mathbf{r}_{c,\text{ar}}^\ell(\tau_{n_s}) \in \mathbb{R}^N$ is computed (see Eq. (18)), for $t = \tau_{n_s}$, by $\mathbf{r}_{c,\text{ar}}^\ell(\tau_{n_s}) = \boldsymbol{\mathcal{N}}_t\big(\mathbf{y}_{c,\text{ar}}^\ell(\tau_{n_s}), \tau_{n_s}, \mathbf{w}_{c,\text{ar}}^\ell\big)$.

(iii) Construction of the diffusion-maps basis. The PLoM [30, 32] uses the diffusion-maps basis [47] for projecting the nonlinear ISDE constructed by the PLoM method. This projection allows for preserving the concentration of the probability measure of $\mathbf{X}^c = (\mathbf{Q}^c, \mathbf{W}^c)$ on the manifold (see the definition of $\mathbf{X}^c$ at the beginning of Section 8.1). The diffusion-maps basis depends only on $\big\{\boldsymbol{\eta}_d^1, \ldots, \boldsymbol{\eta}_d^{N_d}\big\}$ defined by Eq. (30) (initial dataset) and is independent of the iteration algorithm. This basis is represented by the matrix $[g] = [\mathbf{g}^1 \ldots \mathbf{g}^m] \in \mathbb{M}_{N_d,m}$ with $1 < m \leq N_d$. Let $[\mathbb{K}]$ be the symmetric $(N \times N)$ real matrix such that $[\mathbb{K}]_{\ell\ell'} = \exp\big\{ -(4\,\varepsilon_{\text{diff}})^{-1} \|\, \boldsymbol{\eta}_d^\ell - \boldsymbol{\eta}_d^{\ell'}\,\|^2 \big\}$, which depends on a real smoothing parameter $\varepsilon_{\text{diff}} > 0$. Let $[\mathbb{P}]$ be the transition matrix in $\mathbb{M}_{N_d}$ of a Markov chain such that $[\mathbb{P}] = [\mathbb{b}]^{-1}\,[\mathbb{K}]$ in which $[\mathbb{b}]$ is the positive-definite diagonal real matrix such that $[\mathbb{b}]_{i\ell} = \delta_{i\ell} \sum_{\ell'=1}^{N_d} [\mathbb{K}]_{\ell\ell'}$. For m fixed in $\{1, \ldots, N_d\}$, let $\mathbf{g}^1, \ldots, \mathbf{g}^m$ be the right eigenvectors in $\mathbb{R}^{N_d}$ of matrix $[\mathbb{P}]$ such that $[\mathbb{P}]\,\mathbf{g}^\alpha = \kappa_\alpha\, \mathbf{g}^\alpha$ in which the eigenvalues are sorted such that $1 = \kappa_1 > \kappa_2 > \ldots > \kappa_m$. It can easily be proven that the eigenvalues are positive

and the largest is $\kappa_1 = 1$ for which all the components of the corresponding eigenvector $\mathbf{g}^1$ are equal. Considering the normalization $[g]^T\,[\mathbb{b}]\,[g] = [I_m]$, the right-eigenvalue problem of the nonsymmetric matrix $[\mathbb{P}]$ can then be done solving the eigenvalue problem $[\mathbb{b}]^{-1/2}\,[\mathbb{K}]\,[\mathbb{b}]^{-1/2}\,\boldsymbol{\gamma}^\alpha = \kappa_\alpha\,\boldsymbol{\gamma}^\alpha$ related to a positive-definite symmetric real matrix with the orthonormalization $<\boldsymbol{\gamma}^\alpha, \boldsymbol{\gamma}^\beta>= \delta_{\alpha\beta}$. Therefore, $\mathbf{g}^\alpha$ can be deduced from $\boldsymbol{\gamma}^\alpha$ by $\mathbf{g}^\alpha = [\mathbb{b}]^{-1/2}\,\boldsymbol{\gamma}^\alpha$. This construction introduces two hyperparameters: the dimension $m \leq N_d$ and the smoothing parameter $\varepsilon_{\text{diff}} > 0$. We refer the reader to [32, 31] for estimating their values. We introduce the matrix $[a]$ defined by

$$[a] = [g]\,\big([g]^T\,[g]\big)^{-1} \in \mathbb{M}_{N_d,m}\,. \tag{52}$$

(iv) Generator of random vector $\mathbf{H}_{\lambda^i}$ by using PLoM algorithm under constraints. Now and throughout the rest of Section 8.4, we remove the exponent i on $\boldsymbol{\lambda}$. Let $\Big\{\big([\mathcal{Z}_{\boldsymbol{\lambda}}(r)], [\boldsymbol{\mathcal{X}}_{\boldsymbol{\lambda}}(r)]\big), r \geq 0\Big\}$ be the stochastic process with values in $\mathbb{M}_{\nu,m} \times \mathbb{M}_{\nu,m}$, which is the solution of the following ISDE,

$$d[\mathcal{Z}_{\boldsymbol{\lambda}}(r)] = [\boldsymbol{\mathcal{X}}_{\boldsymbol{\lambda}}(r)]\,dr\,, \qquad r > 0\,, \tag{53}$$

$$d[\boldsymbol{\mathcal{X}}_{\boldsymbol{\lambda}}(r)] = [\mathcal{L}_{\boldsymbol{\lambda}}([\mathcal{Z}_{\boldsymbol{\lambda}}(r)])]\,dr - \frac{1}{2}\,f_0[\boldsymbol{\mathcal{X}}_{\boldsymbol{\lambda}}(r)]\,dr + \sqrt{f_0}\,d[\mathbf{W}^{\text{wien}}(r)]\,[a]\,, \qquad r > 0\,, \tag{54}$$

$$[\mathcal{Z}_{\boldsymbol{\lambda}}(0)] = [\eta_d]\,[a] \quad , \quad [\boldsymbol{\mathcal{X}}_{\boldsymbol{\lambda}}(0)] = [v_0]\,[a] \quad a.s.\,, \qquad r = 0\,, \tag{55}$$

in which:

(1) $[a]$ is the matrix defined by Eq. (52), $[\eta_d] = [\boldsymbol{\eta}_d^1 \ldots \boldsymbol{\eta}_d^{N_d}] \in \mathbb{M}_{\nu,N_d}$, and $[v_0] \in \mathbb{M}_{\nu,N_d}$ is such that $\big\{[v_0]_{k\ell}\big\}_{k\ell}$ are $\nu \times N_d$ independent normalized Gaussian real-valued random variables that are fixed and independent of iteration i.

(2) $f_0 > 0$ is a free parameter allowing the dissipation to be controlled in the stochastic dynamical system and therefore to quickly reach the stationary solution corresponding to the invariant measure. This parameter is chosen such that $f_0 \leq 4/\widehat{s}_\nu$. The value 4 is generally a good choice.

(3) $\big\{[\mathbf{W}^{\text{wien}}(r)], r \in \mathbb{R}^+\big\}$ is the stochastic process, defined on $(\Theta, \mathcal{T}, \mathcal{P})$, indexed by $\mathbb{R}^+$, with values in $\mathbb{M}_{\nu,N_d}$, for which the columns of $[\mathbf{W}^{\text{wien}}]$ are N_d independent copies of the $\mathbb{R}^\nu$-valued normalized Wiener process whose matrix-valued autocorrelation function is $\min(r, r')\,[I_\nu]$. This stochastic process is independent of iteration i (this is the same for all the iterations) and is statistically independent of the random control parameter $\mathbf{W}$.

(4) the random matrix $[\mathcal{L}_{\boldsymbol{\lambda}}([\mathcal{Z}_{\boldsymbol{\lambda}}(r)])]$ with values in $\mathbb{M}_{\nu,m}$ is written as $[\mathcal{L}_{\boldsymbol{\lambda}}([\mathcal{Z}_{\boldsymbol{\lambda}}(r)])] = [L_{\boldsymbol{\lambda}}([\mathcal{Z}_{\boldsymbol{\lambda}}(r)]\,[g]^T)]\,[a]$, in which, for $k = 1, \ldots, \nu$, for $\ell = 1, \ldots, N_d$, and for $[u] = [\mathbf{u}^1 \ldots \mathbf{u}^{N_d}] \in \mathbb{M}_{\nu,N_d}$ with $\mathbf{u}^\ell = (u_1^\ell, \ldots, u_\nu^\ell)$, we have

$$[L_{\boldsymbol{\lambda}}([u])]_{k\ell} = \frac{1}{\zeta(\mathbf{u}^\ell)}\frac{\partial \zeta(\mathbf{u}^\ell)}{\partial u_k^\ell} - <\boldsymbol{\lambda}\,, \frac{\partial \mathbf{h}^c(\mathbf{u}^\ell)}{\partial u_k^\ell}>\,, \tag{56}$$

in which ζ is defined by Eq. (32) and where the gradient of ζ is such that

$$\boldsymbol{\nabla}_{\mathbf{u}^\ell}\zeta(\mathbf{u}^\ell) = \frac{1}{\widehat{s}_\nu^2}\frac{1}{N_d}\sum_{j=1}^{N_d}\Big(\frac{\widehat{s}_\nu}{s_\nu}\boldsymbol{\eta}_d^j - \mathbf{u}^\ell\Big)\,\exp\Big\{-\frac{1}{2\widehat{s}_\nu^2}\|\frac{\widehat{s}_\nu}{s_\nu}\boldsymbol{\eta}_d^j - \mathbf{u}^\ell\|^2\Big\}\,. \tag{57}$$

For f_0 fixed, let r_0 be a sufficiently large positive real number in order that the stochastic process $\Big\{([\mathcal{Z}_{\boldsymbol{\lambda}}(r)], [\boldsymbol{\mathcal{X}}_{\boldsymbol{\lambda}}(r)]),\ r \geq r_0\Big\}$ is stationary. For given r_{st} such that $r_{\text{st}} \geq r_0$, the probability distribution of random variable $\big([\mathcal{Z}_{\boldsymbol{\lambda}}(r_{\text{st}})], [\boldsymbol{\mathcal{X}}_{\boldsymbol{\lambda}}(r_{\text{st}})]\big)$ is the invariant measure of Eqs. (53) and (54). As proven in [30, 32], the $\mathbb{R}^\nu$-valued random variable $\mathbf{H}_{\boldsymbol{\lambda}}$ is a column (see Paragraph (v) below) of random matrix $[\mathbf{H}_\lambda]$ with values in $\mathbb{M}_{\nu,N_d}$ such that

$$[\mathbf{H}_\lambda] = [\mathcal{Z}_{\boldsymbol{\lambda}}(r_{\text{st}})]\,[g]^T\,. \tag{58}$$

(v) Generation of n_{MC} realizations of $\mathbf{H}_{\boldsymbol{\lambda}}$. The Störmer-Verlet algorithm (see Appendix A) is used for solving Eqs. (53) to (55) as proposed in [52, 30], which allows for generating n_{MC} realizations $\big\{[z_{\boldsymbol{\lambda}}^1], \ldots, [z_{\boldsymbol{\lambda}}^{n_{\text{MC}}}]\big\}$ in $\mathbb{M}_{\nu,m}$ of random matrix $[\mathcal{Z}_{\boldsymbol{\lambda}}]$. It can then be deduced n_{MC} additional realizations $\big\{\boldsymbol{\eta}_{\boldsymbol{\lambda}}^1, \ldots, \boldsymbol{\eta}_{\boldsymbol{\lambda}}^{n_{\text{MC}}}\big\}$ in $\mathbb{R}^\nu$ of random vector $\mathbf{H}_{\boldsymbol{\lambda}}$ such that, for $k \in \big\{1, \ldots, \nu\big\}$ and for $\ell \in \big\{1, \ldots, n_{\text{MC}}\big\}$, we have $\big\{\boldsymbol{\eta}_{\boldsymbol{\lambda}}^\ell\big\}_k = [\eta_{\boldsymbol{\lambda}}^\ell]_{k\,j_0^\ell}$ in which $[\eta_{\boldsymbol{\lambda}}^\ell] = [z_{\boldsymbol{\lambda}}^\ell]\,[g]^T \in \mathbb{M}_{\nu,N}$ and where j_0^ℓ

is defined in Section 6-(i).

9. Computational statistics method for estimating the derivative of $\mathbf{h}^c$

Equation (56) shows that we have to compute, for any given $\mathbf{u}^\ell = (u_1^\ell, \dots, u_\nu^\ell)$ in $\mathbb{R}^\nu$, the partial derivative $\partial \mathbf{h}^c(\mathbf{u}^\ell)/\partial u_k^\ell$ for $k = 1, \dots, \nu$ in which $\mathbf{h}^c(\mathbf{u}^\ell) = \left(h_\rho^c(\mathbf{u}^\ell), \mathbf{h}_w^c(\mathbf{u}^\ell)\right) \in \mathbb{R}^{1+2n_w}$ (see Section 7-(iii)). The first component h_ρ^c of $\mathbf{h}^c$ involves the nonlinear stochastic equation and since ν can be large, the computation of the partial derivatives is a very difficult problem. The construction of a surrogate model of function $\mathbf{u}^\ell \mapsto h_\rho^c(\mathbf{u}^\ell)$ by using a deterministic approach, such as the meshless methods [58, 59, 60, 61, 62, 63], is not adapted to this difficult and numerically expensive problem taking into account the dimension ν of $\mathbf{u}^\ell$ and the possible complexity of function h_ρ^c. We thus propose to develop a computational statistics method based on statistical conditioning for the first component h_ρ^c of $\mathbf{h}^c$. For the other components, that is to say for function $\mathbf{h}_w^c$, there is no difficulty. We then have to construct, for all $\boldsymbol{\eta}$ in $\mathbb{R}^\nu$, an estimation of the gradient $\boldsymbol{\nabla} h_\rho^c(\boldsymbol{\eta})$ of $h_\rho^c(\boldsymbol{\eta})$ and, for all i in $\left\{1, \dots, 2n_w\right\}$, the gradient $\boldsymbol{\nabla} h_{w,i}^c(\boldsymbol{\eta})$ of $h_{w,i}^c(\boldsymbol{\eta})$.

9.1. Estimation of the gradient of $h_\rho^c(\boldsymbol{\eta})$

For all $\boldsymbol{\eta}$ in $\mathbb{R}^\nu$, the following statistical conditioning is used for calculating $h_\rho^c(\boldsymbol{\eta})$,

$$h_\rho^c(\boldsymbol{\eta}) = E\left\{ \rho^2 \,|\, \mathbf{H} = \boldsymbol{\eta} \right\}, \tag{59}$$

in which the conditional mathematical expectation of ρ^2 given $\mathbf{H} = \boldsymbol{\eta}$ is estimated by using the n_{MC} realizations $\left\{\rho_{\text{ar}}^1, \dots, \rho_{\text{ar}}^{n_{\text{MC}}}\right\}$ of ρ (see Eq. (35)) and the n_{MC} realizations $\left\{\boldsymbol{\eta}_{\text{ar}}^1, \dots, \boldsymbol{\eta}_{\text{ar}}^{\text{MC}}\right\}$ of random vector $\mathbf{H}$ (see Section 6-(i)), computed by the PLoM without constraint. These realizations have been chosen because the PLoM method without constraint already gives a learned dataset of dimension $n_{\text{MC}} \gg N_d$, which already verifies well the stochastic equation.

(i) Normalization and gradient of h_ρ^c. The scaling of $\mathbf{H}$ and of its realizations $\left\{\boldsymbol{\eta}_{\text{ar}}^\ell\right\}_\ell$ is necessary. Let $\underline{\boldsymbol{\eta}}_{\text{ar}} = (\underline{\eta}_{\text{ar},1}, \dots, \underline{\eta}_{\text{ar},\nu})$ and $\boldsymbol{\sigma}_{\text{ar}} = (\sigma_{\text{ar},1}, \dots, \sigma_{\text{ar},\nu})$ be the empirical estimates of the mean value and the standard deviation of $\mathbf{H}$, computed with $\left\{\boldsymbol{\eta}_{\text{ar}}^1, \dots, \boldsymbol{\eta}_{\text{ar}}^{\text{MC}}\right\}$. Let $\widehat{\mathbf{H}} = (\widehat{H}_1, \dots, \widehat{H}_\nu)$ and $\widehat{\boldsymbol{\eta}} = (\widehat{\eta}_1, \dots, \widehat{\eta}_\nu)$ such that, for all $\boldsymbol{\eta} = (\eta_1, \dots, \eta_\nu)$ in $\mathbb{R}^\nu$ and for all $k \in \left\{1, \dots, \nu\right\}$, $\widehat{H}_k = (H_k - \underline{\eta}_{\text{ar},k})/\sigma_{\text{ar},k}$ and $\widehat{\eta}_k = (\eta_k - \underline{\eta}_{\text{ar},k})/\sigma_{\text{ar},k}$. Consequently, Eq. (59) can be rewritten as

$$h_\rho^c(\boldsymbol{\eta}) = \widehat{h}_\rho^c(\widehat{\boldsymbol{\eta}}) \quad , \quad \eta_k = \underline{\eta}_{\text{ar},k} + \sigma_{\text{ar},k}\widehat{\eta}_k \,, \tag{60}$$

$$\widehat{h}_\rho^c(\widehat{\boldsymbol{\eta}}) = E\left\{ \rho^2 \,|\, \widehat{\mathbf{H}} = \widehat{\boldsymbol{\eta}} \right\}, \tag{61}$$

in which $\widehat{\boldsymbol{\eta}} = \widehat{\boldsymbol{\eta}}(\boldsymbol{\eta})$ has to be seen as a function of $\boldsymbol{\eta}$. The realizations of $\widehat{\mathbf{H}}$ are $\left\{\widehat{\boldsymbol{\eta}}_{\text{ar}}^1, \dots, \widehat{\boldsymbol{\eta}}_{\text{ar}}^{\text{MC}}\right\}$ such that, for $\ell = 1, \dots, n_{\text{MC}}$ and $k = 1, \dots, \nu$, we have $\widehat{\eta}_{\text{ar},k}^\ell = (\eta_{\text{ar},k}^\ell - \underline{\eta}_{\text{ar},k})/\sigma_{\text{ar},k}$. From Eqs. (60), it can be deduced that, for all $k \in \left\{1, \dots, \nu\right\}$, we have $\partial h_\rho^c(\boldsymbol{\eta})/\partial \eta_k = \sigma_{\text{ar},k}^{-1}\, \partial \widehat{h}_\rho^c(\widehat{\boldsymbol{\eta}})/\partial \widehat{\eta}_k$.

