Automatised selection of load paths to construct reduced-order models in computational damage micromechanics: from dissipation-driven random selection to Bayesian optimization.

Olivier Goury[1,2,3], David Amsallem[4], Stéphane Pierre Alain Bordas[1,5],
Wing Kam Liu[6], Pierre Kerfriden[1*]

[1]   Cardiff University, School of Engineering,
      Queen's Buildings, The Parade, Cardiff CF24 3AA, Wales, UK
[2]   DEFROST team, Inria Lille-Nord Europe,
      40 avenue du Halley, 59000 Lille, France
[3]   UMR-S 1159 Inserm - Université Paris VI Pierre et Marie Curie,
      75005 Paris, France
[4]   Department of Aeronautics and Astronautics, Stanford University,
      Mail Code 4035, Stanford, CA 94305, U.S.A
[5]   University of Luxembourg,
      Campus Kirchberg, G 007 Luxembourg
[6]   Northwestern University, Department of Mechanical Engineering,
      2145 Sheridan Rd., Evanston, IL 60208

April 12, 2016

Abstract

In this paper, we present new reliable model order reduction strategies for computational micromechanics. The difficulties rely mainly upon the high dimensionality of the parameter space represented by any load path applied onto the representative volume element (RVE). We take special care of the challenge of selecting an exhaustive snapshot set. This is treated by first using a random sampling of energy dissipating load paths and then in a more advanced way using Bayesian optimization associated with an interlocked division of the parameter space. Results show that we can insure the selection of an exhaustive snapshot set from which a reliable reduced-order model (ROM) can be built.

Keywords: model order reduction, computational homogenisation, reduced basis, Hyperreduction, damage mechanics, multiscale

1   Introduction

Multiscale modelling permits to take into account partial microscopic data when deriving engineering-scale working models. In solid mechanics, homogenisation is routinely used to obtain coarse-scale stress/strain relationships that are consistent with some statistical knowledge of the microstructure [1, 2, 3, 4]. This is particularly useful when modelling complex phenomena that would require cumbersome heuristic inference if the subscale physics was ignored. In more advanced applications of upscaling concepts, the conservation laws of the coarse-scale medium themselves may be obtained

1

from lower-scale data [5, 6]. Homogenisation can be seen as one particular class of upscaling technique, whereby coarse-scale models approximate the limit of the underlying microscale model when the scale ratio tends to zero [1, 7]. In the classical setting of micromechanics (see for instance [3, 4, 8]), the homogenisation process leads to two interlinked problems: a macroscale mechanical problem with homogeneous constitutive relations, and a microscale problem set over a representative volume element (RVE) of the microstructure, which is often interpreted as a material point of the homogeneous continuum. The solution to the macroscale problem defines a far-field loading for the RVE, usually in the form of boundary conditions. In turns, the solution of the RVE problem permits to find the homogenised coefficients of the coarse-scale constitutive relations, for instance by using micro/macro energy equivalence.

RVE problems were traditionally solved approximately using analytical or semi-analytical approaches [9, 10, 2, 11, 3]. In the last 20 years, computational homogenisation has emerged as an interesting alternative approach [12, 13, 14, 15, 16], whereby the RVE problem is solved using direct numerical simulation. In linear elasticity, the homogenised constitutive relation can be pre-computed by performing a small set of material tests. The results of these tests are then assembled in the form of a homogenised Hooke tensor that can be readily used at the coarse-scale. In a nonlinear setting, a "naive" implementation of computational homogenisation requires to solve the RVE problem at every (quadrature) point of the macroscopic domain, which, although attractive due to its generality, may render the approach prohibitively expensive. A considerable amount of recent work aims at providing
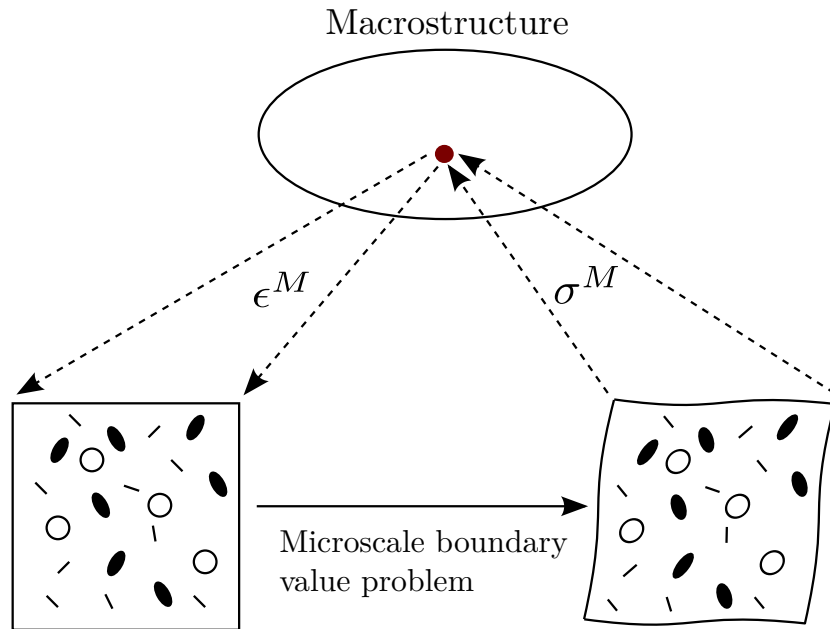

Figure 1: Semi-concurrent homogenisation procedure. At each macrostructural quadrature point, an RVE boundary value poblem can be stated with boundary conditions dictated by the macrostrain at this point. Once the boundary value problem solved, the corresponding macrostress is evaluated as a spatial average of the microstress over the RVE.

an answer to this dilemma. On the one hand, the community that relied heavily on semi-analytical approaches to solve RVE problems has developed methods to circumvent the limitations due to the restrictive assumptions upon which these approaches were traditionally based, at the cost of increased computational requirements. The (non-) uniform transformation analysis [17, 18, 19] (see also [20, 21]) and the Voronoi cell approach developed in [22] are remarkable instances of such developments. On the other hand, the community that relied primarily on computational homogenisation methods has tried to reduce the amount of RVE computations by using meta-modelling, often called meso-modelling in

this context. Such developments include the R3M [23, 24] and the method developed in [25], which both rely on a combination of a proper orthogonal decomposition (POD) expansion [26, 27] for the solution field, and a surface response approach to interpolate the coefficients of this expansion over the space of admissible loading conditions. Our proposed approach is a further step in this direction, which bypasses the need for the surface response step and replaces it by reduced-order modelling (ROM).

Projection-based reduced-order modelling is an increasingly popular technique for the fast solution of parametrised boundary-value problems. The key idea is to represent the parametric variations of the solution in a low-dimensional subspace. This subspace can be identified using the snapshot-POD [28, 29, 30, 31, 32, 33, 34, 35, 36], which compresses the posterior information contained in an exhaustive sampling of the parameter domain, or the Reduced Basis Method [37, 38, 39, 40, 41], which searches for this attractive subspace in the form of a linear combination of samples chosen quasi-optimally via a Greedy algorithm ("offline stage"). In a second stage, the boundary value problem is projected into this subspace, for instance by a Galerkin method, resulting in a reduced model of number of unknowns equal to the dimension of the attractive space. This reduced model is used to deliver an approximation of the solution to the parametric BVP for any set of parameters, and as such can be seen as an implicit interpolation method over the parameter domain ("online stage"). Early contributions concerning these type of methods have shown an increased accuracy compared to traditional response surface methods, for a given sampling of the parameter domain. Perhaps more importantly, these methods are based on approximation theories, and therefore "naturally" incorporate reliability estimates (e.g. [37, 29, 40, 35]).

In this paper, we propose to reformulate the nonlinear RVE problem as a parametrised boundary value problem, and subsequently to approximate it using projection-based ROM. Without loss of generality, we will consider an elastic damageable material represented by a network of damageable beams, with non-homogeneous material properties representing a random distribution of stiff inclusions into a softer matrix. The RVE problem will be parametrised by its far-field loading, represented by homogeneous Dirichlet conditions that belong to a vector space of dimension six (three in two 2D), the time evolution of the coefficients of the associated linear combination being a priori unknown, which effectively results in a parametric space of infinite dimension for the RVE. Therefore, our aim is to characterise the solution of the RVE problem for any history of the far-field load, within the restriction of ellipticity (which implicitly define the bounds of the parameter domain).

In a first attempt to approximate this parametrised solution, we will generate random loadings, enforcing a minimum amount of energy dissipation at each timestep and deploy the Galerkin-POD methodology to derive a reliable ROM. In a second, more advanced approach, we will develop a tailored Reduced Basis Approach to sample the infinite-dimensional parameter space in a reliable and efficient manner. Our procedure relies on two major ingredients. Firstly, we will make use of a gradient algorithm to find points of the parameter space that need to be corrected during the iterates of the greedy algorithm. Although the gradient-based optimisation proposed in [39] is a promising strategy, we will make use of an alternative optimisation technique based on Bayesian optimisation [42]. More precisely, the load path of worst prediction will be found using a Gaussian process regression of an error indicator following [43]. The second ingredient of our approach is to coarsen the a priori infinite-dimensional parameter by applying the complex macroscopic load hierarchically, following a sequence of piecewise linear trajectories. Specifically, we will fully train a reduced basis method in a space of proportional loadings. We will then train a new reduced basis model in an enriched parameter domain, by representing the macroscopic load as a sequence of two piecewise linear loads, and further enrich our parameter domain in this hierarchical manner until a stagnation criterion is reached.

We will pay particular attention in the efficiency of the proposed strategy. In particular, projection-based ROM in the nonlinear setting is known to require an additional level of approximation to remain efficient, known as "hyperreduction" or "system approximation" [38, 44, 45, 31, 46, 32, 33, 47]. We will make use of tailored version of the discrete empirical interpolation method (DEIM) [38, 46], which is, to date, the most widely used system approximation methodology. The original DEIM will be slightly modified to allow for the approximation of a vanishing nonlinear term in the balance equations of the discrete RVE problem. We will also propose a way to choose a good ratio between level of

approximations in the truncation of attractive subspace versus system approximation.

The paper is organised as follows. In section 2, we define the class of nonlinear homogenisation problems that we want to reduce, and explain how these problems can be parametrised. In section 3, we develop specific model order reduction approaches based on the snapshot-POD and the Reduced Basis methodologies. We highlight the pros and cons of these two distinct approaches in the context of nonlinear homogenisation, and show results for each method. Conclusions are drawn in section 4.

2   Computational homogenisation setting

We consider a generic representative volume element (RVE) occupying domain Ω (Figure 2), corresponding to a microscopically heterogeneous structure. The computational homogenisation approach that is considered in this work is a classical FE^2 scheme [12]: the RVE problem is to be solved numerically, under homogeneous Dirichlet boundary conditions, at every quadrature point of the macroscopic domain, which implicitly defines the nonlinear constitutive law at the macroscopic level. We will work under the assumption of small perturbations and isothermal mechanical evolution. The material studied in this paper is damageable elastic, but the methodology is general. In this paper, the RVE will be modelled by a 2D network of damageable beams (see for instance [48, 49] for more details), whose mechanical properties materialise heterogeneities (random distribution of stiff inclusions in our case). However, for the sake of simplicity, the idea of the approach will first be exposed in the context of continuum mechanics and then discretised, the formulation of the spatially discretised continuum-based or lattice-based model being similar.
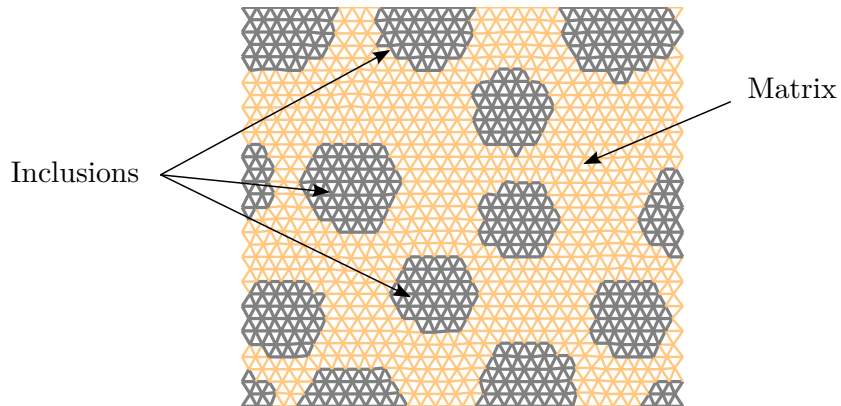

Figure 2: Lattice model of the computational representative volume element. Beams have different mechanical properties that depend on their location with respect to the distribution of heterogeneities in the computational domain. An arbitrary distribution of inclusions is chosen as a test case for this paper.

2.1   RVE boundary value problem

At the RVE level, the displacement field is additively split into a fluctuation $\tilde{\mathbf{u}}$ and a smooth (or "macroscopic") part $\bar{\mathbf{u}}$:

$$\mathbf{u}(\mathbf{x}, t) = \tilde{\mathbf{u}}(\mathbf{x}, t) + \bar{\mathbf{u}}(\mathbf{x}, t) \tag{1}$$

where the fluctuation $\tilde{\mathbf{u}}$ vanishes on the boundary $\partial\Omega$ of RVE domain Ω, t denotes time, and the smooth part of the displacement belongs to a 2-dimensional vector space[1],

$$\bar{\mathbf{u}}(t) = \boldsymbol{\epsilon}^M(t) \, (\mathbf{x} - \bar{\mathbf{x}}) \tag{2}$$

---

[1]We make the formulation in a 2D context, but the same principles apply in 3D.

