Computer Methods in Applied Mechanics and Engineering 422 (2024) 116837

Contents lists available at ScienceDirect

Comput. Methods Appl. Mech. Engrg.

journal homepage: www.elsevier.com/locate/cma

Concurrent multiscale simulations of nonlinear random materials using probabilistic learning

Peiyi Chen [a], Johann Guilleminot [b,*], Christian Soize [c]

[a] Department of Mechanical Engineering and Materials Science, Duke University, Durham, NC 27708, USA
[b] Department of Civil and Environmental Engineering, Duke University, Durham, NC 27708, USA
[c] Université Gustave Eiffel, MSME UMR 8208, 5 bd Descartes, 77454 Marne-la-Vallée, France

ARTICLE INFO

Keywords:
Concurrent methods
Nonlinear elasticity
Probabilistic learning
Random media
Surrogates
Uncertainty quantification

ABSTRACT

This work is concerned with the construction of statistical surrogates for concurrent multiscale modeling in structures comprising nonlinear random materials. The development of surrogates approximating a homogenization operator is a fairly classical topic that has been addressed through various methods, including polynomial- and deep-learning-based models. Such approaches, and their extensions to probabilistic settings, remain expensive and hard to deploy when the nonlinear upscaled quantities of interest exhibit large statistical variations (in the case of non-separated scales, for instance) and potential non-locality. The aim of this paper is to present a methodology that addresses this particular setting from the point of view of probabilistic learning. More specifically, we formulate the approximation problem using conditional statistics, and use probabilistic learning on manifolds to draw samples of the nonlinear constitutive model at mesoscale. Two applications, relevant to inverse problem solving and forward propagation, are presented in the context of nonlinear elasticity. We show that the framework enables accurate predictions (in probability law), despite the small amount of training data and the very high levels of nonlinearity and stochasticity in the considered system.

1. Introduction

1.1. Background

Concurrent nonlinear simulations involve the strong coupling between a macroscopic (or structural) formulation and a microscopic description capturing subscale details [1–8]. One popular approach is the so-called FE^2 method [3,4], where information (in the form of a deformation gradient and any adapted stress variable) is transferred back-and-forth between quadrature points at the macroscale and statistical or representative volume elements (depending on whether the separation of scales exists or not). While versatile and powerful, such frameworks require significant computational resources that often surpass the capabilities of intermediate-power computers, especially when the underlying behavior is highly nonlinear. In this context, the development of surrogate models for large-scale systems has become a very active research domain and has generated a substantial body of literature. Various methodologies have been proposed to address this challenge, including (in a non-exhaustive manner) the development of deterministic representations [9–20] and probabilistic/statistical-based approaches [21–26], and more recently, the integration of machine learning (ML) tools, both with and without probabilistic/statistical formulations; see, e.g., [27–36].

* Corresponding author.
E-mail address: johann.guilleminot@duke.edu (J. Guilleminot).

https://doi.org/10.1016/j.cma.2024.116837
Received 3 January 2024; Received in revised form 5 February 2024; Accepted 6 February 2024
Available online 16 February 2024

In most of the above contributions, the multiscale surrogate is built either through polynomial approximations or deep learning models, and applied *locally* at each point of the coarse scale discretization. In these settings, the intrinsic randomness — and potential non-locality — induced by random media with non-separated scales is very challenging to capture due to representation limitations. In this work, we explore an alternative path to address this problem and seek to construct a statistical surrogate model where the forward map of interest (specifically, the non-local constitutive model) is approximated using statistical conditioning. Instead of calibrating a regression model between the input (e.g., the deformation gradient) and the output (say, a stress measure), we aim to directly generate samples from the input–output joint probability measure, and to estimate quantities of interest through conditional means. This viewpoint requires the use of a generative model capable of accurately capturing measure concentration and the (unknown) geometry of the support of the measure in the small data limit — a task that remains particularly challenging for strongly non-Gaussian distributions in high dimensions. We note that the construction of generative models is a vibrant topic across many scientific communities, and providing an extensive review on existing techniques is beyond the scope of this paper. In the present study, we employ probabilistic learning on manifolds (PLoM) [37–39] to perform this task. The choice of this technique is motivated by (*i*) its capability to sample the probability measure defined by the training dataset and in particular, to respect measure concentration and support information (as demonstrated in [40–51]), (*ii*) relative ease of implementation, and (*iii*) its reliance on low-dimensional, interpretable parameterization. Our main contributions are as follows. First, we formulate the approximation of the non-local homogenized response in nonlinear elasticity as a learning problem. Second, we perform extensive numerical studies and address the validation of the framework under two scenarios relevant to inverse problem solving and forward propagation. In the former case, the approach can be used, for instance, to calibrate hyperparameters in the material model at fine scale, integrating data at the coarse scale. The latter case represents the classical surrogate setting with aleatoric uncertainties induced by subscale randomness (without separation of scales). Notice that while the proposed developments are derived in the context of nonlinear elasticity, they remain applicable to other classes of constitutive models — at the expense of adapting the mechanistic parameterization.

This paper is organized as follows. The multiscale mechanistic framework is first introduced in Section 2. The deterministic scale-coupling problems (and their stochastic counterparts) are presented, together with the stochastic model enabling the representation of material randomness at mesoscale. Section 3 provides a comprehensive overview on the probabilistic learning framework, including both theoretical and algorithmic aspects. In Section 4, the proposed framework is applied in the context of finite elasticity. The two aforementioned scenarios are specifically introduced to assess the robustness of the method (in probability law). Concluding comments are finally provided in Section 5.

1.2. Main notation

(i) *Conventions for variables*.
A lower-case Latin or Greek letter, such as x or η, is a deterministic real variable.
A boldface lower-case Latin or Greek letter, such as $\mathbf{x}$ or $\boldsymbol{\eta}$, is a deterministic vector.
An upper-case Latin or Greek letter, such as X or Ξ, is a real-valued random variable.
A boldface upper-case Latin letter, such as $\mathbf{X}$, is a vector-valued random variable.
A lower- or upper-case Latin letter between brackets, such as $[x]$ or $[X]$, is a deterministic matrix.
A boldface upper-case letter between brackets, such as $[\mathbf{X}]$, is a matrix-valued random variable.
(ii) *Probability space, random variable, probability measure, and probability density function.*
For any finite integer $m \geq 1$, the Euclidean space $\mathbb{R}^m$ is equipped with the σ-algebra $\mathcal{B}_{\mathbb{R}^m}$. If $\mathbf{Y}$ is a $\mathbb{R}^m$-valued random variable defined on the probability space $(\Theta, \mathcal{T}, \mathcal{P})$, $\mathbf{Y}$ is a mapping $\theta \mapsto \mathbf{Y}(\theta)$ from Θ into $\mathbb{R}^m$, measurable from $(\Theta, \mathcal{T})$ into $(\mathbb{R}^m, \mathcal{B}_{\mathbb{R}^m})$, and $\mathbf{Y}(\theta)$ is a realization (sample) of $\mathbf{Y}$ for $\theta \in \Theta$. The probability distribution of $\mathbf{Y}$ is the probability measure $P_{\mathbf{Y}}(d\mathbf{y})$ on the measurable set $(\mathbb{R}^m, \mathcal{B}_{\mathbb{R}^m})$ (we will simply say on $\mathbb{R}^m$). The Lebesgue measure on $\mathbb{R}^m$ is denoted by $d\mathbf{y}$ and $P_{\mathbf{Y}}(d\mathbf{y}) = p_{\mathbf{Y}}(\mathbf{y})\, d\mathbf{y}$, with $p_{\mathbf{Y}}$ the probability density function (pdf) on $\mathbb{R}^m$ of $P_{\mathbf{Y}}(d\mathbf{y})$ with respect to $d\mathbf{y}$. Finally, E denotes the mathematical expectation operator.
(iii) *Algebraic notations*.
$\mathbb{R}$: set of all the real numbers.
$\mathbb{R}^n$: Euclidean vector space on $\mathbb{R}$ of dimension n.
$\mathbb{M}_{n,m}$: set of all the $(n \times m)$ real matrices.
$\mathbb{M}_n$: set of all the square $(n \times n)$ real matrices.
$\mathbb{M}_n^+$: set of all the positive-definite symmetric $(n \times n)$ real matrices.
$[I_n]$: identity matrix in $\mathbb{M}_n$.
$\mathbf{x} = (x_1, \ldots, x_n)$: point in $\mathbb{R}^n$.
$\langle \mathbf{x}, \mathbf{y} \rangle = x_1 y_1 + \cdots + x_n y_n$: inner product in $\mathbb{R}^n$.
$\| \mathbf{x} \|$: norm in $\mathbb{R}^n$ such that $\| \mathbf{x} \| = \langle \mathbf{x}, \mathbf{x} \rangle$.
$[x]^T$: transpose of matrix $[x]$.
$\| [x] \|$: Frobenius norm of matrix $[x]$.
$\delta_{kk'}$: Kronecker's symbol.

2. Description of the mechanistic framework

2.1. Definition of the structural problem

Let Ω_{str} be an open bounded domain in $\mathbb{R}^d$ (here, $d = 2$ without loss of methodological generality) representing the reference configuration for the structure of interest, and denote by $\partial\Omega_{\text{str}}$ the boundary of Ω_{str}. For any material point $\mathbf{x} \in \Omega_{\text{str}}$, the spatial point $\mathbf{x}^\varphi$ in the deformed configuration $\Omega^\varphi_{\text{str}}$ is given by $\mathbf{x}^\varphi = \varphi(\mathbf{x})$, where φ is the deformation map. To make the presentation concrete, we assume that the material (at fine scale) is hyperelastic, compressible and isotropic. For any $\mathbf{x} \in \Omega_{\text{str}}$, the deformation gradient $[F]$ is a second-order tensor defined as $[F] = [\boldsymbol{\nabla}_{\mathbf{x}}\mathbf{x}^\varphi]$. The right Cauchy–Green deformation tensor is defined as $[C] = [F]^T[F]$, and the Green–Lagrange strain tensor defined as $[E_{GL}] = \frac{1}{2}([C] - [I])$. For the sake of simplicity, we consider a Saint Venant–Kirchhoff model, with a strain energy density function given by

$$\psi([E_{GL}]) = \frac{\lambda}{2}[\text{tr}([E_{GL}])]^2 + \mu\,\text{tr}([E_{GL}]^2)\,, \tag{1}$$

where λ and μ are the Lamé parameters (see Section 2.3.1 for details). It is well-known that the above Saint Venant–Kirchhoff material is not polyconvex (see, e.g., Section 4.3 in [52]). The use of this model may thus lead to poor numerical stability and pathological behaviors in general. Such issues were not observed in the applications presented in this paper, given the multiscale surrogate modeling context. In particular, the boundary conditions inherited from the structural boundary value problem did not generate asymptotic behavior. The results supporting the relevance of the proposed methodology (and more specifically, the ability to approximate the homogenized constitutive model) are therefore not expected to be fundamentally affected by this choice. Note also that the proposed approach can accommodate other types of constitutive behaviors, and that the above choice pertaining to the strain energy density function is not expected to impact the methodological results presented in this research.

In a general setting, the strong form (resulting from the balance of linear momentum) of the boundary value problem (BVP) in the reference configuration is stated as [52]

$$\text{Div}\,[P(\mathbf{x})] + \mathbf{b}(\mathbf{x}) = \mathbf{0}\,, \quad \forall\,\mathbf{x} \in \Omega_{\text{str}}\,, \tag{2}$$

$$\mathbf{u}(\mathbf{x}) = \overline{\mathbf{u}}(\mathbf{x})\,, \quad \forall\,\mathbf{x} \in \partial\Omega^D_{\text{str}}\,, \tag{3}$$

$$[P(\mathbf{x})]\cdot\mathbf{n}(\mathbf{x}) = \overline{\mathbf{t}}(\mathbf{x})\,, \quad \forall\,\mathbf{x} \in \partial\Omega^N_{\text{str}}\,, \tag{4}$$

where Div denotes the divergence operator in the reference configuration, $[P]$ is the first Piola–Kirchhoff stress tensor defined as

$$[P] = \frac{\partial\psi([F])}{\partial[F]}\,, \tag{5}$$

the vector $\mathbf{b}$ is the body force, $\mathbf{n}$ is unit vector normal to the boundary in the reference configuration, $\overline{\mathbf{u}}$ and $\overline{\mathbf{t}}$ are given smooth vector fields on the Dirichlet and Neumann boundaries, denoted by $\partial\Omega^D_{\text{str}}$ and $\partial\Omega^N_{\text{str}}$ respectively. The solution to the above problem is classically sought (in an appropriate function space) as a stationary point of the following energy functional [52–54]:

$$\Pi(\varphi) = \int_B \psi([F])\,dV - \int_B \mathbf{b}\cdot\varphi\,dV - \int_{\partial B_N} \overline{\mathbf{t}}\cdot\varphi\,dA\,. \tag{6}$$

In this work, we apply a Dirichlet boundary condition $\overline{\mathbf{u}}$ on $\partial\Omega_{\text{str}}$ (i.e., no traction is applied, and the body force is neglected).

2.2. Definition of the macroscopic problem in the context of concurrent multiscale approaches

Let $\Omega_{\text{mac}} \subset \Omega_{\text{str}}$ denote the reference configuration for the subdomain where the surrogate must be constructed, and denote by $\partial\Omega_{\text{mac}}$ its boundary (see Fig. 1). Let $\overline{\mathbf{u}}_{\text{mac}}$ be the restriction of the solution to the structural problem (defined in the previous section) to the boundary $\partial\Omega_{\text{mac}}$. The strong form of the boundary value problem in the reference configuration of Ω_{mac} is stated as

$$\text{Div}\,[P_{\text{mac}}(\mathbf{x})] = \mathbf{0}\,, \quad \forall\,\mathbf{x} \in \Omega_{\text{mac}}\,, \tag{7}$$

$$\mathbf{u}(\mathbf{x}) = \overline{\mathbf{u}}_{\text{mac}}(\mathbf{x})\,, \quad \forall\,\mathbf{x} \in \partial\Omega_{\text{mac}}\,, \tag{8}$$

where $[P_{\text{mac}}]$ is the first Piola–Kirchhoff stress tensor at macroscale.

In order to define the multiscale setting, we consider a statistical volume element $\Omega_{\text{mes}}(\mathbf{x})$ located at point $\mathbf{x} \in \Omega_{\text{mac}}$, and denote by $\partial\Omega_{\text{mes}}$ the boundary of Ω_{mes} (as shown in Fig. 1). Given a finite element discretization of Ω_{str} and at a given iteration in the nonlinear (Newton–Raphson) solver, the concurrent method proceeds by estimating the deformation gradient $[F_{\text{mac}}(\mathbf{x}^q)]$ at any quadrature point $\mathbf{x}^q$ in Ω_{mac}, and by evaluating the apparent first Piola–Kirchhoff stress tensor defined as

$$[P_{\text{mac}}(\mathbf{x}^q)] = \frac{\partial\overline{\psi}_{\text{mac}}([F_{\text{mac}}(\mathbf{x}^q)];\Omega_{\text{mes}}(\mathbf{x}^q))}{\partial[F_{\text{mac}}(\mathbf{x}^q)]}\,, \tag{9}$$

where $\overline{\psi}_{\text{mac}}(\cdot;\Omega_{\text{mes}}(\mathbf{x}^q))$ is the apparent strain energy density function associated with the mesoscopic domain $\Omega_{\text{mes}}(\mathbf{x}^q)$, using localization (through $[F_{\text{mac}}(\mathbf{x}^q)]$) and homogenization (via $\overline{\psi}_{\text{mac}}(\cdot;\Omega_{\text{mes}}(\mathbf{x}^q))$). Note that as previously pointed out, scale separation is not enforced and thus, all quantities obtained by upscaling are termed *apparent*, following the convention introduced by Huet [55]

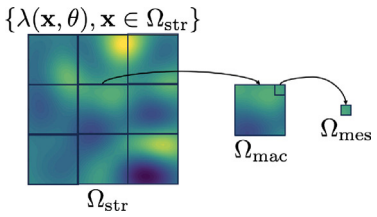


Fig. 1. Definition of scales in the concurrent multiscale simulations. Fluctuations in the first Lamé parameter λ are introduced at fine scale.

(see also [56]). In order to compute $[P_{\text{mac}}]$ at each quadrature point $\mathbf{x}^q \in \Omega_{\text{mac}}$, we use the FE^2 method [4,57] and solve the boundary value problem defined as

$$\text{Div}\,[P_{\text{mes}}(\mathbf{x})] = \mathbf{0}\,, \quad \forall\, \mathbf{x} \in \Omega_{\text{mes}}(\mathbf{x}^q)\,, \tag{10}$$

$$\mathbf{u}(\mathbf{x}) = ([F_{\text{mac}}(\mathbf{x}^q)] - [I])\mathbf{x}\,, \quad \forall\, \mathbf{x} \in \partial\Omega_{\text{mes}}(\mathbf{x}^q)\,, \tag{11}$$

in the reference configuration of $\Omega_{\text{mes}}(\mathbf{x}^q)$, where $[F_{\text{mac}}(\mathbf{x}^q)]$ is the deformation gradient inherited from the macroscale boundary value problem at $\mathbf{x}^q$. The apparent first Piola–Kirchhoff stress tensor at $\mathbf{x}^q$ is then evaluated as [58]

$$[P_{\text{mac}}(\mathbf{x}^q)] = \frac{1}{|\Omega_{\text{mes}}(\mathbf{x}^q)|} \int_{\Omega_{\text{mes}}(\mathbf{x}^q)} [P_{\text{mes}}(\mathbf{x})]\, d\mathbf{x}\,. \tag{12}$$

As we will explain in the next section, the pairs of associated deformation gradients and first Piola–Kirchhoff stress tensors at all quadrature points in the macroscopic domain Ω_{mac} are then used to compute the pairs of associated right Cauchy–Green deformation tensors and second Piola–Kirchhoff stress tensors, which are the (objective) mechanistic variables considered in the probabilistic learning process introduced in Section 3. Note that while the deformation gradient and the first Piola–Kirchhoff stress tensor could also be used in the learning approach, the choice of the right Cauchy–Green deformation tensor and second Piola–Kirchhoff stress tensor as quantities of interest leads to smaller dimensions since both tensors are symmetric. It should also be pointed out that preserving mechanical variables over the entire domain (namely, Ω_{mac}) enables the consideration of a nonlocal apparent constitutive model, as opposed to the calibration of a surrogate at one particular point in the domain (which is more relevant to local constitutive models).

