Supplementary Material: Self-supervised optimization of random material microstructures in the small-data regime

Maximilian Rixner[1] and Phaedon-Stelios Koutsourelakis[*,1,2]

[1] *Technical University of Munich, Department of Engineering Physics and Computation, Professorship of Data-driven Materials Modeling*
[2] *Munich Data Science Institute (MDSI - Core Member)*

[*] *Corresponding author, p.s.koutsourelakis@tum.de*

Supplementary Note

In the following, we provide details specific to the algorithmic implementation and the numerical simulations.

1.1 Process-Structure linkage

As mentioned in the main text, the discretized, two-phase random microstructures employed in the numerical illustrations are represented by a random vector $\boldsymbol{x}$ which arises by thresholding a two-dimensional, zero-mean, unit-variance Gaussian field, in its discretized form denoted by the vector $\boldsymbol{x}_g$. The cutoff threshold x_0 is specified by the desired volume fraction and the parameters φ are associated with the spectral density function (SDF) $G(\boldsymbol{w})$ of the underlying Gaussian field. The SDF $G(\boldsymbol{w})$ arises as the Fourier dual of the autocovariance, where $\boldsymbol{w} = [w_1, w_2]^T \in \mathbb{R}^2$ denotes the wavenumbers. We express the SDF as:

$$G(\boldsymbol{w}) = \sum_{i=1}^{Q} \gamma_i h_i(\boldsymbol{w}; \boldsymbol{\mu}_i, \sigma_i) \tag{1}$$

where the functions h_i are Radial Basis Functions (RBFs) which depend on the parameters $\boldsymbol{\mu}_i, \sigma_i$ and have the functional form:

$$h(\boldsymbol{w}; \boldsymbol{\mu}, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}||\boldsymbol{w}-\boldsymbol{\mu}||^2} \tag{2}$$

This form is adopted because it automatically ensures the positivity of the resulting SDF. Eq. (1) is also known as a *spectral mixture kernel*, which defines a universal approximator for sufficiently large Q [1]. In our simulations, the parameters $\{\boldsymbol{\mu}_i\}$, i.e. the centers of RBFs, were fixed to a uniform grid in $[0, w_{max}]^2$, with $w_{max} = 65.0$ and $\sigma_i = 12.0,\ \forall i$. Finally the weights γ_i are related to the optimization variables φ through a softmax transformation:

$$\gamma_i = \frac{e^{\varphi_i}}{\sum_{j=1}^{Q} e^{\varphi_j}} \tag{3}$$

This is employed so that the resulting SDF integrates to 1 which is the variance of the corresponding Gaussian field. We made use of a spectral representation of the underlying Gaussian field (and therefore of $\boldsymbol{x}_g$) on the basis of its φ-controlled SDF and according to the formulations detailed in [2, 3, 4]. The thresholded Gaussian vector $\boldsymbol{x}_g$ gives rise to the binary microstructure $\boldsymbol{x}$ as described above and we denote summarily the corresponding transformation as:

$$\boldsymbol{x} = \boldsymbol{F}_\varphi(\boldsymbol{\Psi}) \tag{4}$$

where $\boldsymbol{\Psi}$ denotes a vector of so-called random phase angles [2]. It consists of independent random variables uniformly distributed in $[0, 2\pi]$, and its dimension depends on the discretization of the spectral domain. A direct implication of Eq. (4) is that the process-structure density $p(\boldsymbol{x}|\varphi)$ can now be expressed as:

$$p(\boldsymbol{x}|\varphi) = \int \delta(\boldsymbol{x} - \boldsymbol{F}_\varphi(\boldsymbol{\Psi}))\ p(\boldsymbol{\Psi})\ d\boldsymbol{\Psi} \tag{5}$$

where $p(\boldsymbol{\Psi})$ is the product of uniform densities $\mathcal{U}[0, 2\pi]$. As a result, expectations of arbitrary functions, say $f(\boldsymbol{x})$, with respect to $p(\boldsymbol{x}|\varphi)$ can now be written as:

$$\mathbb{E}_{p(\boldsymbol{x}|\varphi)}[f(\boldsymbol{x})] = \int f(\boldsymbol{x})\, \delta(\boldsymbol{x} - \boldsymbol{F}_\varphi(\boldsymbol{\Psi}))\, p(\boldsymbol{\Psi})\ \mathrm{d}\boldsymbol{x}\ \mathrm{d}\boldsymbol{\Psi} \tag{6}$$

$$= \int f(\boldsymbol{F}_\varphi(\boldsymbol{\Psi}))\, p(\boldsymbol{\Psi})\, \mathrm{d}\boldsymbol{\Psi} \tag{7}$$