Macrostructure

Effective damage tensor

Effective strain ϵ

RVE

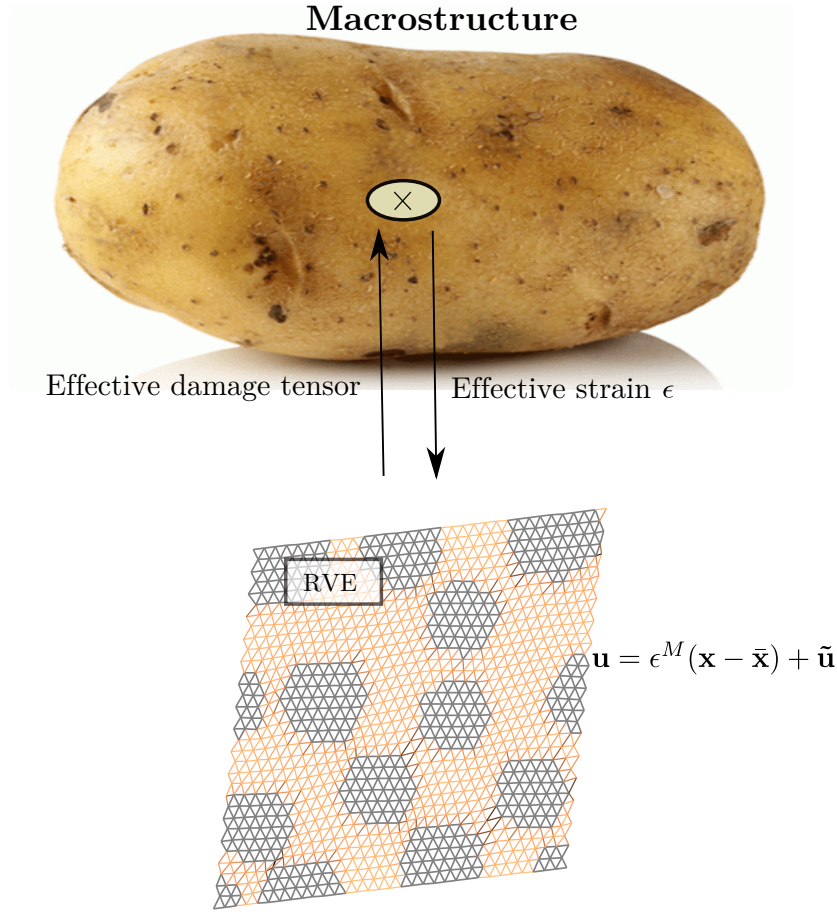$\mathbf{u} = \epsilon^M(\mathbf{x} - \bar{\mathbf{x}}) + \tilde{\mathbf{u}}$

Figure 3: Schematic representation of computational homogenisation. The constitutive law of the macro-structure is defined implicitly. The macroscopic strain is applied as boundary condition to the RVE boundary value problem. In turn, the macroscopic stress field is extracted from the solution of the RVE problem using duality principles.

where $\mathbf{x}$ is the position of a material point of the RVE, while $\bar{\mathbf{x}}$ is its centroid and $\boldsymbol{\epsilon^M}(t)$ gathers three scalar load coordinates that depend on the position of the corresponding material point of the macroscopic structure:

$$\boldsymbol{\epsilon^M}(t) = \begin{pmatrix} \epsilon_{xx}^M(t) & \epsilon_{xy}^M(t) \\ \epsilon_{xy}^M(t) & \epsilon_{yy}^M(t) \end{pmatrix}. \tag{3}$$

The mechanical equilibrium of the RVE is expressed by the principle of virtual work:

$$\int_\Omega \boldsymbol{\sigma^m} : \boldsymbol{\epsilon}(\boldsymbol{\delta u}) \, d\Omega = 0, \qquad \forall \boldsymbol{\delta u} \text{ s.t. } \boldsymbol{\delta u}_{|\partial\Omega} = 0 \tag{4}$$

where $\boldsymbol{\sigma^m}$ is the microscopic Cauchy stress, $\boldsymbol{\epsilon}$ is the strain operator that extracts the symmetric part of the gradient of a displacement vector, and $\boldsymbol{\delta u}$ is a virtual fluctuation field.

The (damageable elastic) constitutive relation of the different micro-constituents of the material is assumed to be known at any time t of the analysis:

$$\boldsymbol{\sigma^m} = \boldsymbol{\sigma^m}((\boldsymbol{\epsilon}(\mathbf{u}(\tau)))_{\tau \leq t}) \tag{5}$$

5

where rate independence, causality and locality are assumed. The history dependence that appears in the previous expression is due to non-reversible damage processes such as plasticity or damage. For the sake of clarity, explicit history-dependance of the variables will be omitted in the remainder of the paper.

2.2 Scale coupling

Following the classical computational homogenisation approach, the relationship between the macroscopic stress $\boldsymbol{\sigma}^M$ and the macroscopic strain at time t and at an arbitrary macroscopic material point can be obtained by using the Hill-Mandel micro-macro energy consistency condition, which reads in the present context:

$$\boldsymbol{\sigma}^M\left((\boldsymbol{\epsilon}(\mathbf{u}^M(\tau)))_{\tau<t}\right) : \boldsymbol{\epsilon}^{M^\star} = \frac{1}{|\Omega|} \int_{\partial\Omega} (\boldsymbol{\sigma}^m \cdot \mathbf{n}) \cdot \boldsymbol{u}^\star \, d\Gamma \tag{6}$$

for any microscopic displacement $\boldsymbol{u}^\star$ and any macroscopic strain $\boldsymbol{\epsilon}^{M^\star}$ related by the "strain averaging" ansatz $\boldsymbol{u}^\star(\mathbf{x}) = \boldsymbol{\epsilon}^{M^\star}(\mathbf{x} - \bar{\mathbf{x}})$. In the previous expression, $\mathbf{u}^M$ denotes the value of the macroscopic displacement field, and $\boldsymbol{\sigma}^m$ is the microscopic stress field that is the solution of the RVE problem under far field load $\bar{\mathbf{u}}(\tau) = \boldsymbol{\epsilon}(\mathbf{u}^M(\tau))(\mathbf{x} - \bar{\mathbf{x}})$ for any time $\tau < t$.

Equation (6) leads directly to the definition of the macroscopic stress as a function of the macroscopic strain history $(\boldsymbol{\epsilon}(\mathbf{u}^M(\tau)))_{\tau<t}$:

$$\boldsymbol{\sigma}^M\left((\boldsymbol{\epsilon}(\mathbf{u}^M(\tau)))_{\tau<t}\right) = \frac{1}{|\Omega|} \int_{\partial\Omega} (\boldsymbol{\sigma}^m \cdot \mathbf{n}) \otimes (\mathbf{x} - \bar{\mathbf{x}}) \, d\Gamma \tag{7}$$

which is subsequently used as constitutive equation for the macroscopic problem.

2.3 Space discretisation and Newton solution algorithm

Equilibrium equation (4), after substitution of the microscopic constitutive relation, is discretised in space using for instance the finite element method (FEM):

$$\forall t, \, \forall \boldsymbol{\delta}\mathbf{u} \text{ s.t. } \mathbf{C}\,\boldsymbol{\delta}\mathbf{u} = 0, \qquad \boldsymbol{\delta}\mathbf{u}^T \mathbf{f}_{\text{int}}((\mathbf{u}(\tau))_{\tau\leq t}) = 0 \, . \tag{8}$$

This equation is complemented by the kinematic admissibility condition $\mathbf{u}(t) = \tilde{\mathbf{u}}(t) + \bar{\mathbf{u}}(t)$, where $\mathbf{u}$ denotes the vector of degrees of freedom of the FEM solution of the RVE problem at time t, the vector $\bar{\mathbf{u}}(t)$ of degrees of freedom corresponding to the smooth "macroscopic" continuous field is known, and the vector of degrees of freedom $\tilde{\mathbf{u}}(t)$ corresponding to the continuous fluctuation field satisfies the discrete version of the vanishing boundary condition $\mathbf{C}\,\tilde{\mathbf{u}}(t) = 0$.

We will use a classical implicit time stepping procedure to discretise the RVE problem in time (i.e. integrate the history dependance in the microscopic constitutive relation). This will be further justified in the next paragraph. The continuous time interval $\mathcal{T}$ is discretised into n_t subintervals ($[t_n \, t_{n+1}]$). Equilibrium and kinematic relations are enforced at successive discrete times t_n, while the continuous history dependency appearing in the constitutive relation is replaced by its discrete counterpart. The fully discrete, non-linear version of the system of equations (8) arising at time t_{n} is solved using a Newton-Raphson algorithm (NR). At each iteration of this algorithm, the following linearisation is computed and solved:

$$\forall \boldsymbol{\delta}\mathbf{u} \text{ s.t. } \mathbf{C}\,\boldsymbol{\delta}\mathbf{u} = 0, \qquad \boldsymbol{\delta}\mathbf{u}^T\left(\mathbf{K}^i \boldsymbol{\Delta}\tilde{\mathbf{u}}^{i+1} + \mathbf{r}^i\right) = 0 \, , \tag{9}$$

where $\mathbf{K}^i = \frac{\partial \mathbf{f}_{\text{int}}}{\partial \mathbf{u}}|_{\mathbf{u}^i}$ is the tangent stiffness matrix, $\mathbf{r}^i = \mathbf{f}_{\text{int}}(\tilde{\mathbf{u}}^i + \bar{\mathbf{u}})$ is the residual vector and $\boldsymbol{\Delta}\tilde{\mathbf{u}}^{i+1} = \tilde{\mathbf{u}}^{i+1} - \tilde{\mathbf{u}}^i = \mathbf{u}^{i+1} - \mathbf{u}^i$ (the second equality is only true if the smooth field is used as an initialisation for the NR algorithm, i.e. $\mathbf{u}^0 = \bar{\mathbf{u}}$) is the variation in the fluctuation vector.

6

2.4 Parametrised RVE problem: description of the macroscopic load

In a FE^2 setting, the RVE problem is solved independently for every quadrature point of the macroscopic mesh. In order to apply our ROM technique, we recast the RVE problem as a family of boundary value problems subject to parameter dependency.

The parameters are the three independent components of the far field load tensor $\boldsymbol{\epsilon}^M$ (ϵ_{xx}^M, ϵ_{yy}^M and ϵ_{xy}^M). Physically, they correspond to scalar descriptors of the loading history applied to the macroscopic material point. We emphasise the fact that these parameters are three functions of time, which is not a classical setting for Model Order Reduction. This high (theoretically infinite) dimensionality is a challenge. Some realisations of the loading functions are depicted in figure 4.

The next step is to define the parameter domain, or in other words the space in which the three load functions can vary freely. This seems to be a largely problem-dependent issue, and we will focus the discussion on the class of rate-independent, damageable elastic materials. In this case, the first remark is that homogenisation loses its meaning once ellipticity is lost at the macroscopic level. Therefore, bounds are implicitly and collectively defined on the values of the loading functions by enforcing that the macroscopic tangent should remain positive definite. A second remark is that the speed at which the load is applied has no influence on the RVE solution; only the load path matters, which eliminates the need to describe loads that would be applied at different speeds but would essentially result in the same path.

We finally define a time integration scheme for the load history by forcing the macroscopic load to vary by a given amount between two successive time steps. More precisely,

$$\forall n \in [\![1, n_t]\!], \quad ||\boldsymbol{\epsilon}^{\mathbf{M}}(t_n) - \boldsymbol{\epsilon}^{\mathbf{M}}(t6_{n-1})||_2 = \Delta l \qquad \text{and} \qquad \boldsymbol{\epsilon}^{\mathbf{M}}(t_0) = 0 \tag{10}$$

Load parameter Δl should be sufficiently small for the constitutive equations of the RVE to be correctly integrated and for the nonlinear solutions algorithms to converge.

Note that in this time-discrete setting, the number of parameters is two^2 times the number of pseudo-time steps n_t, which highlights the high-dimensionality of the problem.
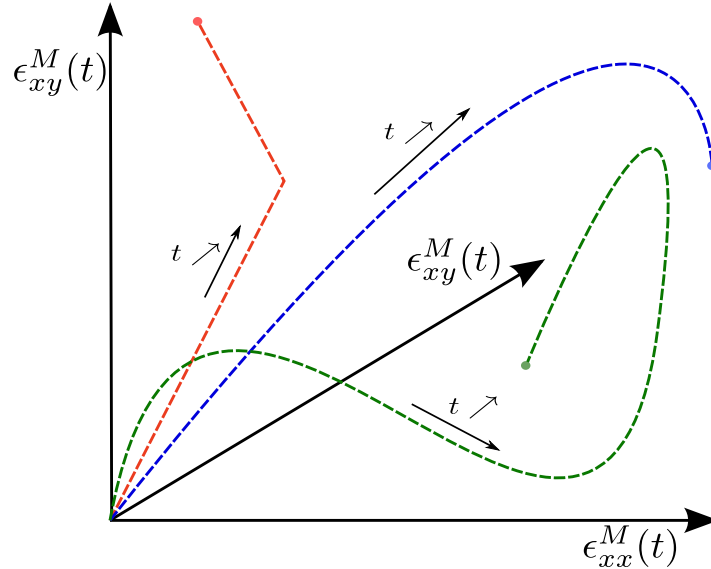

Figure 4: Representation of the parameter domain for the nonlinear RVE problem.

---

^{2}It is not 3 since we fixed the value of the load between two successive time steps.

3 Reduction of the RVE boundary value problem

Our goal is to solve the balance equations of the RVE problem for any history of the macroscopic strain at reduced costs whilst retaining the accuracy of the computed macroscopic stress field. In order to do so, we postulate that for any load applied to the RVE, the fluctuation part of the displacement field can be approximated with an acceptable level of accuracy in a vector space of small dimension, called reduced space. This space being identified, we will find an approximation of the displacement field by looking for the amplitude (*i.e.* generalised coordinates) associated to the (few) basis vectors of this space. In this context, three questions arise:

- How can we identify the reduced space?

- How can we find the generalised coordinates in an efficient and stable manner?

- How can we evaluate the reliability of the approach?

The answer to the second question is now relatively well established in the literature. We will make use of a Petrov-Galerkin projection of the discrete set of balance equations (8) into the reduced space. More precisely, we will proceed in two stages: a first "ideal" Galerkin projection[3], followed by a second stage of approximation, called "system approximation" [32] or "hyperreduction" [45] to make the solution of the projected system computationally tractable.

The answers to the first and third questions are strongly intertwined, and we describe in the following paragraphs two different manners to approach the problem.

A POD-based approach looks for the best reduced space, in the sense of the minimisation of the projection error on average over the parameter domain. In practice, this optimization problem is reduced to a problem of minimum projection error over a representative set of solutions to the parametrised problem, the so-called snapshots [26]. In the case of large parametric dimensions, the sampling of the parameter domain needs to be done in such a way that it overcomes the "curse of dimensionality", for instance by using quasi-random sampling techniques. The reliability of the approach can then be evaluated by resampling (cross-validation, bootstrap, ...) or other statistical tools. This approach suffers from two major drawbacks. Firstly, the optimality of the reduced space is established in an average sense over the parameter domain, which potentially results in inaccurate representation of outliers even for large dimensions of the reduced model. Secondly, the exhaustive sampling of the parameter domain might be prohibitively expensive, and is, in any case, inefficient if performed in a (statistically) uniform manner. The interested reader can find possible ways to tackle this difficult in [50]. Nonetheless, the POD-based methodology remains attractive because the optimization problem associated with the search of the reduced space can be solved using standard linear algebra tools, namely singular value decomposition.