2.3. Description of material uncertainties

2.3.1. Definition of the stochastic model

In this section, we detail the construction of the stochastic model for the strain energy density function defined by Eq. (1). Given the scope of this work, which is focused on the learning perspective rather than stochastic modeling, the hyperelastic model is randomized by defining the first Lamé parameter, λ, as a random field. Models enabling the randomization of all parameters in various classes of strain energy density functions can be found in the references provided after Eq. (15), and in [59,60] for linear elasticity (for all symmetry classes). We also note that results published elsewhere reporting on the (first-order) marginal cross-correlation of elastic moduli suggest that one latent random field may be sufficient to induce multiscale-informed stochasticity in the isotropic case (depending on the material under consideration; see, e.g., [61] for a reinforced composite material).

The first Lamé parameter random field is denoted by $\{\lambda(\mathbf{x}), \mathbf{x} \in \Omega_{\text{str}}\}$, is defined on probability space $(\Theta, \mathcal{T}, P)$, and takes values in $\mathbb{R}^+_{\setminus\{0\}}$. In this paper, we define $\{\lambda(\mathbf{x}), \mathbf{x} \in \Omega_{\text{str}}\}$ as

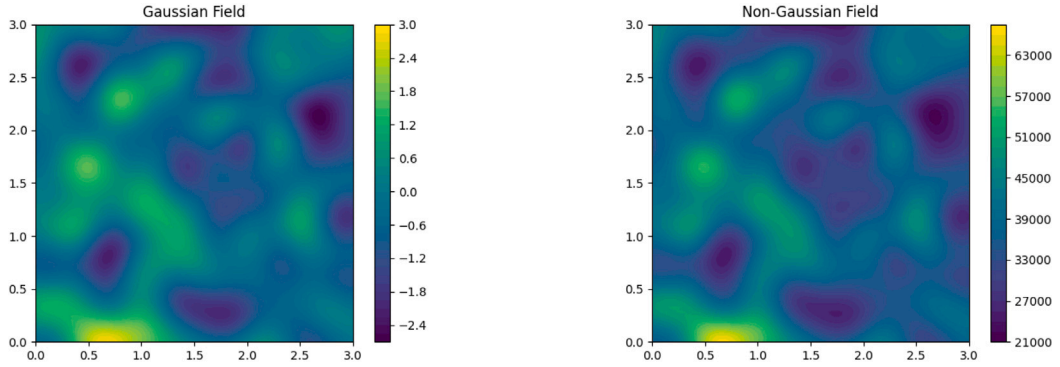
$$\lambda(\mathbf{x}) = \mathcal{H}\,(\Xi(\mathbf{x}))\,, \quad \forall \mathbf{x} \in \Omega_{\text{str}}\,, \tag{13}$$

where $\mathcal{H}$ denotes a so-called transport map, constructed to enforce admissibility (in the almost sure sense), and $\{\Xi(\mathbf{x}), \mathbf{x} \in \mathbb{R}^d\}$ is a centered homogeneous Gaussian random field. This Gaussian field is completely defined by its correlation function $(\mathbf{x}, \mathbf{x}') \mapsto \rho(\mathbf{x}, \mathbf{x}') = E\{\Xi(\mathbf{x})\Xi(\mathbf{x}')\}$, which is taken as

$$\rho(\mathbf{x}, \mathbf{x}') = \prod_{i=1}^{d} \exp\left(-\left(\frac{x_i - x'_i}{\ell_c}\right)^2\right), \quad \forall (\mathbf{x}, \mathbf{x}') \in \mathbb{R}^d \times \mathbb{R}^d\,, \tag{14}$$

for the sake of illustration, with ℓ_c a model parameter such that $\int_0^{+\infty} \exp\left(-(\tau/\ell_c)^2\right)\, d\tau = L_c$, where L_c is the spatial correlation length of the Gaussian random field (which is assumed to be independent of the direction, for simplicity), with $L_c = \ell_c\sqrt{\pi}/2$. Following the methodology introduced in [62] in the context of anisotropic linear elasticity, the transport map is constructed using information theory and the principle of maximum entropy [63–65]; see [66] for an introduction to concepts and methodologies, as well as [67] for specific results in (linear and nonlinear) mechanics of materials. Specifically, $\mathcal{H}$ is defined by imposing that

$$\lambda(\mathbf{x}) = \mathcal{H}\,(\Xi(\mathbf{x})) \sim P_{\text{ME}}\,, \quad \forall \mathbf{x} \in \Omega_{\text{str}}\,, \tag{15}$$



(a) Sample of $\{\Xi(\mathbf{x}), \mathbf{x} \in \Omega_{\text{str}}\}$.

(b) Sample of $\{\lambda(\mathbf{x}), \mathbf{x} \in \Omega_{\text{str}}\}$ computed from (a).

Fig. 2. Realizations of the underlying Gaussian field (left) and material parameter random field (right).

where P_{ME} is the probability measure induced by entropy maximization under constraints. General methodologies and information-theoretic results for a large class of models in nonlinear elasticity can be found in [68,69], for the cases of isotropic incompressible and compressible materials, respectively. Extensions to spatially-dependent anisotropic hyperelastic models can be found in [70,71], and applications including calibration and validation using experimental data are available in [71–73] (see also [74] and the references therein for a review of applications to canonical mechanics problems). Since λ corresponds to an elasticity parameter, results obtained in the context of stochastic linearized elasticity can also be invoked. Accounting for the positiveness constraint, as well as for the existence of second-order moments for the linearized elasticity tensor and its inverse [62,75], it can be shown that P_{ME} corresponds to a Gamma distribution. Denoting by $\underline{\lambda}$ and δ_λ the mean and coefficient of variation of λ, it follows that

$$\mathcal{H} = F^{-1}_{\mathcal{G}(\delta_\lambda^{-2}, \underline{\lambda}\delta_\lambda^{-2})} \circ F_{\mathcal{N}(0,1)}\,, \tag{16}$$

where $F^{-1}_{\mathcal{G}}$ is the inverse cumulative distribution function of the Gamma distribution with shape and scale parameters given by δ_λ^{-2} and $\underline{\lambda}\delta_\lambda^{-2}$, respectively, and $F_{\mathcal{N}(0,1)}$ is the cumulative distribution function of the standard Gaussian distribution. Notice that these hyperparameters can be made spatially-dependent to improve expressiveness in the model: this sophistication is, however, irrelevant for the objectives pursued in this paper.

In the applications presented below, the underlying Gaussian random field $\{\Xi(\mathbf{x}), \mathbf{x} \in \mathbb{R}^d\}$ is sampled using a truncated Karhunen–Loève expansion, with an order of truncation determined such that the L^2 error falls below a given threshold (chosen as 1e−2). Samples for both the Gaussian and non-Gaussian random fields are shown in Fig. 2, for $\underline{\lambda} = 40\,000$, $\delta_\lambda = 0.2$, and $L_c = 0.3$.

2.3.2. Definition of the stochastic boundary value problems

Considering the Lamé parameter random field defined in Section 2.3.1 in the BVPs introduced in Sections 2.1 and 2.2 leads to the definition of stochastic boundary value problems (SBVPs), which are briefly described below for the sake of readability. All equalities below hold in the almost sure sense.

Following the retained modeling setup, the structural stochastic boundary value problem is given by

$$\text{Div}\,[\mathbf{P}(\mathbf{x})] = \mathbf{0}\,, \quad \forall\, \mathbf{x} \in \Omega_{\text{str}}\,, \tag{17}$$

$$\mathbf{U}(\mathbf{x}) = \overline{\mathbf{u}}(\mathbf{x})\,, \quad \forall\, \mathbf{x} \in \partial\Omega^D_{\text{str}}\,, \tag{18}$$

where $[\mathbf{P}]$ is the stochastic Piola–Kirchhoff stress tensor (arising from the randomization of the strain energy density function via λ), $\{\mathbf{U}(\mathbf{x}), \mathbf{x} \in \Omega_{\text{str}}\}$ is the displacement solution random field and $\mathbf{x} \mapsto \overline{\mathbf{u}}(\mathbf{x})$ is the known deterministic field introduced in Section 2.1. Similarly, the macroscopic SBVP on the domain of interest Ω_{mac} (where the statistical surrogate is built) writes

$$\text{Div}\,[\mathbf{P}_{\text{mac}}(\mathbf{x})] = \mathbf{0}\,, \quad \forall\, \mathbf{x} \in \Omega_{\text{mac}}\,, \tag{19}$$

$$\mathbf{U}(\mathbf{x}) = \overline{\mathbf{U}}_{\text{mac}}(\mathbf{x})\,, \quad \forall\, \mathbf{x} \in \partial\Omega_{\text{mac}}\,, \tag{20}$$

where $\{\overline{\mathbf{U}}_{\text{mac}}(\mathbf{x}), \mathbf{x} \in \partial\Omega_{\text{mac}}\}$ is now a random field with values in $\mathbb{R}^d$, due to the fact that the background medium is stochastic. Finally, the SBVP considered in the concurrent approach (for any subdomain $\Omega_{\text{mes}}(\mathbf{x}^q)$ centered at quadrature point $\mathbf{x}^q$ in Ω_{mac}) is

$$\text{Div}\,[\mathbf{P}_{\text{mes}}(\mathbf{x})] = \mathbf{0}\,, \quad \forall\, \mathbf{x} \in \Omega_{\text{mes}}(\mathbf{x}^q)\,, \tag{21}$$

$$\mathbf{U}(\mathbf{x}) = ([\mathbf{F}_{\text{mac}}(\mathbf{x}^q)] - [I])\mathbf{x}\,, \quad \forall\, \mathbf{x} \in \partial\Omega_{\text{mes}}(\mathbf{x}^q)\,, \tag{22}$$

where $[\mathbf{F}_{\text{mac}}(\mathbf{x}^q)]$ is the stochastic deformation gradient at $\mathbf{x}^q$, defined through localization.

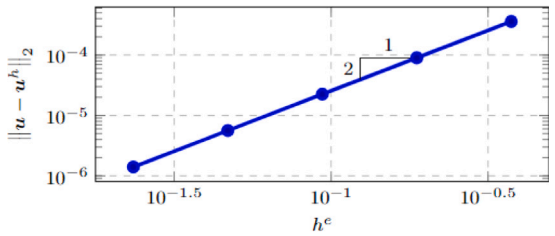
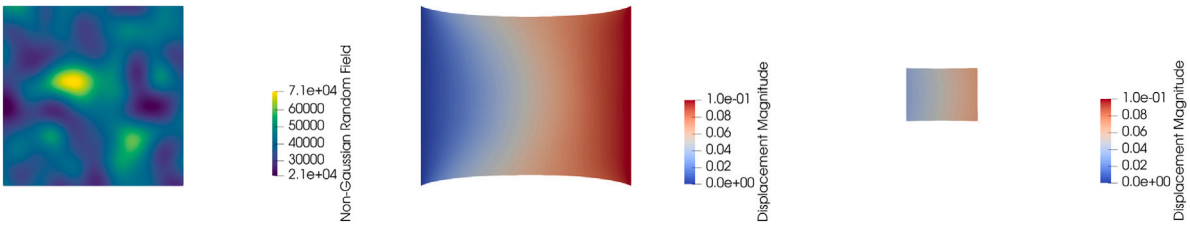


Fig. 3. Convergence of the L^2 error (h-refinement) for the reference solution.



(a) $\{\lambda(\mathbf{x}, \theta), \mathbf{x} \in \Omega_{\text{str}}\}$ (b) $\mathbf{x} \mapsto \|\mathbf{u}_{\text{str}}(\mathbf{x})\|$ (c) $\mathbf{x} \mapsto \|\mathbf{u}_{\text{FE}^2}(\mathbf{x})\|$

Fig. 4. Sample of the material parameter λ (left) and associated displacement magnitudes for the solutions on Ω_{str} (middle) and Ω_{mac} (right).

In this work, we consider the strong stochastic solutions to the weak formulations (using the Galerkin method and a finite element discretization) of the above SBVPs. The Monte Carlo approach is chosen as the stochastic solver. The pairs of right Cauchy–Green deformation tensors and second Piola–Kirchhoff stress tensors (denoted by $\{[C_{\text{mac}}(\mathbf{x}^q)], [S_{\text{mac}}(\mathbf{x}^q)]\}_{q=1}^{Q_{\text{mac}}}$) are collected at all quadrature points, for all samples of the Lamé parameter random field, to constitute the dataset for the probabilistic learning procedure introduced in Section 3.

2.4. Implementation verification for the FE ^{2} method (with deterministic background media)

For the sake of illustration, we consider the structural domain $\Omega_{\text{str}} = [0, 3]^2$, and define the macroscopic domain of interest as $\Omega_{\text{mac}} = [1, 2]^2$. The mesoscopic domain $\Omega_{\text{mes}}(\mathbf{x}^q)$ at quadrature point $\mathbf{x}^q$ is defined by a characteristic length $L_{\Omega_{\text{mes}}} = 0.025$. Spatial discretization is realized using Q4 elements at all mesh resolutions. The number of elements per direction is 15 in Ω_{str}, 5 in Ω_{mac}, and 5 in Ω_{mes}. The Dirichlet boundary conditions applied on the boundary of Ω_{str} are given by

$$\overline{\mathbf{u}}(\mathbf{x}) = \mathbf{0}, \quad x_1 = 0, \quad \forall x_2 \in [0, 3], \tag{23}$$

$$\overline{\mathbf{u}}(\mathbf{x}) = \begin{bmatrix} 0.1 \\ 0 \end{bmatrix}, \quad x_1 = 3, \quad \forall x_2 \in [0, 3]. \tag{24}$$

As described in the previous section, the solution vector $\mathbf{u}$ in Ω_{str} along the boundary $\partial\Omega_{\text{mac}}$ of Ω_{mac} is applied as the Dirichlet boundary condition for the multi-scale problem.

Implementation was performed within the MOOSE finite element framework [76]. A convergence study on the solution on Ω_{str} was first conducted. A manufactured displacement field taken as $\mathbf{u}^{\text{MMS}}(\mathbf{x}) = (0.01\sin(y), 0.01\sin(x))^T$ is considered, with material parameters given by $\lambda = 40\,000$ [kg/cm^2] and $\mu = 10\,000$ [kg/cm^2] (these values, taken from [77], correspond to a soft biological tissue, modeled as a Saint Venant–Kirchhoff material). Dirichlet boundary conditions in accordance with the above solution are prescribed on all boundaries. A body force is defined such that the manufactured solution corresponds to the nonlinear boundary value problem defined in Section 2.1. Regarding numerical solving, a standard Newton–Raphson solver was used with a maximum number of nonlinear iterations set to 25, with a relative tolerance taken as 1e−10, and an absolute tolerance given by 1e−12. The convergence order is measured by the L^2-norm of the difference between the approximation u^h and the reference solution u^{ref} within the domain $[0, 3]^2$. Standard h-convergence is observed, as illustrated in Fig. 3.

Next, the implementation of the FE^2 method was verified by comparing the normalized L^2-norm error between the solution vector (at all nodes) on Ω_{str} without multiscale coupling, and the solution vector (at all nodes) obtained by using the FE^2 method in the subdomain $\Omega_{\text{mac}} = [1, 2]^2$. Fig. 4 shows the first sample of the material random field λ (for $\underline{\lambda} = 40\,000$, $\delta_\lambda = 0.2$, and $L_c = 0.3$), as well as solutions to the structural and macroscale problems. In order to perform a statistical analysis on the error, 500 independent samples of λ were generated. The mean of the normalized L^2 error is 4.69e−4, and the coefficient of variation is 0.22. The probability density function of the normalized L^2 norm error is shown in Fig. 5. These results indicate proper implementation of the concurrent multiscale method, which is used to build the dataset for the probabilistic learning technique (introduced in the next section).

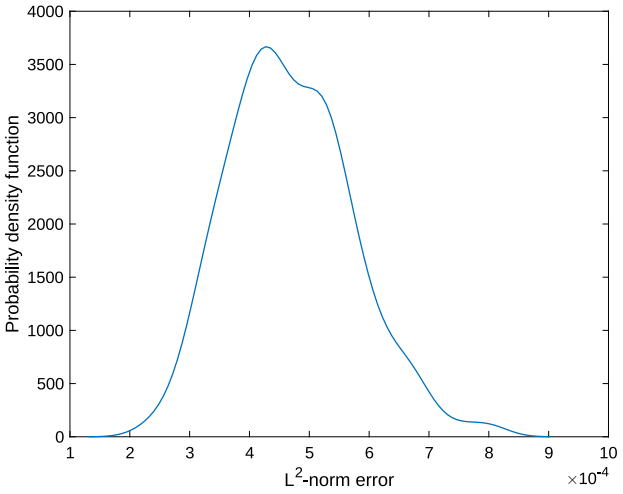


Fig. 5. Probability density function of the normalized L^2-norm error (estimated with 500 samples).

