HAL
open science

Probabilistic learning on manifolds (PLoM) with partition

Christian Soize, Roger Ghanem

To cite this version:

Christian Soize, Roger Ghanem. Probabilistic learning on manifolds (PLoM) with partition. International Journal for Numerical Methods in Engineering, 2022, 123 (1), pp.268-290. 10.1002/nme.6856 . hal-03381363

HAL Id: hal-03381363

https://hal.science/hal-03381363v1

Submitted on 27 Oct 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Probabilistic Learning on Manifolds (PLoM) with partition

Christian Soize[a,*], Roger Ghanem[b]

[a] Univ Gustave Eiffel, MSME UMR 8208, 5 bd Descartes, 77454 Marne-La-Vallée, France
[b] University of Southern California, Viterbi School of Engineering, 210 KAP Hall, Los Angeles, CA 90089, United States

Abstract

The probabilistic learning on manifolds (PLoM) introduced in 2016 [1] has solved difficult supervised problems for the "small data" limit where the number N of points in the training set is small. Many extensions have since been proposed, making it possible to deal with increasingly complex cases. However, the performance limit has been observed and explained for applications for which N is very small and for which the dimension of the diffusion-map basis is close to N. For these cases, we propose a novel extension based on the introduction of a partition in independent random vectors. We take advantage of this development to present improvements of the PLoM such as a simplified algorithm for constructing the diffusion-map basis and a new mathematical result for quantifying the concentration of the probability measure in terms of a probability upper bound. The analysis of the efficiency of this extension is presented through two applications.

Keywords: probabilistic learning, PLoM, partition in independent random vectors, machine learning, uncertainty quantification,

1. Introduction

(i) About the PLoM. The PLoM (probabilistic learning on manifolds) method was proposed in 2016 [1] as a complementary approach to existing methods in machine learning for sampling underlying distributions on manifolds [2, 3, 4, 5, 6, 7, 8, 9]. It allows for solving unsupervised and supervised problems under uncertainty for which the training sets are small. This situation is encountered in many problems of physics and engineering sciences with expensive function evaluations. The exploration of the admissible solution space in these situations is thus hampered by available computational resources. The PLoM was successfully adapted to tackle these challenges for several related problems including nonconvex optimization under uncertainty [10, 11] and the calculation of Sobol's indices [12].

(ii) Brief discussion on the hypotheses and the objectives of PLoM. Hypotheses. The PLoM approach starts from a training set $\mathcal{D}_d$ made up of a relatively small number, N, of data points. For the supervised case, it is assumed that the training set is related to an underlying stochastic manifold related to a $\mathbb{R}^n$-valued random variable $\mathbf{X} = (\mathbf{Q}, \mathbf{W})$ in which $\mathbf{X}$, $\mathbf{Q}$ (quantity of interest), and

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

*Preprint submitted to International Journal for Numerical Methods in Engineering, 2021, doi:**********/nme.*********, October 10, 2021*

$\mathbf{W}$ (control parameter) are $\mathbb{R}^n$-, $\mathbb{R}^{n_q}$-, and $\mathbb{R}^{n_w}$-valued random variables defined on a probability space $(\Theta, \mathcal{T}, \mathcal{P})$ with $n = n_q + n_w$. Let $\mathbf{U}$ (uncontrolled parameter) be another $\mathbb{R}^{n_u}$- valued random variable defined on $(\Theta, \mathcal{T}, \mathcal{P})$. Random variable $\mathbf{Q}$ is written as $\mathbf{Q} = \mathbf{f}(\mathbf{U}, \mathbf{W}) = \mathbf{F}(\mathbf{W})$ in which the measurable mapping $\mathbf{f}$ is not explicitly known (unknown) and $\mathbf{F}$ is such that $\mathbf{F} = \mathbf{f}(\mathbf{U}, \cdot)$ is a random mapping. The probability distributions of the vector-valued random variables $\mathbf{W}$ and $\mathbf{U}$ are assumed to be given. The stochastic manifold is defined by the unknown random graph $\{\mathbf{w}, \mathbf{F}(\mathbf{w})\}$ for $\mathbf{w}$ belonging to an admissible set C_w that is the support of the probability distribution of $\mathbf{W}$. 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 non-Gaussian probability measure of $\mathbf{X}$ is concentrated in a region of $\mathbb{R}^n$ for which the only available information is the cloud of the points of the training set.

What are the construction objectives of PLoM and its characteristics?.

- The construction of PLoM has been specifically developed for small training sets and for arbitrary non-Gaussian probability measures.
- From training set $\mathcal{D}_d$, the PLoM method makes it possible to generate the learned set $\mathcal{D}_{\text{ar}}$ whose $n_{\text{ar}} \gg N$ points (learned realizations) are generated by the non-Gaussian probability measure that is estimated from the training set.
- The estimate of unknown non-Gaussian probability measure cannot be performed from the training set by using an arbitrary estimator. It must be parameterized in a manner that permits convergence to any probability measure as its number of points in the training set goes towards infinity. The PLoM method therefore does not only consist in generating learned points that belong to the region in which the measure is concentrated, but also allows these learned 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 set guarantees that this required fundamental property is satisfied (see [1] and in particular, Section 5.3 of [13]).
- The generation of the learned set is performed while preserving the concentration of the probability measure, defined by the clouds of points of the training set. With PLoM, this concentration is preserved thanks to the use of the diffusion-maps basis that allows enriching the available information from the training set. The preservation of the concentration is quantified by the calculation of a L^2 distance on $(\Theta, \mathcal{T}, \mathcal{P})$ between the learned set and the training set [13].
- Using the learned set, PLoM allows for carrying out any conditional statistics such as $\mathbf{w} \mapsto E\{\mathbf{Q}|\mathbf{W} = \mathbf{w}\}$ from C_w in $\mathbb{R}^{n_q}$, and consequently, to directly construct metamodels in a probabilistic framework.

(iii) Improvement of the PLoM method for certain applications. Since its introduction in 2016, extensions of the original method [1] have been developed in order to address increasingly complex problems for the case of small data: sampling of Bayesian posteriors with a non-Gaussian probabilistic learning on manifolds in very high dimension [14], physics-constrained non-Gaussian probabilistic learning on manifolds, for instance, to integrate information coming from experimental measurements during the learning [15], probabilistic learning on manifolds constrained by nonlinear partial differential equations for small data [16]. During this period, a number of applications were addressed making it possible to refine the method, to validate it, and to better assess and relax its limitations. However, some challenges have remained. These are cases where the number of points (realizations) in the training set is very small and for which the dimension of the subspace generated by the diffusion-map basis is very close to this number. In

this case, the PLoM may not be more efficient than a standard MCMC algorithm that is agnostic to any concentration of the probability measure. One possible way to improve the PLoM for these very challenging cases is to perform a partition of the random vector of which the training set is the realizations, into statistically independent groups in a non-Gaussian framework. In this manner, statistical knowledge about the data set, beyond its localization to a manifold, is relied upon to enhance information extraction and representation. One difficulty with the standard methods that allow a partition into independent components to be carried out (see paragraph (iv) below and its included references) is that they typically divide-up available samples into separate groups, for which each group consists of a smaller number of samples. These types of approaches are not suitable to the present setting given the already small number of points in the training set. A more useful approach, which is adopted in this paper, results in groups that are each equal in size to the number of samples in the training set, but for which the dimension of the diffusion-map basis is significantly reduced. This approach relies on an extension of the PLoM method that we present in this paper.

Remark. The reason of this difficulty, which can appear when the number of points in the training set is close to the subspace generated by the diffusion-maps basis, is the following one. In this case, the dimension of the manifold is close to the dimension of the support of the probability measure. The diffusion-maps basis, which makes it possible to characterize the geometry of this support, then has a similar dimension. The projection on the diffusion-maps basis therefore appears as a simple change of vector basis in the data space. There is therefore no gain compared to a standard MCMC generator and in such a case, the concentration of the probability measure, defined by the points of the training set, deteriorates. There is therefore a great interest in trying to preserve the concentration of the probability measure for these situations. If the training set is made up of heterogeneous data, that is, if there are data subsets in the training set, which are statistically independent, then finding a partition becomes very efficient if it can be identified. The quantification of the gain obtained will be performed by using the L^2-distance criterion introduced in [13], recalled in Section 4.5. In this paper, we also introduce a new probability criterion (see Section 5.7), which gives the probability upper bound of the measure of concentration. As we will see, these two criteria show that PLoM without partition allows to preserve the concentration and that the preservation of this concentration is further improved by using PLoM with partition. When the difficulty mentioned above appears, then PLoM with partition helps to preserve concentration. Examples of such challenging problems can be found in many fields of engineering. Let us cite for example: (1) in the field of solid mechanics, the numerical simulation models of the macroscopic behavior of heterogeneous materials, which require the introduction of random media at the mesoscopic scale (see an illustration of this problem in Section 7), (2) in the field of fluid mechanics, CFD models that make it possible to estimate the unsteady aerodynamic forces exerted on the structure of stadium roofs.

(iv) A novel extension of the PLOM method to get around the difficulty. One of the ingredients for this extension of the PLoM method is the construction of a partition in non-Gaussian independent random vectors, assuming that it exists. Indeed, there may very well be applications for which the partition yields a single group, identical to the initial random vector. For such cases the present approach of PLoM with partitions affords no further reduction. Concerning the construction of a partition of independent random vectors, a popular method for testing the statistical independence of the ν components of a random vector from a given set of N realizations is the use of the frequency distribution [17] coupled with the use of the Pearson chi-squared (χ^2) test [18, 19]. For a relatively large value of ν and a relatively small value of N, such an approach

does not give sufficiently accurate results. In addition, even when this type of method permits testing for independence, the need remains for a fast algorithm for constructing the optimal partition. Independent component analysis (ICA) [20, 21, 22, 23, 24, 25, 26, 27] is a method that consists of extracting independent source signals as a linear mixture of mutually statistically dependent signals, and is often used for source-separation problems. The fundamental hypothesis that is introduced in the ICA methodology is that the observed vector-valued signal is a linear transformation of statistically independent real-valued signals (that is to say, is a linear transformation of a vector-valued signal whose components are mutually independent) and the objective of the ICA algorithms is to identify the best linear operator. In this paper, for the PLoM with partition, we use the procedure proposed in [28], which is an ICA by mutual information and which does not use the construction of a linear transformation. This information-theoretic algorithm permits the identification of an optimal partition in terms of independent random vectors for any non-Gaussian vector in high dimension, which is defined by a relatively small number N of realizations.

(v) Organization of the paper. In Section 2, we present the supervised problem and training set. In Section 3, we introduce the notations for the principal component analysis of random vector $\mathbf{X}$, which is the first step of the PLoM method. Section 4 is devoted to a summary of the PLoM method with no group (No-Group PLoM), which is necessary for understanding the presentation of the PLoM with partition. Section 5 deals with the PLoM analysis with group (With-Group PLoM). Section 6 and Section 7 are both devoted to the presentation of two applications. Finally, a discussion of the method is presented in Section 8. Complements on the probabilistic models used for the two applications are given in Appendices A and B.

(vi) Novelties presented in the paper. The main novelty is the development of the PLoM methodology with partition. We also propose an adapted algorithm for identifying the optimal values of the hyperparameters related to the construction of the reduced-order diffusion-map basis. For covering the cases for which the normalization introduced by the PLoM is lost with the use of a partition, we introduce constraints by using the Kullback-Leibler minimum cross-entropy principle [15]. The quantification of the concentration of the probability measure for the PLoM, which is performed with the distance introduced in [13], is extended for the PLoM with partition, and is completed by a result formulated in terms of a probability upper bound of the measure of concentration.

2. Supervised problem and training set

The mapping $(\mathbf{w}, \mathbf{u}) \mapsto \mathbf{f}(\mathbf{w}, \mathbf{u})$ on $\mathbb{R}^{n_w} \times \mathbb{R}^{n_u}$ with values in $\mathbb{R}^{n_q}$, introduced in Section 1-(ii), represents the solution of a mathematical/computational model. The $\mathbb{R}^{n_w}$-valued random variable $\mathbf{W}$ is the control parameter and the $\mathbb{R}^{n_u}$-valued random variable $\mathbf{U}$ is the non-controlled parameter, defined on a probability space $(\Theta, \mathcal{T}, \mathcal{P})$. The random vectors $\mathbf{W}$ and $\mathbf{U}$ are assumed to be statistically independent and they are generally non-Gaussian. The probability distributions $P_{\mathbf{W}}(d\mathbf{w}) = p_{\mathbf{W}}(\mathbf{w})\, d\mathbf{w}$ and $P_{\mathbf{U}}(d\mathbf{u}) = p_{\mathbf{U}}(\mathbf{u})\, d\mathbf{u}$ are defined by the probability density functions $p_{\mathbf{W}}$ and $p_{\mathbf{U}}$ with respect to the Lebesgue measures $d\mathbf{w}$ and $d\mathbf{u}$ on $\mathbb{R}^{n_w}$ and $\mathbb{R}^{n_u}$. The $\mathbb{R}^{n_q}$-valued random variable $\mathbf{Q} = \mathbf{f}(\mathbf{W}, \mathbf{U})$ represents the quantity of interest (QoI), which is defined on $(\Theta, \mathcal{T}, \mathcal{P})$. For instance, the computational model can be the finite element discretization of an elliptic stochastic boundary value problem as the one presented in Section 7, which involves an

elasticity problem for a random medium. In this application, $\mathbf{U}$ is related to the space discretization of the fourth-order tensor-valued non-Gaussian elasticity random field of the heterogeneous material at mesoscale (random coefficients of the partial differential operator) while the components of $\mathbf{W}$ are related to the spatial correlation lengths of the random medium and to the level of statistical fluctuations in it. The components of random vector $\mathbf{Q}$ will then correspond to the displacement field at some given space points. It is assumed that $N \geq 3$ independent realizations $\{\mathbf{q}_d^j, j = 1, \ldots, N\}$ of $\mathbf{Q}$ have been computed such that $\mathbf{q}_d^j = \mathbf{f}(\mathbf{w}_d^j, \mathbf{u}_d^j)$ in which $\{\mathbf{w}_d^j, j = 1, \ldots, N\}$ and $\{\mathbf{u}_d^j, j = 1, \ldots, N\}$ are N independent realizations of $\mathbf{W}$ and $\mathbf{U}$ (subscript d refers to the training set). We then consider the random variable $\mathbf{X}$ with values in $\mathbb{R}^n$, such that $\mathbf{X} = (\mathbf{Q}, \mathbf{W})$ with $n = n_q + n_w$. The training set (initial data set) related to random vector $\mathbf{X}$ is then made up of the N independent realizations $\{\mathbf{x}_d^j, j = 1, \ldots, N\}$ in which $\mathbf{x}_d^j = (\mathbf{q}_d^j, \mathbf{w}_d^j) \in \mathbb{R}^n$ (note that $\mathbf{U}$ is not included in $\mathbf{X}$). Since generally the data pertains to heterogeneous features with potentially wildly distinct supports, it is assumed that the training set has been suitably scaled for the purpose of computational statistics. Let us assume that the measurable mapping $\mathbf{f}$ is such that the conditional probability distribution $P_{\mathbf{Q}|\mathbf{W}}(d\mathbf{q}|\mathbf{w})$ given $\mathbf{W} = \mathbf{w}$ admits a conditional probability density function. It can be deduced (see [15]) that the probability distribution $P_{\mathbf{X}}(d\mathbf{x})$ of $\mathbf{X}$ admits a density $\mathbf{x} \mapsto p_{\mathbf{X}}(\mathbf{x})$ with respect to the Lebesgue measure $d\mathbf{x}$ on $\mathbb{R}^n$. As recalled in Section 1-(ii), PLoM [1, 13] allows for generating the learned set $\mathcal{D}_{\text{ar}}$ made up of $N_{\text{ar}} \gg N$ realizations $\{\mathbf{x}_{\text{ar}}^\ell, \ell = 1, \ldots, N_{\text{ar}}\}$ that allows for deducing $\{(\mathbf{q}_{\text{ar}}^\ell, \mathbf{w}_{\text{ar}}^\ell) = \mathbf{x}_{\text{ar}}^\ell, \ell = 1, \ldots, N_{\text{ar}}\}$ without using the computational model, but using only the training set $\mathcal{D}_d$ (subscript ar refers to the learned set).