(ii) Estimation of $\widehat{h}_\rho^c(\widehat{\boldsymbol{\eta}})$. Let $(r, \widehat{\boldsymbol{\eta}}) \mapsto p_{R,\widehat{\mathbf{H}}}(r, \widehat{\boldsymbol{\eta}})$ be the joint pdf on $\mathbb{R} \times \mathbb{R}^\nu$ of the positive-valued random variable $R = \rho^2$ and $\widehat{\mathbf{H}}$ with respect to the Lebesgue measure on $\mathbb{R} \times \mathbb{R}^\nu$. We then have

$$p_{\widehat{\mathbf{H}}}(\widehat{\boldsymbol{\eta}}) = \int_{\mathbb{R}} p_{R,\widehat{\mathbf{H}}}(r, \widehat{\boldsymbol{\eta}})\, dr \,. \tag{62}$$

From Eq. (61), it can be deduced that

$$\widehat{h}_\rho^c(\widehat{\boldsymbol{\eta}}) = \frac{1}{p_{\widehat{\mathbf{H}}}(\widehat{\boldsymbol{\eta}})} \int_{\mathbb{R}} r\, p_{R,\widehat{\mathbf{H}}}(r, \widehat{\boldsymbol{\eta}})\, dr \,. \tag{63}$$

Let $\left\{r_{\text{ar}}^1, \dots r_{\text{ar}}^{n_{\text{MC}}}\right\}$ be the realizations of R such that $r_{\text{ar}}^\ell = (\rho_{\text{ar}}^\ell)^2$, and let σ_R be its standard deviation estimated using

$\left\{r_{\rm ar}^1,\dots r_{\rm ar}^{n_{\rm MC}}\right\}$. For $n_{\rm MC}$ sufficiently large, the Gaussian kernel-density estimation [53] of $p_{R,\widehat{\mathbf{H}}}(r,\widehat{\boldsymbol{\eta}})$ can be written as

$$p_{R,\widehat{\mathbf{H}}}(r,\widehat{\boldsymbol{\eta}}) \simeq \frac{1}{n_{\rm MC}}\sum_{\ell=1}^{n_{\rm MC}} \frac{1}{\sigma_R(\sqrt{2\pi}s_{\rm SB})^{\nu+1}} \exp\Big\{-\frac{1}{2s_{\rm SB}^2}\big((\frac{r_{\rm ar}^\ell - r}{\sigma_R})^2 + \|\widehat{\boldsymbol{\eta}}_{\rm ar}^\ell - \widehat{\boldsymbol{\eta}}\|^2\big)\Big\}, \tag{64}$$

in which $s_{\rm SB} = \left\{4/\big(n_{\rm MC}(2+\nu+1)\big)\right\}^{1/(4+\nu+1)}$ is the usual multidimensional optimal Silverman bandwidth. Using Eqs. (62) and (64) yields

$$p_{\widehat{\mathbf{H}}}(\widehat{\boldsymbol{\eta}}) \simeq \frac{1}{n_{\rm MC}}\sum_{\ell=1}^{n_{\rm MC}} \frac{1}{(\sqrt{2\pi}s_{\rm SB})^{\nu}} \exp\Big\{-\frac{1}{2s_{\rm SB}^2}\|\widehat{\boldsymbol{\eta}}_{\rm ar}^\ell - \widehat{\boldsymbol{\eta}}\|^2\Big\}. \tag{65}$$

From Eqs. (63) to (65), it can be deduced that

$$\widehat{h}_\rho^c(\widehat{\boldsymbol{\eta}}) = \frac{\mathcal{A}(\widehat{\boldsymbol{\eta}})}{\mathcal{B}(\widehat{\boldsymbol{\eta}})}, \tag{66}$$

in which $\mathcal{A}(\widehat{\boldsymbol{\eta}})$ and $\mathcal{B}(\widehat{\boldsymbol{\eta}})$ are defined by

$$\mathcal{A}(\widehat{\boldsymbol{\eta}}) = \sum_{\ell=1}^{n_{\rm MC}} r_{\rm ar}^\ell \exp\Big\{-\frac{1}{2s_{\rm SB}^2}\|\widehat{\boldsymbol{\eta}}_{\rm ar}^\ell - \widehat{\boldsymbol{\eta}}\|^2\Big\} \quad , \quad \mathcal{B}(\widehat{\boldsymbol{\eta}}) = \sum_{\ell=1}^{n_{\rm MC}} \exp\Big\{-\frac{1}{2s_{\rm SB}^2}\|\widehat{\boldsymbol{\eta}}_{\rm ar}^\ell - \widehat{\boldsymbol{\eta}}\|^2\Big\}. \tag{67}$$

(iii) Estimation of $\boldsymbol{\nabla}\widehat{h}_\rho^c(\widehat{\boldsymbol{\eta}})$. From Eqs. (66) and (67), it can be seen that

$$\boldsymbol{\nabla}_{\widehat{\boldsymbol{\eta}}}\,\widehat{h}_\rho^c(\widehat{\boldsymbol{\eta}}) = \frac{1}{\mathcal{B}(\widehat{\boldsymbol{\eta}})}\big(\boldsymbol{\nabla}_{\widehat{\boldsymbol{\eta}}}\,\mathcal{A}(\widehat{\boldsymbol{\eta}}) - \widehat{h}_\rho^c(\widehat{\boldsymbol{\eta}})\,\boldsymbol{\nabla}_{\widehat{\boldsymbol{\eta}}}\,\mathcal{B}(\widehat{\boldsymbol{\eta}})\big), \tag{68}$$

in which $\boldsymbol{\nabla}_{\widehat{\boldsymbol{\eta}}}\,\mathcal{A}(\widehat{\boldsymbol{\eta}})$ and $\boldsymbol{\nabla}_{\widehat{\boldsymbol{\eta}}}\,\mathcal{B}(\widehat{\boldsymbol{\eta}})$ are given by

$$\boldsymbol{\nabla}_{\widehat{\boldsymbol{\eta}}}\,\mathcal{A}(\widehat{\boldsymbol{\eta}}) = \frac{1}{s_{\rm SB}^2}\sum_{\ell=1}^{n_{\rm MC}} r_{\rm ar}^\ell(\widehat{\boldsymbol{\eta}}_{\rm ar}^\ell - \widehat{\boldsymbol{\eta}})\exp\Big\{-\frac{1}{2s_{\rm SB}^2}\|\widehat{\boldsymbol{\eta}}_{\rm ar}^\ell - \widehat{\boldsymbol{\eta}}\|^2\Big\} \quad , \quad \boldsymbol{\nabla}_{\widehat{\boldsymbol{\eta}}}\,\mathcal{B}(\widehat{\boldsymbol{\eta}}) = \frac{1}{s_{\rm SB}^2}\sum_{\ell=1}^{n_{\rm MC}}(\widehat{\boldsymbol{\eta}}_{\rm ar}^\ell - \widehat{\boldsymbol{\eta}})\exp\Big\{-\frac{1}{2s_{\rm SB}^2}\|\widehat{\boldsymbol{\eta}}_{\rm ar}^\ell - \widehat{\boldsymbol{\eta}}\|^2\Big\}. \tag{69}$$

(iv) Numerical conditioning. Equations (66) to (69) show the presence of exponentials of $-o^\ell(\widehat{\boldsymbol{\eta}})$ with $o^\ell(\widehat{\boldsymbol{\eta}}) = \frac{1}{2s_{\rm SB}^2}\|\widehat{\boldsymbol{\eta}}_{\rm ar}^\ell - \widehat{\boldsymbol{\eta}}\|^2$. If, for a given $\widehat{\boldsymbol{\eta}}$, $o^\ell(\widehat{\boldsymbol{\eta}})$ is large enough for all $\ell = 1,\dots,n_{\rm MC}$, then we can get a value close to $0/0$ or $c/0$ for the computation of $\boldsymbol{\nabla}_{\widehat{\boldsymbol{\eta}}}\,\widehat{h}_\rho^c(\widehat{\boldsymbol{\eta}})$, which is "almost" indeterminate or "almost" in overflow, since $\exp\big\{-o^\ell(\widehat{\boldsymbol{\eta}})\big\}$ can be very small for all ℓ. It can be seen that if $\exp\big\{-o^\ell(\widehat{\boldsymbol{\eta}})\big\}$ is replaced by $\exp\big\{-(o^\ell(\widehat{\boldsymbol{\eta}}) - \underline{e}(\widehat{\boldsymbol{\eta}})\big\}$ in Eqs. (66) to (69) in which $\underline{e}(\widehat{\boldsymbol{\eta}})$ is independent of ℓ, then $\widehat{h}_\rho^c(\widehat{\boldsymbol{\eta}})$ defined by Eq. (66) and $\boldsymbol{\nabla}_{\widehat{\boldsymbol{\eta}}}\,\widehat{h}_\rho^c(\widehat{\boldsymbol{\eta}})$ defined by Eq. (68) are not modified. Therefore, for improving the numerical conditioning, in Eqs. (66) to (69), $\exp\big\{-\frac{1}{2s_{\rm SB}^2}\|\widehat{\boldsymbol{\eta}}_{\rm ar}^\ell - \widehat{\boldsymbol{\eta}}\|^2\big\}$ is replaced by $\exp\big\{-\big(\frac{1}{2s_{\rm SB}^2}\|\widehat{\boldsymbol{\eta}}_{\rm ar}^\ell - \widehat{\boldsymbol{\eta}}\|^2 - \underline{e}(\widehat{\boldsymbol{\eta}})\big)\big\}$ in which $\underline{e}(\widehat{\boldsymbol{\eta}}) = (1/n_{\rm MC})\sum_{\ell=1}^{n_{\rm MC}}\|\widehat{\boldsymbol{\eta}}_{\rm ar}^\ell - \widehat{\boldsymbol{\eta}}\|^2/(2s_{\rm SB}^2)$.

9.2. Estimation of the gradient of $h_{w,i}^c(\boldsymbol{\eta})$ *for* $i = 1,\dots,2n_w$

From Section 7-(ii) and for all $j \in \{1,\dots,n_w\}$, we have $h_{w,j}^c(\mathbf{H}) = W_j^{(\nu)}$ and $h_{w,j+n_w}^c(\mathbf{H}) = (W_j^{(\nu)})^2$. Using Eq. (37) yields, for all $\boldsymbol{\eta} = (\eta_1,\dots,\eta_\nu) \in \mathbb{R}^\nu$,

$$h_{w,j}^c(\boldsymbol{\eta}) = \big\{\underline{\mathbf{x}}_w + [\psi_w]\,[\xi]^{1/2}\,\boldsymbol{\eta}\big\}_j \quad , \quad h_{w,j+n_w}^c(\boldsymbol{\eta}) = \big(\big\{\underline{\mathbf{x}}_w + [\psi_w]\,[\xi]^{1/2}\,\boldsymbol{\eta}\big\}_j\big)^2. \tag{70}$$

It can easily be deduced that, for all $j \in \{1,\dots,n_w\}$ and for all $k \in \{1,\dots,\nu\}$,

$$\frac{\partial h_{w,j}^c(\boldsymbol{\eta})}{\partial \eta_k} = \big\{[\psi_w]\,[\xi]^{1/2}\big\}_{jk} \quad , \quad \frac{\partial h_{w,j+n_w}^c(\boldsymbol{\eta})}{\partial \eta_k} = 2\big(\big\{\underline{\mathbf{x}}_w + [\psi_w]\,[\xi]^{1/2}\,\boldsymbol{\eta}\big\}_j\big)\big\{[\psi_w]\,[\xi]^{1/2}\big\}_{jk}. \tag{71}$$

10. Application 1: nonlinear stochastic dynamical system

This application deals with the nonstationary stochastic response of a nonlinear dynamic system with one degree of freedom (Duffing oscillator) excited by a nonstationary stochastic process and with a random control parameter. This example was deliberately chosen to be very simple in order to quickly show to the reader the role played by the constraints on the stochastic equation during the learning carried out by the PLoM. For this application, we will not give the algebraic complements necessary for its implementation and we will not do the detailed analysis. These aspects will be presented for Applications 2 and 3.

10.1. Nonlinear stochastic equation

(i) Nonlinear stochastic dynamics equation with initial conditions. We consider the Duffing oscillator with $N = 1$,

$$\ddot{Y}(t) + 2\,\chi_d\, g_1(W_1)\,\dot{Y}(t) + g_1(W_1)^2\big(1 + k_b\, Y(t)^2\big)\,Y(t) = \gamma(t, W_2) \quad , \quad t \in]0, T]\,,$$
$$Y(0) = \dot{Y}(0) = 0 \quad a.s.\,,$$

in which $\mathbf{W} = (W_1, W_2)$ is the $\mathbb{R}^2$-valued random control parameter ($n_w = 2$) defined on the probability space $(\Theta, \mathcal{T}, \mathcal{P})$ such that W_1 and W_2 are independent normalized Gaussian random variables, where $\chi_d = 0.05$, $k_b = 5 \times 10^8$, $T = 0.7325\ s$, and where $Y = \big\{Y(t), t \in [0, T]\big\}$ is the unknown real-valued stochastic process defined on $(\Theta, \mathcal{T}, \mathcal{P})$.

(ii) Nonstationary stochastic excitation and random control parameters. The stochastic excitation is defined by

$$\gamma(t, W_2) = \Big(\frac{t}{T}\, g_2(W_2)\Big)^2 \,\big(1 + 0.05\,\sin(\omega_b\, t)\big)\,\exp\left\{ -\frac{1}{1 - (\frac{2t}{T} - 1)^4} \right\} \quad , \quad \forall\, t \geq 0\,,$$

in which $\omega_b = 2\pi \times 100\, rad/s$. For $j = 1, 2$, we have $g_j(W_j) = \underline{g}_j \big(1 + \sqrt{3}\,\delta_j(2\,\mathcal{U}_j - 1)\big)$ in which $\mathcal{U}_j = \big(1 + \mathrm{erf}(W_j/\sqrt{2})\big)/2$, with $\delta_j = 0.2$ and where $\underline{g}_j = E\big\{g_j(W_j)\big\}$ with $\underline{g}_1 = \omega_b$ and $\underline{g}_2 = 6$.

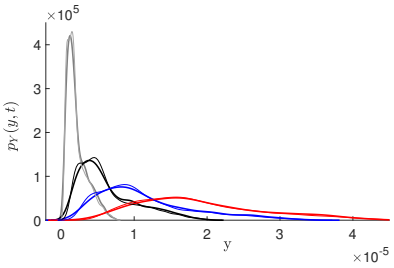
(iii) Values of the main parameters. The time sampling is done with $n_{\text{time}} = 2\,931$ sampling times for which the time step is $\Delta t = 2.5 \times 10^{-4}\ s$. There is no time subsampling, that is to say, $n_{\text{sp}} = n_{\text{time}}$. The number of points in the training dataset is $N_d = 80$, the KL-expansion of stochastic process Y has been performed with $\varepsilon_{\text{KL}} = 10^{-6}$ yielding $n_q = 25$. The PCA of random vector $\mathbf{X}$ of dimension $n_x = n_q + n_w = 27$ has been performed with $\varepsilon_{\text{PCA}} = 10^{-6}$ yielding $\nu = 27$. For $\varepsilon_{\text{diff}} = 63$, the dimension of the diffusion-maps basis is $m = 28$. In the ISDE, $f_0 = 4$ and the parameters for the Störmer-Verlet algorithm (Appendix A) are $\Delta r = 0.19914$, $l_0 = 100$, and $M_0 = 20$.

10.2. Reference stochastic solution

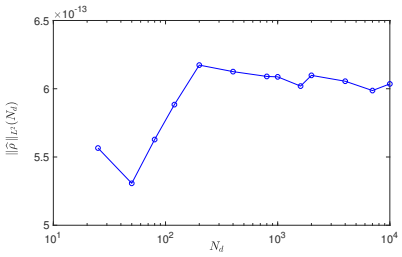
Figure 1 is related to the reference solution that is constructed using a training dataset with $N_d = 1\,000$ points. Figure 1-(a) displays the time evolution of the pdf $y \mapsto p_Y(y, t)$ of random variable $Y(t)$ for $t = 0.1785$, 0.2977, 0.3960, and $0.5535\ s$, which are estimated with $N_d = 1\,000$ and $10\,000$ points in the training dataset. It can be seen that convergence with respect to N_d is obtained for $N_d = 1\,000$ (the curves for $10\,000$ are very close to those for $1\,000$). The convergence of the reference solution constructed with $N_d = 1\,000$ can also be viewed in Fig. 1-(b) that shows the graph of function $N_d \mapsto \|\widehat{\rho}\,\|_{L^2}(N_d)$ of the random residual $\widehat{\rho}$. The values of $\|\widehat{\rho}\,\|_{L^2}(N_d)$ are close for $N_d = 1\,000$ and $N_d = 10\,000$.

10.3. Analysis of the role played by theconstraints on the probability distribution of $\mathbf{W}$

For simplifying the writing, $\mathbf{W}^{(\nu)}$ is simply noted $\mathbf{W}$. Figure 2 allows for analyzing the role played by the constraints on the probability distribution of $\mathbf{W}$ during the minimization of the normalized random residual. The computation is performed with $N_d = 80$ points in the training dataset and $n_{\text{MC}} = 1\,000$ additional realizations for the PLoM under constraints. Figure 2-(a) displays the graph of error function $i \mapsto \mathrm{err}_W(i)$ (see Eq. (47)) computed with *algo* 1 of the PLoM under the constraint on ρ but without the constraints on $\mathbf{W}$. At iteration $i_{\text{opt}} = 10$, the relative error $\mathrm{err}_R(i_{\text{opt}})$ (see Eq. (47)) is minimum while $i \mapsto \mathrm{err}_W(i)$ (see Eq. (47)) is increasing from $i = 5$ and does not reach a minimum for $i_{\text{opt}} = 10$. This phenomenon can also be seen of Figure 2-(b) that displays the pdf $w_2 \mapsto p_{W_2}(w_2)$ of W_2 estimated



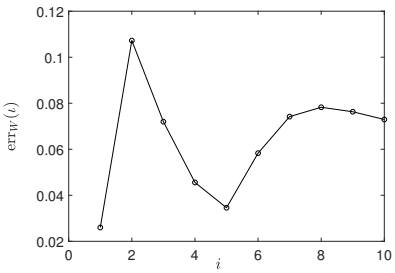
(a) pdf of random variable $Y(t)$ as a function of t



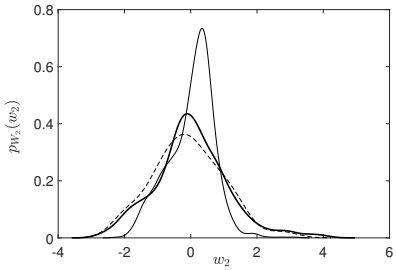
(b) L^2-norm of the random residue $\widehat{\rho}$

Figure 1: (a) time evolution of the pdf $y \mapsto p_Y(y,t)$ for $t = 0.1785\,s$ (green), $t = 0.2977\,s$ (black), $t = 0.3960\,s$ (blue), and $t = 0.5535\,s$ (red) estimated with $N_d = 1\,000$ points (thick line) and with $N_d = 10\,000$ points (thin line) in the training dataset; (b) graph of $N_d \mapsto \|\widehat{\rho}\|_{L^2}(N_d)$.

with the training dataset, with *algo* 1 of the PLoM under the constraint on ρ but without the constraints on $\mathbf{W}$, and with *algo* 3 of the PLoM under the constraints on ρ and $\mathbf{W}$. It can be seen that *algo* 3 allows for obtaining a pdf of W_2, which is close to the pdf that corresponds to the training dataset.



(a) relative error function for $\mathbf{W}$



(b) pdf of W_2

Figure 2: For $N_d = 80$ and $n_{\mathrm{MC}} = 1\,000$: (a) relative error function $i \mapsto \mathrm{err}_W(i)$ computed with *algo* 1 of the PLoM under the constraint on ρ but without the constraints on $\mathbf{W}$; (b) pdf $w_2 \mapsto p_{W_2}(w_2)$ of W_2 estimated with the training dataset (dashed line), with *algo* 1 of the PLoM under the constraint on ρ but without the constraints on $\mathbf{W}$ (thin solid line), and with *algo* 3 of the PLoM under the constraints on ρ and $\mathbf{W}$ (thick solid line).

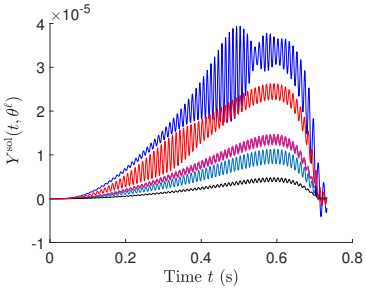
10.4. Results of the PLoM without and under constraints for a given dimension of the training dataset

In this section, the computation is performed with $N_d = 80$ points in the training dataset and $n_{\mathrm{MC}} = 1\,000$ additional realizations for the PLoM under constraints. In order to visualize the type of sample path of stochastic process Y^{sol}, Fig. 3-(a) displays the graph of 5 realizations $\mathbf{y}^\ell = Y^{\mathrm{sol}}(\cdot,\theta^\ell)$ of stochastic process Y^{sol}. Fig. 3-(b) displays $n_{\mathrm{MC}} = 1\,000$ additional realizations of components (W_1^c, W_2^c, Q_2^c) of random vector $\mathbf{X}^c = (\mathbf{Q}^c, \mathbf{W}^c)$. A concentration of these realizations can be observed. Concerning the pdf $y \mapsto p_Y(y,t)$ of random variable $Y(t)$, Fig. 4 shows, for the two instants $t = 0.2977\,s$ and $t = 0.5535\,s$, the reference constructed with $N_d = 1\,000$ and, for $N_d = 80$ points and $n_{\mathrm{MC}} = 1\,000$ additional realizations, the estimation constructed with the PLoM without constraint and the one with the PLoM under constraints using *algo* 3 for which the minimum is obtained for $i_{\mathrm{opt}} = 8$. This figure already shows that the PLoM without constraint gives good results but that the PLoM under constraints significantly improves the result. In particular, it can be seen in figures (c) and (d), which are in a logarithmic scale for the ordinates, that the PLoM under constraints makes it possible to correctly predict the tails of the probability density function of the reference.

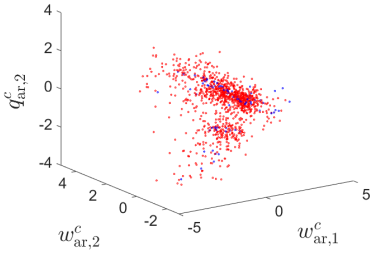
11. Application 2: unsteady 2D Navier-Stokes equation for incompressible fluid

11.1. Boundary value problem

We consider a dimensionless 2D formulation in a Cartesian coordinate system (O, x_1, x_2), whose generic point is $\mathbf{x} = (x_1, x_2)$. The 2D open domain containing the fluid is $\Omega =]0, \ell_{x_1}[\times]0, \ell_{x_2}[$ with $\ell_{x_1} = 0.9$ and $\ell_{x_2} = 1$.