1.2 VB-EM-Algorithm

By making use of Eq. (7) above, we can write the ELBO for the log-expected utility (type (O1) problems) as

$$\begin{aligned}
\log U^{\mathcal{D}}_{1,\mathcal{M}}(\varphi) &= \log \mathbb{E}_{p(\boldsymbol{x}|\varphi)} \left[\int u(\boldsymbol{\kappa})\, p_{\mathcal{M}}(\boldsymbol{\kappa}|\boldsymbol{x}, \mathcal{D})\, \mathrm{d}\boldsymbol{\kappa} \right] \\
&= \log \int u(\boldsymbol{\kappa})\, p_{\mathcal{M}}(\boldsymbol{\kappa}|\boldsymbol{F}_\varphi(\boldsymbol{\Psi}), \mathcal{D})\, p(\boldsymbol{\Psi})\, \mathrm{d}\boldsymbol{\kappa}\, \mathrm{d}\boldsymbol{\Psi} \\
&\geq \mathbb{E}_{q_{\boldsymbol{\xi}}(\boldsymbol{\kappa}, \boldsymbol{\Psi})} \left[\log \frac{u(\boldsymbol{\kappa})\, p_{\mathcal{M}}(\boldsymbol{\kappa}|\boldsymbol{F}_\varphi(\boldsymbol{\Psi}), \mathcal{D})\, p(\boldsymbol{\Psi})}{q_{\boldsymbol{\xi}}(\boldsymbol{\kappa}, \boldsymbol{\Psi})} \right] \\
&= \mathcal{F}(\varphi, q_{\boldsymbol{\xi}}(\boldsymbol{\kappa}, \boldsymbol{\Psi}))
\end{aligned} \tag{8}$$

where expectations with respect to $\boldsymbol{x}$ have been substituted by integrations with respect to the (primal) random variables $\boldsymbol{\Psi}$ arising from the spectral representation. Similarly the variational density is expressed with respect to $\boldsymbol{\Psi}$, i.e. $q_{\boldsymbol{\xi}}(\boldsymbol{\kappa}, \boldsymbol{\Psi})$ (as opposed to $q_{\boldsymbol{\xi}}(\boldsymbol{\kappa}, \boldsymbol{x})$). The ELBO of the (O2)-type problems can similarly be written as

$$\begin{aligned}
\log U^{\mathcal{D}}_{2,\mathcal{M}}(\varphi) &= \int p_{target}(\boldsymbol{\kappa}) \log p_{\mathcal{M}}(\boldsymbol{\kappa}|\varphi, \mathcal{D})\, \mathrm{d}\boldsymbol{\kappa} \\
&\approx \frac{1}{S} \sum_{s=1}^{S} \log p_{\mathcal{M}}\left(\boldsymbol{\kappa}^{(s)} \middle| \varphi, \mathcal{D}\right) \qquad \boldsymbol{\kappa}^{(s)} \overset{\text{i.i.d.}}{\sim} p_{target}(\boldsymbol{\kappa}) \\
&= \frac{1}{S} \sum_{s=1}^{S} \log \int p_{\mathcal{M}}\left(\boldsymbol{\kappa}^{(s)} \middle| \boldsymbol{F}_\varphi(\boldsymbol{\Psi}), \mathcal{D}\right) p(\boldsymbol{\Psi})\, \mathrm{d}\boldsymbol{\Psi} \\
&\geq \frac{1}{S} \sum_{s=1}^{S} \mathbb{E}_{q^{(s)}_{\boldsymbol{\xi}}(\boldsymbol{\Psi})} \left[\log \frac{p_{\mathcal{M}}\left(\boldsymbol{\kappa}^{(s)} \middle| \boldsymbol{F}_\varphi(\boldsymbol{\Psi}), \mathcal{D}\right) p(\boldsymbol{\Psi})}{q^{(s)}_{\boldsymbol{\xi}}(\boldsymbol{\Psi})} \right] \\
&= \sum_{s=1}^{S} \mathcal{F}_s\left(q^{(s)}_{\boldsymbol{\xi}}(\boldsymbol{\Psi}), \varphi\right)
\end{aligned} \tag{9}$$