The Reduced Basis [37] methodology aims at minimising the maximum projection error over the parameter domain. In practice, this is performed in a suboptimal manner using a Greedy algorithm: the reduced-order model is constructed iteratively by enriching the reduced space in order to decrease the error at the point of the parameter domain where some measure of projection error is at its largest. When reliable error estimates are available for the projection, the search for the highest level of error over the parameter domain is very efficient, which makes the approach very attractive. The sampling of the parameter domain is performed in a rational manner, which ensures that the construction of the ROM remains affordable. When error estimates are not available, the approach remains attractive in the context of large parametric dimensions. Indeed, the point of the parameter domain that corresponds to the largest level of projection error can be found using gradient-based optimization, whose numerical complexity may be made independent of the parametric dimension by using the adjoint methodology [39] to compute the sensitivities. In this setting, the "curse" of

---

[3]Some authors advocate the need for a residual minimising approach to ensure the optimality of this step [32, 41]

dimensionality can be overcome whilst retaining reliability of the ROM over the entire parameter domain[4].

In the remainder of this section, we explore these two different possibilities for the reduction of the nonlinear RVE problem. We first propose a snapshot-POD approach, where the sampling is performed randomly, enforcing the random samples to undergo a minimum dissipation at each time step. In a second stage, we will develop a Reduced Basis approach for general loading, which allows for a more continuous approach which takes into account the error of the reduced model not only at the snapshots, but also between the snapshots thanks to a gaussian process regression. We will propose specific ideas to overcome the "curse of dimensionality".

3.1 Galerkin projection of the governing equations in a reduced space

The fluctuating part of the displacement over the RVE[5] is searched for in a reduced space $\mathcal{U}_{\text{MOR}} = \text{span}\left((\boldsymbol{\phi}_i)_{i=1,N}\right)$ of dimension N (see figure 5). The displacement is parametrised by the history of the far field load $\left(\boldsymbol{\epsilon}^{\mathbf{M}}(t)\right)_{t \in [0,T]}$, which will subsequently be denoted by $\boldsymbol{\epsilon}^{\mathbf{M}}$ for simplicity. Mathematically, the surrogate for the displacement can be expressed at any time t as:

$$\mathbf{u}(t; \boldsymbol{\epsilon}^{\mathbf{M}}) = \bar{\mathbf{u}}(t; \boldsymbol{\epsilon}^{\mathbf{M}}) + \tilde{\mathbf{u}}(t; \boldsymbol{\epsilon}^{\mathbf{M}}) \approx \bar{\mathbf{u}}(t; \boldsymbol{\epsilon}^{\mathbf{M}}) + \sum_{i=1}^{N} \boldsymbol{\phi}_i\, \alpha_i(t; \boldsymbol{\epsilon}^{\mathbf{M}}) = \bar{\mathbf{u}}(t; \boldsymbol{\epsilon}^{\mathbf{M}}) + \boldsymbol{\Phi}\, \boldsymbol{\alpha}(t; \boldsymbol{\epsilon}^{\mathbf{M}}). \tag{11}$$

The degrees of freedom of the surrogate are the components of the vector of generalised coordinates $\boldsymbol{\alpha}$. In the previous equation, operator $\boldsymbol{\Phi}$ is the matrix whose columns are the basis vectors of the reduced space $\mathcal{U}_{\text{MOR}}$. In the following, for the sake of being more general, we will refer to the parameter as μ rather than being an explicit loading path defined by a macro-strain $\boldsymbol{\epsilon}^{\mathbf{M}}$.

Substituting the trial and test vectors of balanced equation (8) by surrogate (11) leads to the Galerkin formulation

$$\forall t, \forall \boldsymbol{\delta\alpha}, \qquad \boldsymbol{\delta\alpha}^T \boldsymbol{\Phi}^T \mathbf{f}_{\text{int}}\left((\bar{\mathbf{u}}(t; \boldsymbol{\epsilon}^{\mathbf{M}}) + \boldsymbol{\Phi}\, \boldsymbol{\alpha}(t; \boldsymbol{\epsilon}^{\mathbf{M}}))\right) = 0. \tag{12}$$

This reduced nonlinear system of equations can be solved using a Newton-Raphson algorithm. At iteration i of this algorithm, we solve the linear system

$$\boldsymbol{\Phi}^T \left(\widetilde{\mathbf{K}}^i \boldsymbol{\Phi}\, \boldsymbol{\Delta\alpha}^{i+1} + \tilde{\mathbf{r}}^i\right) = 0, \tag{13}$$

where $\widetilde{\mathbf{K}}^i = \frac{\partial \mathbf{f}_{\text{int}}}{\partial \mathbf{u}}_{|\bar{\mathbf{u}}+\boldsymbol{\Phi}(x)\,\boldsymbol{\alpha}^i}$ is the tangent operator, $\tilde{\mathbf{r}}^i = \mathbf{f}_{\text{int}}(\bar{\mathbf{u}} + \boldsymbol{\Phi}\, \boldsymbol{\alpha}^i)$ is the residual vector. It is important to recall that although the number of degrees of freedom of this system, N, may be small, the cost of assembling the tangent operators and residuals remains expensive. The reduced model cannot be used "online" in this form, which is why an additional "system approximation" is necessary, which will be detailed in section 3.1.2. For now, we will focus on our first proposition to construct a reduced space using the snapshot-POD methodology.

3.1.1 Snapshot POD

Once the snapshot is computed, an optimization problem can be solved to identify the reduced space that minimises a measure of the projection error of the samples. We define the snapshot matrix $\mathbf{S} = \left[\mathbf{s}_1(t_1)\, \mathbf{s}_1(t_2) \ldots \mathbf{s}_1(t_{n_t})\, \mathbf{s}_2(t_1)\, \ldots\, \mathbf{s}_{n_\mu}(t_{n_t})\right]$, whose columns correspond to the computed samples in various far-field load cases over n_t time steps[6].

---

[4]This is arguable as the gradient-based optimizer will converge to a local minimum in the parameter domain, see [39] for a more detailed discussion and the proposition of a remedy.

[5]We work at a fully discrete level with vectors of degrees of freedom corresponding to continuous fields that belong to FE spaces, but we will refer to such quantities as "fields" or simply "displacements" to avoid unecesseray complication of the explanations.

[6]Note that in practice, n_t is different between different load cases. Here we try to keep simple notations.
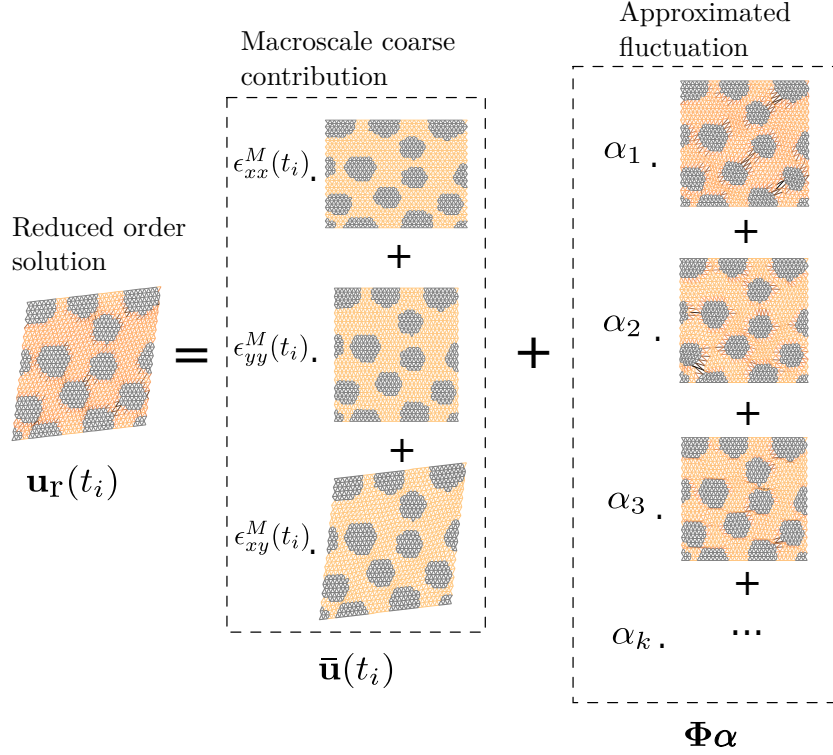
Figure 5: Surrogate model for the displacement field in the RVE. The surrogate is the sum of a macroscopic contribution (known *a priori*) and a fluctuation that is represented as a linear combination of basis vectors and obtained through the ROM.

The POD minimisation problem reads:

$$\begin{cases} \min_{\phi_1,...,\phi_l} J^s_{\langle . \rangle}(\phi_1,...,\phi_l) \\ \langle \phi_i, \phi_j \rangle = \delta_{ij} \, , \end{cases} \tag{14}$$

where the scalar product $\langle . \rangle$ remains to be defined and $\forall \, \mathbf{x}, \|\mathbf{x}\| = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle}$. The cost function is defined as:

$$J^s_{\langle . \rangle}(\phi_1,...,\phi_l) = \sum_{t_j=t_1}^{t_{n_t}} \sum_{i=1}^{n_\mu} \left\| \mathbf{s}_i(t_j) - \sum_{k=1}^{N} \langle \phi_k, \mathbf{s}_i(t_j) \rangle \, \phi_k \right\|^2 . \tag{15}$$

Now, we need to define the scalar product $\langle . \rangle$. The most common choice is the canonical scalar product (i.e. $\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^T \mathbf{y}$) which induces the $\mathcal{L}_2$-norm. In our case, the $\mathcal{L}_2$-norm of the displacement field has little interest. Since we are interested in the energy output of the RVE, we choose a scalar product induced by the initial structure stiffness $\mathbf{K_0}$: $\langle \mathbf{x}, \mathbf{y} \rangle_{\mathbf{K_0}} = \mathbf{x}^T \mathbf{K_0} \mathbf{y}$. This gives a structure specific measure of the displacement quantities. One can then show that solving 14 is equivalent to solve the eigenvalue problem:

$$\mathbf{S}\mathbf{S}^T \mathbf{K_0} \phi_i = \lambda_i \phi_i. \tag{16}$$

This will provide a set of $\mathbf{K_0}$-orthogonal vectors that best represent the snapshot space in terms of elastic energy. We then have the following error (which represents how well the POD basis of order N approximates the snapshot $\mathbf{S}$:

$$\nu_{\text{POD}}(\phi_1,...,\phi_N) = \sqrt{\frac{\sum_{k=N+1}^{n_u} \lambda_k}{\sum_{k=1}^{n_u} \lambda_k}}. \tag{17}$$

10

3.1.2 System Approximation

Constricting the displacement in a low-dimensional space does not provide a significant computational gain, even if the systems to be solved are of smaller dimension. This is because the material of study is nonlinear and history-dependent, and its stiffness varies not only in different areas of the material but also with time. This requires to evaluate the stiffness everywhere in the material and this at each time step of the simulation. This means that the numerical complexity remains despite the simplification on the displacement. Hence, to decrease the numerical complexity, the domain itself needs to be approximated. Several authors have looked into that. Notable contributions include the Hyperreduction method [45], the missing point estimation [44], system approximation [32], discrete empirical interpolation method (DEIM) [51] or more recently the energy-conserving and weighting method [47]. Those methods share the idea that the material properties will be evaluated only at a small set of points or elements within the material domain. They differ in the way of selecting those points and in the treatment of that reduced information. In this paper, we will use the "gappy" method [52], very much like in [32, 51].

Gappy Method

The internal forces generated by the reduced displacement $\mathbf{f}_{\text{int}}(\boldsymbol{\Phi}\,\boldsymbol{\alpha})$[7] will be evaluated only in a small subset of the degrees of freedom $\mathcal{I}$ of the domain Ω. A procedure to select $\mathcal{I}$ will be described later on. All the elements in contact with those degrees of freedom have to be considered. We refer to those as the controlled elements. The internal forces will then be reconstructed by writing the internal forces as a linear combination of a few basis vectors themselves (just like it was made for the displacement).

$$\mathbf{f}_{\text{int}}(\boldsymbol{\Phi}\,\boldsymbol{\alpha}) \approx \sum_{1}^{n_{\text{gap}}} \boldsymbol{\psi}_i \beta_i = \boldsymbol{\Psi}\boldsymbol{\beta}, \tag{18}$$

where $\left[\boldsymbol{\psi_1}, \cdots, \boldsymbol{\psi_{n_{\text{gap}}}}\right] = \boldsymbol{\Psi}$ is the forces basis of size n_{gap} and $\boldsymbol{\beta}$ the associated scalar coefficients.

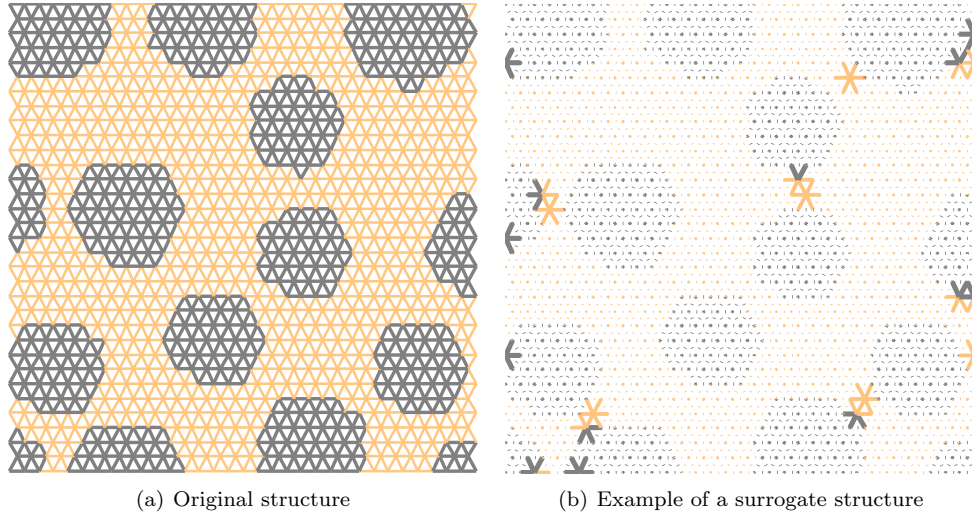(a) Original structure (b) Example of a surrogate structure

Figure 6: Example of a surrogate structure. The stiffness of the structure is evaluated on controlled elements only, while the other ones are just like ghosts

---

[7]To simplify the notations we denote $\mathbf{f}_{\text{int}}(\bar{\mathbf{u}} + \boldsymbol{\Phi}\,\boldsymbol{\alpha})$ by simply $\mathbf{f}_{\text{int}}(\boldsymbol{\Phi}\,\boldsymbol{\alpha})$ in the remaining of this paper.