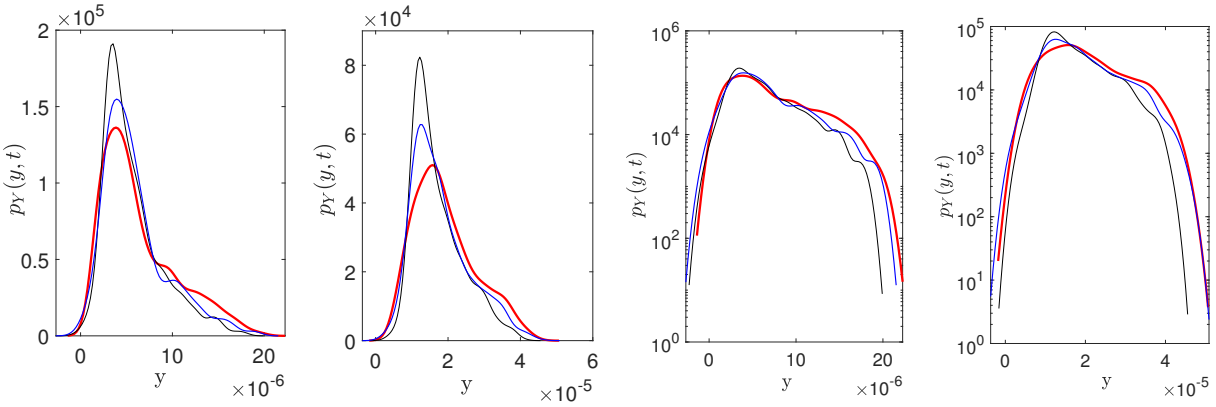


(a) Example of 5 realizations of Y^{sol}



(b) PLoM under constraints

Figure 3: (a) Example of 5 realizations $\mathbf{y}^\ell$ of Y^{sol}. (b) 3D-plot of realizations $(w^c_{\text{ar},1}, w^c_{\text{ar},2}, q^c_{\text{ar},2})$ of (W^c_1, W^c_2, Q^c_2) computed with the PLoM under the constraints: $n_{\text{MC}} = 1\,000$ additional realizations (red symbols) and $N_d = 80$ points from the training dataset (blue symbols).



(a) $y \mapsto p_Y(y,t)$ at $t = 0.29775\,s$ (b) $y \mapsto p_Y(y,t)$ at $t = 0.5535\,s$ (c) $y \mapsto \log_{10} p_Y(y,t)$ at $t = 0.29775\,s$ (d) $y \mapsto \log_{10} p_Y(y,t)$ at $t = 0.5535\,s$

Figure 4: For $N_d = 80$ and $n_{\text{MC}} = 1\,000$: pdf $y \mapsto p_Y(y,t)$ (a)-(b) and $y \mapsto \log_{10} p_Y(y,t)$ (c)-(d) for $t = 0.2977\,s$ (a)-(c) and for $t = 0.5535\,s$ (b)-(d), corresponding to the reference (red thick lines), to the PLoM without constraint (thin black lines), and to the PLoM under constraints (blue med line).

The boundary is written as $\partial\Omega = \Gamma_0 \cup \Gamma_u$ in which $\Gamma_u = \{x_1 \in [0, \ell_{x_1}], x_2 = \ell_{x_2}\}$ and where $\Gamma_0 = \partial\Omega \backslash \Gamma_u$. Let $\mathbf{u}(\mathbf{x},t) = (u_1(\mathbf{x},t), u_2(\mathbf{x},t))$ be the dimensionless velocity field and $p(\mathbf{x},t)$ be the dimensionless pressure field, in which t is the dimensionless time. These quantities are related to the non dimensionless quantities (denoted with a tilde) as follows: $\mathbf{x} = \widetilde{\mathbf{x}}/\widetilde{\ell}_{\text{ref}}$, $t = \widetilde{t}\,\widetilde{V}_{\text{ref}}/\widetilde{\ell}_{\text{ref}}$ in which $\widetilde{\ell}_{\text{ref}}$ is a given reference length and where $\widetilde{V}_{\text{ref}}$ is the reference velocity in m/s (independent of $\mathbf{x}$ and t), $\mathbf{u} = \widetilde{\mathbf{u}}/\widetilde{V}_{\text{ref}}$, and $p = \widetilde{p}/(\widetilde{\rho}_{\text{ref}}\widetilde{V}^2_{\text{ref}})$ with $\widetilde{\rho}_{\text{ref}}$ the mass density in kg/m^3 (independent of $\mathbf{x}$ and t). Let $\mathcal{R}e = \widetilde{V}_{\text{ref}}\widetilde{\ell}_{\text{ref}}/\nu_{\text{CFD}}$ be the Reynolds number in which ν_{CFD} is the kinematic viscosity such that $\nu_{\text{CFD}} = \widetilde{\eta}_{\text{ref}}/\widetilde{\rho}_{\text{ref}}$ where $\widetilde{\eta}_{\text{ref}}$ is the dynamic viscosity in $Pa \times s$. It is assumed that ν_{CFD} is spatially uniform (independent of $\mathbf{x}$ and t) and consequently, the energy equation is not required. The boundary conditions on the non dimensionless boundary $\partial\widetilde{\Omega} = \widetilde{\Gamma}_0 \cup \widetilde{\Gamma}_u$ are written, for all t in $[t_0, T]$, as $\widetilde{\mathbf{u}}(\cdot,t) = (0,0)$ on $\widetilde{\Gamma}_0$ and $\widetilde{\mathbf{u}}(\cdot,t) = (\widetilde{V}_{\text{CFD}}, 0)$ on $\widetilde{\Gamma}_u$ where $\widetilde{V}_{\text{CFD}}$ in m/s is written as $\widetilde{V}_{\text{CFD}} = V_{\text{CFD}} \times \widetilde{V}_{\text{ref}}$ in which V_{CFD} is the given dimensionless velocity that is assumed independent of $\mathbf{x}$ and t. The dimensionless boundary value problem for the unsteady incompressible flow in Ω is defined as follows. For all $t \in]t_0, T]$ and for all $\mathbf{x} \in \Omega$, we have the following dimensionless 2D Navier-Stokes equations for an incompressible flow,

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \boldsymbol{\nabla}\mathbf{u} + \boldsymbol{\nabla} p - \frac{1}{\mathcal{R}e}\boldsymbol{\nabla}^2\mathbf{u} = \mathbf{0}\,, \tag{72}$$

$$\boldsymbol{\nabla} \cdot \mathbf{u} = 0\,, \tag{73}$$

with the boundary conditions that are written for all $t \in [t_0, T]$,

$$\mathbf{u}(\mathbf{x}, t) = (0, 0)\,,\ \mathbf{x} \in \Gamma_0 \quad , \quad \mathbf{u}(\mathbf{x}, t) = (V_{\text{CFD}}, 0)\,,\ \mathbf{x} \in \Gamma_u\,, \tag{74}$$

and with the following initial condition at time t_0, $\mathbf{u}(\mathbf{x}, t_0) = (0, 0)$ for all $\mathbf{x} \in \Omega$. We choose $\widetilde{\ell}_{\text{ref}} = 0.9\ m$ and $\widetilde{V}_{\text{ref}} = \widetilde{\nu}\, V_{\text{CFD}}$ with $\widetilde{\nu} = 1\ m/s$. Consequently, the Reynolds can be rewritten as $\mathcal{R}e = 0.9 V_{\text{CFD}}/\nu_{\text{CFD}}$.

11.2. Prior probability model of uncertainties

Parametric probabilistic uncertainties and their quantification in computational fluid dynamics have extensively been studied (see for instance [64, 65]). The present application is developed to validate the PLoM under constraints. We must therefore introduce uncertainties and we therefore choose a parametric probabilistic approach. The uncertain control parameters are V_{CFD} and ν_{CFD}. They are modeled by two independent positive-valued random variables, defined on the probability space $(\Theta, \mathcal{T}, \mathcal{P})$, with values in $[V_{\min}, V_{\max}]$ and $[\nu_{\min}, \nu_{\max}]$, written as $V_{\text{CFD}} = V_{\min}(1 + \varepsilon_V\, \mathcal{U}_1)$ and $\nu_{\text{CFD}} = \nu_{\min}(1 + \varepsilon_\nu\, \mathcal{U}_2)$, in which $\varepsilon_V = V_{\max}/V_{\min} - 1$ and $\varepsilon_\nu = \nu_{\max}/\nu_{\min} - 1$, and where $\mathcal{U}_1$ and $\mathcal{U}_2$ are independent uniform random variables on $[0, 1]$. The considered numerical values are $V_{\min} = 0.1\ m/s$, $V_{\max} = 0.3\ m/s$, $\nu_{\min} = 10^{-5}$, and $\nu_{\max} = 3 \times 10^{-5}$.

Consequently, the support of the probability distribution of the random Reynolds number $\mathcal{R}e$ is the interval $[3\,000, 27\,000]$. The uniform random variables are classically written as $\mathcal{U}_j = \left(1 + \text{erf}(W_j/\sqrt{2})\right)/2$, in which W_1 and W_2 are independent normalized Gaussian random variables. Let $\mathbf{W} = (W_1, W_2)$ be the $\mathbb{R}^{n_w}$- valued random variable with $n_w = 2$. For $j = 1, 2$, let $\left\{\mathbf{w}_{d,j}^{\ell}, \ell = 1, \ldots, N_d\right\}$ be N_d realizations generated with the prior probability model for constituting the training dataset. The random Reynolds number $\mathcal{R}e$ is defined by a given real-valued mapping $f_{\mathcal{R}e}$ on $\mathbb{R}^2$ such that

$$\mathcal{R}e = f_{\mathcal{R}e}(\mathbf{W})\,.$$

11.3. Computational model, KL expansion, random residual, and PLoM parameters

(i) Time discretization scheme. We use the time discretization scheme proposed in [66]. We choose $t_0 = 0$. Let Δt be the time step, $\left\{t_n = n\,\Delta t,\ n = 1, \ldots n_{\text{time}}\right\}$ be the time sampling, and $T = n_{\text{time}}\,\Delta t$ be the final time. Let us assumed that the velocity field $\mathbf{x} \mapsto \mathbf{u}^{(n)}(\mathbf{x}) = \mathbf{u}(\mathbf{x}, t_n)$ and the pressure field $\mathbf{x} \mapsto p^{(n)}(\mathbf{x}) = p(\mathbf{x}, t_n)$ are known at the n-th sampling time and satisfy Eqs. (72) and (73). The following scheme allows for calculating the approximation of the solution at time t_{n+1} (below, the $\mathbf{x}$ dependence is removed). In Eq. (72), the partial time derivative of $\mathbf{u}$ at time t_{n+1}, which is discretized as $(\mathbf{u}^{(n+1)} - \mathbf{u}^{(n)})/\Delta t$, is written as the sum of three terms: the nonlinear force that is explicitly treated, the viscosity force that is implicitly treated, and the pressure correction term that is implicitly treated for enforcing incompressibility defined by to Eq. (73),

$$\frac{\mathbf{u}^{\text{NL}} - \mathbf{u}^{(n)}}{\Delta t} = -(\mathbf{u} \cdot \boldsymbol{\nabla}\mathbf{u})^{(n)} \quad , \quad \frac{\mathbf{u}^{\text{V}} - \mathbf{u}^{\text{NL}}}{\Delta t} = \frac{1}{\mathcal{R}e}(-\boldsymbol{\nabla}^2\mathbf{u}^{\text{V}}) \quad , \quad \frac{\mathbf{u}^{(n+1)} - \mathbf{u}^{\text{V}}}{\Delta t} = -\boldsymbol{\nabla} p^{(n+1)} \quad , \quad \boldsymbol{\nabla} \cdot \mathbf{u}^{(n+1)} = 0\,. \tag{75}$$

The implicit equation for the calculation of $p^{(n+1)}$ is obtained by substituting $\mathbf{u}^{(n+1)}$ given by the third Eq. (75) into the fourth and gives the elliptic equation,

$$-\boldsymbol{\nabla}^2(\Delta t\, p^{(n+1)}) = -\boldsymbol{\nabla} \cdot \mathbf{u}^{\text{V}}\,. \tag{76}$$

(ii) Spatial discretization. The spatial discretization, the discretization of the spatial derivatives, and the implementation of the boundary conditions, are performed as explained in [66]. A grid is defined using n_{x_1} grid points in the x_1-axis with a spatial step $h_{x_1} = \ell_{x_1}/n_{x_1}$ and n_{x_2} grid points in the x_2-axis with $h_{x_2} = \ell_{x_2}/n_{x_2}$. There are $n_{x_1} \times n_{x_2}$ cells. The spatial discretization is performed on a staggered grid for which u_1 is placed on the x_2 (vertical) cell interfaces, u_2 is placed on the x_1 (horizontal) cell interfaces, and p is placed in the cell midpoints. We then have,

- for the interior points, $N_{u_1} = (n_{x_1} - 1) \times n_{x_2}$ points for velocity component u_1, $N_{u_2} = n_{x_1} \times (n_{x_2} - 1)$ points for velocity component u_2, and $N_p = n_{x_1} \times n_{x_2}$ points for pressure p.
- for the total number of points allowing the boundary conditions to be implemented, $(n_{x_1} + 1) \times (n_{x_2} + 2)$ points for velocity component u_1, $(n_{x_1} + 2) \times (n_{x_2} + 1)$ points for velocity component u_2, and $(n_{x_1} + 2) \times (n_{x_2} + 2)$ points for pressure p.

This staggered grid allows the boundary conditions to be easily implemented. Let be $N_u = N_{u_1} + N_{u_2}$ and $N = N_u + N_p$. At time t, the spatial discretization of random fields $\mathbf{u}(\cdot, t) = \big(u_1(\cdot, t), u_2(\cdot, t)\big)$ and $p(\cdot, t)$ are the $\mathbb{R}^{N_u}$- and $\mathbb{R}^{N_p}$-valued random variables $\mathbf{U}(t)$ and $\mathbf{P}(t)$, and we define the $\mathbb{R}^N$-valued random variable $\mathbf{Y}(t)$ by

$$\mathbf{Y}(t) = \big(\mathbf{U}(t), \mathbf{P}(t)\big) \,. \tag{77}$$

(iii) Computational model. The time discretization scheme and the spatial discretization allow for computing the time-spatial-discretization $\mathbf{Y}(t_n) = \big(\mathbf{U}(t_n), \mathbf{P}(t_n)\big)$, and the time derivative $\dot{\mathbf{U}}(t_n)$, of the solution of the boundary value problem, at time t_n, for all n in $\{1, \ldots, n_{\text{time}}\}$. In that condition, Eq. (1) is rewritten, at sampling time t_n, as

$$\boldsymbol{\mathcal{N}}^{\text{NS}}\big(\dot{\mathbf{U}}(t_n), \mathbf{U}(t_n), \mathbf{P}(t_n), \mathbf{W}\big) = \mathbf{0}_{N_u} \quad a.s., \tag{78}$$

$$\boldsymbol{\mathcal{N}}^{\text{DIV}}\big(\mathbf{U}(t_n)\big) = \mathbf{0}_{N_p} \quad a.s., \tag{79}$$

in which $\boldsymbol{\mathcal{N}}^{\text{NS}}$ is a nonlinear operator and where $\boldsymbol{\mathcal{N}}^{\text{DIV}}$ is a linear operator.

(iv) Values of the parameters for the Navier-Stokes solver. The staggered grid is such that $n_{x_1} = 100$ and $n_{x_2} = 111$. We then have $N_{u_1} = 10\,989$, $N_{u_2} = 11\,000$, $N_u = 21\,989$, $N_p = 11\,100$, and $N = 33\,089$. The time step has been chosen to $\Delta t = 0.03$ and leads to a convergent solution up to the steady (stationary) stochastic flow with a L^2-norm of the residual (level of approximation) that is acceptable (see Section 11.4). Clearly, this approximation could be improved in decreasing the value of Δt but there is no point in unnecessarily increasing the computational costs, because that does not change anything on the validation of the PLoM under constraints. (i) For computation of the steady (time-stationary) stochastic flow (see Section 11.4-(i)), $T = 300$ and $n_{\text{time}} = 10\,000$. For computing the random residual, $n_{\text{sp}} = 500$. (ii) For computation of the reference unsteady (time-nonstationary) stochastic flow (see Section 11.4-(ii)), $t = 30$ is chosen for analyzing the performance of the proposed PLoM method under constraints, which is an instant corresponding to the unsteady flow regime. We then have $T = 30$ and $n_{\text{time}} = 1\,000$.

(v) Reduced-order representations (KL expansions) of the nonstationary stochastic processes $\mathbf{Y} = (\mathbf{U}, \mathbf{P})$ *and* $\dot{\mathbf{U}}$. The KL expansion of the nonstationary $\mathbb{R}^N$-valued stochastic process $\big\{\mathbf{Y}(t), t \in [t_0, T]\big\}$ is performed at sampling times $t_1, \ldots, t_{n_{\text{time}}}$, yielding the representation $\mathbf{Y}(t_n) = \underline{\mathbf{y}}(t_n) + [V(t_n)]\,\mathbf{Q}$. Since $\mathbf{Y} = (\mathbf{U}, \mathbf{P})$, we have

$$\mathbf{U}(t_n) = \underline{\mathbf{u}}(t_n) + [V_u(t_n)]\,\mathbf{Q}\,, \tag{80}$$

$$\mathbf{P}(t_n) = \underline{\mathbf{p}}(t_n) + [V_p(t_n)]\,\mathbf{Q}\,, \tag{81}$$

in which $\underline{\mathbf{u}}(t_n) \in \mathbb{R}^{N_u}$, $[V_u(t_n)] \in \mathbb{M}_{N_u, n_q}$, $\underline{\mathbf{p}}(t_n) \in \mathbb{R}^{N_p}$, and $[V_p(t_n)] \in \mathbb{M}_{N_p, n_q}$. The time derivative of $\mathbf{U}(t)$ at t_n is calculated by $\dot{\mathbf{U}}(t_n) = (\mathbf{U}(t_n) - \mathbf{U}(t_{n-1}))/\Delta t$, which allows for deducing $\underline{\dot{\mathbf{u}}}(t_n)$ and $[\dot{V}_u(t_n)]$. The KL expansion of $\dot{\mathbf{U}}$ can then be written as,

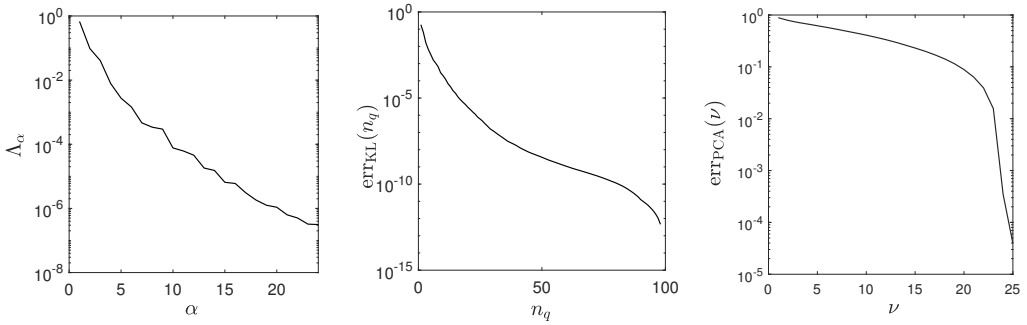
$$\dot{\mathbf{U}}(t_n) = \underline{\dot{\mathbf{u}}}(t_n) + [\dot{V}_u(t_n)]\,\mathbf{Q}\,. \tag{82}$$

For $N_d = 100$ and $T = 30$, Figs. 5-(a) and -(b) show the distribution of the eigenvalues $\alpha \mapsto \Lambda_\alpha$ of the KL-expansion of stochastic process $\big\{\mathbf{Y}(t) = \big(\mathbf{U}(t), \mathbf{P}(t)\big), t \in [t_0, T]\big\}$ and the error function $n_q \mapsto \text{err}_{\text{KL}}(n_q)$ defined by Eq. (15). Fixing the tolerance ε_{KL} to the value 10^{-6} yields the optimal value $n_q = 24$.