3. Overview of the probabilistic learning on manifolds (PLoM) algorithm and its parameterization

In this section, we provide a concise overview of the PLoM algorithm. The reason for providing this review is twofold. First, we aim to assist readers in analyzing and comprehending the underlying parameterization, the values chosen for the parameters, and the results pertaining to algorithmic control and convergence. Second, there is no published paper that summarizes all the ingredients of the PLoM approach that are used in this work. Early developments addressing, for instance, the quantification of probability measure concentration and the estimation of the smoothing parameter in the calculation of the diffusion maps basis and its truncation order, are disseminated in a series of papers (see below). On the other hand, some presented results are new, including the expression of the relaxation parameters as a function of the iteration number (in the implementation of the learning algorithm), and the expression of the drift matrix associated with the normalization condition.

The PLoM approach [37–39] starts with the consideration of a training dataset $\mathcal{D}_d$, comprising a relatively small number n_d of points generated from an underlying stochastic manifold associated with a $\mathbb{R}^n$-valued random variable $\mathbf{X} = (\mathbf{Q}, \mathbf{W})$, defined on a probability space $(\Theta, \mathcal{T}, \mathcal{P})$. Here, $\mathbf{Q}$ represents the quantity of interest and is a $\mathbb{R}^{n_q}$-valued random variable, while $\mathbf{W}$ denotes the control parameter and is a $\mathbb{R}^{n_w}$-valued random variable. The total dimension is $n = n_q + n_w$. Another $\mathbb{R}^{n_u}$-valued random variable $\mathbb{U}$, defined on $(\Theta, \mathcal{T}, \mathcal{P})$, is also considered as an uncontrolled parameter. The random variable $\mathbf{Q}$ is assumed to be expressed as $\mathbf{Q} = \mathbf{f}(\mathbb{U}, \mathbf{W})$, where the measurable mapping $\mathbf{f}$ is not explicitly known. The joint probability distribution $P_{\mathbf{W},\mathbb{U}}(d\mathbf{w}, d\mathbb{u})$ of $\mathbf{W}$ and $\mathbb{U}$ is assumed to be given. The non-Gaussian probability measure $P_{\mathbf{X}}(\mathbf{x}) = P_{\mathbf{Q},\mathbf{W}}(d\mathbf{q}, d\mathbf{w})$ of $\mathbf{X} = (\mathbf{Q}, \mathbf{W})$ is concentrated in a region of $\mathbb{R}^n$, for which the only available information is the cloud of points in the training dataset $\mathcal{D}_d$. The PLoM method enables the generation of the learned dataset $\mathcal{D}_{\text{ar}}$ for $\mathbf{X}$, consisting of $n_{\text{MC}} \gg n_d$ points (learned realizations) generated by the non-Gaussian probability measure estimated using the training dataset. The preservation of the probability measure concentration is guaranteed by the utilization of a diffusion-maps basis, which enriches the available information from the training dataset. Utilizing the learned dataset $\mathcal{D}_{\text{ar}}$, PLoM enables the computation of conditional statistics, such as $\mathbf{w} \mapsto P_{\mathbf{O}|\mathbf{W}}(d\mathbf{o}|\mathbf{W} = \mathbf{w})$, on $\mathcal{C}_w$. Here, $\mathbf{O} = \boldsymbol{\xi}(\mathbf{Q})$, where $\boldsymbol{\xi}$ is a measurable mapping from $\mathbb{R}^{n_q}$ into $\mathbb{R}^{n_o}$, allowing for the construction of statistical surrogate models (metamodels) within a probabilistic framework. The formulas for the computation of conditional mathematical expectations, conditional probability density functions, and conditional cumulative distribution functions, given any $\mathbf{w}_0$ in $\mathcal{C}_w$, are given in Appendix.

The training dataset $\mathcal{D}_d$ comprises n_d independent realizations $\mathbf{x}_d^j = (\mathbf{q}_d^j, \mathbf{w}_d^j)$ in $\mathbb{R}^n = \mathbb{R}^{n_q} \times \mathbb{R}^{n_w}$ for $j \in \{1, \ldots, n_d\}$ of random variable $\mathbf{X} = (\mathbf{Q}, \mathbf{W})$, in which $\mathbf{q}_d^j = \mathbf{f}(\mathbb{u}_d^j, \mathbf{w}_d^j)$. The PLoM method allows for generating the learned dataset $\mathcal{D}_{\text{ar}}$, consisting of $n_{\text{ar}} \gg n_d$ learned realizations $\{\mathbf{x}_{\text{ar}}^\ell, \ell = 1, \ldots, n_{\text{ar}}\}$ of random vector $\mathbf{X}$. Once the learned dataset is constructed, the learned realizations for $\mathbf{Q}$ and $\mathbf{W}$ can be extracted as $(\mathbf{q}_{\text{ar}}^\ell, \mathbf{w}_{\text{ar}}^\ell) = \mathbf{x}_{\text{ar}}^\ell$ for $\ell = 1, \ldots, n_{\text{ar}}$.

3.1. Construction of a reduced representation

The n_d independent realizations $\{\mathbf{x}_d^j, j = 1, \ldots, n_d\}$ are represented by the matrix $[x_d] = [\mathbf{x}_d^1 \ldots \mathbf{x}_d^{n_d}]$ in $\mathbb{M}_{n,n_d}$. Let $[\mathbf{X}] = [\mathbf{X}^1, \ldots, \mathbf{X}^{n_d}]$ be the random matrix with values in $\mathbb{M}_{n,n_d}$, where its columns are n_d independent copies of random vector $\mathbf{X}$. Utilizing Principal Component Analysis (PCA) of $\mathbf{X}$, random matrix $[\mathbf{X}]$ is written as,

$$[\mathbf{X}] = [\underline{x}] + [\varphi]\,[\mu]^{1/2}\,[\mathbf{H}]\,, \tag{25}$$

where $[\mathbf{H}] = [\mathbf{H}^1, \ldots, \mathbf{H}^{n_d}]$ is a $\mathbb{M}_{\nu,n_d}$-valued random matrix ($\nu \leq n$), and $[\mu]$ is the $(\nu \times \nu)$ diagonal matrix of the ν positive eigenvalues of the empirical estimate of the covariance matrix of $\mathbf{X}$. The $(n \times \nu)$ matrix $[\varphi]$ consists of the associated eigenvectors

such $[\varphi]^T[\varphi] = [I_\nu]$. The matrix $[\underline{x}]$ in $\mathbb{M}_{n,n_d}$ has identical columns, each being equal to the empirical estimate $\underline{\mathbf{x}} \in \mathbb{R}^n$ of the mean value of random vector $\mathbf{X}$. The columns of $[\mathbf{H}]$ are n_d independent copies of a random vector $\mathbf{H}$ with values in $\mathbb{R}^\nu$, satisfying the normalization conditions, $E\{\mathbf{H}\} = \mathbf{0}_\nu$ and $E\{\mathbf{H}\otimes\mathbf{H}\} = [I_\nu]$. The realization $[\eta_d] = [\boldsymbol{\eta}_d^1 \dots \boldsymbol{\eta}_d^{n_d}] \in \mathbb{M}_{\nu,n_d}$ of $[\mathbf{H}]$ is computed by $[\eta_d] = [\mu]^{-1/2}[\varphi]^T([x_d] - [\underline{x}])$. The value ν is classically calculated in order that the L^2- error function $\nu \mapsto \mathrm{err}_{\mathbf{X}}(\nu)$, defined by

$$\mathrm{err}_{\mathbf{X}}(\nu) = 1 - \frac{\sum_{\alpha=1}^{\nu}\mu_\alpha}{E\{\|\mathbf{X}\|^2\}}\,, \tag{26}$$

be smaller than $\varepsilon_{\mathrm{PCA}}$. If $\nu < n-1$, statistical reduction occurs.

3.2. Probability measure of $\mathbf{H}$

The probability measure $P_{\mathbf{H}}$ of $\mathbf{H}$ has to be estimated in order to construct the probability measure of random matrix $[\mathbf{H}]$ used in the PLoM methodology. Let $P_{\mathbf{H}}(d\boldsymbol{\eta}) = p_{\mathbf{H}}(\boldsymbol{\eta})\,d\boldsymbol{\eta}$ be the probability measure on $\mathbb{R}^\nu$ of $\mathbf{H}$, whose probability density function $\boldsymbol{\eta}\mapsto p_{\mathbf{H}}(\boldsymbol{\eta})$ on $\mathbb{R}^\nu$ is estimated by using the Gaussian kernel-density estimation (KDE) with the training dataset $\mathcal{D}_{\mathrm{train}}(\boldsymbol{\eta}) = \{\boldsymbol{\eta}^j, j = 1,\dots,n_d\}$,

$$p_{\mathbf{H}}(\boldsymbol{\eta}) = \frac{1}{n_d}\sum_{j=1}^{n_d}\frac{1}{(\sqrt{2\pi}\,\hat{s}\,)^\nu}\exp\left(-\frac{1}{2\hat{s}^2}\,\|\,\frac{\hat{s}}{s_\nu}\,\boldsymbol{\eta}^j - \boldsymbol{\eta}\,\|^2\right)\quad,\quad \forall\boldsymbol{\eta}\in\mathbb{R}^\nu\,. \tag{27}$$

In these equations, $\hat{s}_\nu$ is a modification of the standard Silverman bandwidth s_ν, defined by

$$\hat{s}_\nu = \frac{s_\nu}{\sqrt{s_\nu^2 + \frac{n_d-1}{n_d}}}\quad,\quad s_\nu = \left\{\frac{4}{n_d(2+\nu)}\right\}^{1/(\nu+4)}\,.$$

With such a modification, the normalization of $\mathbf{H}$ is preserved for any value of n_d, that is to say,

$$E\{\mathbf{H}\} = \int_{\mathbb{R}^\nu}\boldsymbol{\eta}\,p_{\mathbf{H}}(\boldsymbol{\eta})\,d\boldsymbol{\eta} = \frac{1}{2\hat{s}^2}\,\underline{\hat{\boldsymbol{\eta}}} = \mathbf{0}_\nu\,,$$

$$E\{\mathbf{H}\otimes\mathbf{H}\} = \int_{\mathbb{R}^\nu}(\boldsymbol{\eta}\otimes\boldsymbol{\eta})\,p_{\mathbf{H}}(\boldsymbol{\eta})\,d\boldsymbol{\eta} = \hat{s}^2\,[I_\nu] + \frac{\hat{s}^2}{s_\nu^2}\,\frac{(n_d-1)}{n_d}\,[\widehat{C}_{\mathbf{H}}] = [I_\nu]\,,$$

where $\underline{\hat{\boldsymbol{\eta}}} \in \mathbb{R}^\nu$ and $[\widehat{C}_{\mathbf{H}}] \in \mathbb{M}_\nu^+$ are the estimates of the mean value and the covariance matrix of $\mathbf{H}$, performed with $\mathcal{D}_{\mathrm{train}}(\boldsymbol{\eta})$. Theorem 3.1 in [38] proves that, for all $\boldsymbol{\eta}$ fixed in $\mathbb{R}^\nu$, Eq. (27) is a consistent estimation of the sequence $\{p_{\mathbf{H}}\}_{n_d}$ for $n_d\to+\infty$.

3.3. Development of a reduced-order basis using diffusion maps

To preserve the concentration of the learned realizations in the region where the points of the training dataset are concentrated, PLoM relies on an algebraic basis in vector space $\mathbb{R}^{n_d}$, constructed using the diffusion-maps basis [78]. Let $[K]$ and $[b]$ be matrices such that, for all i and j in $\{1,\dots,n_d\}$, $[K]_{ij} = \exp\{-(4\,\varepsilon_{\mathrm{DM}})^{-1}\|\boldsymbol{\eta}_d^i - \boldsymbol{\eta}_d^j\|^2\}$ and $[b]_{ij} = \delta_{ij}\,b_i$ with $b_i = \sum_{j=1}^{n_d}[K]_{ij}$, where $\varepsilon_{\mathrm{DM}} > 0$ is a smoothing parameter. Let $[\mathbb{P}] = [b]^{-1}[K]$ be the matrix in $\mathbb{M}_{n_d}$, with positive entries, satisfying $\sum_{j=1}^{n_d}[\mathbb{P}]_{ij} = 1$ for all $i = 1,\dots,n_d$. Matrix $[\mathbb{P}]$ can be regarded as the transition matrix of a Markov chain that represents the probability of transition in one step. The eigenvalues $\lambda_1,\dots,\lambda_{n_d}$ and the associated eigenvectors $\boldsymbol{\psi}^1,\dots,\boldsymbol{\psi}^{n_d}$ of the right-eigenvalue problem $[\mathbb{P}]\,\boldsymbol{\psi}^\alpha = \lambda_\alpha\,\boldsymbol{\psi}^\alpha$ satisfy $1 = \lambda_1 > \lambda_2 \ge \dots \ge \lambda_{n_d}$ and are computed by solving the generalized eigenvalue problem $[K]\,\boldsymbol{\psi}^\alpha = \lambda_\alpha\,[b]\,\boldsymbol{\psi}^\alpha$ with the normalization condition $\langle[b]\,\boldsymbol{\psi}^\alpha,\boldsymbol{\psi}^\beta\rangle = \delta_{\alpha\beta}$. The eigenvector $\boldsymbol{\psi}^1$ associated with $\lambda_1 = 1$ is a constant vector. For a given integer $\kappa \ge 0$, the diffusion-maps basis $\{\mathbf{g}^1,\dots,\mathbf{g}^\alpha,\dots,\mathbf{g}^{n_d}\}$ forms a vector basis of $\mathbb{R}^{n_d}$ defined by $\mathbf{g}^\alpha = \lambda_\alpha^\kappa\,\boldsymbol{\psi}^\alpha$. The reduced-order diffusion-maps basis of order m is defined, for a given integer m, as the set $\{\mathbf{g}^1,\dots,\mathbf{g}^m\}$, represented by the matrix $[g_m] = [\mathbf{g}^1\dots\mathbf{g}^m] \in \mathbb{M}_{n_d,m}$ with $\mathbf{g}^\alpha = (g_1^\alpha,\dots,g_{n_d}^\alpha)$ and $[g_m]_{j\alpha} = g_j^\alpha$. This basis depends on two parameters, $\varepsilon_{\mathrm{DM}}$ and m, which need to be identified. It is proven in [38], that the PLoM method does not depend on κ, which can therefore be chosen to 0. We aim to determine the optimal value $m_{\mathrm{opt}} \le n_d$ for m and the smallest value $\varepsilon_{\mathrm{opt}} > 0$ for $\varepsilon_{\mathrm{DM}}$ such that (see [39])

$$1 = \lambda_1 > \lambda_2(\varepsilon_{\mathrm{opt}}) \simeq \dots \simeq \lambda_{m_{\mathrm{opt}}}(\varepsilon_{\mathrm{opt}}) \gg \lambda_{m_{\mathrm{opt}}+1}(\varepsilon_{\mathrm{opt}}) \ge \dots \ge \lambda_{n_d}(\varepsilon_{\mathrm{opt}}) > 0\,, \tag{28}$$

with an amplitude jump equal to an order of magnitude (a factor 10, as demonstrated in [38]) between $\lambda_{m_{\mathrm{opt}}}(\varepsilon_{\mathrm{opt}})$ and $\lambda_{m_{\mathrm{opt}}+1}(\varepsilon_{\mathrm{opt}})$. A more detailed analysis leads to the following algorithm for estimating $\varepsilon_{\mathrm{opt}}$ and m_{opt}. Let $\varepsilon_{\mathrm{DM}} \mapsto \mathrm{Jump}(\varepsilon_{\mathrm{DM}})$ be the function on $]0,+\infty[$ defined by

$$\mathrm{Jump}(\varepsilon_{\mathrm{DM}}) = \lambda_{m_{\mathrm{opt}}+1}(\varepsilon_{\mathrm{DM}})/\lambda_2(\varepsilon_{\mathrm{DM}})\,. \tag{29}$$

The algorithm is as follows: set the value of m to $m_{\mathrm{opt}} = \nu + 1$ and identify the smallest possible value $\varepsilon_{\mathrm{opt}}$ of $\varepsilon_{\mathrm{DM}}$ such that $\mathrm{Jump}(\varepsilon_{\mathrm{opt}}) \le 0.1$ and Eq. (28) is satisfied.