The coefficients $\boldsymbol{\beta}$ of the expansion are found so that to minimise the norm of the difference between the linear expansion and the nonlinear term over the subset $\mathcal{I}$:

$$\underset{\boldsymbol{\beta}^{\star}}{\operatorname{argmin}} \|\mathbf{f}_{\text{int}}(\boldsymbol{\Phi}\,\boldsymbol{\alpha}) - \boldsymbol{\Psi}\boldsymbol{\beta}^{\star}\|_{\mathbf{P}}, \tag{19}$$

with $\mathbf{P}$ being a matrix so that $\mathbf{P}ij = \begin{cases} 1 & \text{if} \quad i \in \mathcal{I} \quad \text{and} \quad i = j \\ 0 & \text{otherwise} \end{cases}$ and $\|\mathbf{x}\|_{\mathbf{P}} = \|\mathbf{P}^T\mathbf{x}\mathbf{P}\|_2$. $\mathbf{P}$ can be written $\mathbf{E}\mathbf{E}^T$ with $\mathbf{E}$ being an extractor matrix so that $\mathbf{E}^T\mathbf{x}$ is the restriction of $\mathbf{x}$ to the set $\mathcal{I}$. If the number of points in $\mathcal{I}$ is identical to the number of basis vectors $(\boldsymbol{\psi_i})_{i=1,n_{\text{gap}}}$, $\boldsymbol{\beta}^{\star}$ can be found by solving the equation:

$$\mathbf{E}^T\,\boldsymbol{\Psi}\boldsymbol{\beta} = \mathbf{E}^T\,\mathbf{f}_{\text{int}}(\boldsymbol{\Phi}\,\boldsymbol{\alpha}) \tag{20}$$

which implies:

$$\boldsymbol{\beta} = (\mathbf{E}^T\boldsymbol{\Psi})^{-1}\mathbf{E}^T\mathbf{f}_{\text{int}}(\boldsymbol{\Phi}\,\boldsymbol{\alpha}), \tag{21}$$

assuming $\mathbf{E}^T\boldsymbol{\Psi}$ is invertible. This assumption is true when using DEIM, since it insures the linear independence of the restriction of the basis to the reduced integration domain. In other strategies, such as hyperreduction, the size of the reduced integration domain may be chosen larger to guarantee well-posedness of the reduced equations.

At a Newton iteration of our POD-Galerkin framework, this reduces equation (13) to:

$$\boldsymbol{\Phi}^T\boldsymbol{\Psi}(\mathbf{E}^T\boldsymbol{\Psi})^{-1}\mathbf{E}^T\,\widetilde{\mathbf{K}}^i\boldsymbol{\Phi}\boldsymbol{\Delta}\boldsymbol{\alpha} + \boldsymbol{\Phi}^T\boldsymbol{\Psi}(\mathbf{E}^T\boldsymbol{\Psi})^{-1}\mathbf{E}^T\,\widetilde{\mathbf{r}}^i = \mathbf{0}. \tag{22}$$

This can be rewritten in the form:

$$\boldsymbol{\Phi}^T\mathbf{G}\mathbf{E}^T\,\widetilde{\mathbf{K}}^i\boldsymbol{\Phi}\boldsymbol{\Delta}\boldsymbol{\alpha} + \boldsymbol{\Phi}^T\mathbf{G}\,\mathbf{E}^T\,\widetilde{\mathbf{r}}^i = \mathbf{0}, \tag{23}$$

where we define the gappy operator $\mathbf{G} = \boldsymbol{\Psi}(\mathbf{E}\boldsymbol{\Psi})^{-1}$.

Remark: *Note that once the "offline" stage operations are done, the bases $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ are calculated and the set of control points $\mathcal{I}$ is selected and the gappy operator is evaluated. In the "online" stage, all that remains to do is build a system of dimension equal to the size of the displacement basis and solve it which is computationally much cheaper. In particular, the evaluation of $\mathbf{K}$ will be substituted by the evaluation of $\mathbf{E}^T\,\mathbf{K}$, which allows great time savings.*

Selection of the controlled elements

The selection of the control elements will be done using the discrete empirical interpolation method (DEIM) [51]. This method finds a set of degrees of freedom $\mathcal{I}$ in a greedy manner from the internal forces basis $\boldsymbol{\Psi}$. We briefly describe the method.

At iteration j of the greedy algorithm, $j-1$ points have been already selected. We define the extractor $\mathbf{E^j}$ that extracts those j selected degrees of freedom (i.e. for any vector $\mathbf{v}$, $\mathbf{E^j}\mathbf{v}$ is a smaller vector containing only the j entries of $\mathbf{v}$ corresponding to the selected degrees of fredom). The residual $\mathbf{r}_{\text{gap}} = \left|\boldsymbol{\psi_{[1,j]}}\boldsymbol{\beta^j} - \boldsymbol{\psi_{j+1}}\right|$ is evaluated, where $\boldsymbol{\psi_{[1,j]}}$ is the matrix containing the first j vectors of the basis $\boldsymbol{\Psi}$ and $\boldsymbol{\psi_{j+1}}$ is the $j+1^{th}$ vectors in that basis. $\boldsymbol{\beta}$ is the solution of the minimisation problem

$$\boldsymbol{\beta} = \underset{\boldsymbol{\beta}^{\star}}{\operatorname{argmin}} \left\|\mathbf{E^j}\boldsymbol{\psi_{[1,j]}}\boldsymbol{\beta}^{\star} - \mathbf{E^j}\boldsymbol{\psi_{j+1}}\right\|_2. \tag{24}$$

The solution is easily found: $\boldsymbol{\beta} = \left(\mathbf{E^j}\boldsymbol{\psi_{[1,j]}}\right)^{-1}\mathbf{E^j}\boldsymbol{\psi_{j+1}}$. The greedy procedure then selects the index of the highest entry in $\mathbf{r}_{\text{gap}}$ as the $j+1^{th}$ control degree of freedom. This procedure essentially selects the set of degrees of freedom that maximises the conditioning of the system (20). At the end of the greedy algorithm, the number of control degrees of freedom chosen equals the number of basis vectors $(\boldsymbol{\psi_i})_{n_{\text{gap}}}$ which makes system (20) well defined.

3.2 A first "brute force" model reduction approach using snapshot POD on a snapshot randomly generated ensuring dissipation

In this section, we present the construction of a reduced model based on a random selection of the snapshot, constricting the random load paths to dissipate some energy of the structure each timestep. This is done to ensure the variability of the load paths so that maximum knowledge can be gained from the snapshot. In following, we show the method used to approximate the generation of such snapshots.

3.2.1 Random sampling of the parameter domain

To insure load paths that do not "turn back on themselves", we enforce them to dissipate some energy in the structure at each timestep. The idea is that if no energy is dissipated, the structure will deform in an elastic manner, which will not add to the complexity of the snapshot space and will not be informative. We want the snapshot to be as varied as possible so that the reduced basis built from it can be exhaustive (in the sense that it is able to represent any solution resulting from any load path with a controlled error). Note that one could not put any dissipation constraint on the random load paths, but one would have to generate a much larger snapshot for it to statistically extend to the edges of the parameter space. Forcing dissipation saves computational time by computing only the most "informative" solutions.

To generate snapshots following this dissipation property, we will divide the load paths in increments, and enforce that at each increment, the maximum value of load path history is increased in either tension in x, y or shear. This is an approximation, since this is not strictly equivalent to dissipating energy. However, this constraint is explicit, easy to implement, and provides essentially the extended snapshots we are looking for.

In mathematical terms, the parameter space is sampled randomly by iteratively generating random load increments $\widetilde{\boldsymbol{\Delta\epsilon^M}}(t_n) = \begin{bmatrix} \widetilde{\Delta\epsilon_{xx}}(t_n) & \widetilde{\Delta\epsilon_{xy}}(t_n) \\ \widetilde{\Delta\epsilon_{xy}}(t_n) & \widetilde{\Delta\epsilon_{yy}}(t_n) \end{bmatrix}$ of predefined norm Δl. Initialising the loading path to be generated by $\boldsymbol{\epsilon^M}(t_0) = \mathbf{0}$, the path is iteratively incremented as:

$$\boldsymbol{\epsilon^M}(t_{n+1}) = \boldsymbol{\epsilon^M}(t_n) + \widetilde{\boldsymbol{\Delta\epsilon^M}}(t_n), \text{ with } \|\widetilde{\boldsymbol{\Delta\epsilon^M}}(t_n)\| = \Delta l, \tag{25}$$

where $\|\widetilde{\boldsymbol{\Delta\epsilon^M}}(t_n)\| = \sqrt{\widetilde{\Delta\epsilon_{xx}}^2(t_n) + \widetilde{\Delta\epsilon_{yy}}^2(t_n) + \widetilde{\Delta\epsilon_{xy}}^2(t_n)}$. The random load increments $\widetilde{\boldsymbol{\Delta\epsilon^M}}(t_n)$ are forced to create dissipation, by ensuring that at least one of the following inequalities is true at each timestep:

$$\langle\widetilde{\Delta\epsilon_{xx}}(t_n)\rangle^+ > \max_{k \in [\![0,n-1]\!]} \widetilde{\Delta\epsilon_{xx}}(t_k) \tag{26}$$

$$\langle\widetilde{\Delta\epsilon_{yy}}(t_n)\rangle^+ > \max_{k \in [\![0,n-1]\!]} \widetilde{\Delta\epsilon_{yy}}(t_k) \tag{27}$$

$$|\widetilde{\Delta\epsilon_{xy}}(t_n)| > \max_{k \in [\![0,n-1]\!]} |\widetilde{\Delta\epsilon_{xy}}(t_k)|, \tag{28}$$

where $\langle x\rangle^+$ is the positive part of x. These conditions mean that either the tension in x direction, in y direction or shear has to increase at each timestep. When no dissipation is created, the damage law behave essentially linearly and do not add to the complexity of the snapshot space. An example of a few loading paths generated using this method is displayed in Figure 7. The randomness of this procedure will allow to explore the parameter space exhaustively, as long as the number of paths generated is large enough.

To improve the reliability of the procedure, the quality of the ROM could be tested by evaluating the error over some random validation set Ξ_{test} different from the snapshot set Ξ. If the average error over that set is larger than some tolerance, the initial snapshot could be enriched iteratively until that tolerance is achieved. This is described in pseudo-code in algorithm 2. This strategy will not be developed further in this paper.
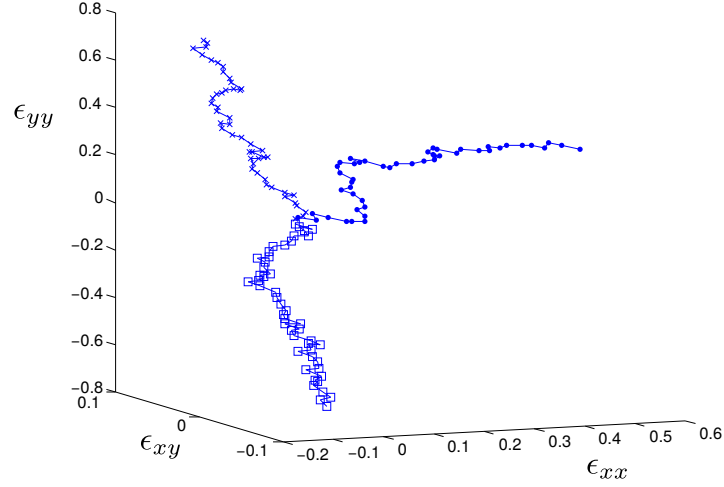
13

Figure 7: Example of loading paths obtained using the random procedure.

Algorithm 1 Example of a "Brute force" sampling method: *the parameter space is randomly sampled with a high number of values and the reduced-order model is built upon it. The random set is enriched until reaching some convergence.*

1: Input target tolerance ν
2: Initialise N
3: $err_{\mathrm{snap}} \leftarrow \infty$
4: **while** $err_{\mathrm{snap}} > \nu$ **do**
5:     Generate N new random "dissipating" snapshots
6:     Build new reduced-order model to achieve tolerance ν
7:     Evaluate err_{snap} by computing the average value of $\mathcal{Q}^{\mathrm{HR}}$ (defined in equation (29)) over some validation set Ξ_{test}

3.2.2 Application of the random snapshot-POD procedure and numerical findings

Displacement basis. We proceed to apply the snapshot-POD procedure with random snapshot selection described in section 3.2. 36 load paths are randomly generated. The first few vectors of the POD expansion are displayed in Figure 8.
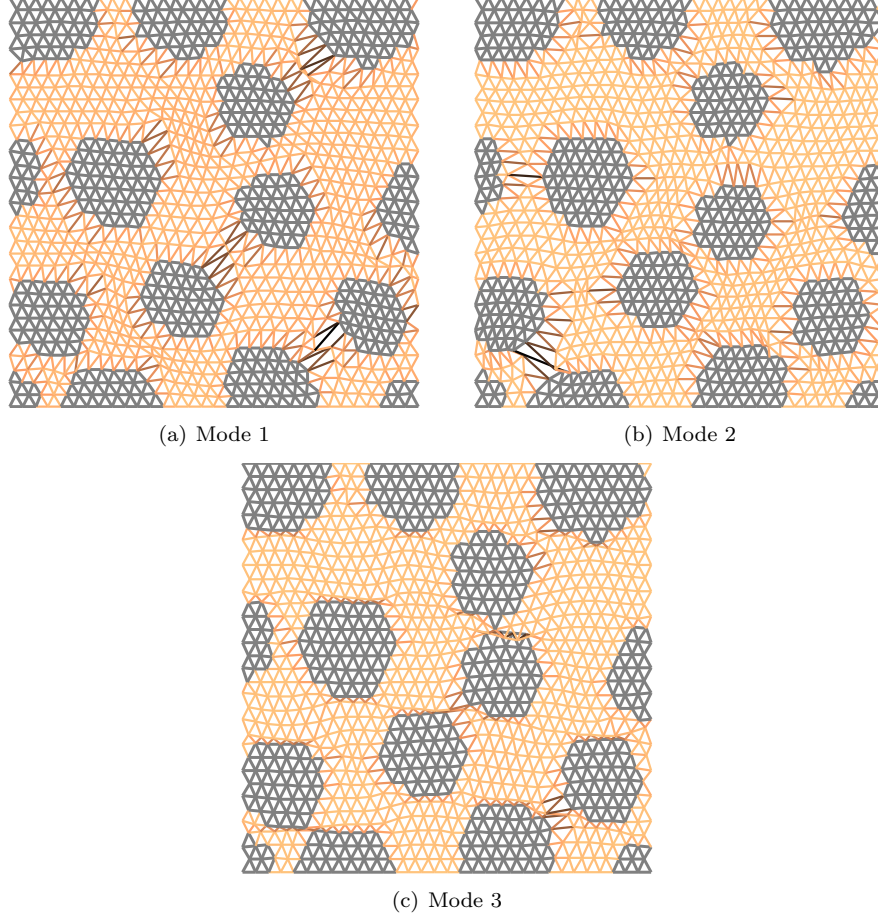

(a) Mode 1

(b) Mode 2

(c) Mode 3

Figure 8: Vectors associated to the 3 largest eigenvalues obtained using the snapshot-POD procedure with random snapshot selection. Darker bars indicates larger damage. The damage localises between pairs of inclusions.