3. Introducing the notations for the principal component analysis (PCA) of random vector X

The first step of the PLoM method consists in performing a PCA of random vector $\mathbf{X}$. For presenting the PLoM with partition, we need to introduce the notations related to this PCA and it is the aim of this section. Let $\underline{\mathbf{x}}_d \in \mathbb{R}^n$ and $[C_{\mathbf{X}}] \in \mathbb{M}_n^+$ be the mean vector and the covariance matrix of $\mathbf{X}$ estimated with the training set. Let $\mu_1 \geq \mu_2 \geq \ldots \geq \mu_\nu > 0$ be the ν largest eigenvalues and let $\boldsymbol{\varphi}^1, \ldots, \boldsymbol{\varphi}^\nu$ be the associated orthonormal eigenvectors of $[C_{\mathbf{X}}]$. The integer $\nu \leq n$ is such that, for a given $\varepsilon_{\text{PCA}} > 0$, we have $\text{err}_{\text{PCA}}(\nu) = 1 - \sum_{\alpha=1}^{\nu} \mu_\alpha / \text{tr}[C_{\mathbf{X}}] \leq \varepsilon_{\text{PCA}}$. The PCA of $\mathbf{X}$ allows for representing $\mathbf{X}$ by $\mathbf{X}^\nu$ such that $\mathbf{X}^\nu = \underline{\mathbf{x}}_d + [\Phi]\,[\mu]^{1/2}\,\mathbf{H}$ such that $E\{\|\mathbf{X} - \mathbf{X}^\nu\|^2\} \leq \varepsilon_{\text{PCA}}\, E\{\|\mathbf{X}\|^2\}$, in which $[\Phi] = [\boldsymbol{\varphi}^1 \ldots \boldsymbol{\varphi}^\nu] \in \mathbb{M}_{n,\nu}$ such that $[\Phi]^T\,[\Phi] = [I_\nu]$ and $[\mu]$ is the diagonal $(\nu \times \nu)$ matrix such that $[\mu]_{\alpha\beta} = \mu_\alpha \delta_{\alpha\beta}$. From a numerical point of view, if $N < n$, then matrix $[C_{\mathbf{X}}]$ is not estimated and $\nu, [\mu]$, and $[\Phi]$ are directly computed using a thin SVD [29] of the matrix whose N columns are $(\mathbf{x}_d^j - \underline{\mathbf{x}}_d)$ for $j = 1, \ldots, N$. The $\mathbb{R}^\nu$-valued random variable $\mathbf{H}$ is obtained by projection, $\mathbf{H} = [\mu]^{-1/2}\,[\Phi]^T\,(\mathbf{X} - \underline{\mathbf{x}}_d)$, and its N independent realizations $\{\boldsymbol{\eta}_d^j, j = 1, \ldots, N\}$ are such that $\boldsymbol{\eta}_d^j = [\mu]^{-1/2}\,[\Phi]^T\,(\mathbf{x}_d^j - \underline{\mathbf{x}}_d) \in \mathbb{R}^\nu$. Using $\{\boldsymbol{\eta}_d^j, j = 1, \ldots, N\}$, the estimates of the mean vector and the covariance matrix of $\mathbf{H}$ verify $\underline{\boldsymbol{\eta}}_d = \mathbf{0}_\nu$ and $[C_{\mathbf{H}}] = [I_\nu]$. We define the matrix $[\eta_d] = [\boldsymbol{\eta}_d^1 \ldots \boldsymbol{\eta}_d^N] \in \mathbb{M}_{\nu,N}$ whose columns are the N realizations of $\mathbf{H}$, which is such that

$$\|[\eta_d]\|^2 = \text{tr}\{[\eta_d]^T\,[\eta_d]\} = \sum_{j=1}^{N} \|\boldsymbol{\eta}_d^j\|^2 = \nu(N-1)\,. \tag{1}$$

4. Summary of the PLoM method with no group (No-Group PLoM) and its improvement

The PLoM method can be found in [1], for which a complete mathematical analysis is proposed in [13]. We present a summary of this method (PLoM with no group also called "No-Group PLoM") in order to introduce the notations and the developments that are required for the presentation of the PLoM with partition (also called "With-Group PLoM").

4.1. Nonparametric estimate of the pdf of $\boldsymbol{H}$

A modification of the multidimensional Gaussian kernel-density estimation method [30, 31, 32, 33] is used for constructing the nonparametric estimate $p_{\mathbf{H}}^{(N)}$ on $\mathbb{R}^\nu$ of the pdf $p_{\mathbf{H}}$ of random vector $\mathbf{H}$, which is written (see Theorem 3.1 of [13] for the convergence with respect to N) as

$$p_{\mathbf{H}}^{(N)}(\boldsymbol{\eta}) = \frac{1}{N}\sum_{j=1}^{N} \frac{1}{(\sqrt{2\pi}\,\widehat{s})^\nu}\, \exp\{-\frac{1}{2\widehat{s}^2}\|\frac{\widehat{s}}{s}\,\boldsymbol{\eta}_d^j - \boldsymbol{\eta}\|^2\} \quad , \quad \forall \boldsymbol{\eta} \in \mathbb{R}^\nu\,, \tag{2}$$

in which $s = (N(\nu+2)/4)^{-1/(\nu+4)}$ is the usual Silverman bandwidth (since $[C_{\mathbf{H}}] = [I_\nu]$, see for instance, [34]) and where $\widehat{s} = s\,(s^2 + (N-1)/N)^{-1/2}$ has been introduced in order that $\int_{\mathbb{R}^\nu} \boldsymbol{\eta}\, p_{\mathbf{H}}^{(N)}(\boldsymbol{\eta})\, d\boldsymbol{\eta} = 0_\nu$ and $\int_{\mathbb{R}^\nu} \boldsymbol{\eta}\otimes\boldsymbol{\eta}\; p_{\mathbf{H}}^{(N)}(\boldsymbol{\eta})\, d\boldsymbol{\eta} = [I_\nu]$.

4.2. Construction of a reduced-order diffusion-map basis (ROB-DM)

To identify the subset around which the points of the training set are concentrated, the PLoM relies on the diffusion-map method [35, 36]. The Gaussian kernel is used. Let $[K]$ and $[b]$ be the matrices such that, for all i and j in $\{1,\ldots,N\}$, $[K]_{ij} = \exp\{-(4\,\varepsilon_{\text{DM}})^{-1}\|\boldsymbol{\eta}_d^i - \boldsymbol{\eta}_d^j\|^2\}$ and $[b]_{ij} = \delta_{ij}\, b_i$ with $b_i = \sum_{j=1}^N [K]_{ij}$, in which $\varepsilon_{\text{DM}} > 0$ is a smoothing parameter (the non symmetric matrix $\mathbb{P} = [b]^{-1}[K] \in \mathbb{M}_N$ is the transition matrix of a Markov chain that yields the probability of transition in one step). The eigenvalues $\lambda_1,\ldots,\lambda_N$ and the associated eigenvectors $\boldsymbol{\psi}^1,\ldots,\boldsymbol{\psi}^N$ of the right-eigenvalue problem $[\mathbb{P}]\,\boldsymbol{\psi}^\alpha = \lambda_\alpha\,\boldsymbol{\psi}^\alpha$ are such that $1 = \lambda_1 > \lambda_2 \geq \ldots \geq \lambda_N$ and are computed by solving the generalized eigenvalue problem $[K]\,\boldsymbol{\psi}^\alpha = \lambda_\alpha\,[b]\,\boldsymbol{\psi}^\alpha$ with the normalization $< [b]\,\boldsymbol{\psi}^\alpha, \boldsymbol{\psi}^\beta >= \delta_{\alpha\beta}$. The eigenvector $\boldsymbol{\psi}^1$ associated with $\lambda_1 = 1$ is a constant vector. For a given integer $\kappa \geq 0$, the diffusion-map basis $\{\mathbf{g}^1,\ldots,\mathbf{g}^\alpha,\ldots,\mathbf{g}^N\}$ is a vector basis of $\mathbb{R}^N$ defined by $\mathbf{g}^\alpha = \lambda_\alpha^\kappa\,\boldsymbol{\psi}^\alpha$. For a given integer m with $3 \leq m \leq N$, the reduced-order diffusion-map basis of order m is defined as the family $\{\mathbf{g}^1,\ldots,\mathbf{g}^m\}$ that is represented by the matrix $[g_m] = [\mathbf{g}^1\ldots\mathbf{g}^m] \in \mathbb{M}_{N,m}$ with $\mathbf{g}^\alpha = (g_1^\alpha,\ldots,g_N^\alpha)$ and $[g_m]_{\ell\alpha} = g_\ell^\alpha$. This ROB-DM depends on two parameters, ε_{DM} and m, which have to be identified. It is proven in [13], that the PLoM method does not depend on κ that can therefore be chosen to 0.

It should be noted that, if $\nu = 1$, then there is no reason to use the ROB-DM and in this case, we propose to take $m = N$ and $[g_N] = [I_N]$. For non-trivial applications analyzed with the PLoM without partition, we always have $\nu > 1$ and even, $1 \ll \nu \leq n$. However for the PLoM method with partition, optimal partitions can be found for which some groups may have dimension 1 (hence the consideration of possible cases of this type).

4.3. Novel algorithm for identifying the optimal values ε_{o} *and* m_{o} *of* ε_{DM} *and* m

Let us assume that $\nu \geq 2$. For estimating the optimal values ε_{o} of ε_{DM} and m_{o} of m, the criterion of the eigenvalues given in Section 5.2 of [13] must be satisfied for the PLoM method

to be applicable. This criterion can be summarized as follows. We have to find the value $m_{\rm o} \leq N$ of m and the smallest value $\varepsilon_{\rm o} > 0$ of $\varepsilon_{\rm DM}$ such that

$$1 = \lambda_1 > \lambda_2(\varepsilon_{\rm o}) \simeq \ldots \simeq \lambda_{m_{\rm o}}(\varepsilon_{\rm o}) \gg \lambda_{m_{\rm o}+1}(\varepsilon_{\rm o}) \geq \ldots \geq \lambda_N(\varepsilon_{\rm o}) > 0\,, \tag{3}$$

with an amplitude jump equal to an order of magnitude (a factor 10 as demonstrated in [13]) between $\lambda_{m_{\rm o}}(\varepsilon_{\rm o})$ and $\lambda_{m_{\rm o}+1}(\varepsilon_{\rm o})$. This property means that we have to find $m_{\rm o} \leq N$ and the smallest positive value $\varepsilon_{\rm o}$ in order (i) to have $\lambda_2(\varepsilon_{\rm o}) < 1$ (one must not have several eigenvalues in the neighborhood of 1) and (ii) to obtain a plateau for $\lambda_2(\varepsilon_{\rm o})$ to $\lambda_{m_{\rm o}}(\varepsilon_{\rm o})$ with a jump of amplitude 10 between $\lambda_{m_{\rm o}}(\varepsilon_{\rm o})$ and $\lambda_{m_{\rm o}+1}(\varepsilon_{\rm o})$. A further in-depth analysis makes it possible to state the following new criterion and algorithm to easily estimate $\varepsilon_{\rm o}$ and $m_{\rm o}$. Let $\varepsilon_{\rm DM} \mapsto \text{Jump}(\varepsilon_{\rm DM})$ be the function on $]0, +\infty[$ defined by $\text{Jump}(\varepsilon_{\rm DM}) = \lambda_{m_{\rm o}+1}(\varepsilon_{\rm DM})/\lambda_2(\varepsilon_{\rm DM})$. The algorithm is thus given in

Algorithm 1 Algorithm for estimating the optimal values $m_{\rm o}$ of m and $\varepsilon_{\rm o}$ of $\varepsilon_{\rm DM}$

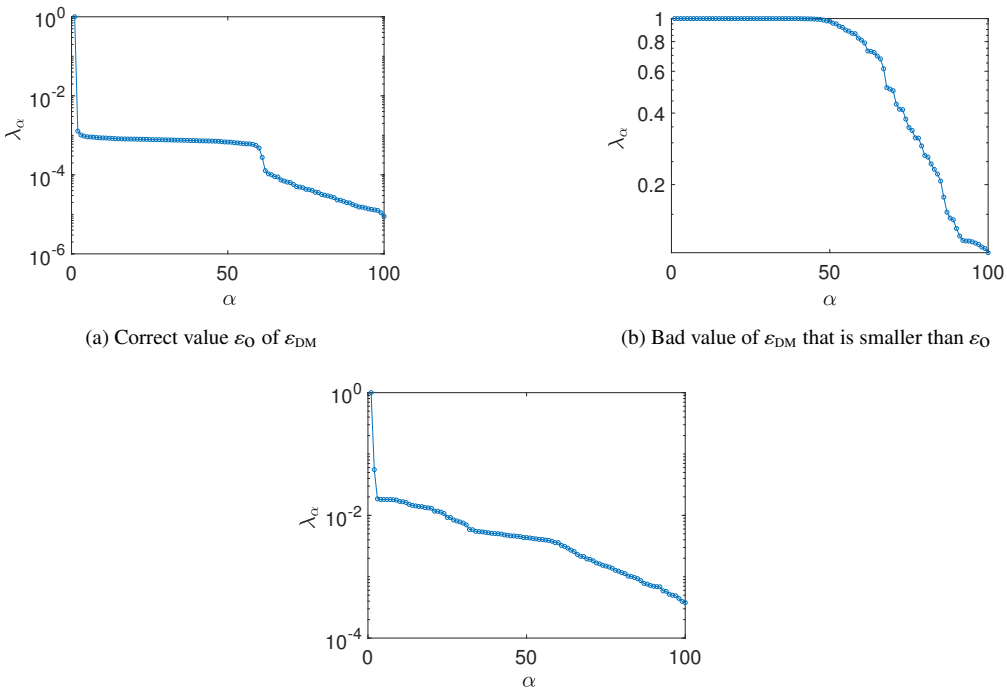
if $\nu = 1$ **then**
  Set the value of m to $m_{\rm o} = N$ and $[g_N] = [I_N]$.
end if
if $\nu \geq 2$ **then**
  Set the value of m to $m_{\rm o} = \nu + 1$.
  Identify the smallest possible value $\varepsilon_{\rm o}$ of $\varepsilon_{\rm DM}$ in order that $\text{Jump}(\varepsilon_{\rm o}) \leq 0.1$ and such that Eq. (3) be verified.
end if

Algorithm 1 and Figure 1 shows an illustration: we have $m_{\rm o} = \nu + 1 = 61$; the optimal value of $\varepsilon_{\rm DM}$ that satisfies the criteria is $\varepsilon_{\rm o} = 65$ and yields Figure 1a; if a smaller value than 65 is chosen, for instance the value 5, then there will be many eigenvalues close to 1 as shown in Figure 1b; if the smallest value for $\varepsilon_{\rm DM}$ is not selected, for example taking the value 100, then the plateau is not obtained as shown in Figure 1c. For these two bad values of $\varepsilon_{\rm DM}$, the calculated diffusion-map basis is not adapted to the PLoM procedure.

Remark concerning the choice of factor 10 *for the jump of the eigenvalues*. This choice is justified in Paper [13], which gives the mathematical results in support of PLoM (we refer the reader to Theorem 7.8 of this paper and its Lemmas 7.5, 7.6, and 7.7 on which the proof is based). In this paper, it is proven that Eq. (3) must hold for preserving the concentration of the probability measure, which is quantified by the square of the L^2-distance (see Eq. (4)), and which is estimated with the $N_{\rm ar} \gg N$ learned realizations. Note that there are two important hypotheses in Eq. (3) that must be verified: one is the existence of the plateau for $m \in \{2, \ldots, m_{\rm o}\}$ and the other one is the existence of a jump between $m = m_{\rm o}$ and $m = m_{\rm o} + 1$. In paper [13], it is proven that the plateau and the jump on the eigenvalues are directly related to the variations of the function $m \mapsto \varepsilon_{\rm DM}(m)$ (denoted as $m \mapsto \varepsilon_d(m)$ in [13]), which has to be rapidly decreasing in m in the neighborhood of $m_{\rm o}$ by the lower integer values and which has to remain much lower than 1 for $m \geq m_{\rm o}$ (see Fig. 1 of [13]). When Eq. (3) holds, then $\varepsilon_{\rm DM}(m) \ll 1$ for $m \geq m_{\rm o}$, that is a fundamental property used for proving Theorem 7.8. This property $\varepsilon_{\rm DM}(m) \ll 1$ for $m \geq m_{\rm o}$ has been quantified by choosing one order of magnitude for the jump of the eigenvalues.