The expression for $\log U^{\mathcal{D}}_{2,\mathcal{M}}(\varphi)$ follows from the observation that minimization of the Kullback-Leibler divergence is equivalent to maximization of $\int p_{target}(\boldsymbol{\kappa}) \log p_{\mathcal{M}}(\boldsymbol{\kappa}|\varphi, \mathcal{D})\, \mathrm{d}\boldsymbol{\kappa}$ due to the following relations:

$$\begin{aligned}
&KL(p_{target}(\boldsymbol{\kappa})||p_{\mathcal{M}}(\boldsymbol{\kappa}|\varphi, \mathcal{D})) \\
&= \mathbb{E}_{p_{target}(\boldsymbol{\kappa})} \left[\log \frac{p_{target}(\boldsymbol{\kappa})}{p_{\mathcal{M}}(\boldsymbol{\kappa}|\varphi, \mathcal{D})} \right] \\
&= -\mathbb{E}_{p_{target}(\boldsymbol{\kappa})} [\log p_{\mathcal{M}}(\boldsymbol{\kappa}|\varphi, \mathcal{D})] - \underbrace{\mathbb{H}[p_{target}(\boldsymbol{\kappa})]}_{\text{independent of } \varphi}
\end{aligned} \tag{10}$$

and consequently

$$\begin{aligned}
\varphi^* &= \arg\min_{\varphi} KL(p_{target}(\boldsymbol{\kappa})||p_{\mathcal{M}}(\boldsymbol{\kappa}|\varphi, \mathcal{D})) \\
&= \arg\max_{\varphi} \mathbb{E}_{p_{target}(\boldsymbol{\kappa})} [\log p_{\mathcal{M}}(\boldsymbol{\kappa}|\varphi, \mathcal{D})] \\
&= \arg\max U^{\mathcal{D}}_{2,\mathcal{M}}(\varphi)
\end{aligned} \tag{11}$$

Since the maximization of the ELBO w.r.t. $\boldsymbol{\xi}$ and φ is not possible in closed form, noisy estimates of the gradients $\hat{\boldsymbol{g}}_\varphi \approx \nabla_\varphi \mathcal{F}(\varphi, q_{\boldsymbol{\xi}}(\boldsymbol{\kappa}, \boldsymbol{\Psi}))$ and $\hat{\boldsymbol{g}}_{\boldsymbol{\xi}} \approx \nabla_{\boldsymbol{\xi}} \mathcal{F}(\varphi, q_{\boldsymbol{\xi}}(\boldsymbol{\kappa}, \boldsymbol{\Psi}))$ are obtained using Monte Carlo and the reparametrization trick ([5] - see ensuing

discussion). For our numerical illustrations, the optimization with respect to φ and $\boldsymbol{\xi}$ is carried out with stochastic gradient ascent and the Adam optimizer [6] in PyTorch [7].

Representation

Instead of the bounded phase angles $\boldsymbol{\Psi}$, we employ the unbounded and normally distributed variables $\boldsymbol{\Psi}_t$ (i.e. $\boldsymbol{\Psi}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$) which are related through the error function $\mathrm{erf}(\cdot)$ as follows:

$$\Psi_i = 0.5 \cdot \left(1 + \mathrm{erf}\left(\frac{\Psi_{t,i}}{\sqrt{2\pi}}\right)\right) \cdot 2\pi \tag{12}$$

As a result, all expectations with respect to $p(\boldsymbol{\Psi})$ are substituted by expectations with respect to $p(\boldsymbol{\Psi}_t) = \mathcal{N}(\mathbf{0}, \boldsymbol{I})$.