(vi) Calculation of the random residual $\widehat{\rho}$ *using the subsampling times*. The subsampling times $\{\tau_{n_s}, n_s = 1, \ldots, n_{\text{sp}}\}$ (see Section 4.2) is built with a constant step $\Delta\tau = (T - t_0)/n_{\text{sp}}$ and $n_{\text{sp}} = 50$. The family $\big\{\dot{\mathbf{U}}(\tau_{n_s}), \mathbf{U}(\tau_{n_s}), \mathbf{P}(\tau_{n_s}), n_s = 1, \ldots, n_{\text{sp}}\big\}$ is extracted from the family $\big\{\dot{\mathbf{U}}(t_n), \mathbf{U}(t_n), \mathbf{P}(t_n), n = 1, \ldots, n_{\text{time}}\big\}$ computed with Eqs. (80) to (82). Taking into account Eqs. (78) and (79) at subsampling time τ_{n_s}, the $\mathbb{R}^N$-valued random residual $\boldsymbol{\mathcal{R}}(\tau_{n_s})$, defined by Eq. (17), is rewritten as $\boldsymbol{\mathcal{R}}(\tau_{n_s}) = \big(\boldsymbol{\mathcal{R}}^{\text{NS}}(\tau_{n_s}), \boldsymbol{\mathcal{R}}^{\text{DIV}}(\tau_{n_s})\big)$ in which the $\mathbb{R}^{N_u}$-valued random variable $\boldsymbol{\mathcal{R}}^{\text{NS}}(\tau_{n_s})$ and the $\mathbb{R}^{N_p}$-valued random variable $\boldsymbol{\mathcal{R}}^{\text{DIV}}(\tau_{n_s})$ are written, for all $n_s \in \{1, \ldots, n_{\text{sp}}\}$, as

$$\boldsymbol{\mathcal{R}}^{\text{NS}}(\tau_{n_s}) = \boldsymbol{\mathcal{N}}^{\text{NS}}(\dot{\mathbf{U}}(\tau_{n_s}), \mathbf{U}(\tau_{n_s}), \mathbf{P}(\tau_{n_s}), \mathbf{W}) \quad a.s., \tag{83}$$

$$\boldsymbol{\mathcal{R}}^{\text{DIV}}(\tau_{n_s}) = \boldsymbol{\mathcal{N}}^{\text{DIV}}(\mathbf{U}(\tau_{n_s})) \quad a.s.. \tag{84}$$



(a) Eigenvalues Λ of the K-L expansion (b) Error induced by K-L expansion (c) Error induced by PCA

Figure 5: For $N_d = 100$, $T = 30$, and $n_{\text{time}} = 1\,000$, KL expansion of nonstationary stochastic process $\mathbf{Y}$: (a) distribution of the eigenvalues $\alpha \mapsto \Lambda_\alpha$ and (b) error function $n_q \mapsto \mathrm{err}_{\mathrm{KL}}(n_q)$. PCA of random vector $\mathbf{X}$: (c) error function $\nu \mapsto \mathrm{err}_{\mathrm{PCA}}(\nu)$.

Let $\boldsymbol{\mathcal{R}}^{\mathrm{NS}}_{u_1}(\tau_{n_s})$ and $\boldsymbol{\mathcal{R}}^{\mathrm{NS}}_{u_2}(\tau_{n_s})$ with values in $\mathbb{R}^{N_{u_1}}$ and $\mathbb{R}^{N_{u_2}}$ be the block decomposition of $\boldsymbol{\mathcal{R}}^{\mathrm{NS}}(\tau_{n_s})$ associated with the components relative to the x_1- and x_2-axes. We then have $\boldsymbol{\mathcal{R}}^{\mathrm{NS}}(\tau_{n_s}) = \big(\boldsymbol{\mathcal{R}}^{\mathrm{NS}}_{u_1}(\tau_{n_s}), \boldsymbol{\mathcal{R}}^{\mathrm{NS}}_{u_2}(\tau_{n_s})\big)$ with values in $\mathbb{R}^{N_u} = \mathbb{R}^{N_{u_1}} \times \mathbb{R}^{N_{u_2}}$. The positive-valued random variable $\mathcal{R}_{\mathrm{norm}}(\tau_{n_s})$ is defined by Eq. (19), but the Euclidean norm is adapted to the structure of random vector $\boldsymbol{\mathcal{R}}(\tau_{n_s})$ as

$$\mathcal{R}_{\mathrm{norm}}(\tau_{n_s}) = \Big\{ \|\boldsymbol{\mathcal{R}}^{\mathrm{NS}}_{u_1}(\tau_{n_s})\|^2 / N^2_{u_1} + \|\boldsymbol{\mathcal{R}}^{\mathrm{NS}}_{u_2}(\tau_{n_s})\|^2 / N^2_{u_2} + \|\boldsymbol{\mathcal{R}}^{\mathrm{DIV}}(\tau_{n_s})\|^2 / N^2_{p} \Big\}^{1/2} . \tag{85}$$

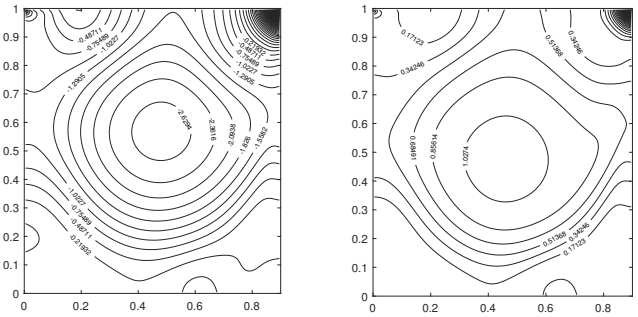
The random residual $\widehat{\rho}$ is then calculated by using Eq. (24). The mean value $\widehat{\underline{\rho}}_{\mathrm{ref}}$ of the reference random residual $\widehat{\rho}_{\mathrm{ref}}$ is calculated for the fixed value $N_{d,\mathrm{ref}} = 100$ of the number N_d of training points in the training dataset.

(vii) Values of the parameters for the PLoM without and under constraints. We consider the case $N_d = 100$ and $T = 30$ for which the KL expansions of stochastic processes $\dot{\mathbf{U}}$, $\mathbf{U}$, and $\mathbf{P}$ have been performed. Random vector $\mathbf{X} = (\mathbf{Q}, \mathbf{W})$ has dimension $n_x = n_q + n_w = 26$. Concerning the PCA of $\mathbf{X}$, Figure 5-(c) shows the error function $\nu \mapsto \mathrm{err}_{\mathrm{PCA}}(\nu)$ defined by Eq. (27). Fixing the tolerance $\varepsilon_{\mathrm{PCA}}$ to the value 10^{-6} yields the optimal value $\nu = 26$. For the PLoM without constraint or under constraints, the diffusion-maps basis is computed with $\varepsilon_{\mathrm{diff}} = 95$, which yields the optimal dimension $m = 27$. Concerning the Störmer-Verlet algorithm for solving the nonlinear ISDE: $f_0 = 4$ and $\Delta r = 0.1974$. The convergence analysis with respect to n_{MC} is performed by considering the following values 100, 400, 800, 1 200, 1 600, 2 000, and 4 000 of n_{MC}.

11.4. Steady and unsteady stochastic flow

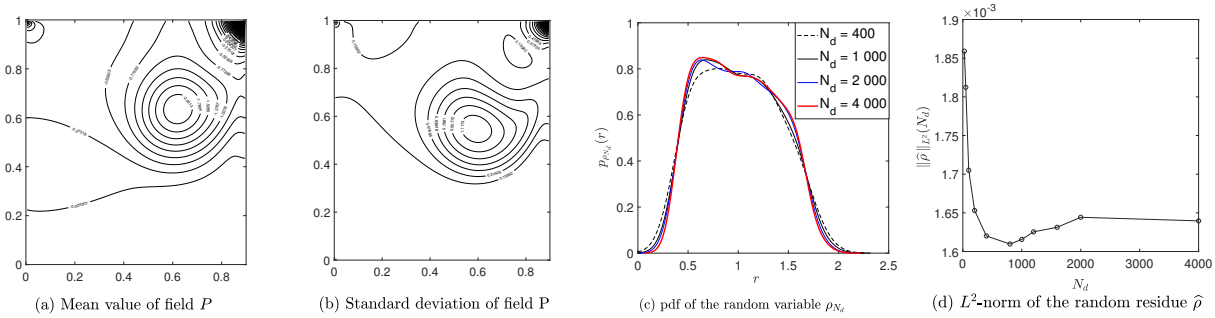
(i)- Steady (time-stationary) stochastic flow. For all the possible values of random control parameters $\nu_{\mathrm{CFD}} \in [1\,,3] \times 10^{-5}$ and $V_{\mathrm{ref}} \in [0.1\,,0.3]\, m/s$ of the prior probability model, the steady flow (time-stationary regime of the stochastic flow) has been obtained at $t = 300$ (almost-sure stochastic solution of the Navier-Stokes equations). For $t \geq 300$, the stochastic solution is stationary in time. The mean value $m_P(\mathbf{x}, t)$ and the standard deviation $\sigma_P(\mathbf{x}, t)$ of the pressure random field $\big\{P(\mathbf{x}, t), \mathbf{x} \in \Omega, t \geq 300\big\}$ are independent of t. For $N_d = 50$ realizations of the training dataset, Fig. 6 shows the estimates of the mean value and standard deviation of the pressure random field at $t = 300$.

(ii)- Reference unsteady (time-nonstationary) stochastic flow. Instant $t = 30$ is selected for analyzing the performance of the proposed method in the unsteady flow regime. A reference unsteady solution has been computed with $N_d = 4\,000$ realizations from the training dataset requiring a significant CPU effort. This reference solution is used for analyzing all the convergence aspects of the PLoM without and under constraints. Figures 7-(a) and (b) show the estimates of the mean value and standard deviation of the pressure random field at $t = 30$. The estimate of $E\big\{\widehat{\rho}\big\}$ is 1.53×10^{-3} while the estimate of $\|\widehat{\rho}\|_{L^2}$ is 1.64×10^{-3}. Figure 7-(c) displays the graphs of the pdf $r \mapsto p_{\rho_{N_d}}(r)$ of the random variable $\rho_{N_d} = \widehat{\rho}/\widehat{\underline{\rho}}_{N_d}$ in which $\widehat{\underline{\rho}}_{N_d} = E\big\{\widehat{\rho}\big\}$ is estimated using N_d realizations. Therefore, $E\big\{\rho_{N_d}\big\} = 1$. For $N_d = 400$, 1 000, 2 000, and 4 000, the standard deviation of ρ_{N_d} is equal to 0.396, 0.383, 0.379, and 0.377,



(a) Mean value (b) Standard deviation

Figure 6: Mean value $\mathbf{x} \mapsto 1\,000 \times m_P(\mathbf{x}, t)$ (a) and standard deviation $\mathbf{x} \mapsto 1\,000 \times \sigma_P(\mathbf{x}, t)$ (b) estimated with $N_d = 50$ realizations of the training dataset, at $t = 300$ of the time-stationary pressure random field $\{P(\mathbf{x}, t), \mathbf{x} \in \Omega, t \geq 300\}$.



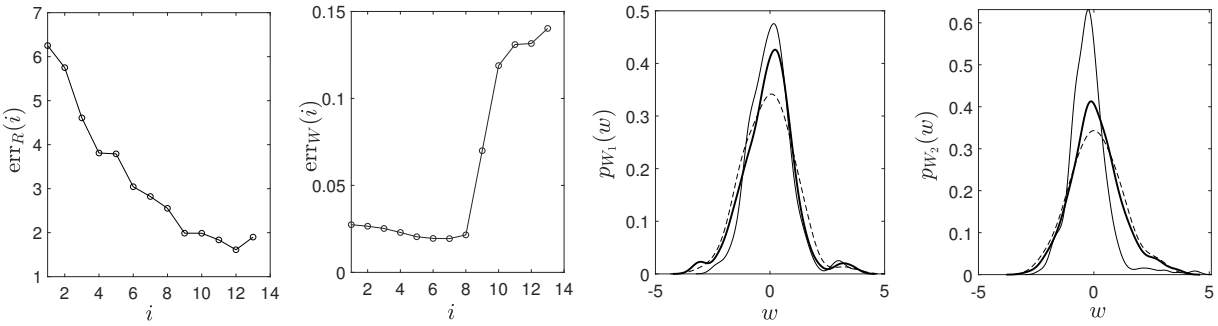
(a) Mean value of field P (b) Standard deviation of field P (c) pdf of the random variable ρ_{N_d} (d) L^2-norm of the random residue $\widehat{\rho}$

Figure 7: Reference unsteady stochastic flow at $t = 30$: (a) mean value $\mathbf{x} \mapsto 1\,000 \times m_P(\mathbf{x}, t)$; (b) standard deviation $\mathbf{x} \mapsto 1\,000 \times \sigma_P(\mathbf{x}, t)$; (c) pdf $r \mapsto p_{\rho_{N_d}}(r)$ of the random variable ρ_{N_d}; (d) L^2-norm $N_d \mapsto \|\widehat{\rho}\|_{L^2}(N_d)$ of the random residual $\widehat{\rho}$.

respectively. Figure 7-(d) displays the graph $N_d \mapsto \|\widehat{\rho}\|_{L^2}(N_d)$. These results show that the convergence with respect to N_d is reasonably obtained for $N_d = 4\,000$.

11.5. Analysis of the role played by the constraints on the probability distribution of $\mathbf{W}$

As in Application 1, for simplifying the writing, $\mathbf{W}^{(\nu)}$ is simply noted $\mathbf{W}$. We consider the training dataset of dimension $N_d = 100$ and $n_{\text{MC}} = 400$ additional realizations (learning) are computed by using the PLoM under constraints. Figures 8-(a) and (b) correspond to *algo* 1 of the PLoM under the constraint on ρ but without the constraints on $\mathbf{W}$. The mean value $\widehat{\underline{\rho}}_{\text{ref}} = E\{\widehat{\rho}_{\text{ref}}\}$ is estimated using the $N_{d,\text{ref}} = 100$ realizations. For $N_d = 100$, $n_{\text{MC}} = 400$, $t = 30$, and using *algo* 1, Figs. 8-(a) and (b) show the error functions $i \mapsto \text{err}_R(i)$ and $i \mapsto \text{err}_W(i)$ defined by Eq. (47). As expected, Fig. 8-(a) shows the decrease in the error function of ρ, which shows that the optimization process decreases the norm of ρ. On the other hand, since there is no constraint on $\mathbf{W}$, the optimization process to decrease the random residual of the stochastic PDE, modifies the prior probability distribution of $\mathbf{W}$, which is seen by the growth of function $i \mapsto \text{err}_W(i)$ that is plotted in Fig. 8-(b). Figures 8-(c) and (d) quantify this modification of the prior probability density function of each component of random vector $\mathbf{W} = (W_1, W_2)$. Figure 8-(c) shows the pdf $w \mapsto p_{W_1}(w)$ of W_1 while Fig. 8-(d) the pdf $w \mapsto p_{W_2}(w)$ of W_2. On each of these two figures, it is plotted the estimated pdf, on the one hand with the $N_d = 100$ realizations (prior pdf estimated with the training), on the other hand with the $n_{\text{MC}} = 400$ additional realizations (learning) calculated using *algo* 1 (converged for $i_{\text{opt}} = 12$) and using *algo* 3 (converged for $i_{\text{opt}} = 5$) of the PLoM under the constraints on ρ and $\mathbf{W}$. It can be seen notable improvement of the probability density function of $\mathbf{W} = (W_1, W_2)$ when the constraints on $\mathbf{W}$ are taken into account.

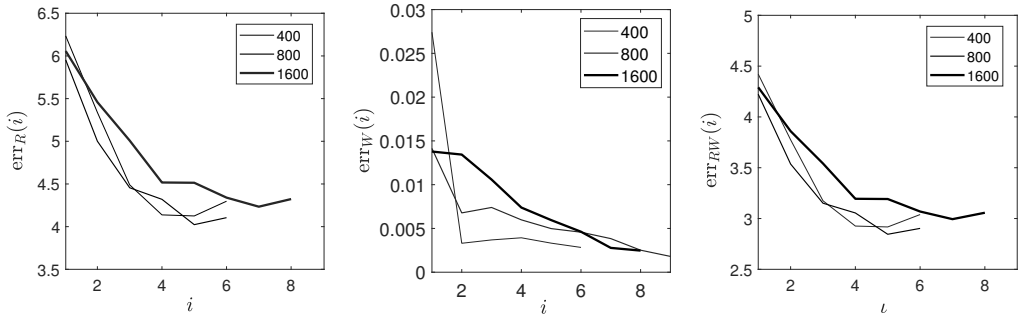


(a) normalized random residue ρ (b) random control parameter $\mathbf{W}$ (c) pdf of W_1 (d) pdf of W_2

Figure 8: For $N_d = 100$, $n_{\mathrm{MC}} = 400$, unsteady stochastic flow at $t = 30$, and for the PLoM computation under the constraint on ρ but without the constraints on $\mathbf{W}$: (a) function $i \mapsto \mathrm{err}_R(i)$; (b) function $i \mapsto \mathrm{err}_W(i)$; (c) pdf $w \mapsto p_{W_1}(w)$ and (d) pdf $w \mapsto p_{W_2}(w)$ estimated with $N_d = 100$ realizations (prior model) (dashed line), with $n_{\mathrm{MC}} = 400$ additional realizations (learning) calculated using *algo* 1 of the PLoM under the constraint on ρ but without the constraints on $\mathbf{W}$ (thin solid line), and with $n_{\mathrm{MC}} = 400$ additional realizations (learning) calculated using *algo* 3 of the PLoM under the constraints on ρ and $\mathbf{W}$ (thick solid line).

11.6. Analysis of the behavior of the iterative algorithms as a function of n_{MC} and convergence criteria

The three algorithms, *algo* 1 to *algo* 3 have deeply been tested. For the reasons given in Section 8.3, we have used and we present the results obtained with *algo* 3 for the iterative algorithm. For unsteady stochastic flow at $t = 30$



(a) Error function err_R (b) Error function err_W (c) Error function err_{RW}

Figure 9: For unsteady stochastic flow at $t = 30$, for $N_d = 100$, and using *algo* 3 for the iteration algorithm, evolution of the error functions as a function of n_{MC}: $i \mapsto \mathrm{err}_R(i)$ (a), $i \mapsto \mathrm{err}_W(i)$ (b), and $i \mapsto \mathrm{err}_{RW}(i)$ (c).

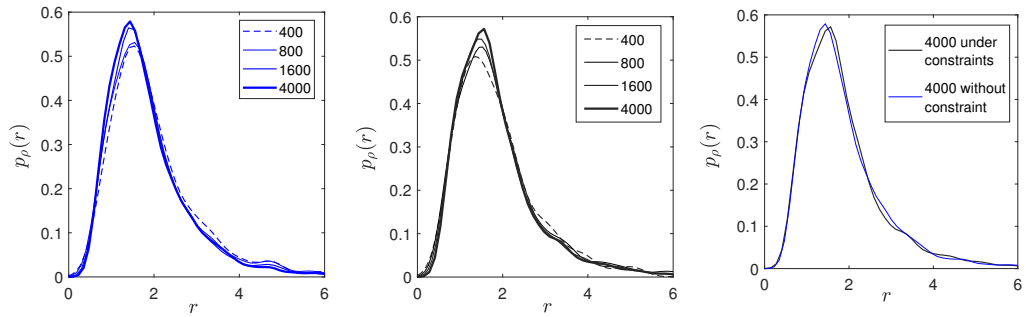
and for $N_d = 100$, Fig. 9 shows the evolution of the error functions $i \mapsto \mathrm{err}_R(i)$, $i \mapsto \mathrm{err}_W(i)$, and $i \mapsto \mathrm{err}_{RW}(i)$ defined in *algo* 3, for $n_{\mathrm{MC}} = 400$, 800, and 1 600. For $n_{\mathrm{MC}} = 400$, a minimum of err_{RW} is reached for $i_{\mathrm{opt}} = 5$ with $\mathrm{err}_{RW}(i_{\mathrm{opt}}) = 2.92$, while, for $n_{\mathrm{MC}} = 1\,600$, a minimum of err_{RW} is reached for $i_{\mathrm{opt}} = 7$ with $\mathrm{err}_{RW}(i_{\mathrm{opt}}) = 2.99$. Although the two values are close, one might think that the solution obtained for $n_{\mathrm{MC}} = 400$ is better than that obtained for $n_{\mathrm{MC}} = 1\,600$. In fact it is not the case because the larger n_{MC} the better the convergence of the statistical estimators of the solution. The effects of the small difference between 2.92 and 2.99 are negligible.

11.7. Results of the PLoM without and under constraints for a given dimension of the training dataset

We consider the training dataset with $N_d = 100$. A large number $n_{\mathrm{MC}} \gg N_d$ of additional realizations is generated with the PLoM under the constraints on ρ and $\mathbf{W}$, by using algorithm *algo* 3. The unsteady stochastic flow is observed at $t = 30$. The challenge is therefore, for this small value N_d of points in the training dataset, to show that, for a large value of n_{MC} (with convergence with respect to n_{MC}), a good approximation of the reference is obtained whereas the use of the PLoM without the constraints gives an acceptable approximation, which can therefore be improved. For this application, the PLoM method under constraints of the stochastic PDE allows an improvement of the PLoM without

constraint [30].