3.4. Reduced-order representation of random matrices [**H**] and [**X**] to preserve probability measure concentration

The diffusion-maps vectors $\mathbf{g}^1, \ldots, \mathbf{g}^m \in \mathbb{R}^{n_d}$ span a subspace of $\mathbb{R}^{n_d}$ that characterizes, for the optimal values m_{opt} and ε_{opt} of m and ε_{DM}, the local geometry structure of dataset $\{\boldsymbol{\eta}_d^j, j = 1, \ldots, n_d\}$. The PLoM method introduces the $\mathbb{M}_{\nu,n_d}$-valued random matrix $[\mathbf{H}_m] = [\mathbf{Z}_m]\,[g_m]^T$ with $m \leq n_d$, corresponding to a data-reduction representation of random matrix [**H**], in which $[\mathbf{Z}_m]$ is a $\mathbb{M}_{\nu,m}$-valued random matrix. The MCMC generator of random matrix $[\mathbf{Z}_m]$ is chosen from the class of Hamiltonian Monte Carlo methods, explicitly described in [37], and mathematically detailed in Theorem 6.3 of [38]. For generating the learned dataset, the best probability measure of $[\mathbf{H}_m]$ is obtained for $m = m_{\text{opt}}$ and by using the previously defined basis $[g_{m_{\text{opt}}}]$. For these optimal quantities m_{opt} and $[g_{m_{\text{opt}}}]$, the generator allows for computing n_{MC} realizations $\{[\mathbf{z}_{\text{ar}}^\ell], \ell = 1, \ldots, n_{\text{MC}}\}$ of $[\mathbf{Z}_{m_{\text{opt}}}]$ and therefore, for deducing the n_{MC} realizations $\{[\boldsymbol{\eta}_{\text{ar}}^\ell], \ell = 1, \ldots, n_{\text{MC}}\}$ of $[\mathbf{H}_{m_{\text{opt}}}]$. The reshaping of matrix $[\boldsymbol{\eta}_{\text{ar}}^\ell] \in \mathbb{M}_{\nu,n_d}$ allows for obtaining $n_{\text{ar}} = n_{\text{MC}} \times n_d$ learned realizations $\{\boldsymbol{\eta}_{\text{ar}}^{\ell'}, \ell' = 1, \ldots, n_{\text{ar}}\}$ of **H**. These learned realizations enable the estimation of converged conditional statistics, which are then utilized to construct statistical surrogate models related to **X**, and subsequently, to (**Q**, **W**).

3.5. Quantifying the probability measure concentration of random matrix $[\mathbf{H}_{m_{\text{opt}}}]$

For $m \leq n_d$, the probability measure concentration of random matrix $[\mathbf{H}_m]$ is defined (see [39]) by:

$$d_{n_d}^2(m) = E\{\|[\mathbf{H}_m] - [\eta_d]\|^2\}/\|[\eta_d]\|^2\,. \tag{30}$$

Let $\mathcal{M}_{\text{opt}} = \{m_{\text{opt}}, m_{\text{opt}} + 1, \ldots, n_d\}$, where m_{opt} represents the optimal value of m as defined earlier. Theorem 7.8 of [38] shows that $\min_{m \in \mathcal{M}_{\text{opt}}} d_{n_d}^2(m) \leq 1 + m_{\text{opt}}/(n_d - 1) < d_{n_d}^2(n_d)$, indicating that the PLoM method, for $m = m_{\text{opt}}$ and $[g_{m_{\text{opt}}}]$, is a better method than the standard one corresponding to $d_{n_d}^2(n_d) = 1 + n_d/(n_d - 1) \simeq 2$. Using the n_{MC} realizations $\{[\boldsymbol{\eta}_{\text{ar}}^\ell], \ell = 1, \ldots, n_{\text{MC}}\}$ of $[\mathbf{H}_{m_{\text{opt}}}]$, we have the estimate $d_{n_d}^2(m_{\text{opt}}) \simeq (1/n_{\text{MC}}) \sum_{\ell=1}^{n_{\text{MC}}} \{\|[\boldsymbol{\eta}_{\text{ar}}^\ell] - [\eta_d]\|^2\}/\|[\eta_d]\|^2$.

3.6. Generation of learned realizations $\{\boldsymbol{\eta}_{\text{ar}}^{\ell'}, \ell' = 1, \ldots, n_{\text{ar}}\}$ for the random vector **H**

The MCMC generator is detailed in [37]. Let $\{([\mathcal{Z}(t)], [\mathcal{Y}(t)]), t \in \mathbb{R}^+\}$ be the unique asymptotic (as $t \to +\infty$) stationary diffusion stochastic process with values in $\mathbb{M}_{\nu,m_{\text{opt}}} \times \mathbb{M}_{\nu,m_{\text{opt}}}$, representing the following reduced-order ISDE (stochastic nonlinear second-order dissipative Hamiltonian dynamic system), for $t > 0$,

$$d[\mathcal{Z}(t)] = [\mathcal{Y}(t)]\,dt\,,$$

$$d[\mathcal{Y}(t)] = [\mathcal{L}([\mathcal{Z}(t)])]\,dt - \frac{1}{2} f_0\,[\mathcal{Y}(t)]\,dt + \sqrt{f_0}\,[d\mathcal{W}^{\text{wien}}(t)]\,,$$

with $[\mathcal{Z}(0)] = [\eta_d]\,[a]$ and $[\mathcal{Y}(0)] = [\mathbf{N}]\,[a]$, in which

$$[a] = [g_{m_{\text{opt}}}]\,([g_{m_{\text{opt}}}]^T\,[g_{m_{\text{opt}}}])^{-1} \in \mathbb{M}_{n_d,m_{\text{opt}}}\,.$$

(i) $[\mathcal{L}([\mathcal{Z}(t)])] = [L([\mathcal{Z}(t)]\,[g_{m_{\text{opt}}}]^T)]\,[a]$ is a random matrix with values in $\mathbb{M}_{\nu,m_{\text{opt}}}$. For all $[u] = [\mathbf{u}^1 \ldots \mathbf{u}^{n_d}]$ in $\mathbb{M}_{\nu,n_d}$ with $\mathbf{u}^j = (u_1^j, \ldots, u_\nu^j)$ in $\mathbb{R}^\nu$, the matrix $[L([u])]$ in $\mathbb{M}_{\nu,n_d}$ is defined, for all $k = 1, \ldots, \nu$ and for all $j = 1, \ldots, n_d$, by

$$[L([u])]_{kj} = \frac{1}{p(\mathbf{u}^j)}\,\{\boldsymbol{\nabla}_{\mathbf{u}^j}\,p(\mathbf{u}^j)\}_k\,, \tag{31}$$

$$p(\mathbf{u}^j) = \frac{1}{n_d} \sum_{j'=1}^{n_d} \exp\{-\frac{1}{2\hat{s}_\nu^2}\|\frac{\hat{s}_\nu}{s_\nu}\boldsymbol{\eta}^{j'} - \mathbf{u}^j\|^2\}\,,$$

$$\boldsymbol{\nabla}_{\mathbf{u}^j}\,p(\mathbf{u}^j) = \frac{1}{\hat{s}_\nu^2\,n_d} \sum_{j'=1}^{n_d} (\frac{\hat{s}_\nu}{s_\nu}\boldsymbol{\eta}^{j'} - \mathbf{u}^j)\,\exp\{-\frac{1}{2\hat{s}_\nu^2}\|\frac{\hat{s}_\nu}{s_\nu}\boldsymbol{\eta}^{j'} - \mathbf{u}^j\|^2\}\,.$$

(ii) $[\mathcal{W}^{\text{wien}}(t)] = [\mathbf{W}^{\text{wien}}(t)]\,[a]$ where $\{[\mathbf{W}^{\text{wien}}(t)], t \in \mathbb{R}^+\}$ is the $\mathbb{M}_{\nu,n_d}$-valued normalized Wiener process.
(iii) $[\mathbf{N}]$ is the $\mathbb{M}_{\nu,n_d}$-valued normalized Gaussian random matrix that is independent of process $[\mathbf{W}^{\text{wien}}]$.
(iv) We then have $[\mathbf{Z}_{m_{\text{opt}}}] = \lim_{t \to +\infty} [\mathcal{Z}(t)]$ in probability distribution. The Störmer–Verlet scheme is used for solving the reduced-order ISDE (see [37]), which allows for generating the learned realizations, $[z_{\text{ar}}^1], \ldots, [z_{\text{ar}}^{n_{\text{MC}}}]$, and then, generating the learned realizations $[\eta_{\text{ar}}^1], \ldots, [\eta_{\text{ar}}^{n_{\text{MC}}}]$ such that $[\eta_{\text{ar}}^\ell] = [z_{\text{ar}}^\ell]\,[g_{m_{\text{opt}}}]^T$. It should be noted that the calculation of the realizations of $[\mathbf{Z}_{m_{\text{opt}}}]$ is done in parallel computation, each realization $[z_{\text{ar}}^\ell]$ of $[\mathbf{Z}_{m_{\text{opt}}}]$ being calculated on a "worker" associated with a realization $[\mathcal{W}^{\text{wien},\ell}]$ of the Wiener process $[\mathcal{W}^{\text{wien}}]$.
(v) The free parameter f_0, satisfying $0 < f_0 < 4/\hat{s}_\nu$, allows for the control of the dissipation term in the nonlinear second-order dynamic system (a dissipative Hamiltonian system) to quickly damp the transient effects induced by the initial conditions. A commonly used value is $f_0 = 4$ (noting that $\hat{s}_\nu < 1$). Consequently, the ISDE is solved over the interval $]0, T]$, where T depends on f_0 and represents the smallest integration final time allowing $[\mathbf{Z}_{m_{\text{opt}}}]$ to be chosen as $[\mathcal{Z}(T)]$ while being in the stationary regime.
(vi) The learned realizations $\{\mathbf{x}_{\text{ar}}^{\ell'}, \ell' = 1, \ldots, n_{\text{ar}}\}$ of random vector **X** are then obtained by reshaping the realizations $\{[x_{\text{ar}}^\ell] = [\underline{x}] + [\varphi]\,[\mu]^{1/2}\,[\eta_{\text{ar}}^\ell], \ell = 1, \ldots, n_{\text{MC}}\}$ (see Eq. (25)) with $n_{\text{ar}} = n_{\text{MC}} \times n_d$.

3.7. Preserving normalization conditions through constraints on second-order moments of $\mathbf{H}$

In general, the mean value of $\mathbf{H}$ estimated using the n_{ar} learned realizations $\{\boldsymbol{\eta}_{\text{ar}}^{\ell'}, \ell' = 1, \dots, n_{\text{ar}}\}$, is sufficiently close to zero. Similarly, the estimate of the covariance matrix of $\mathbf{H}$ is also sufficiently close to the identity matrix. However, there are instances where the mean value may not be sufficiently small, and the diagonal entries of the estimated covariance matrix can fall below 1. The normalization conditions can be reestablished during the learning algorithm described in Section 3.6 by imposing, for all $k = 1, \dots, \nu$, the constraints $E\{H_k\} = 0$ and $E\{(H_k)^2\} = 1$. These constraints can be globally rewritten as

$$E\{\mathbf{h}(\mathbf{H})\} = \mathbf{b} \quad \text{on} \quad \mathbb{R}^{n_c}\,, \tag{32}$$

in which $n_c = 2\nu$. Here, the function $\mathbf{h} = (h_1, \dots, h_{n_c})$ and the vector $\mathbf{b} = (b_1, \dots, b_{n_c})$ are defined such that, for all k in $\{1, \dots, \nu\}$, we have $h_k(\mathbf{H}) = H_k$, $h_{k+\nu}(\mathbf{H}) = (H_k)^2$, $b_k = 0$, and $b_{k+\nu} = 1$.

(i) Methodology for imposing the constraint in the learning algorithm. We apply the Kullback–Leibler minimum cross-entropy principle as proposed in [79,80]. Let $\mathbf{H}_c$ be the $\mathbb{R}^\nu$-valued random variable that satisfies the constraint defined by Eq. (31), expressed as $E\{\mathbf{h}(\mathbf{H}_c)\} = \mathbf{b}$. The learned probability measure $P_{\mathbf{H}_c}(d\boldsymbol{\eta}) = p_{\mathbf{H}_c}(\boldsymbol{\eta})\, d\boldsymbol{\eta}$, represented by a density $p_{\mathbf{H}_c}$ on $\mathbb{R}^\nu$, which satisfies the constraint and which is closest to $p_{\mathbf{H}}$ defined by Eq. (27), is the solution of the following optimization problem,

$$p_{\mathbf{H}_c} = \arg \min_{p \in \mathcal{C}_{\text{ad},p}} \int_{\mathbb{R}^\nu} p(\boldsymbol{\eta}) \log\left(\frac{p(\boldsymbol{\eta})}{p_{\mathbf{H}}(\boldsymbol{\eta})}\right) d\boldsymbol{\eta}\,, \tag{33}$$

in which the admissible set $\mathcal{C}_{\text{ad},p}$ is defined by

$$\mathcal{C}_{\text{ad},p} = \left\{ \boldsymbol{\eta} \mapsto p(\boldsymbol{\eta}) : \mathbb{R}^\nu \to \mathbb{R}^+\,, \int_{\mathbb{R}^\nu} p(\boldsymbol{\eta})\, d\boldsymbol{\eta} = 1\,, \int_{\mathbb{R}^\nu} \mathbf{h}(\boldsymbol{\eta})\, p(\boldsymbol{\eta})\, d\boldsymbol{\eta} = \mathbf{b} \right\}. \tag{34}$$

It is proven that there exists a unique solution to the optimization problem defined by Eqs. (33) and (34), which is reformulated using Lagrange multipliers to account for the constraints in the admissible set (refer to Proposition 1 in [80] for the construction of the probability measure of $\mathbf{H}_c$ and the proof of its existence and uniqueness).

(ii) Learning algorithm implementation. To take into account the constraints in the learning algorithm defined in Section 3.6, Eq. (31) is replaced by the following one,

$$[L_\lambda([u])]_{kj} = \frac{1}{p(\mathbf{u}^j)} \{\boldsymbol{\nabla}_{\mathbf{u}^j}\, p(\mathbf{u}^j)\}_k - \lambda_k - 2\,\lambda_{k+\nu} u_k^j\,.$$

in which the Lagrange multiplier $\lambda \in \mathbb{R}^{n_c}$, associated with the constraint defined by Eq. (32), is calculated using an iteration algorithm (see [80]). At each iteration i, the value of λ is denoted by λ^i and the corresponding random vector $\mathbf{H}_c$ is denoted by $\mathbf{H}_{\lambda^i}$. The value λ^{i+1} is computed as a function of λ^i by

$$\begin{aligned} \lambda^{i+1} &= \lambda^i - \alpha_i [\Gamma''(\lambda^i)]^{-1}\, \boldsymbol{\Gamma}'(\lambda^i) \quad , \quad i \geq 0\,, \\ \lambda^0 &= \mathbf{0}_{n_c}\,, \end{aligned}$$

in which $\boldsymbol{\Gamma}'(\lambda^i) = \mathbf{b} - E\{\mathbf{h}(\mathbf{H}_{\lambda^i})\}$ and $[\Gamma''(\lambda^i)] = [\text{COV}\{\mathbf{h}(\mathbf{H}_{\lambda^i})\}]$ (the covariance matrix). The positive coefficient α_i is a relaxation parameter (less than 1) that is introduced for controlling the convergence of the iteration algorithm. For given $i_2 \geq 2$, for given β_1 and β_2 such that $0 < \beta_1 < \beta_2 \leq 1$, α_i is defined, for $i \leq i_2$, by $\alpha_i = \beta_1 + (\beta_2 - \beta_1)(i-1)/(i_2 - 1)$, and for $i > i_2$, by $\alpha_i = \beta_2$. The convergence of the iteration algorithm is controlled by the error function $i \mapsto \text{err}(i)$ defined by

$$\text{err}(i) = \|\mathbf{b} - E\{\mathbf{h}(\mathbf{H}_{\lambda^i})\}\| / \|\mathbf{b}\|\,. \tag{35}$$

At each iteration i, $E\{\mathbf{h}(\mathbf{H}_{\lambda^i})\}$ and $[\text{COV}\{\mathbf{h}(\mathbf{H}_{\lambda^i})\}]$ are estimated using $n_{\text{ar}} = n_{\text{MC}} \times n_d$ learned realizations of the random vector $\mathbf{H}_{m_{\text{opt}}}(\lambda^i)$, which is obtained by reshaping the n_{MC} learned realizations of the random matrix $[\mathbf{H}_{m_{\text{opt}}}(\lambda^i)]$.

4. Application

4.1. Notation and definition of case studies

In order to deploy and assess the performance of the PLoM approach, two scenarios are introduced as follows.

- In the first scenario, relevant to inverse problem solving, the stochastic boundary conditions on Ω_{mac} and the hyperparameters in the random field model (see Section 2.3) are used as control parameters. Let $\mathbf{W}_{\text{disp}}$ be the random vector corresponding to the discretization of the random field $\{\overline{\mathbf{U}}_{\text{mac}}(\mathbf{x}), \mathbf{x} \in \partial\Omega_{\text{mac}}\}$ (see Section 2.3.2), and let $\mathbf{W}_{\text{hyp}}$ be the random vector associated with the randomization of the mean value, the coefficient of variation, and the correlation length of λ. The control variables are $\mathbf{W}_{\text{hyp}}$ and $\mathbf{W}_{\text{disp}}$.
- In the second scenario, related to propagation, the hyperparameters for the random field model are set to their mean values and are not considered as control variables. The latter only comprise the boundary displacements on Ω_{mac}, gathered in $\mathbf{W}_{\text{disp}}$.

In the context of statistical surrogate modeling, our goal is then to estimate conditional distributions for the mechanistic variables in the homogenized constitutive model (namely, the right Cauchy–Green deformation tensor and the second Piola–Kirchhoff stress tensor), at any quadrature point of interest in the region where concurrent multiscale coupling is considered, given specific values of the control parameters. For the first scenario, this would enable, through the formulation of an ad hoc statistical inverse problem, the identification of the hyperparameters in the material random field model, assuming that (e.g., experimental) data on the above tensors are available. In the second scenario, this would allow for the estimation of the stress and strain variables (defined as conditional means) for given values of the (Dirichlet) boundary conditions on the macroscopic domain — a statistical extension to standard surrogate modeling where the mapping $[C_{\text{mac}}(\mathbf{x}^q,\theta)] \mapsto [S_{\text{mac}}(\mathbf{x}^q,\theta)]$, $1 \le q \le Q_{\text{mac}}$ would be approximated using, e.g., regression.