In addition, instead of the Heaviside function defining the binary vector $\boldsymbol{x}$ from the underlying Gaussian $\boldsymbol{x}_g$ as $x_i = H(x_{g,i} - x_0)$, we employ the differentiable transformation

$$x_i = \frac{\tanh\left(\epsilon\left(x_{g,i} - x_0\right)\right) + 1}{2} \tag{13}$$

We note that as $\epsilon \to \infty$ we recover the Heaviside function and therefore a hard truncation. While the resulting x_i's are only approximately binary for finite ε (we used $\varepsilon = 25$), these were used in all computations involved in the surrogate and the optimization. An advantage is that this enables the use of the reparametrization trick [5] to estimate the ELBO and its gradients as explained in the previous section.

Low-rank Variational Approximation

For $q_{\boldsymbol{\xi}}(\boldsymbol{\kappa}, \boldsymbol{\Psi}_t) \in \mathcal{Q}$ we adopt the choice of a low-rank multivariate Gaussian distribution (with $\boldsymbol{z} = [\boldsymbol{\kappa}, \boldsymbol{\Psi}_t]^T \in \mathbb{R}^{d_z}$), i.e.

$$q_{\boldsymbol{\xi}}(\boldsymbol{z}) = \mathcal{N}\left(\boldsymbol{z} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma} = \mathrm{diag}(\boldsymbol{d}) + \boldsymbol{L}\boldsymbol{L}^T\right) \tag{14}$$

with $\boldsymbol{L} \in \mathbb{R}^{d_z \times M}$ and $M << d_z$. The variational parameters are given by $\boldsymbol{\xi} = \{\boldsymbol{\mu}, \boldsymbol{d}, \boldsymbol{L}\}$ with $\dim(\boldsymbol{\xi}) = \mathcal{O}(d_z \cdot M)$. This particular choice enables to capture enough of the correlation structure to drive the EM updates, while remaining scalable with regards to the (generally large) dimension d_z of the problem (we used $M = 50$). We note that a fully Bayesian treatment of the surrogate could be accomplished by including the neural network parameters $\boldsymbol{\theta}$ in the variational inference framework. While we could also drive the EM-algorithm via, e.g., Markov Chain Monte Carlo or Sequential Monte Carlo, the choice of variational inference is computationally faster and additionally enables monitoring of convergence through the ELBO $\mathcal{F}$.

Tempering

When specifying the material design objective, it is numerically advantageous to pursue a tempering schedule, in particular if the desired material behaviour deviates strongly from the initially observed dataset $\mathcal{D}$, or the properties $\boldsymbol{\kappa}$ associated with the initial guess $\varphi^{(0)}$. In the following we discuss an adaptive tempering strategy which - for the sake of illustration - we explain in the context of a utility function $u(\boldsymbol{\kappa}) = \mathbb{I}_{\mathcal{K}}(\boldsymbol{\kappa})$. Instead of trying to obtain $\varphi^* = \arg\max p(\boldsymbol{\kappa} \in \mathcal{K}|\varphi)$ directly, we instead introduce a sequence of target domains $\mathcal{K}^{(r)}, r = 1..., R$, such that $\mathcal{K}^{(R)} = \mathcal{K}$. To this end we may define $\mathcal{K}^{(0)}$ in such a way, that (according to the model belief) a non-negligible number of samples $\boldsymbol{\kappa} \sim p_{\mathcal{M}}\left(\boldsymbol{\kappa}\middle|\varphi^{(0)}, \mathcal{D}\right)$ fall into the domain $\mathcal{K}^{(0)}$. In order to assess how strongly the tempered target domain $\mathcal{K}^{(r)}$ can be shifted towards the desired $\mathcal{K}$ in each each step r, we can make use of the *effective sample size* (ESS) [8]. For an ensemble of N_w phase angles $\{\boldsymbol{\Psi}^{(n)}\}_{n=1}^{N_w}$ generated from $q(\boldsymbol{\Psi})$, we introduce the corresponding weights as the model-based belief that the material properties $\boldsymbol{\kappa}$ reside in the tempered target domain $\mathcal{K}^{(r)}$

$$w_n^r = \int \mathbb{I}_{\mathcal{K}^{(r)}}(\boldsymbol{\kappa})\, p_{\mathcal{M}}\left(\boldsymbol{\kappa}\middle|\boldsymbol{F}_\varphi\left(\boldsymbol{\Psi}_t^{(n)}\right), \mathcal{D}\right) \mathrm{d}\boldsymbol{\kappa} \tag{15}$$