4.4. Random matrices $[\boldsymbol{H}^N]$, $[\boldsymbol{H}^N_m]$, $[\boldsymbol{H}^N_{m_{\rm o}}]$, *and MCMC generator*

Let $\mathbf{H}^{(N)}$ be the $\mathbb{R}^\nu$-valued random variable defined on $(\Theta, \mathcal{T}, \mathcal{P})$ for which the pdf is $p^{(N)}_{\mathbf{H}}$ defined by Equation (2). Let $[\mathbf{H}^N]$ be the random matrix with values in $\mathbb{M}_{\nu,N}$ such that $[\mathbf{H}^N] =$



(a) Correct value ε_{O} of $\varepsilon_{\mathrm{DM}}$

(b) Bad value of $\varepsilon_{\mathrm{DM}}$ that is smaller than ε_{O}

(c) Bad value of $\varepsilon_{\mathrm{DM}}$ that is larger than ε_{O}

Figure 1: Illustration of the criterion effects defined by Equation (3) for identifying ε_{O}.

$[\mathbf{H}^1 \ldots \mathbf{H}^N]$ in which $\mathbf{H}^1, \ldots, \mathbf{H}^N$ are N independent copies of $\mathbf{H}^{(N)}$. It can be seen that $E\{\mathbf{H}^{(N)}\} = \mathbf{0}_\nu$ and $E\{\mathbf{H}^{(N)} \otimes \mathbf{H}^{(N)}\} = [I_\nu]$. Note that $\mathbf{H}^1, \ldots, \mathbf{H}^N$ are not taken as N independent copies of $\mathbf{H}$ whose pdf $p_{\mathbf{H}}$ is unknown, but are taken as N independent copies of $\mathbf{H}^{(N)}$ whose pdf $p_{\mathbf{H}}^{(N)}$ is known. The PLoM method introduces the $\mathbb{M}_{\nu,N}$-valued random matrix $[\mathbf{H}_m^N] = [\mathbf{Z}_m]\,[g_m]^T$ with $3 \leq m \leq N$, corresponding to a data-reduction representation of random matrix $[\mathbf{H}^N]$, in which $[g_m]$ is the ROB-DM and where $[\mathbf{Z}_m]$ is a $\mathbb{M}_{\nu,m}$-valued random matrix for which its probability measure $p_{[\mathbf{Z}_m]}([z])\,d[z]$ is explicitly described by Proposition 2 of [13]. In the PLoM method, the MCMC generator of random matrix $[\mathbf{Z}_m]$ belongs to the class of Hamiltonian Monte Carlo methods [37], is explicitly described in [1], and is mathematically detailed in Theorem 6.3 of [13]. For generating the learned set, the best probability measure of $[\,\mathbf{H}_m^N]$ is obtained for $m = m_{\mathrm{O}}$ and using the previously defined $[g_{m_{\mathrm{O}}}]$. For these optimal quantities m_{O} and $[g_{m_{\mathrm{O}}}]$, the generator allows for computing n_{MC} realizations $\{[\mathbf{z}_{\mathrm{ar}}^\ell], \ell = 1, \ldots, n_{\mathrm{MC}}\}$ of $[\mathbf{Z}_{m_{\mathrm{O}}}]$ and therefore, for deducing the n_{MC} realizations $\{[\boldsymbol{\eta}_{\mathrm{ar}}^\ell], \ell = 1, \ldots, n_{\mathrm{MC}}\}$ of $[\mathbf{H}_{m_{\mathrm{O}}}^N]$. The reshaping of matrix $[\boldsymbol{\eta}_{\mathrm{ar}}^\ell] \in \mathbb{M}_{\nu,N}$ allows for obtaining $N_{\mathrm{ar}} = n_{\mathrm{MC}} \times N$ additional realizations $\{\boldsymbol{\eta}_{\mathrm{ar}}^{\ell'}, \ell' = 1, \ldots, N_{\mathrm{ar}}\}$ of $\mathbf{H}$. These additional realizations allow for estimating converged statistics on $\mathbf{H}$ and then on $\mathbf{X}$, such as pdf, moments, or conditional expectation of the type $E\{\boldsymbol{\xi}(\mathbf{Q})\,|\,\mathbf{W} = \mathbf{w}_0\}$ for $\mathbf{w}_0$ given in $\mathbb{R}^{n_w}$ and for any given vector-valued function $\boldsymbol{\xi}$ defined on $\mathbb{R}^{n_q}$.

4.5. Quantifying the concentration of the probability measure of random matrix $[\boldsymbol{H}^N_{m_o}]$

In [13], for $3 \leq m \leq N$, we have introduced an L^2-distance $d_N(m)$ of random matrix $[\mathbf{H}^N_m]$ to matrix $[\eta_d]$ in order to quantify the concentration of the probability measure of random matrix $[\mathbf{H}^N_m]$, which is informed by the training set represented by matrix $[\eta_d]$. The square of this distance is defined by

$$d^2_N(m) = E\{\|[\mathbf{H}^N_m] - [\eta_d]\|^2\}/\|[\eta_d]\|^2 \,. \tag{4}$$

Let $\mathcal{M}_o = \{m_o, m_o + 1, \ldots, N\}$ in which m_o is the optimal value of m previously defined. Theorem 7.8 of [13] shows that $\min_{m\in\mathcal{M}_o} d^2_N(m) \leq 1 + m_o/(N-1) < d^2_N(N)$ which means that the PLoM method, for $m = m_o$ and $[g_{m_o}]$ is a better method than the usual one corresponding to $d^2_N(N) = 1 + N/(N-1)$. Using the n_{MC} realizations $\{[\boldsymbol{\eta}^\ell_{ar}], \ell = 1, \ldots, n_{MC}\}$ of $[\mathbf{H}^N_{m_o}]$, we have the estimate $d^2_N(m_o) \simeq (1/n_{MC}) \sum_{\ell=1}^{n_{MC}} \{\|[\boldsymbol{\eta}^\ell_{ar}] - [\eta_d]\|^2\}/\|[\eta_d]\|^2$.

5. PLoM analysis with group (With-Group PLoM)

In this section, for $\nu \geq 2$, we present the extension of the PLoM method for which statistically independent groups are constructed using an optimal partition of random vector $\mathbf{H}$.

5.1. Construction of the optimal partition of $\boldsymbol{H}$

From the training set $\{\boldsymbol{\eta}^j, j = 1, \ldots, N\}$, the optimal partition of $\mathbf{H} = (H_1, \ldots, H_\nu)$ is performed using the algorithm proposed in [28]. Such a partition is composed of n_p groups consisting in n_p mutually independent random vectors $\mathbf{Y}^1, \ldots, \mathbf{Y}^{n_p}$. Since $\mathbf{H}$ is a normalized random vector (zero mean vector and covariance matrix equal to the identity matrix), for $i = 1, \ldots, n_p$, $\mathbf{Y}^i$ is a normalized $\mathbb{R}^{\nu_i}$-valued random variable $\mathbf{Y}^i = (Y^i_1, \ldots, Y^i_{\nu_i}) = (H_{r^i_1}, \ldots, H_{r^i_{\nu_i}})$ in which $1 \leq r^i_1 < r^i_2 < \ldots < r^i_{\nu_i} \leq \nu$, with $\nu = \nu_1 + \ldots + \nu_{n_p}$, and where $Y^i_k = H_{r^i_k}$. Random vector $\mathbf{Y}^i$ is non-Gaussian and such that the estimate of its mean vector is $\underline{\boldsymbol{\eta}^i} = \mathbf{0}_{\nu_i}$ and the estimate of its covariance matrix is $[C_{\mathbf{Y}^i}] = [I_{\nu_i}]$. We then have $\mathbf{H} = perm(\mathbf{Y}^1, \ldots, \mathbf{Y}^{n_p})$ in which $perm$ is the permutation operator acting on the components of vector $\widetilde{\mathbf{H}} = (\mathbf{Y}^1, \ldots, \mathbf{Y}^{n_p})$ in order to reconstitute $\mathbf{H} = perm(\widetilde{\mathbf{H}})$. For each group i, the training set is represented by the matrix $[\eta^i_d] \in \mathbb{M}_{\nu_i,N}$ whose columns are the N realizations $\{\boldsymbol{\eta}^{i,j}_d, j = 1, \ldots, N\}$ of the $\mathbb{R}^{\nu_i}$-valued random variable $\mathbf{Y}^i$, which are deduced from an adapted extraction (due to the permutations) of the components of vectors $\{\boldsymbol{\eta}^j_d, j = 1, \ldots, N\}$. The partition is identified by constructing the function $i_{ref} \mapsto \tau(i_{ref})$ of the mutual information defined by Eq. (3.44) of [28] and then by deducing the optimal level i^o_{ref} defined by Eq. (3.46) of [28]. These two equations are recalled below.

(i) Mutual information of the random vectors related to the partition. The mutual information $i^\nu(\mathbf{Y}^1, \ldots, \mathbf{Y}^{n_p})$ between the random vectors $\mathbf{Y}^1, \ldots, \mathbf{Y}^{n_p}$ is defined by

$$i^\nu(\mathbf{Y}^1, \ldots, \mathbf{Y}^{n_p}) = -E\left\{\log\left(\frac{p_{\mathbf{Y}^1}(\mathbf{Y}^1) \times \ldots \times p_{\mathbf{Y}^{n_p}}(\mathbf{Y}^{n_p})}{p_{\mathbf{Y}^1,\ldots,\mathbf{Y}^{n_p}}(\mathbf{Y}^1, \ldots, \mathbf{Y}^{n_p})}\right)\right\} ,$$

in which the conventions $0 \log(0/a) = 0$ for $a \geq 0$ and $b \log(b/0) = +\infty$ for $b > 0$ are used, where $p_{\mathbf{Y}^i}$ is the pdf of $\mathbf{Y}^i$, and where $p_{\mathbf{Y}^1,\ldots,\mathbf{Y}^{n_p}}$ is the joint pdf of $\mathbf{Y}^1, \ldots, \mathbf{Y}^{n_p}$. Let $\mathbf{G}$ be the Gaussian second-order centered $\mathbb{R}^\nu$-valued random vector for which its covariance matrix is $[I_\nu]$. Consequently, its components are mutually independent. Applying to $\mathbf{G}$ the same partition that the one defined for $\mathbf{Y}$, we can write $\mathbf{G} = (\mathbf{G}^1, \ldots, \mathbf{G}^{n_p})$, and its mutual information is $i^\nu(\mathbf{G}^1, \ldots, \mathbf{G}^{n_p})$.

(ii) Optimal level $i^{\mathrm{o}}_{\mathrm{ref}}$. Let $i_{\mathrm{ref}} \geq 0$ be any fixed real value of the mutual information for two real-valued random variables. Let $\tau_\nu(i_{\mathrm{ref}})$ be the mutual information defined by

$$\tau_\nu(i_{\mathrm{ref}}) = 1 - \frac{i^\nu(\mathbf{Y}^1, \ldots, \mathbf{Y}^{n_p})}{i^\nu(\mathbf{G}^1, \ldots, \mathbf{G}^{n_p})} \,.$$

The optimal level $i^{\mathrm{o}}_{\mathrm{ref}}$ is such that

$$i^{\mathrm{o}}_{\mathrm{ref}} = \inf_{i_{\mathrm{ref}}} \{ \arg\max_{i_{\mathrm{ref}} \geq 0} \ \tau_\nu(i_{\mathrm{ref}}) \} \,.$$

(iii) Numerical aspects. For calculating the mutual information, the pdf are estimated by using the multidimensional Gaussian kernel density estimation method with the points of the training set. For each given i_{ref}, the partition is constructed using a graph theory algorithm (see [28]).

5.2. Use of the PLoM for each independent group

Let i be fixed in $\{1, \ldots, n_p\}$. The PLoM method (summarized in Section 4) is applied to the $\mathbb{R}^{\nu_i}$-valued random variable $\mathbf{Y}^i$ of the optimal partition $\mathbf{Y}^1, \ldots, \mathbf{Y}^{n_p}$ of $\mathbf{H} = perm(\mathbf{Y}^1, \ldots, \mathbf{Y}^{n_p})$. The parameters of the PLoM are thus the following.
1) The Silverman bandwidth is $s_i = (N(\nu_i + 2)/4)^{-1/(\nu_i+4)}$ (since $[C_{\mathbf{Y}^i}] = [I_{\nu_i}]$) and the modified bandwidth is $\widehat{s}_i = s_i \, (s_i^2 + (N-1)/N)^{-1/2}$.
2) Algorithm 1 is used. If $\nu_i = 1$, then $m_{i,\mathrm{o}} = N$ and $[g^i_N] = N$. If $\nu_i \geq 2$, the optimal parameter $m_{i,\mathrm{o}}$ of the dimension m_i of the ROB-DM^i is such that $m_{i,\mathrm{o}} = \nu_i + 1$. The optimal parameter $\varepsilon_{i,\mathrm{o}}$ of $\varepsilon_{i,\mathrm{DM}}$ is calculated as explained in Section 4.3. The ROB-DM^i of order $m_{i,\mathrm{o}}$ is represented by the matrix $[g^i_{m_{i,\mathrm{o}}}] \in \mathbb{M}_{N,m_{i,\mathrm{o}}}$.
3) The learned set of the random matrix $[\,\mathbf{Y}^{N,i}_{m_{i,\mathrm{o}}}\,] = [\mathbf{Z}^i_{m_{i,\mathrm{o}}}]\,[g^i_{m_{i,\mathrm{o}}}]^T$ is computed for $m_i = m_{i,\mathrm{o}}$ and by using $[g^i_{m_{i,\mathrm{o}}}]$. Finally, the n_{MC} realizations $\{[\boldsymbol{\eta}^{i,\ell}_{\mathrm{ar}}], \ell = 1, \ldots, n_{\mathrm{MC}}\}$ of $[\mathbf{Y}^{N,i}_{m_{i,\mathrm{o}}}]$ are computed with the MCMC generator and by reshaping, we obtain the $N_{\mathrm{ar}} = n_{\mathrm{MC}} \times N$ additional realizations $\{\boldsymbol{\eta}^{i,\ell'}_{\mathrm{ar}}, \ell' = 1, \ldots, N_{\mathrm{ar}}\}$.

5.3. Possible loss of normalization

Numerical experiments have been done for numerous cases with respect to the number of groups and the dimension of each group. These experiments have shown the following. In general, the mean value of $\mathbf{Y}^i$, estimated using the N_{ar} additional realizations $\{\boldsymbol{\eta}^{i,\ell'}_{\mathrm{ar}}, \ell' = 1, \ldots, N_{\mathrm{ar}}\}$, is sufficiently close to zero. Likewise, the estimate of the covariance matrix of $\mathbf{Y}^i$, which must be the identity matrix, is sufficiently close to a diagonal matrix. However, sometimes the diagonal entries of the estimated covariance matrix can be lower than 1 (for instance 0.5). Such a case can occur for relatively small value of ν_i (but it is not systematic because this type of behavior is application-dependent). Normalization can be recovered by imposing constraints in the PLoM method.