For the sake of illustration, non-informative prior models are chosen in the first scenario. More specifically, the mean, coefficient of variation, and the correlation lengths are assumed to be statistically independent and uniformly distributed on $[32000, 48000]$, $[0.1, 0.3]$, and $[0.2, 0.4]$, respectively. The truncation order in the Karhunen–Loève expansion is computed for each sample of the correlation length, using the same threshold (1e−2).

The relationships between the notations used for the PLoM algorithm summarized in Section 3 and the notations of mechanical quantities are as follows. We recall that, for the $n_p = 100$ Gauss points referred to by the set of indices $\mathcal{I} = \{i = 1, \dots, n_p\}$, $\{C_{ij}, j = 1,2,3\}$ represents the 3 components of the random Cauchy–Green tensor at index point $i \in \mathcal{I}$, and $\{S_{ij}, j = 1,2,3\}$ represents the 3 components of the random second Piola–Kirchhoff stress tensor. We will also use the notations $\mathbf{C}$ and $\mathbf{S}$ for the reshaping of these two random tensors, which are then random vectors with values in $\mathbb{R}^{3n_p}$. Finally, plots of probability density functions are obtained using (non-parametric) kernel density estimators, while the conditional probability density functions are estimated using Eq. (31).

(i) Quantity of interest. The $\mathbb{R}^n$-valued random variable $\mathbf{Q}$ is defined by $\mathbf{Q} = (\mathbf{S}, \mathbf{C})$ with $n_q = 2 \times (3\, n_p) = 600$.

(ii) Control parameter. Following the previous discussion, let $\mathbf{W}_{\text{disp}}$ be the $\mathbb{R}^{n_{w,\text{disp}}}$-valued random variable for which the $n_{w,\text{disp}} = 48$ components are the discretized displacements on the boundary. Let $\mathbf{W}_{\text{hyp}} = (W_{\text{hyp},1}, W_{\text{hyp},2}, W_{\text{hyp},3})$ be the hyperparameters that control the prior probability model of the random medium (defined in Section 2.3). For the construction of the training and learned datasets, the definition of the $\mathbb{R}^{n_w}$-control parameter $\mathbf{W}$ depends on the scenario.

- Scenario 1: the training and learned datasets are constructed with $\mathbf{W} = (\mathbf{W}_{\text{hyp}}, \mathbf{W}_{\text{disp}})$ and $n_w = 3 + n_{w,\text{disp}} = 3 + 48 = 51$. The conditional statistics are constructed for given $\mathbf{W}_{\text{hyp}} = \mathbf{w}_{\text{hyp},o} \in \mathbb{R}^3$.
- Scenario 2: the training and learned datasets are constructed with $\mathbf{W} = \mathbf{W}_{\text{disp}}$ and $n_w = n_{w,\text{disp}} = 48$. The value of $\mathbf{W}_{\text{hyp}}$ is fixed to the statistical mean value $\underline{\mathbf{w}}_{\text{hyp}} = (\underline{w}_{\text{hyp},1}, \underline{w}_{\text{hyp},2}, \underline{w}_{\text{hyp},3})$ of the prior probability model of $\mathbf{W}_{\text{hyp}}$. The conditional statistics are constructed for given $\mathbf{W}_{\text{disp}} = \mathbf{w}_{\text{disp},o} \in \mathbb{R}^{48}$.

The random variable $\mathbb{U}$ (uncontrolled parameter) corresponds to the stochastic germs in the Karhunen–Loève expansion of the underlying Gaussian random field, which are statistically independent normalized Gaussian random variables.

4.2. Conditional statistics

For the validation of the proposed methodology, the conditional statistics estimated with the learned dataset and those estimated with the validation dataset (considered as a reference) will be compared. Below, we define the considered conditional statistics, including the conditional probability density functions and the conditional mean functions (given the control parameter).

- Scenario 1: for all $i \in \mathcal{I}$, for $j \in \{1,2,3\}$, for $\mathbf{w}_o$ given in $\mathbb{R}^3$, and for all q in $\mathbb{R}$, we consider

$$q \mapsto p_{C_{ij}\,|\,\mathbf{W}_{\text{hyp}}}(q \,|\, \mathbf{w}_o), \quad q \mapsto p_{S_{ij}\,|\,\mathbf{W}_{\text{hyp}}}(q \,|\, \mathbf{w}_o),$$

$$i \mapsto E\{C_{ij} \,|\, \mathbf{W}_{\text{hyp}} = \mathbf{w}_o\} = \int_{\mathbb{R}} q\, p_{C_{ij}\,|\,\mathbf{W}_{\text{hyp}}}(q \,|\, \mathbf{w}_o)\, dq$$

and

$$i \mapsto E\{S_{ij} \,|\, \mathbf{W}_{\text{hyp}} = \mathbf{w}_o\} = \int_{\mathbb{R}} q\, p_{S_{ij}\,|\,\mathbf{W}_{\text{hyp}}}(q \,|\, \mathbf{w}_o)\, dq\,.$$

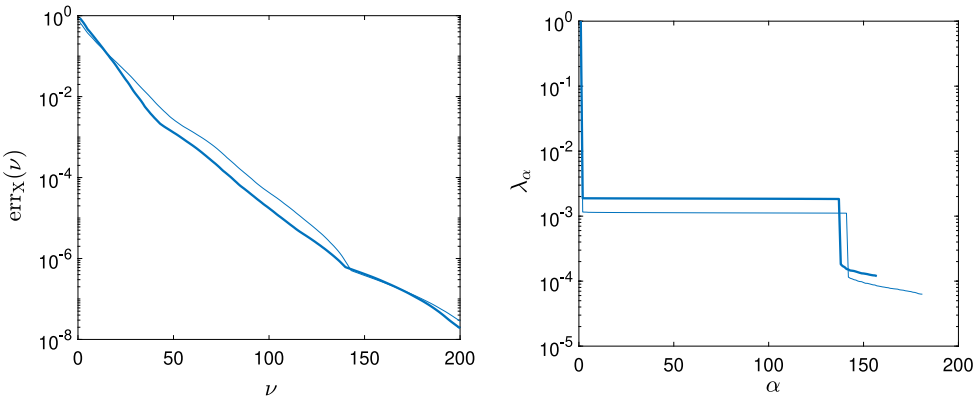
- Scenario 2: for all $i \in \mathcal{I}$, for $j \in \{1,2,3\}$, for $\mathbf{w}_o$ given in $\mathbb{R}^{48}$, and for all q in $\mathbb{R}$, we consider

$$q \mapsto p_{C_{ij}\,|\,\mathbf{W}_{\text{disp}}}(q \,|\, \mathbf{w}_o), \quad q \mapsto p_{S_{ij}\,|\,\mathbf{W}_{\text{disp}}}(q \,|\, \mathbf{w}_o),$$

$$i \mapsto E\{C_{ij} \,|\, \mathbf{W}_{\text{disp}} = \mathbf{w}_o\} = \int_{\mathbb{R}} q\, p_{C_{ij}\,|\,\mathbf{W}_{\text{disp}}}(q \,|\, \mathbf{w}_o)\, dq,$$

and

$$i \mapsto E\{S_{ij} \,|\, \mathbf{W}_{\text{disp}} = \mathbf{w}_o\} = \int_{\mathbb{R}} q\, p_{S_{ij}\,|\,\mathbf{W}_{\text{disp}}}(q \,|\, \mathbf{w}_o)\, dq\,.$$



(a) Graph of function $\nu \mapsto \mathrm{err}_{\mathbf{X}}(\nu)$ (b) Graph of the eigenvalues $\alpha \mapsto \lambda_\alpha$

Fig. 6. Convergence of the PCA reduced representation (a). Eigenvalues of the transition matrix $[\mathbb{P}]$ for the diffusion-maps basis (b). Scenario 1 (thin line), Scenario 2 (thick line).

4.3. Parameter values and convergence analysis of PLoM algorithms for scenarios 1 and 2

In this section, we define the parameter values used by the PLoM algorithms (as summarized in Section 3), and we present the convergence analysis for both scenarios. Notations are those introduced in Section 3. To simplify referencing with respect to each scenario, the first provided value corresponds to Scenario 1, while the second value corresponds to Scenario 2. For instance, "$n_c = 280$ and 272" means that $n_c = 280$ for Scenario 1 and $n_c = 272$ for Scenario 2. When a single value is given, it applies to both scenarios. For example "$n_d = 500$" means that $n_d = 500$ for Scenario 1 and Scenario 2.

(i) Values of the general parameters. The total dimension of $\mathbf{X} = (\mathbf{Q}, \mathbf{W})$ is $n = n_q + n_w = 651$ and 648. The number of points in the training dataset D_d is $n_d = 500$.

(ii) Reduced representation and diffusion-maps basis. Fig. 6(a) displays the graph of the function $\nu \mapsto \mathrm{err}_{\mathbf{X}}(\nu)$ defined by Eq. (26). For Scenario 1, convergence of the representation is achieved at $\nu = 140$, resulting in an error of $\mathrm{err}_{\mathbf{X}}(140) = 2.85 \times 10^{-4}$. In Scenario 2, convergence occurs at $\nu = 136$, with an error of $\mathrm{err}_{\mathbf{X}}(136) = 2.99 \times 10^{-4}$. Regarding the computation of the diffusion-maps basis introduced in Section 3.3, the optimal value of the smoothing parameter $\varepsilon_{\mathrm{DM}}$ is determined as $\varepsilon_{\mathrm{opt}} = 454$ and 278, corresponding to the optimal value $m_{\mathrm{opt}} = 141$ and 137 of parameter m. Fig. 6(b) shows the graph of the function $\alpha \mapsto \lambda_\alpha$ representing the eigenvalues of the transition matrix $[\mathbb{P}]$.

(iii) Generating the learned realization. The learned realizations are generated as explained in Section 3.6, incorporating the constraints outlined in Section 3.7. The free parameter f_0 is chosen as 4, and the integration step Δt of the Störmer–Verlet scheme is 0.21. Each realization $[z_\alpha^\ell]$ represents the ℓth realization of $[\mathcal{Z}(T)]$ for $T = 30 \times \Delta t$ (due to the damping controlled by $f_0 = 4$, T is a time at which the stationary response is reached). We have chosen $n_{\mathrm{MC}} = 100$, resulting in $n_{\mathrm{ar}} = n_{\mathrm{MC}} \times n_d = 50\,000$.

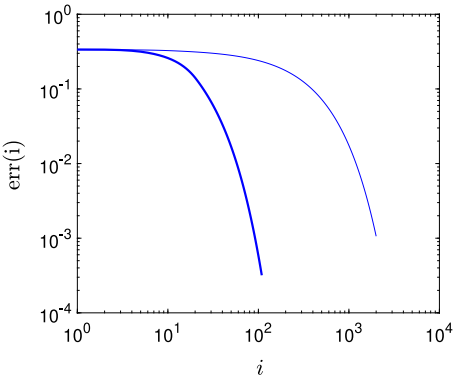
(iv) Constraints on second-order moments of $\mathbf{H}$. For Scenario 1, the number of constraints is $n_c = 280$, and the relaxation parameter α_i is defined by $\beta_1 = 0.01, i_2 = 10$, and $\beta_2 = 0.1$. For Scenario 2, $n_c = 272$, and $\beta_1 = 0.01, i_2 = 20$, and $\beta_2 = 0.5$. The convergence of the iterative algorithm presented in Section 3.7-(ii), to take into account the constraints on second-order moments of $\mathbf{H}$ in PLoM, as a function of iteration number i, is studied in analyzing the graph of the error function $i \mapsto \mathrm{err}(i)$ defined by Eq. (35) (see Fig. 7(a)) and the graph of the function $i \mapsto \|\lambda^i\|$ (see Fig. 7(b)). A very good convergence is observed, with $\mathrm{err}(2000) = 10^{-3}$ (Scenario 1) and $\mathrm{err}(109) = 3.2 \times 10^{-4}$ (Scenario 2). For both scenarios and for all k in $\{1, \ldots, \nu\}$, at convergence, it holds that $|E\{H_{c,k}\}| \leq 10^{-7}$ and $0.999 \leq E\{H_{c,k}^2\} \leq 1$.

(v) Concentration of the learned probability measure. As explained in Section 3.5, the quality of the PLoM algorithm is assessed by examining the value of $d_{n_d}^2(m_{\mathrm{opt}})$, defined by Eq. (30). At convergence, we obtain $d_{n_d}^2(m_{\mathrm{opt}}) = 0.17$ and 0.16, indicating excellent quality of PLoM to preserve the concentration of the learned probability measure.

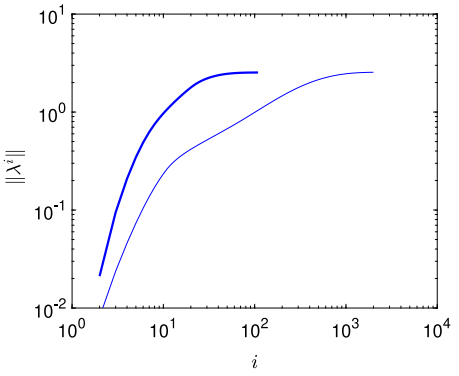
(vi) Illustration of the learned pdf of components of $\mathbf{H}$ *and of the clouds of the learned points*. In $\mathbf{H}_c$, the subscript c is removed to simplify the writing. Fig. 8 (Scenario 1) and Fig. 9 (Scenario 2) depict the probability density functions of components H_1, H_{30}, and H_{70} of $\mathbf{H}$, estimated using the training dataset and the learned dataset. These figures exhibit a good coherence. It should be noted that the convergence of these estimates is not the same, as there are $n_d = 500$ points in the training dataset and $n_{\mathrm{ar}} = 50\,000$ points in the learned dataset. Fig. 10(a) (Scenario 1) and Fig. 10(b) (Scenario 2) display the clouds of the learned points and the training points of $\mathbf{H}$, associated with components H_1, H_{30}, and H_{70}. These results illustrate the preservation of the concentration of the probability measure.

4.4. Validation analysis

In this section, we present a validation of the proposed methodology. This methodology is based on the construction of conditional statistics (statistical surrogate model) defined in Section 4.2, which are estimated with the n_{ar} points of the learned dataset, generated

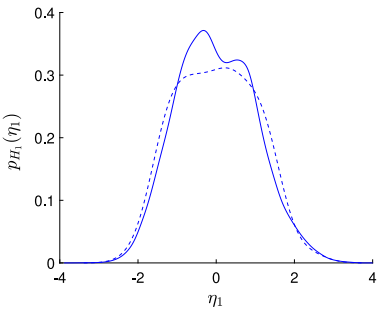


(a) Graph of function $i \mapsto \mathrm{err}(i)$

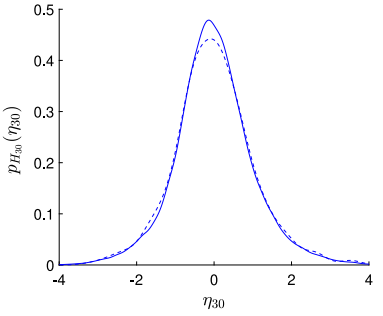


(b) Graph of function $i \mapsto \|\boldsymbol{\lambda}^i\|$

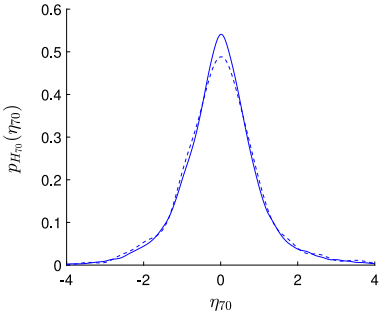
Fig. 7. Convergence analysis of the iterative algorithm to take into account the constraints on second-order moments of **H** in PLoM as a function of iteration number i, presented in log–log scales. Scenario 1 (thin line), Scenario 2 (thick line).



(a) PDF $\eta_1 \mapsto p_{H_1}(\eta_1)$

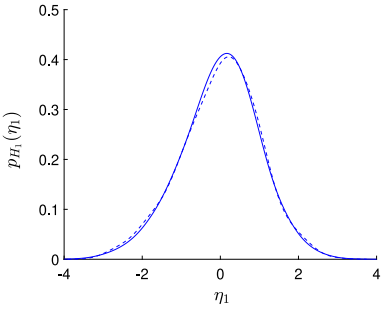


(b) PDF $\eta_{30} \mapsto p_{H_{30}}(\eta_{30})$

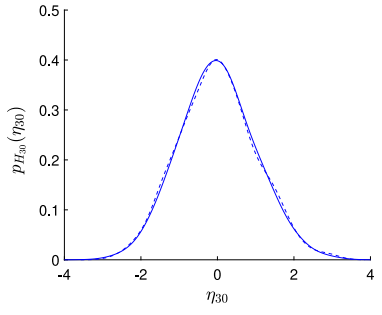


(c) PDF $\eta_{70} \mapsto p_{H_{70}}(\eta_{70})$

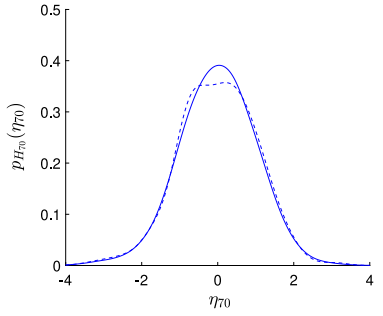
Fig. 8. Scenario 1: probability density functions of components H_1, H_{30}, and H_{70} of **H**, estimated with the training dataset (dashed line) and with the learned dataset (solid line).