System approximation. We follow the procedure described in 3.1.2. The basis $\boldsymbol{\Psi}$ is extracted from the snapshot space generated by the same loading paths used for the displacement basis $\boldsymbol{\Phi}$. The set of controlled elements is selected using the DEIM [51]. The amount of vectors in the basis $\boldsymbol{\Psi}$ is chosen so that the error generated by the system approximation is of the same order than the global error of the reduced-order model.

To this purpose we define the quantity of interest $\mathcal{Q}$ as the norm of the error. More specifically, we denote $\mathcal{Q}^{\mathrm{R}}$ the average norm over time of the error between exact and reduced-order solution using no hyperreduction and $\mathcal{Q}^{\mathrm{HR}}$ the average norm over time of the error between exact and "hyperreduced-order solution" actually using the hyperreduction:

$$\mathcal{Q}^{\mathrm{R}}(\mu)^2 = \sum_{t=t_0}^{t_{n_t}} \frac{\left\| \mathbf{u}(\mu,t) - \mathbf{u}^{\mathrm{R}}(\mu,t) \right\|_{\mathbf{K_0}}^2}{n_t+1} \quad \text{and} \quad \mathcal{Q}^{\mathrm{HR}}(\mu)^2 = \sum_{t=t_0}^{t_{n_t}} \frac{\left\| \mathbf{u}(\mu,t) - \mathbf{u}^{\mathrm{HR}}(\mu,t) \right\|_{\mathbf{K_0}}^2}{n_t+1}, \qquad (29)$$

15

with $\mathbf{u}(t)$ the exact solution, $\mathbf{u}^{\mathrm{R}}(\mu, t; \boldsymbol{\Phi})$, the reduced-order solution without the system approximation using the displacement basis $\boldsymbol{\Phi}$, and $\mathbf{u}^{\mathrm{HR}}(\mu, t; \boldsymbol{\Phi}, \boldsymbol{\Psi})$ the complete reduced-order model with system approximation using the displacement basis $\boldsymbol{\Phi}$ and the static basis $\boldsymbol{\Psi}$. Note that we skip the dependency of the solution on the bases $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ in the following for simplicity of the notations. $\mathcal{Q}^{\mathrm{HR}}(\mu)^2$ can then be decomposed in the following way:

$$\mathcal{Q}^{\mathrm{HR}}(\mu)^2 = \sum_{t=t_0}^{t_{n_t}} \frac{\left\| \mathbf{u}(\mu, t) - \mathbf{u}^{\mathrm{R}}(\mu, t) + \mathbf{u}^{\mathrm{R}}(\mu, t) - \mathbf{u}^{\mathrm{HR}}(\mu, t) \right\|_{\mathbf{K_0}}^2}{n_t + 1} \tag{30}$$

$$\leq \mathcal{Q}^{\mathrm{R}}(\mu)^2 + \underbrace{\sum_{t=t_0}^{t_{n_t}} \frac{\left\| \mathbf{u}^{\mathrm{R}}(\mu, t) - \mathbf{u}^{\mathrm{HR}}(\mu, t) \right\|_{\mathbf{K_0}}^2}{n_t + 1}}_{\overset{def}{=} \widetilde{\mathcal{Q}^{\mathrm{HR}}(\mu)}^2} . \tag{31}$$

Taking this in consideration, the basis $\boldsymbol{\Psi}$ is chosen to be the smallest (i.e. the one with the least amount of vectors) that verifies the inequality:

$$\widetilde{\mathcal{Q}^{\mathrm{HR}}(\mu)} \leq \mathcal{Q}^{\mathrm{R}}(\mu) \tag{32}$$

This guarantees that the error generated by the system approximation is controlled by the error generated by approximating the displacement. The location of controlled elements (which are all the elements in contact with the control degrees of freedom) is shown in Figure 9 for various basis sizes. It is interesting to remark that the controlled elements gather around inclusions where damage is the highest. Figure 10 illustrate this effect.

Remark: *Note that in equation (31), we defined the quantity $\widetilde{\mathcal{Q}^{HR}(\mu)}$ which defines the error between the reduced and the hyperreduced model which is different from $\mathcal{Q}^{HR}(\mu)$, which defines the error between the exact solution and the hyperreduced model.*
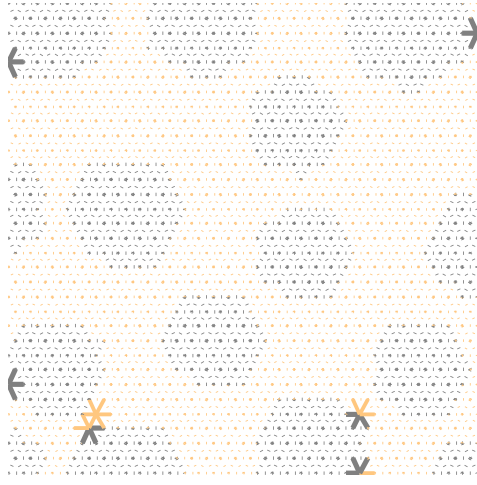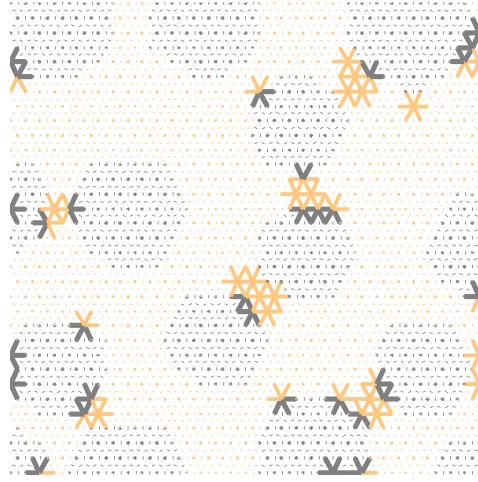
(a) Controlled elements with 3 displacement basis vectors. 10 "static" basis vectors are needed to achieve the minimal accuracy condition (32)
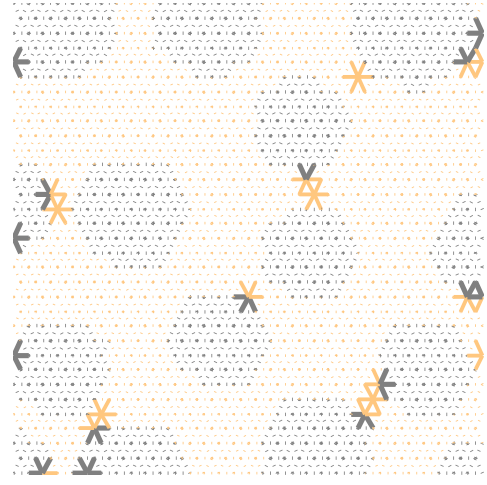
(b) Controlled elements with 5 displacement basis vectors. 28 "static" basis vectors are needed to achieve the minimal accuracy condition (32)

(c) Controlled elements with 15 "dynamic" basis vectors. 60 "static" basis vectors are needed to achieve the minimal accuracy condition (32)

Figure 9: Controlled elements selected using various basis sizes. The larger the basis, the more controlled elements are needed. The elements tend to gather around the regions where the variation of displacement is the highest, hence where the variation of the internal forces will be high.
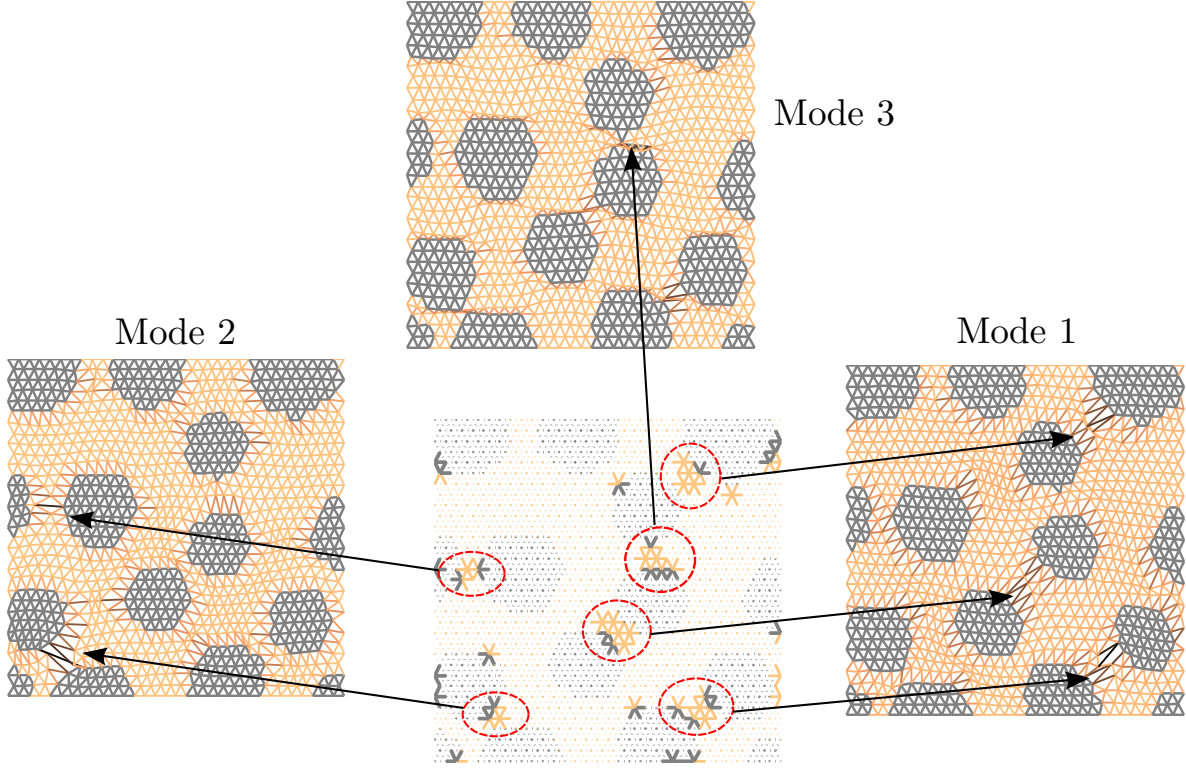
Figure 10: Regions of interest selected by the system approximation procedure. Those regions (circled in the Figure) are matching the areas of higher displacement found in the POD bases. This is intuitively good, since those elements have to give enough information to be able to reconstruct the internal forces over the entire domain. Those are the elements whose behaviour vary the most when changing the loading path (which is the parameter of the reduced model), hence containing the core information necessary to build up an accurate reconstruction.

Numerical savings   In this section, we will test the performance of the method by comparing the relative error between the "truth" solution of the RVE problem, which is the solution obtained when using the full order model, and the reduced-order model.

The following load path considered for testing the efficiency of the model is set using the following effective strain: $\boldsymbol{\epsilon}^{\mathbf{M}}(t) = \frac{t}{T} \cdot \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$. Note that this case is not in the snapshot set.

We then proceed to solve the RVE boundary value problem subjected to this loading path using both the full order model and the reduced-order model while varying the sizes of the displacement and static bases. Induced errors and times gained are displayed in Figure 11.

(a) Evolution of the error varying the number of displacement and static basis vectors.

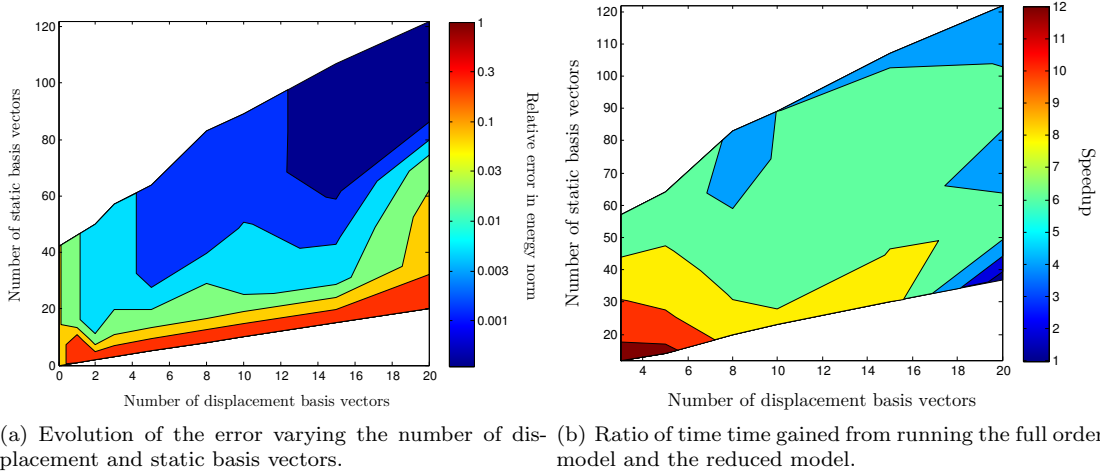(b) Ratio of time time gained from running the full order model and the reduced model.

Figure 11: Numerical results tested on a loading path not included in the snapshot space. Here, the snapshot selection was arbitrary and relatively fine which allows to consider various number of basis sizes. Note that one could use these plots as a way to determine the sizes of the bases to maximise the speedup for a specific target error.

Several remarks can be made:

- As expected, the error decreases when the number of either the displacement or static bases vectors increases. A higher dimensional representation of the solution leads unsurprisingly to more accuracy.

- The time gained using the reduced model becomes more and more important when the number of vectors in the bases decreases.