5.4. Constraints on the second-order moments of the components of $\mathbf{Y}^i$ if loss of normalization occurs

As explained in Section 5.3, if appropriate for group i, constraints $\{E\{(Y^i_k)^2\} = 1, k = 1, \ldots, \nu_i\}$ can readily be introduced in the PLoM. For that, we use the method and the iterative algorithm presented in Sections 5.5 and 5.6 of [15]. The method consists of constructing the generator using the PLoM for each independent group (see Section 5.2) and the Kullback-Leibler

minimum cross-entropy principle. The resulting optimization problem is formulated using Lagrange multipliers associated with the constraints. The optimal solution of the Lagrange multipliers is computed using an efficient iterative algorithm. At each iteration, the MCMC generator of the PLoM is used. The constraints are rewritten as

$$E\{\mathbf{h}^i(\mathbf{Y}^i)\} = \mathbf{b}^i\,, \tag{5}$$

in which the function $\mathbf{h}^i = (h^i_1, \ldots, h^i_{\nu_i})$ and the vector $\mathbf{b}^i = (b^i_1, \ldots, b^i_{\nu_i})$ are such that $h^i_k(\mathbf{Y}^i) = (Y^i_k)^2$ and $b^i_k = 1$ for k in $\{1, \ldots, \nu_i\}$. Equations (71) and (72) of [15] involve the Lagrange multiplier $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_{\nu_i}) \in \mathbb{R}^{\nu_i}$ associated with the constraints defined by Equation (5). These two equations, which define the nonlinear mapping $[u] \mapsto [L^i_{\boldsymbol{\lambda}}([u])]$ from $\mathbb{M}_{\nu_i,N}$ into $\mathbb{M}_{\nu_i,N}$ (drift of the Itô stochastic differential equation of the PLoM generator), have to be modified as follows. For $\alpha = 1, \ldots, \nu_i$, for $\ell = 1, \ldots, N$, and for $[u] = [\mathbf{u}^1 \ldots \mathbf{u}^N]$ in $\mathbb{M}_{\nu_i,N}$, we have

$$[L^i_{\boldsymbol{\lambda}}([u])]_{\alpha\ell} = \frac{1}{\rho_i(\mathbf{u}^\ell)} \frac{\partial \rho_i(\mathbf{u}^\ell)}{\partial u^\ell_\alpha} - 2\lambda_\alpha u^\ell_\alpha\,,$$

$$\rho_i(\mathbf{u}^\ell) = \frac{1}{N} \sum_{j=1}^{N} \frac{1}{(\sqrt{2\pi}\,\widehat{s}_i)^{\nu_i}} \exp\{-\frac{1}{2\widehat{s}_i^2} \|\frac{\widehat{s}_i}{s_i}\,\boldsymbol{\eta}^{i,j}_d - \mathbf{u}^\ell\|^2\}\,.$$

The iteration algorithm computes the sequence $\{\boldsymbol{\lambda}^\iota\}_{\iota\geq 1}$ that is convergent. If difficulties of convergence appear, a relaxation factor (less than 1) is introduced for computing $\boldsymbol{\lambda}^{\iota+1}$ as a function of $\boldsymbol{\lambda}^\iota$. For controlling the convergence as a function of iteration number ι, we use the error function $\iota \mapsto \mathrm{err}_i(\iota)$ defined by

$$\mathrm{err}_i(\iota) = \|\mathbf{b}^i - E\{\mathbf{h}^i(\mathbf{Y}^i_{\boldsymbol{\lambda}^\iota})\}\|/\|\mathbf{b}^i\|\,. \tag{6}$$

At each iteration ι, $E\{\mathbf{h}^i(\mathbf{Y}^i_{\boldsymbol{\lambda}^\iota})\}$ is estimated with the N_{ar} additional realizations deduced by reshaping of the n_{MC} realizations of the $\mathbb{M}_{\nu_i,N}$-valued random matrix $[\mathbf{Y}^{N,i}_{m_{i,\mathrm{o}}}(\boldsymbol{\lambda}^\iota)]$ that depends on $\boldsymbol{\lambda}^\iota$. These realizations are generated by the MCMC algorithm of the PLoM under the constraints.

5.5. Learned data set generated by With-Group PLoM

We have seen above (see Section 5.2 how the learned set $\{[\boldsymbol{\eta}^{i,\ell}_{\mathrm{ar}}], \ell = 1, \ldots, n_{\mathrm{MC}}\}$ of random matrix $[\mathbf{Y}^{N,i}_{m_{i,\mathrm{o}}}]$ are generated using With-Group PLoM for each group $i = 1, \ldots, n_p$ (using or not the constraints on the second-order moments of the components of $\mathbf{Y}^i$). From this information, we can directly deduce the learned set $\{[\boldsymbol{\eta}^{wg,\ell}_{\mathrm{ar}}], \ell = 1, \ldots, n_{\mathrm{MC}}\}$ of $[\mathbf{H}^{wg,N}_{\mathbf{m}_\mathrm{o}}]$ that corresponds to the concatenation with an adapted extraction of the rows (due to the permutations) of matrices $\{[\mathbf{Y}^{N,i}_{m_{i,\mathrm{o}}}], i = 1, \ldots, n_p\}$ and where $\mathbf{m}_\mathrm{o} = (m_{1,\mathrm{o}}, \ldots, m_{n_p,\mathrm{o}})$. We have introduced a superscript *wg* for distinguishing With-Group PLoM from No-Group PLoM. The reshaping of matrix $[\boldsymbol{\eta}^{wg,\ell}_{\mathrm{ar}}] \in \mathbb{M}_{\nu,N}$ allows for obtaining $N_{\mathrm{ar}} = n_{\mathrm{MC}} \times N$ additional realizations $\{\boldsymbol{\eta}^{wg,\ell'}_{\mathrm{ar}}, \ell' = 1, \ldots, N_{\mathrm{ar}}\}$ of $\mathbf{H}$, computed using With-Group PLoM.

5.6. Quantifying the concentration of the probability measure of random matrices $[\boldsymbol{H}^{wg,N}_{\boldsymbol{m}_\mathrm{o}}]$ and $\{[\boldsymbol{Y}^{N,i}_{m_\mathrm{o}}], i = 1, \ldots, n_p\}$

For $\mathbf{m}_\mathrm{o} = (m_{1,\mathrm{o}}, \ldots, m_{n_p,\mathrm{o}})$, the square of the distance of the random matrix $[\mathbf{H}^{wg,N}_{\mathbf{m}_\mathrm{o}}]$ to matrix $[\eta_d]$ is directly given by Equation (4) which is rewritten here as,

$$d^2_{wg,N}(\mathbf{m}_\mathrm{o}) = E\{\|[\mathbf{H}^{wg,N}_{\mathbf{m}_\mathrm{o}}] - [\eta_d]\|^2\}/\|[\eta_d]\|^2\,, \tag{7}$$

in which the mathematical expectation is estimated using the $n_{\rm MC}$ realizations $\{[\boldsymbol{\eta}_{\rm ar}^{wg,\ell}], \ell = 1, \ldots, n_{\rm MC}\}$. Using again Equation (4), for $i \in \{1, \ldots, n_p\}$, the square of the distance of random matrix $[\mathbf{Y}_{m_{i,\rm o}}^{N,i}]$ to matrix $[\eta_d^i]$ is given by

$$d_{i,N}^2(m_{i,\rm o}) = E\{\|[\mathbf{Y}_{m_{i,\rm o}}^{N,i}] - [\eta_d^i]\|^2\}/\|[\eta_d^i]\|^2 \,, \tag{8}$$

which is estimated using the $n_{\rm MC}$ realizations $\{[\boldsymbol{\eta}_{\rm ar}^{i,\ell}], \ell = 1, \ldots, n_{\rm MC}\}$. Equation (7) contains the information defined by Equation (8). Indeed it is easy to verify that we have the relation

$$d_{wg,N}^2(\mathbf{m}_{\rm o}) = \sum_{i=1}^{n_p} (\nu_i/\nu)\, d_{i,N}^2(m_{i,\rm o}) \,. \tag{9}$$

5.7. How to quantify the gain obtained by using With-Group PLoM instead of No-Group PLoM when $n_p > 1$

For a given application, the first method consists in numerically comparing the estimates of $d_{wg,N}^2(\mathbf{m}_{\rm o})$ defined by Equation (7) with $d_N^2(m_{\rm o})$ defined by Equation (4). If there is a gain, we must have

$$d_{wg,N}^2(\mathbf{m}_{\rm o}) < d_N^2(m_{\rm o}) \,. \tag{10}$$

This expected inequality for any applications for which $n_p > 1$ is reinforced by the second method, which is encapsulated by the following proposition.

Proposition 1 (Probability upper bound of the measure of concentration). *Let ε be a given real number such that $0 < \varepsilon < 1$. Let $d_N^2(m_{\rm o})$ be defined by Equation* (4) *for $m = m_{\rm o}$. We then have*

$$\text{Proba}\{\|[\boldsymbol{H}_{m_{\rm o}}^N] - [\eta_d]\|^2/\|[\eta_d]\|^2 \geq \varepsilon\} \leq d_N^2(m_{\rm o})/\varepsilon \,. \tag{11}$$

Let r be the positive real number (geometric mean) such that $r = \{\Pi_{i=1}^{n_p} d_{i,N}^2(m_{i,\rm o})\}^{1/n_p}$ in which $d_{i,N}^2(m_{i,\rm o})$ is defined by Equation (8)*. We then have*

$$\text{Proba}\{\|[\boldsymbol{H}_{\boldsymbol{m}_{\rm o}}^{wg,N}] - [\eta_d]\|^2/\|[\eta_d]\|^2 \geq \varepsilon\} \leq (r/\varepsilon)^{n_p} \,. \tag{12}$$

PROOF. (i) Using the Markov inequality to the left hand-side member of Equation (11) directly yields Equation (11). (ii) Let us introduce the simplified following notations: $\Xi_i = \|[\mathbf{Y}_{m_{i,\rm o}}^{N,i}] - [\eta_d^i]\|^2$ and $\zeta_i = \|[\eta_d^i]\|^2$. Therefore, Equation (8) can be rewritten as $d_{i,N}^2(m_{i,\rm o}) = E\{\Xi_i\}/\zeta_i$. If $\forall i \in \{1, \ldots, n_p\}$ we have $\Xi_i \geq \varepsilon\,\zeta_i$ *a.s*, then $\|[\mathbf{H}_{\mathbf{m}_{\rm o}}^{wg,N}] - [\eta_d]\|^2 = \sum_{i=1}^{n_p} \Xi_i \geq \varepsilon \sum_{i=1}^{n_p} \zeta_i = \varepsilon\,\|[\eta_d]\|^2$ *a.s*, that is to say $\|[\mathbf{H}_{\mathbf{m}_{\rm o}}^{wg,N}] - [\eta_d]\|^2/\|[\eta_d]\|^2 \geq \varepsilon$ *a.s.* (iii) Using result (ii) above, it can be deduced that $\text{Proba}\{\cap_{i=1}^{n_p}\{\Xi_i/\zeta_i \geq \varepsilon\}\} = \text{Proba}\{\|[\mathbf{H}_{\mathbf{m}_{\rm o}}^{wg,N}] - [\eta_d]\|^2/\|[\eta_d]\|^2 \geq \varepsilon\}$. (iv) Due to the partition, the random matrices $[\mathbf{Y}_{m_{1,\rm o}}^{N,1}], \ldots, [\mathbf{Y}_{m_{n_p,\rm o}}^{N,n_p}]$ are statistically independent, and thus $\Xi_1, \ldots, \Xi_{n_p}$ are statistically independent. Therefore, we can write, $\text{Proba}\{\cap_{i=1}^{n_p}\{\Xi_i/\zeta_i \geq \varepsilon\}\} = \Pi_{i=1}^{n_p}\text{Proba}\{\Xi_i/\zeta_i \geq \varepsilon\}$. (v) The results (iii) and (iv) above yield $\text{Proba}\{\|[\mathbf{H}_{\mathbf{m}_{\rm o}}^{wg,N}] - [\eta_d]\|^2/\|[\eta_d]\|^2 \geq \varepsilon\} = \Pi_{i=1}^{n_p}\text{Proba}\{\Xi_i/\zeta_i \geq \varepsilon\}$. The use of the Markov inequality allows us to write, $\text{Proba}\{\Xi_i/\zeta_i \geq \varepsilon\} \leq E\{\Xi_i\}/(\varepsilon\,\xi_i) = d_{i,N}^2(m_{i,\rm o})/\varepsilon$. Substituting this inequation into the right hand-side member of the last equality allows us to write $\text{Proba}\{\|[\mathbf{H}_{\mathbf{m}_{\rm o}}^{wg,N}] - [\eta_d]\|^2/\|[\eta_d]\|^2 \geq \varepsilon\} \leq \Pi_{i=1}^{n_p}\{d_{i,N}^2(m_{i,\rm o})/\varepsilon\} = (r/\varepsilon)^{n_p}$, which is Equation (12).

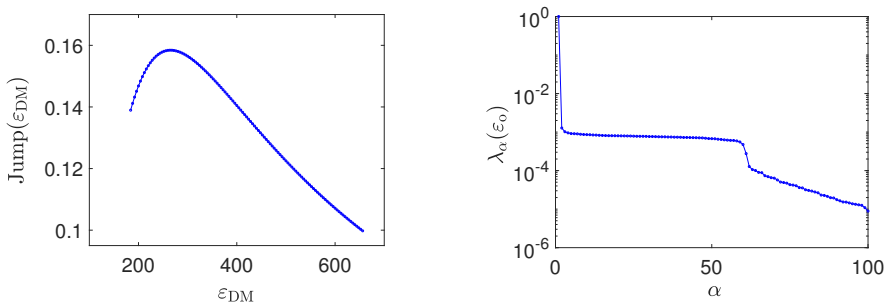
6. Application 1

The probabilistic model is chosen so that the partition in terms of statistically independent groups is known. This will serve to validate the proposed methodology. This application can easily be reproduced. We directly construct the normalized non-Gaussian $\mathbb{R}^\nu$-valued random variable $\mathbf{H} = (H_1, \ldots, H_\nu)$ with $\nu = 60$. Its probabilistic model is described in Appendix Appendix A. The random vector $\mathbf{X}$ from which $\mathbf{H}$ is deduced by a PCA is not constructed. It should be noted that this application is very difficult for the learning methods taking into account the high degree of the polynomials in the model, which induces a complexity of the geometry of the support of the probability measure of $\mathbf{H}$.

A reference data set with $N_{\text{ref}} = 1\,000\,000$ independent realizations and the training set with $N = 1\,200$ independent realizations $\{\boldsymbol{\eta}_d^j, j = 1, \ldots, N\}$ are generated using the probabilistic model of $\mathbf{H}$. The learned set is generated by the PLoM method (without or with groups) with $N_{\text{ar}} = 1\,200\,000$ realizations $\{\boldsymbol{\eta}_{\text{ar}}^\ell, \ell = 1, \ldots, N_{\text{ar}}\}$ ($N_{\text{ar}} = n_{\text{MC}} \times N$ with $n_{\text{MC}} = 1\,000$). It should be noted that the mean vector $\underline{\boldsymbol{\eta}}$ and the covariance matrix $[C_{\mathbf{H}}]$ of $\mathbf{H}$, which are estimated with the N realizations of the training set, are such that $\underline{\boldsymbol{\eta}}_d = \mathbf{0}_\nu$ and $[C_{\mathbf{H}}] = [I_\nu]$.