(a) PDF $\eta_1 \mapsto p_{H_1}(\eta_1)$

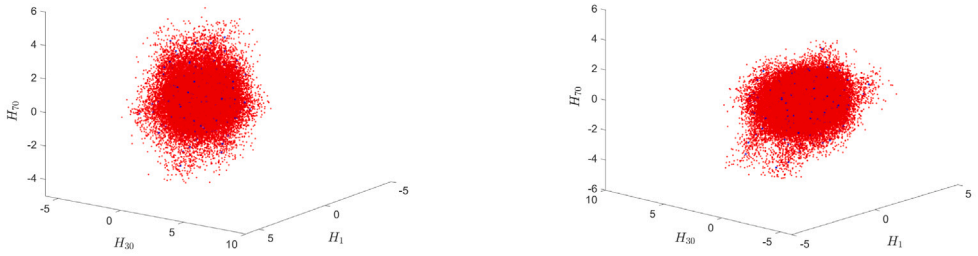


(b) PDF $\eta_{30} \mapsto p_{H_{30}}(\eta_{30})$



(c) PDF $\eta_{70} \mapsto p_{H_{70}}(\eta_{70})$

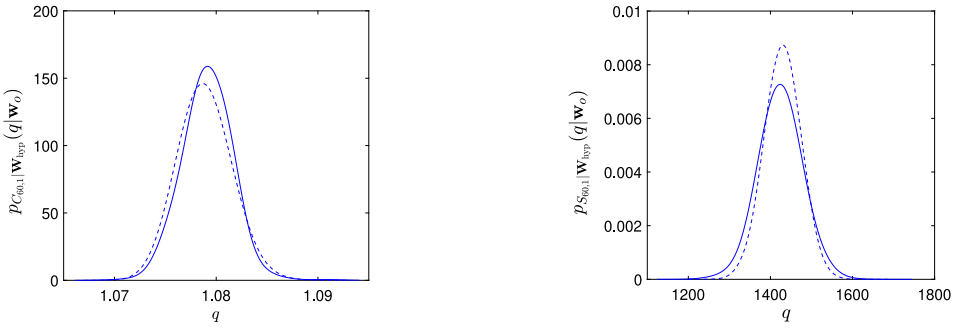
Fig. 9. Scenario 2: probability density functions of components H_1, H_{30}, and H_{70} of **H**, estimated with the training dataset (dashed line) and with the learned dataset (solid line).



(a) Scenario 1: cloud of points for training (blue star) and learning (red hexagram)

(b) Scenario 2: cloud of points for training (blue star) and learning (red hexagram)

Fig. 10. Clouds of the learned points and the training points of $\mathbf{H}$, associated with components H_1, H_{30}, and H_{70}.



(a) Graph of function $q \mapsto p_{C_{60,1}|\mathbf{W}_{\text{hyp}}}(q|\mathbf{w}_o)$

(b) Graph of function $q \mapsto p_{S_{60,1}|\mathbf{W}_{\text{hyp}}}(q|\mathbf{w}_o)$

Fig. 11. Validation of Scenario 1: Graph of the conditional pdf given $\mathbf{W}_{\text{hyp}} = \mathbf{w}_0$ of component 1 of the random tensors $\mathbf{C}$ and $\mathbf{S}$ at point 60. Learned dataset (solid line), validation dataset (dashed line).

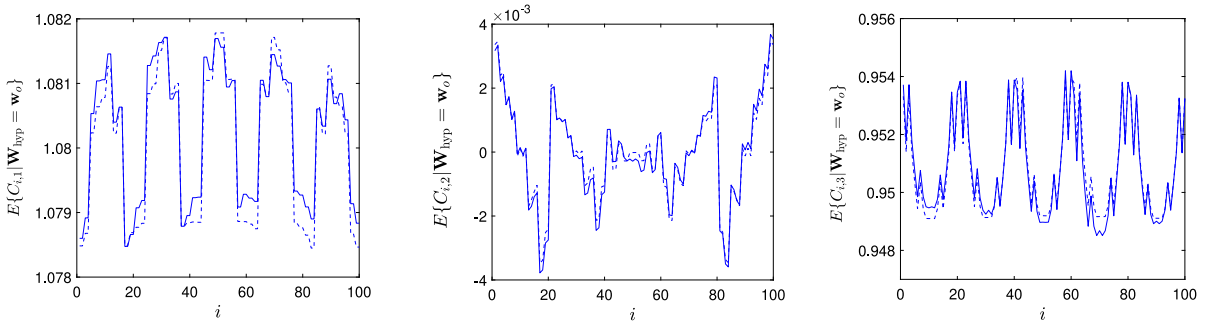
with the PLoM algorithm. The estimates of these conditional statistics are converged because n_{ar} is large. For validation purposes, these conditional statistics must be compared with a reference. This reference can only be obtained by constructing a validation dataset with n_v points generated with the nonlinear computational model used to construct the n_d points of the training dataset. Ideally, $n_v \simeq n_{\text{ar}}$ and the construction of a validation dataset can be achieved for both scenarios. Here, due to limitations in computational resources, we only consider the construction of a new validation dataset for Scenario 2. This validation dataset $\mathcal{D}_v$ comprises $n_v = 20\,000$ independent realizations $(\mathbf{q}_v^\ell, \mathbf{w}_v^\ell)$ in $\mathbb{R}^n = \mathbb{R}^{n_q} \times \mathbb{R}^{n_w}$ with $n_q = 600$ and $n_w = 48$, for $\ell \in \{1, \ldots, n_v\}$ of the random variable $(\mathbf{Q}, \mathbf{W})$ (note that $n_v < n_{\text{ar}}$ in this case). The constitution of this dataset requires solving $2\,040\,000$ nonlinear finite-element computations overall. The following settings are then considered.

- For Scenario 1: as explained in Section 4.1, we have $\mathbf{W} = (\mathbf{W}_{\text{hyp}}, \mathbf{W}_{\text{disp}})$ and $n_w = 3 + 48 = 51$. To construct the reference conditional statistics with the validation dataset $\mathcal{D}_v$, we can therefore consider a single value $n_o = 1$, $\mathbf{w}_o = \underline{\mathbf{w}}_{\text{hyp}} \in \mathbb{R}^3$ of the random control parameter $\mathbf{W}_{\text{hyp}}$. The realizations $\{\mathbf{w}_v^\ell\}_{\ell \geq 1}$ are not useful, and only the realizations $\{\mathbf{q}_v^\ell\}_{\ell \geq 1}$ are used.
- For Scenario 2: for the validation of the conditional statistics, the control variable is $\mathbf{W} = \mathbf{W}_{\text{disp}}$ with $n_w = n_{w,\text{disp}} = 48$. To construct the reference conditional statistics with the validation dataset $\mathcal{D}_v$, we introduce n_o values, $\{\mathbf{w}_{o,k} \in \mathbb{R}^{n_w}, k = 1, \ldots, n_o\}$, of the random control parameter $\mathbf{W}$ with values in $\mathbb{R}^{n_w}$. We have randomly drawn, with a uniform distribution, the n_o vectors $\{\mathbf{w}_{o,k}, k = 1, \ldots, n_o\}$ from the set $\{\mathbf{w}_v^\ell, \ell = 1, \ldots, n_v\}$. Due to space limitations, we consider $n_o = 2$: these two vectors correspond to the realizations #1882 and #19 502 in the set $\{\mathbf{w}_v^\ell, \ell = 1, \ldots, 20\,000\}$.

The validation results are presented below for each scenario.

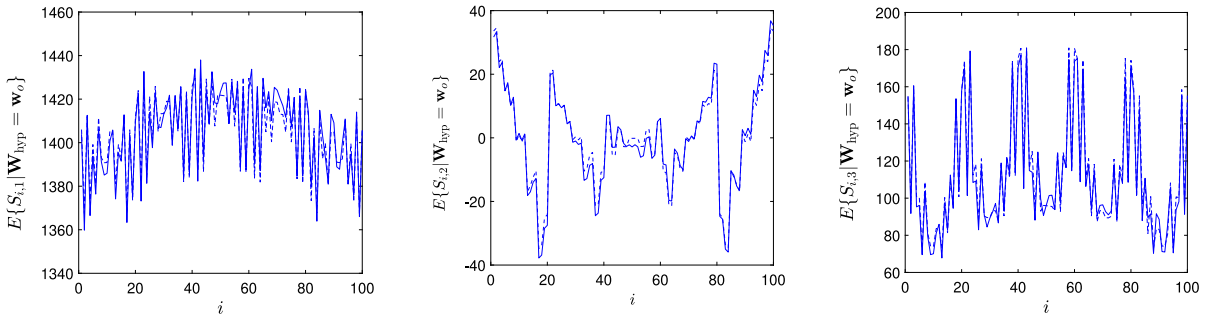
(i) Validation for Scenario 1. Concerning the conditional probability density functions, we select the first components at point #60, corresponding to the random variables $C_{60,1}$ and $S_{60,1}$ relative to the random tensors $\mathbf{C}$ and $\mathbf{S}$, as quantities of interest. This point is relatively central in the macroscopic domain, and the first components at this location present significant fluctuations — hence making this choice a reasonable one from a validation standpoint. Fig. 11 displays the graphs of the conditional probability density functions $q \mapsto p_{C_{60,1}|\mathbf{W}_{\text{hyp}}}(q|\mathbf{w}_o)$ of random variable $C_{60,1}$ and $q \mapsto p_{S_{60,1}|\mathbf{W}_{\text{hyp}}}(q|\mathbf{w}_o)$ of random variable $S_{60,1}$ given $\mathbf{W}_{\text{hyp}} = \mathbf{w}_0$, estimated with the learned dataset and the validation dataset. It can be seen that the width of the supports, which control the variances, is well predicted.

For the three components indexed by $j \in \{1, 2, 3\}$, Fig. 12 displays the graphs of the conditional mathematical expectations $i \mapsto E\{C_{i,j}|\mathbf{W}_{\text{hyp}} = \mathbf{w}_o\}$ for the family $\{C_{i,j}, i \in \mathcal{I}\}$ of random variables, while Fig. 13 displays the graphs of $i \mapsto E\{S_{i,j}|\mathbf{W}_{\text{hyp}} = \mathbf{w}_o\}$



(a) Graph of function $i \mapsto E\{C_{i,1}|\mathbf{W}_{\text{hyp}} = \mathbf{w}_o\}$ (b) Graph of function $i \mapsto E\{C_{i,2}|\mathbf{W}_{\text{hyp}} = \mathbf{w}_o\}$ (c) Graph of function $i \mapsto E\{C_{i,3}|\mathbf{W}_{\text{hyp}} = \mathbf{w}_o\}$

Fig. 12. Validation of Scenario 1: Graph of the conditional mathematical expectation as function of points i given $\mathbf{W}_{\text{hyp}} = \mathbf{w}_0$ for the three components of the random tensor $\mathbf{C}$. Learned dataset (solid line), validation dataset (dashed line).



(a) Graph of function $i \mapsto E\{S_{i,1}|\mathbf{W}_{\text{hyp}} = \mathbf{w}_o\}$ (b) Graph of function $i \mapsto E\{S_{i,2}|\mathbf{W}_{\text{hyp}} = \mathbf{w}_o\}$ (c) Graph of function $i \mapsto E\{S_{i,3}|\mathbf{W}_{\text{hyp}} = \mathbf{w}_o\}$

Fig. 13. Validation of Scenario 1: Graph of the conditional mathematical expectation as function of points i given $\mathbf{W}_{\text{hyp}} = \mathbf{w}_0$ for the three components of the random tensor $\mathbf{S}$. Learned dataset (solid line), validation dataset (dashed line).

for the family $\{S_{i,j}, i \in \mathcal{I}\}$. While very large variations are observed over the quadrature points, the accuracy in the predictions of the conditional expectations remains remarkably good given the small number of training data. It should be noted that the graphs in Figs. 12(b) and 13(b) differ only from a constant.

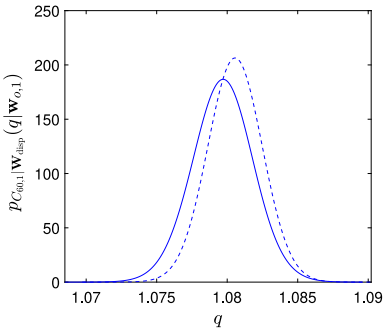
(ii) Validation for Scenario 2. The conditional probability density functions are estimated for the first components of the tensorial quantities of interest at point #60, as for Scenario 1. For the two values of the control parameter, $k = 1$ and $k = 2$, Fig. 14 displays the graphs of the conditional probability density functions $q \mapsto p_{C_{60,1}|\mathbf{W}_{\text{disp}}}(q|\mathbf{w}_{o,k})$ for random variable $C_{60,1}$ given $\mathbf{W}_{\text{disp}} = \mathbf{w}_{0,k}$, estimated with the learned dataset and the validation dataset. Similarly, Fig. 15 displays the graphs of $q \mapsto p_{S_{60,1}|\mathbf{W}_{\text{disp}}}(q|\mathbf{w}_{o,k})$ for random variable $S_{60,1}$ given $\mathbf{W}_{\text{disp}} = \mathbf{w}_{0,k}$. Again, it is seen that the supports of the (marginal) distributions are well predicted, and the validation results are reasonably good.

For $j \in \{1, 2, 3\}$ and the two values of the control parameter ($k = 1$ and $k = 2$), Figs. 16 and 17 display the graphs of the conditional mathematical expectations $i \mapsto E\{C_{i,j}|\mathbf{W}_{\text{disp}} = \mathbf{w}_{o,k}\}$ for the family $\{C_{i,j}, i \in \mathcal{I}\}$ of random variables, while Figs. 18 and 19 display the graphs of $i \mapsto E\{S_{i,j}|\mathbf{W}_{\text{hyp}} = \mathbf{w}_{o,k}\}$ for the family $\{S_{i,j}, i \in \mathcal{I}\}$. Good accuracy is observed over all quadrature points, which demonstrates the capability of the framework to capture the non-smooth large variations generated by the localization in the nonlinear stochastic boundary value problems.

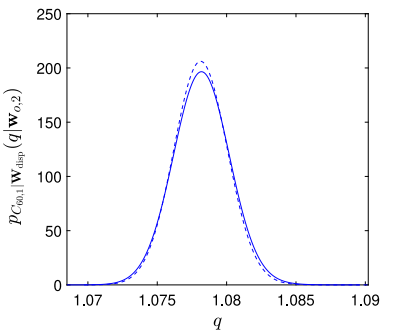
4.4.1. Remarks on CPU time

For the probabilistic learning algorithm, the total CPU time is due to the construction of the learned dataset, the numerical cost of conditional statistics being completely negligible. For the construction of learned datasets, the CPU time is directly proportional to the number of iterations required to satisfy the normalization constraints. For the first scenario, 2000 are used and the total CPU time is 4 hr and 26 min. For the second scenario, only 109 iterations are necessary to reach excellent convergence and the total CPU time is 19 min. It should be noted that for scenario 1, the convergence tolerance could have been reduced without significantly penalizing the quality of the results, and this would have significantly reduced the number of iterations and therefore CPU time.

Regarding the generation of the training dataset using nonlinear finite element simulations and the FE^2 approach, the average CPU time to complete one simulation is 449.80 s, and one multiscale simulation at any Gauss point (and final increment) takes from 0.5 to 5.401 s (depending on both the sample of the local material properties and the applied boundary conditions). Assuming that conditional statistics would be computed using 50,000 samples, the CPU time associated with a brute force approach (with

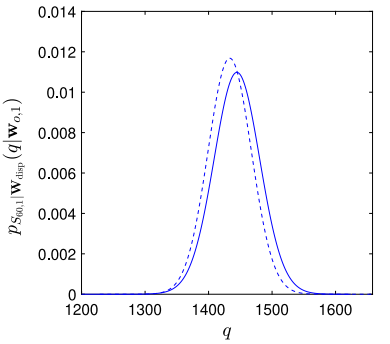


(a) Graph of function $q \mapsto p_{C_{60,1}|\mathbf{W}_{\text{disp}}}(q|\mathbf{w}_{o,1})$

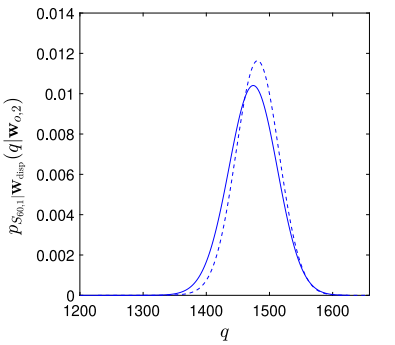


(b) Graph of function $q \mapsto p_{C_{60,1}|\mathbf{W}_{\text{disp}}}(q|\mathbf{w}_{o,2})$

Fig. 14. Validation of Scenario 2: Graph of the conditional pdf given $\mathbf{W}_{\text{disp}} = \mathbf{w}_{0,1}$ or $\mathbf{w}_{0,2}$ of component 1 of the random tensor $\mathbf{C}$ at point 60. Learned dataset (solid line), validation dataset (dashed line).