- Looking at Figure 11(b), it can be seen that the speedup is roughly dependent on the size of the static bases, rather than on the displacement basis. Indeed, the number of controlled elements, which is linked to the amount of computations to be done, is directly linked to the dimension of the static basis $\boldsymbol{\Psi}$.

- To have a well defined reduced-order model, the dimension of the static basis $\boldsymbol{\Psi}$ should at least match the dimension of the displacement basis $\boldsymbol{\Phi}$. However, it can be seen that to achieve a reasonable tolerance on the error, the dimension of the static basis should actually be relatively larger.

The error with respect to the speedup for a range of reduced space sizes is displayed in Figure 12. What we call speedup here is the ratio of the elapsed time of the full order simulation over the elapsed time of the reduced model. It represents how many times faster is the reduced-order model compared to the full order model.
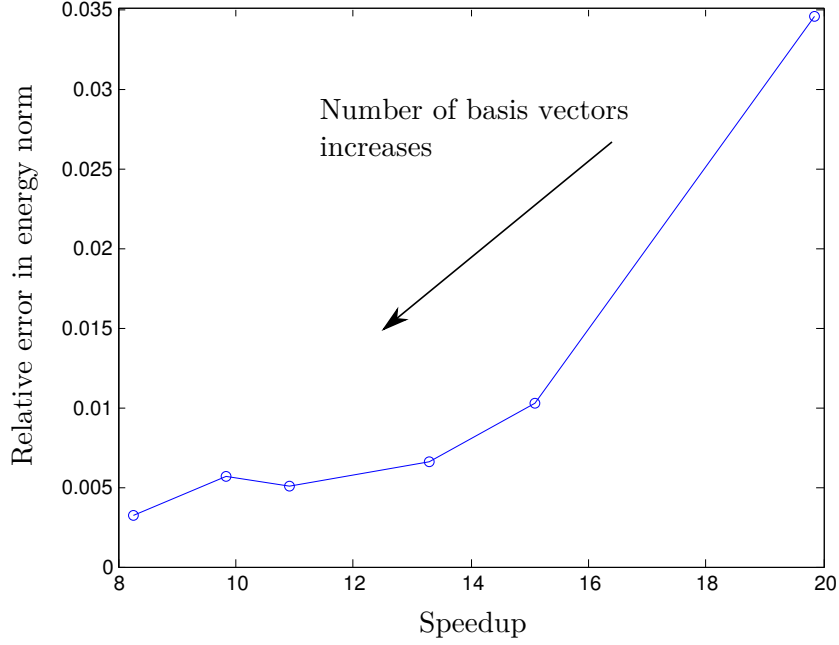
19

Figure 12: Evolution of the error with respect to the speedup while increasing the number of basis vectors. For each size of the displacement basis $\mathbf{\Phi}$, the number of vectors in the static basis $\mathbf{\Psi}$ is chosen according to the rule defined in equation (32).

It can be seen that there is a proportional relation between speedup and error: as the number of basis functions increases, the speedup and the error decrease. The user can reduce the error at the price of having a slower simulation. What makes the reduced model faster is purely the bypassing of most of the elements when computing the internal forces or the tangent stiffness (this bypassing is possible thanks to the system approximation technique). Note that the speedup is not purely equal to the ratio between controlled elements and total number of elements since the Newton-Raphson procedure requires more steps to converge in the reduced-order model scheme than in the full order model. Another remark is that beyond a certain dimension of the reduced space, the error does not decrease very much and reaches a plateau. This means that no matter how many vectors in the basis, a maximum accuracy is achieved. This can be explained by the fact that the loading path tested is not part of the snapshot. The only way to decrease this residual error is to enrich the snapshot space. Let us define $\mathbf{u}_{\mathrm{snap}}(t)$ as the projection of the exact solution onto the snapshot space. Using the same principle than equation (31), we can decompose the error further (dropping parameters for clarity):

$$\mathcal{Q}^{\mathrm{HR}^2} = \sum_{t=t_0}^{t_{n_t}} \frac{\left\| \mathbf{u}(t) - \mathbf{u}_{\mathrm{snap}}(t) + \mathbf{u}_{\mathrm{snap}}(t) - \mathbf{u}^{\mathrm{R}}(t) + \mathbf{u}^{\mathrm{R}}(t) - \mathbf{u}^{\mathrm{HR}}(t) \right\|_{\mathbf{K_0}}^2}{n_t + 1} \tag{33}$$

$$\leq \sum_{t=t_0}^{t_{n_t}} \frac{\left\| \mathbf{u}(t) - \mathbf{u}_{\mathrm{snap}}(t) \right\|_{\mathbf{K_0}}^2}{n_t + 1} + \sum_{t=t_0}^{t_{n_t}} \frac{\left\| \mathbf{u}_{\mathrm{snap}}(t) - \mathbf{u}^{\mathrm{R}}(t) \right\|_{\mathbf{K_0}}^2}{n_t + 1} + \widetilde{\mathcal{Q}^{\mathrm{HR}}}^2. \tag{34}$$

$\sum_{t=t_0}^{t_{n_t}} \frac{\left\| \mathbf{u}_{\mathrm{snap}}(t) - \mathbf{u}^{\mathrm{R}}(t) \right\|_{\mathbf{K_0}}^2}{n_t + 1}$ and $\widetilde{\mathcal{Q}^{\mathrm{HR}}}^2$ can be made as small as desired by taking high dimensional bases $\mathbf{\Phi}$ and $\mathbf{\Psi}$. The residual error that remains is $\sum_{t=t_0}^{t_{n_t}} \frac{\left\| \mathbf{u}(t) - \mathbf{u}_{\mathrm{snap}}(t) \right\|_{\mathbf{K_0}}^2}{n_t + 1}$, which entirely depends upon the richness of the snapshot space.

We will deal with this issue in the next section by using a Bayesian-optimized snapshot selection which will allow to guess the error between the discrete solutions computed for the snapshot set.

20

3.3 Model reduction using a POD-greedy algorithm based on a Bayesian-optimized snapshot selection designed for dealing with high-dimensional parameter spaces

As said in the previous section, it may not be satisfactory to use an arbitrary sampling method, since some important information could be unwittingly dropped out. The accuracy of the reduced model greatly depends on the snapshot space and how well it samples the parameter space. Here, the parameter space contains any load path (based on the macro-strain $\epsilon^M(t)$) over a certain period of time until ellipticity of the mechanical problem is lost. After time discretisation, the parameter space is of dimension $2 \times n_t$, since in 2 dimensions the load can be uniaxial in the x or y direction or in shear, and we set a fixed load increment norm between two timesteps. n_t stands for the number of time steps required to reach fracture.

This section attempts to address the problem of ensuring that the snapshot space is sufficiently fine so that a reduced model of sufficient accuracy can be built upon it. Given the high dimension of the parameter space, its effective sampling is based on a combination of three necessary cost-effective strategies:

- First, the high-dimensional parameter space $\mathcal{P}$ is restricted to a hierarchical sequence of much lower dimensional pseudo-parameter spaces $\widehat{\mathcal{P}}^n$ which enable to avoid the "curse of dimensionality". Starting from a pseudo-parameter space $\widehat{\mathcal{P}}^0$ containing proportional loadings only, it is iteratively refined until reaching some "convergence". This approach is described in section 3.3.1.

- Second, within each pseudo-parameter spaces $\widehat{\mathcal{P}}^n$, rather than a random and fine sampling typically used in traditional POD-greedy approaches, an effective selection procedure allowing few evaluations of an error indicator is done using a Gaussian process predictor. This strategy is explained in section 3.3.2.

- A statistical correspondence between the error indicator and the true error is built using Gaussian process regression to control the convergence of the procedure. This is described in section 3.3.3.

3.3.1 Definition of a sequence of surrogate parameter spaces of low dimension

We would like to sample the parameter space exhaustively. To this purpose, the use of Gaussian process methods [42, 53] are attractive as they provide an excellent predictor as well as statistical information that allows to trust the model. However, it is almost inapplicable in a high-dimensional context, as it requires a decent amount of data in proportion with the dimensionality of the phenomenon to study. To circumvent the curse of dimensionality, we propose to define a sequence of low-dimensional surrogate parameter spaces of progressively finer dimension $\widehat{\mathcal{P}}^i$. The initial surrogate space $\widehat{\mathcal{P}}^0$ contains all proportional loadings only (i.e. the loading path is a straight line). This space only is of dimension 2 (well-defined by two angles when considering spherical coordinates). The surrogate space $\widehat{\mathcal{P}}^1$ is defined as 2 successive proportional loadings with different directions (Figure 13). This space has 5 dimensions since 2 dimensions can be counted for each proportional part of the loading plus one more dimension for the length of the load at which the second proportional loading starts. This sequence can carry on with space n, the space of loads with $n+1$ distinct proportional loadings, which has dimension $2 \times n + (n-1)$. We also have the property that surrogate space n is included in surrogate space $n+1$:

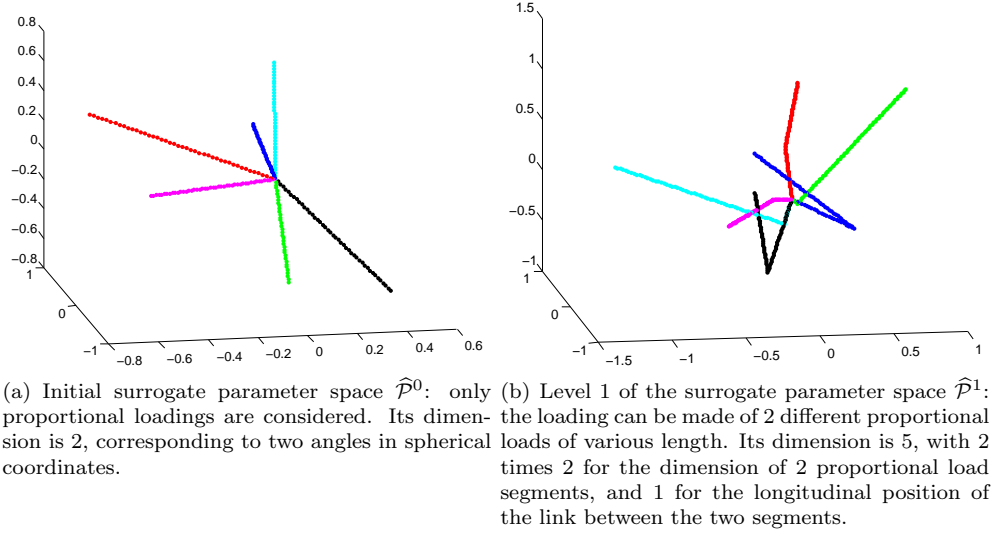$$\forall n \in \mathbb{N}, \ \widehat{\mathcal{P}}^n \subset \widehat{\mathcal{P}}^{n+1}. \tag{35}$$

(a) Initial surrogate parameter space $\widehat{\mathcal{P}}^0$: only proportional loadings are considered. Its dimension is 2, corresponding to two angles in spherical coordinates.

(b) Level 1 of the surrogate parameter space $\widehat{\mathcal{P}}^1$: the loading can be made of 2 different proportional loads of various length. Its dimension is 5, with 2 times 2 for the dimension of 2 proportional load segments, and 1 for the longitudinal position of the link between the two segments.

Figure 13: Examples of loadings paths for level 0 and 1 of the surrogate parameter spaces.

Now, with such an inclusive decomposition of the parameter space, it becomes possible to infer what level of refinement is necessary to consider, for building an accurate reduced model. Indeed, a reduced-order model can be constructed based on snapshots from surrogate parameter space $\widehat{\mathcal{P}}^n$, and if it represents well any solution in parameter space $\widehat{\mathcal{P}}^{n+1}$ (which is a space with a finer discretisation of the loading paths $\boldsymbol{\epsilon}^M$), we can assume that the current reduced model is satisfactory and there is no need to consider finer parameter spaces. The procedure is described in Algorithm 2.

Algorithm 2 Main loop of the procedure

1: Define a tolerance target ϵ for the reduced-order model
2: $n \leftarrow 0$
3: Initialise $\boldsymbol{\Phi}^0$ to the empty matrix
4: **while** no convergence **do**
5:     Update the reduced-order model basis $\boldsymbol{\Phi}^n$ with pseudo parameter space $\widehat{\mathcal{P}}^n$ to achieve tolerance ϵ following Algorithm 4
6:     IF convergence
7:         Break;
8:     $n \leftarrow n + 1$

3.3.2 Exhaustive sampling of the surrogate parameter spaces using a Gaussian process predictor

In this section, given a dimension for the surrogate parameter space, we are looking for the value of the parameter leading to the highest error between the exact solution and the solution computed using our reduced model.

Standard POD-greedy procedure. In traditional POD-greedy strategies [40], an a posteriori error bound $\Delta^k(\mu)$ inexpensive to compute is assumed available, which allows for the estimation of the error between the full order model and the reduced-order model at step k of the procedure, on a fine discretisation $\Xi \in \mathcal{P}$ of the parameter space. At step k of the procedure, the full order model is

evaluated at the parameter value $\mu_{\max}^k$ satisfying:

$$\mu_{\max}^k = \underset{\mu \in \Xi}{\operatorname{argmax}} \Delta^k(\mu). \tag{36}$$

The reduced-order model is updated with this new information and the algorithm carries on until reaching some tolerance.

In practice, error bounds $\Delta^k(\mu)$ are available for linear problems. In the general nonlinear case, no sharp error bound is available, and one has to rely on an error indicator at the parameter value μ instead: $\mathcal{J}(\mu)$. This error indicator, does not provide a bound on the error, but rather a measure of its magnitude. Though less expensive than computing the exact solution at μ, we will see in the next section that the evaluation of this indicator at all values of a fine discretisation of the parameter space is not affordable. To alleviate this issue, the error indicator surface over the parameter space will be approximated using a Gaussian process predictor to allow for only a few, well chosen, evaluations of the indicator.

Definition of an error indicator based on the residual. In our case the quantity of interest is the error $\mathcal{Q}^{\mathrm{HR}}$ defined in equation (29). Having no rigorous error bound at hand, we will use an error indicator instead. The norm of the residual $\mathbf{r}\,(t_k, \mu; \boldsymbol{\Phi}, \boldsymbol{\Psi})$ can be used as such an indicator at each timestep (as described in [43, 54]). Here the residual at timestep t_k is defined as:

$$\mathbf{r}\,(t_k, \mu; \boldsymbol{\Phi}, \boldsymbol{\Psi}) = \mathbf{f}_{\mathrm{int}}\left((\bar{\mathbf{u}}(t_k; \mu) + \boldsymbol{\Phi}\,\boldsymbol{\alpha}(t_k; \mu))\right), \tag{37}$$

where $\boldsymbol{\alpha}(t_k; \mu)$ is the converged solution at $t = t_k$ obtained from the Newton procedure described in equation (23).