6.1. PLoM analysis with no group (No-Group PLoM)

Algorithm 1 is used for the calculation of the reduced-order diffusion-map basis $[g_{m_{\text{o}}}]$ of the $\mathbb{R}^\nu$-valued random variable $\mathbf{H}$. The optimal dimension is $m_{\text{o}} = \nu + 1 = 61$. Figure 2a displays the function $\varepsilon_{\text{DM}} \mapsto \text{Jump}(\varepsilon_{\text{DM}})$ and shows that the optimal value ε_{o} of the smoothing parameter ε_{DM} is $\varepsilon_{\text{o}} = 656$ for which $\text{Jump}(\varepsilon_{\text{o}}) = 0.1$. For this value ε_{o} of ε_{DM}, Figure 2b shows the graph of function $\alpha \mapsto \lambda_\alpha(\varepsilon_{\text{o}})$. It can be seen that the criterion defined by Equation (3) is satisfied. The PLoM algorithm with no group is then used for generating the learned set



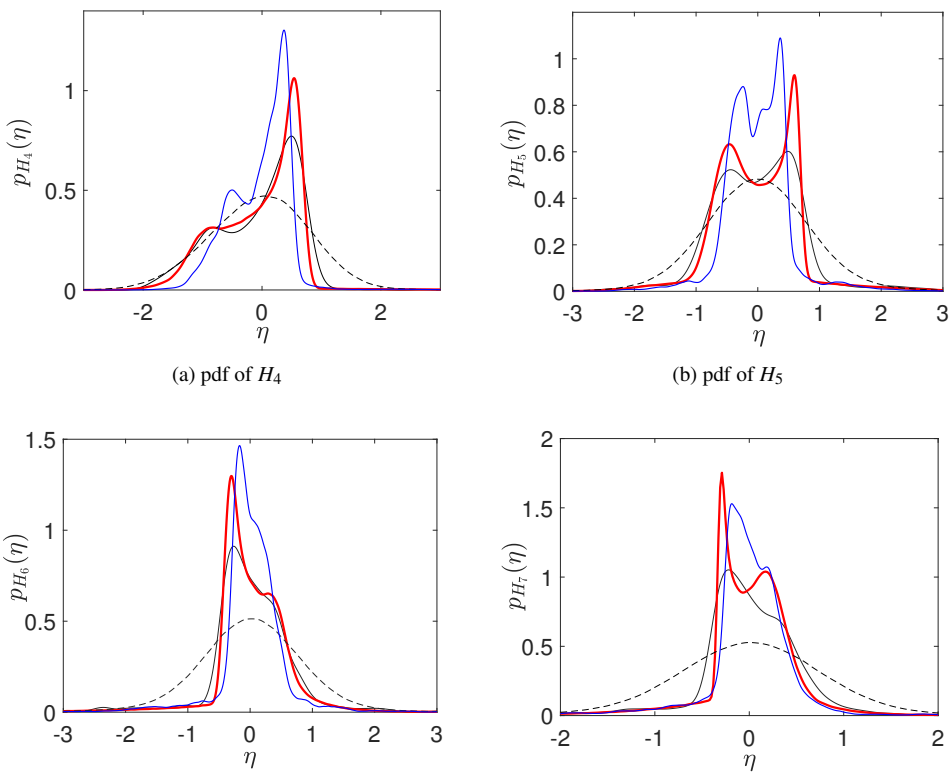
(a) Identifying the value ε_{o} of ε_{DM}

(b) Eigenvalues of the transition matrix

Figure 2: Construction of the diffusion map basis of the standard PLoM (No group).

$\{\boldsymbol{\eta}_{\text{ar}}^\ell, \ell = 1, \ldots, N_{\text{ar}}\}$. Figure 3 shows the pdf of each one of the random variables H_4, H_5, H_6, and H_7 estimated with the learned set. Each pdf is estimated (i) with the N realizations of the training set, (ii) with the N_{ref} realizations of the reference data set, (iii) with the N_{ar} additional realizations generated with the Hamiltonian MCMC algorithm corresponding to the PLoM with $m_{\text{o}} = N$ and $[g_{m_{\text{o}}}] = [I_N]$, and referenced as "*No-PLoM*", and finally, with the N_{ar} realizations of the learned set constructed with the PLoM for which the partition in groups is not taken into account and referenced as "*No-Group PLoM*" (in this case no constraints are applied). It can be seen that the

No-PLoM estimation yields a big scattering with an important increase of the dispersion (and thus a loss of the concentration of the probability measure) while No-Group PLoM preserves the concentration of the probability measure (as expected) and the pdfs' estimations are good enough. These estimations will be improved by using the PLoM with groups and referenced as "*With-Group PLoM*".



(a) pdf of H_4

(b) pdf of H_5

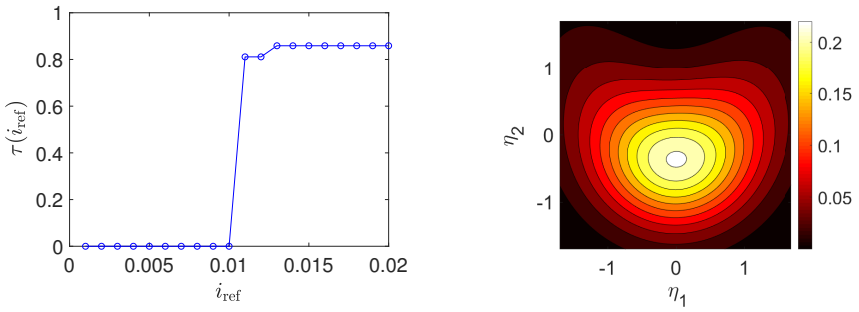
(c) pdf of H_6

(d) pdf of H_7

Figure 3: pdf estimated with (i) the training set (black thin), (ii) the reference data set (red thick), (iii) No-PLoM (dashed) , and (iv) No-Group PLoM (blue thin).

6.2. *Computing the partition*

The optimal partition is computed as explained in Section 5.1. Figure 4a displays the graph of $i_{\text{ref}} \mapsto \tau(i_{\text{ref}})$, which shows that $i_{\text{ref}}^{\text{o}} = 0.013$. Finally, the algorithm identifies the partition and finds $n_p = 3$ groups with $\nu_1 = 10$, $\nu_2 = 20$, and $\nu_3 = 30$ and with $\mathbf{Y}^1 = (H_1, \ldots, H_{10})$, $\mathbf{Y}^2 = (H_{11}, \ldots, H_{30})$, and $\mathbf{Y}^3 = (H_{31}, \ldots, H_{60})$, which correspond to the model introduced in Appendix Appendix A for generating the training set. This result constitutes an additional validation of the optimal partition algorithm that is used for non-Gaussian random vectors. For illustration, Figure 4b displays the graph of the joint pdf of random variables H_1 and H_2.



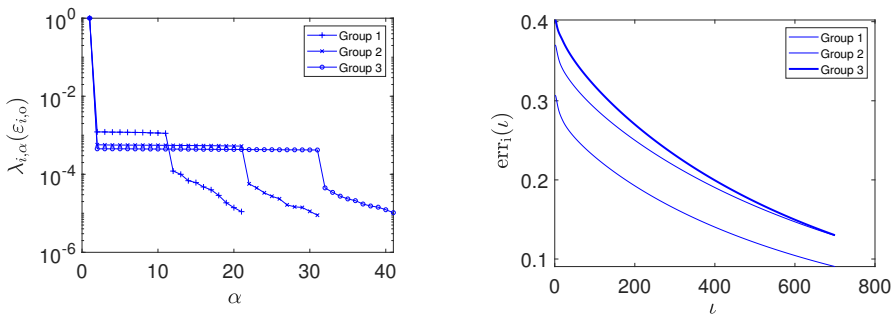
(a) Graph of function $i_{\mathrm{ref}} \mapsto \tau(i_{\mathrm{ref}})$

(b) Joint pdf of H_1 and H_2

Figure 4: Partition of $\mathbf{H}$ in n_p mutually independent random vectors $\mathbf{Y}^1, \ldots, \mathbf{Y}^{n_p}$.

6.3. PLoM analysis with groups (With-Group PLoM)

Algorithm 1 is used for each group $i = 1, 2, 3$. We then have $m_{i,\mathrm{o}} = \nu_i + 1$. The training set $\{\boldsymbol{\eta}_d^{i,j}, j = 1, \ldots, N\}$ of $\mathbf{Y}^i$ is used. A similar graph to the one shown in Figure 2a is constructed for identifying the optimal value $\varepsilon_{i,\mathrm{o}}$ of the smoothing parameter $\varepsilon_{i,\mathrm{DM}}$ yielding $\varepsilon_{1,\mathrm{o}} = 412$, $\varepsilon_{2,\mathrm{o}} = 896$, and $\varepsilon_{3,\mathrm{o}} = 1\,132$. Figure 5a shows the distribution of the eigenvalues of the transition matrix of each group i computed for $\varepsilon_{i,\mathrm{DM}} = \varepsilon_{i,\mathrm{o}}$. It can be seen that all the required criteria are satisfied. For each group $i = 1, 2, 3$, the PLoM method with groups is used for generating the

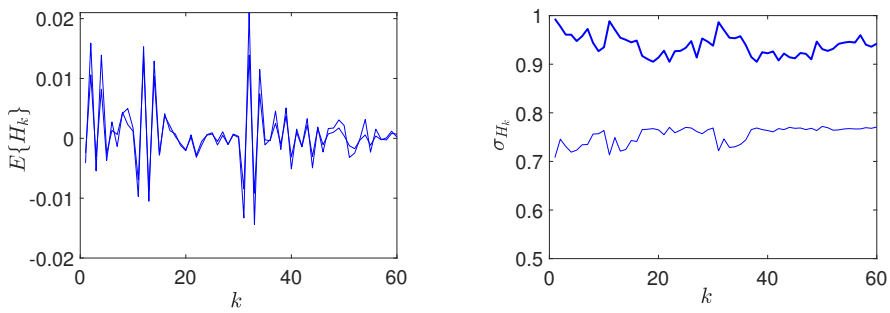


(a) Eigenvalues of the transition matrix of group i computed for $\varepsilon_{i,\mathrm{DM}} = \varepsilon_{i,\mathrm{o}}$

(b) Error function $\iota \mapsto \mathrm{err}_i(\iota)$ of group i for iteration number ι of the iteration algorithm

Figure 5: Diffusion map basis and error function of the iteration algorithm for each one of the 3 groups, $i = 1, 2, 3$.

learned set $\{\boldsymbol{\eta}_{\mathrm{ar}}^{i,j}, j = 1, \ldots, N_{\mathrm{ar}}\}$. The constraints $E\{(Y_k^i)^2\} = 1$ for $k \in \{1, \ldots, \nu_i\}$ are applied and the iterative algorithm introduced in Section 5.4 is used. Figure 5b displays the error function $\iota \mapsto \mathrm{err}_i(\iota)$ of group i, defined by Equation (6), which shows the convergence of the iterative algorithm. It should be noted that the convergence could have been pushed further, but the numerical experiments showed that the additional gain obtained is negligible. In addition, numerical experiments have been carried out to compare the efficiency of the type of constraints. We have verified that taking into account all the constraints (mean of $\mathbf{Y}^i$ equal to $\mathbf{0}$ and covariance matrix $[C_{\mathbf{Y}^i}] = [I_{\nu_i}]$) did not provide significant improvements on the preservation of the concentration of the probability measure compared to the sole application of the constraints $E\{(Y_k^i)^2\} = 1$ for



(a) Mean value of H_k

(b) Standard deviation of H_k

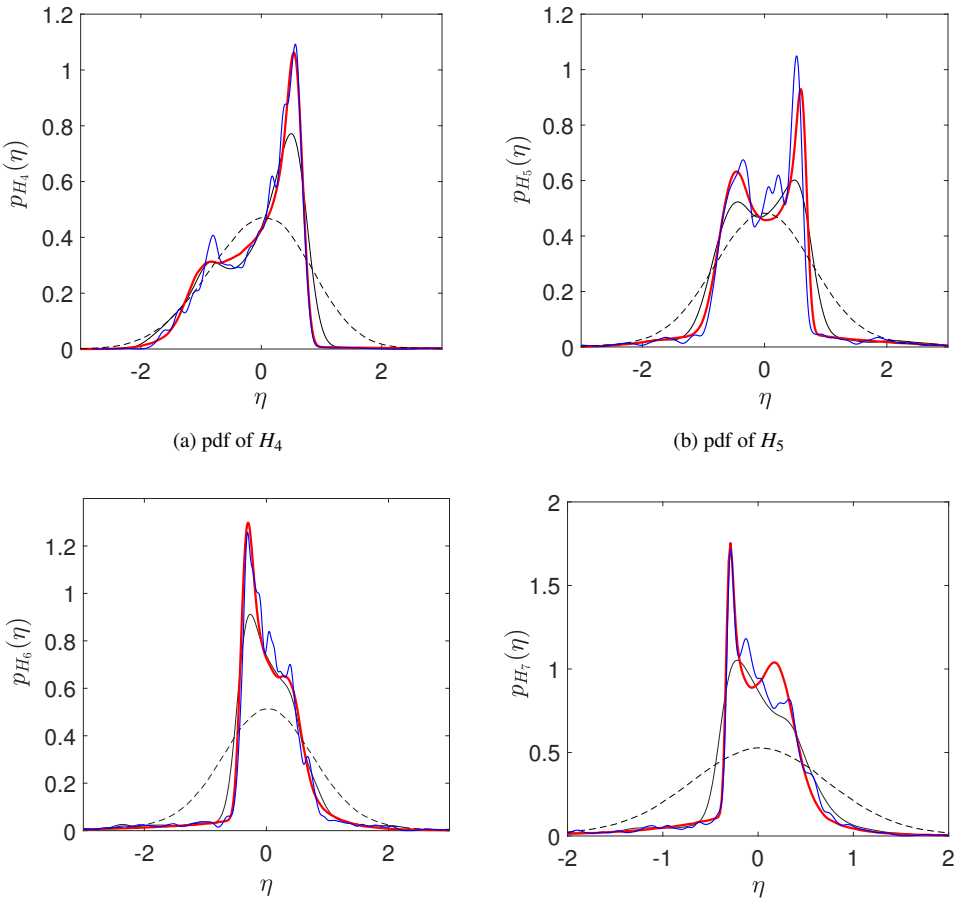
Figure 6: Mean value and standard deviation of the components H_k, $k = 1, \ldots, \nu$, of $\mathbf{H}$ estimated using the learned set generated by No-Group PLoM (thin line) and by With-Group PLoM (thick line).

$k \in \{1, \ldots, \nu_i\}$. Figure 6 shows the mean value and the standard deviation of the components H_k, $k = 1, \ldots, \nu$ of $\mathbf{H}$ estimated using the learned set generated by No-Group PLoM (see Section 6.1) and by With-Group PLoM. Figure 6a shows that the mean values are reasonably small with respect to 1 and therefore that it is not necessary to improve it by introducing the constraints for the mean. Figure 6b shows that the standard deviations are improved by using With-Group PLoM for which the constraints are taken into account. Figure 7 shows the pdf of H_4, H_5, H_6, and H_7 estimated with the learned set. Similarly to Section 6.1, each pdf is estimated (i) with the N realizations of the training set, (ii) with the N_{ref} realizations of the reference data set, (iii) with the N_{ar} additional realizations computed by No-PLoM, and (iv) with the N_{ar} realizations computed by With-Group PLoM. Comparing Figure 3 with Figure 7, it can be seen that the use of groups improves the pdfs' estimations as expected. It can also be noted that the estimates are excellent for this very difficult case in particular by comparing with the usual approach (see the dashed lines corresponding to No-PLoM).

6.4. Quantifying the concentration of the probability measure

For No PLoM, the computations are performed as explained in Section 6.1, for No-Group PLoM as in Section 4 and Section 6.1, and for With-Group PLoM as in Section 5 and Section 6.3.

(i) The results concerning the concentration of the probability measure are summarized in Table 1. For No PLoM, $d_N^2(N)$ is computed with Equation (4) for which $m = N = 1\,200$. For No-Group PLoM, $d_N^2(m_{\text{o}})$ is also computed with Equation (4) but with $m = m_{\text{o}} = 61$. For With-Group PLoM, $d_{wg,N}^2(\mathbf{m}_{\text{o}})$ is computed with Equation (7) for which $\mathbf{m}_{\text{o}} = (m_{1,\text{o}}, m_{2,\text{o}}, m_{3,\text{o}})$ with $m_{1,\text{o}} = 11$, $m_{2,\text{o}} = 21$, and $m_{3,\text{o}} = 31$, and where $d_{i,N}^2(m_{i,\text{o}})$ are computed using Equation (8), which yields $d_{1,N}^2(m_{1,\text{o}}) = 0.012$, $d_{2,N}^2(m_{2,\text{o}}) = 0.015$, and $d_{3,N}^2(m_{3,\text{o}}) = 0.019$. The results obtained are those that were hoped for. Without using the PLoM method, we find numerically $d_N^2(N) \simeq 2$ that is the theoretical value (see Section 4.5). We also see that $d_N^2(m_{\text{o}}) = 0.094 \ll 2$, which shows that the usual PLoM method (without group) effectively preserves the concentration of the probability measure unlike the usual MCMC method that does not allow it. For the PLoM with groups, an improvement is observed relative to the PLoM without group as indicated by the evaluation $d_{wg,N}^2(\mathbf{m}_{\text{o}}) = 0.016 \ll d_N^2(m_{\text{o}}) = 0.094$. The quantification of the probability of the random relative distance defined by Equation (12) confirms this improvement. Note that the probability $(r/\varepsilon)^{n_p}$ corresponds to an upper value, the probability being certainly smaller.