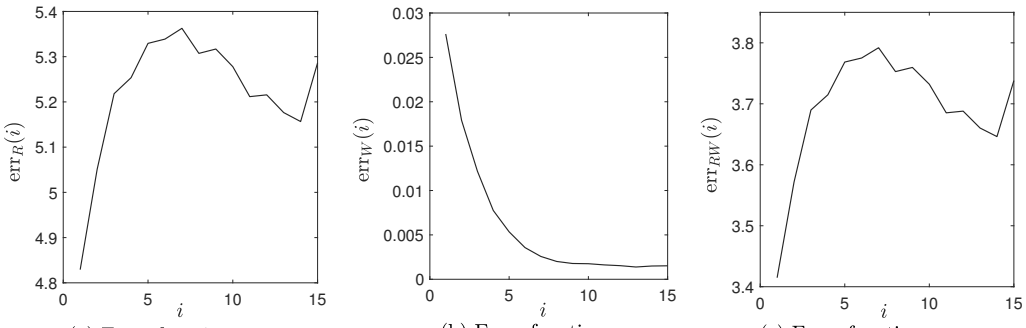
(i) Figure 10 shows the probability density function $r \mapsto p_\rho(r)$ of ρ as a function of the number n_{MC} of additional realizations generated with the PLoM without constraint and under constraints. Figure 10-(c) shows, for $n_{\text{MC}} = 4\,000$, the comparison of this pdf estimated with the PLoM without constraint and under constraints. The analysis of these figures shows a good convergence of the pdf of ρ with respect to n_{MC}.



(a) PLoM without constraint (b) PLoM under constraints (c) PLoM without and under constraints

Figure 10: For unsteady stochastic flow at $t = 30$ and for $N_d = 100$, graph of pdf $r \mapsto p_\rho(r)$ of ρ as a function of n_{MC}, estimated (a) with the PLoM without constraint, (b) with the PLoM under constraints with *algo* 3, and (c) for $n_{\text{MC}} = 4\,000$, the PLoM without and under constraints.

(ii) For the unsteady stochastic flow at $t = 30$, for $N_d = 100$ and $n_{\text{MC}} = 4\,000$, Fig. 11 shows the evolution of the error functions $i \mapsto \text{err}_R(i)$, $i \mapsto \text{err}_W(i)$, and $i \mapsto \text{err}_{RW}(i)$ computed with *algo* 3. The iterative algorithm is stopped at iteration 14 for which there is a local minimum that is $\text{err}_{RW}(14) = 3.65$ and for which $\text{err}_R(14) = 5.16$ and $\text{err}_W(14) = 0.0015$. Figure 12 shows the prior probability density function $w \mapsto p_{W_1}(w)$ and $w \mapsto p_{W_2}(w)$ of



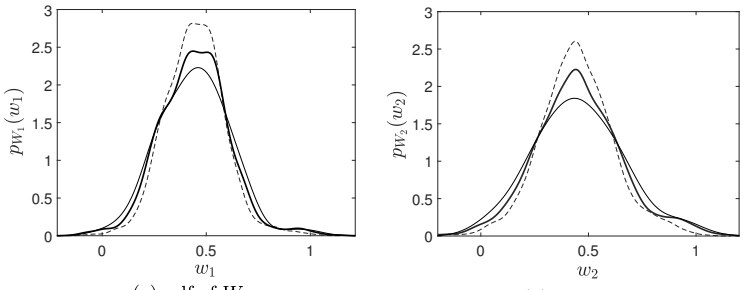
(a) Error function err_R (b) Error function err_W (c) Error function err_{RW}

Figure 11: For unsteady stochastic flow at $t = 30$, $N_d = 100$, $n_{\text{MC}} = 4\,000$, and using *algo* 3, evolution of the error functions $i \mapsto \text{err}_R(i)$ (a), $i \mapsto \text{err}_W(i)$ (b), and $i \mapsto \text{err}_{RW}(i)$ (c).

each component W_1 and W_2 of random vector $\mathbf{W}$. In each figure is plotted the estimated pdf, on the one hand with the $N_d = 100$ realizations from the training dataset (prior pdf estimated with the training), on the other hand with the $n_{\text{MC}} = 4\,000$ additional realizations (learning) estimated with the PLoM without constraint and with the PLoM under the constraints on ρ and $\mathbf{W}$. The mean value $E\{\widehat{\rho}\}$ of $\widehat{\rho}$ is 3.23×10^{-3}, its standard deviation is 2.74×10^{-3}, and $\|\widehat{\rho}\,\|_{L^2}$ is 4.23×10^{-3}. These values must be compared with those of the reference given in Section 11.4-(ii).

To illustrate the distribution of the realizations generated by the PLoM under constraints, Fig. 13 displays the 4 000 additional realizations of components of $\mathbf{X}^c = (\mathbf{Q}^c, \mathbf{W}^c)$. A concentration of these realizations can be observed.

Figure 14 displays the 2D-plot of $\mathbf{x} \mapsto 1\,000 \times m_P(\mathbf{x}, t)$ and $\mathbf{x} \mapsto 1\,000 \times \sigma_P(\mathbf{x}, t)$ estimated with $n_{\text{MC}} = 4\,000$ additional realizations generated by the PLoM without constraint and are compared to the reference (see Fig. 7).



(a) pdf of W_1 (b) pdf of W_2

Figure 12: For $N_d = 100$ and unsteady stochastic flow at $t = 30$, pdf $w \mapsto p_{W_1}(w)$ (a) and pdf $w \mapsto p_{W_2}(w)$ (b), estimated with the $N_d = 100$ realizations (prior model) from the training dataset (solid line), with $n_{\mathrm{MC}} = 4\,000$ additional realizations (learning) estimated with the PLoM without constraint (dashed line) and under the constraints (thick solid line).

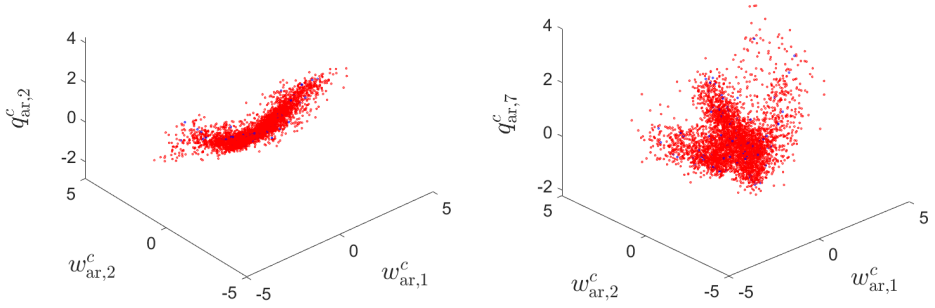
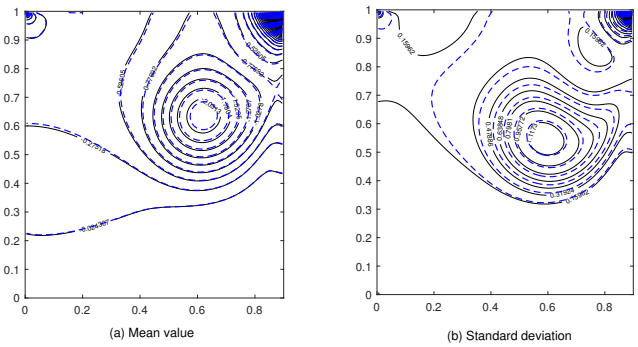


Figure 13: 3D-plot of realizations $(w^c_{\mathrm{ar},1}, w^c_{\mathrm{ar},2}, q^c_{\mathrm{ar},2})$ of (W^c_1, W^c_2, Q^c_2) (left figure) and $(w^c_{\mathrm{ar},1}, w^c_{\mathrm{ar},2}, q^c_{\mathrm{ar},7})$ of (W^c_1, W^c_2, Q^c_7) (right figure), computed with the PLoM under constraints: $n_{\mathrm{MC}} = 4\,000$ additional realizations (red symbols) and $N_d = 100$ points from the training dataset (blue symbols).



(a) Mean value (b) Standard deviation

Figure 14: For unsteady stochastic flow at $t = 30$ and for $N_d = 100$, 2D-plot of mean value $\mathbf{x} \mapsto 1\,000 \times m_P(\mathbf{x}, t)$ (a) and of standard deviation $\mathbf{x} \mapsto 1\,000 \times \sigma_P(\mathbf{x}, t)$ (b) of the pressure random field corresponding to the reference (black thin solid lines) and estimated with $n_{\mathrm{MC}} = 4\,000$ additional realizations computed with the PLoM without constraint (blue thick dashed lines).

These results show that the standard PLoM method (that is to say without constraint) is already good. It allows for generating a large number n_{MC} of additional realizations (learning) from a small number N_d of points of the training dataset. These realizations allow for estimating the second-order quantities of the random field, which are good for the mean and less good for the standard deviation (see the zoom and the associated comment of Fig. 15-(c)).

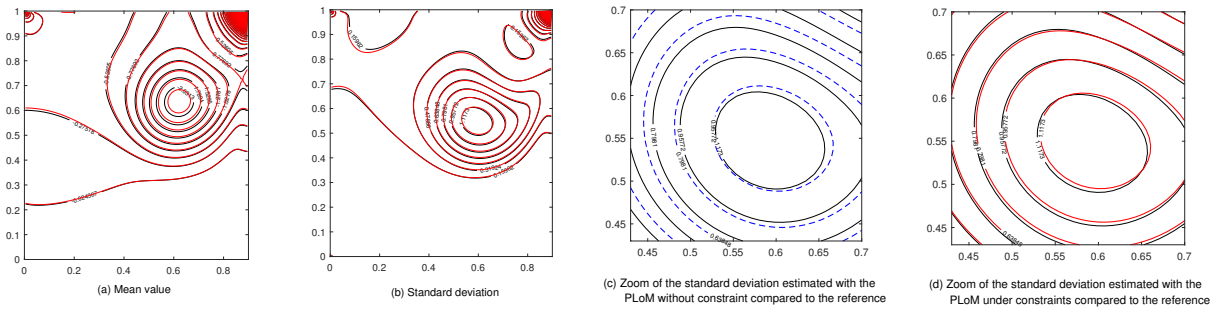


Figure 15: For unsteady stochastic flow at $t = 30$ and for $N_d = 100$, 2D-plot of mean value $\mathbf{x} \mapsto 1\,000 \times m_P(\mathbf{x}, t)$ (a) and of standard deviation $\mathbf{x} \mapsto 1\,000 \times \sigma_P(\mathbf{x}, t)$ (b) of the pressure random field corresponding to the reference (black thin solid lines) and estimated with $n_{\text{MC}} = 4\,000$ additional realizations computed with the PLoM under constraints (red thick solid lines). Fig. (c) is a zoom of Fig. 14-(b). Fig. (d) is a zoom of Fig. (b).

Similarly, Fig. 15 displays the 2D-plot of $\mathbf{x} \mapsto 1\,000 \times m_P(\mathbf{x}, t)$ and $\mathbf{x} \mapsto 1\,000 \times \sigma_P(\mathbf{x}, t)$ estimated with $n_{\text{MC}} = 4\,000$ additional realizations computed with the PLoM under constraints and shows the comparison with the reference (see Fig. 7). These results show that the PLoM under constraints give an excellent prediction of the second-order quantities, for both the mean and the standard deviation of the random fields. Figure 15-(c) corresponds to a zoom of Fig. 14-(b). At the center of the vortex, it can be seen that the value of the innermost level line (in the center of the vortex) is 0.95772 for the reference solution (solid line) while that which corresponds for the PLoM without constraint, is not the proximal dashed line, but the second deviating from the center. The value of the proximal dashed line is 1.1173 instead of 0.95772. In contrast, Fig. 15-(d) that is a zoom of Fig. 15-(b) shows an excellent coincidence of the level lines for the same region of the fluid domain.

11.8. Discussion

The results obtained for this application devoted to unsteady nonlinear stochastic computational fluid dynamics show that the PLoM without constraint is already giving good results (as it was expected, taking into account the previous works). However the novel proposed extension, consisting in constraining the PLoM method by the stochastic PDE allows to improve the results, which are then quite excellent (see zoom in Fig. 15-(d)).

12. Application 3: nonlinear 3D solid dynamics

12.1. Description of the nonlinear 3D dynamical system

(i) System definition and geometry. It is a 3D MEMS accelerometer whose scheme is shown in Fig. 1-(A) of Ref. [42]. The mobile part is constituted of a deformable rectangular frame in silicon with one vertical deformable central beam attached to it. The suspended part is made up of a deformable parallelepipedic solid with two attached vertical deformable beams that are around the central beam, and is attached to the mobile part by a 3D suspension constituted of 20 linear springs. The MEMS is represented by an 3D-domain $\Omega \subset \mathbb{R}^3$ and is described in a Cartesian coordinate system $Ox_1x_2x_3$ that is attached to its mobile part and adopted as the reference coordinate system. The origin O is located at the bottom left corner of the device. The axis Ox_1 is horizontal and oriented positively from left to right. The axis Ox_2 is vertical and oriented positively from bottom to top. The axis Ox_3 is perpendicular to the plane Ox_1x_2. The overall dimensions in microns ($\times 10^{-6}\, m$) of the rectangular frame are $30 \times 31 \times 4$.

(ii) Materials. The suspended and mobile parts are made of a homogeneous, orthotropic, linear elastic material, which is a standard (100) silicon wafer [67], whose Young moduli are $\underline{E}_{11} = \underline{E}_{22} = 169 \times 10^9\ N/m^2$, $\underline{E}_{33} = 130 \times 10^9\ N/m^2$, with Poisson ratios $\underline{\nu}_{23} = 0.36$, $\underline{\nu}_{31} = 0.28$, $\underline{\nu}_{12} = 0.064$, and with shear moduli $\underline{G}_{23} = \underline{G}_{31} = 79.6 \times 10^9\ N/m^2$ and $\underline{G}_{12} = 50.9 \times 10^9\ N/m^2$. Let $[\underline{C}] \in \mathbb{M}_6^+$ be the Kelvin representation of the fourth-order elasticity tensor of this medium, which belongs to the class of orthotropic symmetry. The mass density is $2\,330\ Kg/m^3$. There are three types

of spring. Each spring acts in one direction. In x_1-, x_2-, and x_3-directions, the stiffnesses are $\underline{k}_1^s = 4\ N/m$, $\underline{k}_2^s = 6\ N/m$, and $\underline{k}_3^s = 1.5\ N/m$. A nonlinear elastic material is inserted between the two vertical beams of the suspended part and the vertical central beam of the mobile part. Its constitutive equation corresponds to a cubic, elastic, restoring force whose elastic constant is $k_b = 2 \times 10^{12}\ N/m$.

(iii) Time-dependent excitation. The base Γ_0 of the mobile part is submitted to a given acceleration in the x_1-direction. The time-dependent modulation is $\gamma(t) = \gamma_0 \left\{ \sin\left(t(\omega_c + \Delta\omega_c/2)\right) - \sin\left(t(\omega_c - \Delta\omega_c/2)\right) \right\}/(\pi\, t)$, in which $\gamma_0 = 120\, m/s^2$, $\omega_c = 2\,\pi \times 13 \times 10^6\ rad/s$ is the central angular frequency and where $\Delta\omega_c = 2\,\pi \times 10 \times 10^6\ rad/s$ is the bandwidth. In the positive-frequency domain, the energy of signal $\gamma(t)$ is mainly concentrated in the frequency band $[\omega_{\min}, \omega_e]$, with $\omega_{\min} = 2\pi \times 8 \times 10^6\ rad/s$ and $\omega_e = 2\pi \times 18 \times 10^6\ rad/s$.

(iv) Time interval of analysis, frequency band of analysis, and observations. This nonlinear dynamical system is analyzed for $t \in [t_0, T]$ with $t_0 = -1.3889 \times 10^{-5}\ s$ and $T = 3.1014 \times 10^{-5}\ s$. At time t_0, the device is at rest. The frequency band of analysis is $[0\,, \omega_{\max}]$ with $\omega_{\max} = 2\pi \times 72 \times 10^6\ rad/s$. The frequency band of observation is $\mathcal{B}_o = [0, \omega_o]$, where $\omega_o = 2\,\pi \times 70 \times 10^6\ rad/s$. If the dynamical system was linear, the energy of the response signal would be concentrated in the same frequency band $[\omega_{\min}, \omega_e]$ as that of the energy of the excitation signal. Due to the nonlinearity however, part of the energy of the excitation signal is transferred outside its frequency band and consequently, the frequency band of the response is not $[\omega_{\min}, \omega_e]$ but $[0, \omega_{\max}]$, where $\omega_{\max} > \omega_e$. The results will be presented for two selected observations Obs_1 and Obs_6. The first one is the x_1-dof, denoted as dof_1, of the node of coordinates $(14.5, 8.0, 0.0) \times 10^{-6}\, m$ located on the vertical beam of the mobile part. The second one is the x_1-dof, denoted as dof_6, of the node of coordinates $(23.0, 17.0, 0.0) \times 10^{-6}\, m$ located at the down right corner of the suspended part. The results will be presented in the frequency domain (in Hz) for each one of the two observations and will be the random function $\nu_{\text{freq}} \mapsto \text{dB}_j(\nu_{\text{freq}}) = \log_{10}(|\, \widehat{Y}_{\text{dof}_j}(2\pi\nu_{\text{freq}})\,|)$ for $j = 1$ and $j = 6$, in which $\omega \mapsto \widehat{Y}_{\text{dof}_j}(\omega)$ is the Fourier transform of the displacement $t \mapsto Y_{\text{dof}_j}(t)$.

12.2. Prior probability model of uncertainties

(i) Prior probability model of the springs stiffnesses. Let $\mathbf{K}^s = (K_1^s, K_2^s, K_3^s)$ be the $\mathbb{R}^3$-valued random variable defined on a probability space $(\Theta, \mathcal{T}, \mathcal{P})$. For $j \in \{1, 2, 3\}$, K_j^s is the random stiffness of the spring of type j, with mean value $\underline{k}_j^s$ and with a uniform probability distribution on $[\underline{k}_j^s(1 - \sqrt{3}\,\delta_s)\,, \underline{k}_j^s(1 + \sqrt{3}\,\delta_s)]$, in which $\delta_s = 0.2$ for all j. Random variable K_j^s can then be written as $K_j^s = \underline{k}_j^s\big(1 + \sqrt{3}\,\delta_s(2\,\mathcal{U}_j^s - 1)\big)$, in which $\mathcal{U}_j^s$ is a uniform random variable on $[0, 1]$ such that $\mathcal{U}_j^s = \big(1 + \text{erf}(W_j^s/\sqrt{2})\big)/2$, where W_j^s is a normalized Gaussian random variable. The random variables W_1^s, W_2^s, W_3^s are assumed to be independent and we introduce the $\mathbb{R}^3$-valued random variable $\mathbf{W}^s = (W_1^s, W_2^s, W_3^s)$.

(ii) Prior probability model of the elasticity tensor of the silicon. The material in Ω is a random homogeneous medium. The Kelvin matrix representation $[\mathbf{C}]$ of the random fourth-order elasticity tensor is defined on $(\Theta, \mathcal{T}, \mathcal{P})$, independent of $\mathbf{W}^s$, with values in $\mathbb{M}_6^+$. Its mean value is matrix $[\underline{C}] \in \mathbb{M}_6^+$ whose Cholesky factorization is $[\underline{C}] = [\underline{L}_C]^T\, [\underline{L}_C]$. The statistical fluctuations of $[\mathbf{C}]$ around $[\underline{C}]$ can be constructed either in the class of orthotropic symmetry [68] or both in the orthotropic symmetry class and the anisotropic symmetry class [69, 70]. Here, we consider the statistical fluctuations in the anisotropy symmetry class [71, 70] and consequently, $[\mathbf{C}] = [C_\epsilon] + [\underline{L}_\epsilon]^T\, [\mathbf{G}]\, [\underline{L}_\epsilon]$ in which $[C_\epsilon] = \epsilon(1 + \epsilon)^{-1}[\underline{C}]$ and $[\underline{L}_\epsilon] = (1 + \epsilon)^{-1/2}[\underline{L}_C]$ with $\epsilon = 10^{-6}$. The $\mathbb{M}_6^+$-valued random matrix $[\mathbf{G}]$ is written as $[\mathbf{G}] = [\mathbf{L}_G]^T\, [\mathbf{L}_G]$ in which $[\mathbf{L}_G]$ is a random (6×6) upper triangular matrix defined as follows [70]. Let $\{\mathcal{W}_{ij}\}_{1 \le i \le j \le 6}$ be 21 independent normalized Gaussian real-valued random variables defined on $(\Theta, \mathcal{T}, \mathcal{P})$, independent of $\mathbf{W}^s$. The upper part of matrix $[\mathbf{L}_G]$ is such that $[\mathbf{L}_G]_{ij} = (\delta_C/\sqrt{7})\,\mathcal{W}_{ij}$ for $1 \le i < j \le 6$ and $[\mathbf{L}_G]_{ii} = (\delta_C/\sqrt{7})\big(2\beta(\mathcal{W}_{ii}, \alpha_i)\big)^{1/2}$ for $i \in \big\{1, \ldots, 6\big\}$ in which $\alpha_i = 7/(2\delta_C^2) + (1 - i)/2$ and where $w \mapsto \beta(w, \alpha_i)$ is such that $\Gamma_{\alpha_i} = \beta(\mathcal{W}_{ii}, \alpha_i)$ is the Gamma random variable with parameter α_i. We have $E\big\{[\mathbf{G}]\big\} = [I_6]$. The statistical fluctuations are controlled by the hyperparameter $\delta_C = \big\{(1/6)\, E\{\|\, [\mathbf{G}] - [I_6]\,\|_F^2\}\big\}^{1/2}$ set here to the value 0.1 in which $\|\cdot\|_F$ is the Frobenius norm. Finally, we introduce the random vector $\mathbf{W}^C = (W_1^C, \ldots, W_{21}^C)$ such that $W_i^C = \mathcal{W}_{ii}$ for $i \in \big\{1, \ldots, 6\big\}$ and $W_{6+k}^C = \mathcal{W}_{ij}$ for $1 \le i < j \le 6$ with $k = (i, j) \in \big\{1, \ldots, 15\big\}$.