Since we would like an error indicator that is taking into account the entire time-history of the solution, we define a time-independent norm of the residual for the complete reduced-order model (i.e. including both approximation of the displacement in a low-dimensional space and approximation of the internal forces):

$$\mathcal{R}\,(\mu; \boldsymbol{\Phi}, \boldsymbol{\Psi}) = \sqrt{\frac{\sum_{k=0}^{k=N} \|\mathbf{r}\,(t_k, \mu; \boldsymbol{\Phi}, \boldsymbol{\Psi})\|^2}{N+1}}. \tag{38}$$

<u>Remark:</u> *Note that the residual $\mathcal{R}$ will almost always not be null. Indeed, what is solved in the reduced model leads to the cancellation, at each timestep, of the projected residual:*
$\boldsymbol{\Phi}^T \mathbf{G} \mathbf{E}^T \mathbf{f}_{int}\left((\bar{\mathbf{u}}(t_k; \boldsymbol{\epsilon}^{\mathbf{M}}) + \boldsymbol{\Phi}\,\boldsymbol{\alpha}(t_k; \boldsymbol{\epsilon}^{\mathbf{M}}))\right)$*, using Newton iterations.*

Now, since the residual is defined on the fine discretisation of the domain, the cost of its evaluation presents still a significant cost. This can be partly alleviated by evaluating a surrogate $\widetilde{\mathcal{R}}$ of the global residual $\mathcal{R}$ over only a subset of the timesteps. We then define the error indicator:

$$\mathcal{J}\,(\mu; \boldsymbol{\Phi}, \boldsymbol{\Psi}) = \widetilde{\mathcal{R}}\,(\mu; \boldsymbol{\Phi}, \boldsymbol{\Psi}) = \sqrt{\frac{\sum_{t_k \in \widetilde{\mathcal{T}}} \|\mathbf{r}\,(t_k, \mu; \boldsymbol{\Phi}, \boldsymbol{\Psi})\|^2}{|\widetilde{\mathcal{T}}|}}, \tag{39}$$

where $\widetilde{\mathcal{T}}$ is a subset of the time discretisation used to compute the simulation; for example $\widetilde{\mathcal{T}}$ may contain only one in every five timesteps. In section 3.3.4, we may refer to $\mathcal{J}\,(\mu; \boldsymbol{\Phi}, \boldsymbol{\Psi})$ as $\mathcal{J}^{\mathrm{R}}$ or $\mathcal{J}^{\mathrm{HR}}$ depending on if we are considering the residual of the ROM without or with hyperreduction respectively. The main advantage is that it can provide an indicator of the magnitude of the error for various values of the parameter μ (which in our problem is the far-field strain $\boldsymbol{\epsilon}^{\mathbf{M}}$) at a much cheaper cost that having to evaluate both the exact and reduced solution. Nevertheless $\mathcal{J}$ remains a non-negligible quantity to compute and we show in the next paragraph how to exhaustively explore the parameter space despite a limitation on the number of evaluation of this error indicator and hence of that residual.

Gaussian process regression of the residual surface for efficient evaluation of the error. In traditional POD-greedy procedures, a discrete set $\Xi \in \mathcal{P}^n$ is built arbitrarily to sample the parameter space. It is typically very fine. The goal of this section is to define a set Ξ that is of relatively small cardinality but is chosen so that it is likely to contain the values of the parameter leading to the highest error. To this purpose, we follow a procedure similar to the one described in [42, 43].

The first ingredient is Gaussian process regression [53] (also called kriging in the literature): starting from an initial set Ξ_0 chosen randomly containing few values of the parameter and an associated set of values of the error indicator $\{\mathcal{J}(\mu_i) \,|\, \mu_i \in \Xi_0\}$, a Gaussian process regression approximating the error indicator $\mathcal{J}$ with a confidence interval over the entire parameter domain $\mathcal{P}^n$, will be constructed for each step m of the sampling process. This regression will be used to iteratively enrich Ξ_m with values of the parameter $\boldsymbol{\mu}_m$ where the probability of having large values of the error indicator $\mathcal{J}$ is the highest.

The method is based on the assumption that the data studied is following a joint Gaussian distribution defined by a mean $\bar{\mathbf{m}}$, which can be unknown, and a covariance matrix, whose shape is defined a priori by the user. A common covariance function is the squared exponential:

$$cov(\mathbf{x^p}, \mathbf{x^q}) = \sigma^2 \exp((\mathbf{x^p} - \mathbf{x^q})^T \mathbf{I_\theta}(\mathbf{x^p} - \mathbf{x^q})) \tag{40}$$

with $\mathbf{I_\theta}$ being a diagonal matrix with diagonal element θ_i on the ith row. For the observations at parameter values Ξ (of cardinality N), $\mathcal{J}$ follows the joint Gaussian distribution of mean $\bar{\mathbf{m}}$ and covariance $\mathbf{Cov}(\Xi, \Xi)$:

$$\mathcal{J}_{\bar{\mathbf{m}}, \sigma^2, \boldsymbol{\theta}}(\mathbf{x}) \sim \frac{1}{(2\pi)^{N/2}|\mathbf{Cov}(\Xi, \Xi)|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\bar{\mathbf{m}})(\mathbf{Cov}(\Xi, \Xi))^{-1}(\mathbf{x}-\bar{\mathbf{m}})}, \tag{41}$$

or more simply written:

$$\mathcal{J} \sim \mathcal{N}(\bar{\mathbf{m}}, \mathbf{Cov}(\Xi, \Xi)), \tag{42}$$

where the element in i^{th} row and j^{th} column of $\mathbf{Cov}(\Xi, \Xi)$ is $\{\mathbf{Cov}_{ij}\} = cov(\mathbf{x^i}, \mathbf{x^j})$ with $\mathbf{x^i}, \mathbf{x^j} \in \Xi$. $\bar{\mathbf{m}}$, $\boldsymbol{\theta}$ and σ^2 are hyperparameters that need to be determined. Assuming the knowledge of the data Ξ, it is done through the maximisation of the likelihood function:

$$\mathcal{L}_\Xi(\bar{\mathbf{m}}, \sigma^2, \boldsymbol{\theta}) = \frac{1}{(2\pi)^{N/2}|\mathbf{Cov}(\Xi, \Xi)|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\bar{\mathbf{m}})(\mathbf{Cov}(\Xi, \Xi))^{-1}(\mathbf{x}-\bar{\mathbf{m}})}. \tag{43}$$

This maximization aims to make the Gaussian process distribution we are constructing as consistent as possible with the data at hand. Poorly chosen values of the hyperparameters will lead to poor predictions. This is typically the case with little data at hand but improves as more and more samples are computed. See [53] or [42] for more details on the determination of the hyperparameters.

Then, adding test inputs, we have (with the input parameters Ξ carrying a subscript $\star$ being the test inputs, the other ones being the training inputs):

$$\begin{bmatrix} \mathcal{J} \\ \mathcal{J}_\star \end{bmatrix} \sim \mathcal{N}\left(\bar{m}, \begin{bmatrix} \mathbf{Cov}(\Xi, \Xi) & \mathbf{Cov}(\Xi, \Xi_\star) \\ \mathbf{Cov}(\Xi_\star, \Xi) & \mathbf{Cov}(\Xi_\star, \Xi_\star) \end{bmatrix}\right). \tag{44}$$

This distribution makes no use of the data we have at end. To obtain the posterior distribution which actually uses the knowledge of the data points, one can condition the prior distribution to the observations and obtain the distribution:

$$\mathcal{J}_\star | \Xi_\star, \Xi, \mathcal{J} \sim \mathcal{N}\left(\bar{m} + \mathbf{Cov}(\Xi_\star, \Xi)\mathbf{Cov}(\Xi, \Xi)^{-1}(\mathcal{J} - \bar{m}),\right. \tag{45}$$

$$\left.\mathbf{Cov}(\Xi_\star, \Xi_\star) - \mathbf{Cov}(\Xi_\star, \Xi)\mathbf{Cov}(\Xi, \Xi)^{-1}\mathbf{Cov}(\Xi, \Xi_\star)\right). \tag{46}$$

From this expression one can deduce a predictor at any parameter value μ by taking the mean value,

$$\mathcal{J}_\star^m(\mu) = \bar{m} + \mathbf{Cov}(\{\mu\}, \Xi)\mathbf{Cov}(\Xi, \Xi)^{-1}(\mathcal{J} - \bar{m}), \tag{47}$$

together with a standard error, using the covariance:

$$s(\mu)^2 = \mathbf{Cov}(\{\mu\}, \{\mu\}) - \mathbf{Cov}(\{\mu\}, \Xi)\mathbf{Cov}(\Xi, \Xi)^{-1}\mathbf{Cov}(\Xi, \{\mu\}). \tag{48}$$

Using this regression at step m of the sampling process, the maximum value $\mathcal{J}^m_{\star \max}$ of the current Gaussian predictor $\mathcal{J}^m_\star$ over the parameter domain is computed. Rather than considering the parameter value achieving this maximum as a new test point, the procedure searches for the parameter value that has the highest probability of improving that maximum value by some predefined percentage, which defines a target value $T(\mathcal{J}^m_{\star \max})$. Indeed, this allows to take into account the trade-off between maximal value of the predictor and uncertainty characterized by the standard error. Since $\mathcal{J}_\star(\mu)$ has a normal distribution with mean $\mathcal{J}^m_\star(\mu)$ and standard error $s(\mu)$, the probability of improvement of $\mathcal{J}_\star(\mu)$ beyond the target $T(\mathcal{J}^m_{\star \max})$ is:

$$\pi(\boldsymbol{\epsilon}^{\boldsymbol{M}}; T) = \varphi\left(\frac{\mathcal{J}^m_\star(\mu) - T(\mathcal{J}_{\star \max})}{s(\mu)}\right), \tag{49}$$

where φ is the normal cumulative distribution function. The parameter μ_m maximising this probability is then added to the set Ξ_{m-1}, creating the new set Ξ_m that will be used to build the Gaussian predictor for the next step (with new values of the hyperparameters $\bar{\mathbf{m}}, \sigma, \boldsymbol{\theta}$):

$$\Xi_m = \Xi_0 \cup \{\boldsymbol{\mu_1}\} \cup \{\boldsymbol{\mu_2}\} \cup \ldots \cup \{\boldsymbol{\mu_m}\} \tag{50}$$

The procedure stops at some step M (which can either be defined arbitrarily or by some tolerance on the value of the error indicator , and the value of the parameter that maximises the error indicator over the set Ξ_M is then selected as $\mu^k_{\max}$ (defined in equation (36)), where the exact solution will be computed:

$$\mu^k_{\max} = \operatorname*{argmax}_{\mu \in \Xi_m} \mathcal{J}^m_\star(\mu) \tag{51}$$

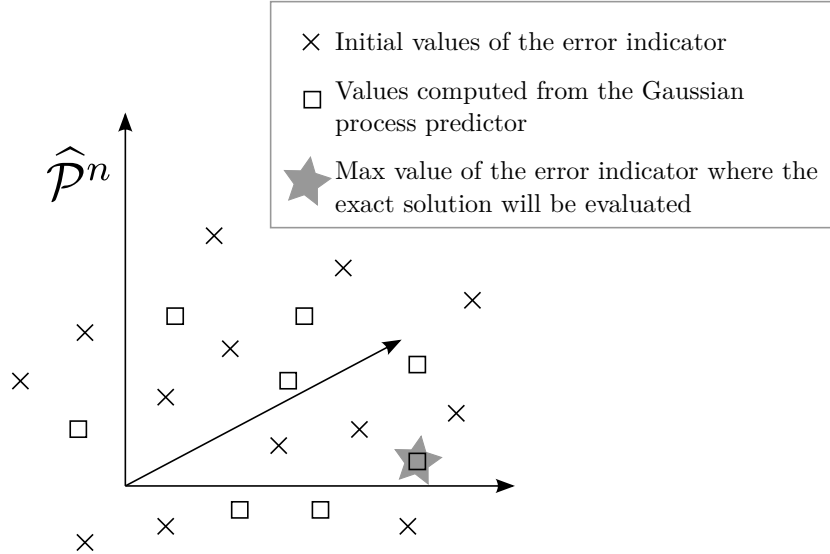This is graphically sketched in Figure 14. The process is also described in Algorithm 3.

Figure 14: Computing the error indicator $\mathcal{J}$ for an initial arbitrary selection of parameters (denoted by the crosses), a Gaussian regression is iteratively computed to evaluate the indicator at locations where it is likely to be the highest (shown as the squares). Eventually, the parameter value selected is the one where the indicator indeed is the highest among this discrete set (shown as the star).

Algorithm 3 Gaussian process sampling algorithm to find parameter value leading to the maximum value of the error indicator

1: Input: Current ROM bases, $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$.
2: Output: Load $\mu_{\max}^k$ with maximum value of the error indicator
3: Define a subset of random loading values Ξ_0
4: $\Xi_{\mathrm{curr}} \leftarrow \Xi_0$
5: $m \leftarrow 0$
6: **while** $m < n_{\mathrm{steps}}$ **do**
7:     Evaluate estimator for each parameter value $\mu \in \Xi_m$
8:     Build kriging response surface and find parameter μ_m of maximum probability of improvement π
9:     $\Xi_{m+1} \leftarrow \Xi_m \cup \{\mu_m\}$
10:     $m \leftarrow m+1$
11: $\mu_{\max}^k = max(\Xi_m)$ (The load among Ξ_m that leads to the maximum value of the error estimator)

3.3.3 Construction of a Gaussian regression between the exact error and the error estimator to monitor the convergence of the procedure.