(a) pdf of H_4

(b) pdf of H_5

(c) pdf of H_6

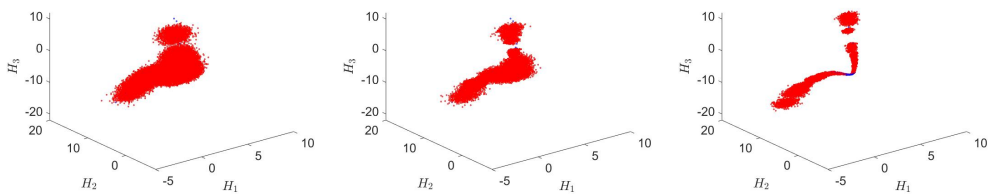
(d) pdf of H_7

Figure 7: pdf estimated with (i) the training set (black thin), (ii) the reference data set (red thick), (iii) No-PLoM (dashed) , and (iv) With-Group PLoM (blue thin).

Table 1: Concentration of the probability measure for Application 1

No PLoM	**PLoM No Group**	**PLoM With Group**		
	$m_{\mathrm{o}} = 61$		Proba by Eq. Equation (12)	
$d_N^2(N)$	$d_N^2(m_{\mathrm{o}})$	$d_{wg,N}^2(\mathbf{m}_{\mathrm{o}})$	ε	$\leq (r/\varepsilon)^{n_p}$
2.00	0.094	0.016	0.05	≤ 0.028
			0.10	≤ 0.0034

(ii) Concerning the visualization of the concentration of the probability measure, Figure 8 shows the clouds of points $\{\eta_{\text{ar}}^{\ell}, \ell = 1, \ldots, N_{\text{ar}}\}$ for the components (H_1, H_2, H_3), generated (a) without using PLoM (No-PLoM), (b) using the PLoM method without group (No-Group PLoM), and (c) using the PLoM method with groups (With-Group PLoM). The three figures confirm the analysis presented in point (i) above.



(a) No PLoM (b) No-Group PLoM (c) With-Group PLoM

Figure 8: Clouds of the $N_{\text{ar}} = 1\,200\,000$ realizations of (H_1, H_2, H_3) computed with No-PLoM, No-Group PLoM, and With-Group PLoM.

7. Application 2

The second application is devoted to a supervised learning problem $\mathbf{Q} = \mathbf{f}(\mathbf{W}, \mathbf{U})$ (see Section 2) in high dimension, for which the uncontrolled random parameter is the $\mathbb{R}^{n_u}$-valued random variable $\mathbf{U}$ with $n_u = 420\,000$, the random control random parameter is the $\mathbb{R}^{n_w}$-valued random variable $\mathbf{W}$ with $n_w = 2$, and the QoI is the $\mathbb{R}^{n_q}$-valued random variable $\mathbf{Q}$ with $n_q = 10\,098$.

7.1. Generation of the training set and reference data set

This application relates to a linear elastic system modeled by an elliptic stochastic boundary problem (BVP) in a 3D bounded domain Ω, described in the SI Units. The generic point of Ω is $\boldsymbol{\zeta} = (\zeta_1, \zeta_2, \zeta_3)$ in an orthonormal Cartesian coordinate system $(O, \zeta_1, \zeta_2, \zeta_3)$ with $O = (0, 0, 0)$. The outward unit normal to $\partial\Omega = \Gamma_0 \cup \Gamma$ is denoted by $\mathbf{n}(\boldsymbol{\zeta})$. There is a zero Dirichlet condition on Γ_0 and a Neumann condition on Γ. Domain Ω is occupied by a random linear elastic medium (heterogeneous material). The uncontrolled parameter $\mathbf{U}$ of the system is related to the finite element discretization of the fourth-order tensor-valued non-Gaussian elasticity random field $\{\mathbb{K} = \{\mathbb{K}_{ijkh}(\boldsymbol{\zeta})\}_{1\le i,j,k,h\le 3}, \boldsymbol{\zeta} \in \Omega\}$ (random coefficients of the partial differential operator) for which the mean value is isotropic and the statistical fluctuations are anisotropic. The control parameter $\mathbf{w} = (w_1, w_2)$ of the system consists of $w_1 = \log(L_{\text{corr}})$ in which L_{corr} is a spatial correlation length and of $w_2 = \log(\delta_G)$ in which δ_G is a dispersion parameter, which allows the statistical fluctuations of $\mathbb{K}$ to be controlled. The observation of the system is the $\mathbb{R}^3$-valued random displacement field $\mathbf{V} = (V_1, V_2, V_3)$ on Ω, which is the strong stochastic solution of the weak formulation of the stochastic BVP,

$$
\begin{aligned}
-\text{div}\, \boldsymbol{\Sigma} &= \mathbf{0} \quad \text{in} \quad \Omega\,, \\
\mathbf{V} &= \mathbf{0} \quad \text{on} \quad \Gamma_0\,, \\
\boldsymbol{\Sigma}\,\mathbf{n} &= \boldsymbol{\mathcal{G}}^{\Gamma} \quad \text{on} \quad \Gamma\,.
\end{aligned}
$$

The stress tensor $\mathbf{\Sigma} = \{\Sigma_{ij}\}_{1\leq i,j\leq 3}$ is related to the strain tensor $\mathbf{E}(\mathbf{V}) = \{E_{kh}(\mathbf{V})\}_{1\leq k,h\leq 3}$ by the constitutive equation, $\Sigma_{ij}(\boldsymbol{\zeta}) = \mathbb{K}_{ijkh}(\boldsymbol{\zeta})\, E_{kh}(\mathbf{V}(\boldsymbol{\zeta}))$ in which the strain tensor is such that $E_{kh}(\mathbf{V}) = (\partial V_k/\partial\zeta_h + \partial V_h/\partial\zeta_k)/2$. The geometry, the surface force field $\boldsymbol{\mathcal{G}}^{\Gamma}$, the probabilistic model of the elasticity random field $\mathbb{K}$ that depends on parameter $\mathbf{w}$, and the finite element discretization of the weak formulation of the stochastic BVP are detailed in Appendix Appendix B.

The control parameter $\mathbf{w}$ is modelled by a $\mathbb{R}^2$-valued random variable $\mathbf{W} = (W_1, W_2)$. The random vectors $\mathbf{U}$, $\mathbf{W}$, and $\mathbf{Q}$, for which the dimensions are $n_u = 420\,000$, $n_w = 2$, and $n_q = 10\,098$, are defined in Appendix Appendix B. The random vectors $\mathbf{W}$ and $\mathbf{U}$ are statistically independent. The dimension $n = n_q + n_w$ of random vector $\mathbf{X} = (\mathbf{Q}, \mathbf{W})$ is thus $n = 10\,100$.

The training set is generated as explained in Section 2 for which $N = 100$ independent realizations, $\mathbf{u}_d^j$ and $\mathbf{w}_d^j$, of $\mathbf{U}$ and $\mathbf{W}$ are generated using the probabilistic model detailed in Appendix Appendix B. For each $j \in \{1, \ldots, N\}$, the realization $\mathbf{q}_d^j$ of $\mathbf{Q}$ is computed by solving the BVP using the computational model (finite element discretization of the BVP), which is such that $\mathbf{q}_d^j = \mathbf{f}(\mathbf{w}_d^j, \mathbf{u}_d^j)$ (note that $\mathbf{f}$ is not explicitly known and results from the solution of the BVP). The training set related to random vector $\mathbf{X} = (\mathbf{Q}, \mathbf{W})$ is then made up of the N independent realizations $\{\mathbf{x}_d^j, j = 1, \ldots, N\}$ in which $\mathbf{x}_d^j = (\mathbf{q}_d^j, \mathbf{w}_d^j) \in \mathbb{R}^n$.

The reference data set $\{\mathbf{x}_{\text{ref}}^{\ell}, \ell = 1, \ldots, N_{\text{ref}}\}$ for $\mathbf{X}$ is generated as the training set but with $N_{\text{ref}} = 20\,000$ independent realizations. Computations have been made for $N_{\text{ref}} = 15\,000$, $18\,000$, and $20\,000$, which have shown that the pdf of each observed component of $\mathbf{Q}$ were converged for $N_{\text{ref}} = 20\,000$ (note that the construction of the reference has been very CPU time consuming).

The learned sets generated without using the PLoM method (No PLoM), or using the PLoM method with no group (No-Group PLoM), or with groups (With-Group PLoM) will be all performed with $N_{\text{ar}} = 200\,000$ realizations $\{\boldsymbol{\eta}_{\text{ar}}^{\ell}, \ell = 1, \ldots, N_{\text{ar}}\}$ ($N_{\text{ar}} = n_{\text{MC}} \times N$ with $n_{\text{MC}} = 2\,000$).

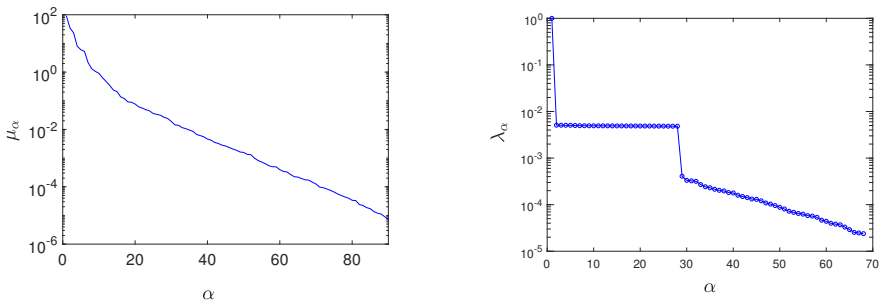
7.2. PLoM analysis without and with partition

In this section, we give the main results without too many details (paper length limitation), knowing that we have already presented a detailed analysis for Application 1.

(i) PCA of random vector $\boldsymbol{X}$. Since $n = 10\,000 \gg N = 100$, the eigenvalue problem of $[C_{\mathbf{X}}]$ is solved using a thin SVD of matrix $[x_d] = [\mathbf{x}_d^1 \ldots \mathbf{x}_d^N] \in \mathbb{M}_{n,N}$, which thus does not require the assembling of $[C_{\mathbf{X}}]$ (as explained in Section 3). Figure 9a shows the distribution of the eigenvalues μ_α. For constructing the PCA representation, $\mathbf{X}^\nu = \underline{\mathbf{x}}_d + [\Phi]\,[\mu]^{1/2}\,\mathbf{H}$, of $\mathbf{X}$, we have chosen $\varepsilon_{\text{PCA}} = 0.001$ that yields $\nu = 27$. Following Section 3, the matrix $[\eta_d] \in \mathbb{M}_{\nu,N}$ is constructed with the N realizations $\boldsymbol{\eta}_d^1, \ldots, \boldsymbol{\eta}_d^N$ of the $\mathbb{R}^\nu$-valued random variable $\mathbf{H}$.

(ii) Reduced-order diffusion-map basis for No-Group PLoM. Algorithm 1 is used for the calculation of the reduced-order diffusion-map basis $[g_{m_{\text{o}}}]$ of the $\mathbb{R}^\nu$-valued random variable $\mathbf{H}$. For the optimal values $m_{\text{o}} = \nu + 1 = 28$ and $\varepsilon_{\text{o}} = 103$, Figure 9b shows the eigenvalues λ_α of the transition matrix.

(iii) Construction of the optimal partition of $\boldsymbol{H}$. The optimal partition is computed as explained in Section 5.1. Figure 10a displays the graph of $i_{\text{ref}} \mapsto \tau(i_{\text{ref}})$, which shows that $i_{\text{ref}}^{\text{o}} = 0.112$. The algorithm identifies the partition and finds $n_p = 9$ groups such that $\nu_1 = 5$ with $\mathbf{Y}^1 = (H_1, H_2, H_4, H_{16}, H_{19})$, $\nu_2 = 1$ with $Y^2 = H_3$, $\nu_3 = 15$ with $\mathbf{Y}^3 = (H_5 \text{ to } H_{11}, H_{14}, H_{15}, H_{17}, H_{18}, H_{20}, H_{24} \text{ to } H_{26})$, and $\nu_4 = \ldots = \nu_9 = 1$ with $Y^4 = H_{12}$, $Y^5 = H_{13}$, $Y^6 = H_{21}$, $Y^7 = H_{22}$, $Y^8 = H_{23}$, and $Y^9 = H_{27}$. For each one of the two groups $i = 1$ and 3 (having a length greater than 1), the optimal values are $m_{1,\text{o}} = 6$, $m_{3,\text{o}} = 16$, $\varepsilon_{1,\text{o}} = 37.7$, and $\varepsilon_{3,\text{o}} = 103$. For these optimal values of

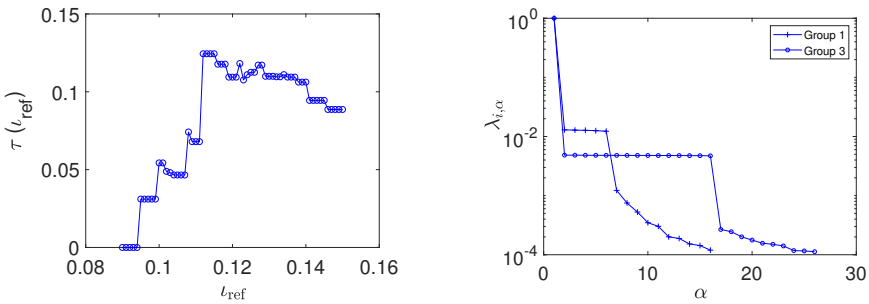


(a) Eigenvalues of the covariance matrix of $\mathbf{X}$

(b) Eigenvalues of the transition matrix

Figure 9: PCA eigenvalues and diffusion map eigenvalues of the PLoM without group (No-Group PLoM).

$\varepsilon_{i,\mathrm{o}}$, Figure 10b shows the distribution of eigenvalues $\lambda_{i,\alpha}$ of the transition matrix. For the groups $i \notin \{1, 3\}$, we have $m_{i,\mathrm{o}} = N$ (see Algorithm 1).



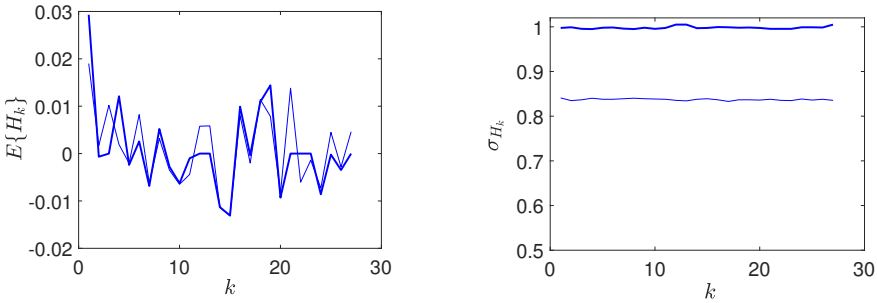
(a) Graph of function $i_{\mathrm{ref}} \mapsto \tau(i_{\mathrm{ref}})$

(b) Eigenvalues of the transition matrix

Figure 10: Partition of $\mathbf{H}$ in $n_p = 9$ mutually independent random vectors $\mathbf{Y}^1, \ldots, \mathbf{Y}^{n_p}$ and diffusion map eigenvalues of the PLoM for groups 1 and 3.