(iii) Random vector $\mathbf{W}$. Let $\mathbf{W} = (W_1, \ldots, W_{n_w})$ be the $\mathbb{R}^{n_w}$-valued random variable defined by $\mathbf{W} = (\mathbf{W}^s, \mathbf{W}^C)$ with $n_w = 3 + 21 = 24$. For constituting the training dataset, the N_d realizations $\{\mathbf{w}_d^\ell, \ell = 1, \ldots, N_d\}$ of $\mathbf{W}$ are generated with its prior probability model. The random stiffnesses $\mathbf{K}^s$ and the random elasticity matrix $[\mathbf{C}]$ are defined by a given $\mathbb{R}^{n_w}$-valued mapping f_{param} on $\mathbb{R}^{n_w}$ such that

$$\{\mathbf{K}^s, [\mathbf{C}]\} = f_{\text{param}}(\mathbf{W}) \,. \tag{86}$$

12.3. Computational model, KL expansion, random residual, and PLoM parameters

(i) Discrete nonlinear dynamical equation. The semidiscretization of the boundary value problem of the nonlinear 3D dynamical system is performed using the Finite Element (FE) method. The FE mesh is shown in Fig. 1-(B) of Ref. [42]. There are 7 328 eight-nodes solid elements, 10 675 nodes, and $N = 31\,410$ free dofs. In the reference coordinate system, the semidiscretized stochastic dynamical system is written as

$$[M]\,\ddot{\mathbf{Y}}(t) + [D]\,\dot{\mathbf{Y}}(t) + [K(\mathbf{W})]\,\mathbf{Y}(t) + \mathbf{f}_{\text{NL}}(\mathbf{Y}(t)) = \gamma(t)\,\mathbf{f}_M \,, \quad t \in]t_0, T] \,, \tag{87}$$

$$\mathbf{Y}(t_0) = \mathbf{0}_N \;,\; \dot{\mathbf{Y}}(t_0) = \mathbf{0}_N \,. \tag{88}$$

In Eq. (87), $\mathbf{Y}(t) = (Y_1(t), \ldots, Y_N(t))$ is the vector of the N free degrees of freedom of the displacements in the reference coordinate system. The corresponding velocity and acceleration vectors are $\dot{\mathbf{Y}}(t)$ and $\ddot{\mathbf{Y}}(t)$. The matrices $[M]$, $[D]$, and $[K(\mathbf{W})]$ are the FE mass, damping, and stiffness matrices with values in $\mathbb{M}_N^+$. The stiffness matrix depends on $\mathbf{W}$ via the use of Eq. (86) and $\mathbf{f}_{\text{NL}}(\mathbf{Y}(t))$ is the $\mathbb{R}^N$-vector of the nonlinear internal forces at time t induced by the nonlinear material. Finally, $\mathbf{f}_M$ is the $\mathbb{R}^N$ vector independent of time t, which represents the spatial part of the external forces in the reference coordinate system, due to the imposed acceleration of boundary Γ_0. Note that, for the dynamical system under consideration, $\mathbf{f}_M$ is independent of the damping and stiffness matrices. The time-dependent load vector is such that $\|\mathbf{f}_M\| = 9.64 \times 10^{-12}$, $\max_t |\gamma(t)| = 2 \times 10^7$, and consequently, $\max_t \|\gamma(t)\mathbf{f}_M\| = 5.59 \times 10^{-6}$.

(ii) Damping modeling. Let $\boldsymbol{\varphi}^1, \ldots, \boldsymbol{\varphi}^{n_{\text{mode}}}$ be the first $n_{\text{mode}} = 40$ elastic modes, normalized with respect to the mass matrix, associated with the first n_{mode} lowest eigenfrequencies $0 < \omega_1 \leq \ldots \leq \omega_{n_{\text{mode}}}$ such that $[K(\underline{\mathbf{w}})]\,\boldsymbol{\varphi}^\beta = \omega_\beta^2\,[M]\,\boldsymbol{\varphi}^\beta$ in which $\underline{\mathbf{w}} = E\{\mathbf{W}\}$. We have $\omega_1 = 2\pi \times 4.1341 \times 10^5$ rad/s and $\omega_{\text{mode}} = 2\pi \times 1.2400 \times 10^8$ rad/s. There are 25 elastic modes in the frequency band $\mathcal{B}_o$. The computational model of matrix $[D] \in \mathbb{M}_N^+$ is derived from the one proposed Page 93 in [72], and is written as $[D] = \sum_{\beta=1}^{n_{\text{mode}}} 2\,\chi_d\,(\omega_\beta - \omega_{n_{\text{mode}}})\,[M]\,\boldsymbol{\varphi}^\beta\,([M]\,\boldsymbol{\varphi}^\beta)^T + 2\,\chi_d\,\omega_{n_{\text{mode}}}\,[M]$, in which $\chi_d = 0.02$. This damping model yields a constant damping rate χ_d for the first n_{mode} elastic modes and yields the damping rates $\{\chi_d\,\omega_{n_{\text{mode}}}/\omega_{n_{\text{mode}}+1}, \ldots, \chi_d\,\omega_{n_{\text{mode}}}/\omega_N\}$ for the $N - n_{\text{mode}}$ higher elastic modes.

(iii) Time sampling and discretization scheme. Let $\Delta t = 6.9444 \times 10^{-9} s$ be the time step. The sampling times are $t_1, \ldots, t_{n_{\text{time}}}$ such that $t_n = t_0 + n\,\Delta t$ for $n = 1, \ldots n_{\text{time}}$ in which $n_{\text{time}} = 6\,466$ and $T - t_0 = n_{\text{time}}\,\Delta t = 4.4903 \times 10^{-5} s$. Using Eqs. (87) and (88) yields, because $\mathbf{f}_{\text{NL}}(\mathbf{0}_N) = \mathbf{0}_N$,

$$\mathbf{Y}(t_0) = \mathbf{0}_N \;,\; \dot{\mathbf{Y}}(t_0) = \mathbf{0}_N \;,\; \ddot{\mathbf{Y}}(t_0) = \ddot{\mathbf{y}}(t_0) \;,\; \ddot{\mathbf{y}}(t_0) = \gamma(t_0)\,[M]^{-1}\,\mathbf{f}_M \,. \tag{89}$$

The centered Newmark integration method (implicit scheme) is used. At each sampling time, the nonlinear algebraic equation is solved by using a fixed point method with a relative precision of 10^{-6}. In order to guarantee the convergence of the fixed point method, a local adaptive time step is used. Knowing $\mathbf{Y}(t_{n-1})$, $\dot{\mathbf{Y}}(t_{n-1})$, and $\ddot{\mathbf{Y}}(t_{n-1})$ at sampling time t_{n-1}, and knowing $\mathbf{Y}(t_n)$, the following equations are used for calculating $\ddot{\mathbf{Y}}(t_n)$ and $\dot{\mathbf{Y}}(t_n)$,

$$\ddot{\mathbf{Y}}(t_n) = a_0(\mathbf{Y}(t_n) - \mathbf{Y}(t_{n-1})) - a_1\dot{\mathbf{Y}}(t_{n-1}) - a_2\ddot{\mathbf{Y}}(t_{n-1}) \,, \tag{90}$$

$$\dot{\mathbf{Y}}(t_n) = \dot{\mathbf{Y}}(t_{n-1}) + a_3(\ddot{\mathbf{Y}}(t_{n-1}) + \ddot{\mathbf{Y}}(t_n)) \,, \tag{91}$$

in which $a_0 = 4/\Delta t^2$, $a_1 = 4/\Delta t$, $a_2 = 1$, and $a_3 = \Delta t/2$.

(iv) Computational model. The FE discretization and time discretization scheme previously defined allow for computing the $\mathbb{R}^N$-valued time series $\{(\mathbf{Y}(t_n), \dot{\mathbf{Y}}(t_n), \ddot{\mathbf{Y}}(t_n)), n = 1, \ldots, n_{\text{time}}\}$. In that condition, Eq. (1) is rewritten, at

sampling time t_n, as

$$\boldsymbol{\mathcal{N}}^{\text{SD}}(\ddot{\mathbf{Y}}(t_n), \dot{\mathbf{Y}}(t_n), \mathbf{Y}(t_n), t_n, \mathbf{W}) = \mathbf{0}_N \quad a.s., \tag{92}$$

in which $\boldsymbol{\mathcal{N}}^{\text{SD}}$ is a nonlinear operator with values in $\mathbb{R}^N$.

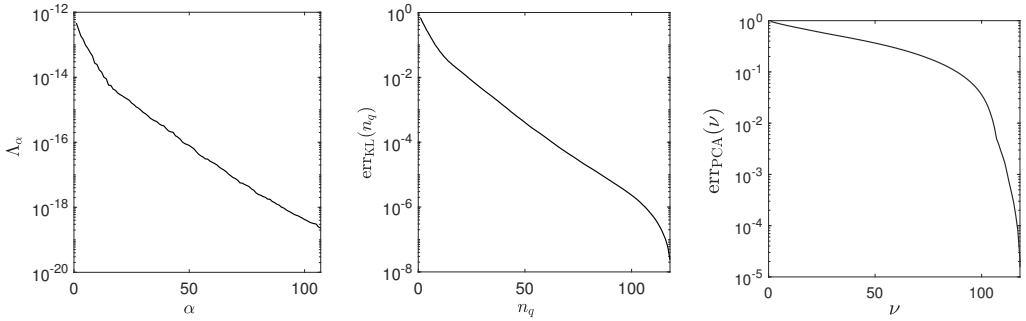
(v) KL expansion of the nonstationary stochastic processes $\mathbf{Y}$. The KL expansion of the nonstationary $\mathbb{R}^N$-valued stochastic process $\{\mathbf{Y}(t), t \in [t_0, T]\}$ is performed at the sampling times, $t_1, \ldots, t_{n_{\text{time}}}$, yielding the representation $\mathbf{Y}(t_n) = \underline{\mathbf{y}}(t_n) + [V(t_n)]\,\mathbf{Q}$. Knowing $\underline{\mathbf{y}}(t_n)$ and $[V(t_n)]$ for $n \in \{1, \ldots, n_{\text{time}}\}$, their first and second time derivatives are computed by the following recurrences based on the centered Newmark integration method used in Eqs. (90) and (91).

(a) Initialization of the recurrences. $\mathbf{Y}(t_0) = \dot{\mathbf{Y}}(t_0) = \mathbf{0}_N$ and $\ddot{\mathbf{Y}}(t_0) = \ddot{\mathbf{y}}(t_0)$ in which $\ddot{\mathbf{y}}(t_0)$ is defined in Eq. (89). Choosing $\underline{\mathbf{y}}(t_0) = \dot{\underline{\mathbf{y}}}(t_0) = \mathbf{0}_N$ and $\ddot{\underline{\mathbf{y}}}(t_0) = \ddot{\mathbf{y}}(t_0)$, it can then be deduced that $[V(t_0)] = [\dot{V}(t_0)] = [\ddot{V}(t_0)] = [0_{N,n_q}]$.

(b) Recurrences. For computing the mean function $t \mapsto \underline{\mathbf{y}}(t)$, the recurrence is written, for $n \in \{1, \ldots, n_{\text{time}}\}$, as $\ddot{\underline{\mathbf{y}}}(t_n) = a_0\,(\underline{\mathbf{y}}(t_n) - \underline{\mathbf{y}}(t_{n-1})) - a_1\,\dot{\underline{\mathbf{y}}}(t_{n-1}) - a_2\,\ddot{\underline{\mathbf{y}}}(t_{n-1})$ with $\dot{\underline{\mathbf{y}}}(t_n) = \dot{\underline{\mathbf{y}}}(t_{n-1}) + a_3\,(\ddot{\underline{\mathbf{y}}}(t_{n-1}) + \ddot{\underline{\mathbf{y}}}(t_n))$, while for the function $t \mapsto [V(t)]$, the recurrence is $[\ddot{V}(t_n)] = a_0\,([V(t_n)] - [V(t_{n-1})]) - a_1\,[\dot{V}(t_{n-1})] - a_2\,[\ddot{V}(t_{n-1})]$ with $[\dot{V}(t_n)] = [\dot{V}(t_{n-1})] + a_3\,([\ddot{V}(t_{n-1})] + [\ddot{V}(t_n)])$. Consequently, the KL expansions of stochastic processes $\mathbf{Y}$, $\dot{\mathbf{Y}}$, and $\ddot{\mathbf{Y}}$ at the sampling times $t_1, \ldots, t_{n_{\text{time}}}$ are

$$\mathbf{Y}(t_n) = \underline{\mathbf{y}}(t_n) + [V(t_n)]\,\mathbf{Q} \quad , \quad \dot{\mathbf{Y}}(t_n) = \dot{\underline{\mathbf{y}}}(t_n) + [\dot{V}(t_n)]\,\mathbf{Q} \quad , \quad \ddot{\mathbf{Y}}(t_n) = \ddot{\underline{\mathbf{y}}}(t_n) + [\ddot{V}(t_n)]\,\mathbf{Q}\,. \tag{93}$$

For $N_d = 120$, Figs. 16-(a) and -(b) show the distribution of the eigenvalues $\alpha \mapsto \Lambda_\alpha$ of the KL-expansion of stochastic process $\{\mathbf{Y}(t), t \in [t_0, T]\}$ and the error function $n_q \mapsto \text{err}_{\text{KL}}(n_q)$ defined by Eq. (15). Fixing the tolerance ε_{KL} to the value 10^{-6} yields the optimal value $n_q = 107$.



(a) Eigenvalues Λ_α of the K-L expansion (b) Error induced by K-L expansion (c) Error induced by PCA

Figure 16: For $N_d = 120$, for $T - t_0 = 4.4903 \times 10^{-5}\,s$, and for $n_{\text{time}} = 6\,466$, KL expansion of nonstationary stochastic process $\mathbf{Y}$: (a) distribution of the eigenvalues $\alpha \mapsto \Lambda_\alpha$ and (b) error function $n_q \mapsto \text{err}_{\text{KL}}(n_q)$. PCA of random vector $\mathbf{X}$: (c) error function $\nu \mapsto \text{err}_{\text{PCA}}(\nu)$.

(vi) Calculation of random residual $\widehat{\rho}$ *using the subsampling times* . Let $b(t_n) = (1/N_d)\sum_{\ell=1}^{N_d} \|\,\mathbf{y}^\ell(t_n)\,\|$ be the estimate of the mean value of $\|\,\mathbf{Y}(t_n)\,\|$ computed with the training dataset. The subsampling times $\{\tau_{n_s}, n_s = 1, \ldots, n_{\text{sp}}\} \subset \{t_n, n = 1, \ldots, n_{\text{time}}\}$ are chosen as the n_{sp} sampling times for which $b(\tau_1), \ldots, b(\tau_{n_{\text{sp}}})$ correspond to the n_{sp} largest values of the set $\{b(t_1), \ldots, b(t_n)\}$. For calculating the random residual with the KL expansions, the family $\{\mathbf{Y}(\tau_{n_s}), \dot{\mathbf{Y}}(\tau_{n_s}), \ddot{\mathbf{Y}}(\tau_{n_s}), n_s = 1, \ldots, n_{\text{sp}}\}$ is extracted from the family $\{\mathbf{Y}(t_n), \dot{\mathbf{Y}}(t_n), \ddot{\mathbf{Y}}(t_n), n = 1, \ldots, n_{\text{time}}\}$ computed with Eq. (93). Taking into account Eq. (92) at subsampling time τ_{n_s}, the $\mathbb{R}^N$-valued random residual $\boldsymbol{\mathcal{R}}(\tau_{n_s})$, defined by Eq. (17), is rewritten as

$$\boldsymbol{\mathcal{R}}(\tau_{n_s}) = \boldsymbol{\mathcal{N}}^{\text{SD}}(\ddot{\mathbf{Y}}(\tau_{n_s}), \dot{\mathbf{Y}}(\tau_{n_s}), \mathbf{Y}(\tau_{n_s}), \tau_{n_s}, \mathbf{W}) \quad a.s. \tag{94}$$

The positive-valued random variable $\mathcal{R}_{\text{norm}}(\tau_{n_s})$ is defined by Eq. (19) and writes $\mathcal{R}_{\text{norm}}(\tau_{n_s}) = N^{-1/2}\,\|\,\boldsymbol{\mathcal{R}}(\tau_{n_s})\,\|$. The random residual $\widehat{\rho}$ is then calculated using Eq. (24) with $n_{\text{sp}} = 60$. The mean value $\widehat{\rho}_{\text{ref}}$ is calculated for the fixed value $N_{d,\text{ref}} = 120$. For $N_d = 120$ and using the KL expansion, the mean value of $\widehat{\rho}$ is 5.48×10^{-7} and $\|\,\widehat{\rho}\,\|_{L^2} = 6.93 \times 10^{-7}$.

(vii) Values of the parameters for the PLoM without and under constraints. We consider always the case $N_d = 120$ for which the KL expansion of stochastic process $\mathbf{Y}$ has been performed. The dimension of $\mathbf{X} = (\mathbf{Q}, \mathbf{W})$ is $n_x = n_q + n_w = 107 + 24 = 131$. Concerning the PCA of $\mathbf{X}$, Fig. 16-(c) shows the error function $\nu \mapsto \mathrm{err}_{\mathrm{PCA}}(\nu)$ defined by Eq. (27). Fixing the tolerance $\varepsilon_{\mathrm{PCA}}$ to the value 10^{-3} yields the optimal value $\nu = 112$. For the PLoM without constraint or under constraints, the diffusion-maps basis is computed with $\varepsilon_{\mathrm{diff}} = 240$, which yields the optimal dimension $m = 113$. Concerning the Störmer-Verlet algorithm for solving the nonlinear ISDE: $f_0 = 4$ and $\Delta r = 0.2147$.