Denoting the normalized weights as $\tilde{w}_n^r = w_n^r / \left(\sum_{n=1}^{N_w} w_n^r\right)$, the ESS is defined as

$$\mathrm{ESS} = \left(\sum_{n=1}^{N_w} {\tilde{w}_n^r}^2\right)^{-1} = \frac{\left(\sum_{n=1}^{N_w} w_n^r\right)^2}{\sum_{n=1}^{N_w} {w_n^r}^2} \tag{16}$$

where $\mathrm{ESS} \in [0, 1]$ represents the deterioration of sample quality induced by shifting the domain $\mathcal{K}^{(r)}$. Let $q(\boldsymbol{\Psi}_t)$ be an approximation to the posterior over the phase angles conditional on the optimality criteria (i.e. $\mathcal{K}^{(r)}$) and the current value of φ. One may then adaptively chose to shift the target domain $\mathcal{K}^{(r)} \to \mathcal{K}^{(r+1)}$ in such a manner, that the ESS of the samples generated from $q(\boldsymbol{\Psi}_t)$ does not deteriorate beyond a certain threshold value (e.g. using a bisection approach). When defining the material design objective by means of a target distribution $p_{target}(\boldsymbol{\kappa})$, similarly one may introduce tempering by gradually shifting the sample representation giving rise to the evidence lower bound.

1.3 Structure-Property linkage

Probabilistic Surrogate

The probabilistic surrogate employed is based on a parametric convolutional neural network (see illustration in main text), where a split in the final dense layers gives rise to (separately) the mean vector $\boldsymbol{m}_{\boldsymbol{\theta}}(\boldsymbol{x})$, as well as the covariance matrix $\boldsymbol{S}_{\boldsymbol{\theta}}(\boldsymbol{x})$ (assumed to be diagonal). The specific choices made regarding the neural network architecture are based on prior published work (e.g. [9, 10]). Each block displayed in the illustration corresponds to 2d convolutions with a subsequent non-linear activation function (Leaky ReLU) ,followed by average pooling. The convolutional layers employ a (3×3) kernel, which in combination with appropriate padding leaves the size of the feature map unchanged [1]. The subsequent average pooling always employs a (2×2) kernel (and identical stride), such that the size of the feature maps is reduced by half in each block. For the numerical results presented, after a sequence of 4 such blocks (with an increasing depth of $4, 8, 12$ and 16 channels in the feature maps), the resulting feature representation extracted from the microstructure is flattened and enters first a shared hidden layer (of width 30), subsequently splitting up into two more layers that map to the mean $\boldsymbol{\mu}_{\boldsymbol{\theta}}$ and the diagonal covariance matrix $\boldsymbol{S}_{\boldsymbol{\theta}}$ (the positivity of the latter is ensured via an exponential transformation). The two phases were encoded as a $(+1)$ and (-1) for the CNN, as this is numerically more expedient compared to an $\{1, 0\}$ representation of the phases. For all numerical experiments presented, the neural network was trained with a batch size of $N_{bs} = 128$. To add regularization, a weight decay of 10^{-5} was used, and additionally a dropout layer (with $p = 0.05$) was introduced before the first dense layer. The neural network was trained on the log-likelihood of the data making again use of the Adam optimizer for the stochastic updates of the parameters $\boldsymbol{\theta}$.