(iv) Influence of the constraints of all the components of $\boldsymbol{Y}^i$. No-Group PLoM is performed without any constraints applied to random vector $\mathbf{H}$. With-Group PLoM is performed, group by group, by applying, for $i = 1, \ldots, n_p = 9$, the constraints $E\{(Y_k^i)^2\} = 1$ for $k \in \{1, \ldots, \nu_i\}$. For all the components $k = 1, \ldots, \nu = 27$ of $\mathbf{H}$, Figure 11 shows the mean value $E\{H_k\}$ and the standard deviation σ_{H_k} that are estimated by No-Group PLoM and by With-Group PLoM. We can see that the mean values remain much lower than 1 although no constraint is applied to the mean, as well for No-Group PLoM as for With-Group PLoM. We can also see that the standard deviation of the components are already close to 1 for No-Group PLoM although no constraint is applied to the second-order moments. As expected, for With-Group PLoM for which the constraints are applied to the second-order moments, the standard deviations are almost equal to 1.

(v) pdf of observations estimated by the PLoM. The pdf of components Q_{17} and Q_{7740} of $\mathbf{Q}$ are presented in Figure 12. Component 17 corresponds to the ζ_2-axis displacement $V_2(\boldsymbol{\zeta})$ at point

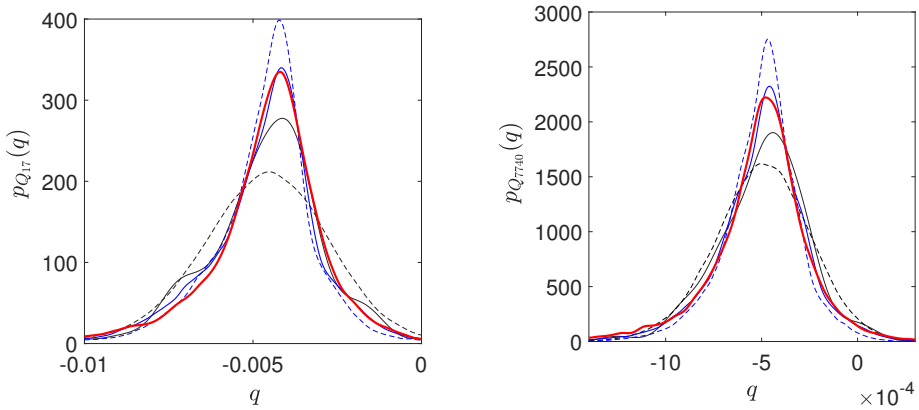


(a) Mean value of H_k

(b) Standard deviation of H_k

Figure 11: Mean value and standard deviation of the components H_k, $k = 1, \ldots, \nu$, of $\mathbf{H}$ estimated using the learned set generated with No-Group PLoM (thin line) and With-Group PLoM (thick line).

$\boldsymbol{\zeta} = (0, 0, 0.1)$ while component 7740 corresponds to the ζ_3-axis displacement $V_3(\boldsymbol{\zeta})$ at point $\boldsymbol{\zeta} = (0.78, 0, 0.1)$. Figure 12 shows the pdf estimated (i) with the $N_d = 100$ points of the training



(a) pdf of Q_{17}

(b) pdf of Q_{7740}

Figure 12: pdf estimated with (i) the training set (black thin), (ii) the reference data set (red thick), (iii) No-PLoM (dashed), (iv) No-Group PLoM (blue thin dashed), and (v) With-Group PLoM (blue thick).

set, (ii) with the $N_{\text{ref}} = 20\,000$ points of the reference data set, (iii) with $N_{\text{ar}} = 200\,000$ additional realizations generated with an usual MCMC generator (without using the PLoM method), (iv) with the $N_{\text{ar}} = 200\,000$ additional realizations of the learned set generated by No-Group PLoM, and finally, (v) with the $N_{\text{ar}} = 200\,000$ additional realizations of the learned set generated by With-Group PLoM for which a partition in $n_p = 9$ groups has been identified. It can be seen that the usual MCMC method (no PLoM) is not good at all, that No-Group PLoM already gives a good estimation in comparison with the reference, and finally, that With-Group PLoM gives an excellent estimation when compared to the reference.

(vi) Quantifying the concentration of the probability measure. The analysis is carried out as in Section 6.4. The results concerning the concentration of the probability measure is summarized in Table 2 and in Figure 13. For No PLoM, $d_N^2(N)$ is computed with Equation (4) for which

Table 2: Concentration of the probability measure for Application 2

No PLoM	**PLoM No Group**	**PLoM With Group**		
	$m_o = 28$		Proba by Eq. Equation (12)	
$d_N^2(N)$	$d_N^2(m_o)$	$d_{wg,N}^2(\mathbf{m}_o)$	ε	$\leq (r/\varepsilon)^{n_p}$
2.00	0.16	0.044	0.05	$\leq 4.3 \times 10^{-5}$
			0.1	$\leq 8.3 \times 10^{-8}$

$m = N = 100$. For No-Group PLoM, $d_N^2(m_o)$ is also computed with Equation (4) but with $m = m_o = 28$. For With-Group PLoM, $d_{wg,N}^2(\mathbf{m}_o)$ is computed with Equation (7) for which $\mathbf{m}_o = (m_{1,o}, \ldots, m_{9,o})$. The graph $i \mapsto d_{i,N}^2(m_{i,o})$ is computed using Equation (8) and is plotted in Figure 13. Without using the PLoM method, we find numerically $d_N^2(N) = 2$ that is the theoretical value (see Section 4.5). We also see that $d_N^2(m_o) = 0.16 \ll 2$, which shows that the usual PLoM method (without group) effectively preserves the concentration of the probability measure unlike the usual MCMC methods that do not allow it. For the PLoM with groups, it can be seen an improvement with respect to the PLoM without group because $d_{wg,N}^2(\mathbf{m}_o) = 0.044 \ll d_N^2(m_o) = 0.16$. The quantification of the probability of the random relative distance defined by Equation (12) confirms this improvement. Note that the probability $(r/\varepsilon)^{n_p}$ corresponds to an upper value, the probability being certainly smaller.

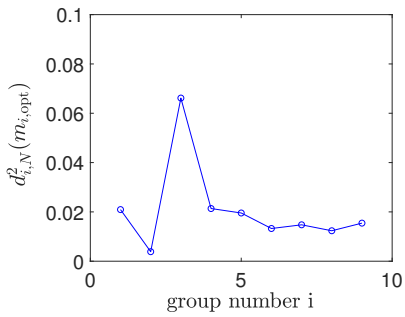


Figure 13: Concentration of the probability measure for each group $i \in \{1, \ldots, 9\}$: graph of $i \mapsto d_{i,N}^2(m_{i,o})$ computed using Equation (8).

(vii) Computational cost according to PLoM usage configurations and to the level of preservation of the concentration of the probability measure. Table 3 presents the numerical cost in terms of Elapsed Time (in seconds) of the analyses performed with a one node computer having 110 cores. In order to compare the numerical costs with the accuracy given for each case, we have given the value of $d_N^2(m_o)$ that characterizes the preservation of the concentration of the probability measure. As expected, the "With-Group PLoM with constraints" yields the best accuracy for an

elapsed time that is very small with respect to the one generated for constructing the converged reference (reference data set) and which is the one that is required if PLoM is not used.

Table 3: Numerical cost of the analyses for Application 2

		PLoM No Group		**PLoM With Group**	
	Reference data set	No constraint	With constraints	No constraint	With constraints
$d_N^2(m_o)$		0.16	0.14	0.30	0.044
Elaspsed Time(s)	18 000	15	215	20	649

8. Discussion and conclusion

The implementation of a partition in the PLoM method has provided an opportunity to revisit, improve the efficiency, and simplify the algorithm to identify the optimal values of the hyper-parameters of the reduced-order diffusion-map basis. This was made necessary for the PLoM method with partition, because the number of groups identified can be large and for each group of dimension greater than 1, the reduced-order diffusion-map basis must be constructed. This new efficient algorithm is common to PLoM with or without partition.

Still within the framework of the PLoM carried out with partition, we have made the following observations. If a group of the partition has a relatively small dimension (a few units, or even one or two dozen) and if the support of its probability measure has a complex geometry, one could obtain a significant loss of normalization compared to 1 (for instance 0.6 or 0.7 instead of 0.9 or 1). For instance, such a situation can be encountered by the presence of numerous non-Gaussian stochastic germs that generate strong statistical fluctuations (for example, up to ten times the standard deviation for some components). For these cases, we have proposed to introduce constraints on the second-order moments of the components of such a group, by reusing the Kullback-Leibler minimum cross-entropy principle that we have previously used for taking into account physics constraints in the PLoM method. For instance, Application 1 is very difficult due to the high degree of the polynomials in the model; although the realizations of the training set are centered and have a covariance matrix equal to the identity matrix, the fluctuations vary between -10 and $+10$ for some components (which must be compared to a magnitude of 1). It should be noted that we have also developed, tested, and implemented the general case of introducing constraints on the mean vector (zero mean) and on the covariance matrix (identity matrix). One then increases considerably the number of Lagrange multipliers to be calculated by the iterative method, which induces a significant numerical additional cost. We have not seen any significant improvement compared to the only constraint related to the diagonal of the covariance matrix (second-order moments equal to 1 knowing that the centering is reasonably well obtained without constraint on the mean vector). Under these conditions we have only presented the simplest case of constraints and we have demonstrated it on two applications.

In the recently published mathematical foundations of PLoM [13], to establish the main theorem, we introduced a distance between the random matrix defined by PLoM and the deterministic matrix that represent all the given points of the training set. In the present paper, and in order

to facilitate the quantification of the preservation of the concentration of the probability measure between the usual MCMC method (No PLoM), the PLoM method without partition (No-Group PLoM), and the PLoM method with partition (With-Group PLoM), we apply this distance to each group of the partition. We have assessed it numerically for the two applications. The results obtained confirm the theoretical results: there is a significant loss of concentration of the probability measure for the usual MCMC method (No PLoM) while the PLoM method without partition (No-Group PLoM) preserves well the concentration of the probability measure. In addition, this distance shows that the PLoM method with partition further improves the preservation of the concentration of the probability measure compared to the PLoM method without partition, which was hoped for. Finally, to complete the quantification of the concentration of the probability measure by the distance, we have also proven a mathematical result of this quantification in terms of probability. This result shows that if the number of groups of the partition increases, then the gain of With-Group PLoM can be significantly improved compared to No-Group PLoM. We have numerically quantified these probabilities for the two applications.

This work contributes to the improvement of the PLoM method. The results presented illustrate the gains made for the two applications, which, although quite distinct, present significant challenges for other statistical learning methods.

NOTATIONS

The following notations are used:
A lower case letter such as x, η, or u, is a real deterministic variable.
A boldface lower case letter such as $\mathbf{x}$, $\boldsymbol{\eta}$, or $\mathbf{u}$ is a real deterministic vector.
An upper case letter such as X, H, or U, is a real random variable (except for E).
A boldface upper case letter, $\mathbf{X}$, $\mathbf{H}$, or $\mathbf{U}$, is a real random vector.
A letter between brackets such as $[x]$, $[\eta]$, $[u]$ or $[C]$, is a real deterministic matrix.
A boldface upper case letter between brackets such as $[\mathbf{X}]$, $[\mathbf{H}]$, or $[\mathbf{U}]$, is a real random matrix.
n, n_q, n_u, n_w : dimensions of random vectors $\mathbf{X}, \mathbf{Q}, \mathbf{U}, \mathbf{W}$.
n_{MC} : number of additional realizations for random matrix $[\mathbf{H}^N_{m_{\mathrm{o}}}]$.
n_p : number of groups in the partition of $\mathbf{H}$.
m, m_i : dimension of the reduced-order diffusion-map bases $[g_m]$, $[g^i_m]$.
$m_{\mathrm{o}}, m_{i,\mathrm{o}}$: optimal value of m, m_i.
E : mathematical expectation.
N, N_{ar} : number of points in the training, learned sets.
ν, ν_i : dimensions of random vectors $\mathbf{H}, \mathbf{Y}^i$.
$\mathbb{R}, \mathbb{R}^n$: real line, Euclidean vector space of dimension n.
$\mathbb{M}_n, \mathbb{M}_{n,N}$: sets of all the $(n \times n), (n \times N)$ real matrices.
$\mathbb{M}^+_n$: set of all the positive-definite symmetric $(n \times n)$ real matrices.
$\|\mathbf{x}\|$: Euclidean norm when $\mathbf{x}$ is the vector or Frobenius norm when $[x]$ is the matrix.

ACKNOWLEDGEMENTS

Support for this work was partially provided through the Scientific Discovery through Advanced Computing (SciDAC) program funded by the U.S. Department of Energy, Office of Science, Advanced Scientific Computing Research

ORCID

Christian Soize, https://orcid.org/0000-0002-1083-6771
Roger Ghanem, https://orcid.org/0000-0002-1890-920X

References

[1] 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.

[2] A. Talwalkar, S. Kumar, H. Rowley, Large-scale manifold learning, in: 2008 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2008, pp. 1–8. doi:10.1109/CVPR.2008.4587670.

[3] D. Gorissen, I. Couckuyt, P. Demeester, T. Dhaene, K. Crombecq, A surrogate modeling and adaptive sampling toolbox for computer based design, Journal of Machine Learning Research 11 (68) (2010) 2051–2055. URL `http://jmlr.org/papers/v11/gorissen10a.html`

[4] A. C. Öztireli, M. Alexa, M. Gross, Spectral sampling of manifolds, ACM Transactions on Graphics (TOG) 29 (6) (2010) 1–8. doi:10.1145/1882261.1866190.

[5] Y. Marzouk, T. Moselhy, M. Parno, A. Spantini, Sampling via measure transport: An introduction, Handbook of uncertainty quantification (2016) 1–41doi:10.1007/978-3-319-11259-6_23-1.

[6] M. D. Parno, Y. M. Marzouk, Transport map accelerated markov chain Monte Carlo, SIAM/ASA Journal on Uncertainty Quantification 6 (2) (2018) 645–682. doi:10.1137/17M1134640.

[7] G. Perrin, C. Soize, N. Ouhbi, Data-driven kernel representations for sampling with an unknown block dependence structure under correlation constraints, Computational Statistics & Data Analysis 119 (2018) 139–154. doi:10.1016/j.csda.2017.10.005.

[8] P. Tsilifis, R. Ghanem, Bayesian adaptation of chaos representations using variational inference and sampling on geodesics, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences 474 (2217) (2018) 20180285. doi:10.1098/rspa.2018.0285.

[9] Y. Kevrekidis, Manifold learning for parameter reduction, Bulletin of the American Physical Society 65. doi:10.1016/j.jcp.2019.04.015.

[10] 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 Engineeringdoi:10.1002/nme.6430.

[11] 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.

[12] M. Arnst, C. Soize, K. Bulthies, Computation of sobol indices in global sensitivity analysis from small data sets by probabilistic learning on manifolds, International Journal for Uncertainty Quantification 11 (2) (2021) 1–23. doi:10.1615/Int.J.UncertaintyQuantification.2020032674.

[13] C. Soize, R. Ghanem, Probabilistic learning on manifolds, Foundations of Data Science 2 (3) (2020) 279–307. doi:10.3934/fods.2020013.

[14] 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.

[15] 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.

[16] C. Soize, R. Ghanem, Probabilistic learning on manifolds constrained by nonlinear partial differential equations for small datasets, Computer Methods in Applied Mechanics and Engineering 380 (2021) 113777. doi:10.1016/j.cma.2021.113777.

[17] J. L. Fleiss, B. Levin, M. C. Paik, Statistical Methods for Rates and Proportions, john wiley & sons, 2013.

[18] P. E. Greenwood, M. S. Nikulin, A guide to Chi-Squared Testing, Vol. 280, John Wiley & Sons, 1996.

[19] K. Pearson, X. on the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling, The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, Series 5 50 (302) (1900) 157–175. doi:10.1080/14786440009463897.

[20] R. Boscolo, H. Pan, V. P. Roychowdhury, Independent component analysis based on nonparametric density estimation, IEEE Transactions on Neural Networks 15 (1) (2004) 55–65. doi:10.1109/TNN.2003.820667.

[21] P. Comon, Independent component analysis, a new concept?, Signal processing 36 (3) (1994) 287–314. doi:10.1016/0165-1684(94)90029-9.