12.4. Stochastic solution of reference and predictions by the PLoM without constraint

(i) Solution of reference. A reference solution has been computed using $N_d = 1\,000$ realizations from the training dataset requiring a significant CPU effort. Figure 17 shows, for each one of the two observations, two types of graphs. The first type corresponds to the deterministic function $\nu_{\mathrm{freq}} \mapsto \mathrm{dB}_{\mathrm{lin},j}(\nu_{\mathrm{freq}})$ computed by using the deterministic linear computational model for which the control parameters are fixed to $\underline{\mathbf{w}}$ and the nonlinear material between the beams are removed. The second type corresponds, for the probability level $p_c = 0.98$, to the confidence region of the random function $\nu_{\mathrm{freq}} \mapsto \mathrm{dB}_j(\nu_{\mathrm{freq}})$ computed using with the nonlinear stochastic computational model estimated using the $N_d = 1\,000$ realizations. It can be observed that the effect of the structural nonlinearity is very important outside the frequency band $[\omega_{\min}, \omega_e]$ of the excitation. Figure 18-(a) displays the graphs of the pdf $r \mapsto p_{\rho_{N_d}}(r)$ of $\rho_{N_d} = \widehat{\rho}/\underline{\widehat{\rho}}_{N_d}$ in

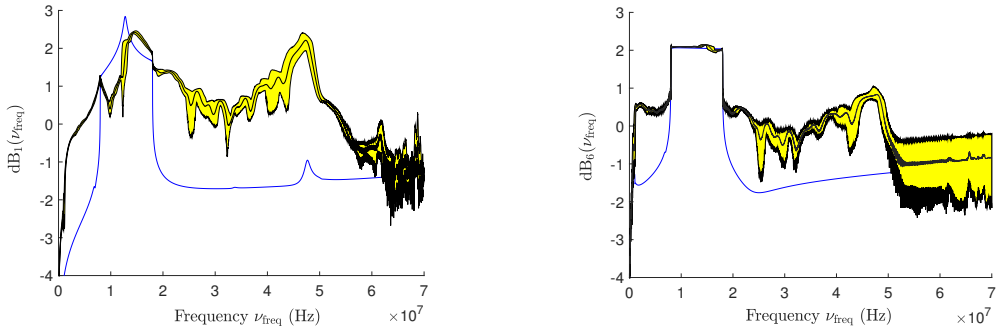


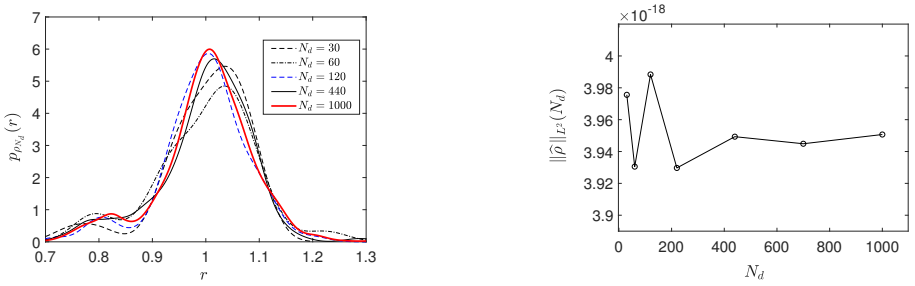
Figure 17: Reference stochastic solution for $N_d = 1\,000$: for Obs_j, $j = 1$ (left figure) and $j = 6$ (right figure), graph of the deterministic function $\nu_{\mathrm{freq}} \mapsto \mathrm{dB}_{\mathrm{lin},j}(\nu_{\mathrm{freq}})$ (blue thin line), and graph of the confidence region (yellow region with black thin lines for the upper and lower envelopes) of the random function $\nu_{\mathrm{freq}} \mapsto \mathrm{dB}_j(\nu_{\mathrm{freq}})$. The black thick line inside the yellow region is the estimate of mean value $\nu_{\mathrm{freq}} \mapsto E\{\mathrm{dB}_j(\nu_{\mathrm{freq}})\}$.

which $\underline{\widehat{\rho}}_{N_d} = E\{\widehat{\rho}\}$ is estimated using N_d realizations. We then have $E\{\rho_{N_d}\} = 1$ and the standard deviations are, for $N_d = 30$, 60, 120, 440, and 1 000, respectively, 0.080, 0.094, 0.091, 0.087, and 0.087. Figure 18-(b) shows the graph $N_d \mapsto \|\widehat{\rho}\|_{L^2}(N_d)$. These results show that the convergence with respect to N_d is reasonably obtained for $N_d = 1\,000$, that is the value selected for the reference.

(ii) Predictions using the PLoM without constraint. For $N_d = 120$ and using the nonlinear stochastic computational model, for each one of the two observations and for the probability level $p_c = 0.98$, Fig. 19 shows the confidence region of the random function $\nu_{\mathrm{freq}} \mapsto \mathrm{dB}_j(\nu_{\mathrm{freq}})$ estimated using $n_{\mathrm{MC}} = 1\,000$ additional realizations generated with the PLoM without constraint. These confidence regions are almost identical to those of the reference stochastic solution shown in Fig. 17 (see also the discussion given in Section 12.7).

12.5. Analysis of the role played by constraints on the probability distribution of random control parameter $\mathbf{W}$

As previously, for simplifying the writing, $\mathbf{W}^{(\nu)}$ is simply noted $\mathbf{W}$. For $N_d = 120$, $n_{\mathrm{MC}} = 1\,000$, and using *algo* 1, Fig. 20-(a) shows the decrease in the error function $i \mapsto \mathrm{err}_R(i)$ while Fig. 20-(b) shows the growth of function $i \mapsto \mathrm{err}_W(i)$ for $i > 24$, proving that the constraint on $\mathbf{W}$ must be taken into account.



(a) pdf of the normalized random residue ρ_{N_d}

(b) L^2-norm of the random residue $\widehat{\rho}$

Figure 18: Reference stochastic solution: (a) evolution of the pdf $r \mapsto p_{\rho_{N_d}}(r)$ of ρ_{N_d} as a function of the number N_d of realizations in the training dataset; (b) $N_d \mapsto \|\widehat{\rho}\|_{L^2}(N_d)$ of the random residual $\widehat{\rho}$.

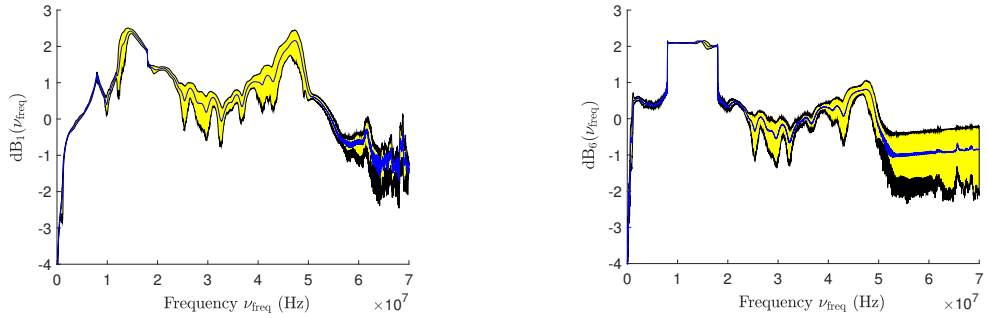
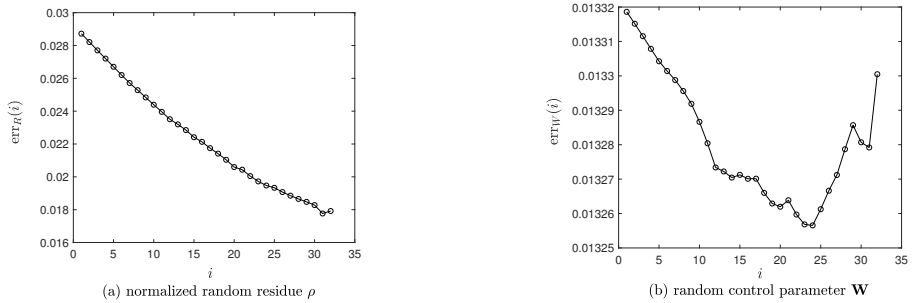


Figure 19: For $N_d = 120$, for $n_{\mathrm{MC}} = 1\,000$ additional realizations generated by the PLoM without constraints, for Obs_j, $j = 1$ (left figure) and $j = 6$ (right figure), graph of the confidence region (yellow region with thin lines for the upper and lower envelopes) of the random function $\nu_{\mathrm{freq}} \mapsto \mathrm{dB}_j(\nu_{\mathrm{freq}})$. The thick line inside the yellow region is the estimate of mean value $\nu_{\mathrm{freq}} \mapsto E\{\mathrm{dB}_j(\nu_{\mathrm{freq}})\}$.



(a) normalized random residue ρ

(b) random control parameter $\mathbf{W}$

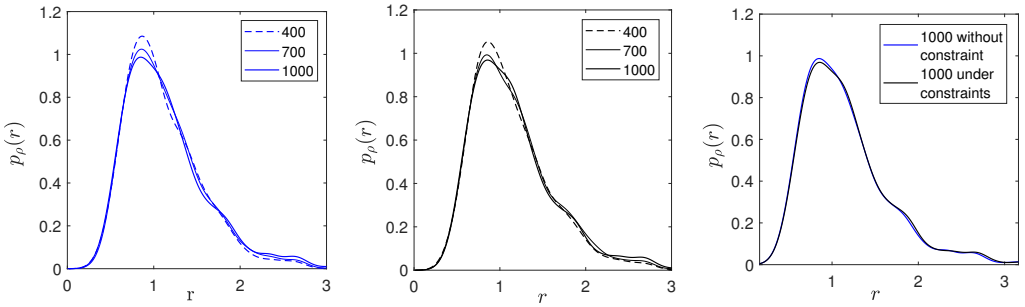
Figure 20: For $N_d = 120$ and with $n_{\mathrm{MC}} = 1\,000$ for the PLoM computation under the constraint on ρ but without constraint on $\mathbf{W}$ (*algo* 1): (a) function $i \mapsto \mathrm{err}_R(i)$ and (b) function $i \mapsto \mathrm{err}_W(i)$.

12.6. Results of the PLoM without and under constraints for a given dimension of the training dataset

The training dataset has $N_d = 120$ realizations and $n_{\mathrm{MC}} \gg N_d$ additional realizations are generated with the PLoM under constraints using algorithm *algo* 3.

(i) Figure 21 shows the pdf $r \mapsto p_\rho(r)$ of ρ as a function of the number n_{MC} of additional realizations generated with the PLoM without any constraint and under the constraints. The analysis of these figures shows a good convergence of the pdf of ρ with respect to n_{MC}.

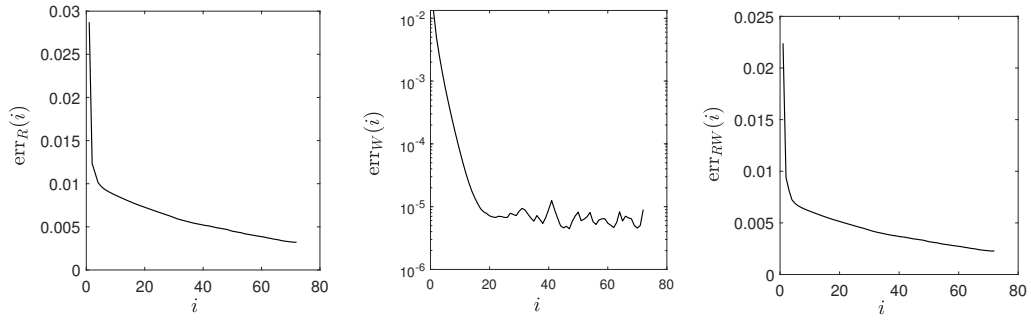
(ii) The case $N_d = 120$ is analyzed with $n_{\mathrm{MC}} = 1\,000$ additional realizations generated with the PLoM under constraints. Figure 22 shows the evolution of the error functions $i \mapsto \mathrm{err}_R(i)$, $i \mapsto \mathrm{err}_W(i)$, and $i \mapsto \mathrm{err}_{RW}(i)$. There is a local minimum for $i_{\mathrm{opt}} = 72$ such that $\mathrm{err}_{RW}(72) = 0.0032$, for which $\mathrm{err}_R(72) = 8.9 \times 10^{-6}$ and $\mathrm{err}_W(72) = 0.0023$.



(a) PLoM without constraint (b) PLoM under constraints (c) PLoM without and under constraints

Figure 21: For $N_d = 120$ and using *algo* 3, graph of pdf $r \mapsto p_\rho(r)$ of ρ as a function of n_{MC}, estimated with the PLoM without constraint (a), under constraints (b), and for $n_{\text{MC}} = 1\,000$, comparison of the estimates obtained with the PLoM without and under constraints (c).

Figure 23 shows the pdf $w \mapsto p_{W_2}(w)$ and the pdf $w \mapsto p_{W_5}(w)$. In each figure is plotted the estimated pdf, on the



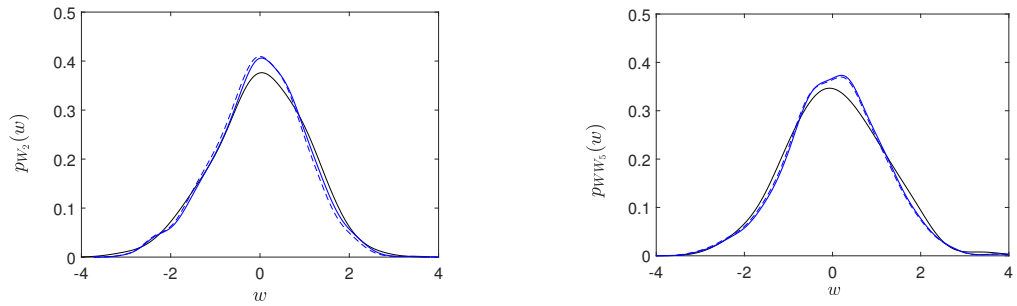
(a) Error function err_R (b) Error function err_W (c) Error function err_{RW}

Figure 22: For $N_d = 120$ with $n_{\text{MC}} = 1\,000$, and using *algo* 3, evolution of the error functions $i \mapsto \text{err}_R(i)$ (a), $i \mapsto \text{err}_W(i)$ (b), and $i \mapsto \text{err}_{RW}(i)$ (c).

one hand with the $N_d = 120$ realizations from the training dataset, on the other hand with the $n_{\text{MC}} = 1\,000$ additional realizations (learning) estimated with the PLoM without constraint and with the PLoM under the constraints on ρ and $\mathbf{W}$ by using *algo* 3. The figures show that the estimation performed with the PLoM without constraint is already good. Indeed, Fig. 22-(b) shows that the error $\text{err}_W(1) = 0.013$ at iteration $i = 1$, which corresponds to PLoM without constraint, is already small. So, for this case, the introduction of the constraint on $\mathbf{W}$ is not decisive, although it allows for improving the little that there is to improve. The mean value of $\widehat{\rho}$ is 6.35×10^{-7}, its standard deviation is 2.74×10^{-7}, and $\|\widehat{\rho}\,\|_{L^2} = 6.95 \times 10^{-7}$. In order to illustrate the distribution of the additional realizations generated by the PLoM under constraints, Fig. 24 displays the $n_{\text{MC}} = 1\,000$ additional realizations of components of random vector $\mathbf{X}^c = (\mathbf{Q}^c, \mathbf{W}^c)$. The probability distribution of (W_1^c, W_2^c, Q_1^c) is concentrated while the one for (W_4^c, W_{10}^c, Q_1^c) is not. For $N_d = 120$ and using the nonlinear stochastic computational model, for each one of the two observations, and for the probability level $p_c = 0.98$, Fig. 25 shows the confidence region of the random function $\nu_{\text{freq}} \mapsto \text{dB}_j(\nu_{\text{freq}})$ estimated using $n_{\text{MC}} = 1\,000$ additional realizations generated with the PLoM under constraints. These confidence regions are almost identical to those of the reference stochastic solution shown in Fig. 17.

12.7. Discussion

Similarly to the CFD application, the results obtained for this application devoted to the nonlinear computational stochastic dynamics of elastic solids show that the PLoM method without constraints gives good results, which was expected for but not at this level of quality. The use of the proposed extension, consisting in constraining the PLoM method by the stochastic PDE allows to verify the convergence and the robustness of the algorithms. The examination



(a) pdf of W_2 (b) pdf of W_5

Figure 23: For $N_d = 120$, pdf $w \mapsto p_{W_2}(w)$ (a) and pdf $w \mapsto p_{W_5}(w)$ (b), estimated with the $N_d = 120$ realizations (prior model) from the training dataset (black thin solid line), with $n_{\text{MC}} = 1\,000$ additional realizations (learning) estimated with the PLoM without constraint (blue dashed line) and under the constraints by using *algo* 3 (blue thick solid line).

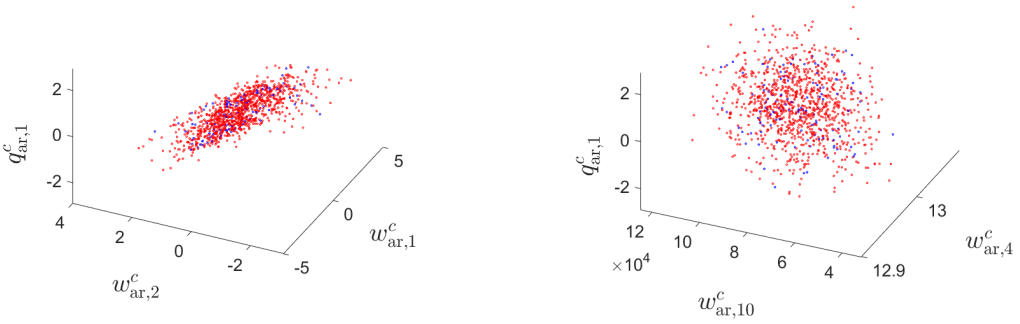


Figure 24: 3D-plot of realizations $(w^c_{\text{ar},1}, w^c_{\text{ar},2}, q^c_{\text{ar},1})$ of (W^c_1, W^c_2, Q^c_1) (left figure) and $(w^c_{\text{ar},4}, w^c_{\text{ar},10}, q^c_{\text{ar},1})$ of (W^c_4, W^c_{10}, Q^c_1) (right figure), computed with the PLoM under constraints: $n_{\text{MC}} = 1\,000$ additional realizations (red symbols) and $N_d = 120$ points from the training dataset (blue symbols).

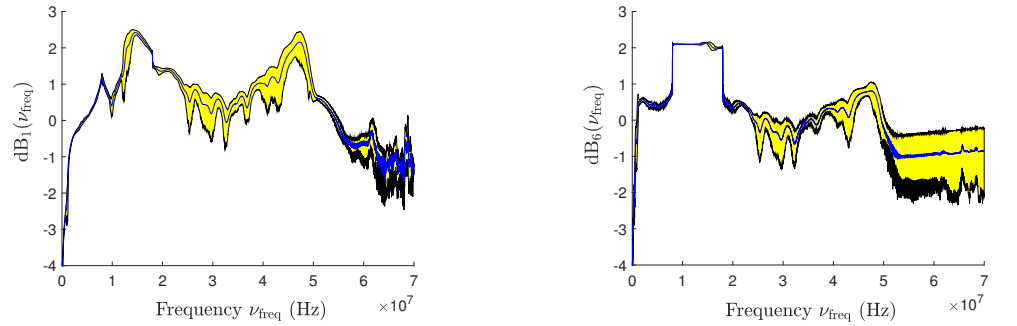


Figure 25: For $N_d = 120$, for $n_{\text{MC}} = 1\,000$ additional realizations generated by the PLoM under constraints, for Obs_j, $j = 1$ (left figure) and $j = 6$ (right figure), graph of the confidence region (yellow region with thin lines for the upper and lower envelopes) of the random function $\nu_{\text{freq}} \mapsto \text{dB}_j(\nu_{\text{freq}})$. The thick line inside the yellow region is the estimate of mean value $\nu_{\text{freq}} \mapsto E\{\text{dB}_j(\nu_{\text{freq}})\}$.

of the confidence domains of the stochastic responses in the frequency domain does not show any significant differences (compare Fig. 19 with Fig. 25). This application therefore also provides an additional validation of the PLoM method without constraint.

13. Conclusion and discussion

A novel extension of the PLoM method has been presented. It makes it possible to generate a large number of time-dependent additional realizations that verify the stochastic PDE used to generate the training dataset. This is paramount for the case where only a small number of solutions can be secured from the numerical model due to computational cost. The method is general and can be applied to linear and nonlinear time-dependent stochastic boundary value problems. Its implementation requires specialized algorithms related to the Karhunen-Loève expansions of high-dimension nonstationary stochastic processes, in order to circumvent potential difficulties associated with RAM limitations, thus providing an efficient alternative to I/O (Input-Output) on external memory. Taking into account the scalar constraint related to the random residual of the stochastic PDE, a novel statistical formulation and its implementation have been developed for estimating the gradient of the potential that appears in the drift vector of the reduced nonlinear Itô stochastic differential equation, which is the MCMC generator. The prior probability model of the vector-valued random control parameter of the stochastic boundary value problem is given and must not be modified by the PLoM under the constraint related to the random residual. In order to guarantee this property, we have had to introduce a vector constraint on the random control parameters. The results obtained through the presented applications in the field of nonlinear computational fluid and solid stochastic dynamics, allow the following conclusions to be drawn.

- The PLoM constrained by PDE provides further validation of the PLoM method without constraint, proposed in [30] for which a complete mathematical analysis has been proposed in [32]. Although the obtained results concerning the PLoM without constraint were expected, they are particularly good and interesting in terms of statistical methods of learning with small data.

- The PLoM constrained by stochastic PDE allows a direct control of the random residual of the stochastic PDE and thus one is assured that the probability measure constructed by PLoM under constraint, checks better the stochastic boundary value problem that is considered. This novel approach makes it possible to improve learning from small data when the random residual of the stochastic PDE computed with the PLoM without constraint is not sufficiently small. We observe this for the CFD application for which the PLoM under constraints gives excellent results and improve the PLoM without constraint. It should also be noted that the robustness of the PLoM algorithms under constraints allows for maintaining the quality of the results given by the PLoM without constraint when these results are already good, as can be seen in application 3 devoted to nonlinear solid dynamics.