Physical model for the computation of properties $\boldsymbol{\kappa}$

In the following we provide a more detailed description of the physical models in the structure-property linkage abstractly represented as $\boldsymbol{\kappa}(\boldsymbol{x})$, i.e., the link between the microstructures and their physical properties we want to control in this study. Note that the specific choice of $\boldsymbol{\kappa}(\boldsymbol{x})$ does not have any direct bearing on the optimization, as the structure-property linkage only enters into the generation of the training data $\mathcal{D}$ for the surrogate. For our numerical illustrations we make use of both the effective *thermal* as well as *mechanical* properties of the microstructures $\boldsymbol{x} \sim p(\boldsymbol{x}|\varphi)$. We present details regarding the numerical computation of the effective mechanical properties, with the thermal properties following by analogy. In order to quantify the macroscopic response of a microstructure, we consider a linear, isotropic elasticity problem on the microscopic scale for a representative volume element (RVE). At this scale the behaviour of the microstructure is characterized by the balance equation (BE) and constitute equation (CE)

$$\text{(BE)}: \quad \mathrm{div}(\boldsymbol{\sigma}) = \mathbf{0} \qquad \forall \boldsymbol{s} \in \Omega_{\mathrm{RVE}} \tag{17}$$

$$\text{(CE)}: \quad \boldsymbol{\sigma} = \mathbb{C}(\boldsymbol{s}) : \boldsymbol{\epsilon} \qquad \forall \boldsymbol{s} \in \Omega_{\mathrm{RVE}} \tag{18}$$

[1] The discretized microstructures regarded as a vectors $\boldsymbol{x} \in \{0, 1\}^{4096}$ are of course reshaped into their original (64×64) un-flattened tensor representation for the CNN.

where $\boldsymbol{\sigma}, \boldsymbol{\epsilon}$ denote microscopic stress and strain, while $\mathbb{C}(\boldsymbol{s})$ constitutes the heterogeneous elasticity tensor $\mathbb{C}(\boldsymbol{s})$. For a binary microstructures where the two phases occupy (random) subdomains $\mathcal{V}_0$ and $\mathcal{V}_1$ (with $\mathcal{V}_0 \cap \mathcal{V}_1 = \emptyset$ and $\mathcal{V}_0 \cup \mathcal{V}_1 = \Omega_{\text{RVE}}$) the elasticity tensor follows as:

$$\mathbb{C}(\boldsymbol{s}) = \begin{cases} \mathbb{C}_1, & \text{if } \boldsymbol{s} \in \mathcal{V}_1 \\ \mathbb{C}_0, & \text{if } \boldsymbol{s} \in \mathcal{V}_0 \end{cases} \tag{19}$$

In our case, the elasticity tensors $\mathbb{C}_0$ and $\mathbb{C}_1$ are fully defined by the Young's moduli E_0 and E_1 of the two phases (a common Poisson's ratio of $\nu = 0.3$ was used). We define the *macroscopic* stress $\boldsymbol{\Sigma} = \langle \boldsymbol{\sigma} \rangle$ as well as macroscopic strain $\boldsymbol{E} = \langle \boldsymbol{\epsilon} \rangle$, where $\langle \cdot \rangle$ denotes a *spatial* average of microscopic quantities over Ω_{RVE}. We then characterize the macroscopic, effective behaviour of the microstructure via [11]

$$\boldsymbol{\Sigma} = \mathbb{C}^{\text{eff}} : \boldsymbol{E} \tag{20}$$

under the constraint that (20) satisfies the averaging theorem by Hill [11, 12]

$$\boldsymbol{\Sigma} : \boldsymbol{E} = \frac{1}{|\Omega_{\text{RVE}}|} \int_{\partial\Omega_{\text{RVE}}} \boldsymbol{t} \cdot \boldsymbol{u} \, \mathrm{d}A \tag{21}$$

with microscopic tractions $\boldsymbol{t}$ and displacements $\boldsymbol{u}$. The homogenized properties $\boldsymbol{\kappa}$ are based on various entries of $\mathbb{C}^{\text{eff}}$ which is computed by solving an ensemble of elementary load cases, as given by the following macroscopic strain modes

$$\hat{\boldsymbol{E}}^{(1)} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \quad \hat{\boldsymbol{E}}^{(2)} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, \quad \hat{\boldsymbol{E}}^{(3)} = \begin{bmatrix} 0 & 0.5 \\ 0.5 & 0 \end{bmatrix} \tag{22}$$