[22] P. Comon, C. Jutten, J. Herault, Blind separation of sources, part ii: Problems statement, Signal processing 24 (1) (1991) 11–20. doi:10.1016/0165-1684(91)90080-3.

[23] J. Herault, C. Jutten, Space or time adaptive signal processing by neural network models, in: AIP conference proceedings, Vol. 151(1), American Institute of Physics, 1986, pp. 206–211.
[24] A. Hyvarinen, Fast and robust fixed-point algorithms for independent component analysis, IEEE transactions on Neural Networks 10 (3) (1999) 626–634. doi:10.1109/72.761722.
[25] A. Hyvärinen, E. Oja, Independent component analysis: algorithms and applications, Neural networks 13 (4-5) (2000) 411–430. doi:10.1016/S0893-6080(00)00026-5.
[26] C. Jutten, J. Herault, Blind separation of sources, part i: An adaptive algorithm based on neuromimetic architecture, Signal processing 24 (1) (1991) 1–10. doi:10.1016/0165-1684(91)90079-X.
[27] T.-W. Lee, M. Girolami, A. J. Bell, T. J. Sejnowski, A unifying information-theoretic framework for independent component analysis, Computers & Mathematics with Applications 39 (11) (2000) 1–21. doi:10.1016/S0898-1221(00)00101-2.
[28] C. Soize, Optimal partition in terms of independent random vectors of any non-gaussian vector defined by a set of realizations, SIAM/ASA Journal on Uncertainty Quantification 5 (1) (2017) 176–211. doi:10.1137/16M1062223.
[29] G. H. Golub, C. F. Van Loan, Matrix Computations, Second Edition, Johns Hopkins University Press, Baltimore and London, 1993.
[30] T. Duong, A. Cowling, I. Koch, M. Wand, Feature significance for multivariate kernel density estimation, Computational Statistics & Data Analysis 52 (9) (2008) 4225–4242. doi:10.1016/j.csda.2008.02.035.
[31] T. Duong, M. L. Hazelton, Cross-validation bandwidth matrices for multivariate kernel density estimation, Scandinavian Journal of Statistics 32 (3) (2005) 485–506. doi:10.1111/j.1467-9469.2005.00445.x.
[32] M. Filippone, G. Sanguinetti, Approximate inference of the bandwidth in multivariate kernel density estimation, Computational Statistics & Data Analysis 55 (12) (2011) 3104–3122. doi:10.1016/j.csda.2011.05.023.
[33] N. Zougab, S. Adjabi, C. C. Kokonendji, Bayesian estimation of adaptive bandwidth matrices in multivariate kernel density estimation, Computational Statistics & Data Analysis 75 (2014) 28–38. doi:10.1016/j.csda.2014.02.002.
[34] 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.
[35] R. Coifman, S. Lafon, Diffusion maps, Applied and Computational Harmonic Analysis 21 (1) (2006) 5–30. doi:10.1016/j.acha.2006.04.006.
[36] S. Lafon, A. B. Lee, Diffusion maps and coarse-graining: A unified framework for dimensionality reduction, graph partitioning, and data set parameterization, IEEE transactions on pattern analysis and machine intelligence 28 (9) (2006) 1393–1403. doi:10.1109/TPAMI.2006.184.
[37] R. Neal, MCMC using hamiltonian dynamics, in: S. Brooks, A. Gelman, G. Jones, X.-L. Meng (Eds.), Handbook of Markov Chain Monte Carlo, Chapman and Hall-CRC Press, Boca Raton, 2011, Ch. 5. doi:10.1201/b10905-6.
[38] 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.
[39] 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.

Appendix A. Probabilistic model of the random generator for Applications 1

In this Appendix, any second-order random quantity $\mathbf{S}$ is defined on a probability space $(\Theta, \mathcal{T}, \mathcal{P})$ and its mathematical expectation $E\{\mathbf{S}\}$ is estimated by $\underline{\mathbf{s}} = (1/N)\sum_{j=1}^{N} \mathbf{s}^j$ using N independent realizations $\{\mathbf{s}^j = \mathbf{S}(\theta_j), j = 1, \ldots, N\}$ of $\mathbf{S}$ with $\theta_j \in \Theta$.

The $\mathbb{R}^\nu$-valued random variable $\mathbf{H} = (H_1, \ldots, H_\nu)$ is written as a partition of $n_p = 3$ independent random vectors $\mathbf{Y}^1, \ldots, \mathbf{Y}^{n_p}$ such that, for $i = 1, \ldots, n_p$, the normalized $\mathbb{R}^{\nu_i}$-valued random variable $\mathbf{Y}^i = (Y^i_1, \ldots, Y^i_{\nu_i})$ is non-Gaussian and such that the estimate of its mean vector is $\underline{\boldsymbol{\eta}}^i = \mathbf{0}_{\nu_i}$ and the estimate of its covariance matrix is $[C_{\mathbf{Y}^i}] = [I_{\nu_i}]$. We have $\nu = \nu_1 + \ldots + \nu_{n_p}$ and we choose $\nu_1 = 10$, $\nu_2 = 20$, and $\nu_3 = 30$.

For $i = 1, 2, 3$, let $[\,b^i]$ be the deterministic matrix in $\mathbb{M}_{\nu_i}$ defined by: $rng('default')$; $[\,b^i] = (0.15 \times rand(\nu_i, \nu_i) + 0.85)/\nu_i$ (in which rng and $rand$ are the Matlab functions). Let $\mathbf{U}^i = 2\,[\,b^i]\,\boldsymbol{\mathcal{U}}^i - \mathbb{1}$ be the $\mathbb{R}^{\nu_i}$-valued random variable in which $\mathbb{1} = (1, \ldots, 1)$ and where $\boldsymbol{\mathcal{U}}^i = (\mathcal{U}^i_1, \ldots, \mathcal{U}^i_{\nu_i})$ is the random vector constituted of ν_i independent uniform random variables on $[0, 1]$, whose N independent realizations are $\{\boldsymbol{\mathcal{U}}^i(\theta_j), j = 1, \ldots, N\}$. The random vectors $\boldsymbol{\mathcal{U}}^1$,

$\mathcal{U}^2$, and $\mathcal{U}^3$ are statistically independent. Let $\boldsymbol{\mathcal{M}}^i = (\mathcal{M}^i_1, \ldots, \mathcal{M}^i_{\nu_i})$ be the $\mathbb{R}^{\nu_i}$-valued random variable in which, for $k = 1, \ldots, \nu_i$, $\mathcal{M}^i_k$ is the random monomial $\mathcal{M}^i_k = \sqrt{k!}\,(U^i_k)^k$ (thus the degree of this monomial is k). Let be $\boldsymbol{\mathcal{M}}^i_c = \boldsymbol{\mathcal{M}}^i - \underline{\mathbf{m}}^i$ in which $\underline{\mathbf{m}}^i$ is the estimate of the mean value of $\boldsymbol{\mathcal{M}}^i$. Let $[C_{\boldsymbol{\mathcal{M}}^i}]$ be the estimate of the covariance matrix of $\boldsymbol{\mathcal{M}}^i$ and let $[L^i]$ be the upper triangular matrix computed from the Cholesky factorization $[C_{\boldsymbol{\mathcal{M}}^i}] = [L^i]^T\,[L^i]$. Therefore, the random vector $\mathbf{Y}^i$ is constructed as $\mathbf{Y}^i = ([L^i]^T)^{-1}\,\boldsymbol{\mathcal{M}}^i_c$ whose the N independent realizations $\{\boldsymbol{\eta}^{i,j}_d, j = 1, \ldots, N\}$ are such that $\boldsymbol{\eta}^{i,j}_d = ([L^i]^T)^{-1}\,\boldsymbol{\mathcal{M}}^i_c(\theta_j)$. The N independent realizations $\{\boldsymbol{\eta}^j_d, j = 1, \ldots, N\}$ of $\mathbf{H}$ are such that $\boldsymbol{\eta}^j_d = (\boldsymbol{\eta}^{1,j}_d, \boldsymbol{\eta}^{2,j}_d, \boldsymbol{\eta}^{3,j}_d) \in \mathbb{R}^\nu = \mathbb{R}^{\nu_1} \times \mathbb{R}^{\nu_2} \times \mathbb{R}^{\nu_3}$. Using these N realizations, the estimate of the mean vector of $\mathbf{H}$ is $\underline{\boldsymbol{\eta}} = \mathbf{0}_\nu$ and the estimate of its covariance matrix is $[C_{\mathbf{H}}] = [I_\nu]$. By construction, we have $\mathbf{Y}^1 = (H_1, \ldots, H_{10})$, $\mathbf{Y}^2 = (H_{11}, \ldots, H_{30})$, and $\mathbf{Y}^3 = (H_{31}, \ldots, H_{60})$.

Appendix B. Model and data for Applications 2

(i) Geometry and surface force field $\boldsymbol{\mathcal{G}}^\Gamma$. The 3D bounded domain is defined by $\Omega =]0, 1.0[\times]0, 0.2[\times]0, 0.1[$. Its boundary is written as $\partial\Omega = \Gamma_0 \cup \Gamma$ in which $\Gamma = \partial\Omega \backslash \Gamma_0$ with $\Gamma_0 = \{\zeta_1 = 1.0\,,\, 0 \leq \zeta_2 \leq 0.2\,,\, 0 \leq \zeta_3 \leq 0.1\}$. The surface force field $\boldsymbol{\mathcal{G}}^\Gamma = (\mathcal{G}^\Gamma_1, \mathcal{G}^\Gamma_2, \mathcal{G}^\Gamma_3)$ is zero on all Γ except on the part $\{\zeta_1 = 0.0\,,\, 0 \leq \zeta_2 \leq 0.2\,,\, 0 \leq \zeta_3 \leq 0.1\}$ for which $\mathcal{G}^\Gamma_1(\zeta) = -4.95 \times 10^7\, N/m^2$, $\mathcal{G}^\Gamma_2(\zeta) = -4.29 \times 10^5\, N/m^2$, and $\mathcal{G}^\Gamma_3(\zeta) = -1.65 \times 10^5\, N/m^2$.

(ii) Probabilistic modeling of the elasticity random field. Random field $\mathbb{K}$ is rewritten as $\mathbb{K}_{ijkh}(\boldsymbol{\zeta}) = [\mathbf{K}(\boldsymbol{\zeta})]_{IJ}$ with $I = (i, j)$ and $J = (k, h)$, in which indices I and J belong to $\{1, \ldots, 6\}$, and where $\{[\mathbf{K}(\boldsymbol{\zeta})], \boldsymbol{\zeta} \in \mathbb{R}^3\}$ is a second-order $\mathbb{M}^+_6$-valued non-Gaussian random field indexed by $\mathbb{R}^3$, which is assumed to be statistically homogeneous. Its mean function $[\underline{K}] \in \mathbb{M}^+_6$ is thus independent of $\boldsymbol{\zeta}$ and corresponds to the elasticity tensor of a homogeneous isotropic elastic medium whose Young modulus is $10^{10}\, N/m^2$ and Poisson coefficient 0.15. The statistical fluctuations of random field $[\mathbf{K}]$ around $[\underline{K}]$ are those of a heterogeneous anisotropic elastic medium. The non-Gaussian $\mathbb{M}^+_6$-valued random field $\{[\mathbf{K}(\boldsymbol{\zeta})]\,, \boldsymbol{\zeta} \in \Omega\}$ is constructed using the stochastic model [38, 39] of random elasticity fields for heterogeneous anisotropic elastic media that are isotropic in statistical mean and exhibit anisotropic statistical fluctuations. Its parameterization consists of three spatial-correlation lengths, one dispersion parameter, and a positive-definite lower bound. Random field $[\mathbf{K}]$ is written, for all $\boldsymbol{\zeta}$ in $\mathbb{R}^3$, as $[\mathbf{K}(\boldsymbol{\zeta})] = [K_\ell] + [\underline{L}_\varepsilon]^T\,[\mathbf{G}_0(\boldsymbol{\zeta})]\,[\underline{L}_\varepsilon]$. The lower-bound matrix is defined by $[K_\ell] = \varepsilon(1 + \varepsilon)^{-1}\,[\underline{K}] \in \mathbb{M}^+_6$ in which ε is chosen equal to 10^{-6}. The upper triangular matrix $[\underline{L}_\varepsilon] \in \mathbb{M}_6$ is written as $[\underline{L}_\varepsilon] = (1+\varepsilon)^{-1/2}\,[\underline{L}]$ in which $[\underline{K}] = [\underline{L}]^T\,[\underline{L}]$ (Cholesky factorization). The non-Gaussian random field $\{[\mathbf{G}_0(\boldsymbol{\zeta})], \boldsymbol{\zeta} \in \mathbb{R}^3\}$, which is indexed by $\mathbb{R}^3$ with values in $\mathbb{M}^+_6$, is homogeneous in $\mathbb{R}^3$ and is a second-order random field (see[39], pp 272-274 for details related to its stochastic modeling and the associated random generator). For all $\boldsymbol{\zeta}$ in $\mathbb{R}^3$, the random matrix $[\mathbf{G}_0(\boldsymbol{\zeta})]$ is written as $[\mathbf{G}_0(\boldsymbol{\zeta})] = [\mathbf{L}_G(\boldsymbol{\zeta})]^T\,[\mathbf{L}_G(\boldsymbol{\zeta})]$ in which $[\mathbf{L}_G(\boldsymbol{\zeta})]$ is an upper (6×6) real triangular random matrix that depends of $n_G = 21$ independent normalized Gaussian random variables. Random field $[\mathbf{G}_0]$ depends on 3 spatial correlation lengths, $L_{\text{corr},1}$, $L_{\text{corr},2}$, $L_{\text{corr},3}$, relative to each one of the three directions ζ_1-, ζ_2-, and ζ_3-axes. It also depends on the dispersion parameter $\delta_G > 0$ that allows for controlling the level of statistical fluctuations. As explained in Section 7.1, only two hyperparameters are kept: L_{corr} and δ_G, for which we have chosen $L_{\text{corr},1} = L_{\text{corr},2} = L_{\text{corr},3} = L_{\text{corr}}$.

(iii) Finite element approximation of the stochastic boundary value problem and definition of random vector $\boldsymbol{U}$. Domain Ω is meshed with $50\times10\times5 = 2\,500$ finite elements using 8-nodes finite elements. There are 3 366 nodes and 10 098 dofs (degrees of freedom) before applying the Dirichlet conditions. The displacements are locked at all the 66 nodes belonging to surface Γ_0 and therefore, there are 198 zero Dirichlet conditions. There are 8 integration points in each finite element. Consequently, there are $N_i = 20\,000$ integration points $\boldsymbol{\zeta}^1, \ldots, \boldsymbol{\zeta}^{N_i}$. Let us consider the $\mathbb{R}^{n_u}$-valued random variable $\mathbf{U}$ made up of all the $n_u = N_i \times n_G = 20\,000\times21 = 420\,000$ independent normalized Gaussian random variables that allow the set $\{[\mathbf{L}_G(\boldsymbol{\zeta}^1)], \ldots, [\mathbf{L}_G(\boldsymbol{\zeta}^{N_i})]\}$ of random matrices to be generated.

(iv) Construction of random vectors $\boldsymbol{Q}$*, and* $\boldsymbol{W}$. The $\mathbb{R}^{n_q}$-valued random variable $\mathbf{Q}$ of the QoIs are constituted of the 10 098 dofs of the discretization of the random displacement field $\mathbf{V}$. The random vector $\mathbf{W} = (W_1, W_2)$ in such that $W_1 = \log(L_{\text{corr}})$ and $W_2 = log(\delta_G)$. The random variables L_{corr} and δ_G are independent and uniform on $[0.1\,, 1.0]$ and $[0.1\,0.5]$, respectively. We then have $L_{\text{corr}} = 0.9\,\mathcal{U}_1 + 0.1$ and $\delta_G = 0.4\,\mathcal{U}_2 + 0.1$ in which $\mathcal{U}_1$ and $\mathcal{U}_2$ are two independent uniform random variable on $[0\,, 1]$.