Acknowledgments

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Agreement No. HR00111990032 and by FASTMATH, a SciDAC Institute (Scientific Discovery through Advanced Computing) funded by the U.S. Department of Energy (DOE), Office of Science, Advanced Scientific Computing Research (ASCR).

Appendix A. Störmer-Verlet algorithm for solving Eqs. (53) to (55)

The ISDE defined by Eqs. (53) to (55) is solved for $r \in [0, r_{\max}]$ with $r_{\max} = (\ell_0 + n_{\text{MC}} \times M_0)\,\Delta r$ in which Δr is the sampling step, where ℓ_0 is chosen in order that the solution of Eqs. (53) to (55) has reached the stationary regime, and where M_0 allows for controlling the number of sampling steps between two consecutive realizations for obtaining a quasi-independence of the computed realizations. Taking as the origin of ℓ, the value ℓ_0, for $\ell = 0, 1, \ldots, n_{\text{MC}} \times M_0$, we consider the sampling points $r_\ell = \ell\,\Delta r$ and the following notations: $[\mathcal{Z}_\ell] = [\mathcal{Z}_\lambda(r_\ell)]$, $[\boldsymbol{\mathcal{X}}_\ell] = [\boldsymbol{\mathcal{X}}_\lambda(r_\ell)]$, and $[\mathbf{W}^{\text{wien}}_\ell] = [\mathbf{W}^{\text{wien}}(r_\ell)]$. The Störmer-Verlet scheme is written, for $\ell = 0, 1, \ldots, n_{\text{MC}} \times M_0$, as

$$[\mathcal{Z}_{\ell+\frac{1}{2}}] = [\mathcal{Z}_\ell] + \frac{\Delta r}{2}\,[\boldsymbol{\mathcal{X}}_\ell]\,,$$

$$[\boldsymbol{\mathcal{X}}_{\ell+1}] = \frac{1-\beta}{1+\beta}\,[\boldsymbol{\mathcal{X}}_\ell] + \frac{\Delta r}{1+\beta}\,[\boldsymbol{\mathcal{L}}_{\ell+\frac{1}{2}}] + \frac{\sqrt{f_0}}{1+\beta}\,[\Delta \mathbf{W}^{\text{wien}}_{\ell+1}]\,[a]\,,$$

$$[\mathcal{Z}_{\ell+1}] = [\mathcal{Z}_{\ell+\frac{1}{2}}] + \frac{\Delta r}{2}\,[\boldsymbol{\mathcal{X}}_{\ell+1}]\,,$$

with the initial condition defined by Eq. (55), where $\beta = f_0\,\Delta r\,/4$, and where $[\boldsymbol{\mathcal{L}}_{\ell+\frac{1}{2}}]$ is the $\mathbb{M}_{\nu,m}$-valued random variable such that $[\boldsymbol{\mathcal{L}}_{\ell+\frac{1}{2}}] = [\mathcal{L}_\lambda([\mathcal{Z}_{\ell+\frac{1}{2}}])] = [L_\lambda([\mathcal{Z}_{\ell+\frac{1}{2}}]\,[g]^T)]\,[a]$. In the above equation, $[\Delta\mathbf{W}^{\text{wien}}_{\ell+1}]$ is a random variable with values in $\mathbb{M}_{\nu,N_d}$, in which the increment $[\Delta\mathbf{W}^{\text{wien}}_{\ell+1}] = [\mathbf{W}^{\text{wien}}(r_{\ell+1})] - [\mathbf{W}^{\text{wien}}(r_\ell)]$. The increments are statistically independent. For all $\alpha = 1,\ldots,\nu$ and for all $j = 1,\ldots,N_d$, the real-valued random variables $\big\{[\Delta\mathbf{W}^{\text{wien}}_{\ell+1}]_{\alpha j}\big\}_{\alpha j}$ are independent, Gaussian, second-order, and centered random variables such that $E\big\{[\Delta\mathbf{W}^{\text{wien}}_{\ell+1}]_{\alpha j}[\Delta\mathbf{W}^{\text{wien}}_{\ell+1}]_{\alpha' j'}\big\} = \Delta r\,\delta_{\alpha\alpha'}\,\delta_{jj'}$.

References

[1] K. Mills, M. Spanner, I. Tamblyn, Deep learning and the schrödinger equation, Physical Review A 96 (4) (2017) 042113. doi:10.1103/PhysRevA.96.042113.

[2] J.-X. Wang, J.-L. Wu, H. Xiao, Physics-informed machine learning approach for reconstructing reynolds stress modeling discrepancies based on DNS data, Physical Review Fluids 2 (3) (2017) 034603. doi:10.1103/PhysRevFluids.2.034603.

[3] A. P. Singh, S. Medida, K. Duraisamy, Machine-learning-augmented predictive modeling of turbulent separated flows over airfoils, AIAA Journal (2017) 2215–2227doi:10.2514/1.J055595.

[4] J.-L. Wu, H. Xiao, E. Paterson, Physics-informed machine learning approach for augmenting turbulence models: A comprehensive framework, Physical Review Fluids 3 (7) (2018) 074602. doi:10.1103/PhysRevFluids.3.074602.

[5] X. Yang, S. Zafar, J.-X. Wang, H. Xiao, Predictive large-eddy-simulation wall modeling via physics-informed neural networks, Physical Review Fluids 4 (3) (2019) 034602. doi:10.1103/PhysRevFluids.4.034602.

[6] R. Swischuk, L. Mainini, B. Peherstorfer, K. Willcox, Projection-based model reduction: Formulations for physics-based machine learning, Computers & Fluids 179 (2019) 704–717. doi:10.1016/j.compfluid.2018.07.021.

[7] M. Raissi, P. Perdikaris, G. E. Karniadakis, Machine learning of linear differential equations using gaussian processes, Journal of Computational Physics 348 (2017) 683–693. doi:10.1016/j.jcp.2017.07.050.

[8] J. Sirignano, K. Spiliopoulos, DGM: A deep learning algorithm for solving partial differential equations, Journal of Computational Physics 375 (2018) 1339–1364. doi:10.1016/j.jcp.2018.08.029.

[9] M. Raissi, G. E. Karniadakis, Hidden physics models: Machine learning of nonlinear partial differential equations, Journal of Computational Physics 357 (2018) 125–141. doi:10.1016/j.jcp.2017.11.039.

[10] M. Raissi, P. Perdikaris, G. E. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, Journal of Computational Physics 378 (2019) 686–707. doi:10.1016/j.jcp.2018.10.045.

[11] B. Schölkopf, A. Smola, K. Müller, Kernel principal component analysis, in: W. Gerstner, A. Germond, M. Hasler, J. Nicoud (Eds.), Artificial Neural Networks ICANN 97, Vol. 1327 of Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 1997, pp. 583–588.

[12] V. Vapnik, The Nature of Statistical Learning Theory, Springer, New York, 2000. doi:10.1007/978-1-4757-3264-1.

[13] C. C. Aggarwal, C. Zhai, Mining Text Data, Springer Science & Business Media, 2012. doi:10.1007/978-1-4614-3223-4.

[14] A. S. Dalalyan, A. B. Tsybakov, Sparse regression learning by aggregation and Langevin Monte-Carlo, Journal of Computer and System Sciences 78 (5) (2012) 1423–1443. doi:10.1016/j.jcss.2011.12.023.

[15] K. P. Murphy, Machine Learning: A Probabilistic Perspective, MIT press, 2012.

[16] M.-F. F. Balcan, V. Feldman, Statistical active learning algorithms, in: C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, K. Q. Weinberger (Eds.), Advances in Neural Information Processing Systems 26, Curran Associates, Inc., 2013, pp. 1295–1303.
URL http://papers.nips.cc/paper/5101-statistical-active-learning-algorithms.pdf

[17] G. James, D. Witten, T. Hastie, R. Tibshirani, An Introduction to Statistical Learning, Vol. 112, Springer, 2013.

[18] X. Dong, E. Gabrilovich, G. Heitz, W. Horn, N. Lao, K. Murphy, T. Strohmann, S. Sun, W. Zhang, Knowledge vault: A web-scale approach to probabilistic knowledge fusion, Proceedings of the 20th ACM SIGKDD International Conference on Knowledge discovery and data mining (2014) 601–610doi:10.1145/2623330.2623623.

[19] Z. Ghahramani, Probabilistic machine learning and artificial intelligence, Nature 521 (7553) (2015) 452–459. doi:10.1038/nature14541.

[20] J. Taylor, R. J. Tibshirani, Statistical learning and selective inference, Proceedings of the National Academy of Sciences 112 (25) (2015) 7629–7634. doi:10.1073/pnas.1507583112.

[21] D. Jones, M. Schonlau, W. Welch, Efficient global optimization of expensive black-box functions, Journal of Global Optimization 13 (4) (1998) 455–492. doi:10.1023/A:1008306431147.

[22] X. Du, W. Chen, Sequential optimization and reliability assessment method for efficient probabilistic design, ASME Journal of Mechanical Design 126 (2) (2004) 225–233. doi:10.1115/1.1649968.

[23] N. Queipo, R. Haftka, W. Shyy, T. Goel, R. Vaidyanathan, K. Tucker, Surrogate-based analysis and optimization, Progress in Aerospace Science 41 (1) (2005) 1–28. doi:10.1016/j.paerosci.2005.02.001.

[24] R. Byrd, G. Chin, W. Neveitt, J. Nocedal, On the use of stochastic Hessian information in optimization methods for machine learning, SIAM Journal of Optimization 21 (3) (2011) 977–995. doi:10.1137/10079923X.

[25] M. Eldred, Design under uncertainty employing stochastic expansion methods, International Journal for Uncertainty Quantification 1 (2) (2011) 119–146. doi:10.1615/Int.J.UncertaintyQuantification.v1.i2.20.

[26] W. Yao, X. Chen, W. Luo, M. vanTooren, J. Guo, Review of uncertainty-based multidisciplinary design optimization methods for aerospace vehicles, Progress in Aerospace Sciences 47 (6) (2011) 450–479. doi:10.1016/j.paerosci.2011.05.001.

[27] T. Homem-de Mello, G. Bayraksan, Monte Carlo sampling-based methods for stochastic optimization, Surveys in Operations Researh and Management Science 19 (1) (2014) 56–85. doi:10.1016/j.sorms.2014.05.001.

[28] V. Papadopoulos, G. Soimiris, D. Giovanis, M. Papadrakakis, A neural network-based surrogate model for carbon nanotubes with geometric nonlinearities, Computer Methods in Applied Mechanics and Engineering 328 (2018) 411–430. doi:10.1016/j.cma.2017.09.010.
[29] D. G. Giovanis, M. D. Shields, Data-driven surrogates for high dimensional models using gaussian process regression on the grassmann manifold, Computer Methods in Applied Mechanics and Engineering 370 (2020) 113269. doi:10.1016/j.cma.2020.113269.
[30] C. Soize, R. Ghanem, Data-driven probability concentration and sampling on manifold, Journal of Computational Physics 321 (2016) 242–258. doi:10.1016/j.jcp.2016.05.044.
[31] C. Soize, R. Ghanem, C. Safta, X. Huan, Z. P. Vane, J. C. Oefelein, G. Lacaze, H. N. Najm, Q. Tang, X. Chen, Entropy-based closure for probabilistic learning on manifolds, Journal of Computational Physics 388 (2019) 528–533. doi:10.1016/j.jcp.2018.12.029.
[32] C. Soize, R. Ghanem, Probabilistic learning on manifolds, Foundations of Data Science 2 (3) (2020) 279–307. doi:10.3934/fods.2020013.
[33] C. Soize, R. Ghanem, Polynomial chaos representation of databases on manifolds, Journal of Computational Physics 335 (2017) 201–221. doi:10.1016/j.jcp.2017.01.031.
[34] C. Soize, R. Ghanem, Physics-constrained non-gaussian probabilistic learning on manifolds, International Journal for Numerical Methods in Engineering 121 (1) (2020) 110–145. doi:10.1002/nme.6202.
[35] C. Soize, R. Ghanem, C. Desceliers, Sampling of bayesian posteriors with a non-gaussian probabilistic learning on manifolds from a small dataset, Statistics and Computing 30 (5) (2020) 1433–1457. doi:10.1007/s11222-020-09954-6.
[36] R. Ghanem, C. Soize, Probabilistic nonconvex constrained optimization with fixed number of function evaluations, International Journal for Numerical Methods in Engineering 113 (4) (2018) 719–741. doi:10.1002/nme.5632.
[37] R. Ghanem, C. Soize, C. Safta, X. Huan, G. Lacaze, J. C. Oefelein, H. N. Najm, Design optimization of a scramjet under uncertainty using probabilistic learning on manifolds, Journal of Computational Physics 399 (2019) 108930. doi:10.1016/j.jcp.2019.108930.
[38] C. Soize, R. Ghanem, C. Safta, X. Huan, Z. P. Vane, J. C. Oefelein, G. Lacaze, H. N. Najm, Enhancing model predictability for a scramjet using probabilistic learning on manifolds, AIAA Journal 57 (1) (2019) 365–378. doi:10.2514/1.J057069.
[39] R. Ghanem, C. Soize, C. Thimmisetty, Optimal well-placement using probabilistic learning, Data-Enabled Discovery and Applications 2 (1) (2018) 4,1–16. doi:10.1007/s41688-017-0014-x.
[40] C. Soize, Design optimization under uncertainties of a mesoscale implant in biological tissues using a probabilistic learning algorithm, Computational Mechanics 62 (3) (2018) 477–497. doi:10.1007/s00466-017-1509-x.
[41] C. Farhat, R. Tezaur, T. Chapman, P. Avery, C. Soize, Feasible probabilistic learning method for model-form uncertainty quantification in vibration analysis, AIAA Journal 57 (11) (2019) 4978–4991. doi:10.2514/1.J057797.
[42] C. Soize, C. Farhat, Probabilistic learning for modeling and quantifying model-form uncertainties in nonlinear computational mechanics, International Journal for Numerical Methods in Engineering 117 (2019) 819–843. doi:10.1002/nme.5980.
[43] J. Guilleminot, J. E. Dolbow, Data-driven enhancement of fracture paths in random composites, Mechanics Research Communications 103 (2020) 103443. doi:10.1016/j.mechrescom.2019.103443.
[44] R. Ghanem, C. Soize, L. Mehrez, V. Aitharaju, Probabilistic learning and updating of a digital twin for composite material systems, International Journal for Numerical Methods in Engineering (2020) 1–17doi:10.1002/nme.6430.
[45] M. Arnst, C. Soize, K. Bulthies, Computation of sobol indices in global sensitivity analysis from samll data sets by probabilistic learning on manifolds, International Journal for Uncertainty Quantification online (2020). doi:10.1615/Int.J.UncertaintyQuantification.2020032674.
[46] C. Soize, A. Orcesi, Machine learning for detecting structural changes from dynamic monitoring using the probabilistic learning on manifolds, Structure and Infrastructure Engineering Journal online (2020) 1–13. doi:10.1080/15732479.2020.1811991.
[47] R. Coifman, S. Lafon, A. Lee, M. Maggioni, B. Nadler, F. Warner, S. Zucker, Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps, PNAS 102 (21) (2005) 7426–7431. doi:10.1073/pnas.0500334102.
[48] I. M. Gelfand, N. I. Vilenkin, Les Distributions. Tome 4. Application de l'Analyse Harmonique, Dunod, 1967. doi:10.1016/0375-9474(67)90547-7.
[49] G. H. Golub, C. F. Van Loan, Matrix Computations, Second Edition, Johns Hopkins University Press, Baltimore and London, 1993.
[50] J. L. Doob, Stochastic processes, John Wiley & Sons, New York, 1953.
[51] P. Krée, C. Soize, Mathematics of Random Phenomena, Reidel Pub. Co, 1986, (first published by Bordas in 1983 and also published by Springer Science & Business Media in 2012).
[52] C. Soize, Polynomial chaos expansion of a multimodal random vector, SIAM-ASA Journal on Uncertainty Quantification 3 (1) (2015) 34–60. doi:10.1137/140968495.
[53] A. Bowman, A. Azzalini, Applied Smoothing Techniques for Data Analysis: The Kernel Approach With S-Plus Illustrations, Vol. 18, Oxford University Press, Oxford: Clarendon Press, New York, 1997. doi:10.1007/s001800000033.
[54] A. Bhattacharyya, On the measures of divergence between two statistical populations defined by their probability distributions, Bulletin of the Calcultta Mathematical Society 35 (1943) 99–109.
[55] S. Kullback, R. A. Leibler, On information and sufficiency, The Annals of Mathematical Statistics 22 (1) (1951) 79–86. doi:10.1214/aoms/1177729694.
[56] J. N. Kapur, H. K. Kesavan, Entropy Optimization Principles with Applications, Academic Press, San Diego, 1992.
[57] N. Agmon, Y. Alhassid, R. D. Levine, An algorithm for finding the distribution of maximal entropy, Journal of Computational Physics 30 (2) (1979) 250–258. doi:10.1016/0021-9991(79)90102-5.
[58] B. Nayroles, G. Touzot, P. Villon, Generalizing the finite element method: diffuse approximation and diffuse elements, Computational mechanics 10 (5) (1992) 307–318. doi:10.1007/BF00364252.
[59] T. Belytschko, Y. Krongauz, D. Organ, M. Fleming, P. Krysl, Meshless methods: an overview and recent developments, Computer methods in applied mechanics and engineering 139 (1-4) (1996) 3–47. doi:10.1016/S0045-7825(96)01078-X.
[60] C. A. Duarte, J. T. Oden, H-p clouds, an h-p meshless method, Numerical Methods for Partial Differential Equations: An International Journal 12 (6) (1996) 673–705. doi:10.1002/(SICI)1098-2426(199611)12:6¡673::AID-NUM3¿3.0.CO;2-P.
[61] P. Breitkopf, A. Rassineux, G. Touzot, P. Villon, Explicit form and efficient computation of mls shape functions and their derivatives, International Journal for Numerical Methods in Engineering 48 (3) (2000) 451–466. doi:10.1002/(SICI)1097-0207(20000530)48:3¡451::AID-NME892¿3.0.CO;2-1.

[62] A. Rassineux, P. Villon, J.-M. Savignat, O. Stab, Surface remeshing by local hermite diffuse interpolation, International Journal for numerical methods in Engineering 49 (1-2) (2000) 31–49. doi:10.1002/1097-0207(20000910/20)49:1/23.0.CO;2-6.
[63] X. Zhang, K. Z. Song, M. W. Lu, X. Liu, Meshless methods based on collocation with radial basis functions, Computational mechanics 26 (4) (2000) 333–343. doi:10.1007/s004660000181.
[64] O. Le Maître, O. M. Knio, Spectral Methods for Uncertainty Quantification: With Applications to Computational Fluid Dynamics, Springer Science & Business Media, 2010.
[65] H. Bijl, D. Lucor, S. Mishra, C. Schwab, Uncertainty Quantification in Computational Fluid Dynamics, Vol. 92, Springer Science & Business Media, Heidelberg, 2013. doi:10.1007/978-3-319-00885-1.
[66] M. Griebel, T. Dornseifer, T. Neunhoeffer, Numerical Simulation in Fluid Dynamics: a Practical Introduction, Society for Industrial and Applied Mathematics, 1998. doi:10.1137/1.9780898719703.
[67] M. A. Hopcroft, W. D. Nix, T. W. Kenny, What is the young's modulus of silicon?, Journal of microelectromechanical systems 19 (2) (2010) 229–238. doi:10.1109/JMEMS.2009.2039697.
[68] J. Guilleminot, C. Soize, On the statistical dependence for the components of random elasticity tensors exhibiting material symmetry properties, Journal of Elasticity 111 (2) (2013) 109–130. doi:10.1007/s10659-012-9396-z.
[69] J. Guilleminot, C. Soize, Stochastic model and generator for random fields with symmetry properties: application to the mesoscopic modeling of elastic random media, Multiscale Modeling & Simulation (A SIAM Interdisciplinary Journal) 11 (3) (2013) 840–870. doi:10.1137/120898346.
[70] C. Soize, Uncertainty Quantification. An Accelerated Course with Advanced Applications in Computational Engineering, Springer, New York, 2017. doi:10.1007/978-3-319-54339-0.
[71] C. Soize, Non gaussian positive-definite matrix-valued random fields for elliptic stochastic partial differential operators, Computer Methods in Applied Mechanics and Engineering 195 (1-3) (2006) 26–64. doi:10.1016/j.cma.2004.12.014.
[72] R. Ohayon, C. Soize, Structural Acoustics and Vibration: Mechanical Models, Variational Formulations and Discretization, Academic Press, San Diego, London, 1998.