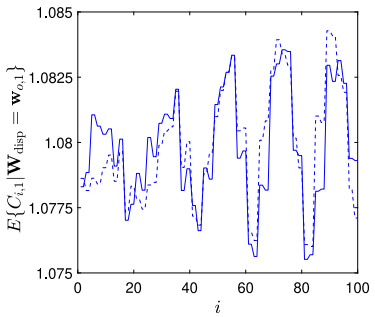


(a) Graph of function $q \mapsto p_{S_{60,1}|\mathbf{W}_{\text{disp}}}(q|\mathbf{w}_{o,1})$

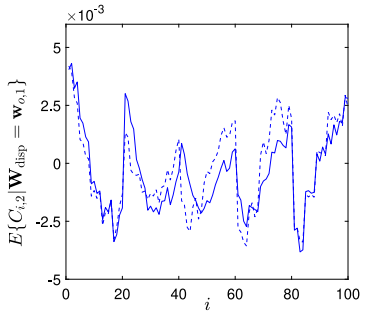


(b) Graph of function $q \mapsto p_{S_{60,1}|\mathbf{W}_{\text{disp}}}(q|\mathbf{w}_{o,2})$

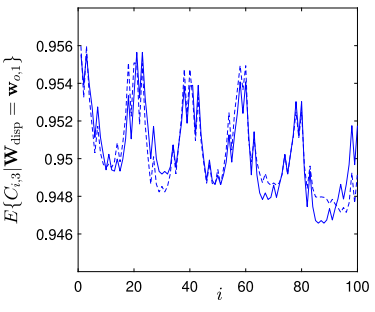
Fig. 15. Validation of Scenario 2: Graph of the conditional pdf given $\mathbf{W}_{\text{disp}} = \mathbf{w}_{0,1}$ or $\mathbf{w}_{0,2}$ of component 1 of the random tensor $\mathbf{S}$ at point 60. Learned dataset (solid line), validation dataset (dashed line).



(a) Graph of function $i \mapsto E\{C_{i,1}|\mathbf{W}_{\text{disp}} = \mathbf{w}_{o,1}\}$



(b) Graph of function $i \mapsto E\{C_{i,2}|\mathbf{W}_{\text{disp}} = \mathbf{w}_{o,1}\}$



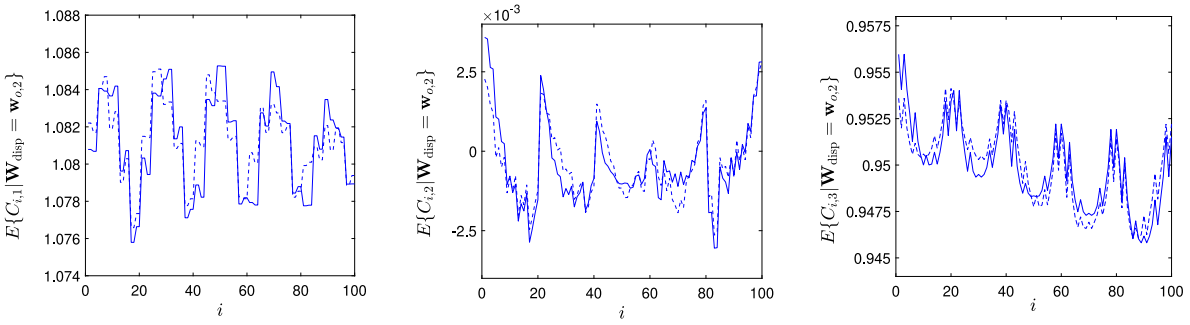
(c) Graph of function $i \mapsto E\{C_{i,3}|\mathbf{W}_{\text{disp}} = \mathbf{w}_{o,1}\}$

Fig. 16. Validation of Scenario 2: Graph of the conditional mathematical expectation as function of points i given $\mathbf{W}_{\text{disp}} = \mathbf{w}_{0,1}$ for the three components of the random tensor $\mathbf{C}$. Learned dataset (solid line), validation dataset (dashed line).

sequential execution) would then be $\approx 50,000 \times 450 = 6250$ h. While this time can be reduced using distributed computing, it remains much larger than the cost induced by the learning method.

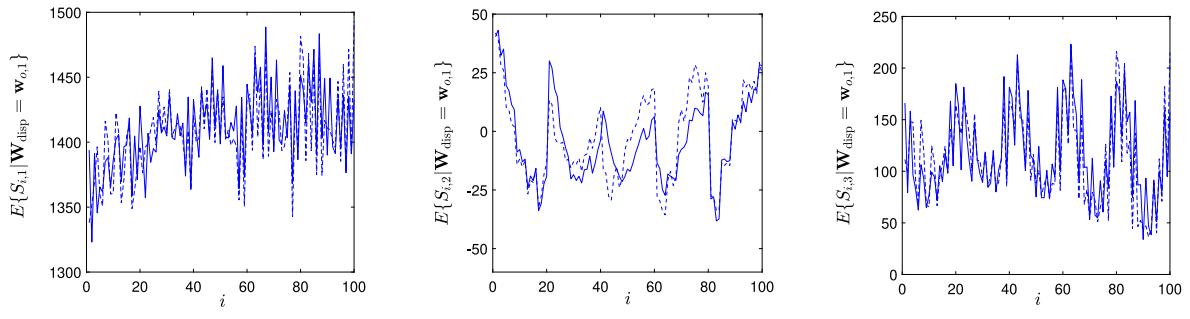
5. Conclusion

In this work, we have introduced a statistical surrogate model for concurrent multiscale simulations involving nonlinear materials with non-separated scales. The methodology combines probabilistic learning on manifolds, a generative model that allows for



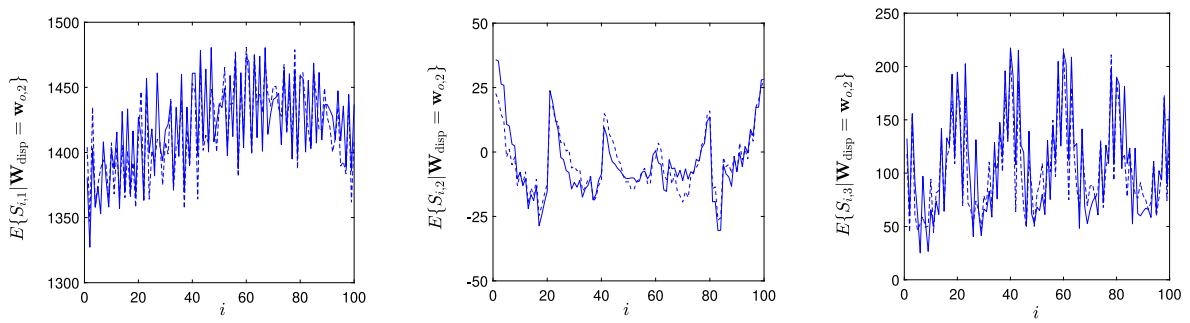
(a) Graph of function $i \mapsto E\{C_{i,1}|\mathbf{W}_{\text{disp}} = \mathbf{w}_{o,2}\}$ (b) Graph of function $i \mapsto E\{C_{i,2}|\mathbf{W}_{\text{disp}} = \mathbf{w}_{o,2}\}$ (c) Graph of function $i \mapsto E\{C_{i,3}|\mathbf{W}_{\text{disp}} = \mathbf{w}_{o,2}\}$

Fig. 17. Validation of Scenario 2: Graph of the conditional mathematical expectation as function of points i given $\mathbf{W}_{\text{disp}} = \mathbf{w}_{0,2}$ for the three components of the random tensor $\mathbf{C}$. Learned dataset (solid line), validation dataset (dashed line).



(a) Graph of function $i \mapsto E\{S_{i,1}|\mathbf{W}_{\text{disp}} = \mathbf{w}_{o,1}\}$ (b) Graph of function $i \mapsto E\{S_{i,2}|\mathbf{W}_{\text{disp}} = \mathbf{w}_{o,1}\}$ (c) Graph of function $i \mapsto E\{S_{i,3}|\mathbf{W}_{\text{disp}} = \mathbf{w}_{o,1}\}$

Fig. 18. Validation of Scenario 2: Graph of the conditional mathematical expectation as function of points i given $\mathbf{W}_{\text{disp}} = \mathbf{w}_{0,1}$ for the three components of the random tensor $\mathbf{S}$. Learned dataset (solid line), validation dataset (dashed line).



(a) Graph of function $i \mapsto E\{S_{i,1}|\mathbf{W}_{\text{disp}} = \mathbf{w}_{o,2}\}$ (b) Graph of function $i \mapsto E\{S_{i,2}|\mathbf{W}_{\text{disp}} = \mathbf{w}_{o,2}\}$ (c) Graph of function $i \mapsto E\{S_{i,3}|\mathbf{W}_{\text{disp}} = \mathbf{w}_{o,2}\}$

Fig. 19. Validation of Scenario 2: Graph of the conditional mathematical expectation as function of points i given $\mathbf{W}_{\text{disp}} = \mathbf{w}_{0,2}$ for the three components of the random tensor $\mathbf{S}$. Learned dataset (solid line), validation dataset (dashed line).

measure concentration and support information to be accurately captured, with the use of conditional statistics to approximate the mapping between apparent strain and stress variables — namely, the right Cauchy–Green tensor and the second Piola–Kirchhoff stress tensor — at a finite set of points (defining a subregion of interest where concurrent coupling must be deployed). As opposed to standard techniques relying on, e.g., polynomial or neural network surrogates, the proposed approach (1) can readily accommodate the (aleatoric) randomness raised by the multiscale setting, (2) enables the seamless integration of nonlocal interactions through the consideration of joint distributions, and (3) can perform efficiently in the small data regime. Two applications, relevant to inverse problem solving and forward propagation, were presented in the context of nonlinear elasticity. In the first case, the hyperparameters for the prior model defining the random media and boundary conditions at mesoscale are considered as control parameters. This setting can be used, for instance, to identify the hyperparameters when experimental observations are available. In the second application, only mesoscopic boundary displacements are used as control variables. In this case, the prior model at fine scale is

fixed, and the effect of material randomness can be quantified. In both cases, large spatial variations, large statistical fluctuations, and strong non-Gaussianity are observed. It was shown that despite these challenges, the framework remains capable of delivering reasonably accurate estimations, even with a fairly limited amount of training data. The information contained in the latter is a critical aspect in the methodology: this information must be rich enough to learn the probability measure and discover the geometry of the manifold defining its support.

CRediT authorship contribution statement

Peiyi Chen: Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Johann Guilleminot:** Writing – review & editing, Writing – original draft, Validation, Software, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization. **Christian Soize:** Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization.

Data availability

Data will be made available on request.

Acknowledgments

The work of the P. C. and J. G. was supported by the National Science Foundation, United States, under awards DGE-2022040 and CMMI-1942928.

Appendix. Formulas for conditional statistics

In this Appendix, we use the following notation. The real-valued random variable Q denotes any component of the $\mathbb{R}^{n_q}$-valued random variable $\mathbf{Q}$, and the real variable q represents the corresponding component of the vector $\mathbf{q}$ in $\mathbb{R}^{n_q}$. Let $\widetilde{Q}$ and $\widetilde{\mathbf{W}} = (\widetilde{W}_1, \ldots, \widetilde{W}_{n_w})$ be the normalized random variables defined by

$$\widetilde{Q} = (Q - \underline{q})/\sigma_Q \quad , \quad \widetilde{W}_k = (W_k - \underline{w}_k)/\sigma_{W_k} \quad , \quad k = 1, \ldots, n_w \, , \tag{A.1}$$

where $\underline{q}$, $\underline{w}_k$, and σ_Q, σ_{W_k} are the mean values and standard deviations of the random variables Q and W_k. These values are estimated using empirical statistical estimators based on the learned realizations $\{(\mathbf{q}_{\rm ar}^\ell, \mathbf{w}_{\rm ar}^\ell), \ell = 1, \ldots, n_{\rm ar}\}$. For any value $\mathbf{w}_0 = (w_{0,1}, \ldots, w_{0,n_w})$ of the control parameter given in $C_w \subset \mathbb{R}^{n_w}$, we define the vector $\widetilde{\mathbf{w}}_0$ in $\mathbb{R}^{n_w}$ such that

$$\widetilde{w}_{0,k} = (w_{0,k} - \underline{w}_k)/\sigma_{W_k} \quad , \quad k = 1, \ldots, n_w \, . \tag{A.2}$$

The Gaussian KDE estimation of the joint probability distribution of $\widetilde{Q}$ and $\widetilde{\mathbf{W}}$, with respect to $d\widetilde{q}\, d\widetilde{\mathbf{w}}$, is written as

$$p_{\widetilde{Q},\widetilde{\mathbf{W}}}(\widetilde{q}, \widetilde{\mathbf{w}}) = \frac{1}{n_{\rm ar}} \sum_{\ell=1}^{n_{\rm ar}} \frac{1}{\sqrt{2\pi}s} \exp(-\frac{1}{2s^2}(\widetilde{q} - \widetilde{q}_{\rm ar}^\ell)^2) \frac{1}{(\sqrt{2\pi}s)^{n_w}} \exp(-\frac{1}{2s^2}\|\widetilde{\mathbf{w}} - \widetilde{\mathbf{w}}_{\rm ar}^\ell\|^2) \, . \tag{A.3}$$

In Eq. (A.3), s is the Silverman bandwidth defined by

$$s = \left\{ \frac{4}{n_{\rm ar}(2+n)} \right\}^{1/(n+4)} \quad , \quad n = 1 + n_w \, , \tag{A.4}$$

and, for $\ell = 1, \ldots, n_{\rm ar}$, $\widetilde{q}_{\rm ar}^\ell$ and $\widetilde{w}_{{\rm ar},k}^\ell$ are defined by

$$\widetilde{q}_{\rm ar}^\ell = (q_{\rm ar}^\ell - \underline{q})/\sigma_Q \quad , \quad \widetilde{w}_{{\rm ar},k}^\ell = (w_{{\rm ar},k}^\ell - \underline{w}_k)/\sigma_{W_k} \quad , \quad k = 1, \ldots, n_w \, . \tag{A.5}$$

From Eq. (A.3), the following formulas for conditional statistics are derived.
(i) The conditional mathematical expectation $E\{Q \,|\, \mathbf{W} = \mathbf{w}_o\}$ of Q given $\mathbf{W} = \mathbf{w}_0$ in C_w is given by

$$E\{Q \,|\, \mathbf{W} = \mathbf{w}_0\} = \underline{q} + \sigma_Q \frac{\sum_{\ell=1}^{n_{\rm ar}} \widetilde{q}_{\rm ar}^\ell \times \exp(-\frac{1}{2s^2}\|\widetilde{\mathbf{w}}_0 - \widetilde{\mathbf{w}}_{\rm ar}^\ell\|^2)}{\sum_{\ell=1}^{n_{\rm ar}} \exp(-\frac{1}{2s^2}\|\widetilde{\mathbf{w}}_0 - \widetilde{\mathbf{w}}_{\rm ar}^\ell\|^2)} \, . \tag{A.6}$$

(ii) The conditional probability density function $p_{Q|\mathbf{W}}(q \,|\, \mathbf{w}_0)$ with respect to dq of Q given $\mathbf{W} = \mathbf{w}_0$ in C_w is defined as

$$p_{Q|\mathbf{W}}(q \,|\, \mathbf{w}_0) = \frac{1}{\sqrt{2\pi}\, s\, \sigma_Q} \frac{\sum_{\ell=1}^{n_{\rm ar}} \exp(-\frac{1}{2s^2}(\widetilde{q} - \widetilde{q}_{\rm ar}^\ell)^2) \times \exp(-\frac{1}{2s^2}\|\widetilde{\mathbf{w}}_0 - \widetilde{\mathbf{w}}_{\rm ar}^\ell\|^2)}{\sum_{\ell=1}^{n_{\rm ar}} \exp(-\frac{1}{2s^2}\|\widetilde{\mathbf{w}}_0 - \widetilde{\mathbf{w}}_{\rm ar}^\ell\|^2)} \quad , \quad \widetilde{q} = (q - \underline{q})/\sigma_Q \, . \tag{A.7}$$

(iii) The conditional cumulative distribution function $F_{Q|\mathbf{W}}(q^*|\mathbf{w}_0) = \text{Proba}\{Q \le q^* \,|\, \mathbf{W} = \mathbf{w}_0\}$ of Q given $\mathbf{W} = \mathbf{w}_0$ in C_w is estimated using

$$F_{Q|\mathbf{W}}(q^*|\mathbf{w}_0) = \frac{\sum_{\ell=1}^{n_{\rm ar}} \widetilde{F}^\ell(\widetilde{q}^*) \times \exp(-\frac{1}{2s^2}\|\widetilde{\mathbf{w}}_0 - \widetilde{\mathbf{w}}^\ell\|^2)}{\sum_{\ell=1}^{n_{\rm ar}} \exp(-\frac{1}{2s^2}\|\widetilde{\mathbf{w}}_0 - \widetilde{\mathbf{w}}^\ell\|^2)} \quad , \quad \widetilde{q}^* = (q^* - \underline{q})/\sigma_Q \, , \tag{A.8}$$

with

$$\widetilde{F}^{\ell}(\widetilde{q}^{*}) = \frac{1}{2} + \frac{1}{2}\mathrm{erf}\left(\frac{1}{\sqrt{2}\,s}(\widetilde{q}^{*} - \widetilde{q}^{\ell}_{\mathrm{ar}})\right) \quad , \quad \mathrm{erf}(y) = \frac{2}{\sqrt{\pi}}\int_0^y e^{-t^2}\,dt\,. \tag{A.9}$$

References

[1] S. Ghosh, K. Lee, S. Moorthy, Two scale analysis of heterogeneous elastic-plastic materials with asymptotic homogenization and Voronoi cell finite element model, Comput. Methods Appl. Mech. Engrg. 132 (1–2) (1996) 63–116.