corresponding to either a pure tension or shear mode, such that $\mathbb{C}^{\text{eff}}$ is recovered by integrating the microscopic stress $\boldsymbol{\sigma}$ to obtain $\boldsymbol{\Sigma} = \langle \boldsymbol{\sigma} \rangle$. One possible approach of imposing these elementary strain modes is based on the introduction of periodic boundary condition (as opposed to defining load cases based on displacement or tractions), which augments Eq. (17) and (18) by

$$\boldsymbol{\epsilon} = \hat{\boldsymbol{E}}^{(c)} + \nabla_{\boldsymbol{s}} \boldsymbol{v} \qquad \text{in } \Omega_{\text{RVE}} \tag{23}$$

$$\boldsymbol{v} \qquad \text{is } \Omega_{\text{RVE}}\text{-periodic} \tag{24}$$

$$\boldsymbol{t} = \boldsymbol{\sigma} \cdot \boldsymbol{n} \qquad \text{is } \Omega_{\text{RVE}}\text{-antiperiodic} \tag{25}$$

Here $\boldsymbol{v}$ denotes a periodic fluctuation (i.e., $\boldsymbol{u} = \boldsymbol{E}\boldsymbol{s} + \boldsymbol{v}$), and $\boldsymbol{t}$ are antiperiodic tractions on the boundary of the domain Ω_{RVE}. We solve for the periodic fluctuations $\boldsymbol{v}$ for all three elementary load cases $c = \{1, 2, 3\}$ using the Bubnov-Galerkin approach and the standard Finite Element Method (an additional Lagrange multiplier has to be included in the variational problem to disambiguate it with regards to rigid body transformations). The effective tangent moduli $\mathbb{C}^{\text{eff}}$ of the RVE thus obtained by the solution of the differential equations (Eq. (17), (18), (23), (24) and (25)) can be shown [11] to satisfy the averaging theorem by Hill. We finally note that $\mathbb{C}^{\text{eff}}$ and the properties of interest $\boldsymbol{\kappa}$ vary depending on the underlying, random microstructure $\boldsymbol{x}$ and the need for their repeated computation represents the computational bottleneck for the optimization of the process parameters φ.

Supplementary References

[1] Wilson, A. & Adams, R. Gaussian process kernels for pattern discovery and extrapolation. In *International conference on machine learning*, 1067–1075 (PMLR, 2013).

[2] Shinozuka, M. & Deodatis, G. Simulation of Multi-Dimensional Gaussian Stochastic Fields by Spectral Representation. *Applied Mechanics Reviews* **49**, 29–53 (1996). URL https://doi.org/10.1115/1.3101883. https://asmedigitalcollection.asme.org/appliedmechanicsreviews/article-pdf/49/1/29/5437403/29_1.pdf.

[3] Hu, B. & Schiehlen, W. On the simulation of stochastic processes by spectral representation. *Probabilistic engineering mechanics* **12**, 105–113 (1997).

[4] Sternfels, R. & Koutsourelakis, P.-S. Stochastic design and control in random heterogeneous materials. *International Journal for Multiscale Computational Engineering* **9** (2011).

[5] Kingma, D. P. & Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).

[6] Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[7] Paszke, A. *et al.* Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems* **32**, 8026–8037 (2019).

[8] Del Moral, P., Doucet, A. & Jasra, A. Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society Series B-Statistical Methodology* **68**, 411–436 (2006).

[9] Yang, Z. *et al.* Deep learning approaches for mining structure-property linkages in high contrast composites from simulation datasets. *Computational Materials Science* **151**, 278–287 (2018).

[10] Cecen, A., Dai, H., Yabansu, Y. C., Kalidindi, S. R. & Song, L. Material structure-property linkages using three-dimensional convolutional neural networks. *Acta Materialia* **146**, 76–84 (2018).

[11] Miehe, C. & Koch, A. Computational micro-to-macro transitions of discretized microstructures undergoing small strains. *Archive of Applied Mechanics* **72**, 300–317 (2002).

[12] Hill, R. On constitutive macro-variables for heterogeneous solids at finite strain. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences* **326**, 131–147 (1972).