[2] R.J. Smit, W.M. Brekelmans, H.E. Meijer, Prediction of the mechanical behavior of nonlinear heterogeneous systems by multi-level finite element modeling, Comput. Methods Appl. Mech. Engrg. 155 (1–2) (1998) 181–192.

[3] F. Feyel, Multiscale FE2 elastoviscoplastic analysis of composite structures, Comput. Mater. Sci. 16 (1–4) (1999) 344–354.

[4] F. Feyel, J.-L. Chaboche, FE2 multiscale approach for modelling the elastoviscoplastic behaviour of long fibre SiC/Ti composite materials, Comput. Methods Appl. Mech. Engrg. 183 (3–4) (2000) 309–330.

[5] K. Terada, N. Kikuchi, A class of general algorithms for multi-scale analyses of heterogeneous media, Comput. Methods Appl. Mech. Engrg. 190 (40–41) (2001) 5427–5464.

[6] V. Kouznetsova, M.G. Geers, W.M. Brekelmans, Multi-scale constitutive modelling of heterogeneous materials with a gradient-enhanced computational homogenization scheme, Internat. J. Numer. Methods Engrg. 54 (8) (2002) 1235–1260.

[7] M. Geers, J. Yvonnet, Multiscale modeling of microstructure–property relations, MRS Bull. 41 (8) (2016) 610–616.

[8] K. Raju, T.-E. Tay, V.B.C. Tan, A review of the FE 2 method for composites, Multiscale Multidiscip. Model. Exp. Design 4 (2021) 1–24.

[9] J. Yvonnet, Q.-C. He, The reduced model multiscale method (R3M) for the non-linear homogenization of hyperelastic media at finite strains, J. Comput. Phys. 223 (1) (2007) 341–368.

[10] E. Monteiro, J. Yvonnet, Q.-C. He, Computational homogenization for nonlinear conduction in heterogeneous materials using model reduction, Comput. Mater. Sci. 42 (4) (2008) 704–712.

[11] J. Yvonnet, D. Gonzalez, Q.-C. He, Numerically explicit potentials for the homogenization of nonlinear elastic heterogeneous materials, Comput. Methods Appl. Mech. Engrg. 198 (33–36) (2009) 2723–2737.

[12] I. Temizer, P. Wriggers, An adaptive method for homogenization in orthotropic nonlinear elasticity, Comput. Methods Appl. Mech. Engrg. 196 (35–36) (2007) 3409–3423.

[13] I. Temizer, T. Zohdi, A numerical method for homogenization in non-linear elasticity, Comput. Mech. 40 (2007) 281–298.

[14] F. Fritzen, T. Böhlke, Reduced basis homogenization of viscoelastic composites, Compos. Sci. Technol. 76 (2013) 84–91.

[15] S. Bhattacharjee, K. Matouš, A nonlinear manifold-based reduced order model for multiscale analysis of heterogeneous hyperelastic materials, J. Comput. Phys. 313 (2016) 635–653.

[16] J. Yvonnet, E. Monteiro, Q.-C. He, Computational homogenization method and reduced database model for hyperelastic heterogeneous structures, Int. J. Multiscale Comput. Eng. 11 (3) (2013).

[17] F. Fritzen, O. Kunc, Two-stage data-driven homogenization for nonlinear solids using a reduced order model, Eur. J. Mech. A Solids 69 (2018) 201–220.

[18] Z. Liu, M. Bessa, W.K. Liu, Self-consistent clustering analysis: an efficient multi-scale scheme for inelastic heterogeneous materials, Comput. Methods Appl. Mech. Engrg. 306 (2016) 319–341.

[19] X. Han, J. Gao, M. Fleming, C. Xu, W. Xie, S. Meng, W.K. Liu, Efficient multiscale modeling for woven composites based on self-consistent clustering analysis, Comput. Methods Appl. Mech. Engrg. 364 (2020) 112929.

[20] Y. Feng, H. Yong, Y. Zhou, A concurrent multiscale framework based on self-consistent clustering analysis for cylinder structure under uniaxial loading condition, Compos. Struct. 266 (2021) 113827.

[21] Y. Efendiev, T.Y. Hou, V. Ginting, Multiscale finite element methods for nonlinear problems and their applications, Commun. Math. Sci. 2 (4) (2004) 553–589.

[22] Y. Efendiev, T.Y. Hou, Multiscale Finite Element Methods: Theory and Applications, vol. 4, Springer Science & Business Media, 2009.

[23] E. Weinan, B. Engquist, The heterognous multiscale methods, Commun. Math. Sci. 1 (1) (2003) 87–132.

[24] E. Weinan, B. Engquist, X. Li, W. Ren, E. Vanden-Eijnden, Heterogeneous multiscale methods: a review, Commun. Comput. Phys. 2 (3) (2007) 367–450.

[25] A. Abdulle, Y. Bai, Adaptive reduced basis finite element heterogeneous multiscale method, Comput. Methods Appl. Mech. Engrg. 257 (2013) 203–220.

[26] A. Abdulle, E. Weinan, B. Engquist, E. Vanden-Eijnden, The heterogeneous multiscale method, Acta Numer. 21 (2012) 1–87.

[27] B. Le, J. Yvonnet, Q.-C. He, Computational homogenization of nonlinear elastic materials using neural networks, Internat. J. Numer. Methods Engrg. 104 (12) (2015) 1061–1084.

[28] C. Rao, Y. Liu, Three-dimensional convolutional neural network (3D-CNN) for heterogeneous material homogenization, Comput. Mater. Sci. 184 (2020) 109850.

[29] X. Lu, D.G. Giovanis, J. Yvonnet, V. Papadopoulos, F. Detrez, J. Bai, A data-driven computational homogenization method based on neural networks for the nonlinear anisotropic electrical response of graphene/polymer nanocomposites, Comput. Mech. 64 (2019) 307–321.

[30] W.T. Leung, G. Lin, Z. Zhang, NH-PINN: Neural homogenization-based physics-informed neural network for multiscale problems, J. Comput. Phys. 470 (2022) 111539.

[31] V.M. Nguyen-Thanh, L.T.K. Nguyen, T. Rabczuk, X. Zhuang, A surrogate model for computational homogenization of elastostatics at finite strain using the HDMR-based neural network approximator, 2019, arXiv preprint arXiv:1906.02005.

[32] V. Minh Nguyen-Thanh, L. Trong Khiem Nguyen, T. Rabczuk, X. Zhuang, A surrogate model for computational homogenization of elastostatics at finite strain using high-dimensional model representation-based neural network, Internat. J. Numer. Methods Engrg. 121 (21) (2020) 4811–4842.

[33] N. Feng, G. Zhang, K. Khandelwal, Finite strain FE2 analysis with data-driven homogenization using deep neural networks, Comput. Struct. 263 (2022) 106742.

[34] J. Han, Y. Lee, A neural network approach for homogenization of multiscale problems, Multiscale Model. Simul. 21 (2) (2023) 716–734.

[35] H. Wessels, C. Böhm, F. Aldakheel, M. Hüpgen, M. Haist, L. Lohaus, P. Wriggers, Computational homogenization using convolutional neural networks, in: Current Trends and Open Problems in Computational Mechanics, Springer, 2022, pp. 569–579.

[36] N. Black, A.R. Najafi, Deep neural networks for parameterized homogenization in concurrent multiscale structural optimization, Struct. Multidiscip. Optim. 66 (1) (2023) 20.

[37] C. Soize, R. Ghanem, Data-driven probability concentration and sampling on manifold, J. Comput. Phys. 321 (2016) 242–258, http://dx.doi.org/10.1016/j.jcp.2016.05.044.

[38] C. Soize, R. Ghanem, Probabilistic learning on manifolds, Found. Data Sci. 2 (3) (2020) 279–307, http://dx.doi.org/10.3934/fods.2020013.

[39] C. Soize, R. Ghanem, Probabilistic learning on manifolds (PLoM) with partition, Internat. J. Numer. Methods Engrg. 123 (1) (2022) 268–290, http://dx.doi.org/10.1002/nme.6856.

[40] C. Farhat, R. Tezaur, T. Chapman, P. Avery, C. Soize, Feasible probabilistic learning method for model-form uncertainty quantification in vibration analysis, AIAA J. 57 (11) (2019) 4978–4991, http://dx.doi.org/10.2514/1.J057797.

[41] 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, J. Comput. Phys. 399 (2019) 108930, http://dx.doi.org/10.1016/j.jcp.2019.108930.

[42] M. Arnst, C. Soize, K. Bulthies, Computation of sobol indices in global sensitivity analysis from small data sets by probabilistic learning on manifolds, Int. J. Uncertain. Quantif. 11 (2) (2021) 1–23, http://dx.doi.org/10.1615/Int.J.UncertaintyQuantification.2020032674.

[43] E. Capiez-Lernout, C. Soize, Nonlinear stochastic dynamics of detuned bladed disks with uncertain mistuning and detuning optimization using a probabilistic machine learning tool, Int. J. Non-Linear Mech. 143 (2022) 104023, http://dx.doi.org/10.1016/j.ijnonlinmec.2022.104023.

[44] R. Ghanem, C. Soize, L. Mehrez, V. Aitharaju, Probabilistic learning and updating of a digital twin for composite material systems, Internat. J. Numer. Methods Engrg. 123 (13) (2022) 3004–3020, http://dx.doi.org/10.1002/nme.6430.

[45] C. Safta, R.G. Ghanem, M.J. Grant, M. Sparapany, H.N. Najm, Trajectory design via unsupervised probabilistic learning on optimal manifolds, Data-Centr. Eng. 3 (2022) e26, http://dx.doi.org/10.1017/dce.2022.26.

[46] A. Sinha, C. Soize, C. Desceliers, G. Cunha, Aeroacoustic liner impedance metamodel from simulation and experimental data using probabilistic learning, AIAA J. 61 (11) (2023) 4926–4934, http://dx.doi.org/10.2514/1.J062991.

[47] K. Zhong, J.G. Navarro, S. Govindjee, G.G. Deierlein, Surrogate modeling of structural seismic response using Probabilistic Learning on Manifolds, Earthq. Eng. Struct. Dyn. 52 (8) (2023) 2407–2428, http://dx.doi.org/10.1002/eqe.3839.

[48] O. Ezvan, C. Soize, C. Desceliers, R. Ghanem, Updating an uncertain and expensive computational model in structural dynamics based on one single target FRF using a probabilistic learning tool, Comput. Mech. 71 (2023) 1161–1177, http://dx.doi.org/10.1007/s00466-023-02301-2.

[49] J.O. Almeida, F.A. Rochinha, A probabilistic learning approach applied to the optimization of wake steering in wind farms, J. Comput. Inf. Sci. Eng. 23 (1) (2023) 011003, http://dx.doi.org/10.1115/1.4054501.

[50] K. Zhong, J.G. Navarro, S. Govindjee, G.G. Deierlein, Surrogate modeling of structural seismic response using probabilistic learning on manifolds, Earthq. Eng. Struct. Dyn. 52 (8) (2023) 2407–2428.

[51] C. Soize, R. Ghanem, Probabilistic-learning-based stochastic surrogate model from small incomplete datasets for nonlinear dynamical systems, Comput. Methods Appl. Mech. Engrg. 418 (2023) 116498, http://dx.doi.org/10.1016/j.cma.2023.116498.

[52] P.G. Ciarlet, Mathematical Elasticity: Three-Dimensional Elasticity, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2021.

[53] P. Wriggers, Nonlinear Finite Element Methods, Springer Science & Business Media, 2008.

[54] J. Bonet, R.D. Wood, Nonlinear Continuum Mechanics for Finite Element Analysis, second ed., Cambridge University Press, 2008, http://dx.doi.org/10.1017/CBO9780511755446.

[55] C. Huet, Application of variational concepts to size effects in elastic heterogeneous bodies, J. Mech. Phys. Solids 38 (6) (1990) 813–841.

[56] M. Ostoja-Starzewski, Microstructural Randomness and Scaling in Mechanics of Materials, Chapman & Hall, 2007.

[57] F. Feyel, A multilevel finite element method (FE2) to describe the response of highly non-linear structures using generalized continua, Comput. Methods Appl. Mech. Eng. 192 (28–30) (2003) 3233–3244.

[58] R. Hill, On constitutive macro-variables for heterogeneous solids at finite strain, Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci. 326 (1565) (1972) 131–147.

[59] J. Guilleminot, C. Soize, On the statistical dependence for the components of random elasticity tensors exhibiting material symmetry properties, J. Elasticity 111 (2) (2013) 109–130.

[60] B. Staber, J. Guilleminot, Stochastic modeling and generation of random fields of elasticity tensors: A unified information-theoretic approach, C.R. Mech. 345 (6) (2017) 399–416.

[61] D.-A. Hun, J. Guilleminot, J. Yvonnet, M. Bornert, Stochastic multiscale modeling of crack propagation in random heterogeneous media, Internat. J. Numer. Methods Engrg. 119 (13) (2019) 1325–1344.

[62] C. Soize, Non-Gaussian positive-definite matrix-valued random fields for elliptic stochastic partial differential operators, Comput. Methods Appl. Mech. Engrg. 195 (1) (2006) 26–64.

[63] C.E. Shannon, A mathematical theory of communication, Bell Syst. Tech. J. 27 (1948) 379–423/623–659.

[64] E. Jaynes, Information theory and statistical mechanics I, Phys. Rev. 106 (4) (1957) 620–630.

[65] E. Jaynes, Information theory and statistical mechanics II, Phys. Rev. 108 (2) (1957) 171–190.

[66] C. Soize, Uncertainty Quantification: An Accelerated Course with Advanced Applications in Computational Engineering, Springer Cham, 2017.

[67] J. Guilleminot, 12 - Modeling non-Gaussian random fields of material properties in multiscale mechanics of materials, in: Y. Wang, D.L. McDowell (Eds.), Uncertainty Quantification in Multiscale Materials Modeling, in: Elsevier Series in Mechanics of Advanced Materials, Woodhead Publishing, 2020, pp. 385–420, http://dx.doi.org/10.1016/B978-0-08-102941-1.00012-2, URL https://www.sciencedirect.com/science/article/pii/B9780081029411000122.

[68] B. Staber, J. Guilleminot, Stochastic modeling of a class of stored energy functions for incompressible hyperelastic materials with uncertainties, C.R. Mech. 343 (9) (2015) 503–514.

[69] B. Staber, J. Guilleminot, Stochastic modeling of the ogden class of stored energy functions for hyperelastic materials: the compressible case, J. Appl. Math. Mech. 97 (3) (2017) 273–295, http://dx.doi.org/10.1002/zamm.201500255.

[70] B. Staber, J. Guilleminot, A random field model for anisotropic strain energy functions and its application for uncertainty quantification in vascular mechanics, Comput. Methods Appl. Mech. Engrg. 333 (2018) 94–113, http://dx.doi.org/10.1016/j.cma.2018.01.001, URL https://www.sciencedirect.com/science/article/pii/S0045782518300033.

[71] P. Chen, J. Guilleminot, Spatially-dependent material uncertainties in anisotropic nonlinear elasticity: Stochastic modeling, identification, and propagation, Comput. Methods Appl. Mech. Engrg. 394 (2022) 114897.

[72] B. Staber, J. Guilleminot, Stochastic hyperelastic constitutive laws and identification procedure for soft biological tissues with intrinsic variability, J. Mech. Behav. Biomed. Mater. 65 (2017) 743–752.

[73] B. Staber, J. Guilleminot, C. Soize, J. Michopoulos, A. Iliopoulos, Stochastic modeling and identification of a hyperelastic constitutive model for laminated composites, Comput. Methods Appl. Mech. Engrg. 347 (2019) 425–444.

[74] A. Mihai, Stochastic Elasticity: A Nondeterministic Approach to the Nonlinear Field Theory, Springer Cham, 2022.

[75] J. Guilleminot, C. Soize, Encyclopedia of Continuum Mechanics, Springer Berlin Heidelberg, Berlin, Heidelberg, 2017, pp. 1–9.

[76] C.J. Permann, D.R. Gaston, D. Andrš, R.W. Carlsen, F. Kong, A.D. Lindsay, J.M. Miller, J.W. Peterson, A.E. Slaughter, R.H. Stogner, R.C. Martineau, MOOSE: Enabling massively parallel multiphysics simulation, SoftwareX 11 (2020) 100430.

[77] G. Picinbono, H. Delingette, N. Ayache, Non-linear anisotropic elasticity for real-time surgery simulation, Graph. Models 65 (5) (2003) 305–321.

[78] R. Coifman, S. Lafon, A. Lee, M. Maggioni, B. Nadler, F. Warner, S. Zucker, Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps, Proc. Natl. Acad. Sci. USA 102 (21) (2005) 7426–7431, http://dx.doi.org/10.1073/pnas.0500334102.

[79] C. Soize, R. Ghanem, Physics-constrained non-Gaussian probabilistic learning on manifolds, Internat. J. Numer. Methods Engrg. 121 (1) (2020) 110–145, http://dx.doi.org/10.1002/nme.6202.

[80] C. Soize, Probabilistic learning inference of boundary value problem with uncertainties based on Kullback-Leibler divergence under implicit constraints, Comput. Methods Appl. Mech. Engrg. 395 (2022) 115078, http://dx.doi.org/10.1016/j.cma.2022.115078.