One important matter in this procedure is that the error indicator (the norm of the residual in our case), for computational saving reasons, is driving the control of our algorithm. Beyond a certain proportionality, which drives the algorithm assuming smaller error indicators lead to smaller errors, there is no control on the magnitude of the actual error. This is a problem as we would like to build a reduced model that is accurate up to a certain tolerance that is chosen by the user. Hence, it is necessary to build some sort of map between estimator and error, which can then be used as a stopping criterion in our greedy algorithm by linking the value of the indicator to the actual error. This problem has been treated in [43] by using a linear regression using data samples obtained from the snapshots, that regression being updated each time a new snapshot is available. In this paper, we use again a Gaussian regression (in a similar way to [55]), just like it was done for the regression of the error indicator against the parameters. However, in this case, we consider noise, since there is not an exact monotonic match between error indicator and exact error a priori. This implies that the covariance expression changes slightly:

$$cov_n(\mathbf{x}, \mathbf{y}) = \sigma^2 \exp((\mathbf{x^P} - \mathbf{x^q})^T \mathbf{I_\theta}(\mathbf{x^P} - \mathbf{x^q})) + \sigma_n^2 \delta_{pq}, \tag{52}$$

where δ_{pq} is the Kronecker delta, and σ_n^2 is the variance of a noise assumed Gaussian, which is a new hyperparameter. Again, more details can be found in [53]. The main advantage over a linear regression is that it provides a more flexible fit as well as a confidence interval that can be used to ensure a bound on the error. Examples of regression with confidence intervals of one standard error from various numbers of data samples are displayed in Figure 15 and 16. We define $\widehat{\mathcal{Q}^+}$, the "pessimistic" estimate of the quantity of interest $\mathcal{Q}^{\mathrm{HR}}$, computed from a value of the error indicator $\mathcal{J}^{\mathrm{HR}}$ through the Gaussian map $\mathcal{G}$ plus one standard deviation σ (estimated from the data in the gaussian process regression), which gives about an 85% confidence that the actual error is below this value:

$$\widehat{\mathcal{Q}^+} = \mathcal{G}(\mathcal{J}^{\mathrm{HR}}) + \sigma. \tag{53}$$

(a) Gaussian regression with two snapshots.
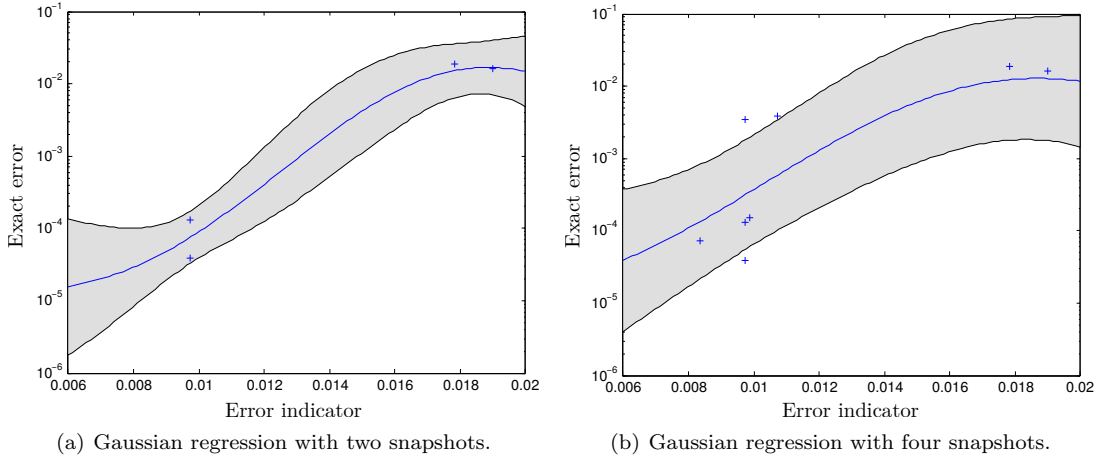
(b) Gaussian regression with four snapshots.

Figure 15: Evolution of the map between error indicator and exact error as the number of observations (snapshots) increases. From each snapshot, two errors can be computed: one before the enrichment of the basis and one afterwards. This allows to include more points to build the map.
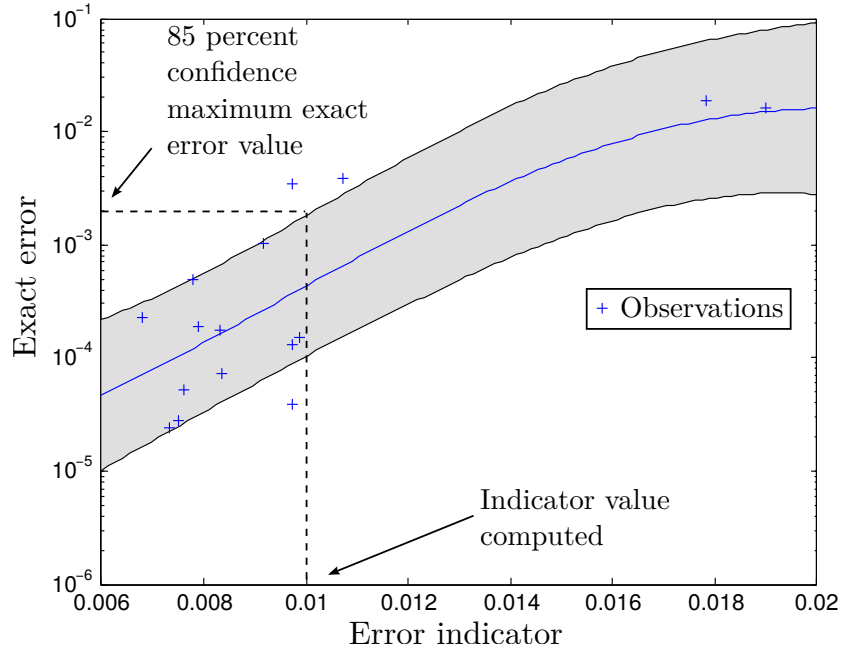

Figure 16: Gaussian process regression between error estimator and exact error, with 70 percent confidence interval. Given the current observations (i.e. the snapshots in our reduced-order modelling jargon), a Gaussian regression is performed to establish a map between exact error and error estimator with statistical knowledge, which gives a confidence interval. Given a value of the error indicator, the user can use the map to have a safe estimate of the exact error using the confidence interval so as not to underestimate the error.

3.3.4 Optimal choice of the size of the reduced spaces to achieve a user-defined tolerance

One important matter, once a new solution has been computed together with a singular value decomposition of its error on the current reduced basis $\boldsymbol{\Phi}$, is to choose how many basis vectors $(\boldsymbol{\phi}_{\mathrm{add}i})_{i=1..n_{\mathrm{add}}}$ should be concatenated to the basis $\boldsymbol{\Phi}$, so that the ROM achieves some user-defined target tolerance ϵ. The same question goes for the number of basis vectors $\boldsymbol{\Psi}$ representing the internal forces for the system approximation.

We choose to tackle this issue in two stages, by first making sure the size of the displacement reduced basis $\boldsymbol{\Phi}$ is large enough for $\mathcal{Q}^{\mathrm{R}}$ to achieve a certain fraction of the tolerance ϵ, and then choosing the dimension of $\boldsymbol{\Psi}$ to achieve the tolerance as well as insuring a monotonic decrease of the reduced and hyperreduced residuals $\mathcal{J}^{\mathrm{R}}$ and $\mathcal{J}^{\mathrm{HR}}$.

The procedure starts by computing an initial solution, chosen for an arbitrary value of the parameter μ, as well as its residuals $\mathcal{J}_{ini}^{\mathrm{R}}, \mathcal{J}_{ini}^{\mathrm{HR}}$. These two residuals will be used as initial residual tolerances.

Determining the size of the displacement reduced basis $\boldsymbol{\Phi}$. Assume we are at step k of the greedy algorithm. We denote the displacement basis $\boldsymbol{\Phi}^k$. The snapshot was enriched with a new exact solution whose projection error with the current reduced basis $\boldsymbol{\Phi}^k$ was decomposed into a POD expansion $\boldsymbol{\Phi}_{\mathrm{add}}$, i.e. $e_{\mathrm{proj}} \simeq \sum_i^{n_{\mathrm{add}}} \alpha_i \boldsymbol{\phi}_{\mathrm{add}i}$.

We successively evaluate the quantity of interest $\mathcal{Q}^{\mathrm{R}}$ (defined in equation (29)) with an increasing number of basis vectors until $\mathcal{Q}^{\mathrm{R}} < \gamma_Q^{\mathrm{R}}.\epsilon$ and $\mathcal{J}^{\mathrm{R}} < \nu_{\mathrm{current}}^{\mathrm{R}}$, where γ_Q^{R} is a scalar smaller than 1 that forces the reduced model (not hyperreduced) to achieve "comfortably" the tolerance ϵ, allowing the hyperreduced model, which is an approximation of the reduced model, to actually achieve the tolerance ϵ. In mathematical terms, this can be written:

$$\min \quad \dim(\boldsymbol{\Phi}_{\mathrm{add}}) \quad such\ that \quad \mathcal{Q}^{\mathrm{R}}(\boldsymbol{\Phi}^{k+1}) < \gamma_Q^{\mathrm{R}}.\epsilon \quad and \quad \mathcal{J}^{\mathrm{R}}(\boldsymbol{\Phi}^{k+1}) < \nu_{\mathrm{current}}^{\mathrm{R}}, \tag{54}$$

where $\boldsymbol{\Phi}^{k+1} = [\boldsymbol{\Phi}^k, \boldsymbol{\Phi}_{\mathrm{add}}]$ (i.e. $\boldsymbol{\Phi}^{k+1}$ is the concatenation of $\boldsymbol{\Phi}^k$ and $\boldsymbol{\Phi}_{\mathrm{add}}$).

The residual tolerance is updated as: $\nu_{\mathrm{current}}^{\mathrm{R}} = \gamma_\nu^{\mathrm{R}}.\nu_{\mathrm{current}}^{\mathrm{R}}$, with $\gamma_\nu^{\mathrm{R}} < 1$. The condition on the residual ensures its decrease throughout the procedure. This is important since the indicator quantity $\mathcal{J}^{\mathrm{R}}$ (influencing $\mathcal{J}^{\mathrm{HR}}$) drives the procedure, the exact error being used for the stopping criterion only (through the Gaussian process regression between error indicator and actual error). The value of $\nu_{\mathrm{current}}^{\mathrm{R}}$ is initialized with the value of the initial residual of the initial ROM, whose size is chosen minimal, typically only of dimension 1 to start with.

Note that this step is quite expensive, as it requires to evaluate the reduced solution several times with no hyperreduction. It could be made cheaper by substituting the evaluations of the non-hyperreduced ROM by a finely (i.e. with a high-dimensional basis $\boldsymbol{\Psi}$) hyperreduced ROM which would be cheaper to evaluate. However, the construction of the hyperreduction ROM is expensive in itself since it requires evaluation of the non-hyperreduced counterpart to build the snapshot necessary to build the internal forces basis $\boldsymbol{\Psi}$. A trade-off would have to be found. In our case, we keep the strategy as it is, keeping in mind that although computationally intensive, this procedure is performed offline.

Determining the size of the internal forces reduced basis $\boldsymbol{\Psi}$. In a similar way, we will successively evaluate the quantity of interest $\mathcal{Q}^{\mathrm{HR}}$ with an increasing number of basis vectors until the tolerance is reached for all the values of the parameter in the current snapshot:

$$\min \quad \dim(\boldsymbol{\Psi}) \quad such\ that \quad \left(\max_{\mu \in \Xi_k} \mathcal{Q}^{\mathrm{HR}}(\boldsymbol{\Psi}, \mu)\right) < \epsilon \quad and \quad \left(\max_{\mu \in \Xi_k} \mathcal{J}^{\mathrm{HR}}(\boldsymbol{\Psi})\right) < \nu_{\mathrm{current}}^{\mathrm{HR}}. \tag{55}$$

In this case, the tolerance ϵ has to be reached for all solutions in the current snapshot Ξ_k, and not only the last one computed. This guarantees the stability of the method. Indeed, unlike when enriching the displacement basis $\boldsymbol{\Phi}$, there is no guarantee on the monotonic decrease of the quantity of interest. To guarantee the monotonic convergence of the error indicator, the residual is updated at each step:

$\nu_{\text{current}}^{\text{HR}} = \gamma_\nu^{\text{HR}} \nu_{\text{current}}^{\text{HR}}$. The general construction of the reduced basis within one pseudo-parameter space $\widehat{\mathcal{P}}^n$ is described in Algorithm 4.

Remark: *Note that this step is not computationally expensive since it only requires evaluations of the hyperreduced model.*

Algorithm 4 Sampling Algorithm within one pseudo parameter space $\widehat{\mathcal{P}}^n$.

1: Define error target tolerance ϵ, initial load $\mu_{\max}^0$, and tolerance parameters smaller than 1: $\gamma_Q^{\text{R}}, \gamma_\nu^{\text{R}}$ and γ_ν^{HR}.
2: Evaluate solution for load $\mu_{\max}^0$ and build initial ROM $\mathbf{\Phi}, \mathbf{\Psi}$
3: Initialise residual tolerances $\nu_{\text{current}}^{\text{R}}$ and $\nu_{\text{current}}^{\text{HR}}$
4: $\widehat{\mathcal{Q}^+} \leftarrow \infty$
5: $k \leftarrow 0$
6: **while** $\widehat{\mathcal{Q}^+} > \epsilon$ **do**
7:     $k \leftarrow k + 1$
8:     $\mu_{\max}^k \leftarrow$ **GaussianProcessPredictor**(Φ, Ψ) *defined in* **Algorithm** 3
9:     Evaluate solution for load $\mu_{\max}^k$
10:    Update the ROM using the tolerances ϵ, $\nu_{\text{current}}^{\text{R}}$ and $\nu_{\text{current}}^{\text{HR}}$:
11:        increase the size of $\mathbf{\Phi}$ until $\mathcal{Q}^{\text{R}} < \gamma_Q^{\text{R}} \epsilon$ and $\mathcal{J}^{\text{R}}(\mathbf{\Phi}^{k+1}) < \nu_{\text{current}}^{\text{R}}$
12:        increase the size of $\mathbf{\Psi}$ until $\mathcal{Q}^{\text{HR}}(\mathbf{\Psi}, \mu) < \epsilon$ AND error indicator $\mathcal{J}^{\text{HR}}(\mathbf{\Psi}) < \nu_{\text{current}}^{\text{HR}}$ FOR ALL elements of the snapshot
13:    $\nu_{\text{current}}^{\text{R}} \leftarrow \gamma_\nu^{\text{R}} \nu_{\text{current}}^{\text{R}}$
14:    $\nu_{\text{current}}^{\text{HR}} \leftarrow \gamma_\nu^{\text{HR}} \nu_{\text{current}}^{\text{HR}}$
15:    Update Gaussian regression between error and residual (as shown in section 3.3.3)
16:    Evaluate the pessimistic error $\widehat{\mathcal{Q}^+}$ from $\mu_{\max}^k$ using the updated Gaussian map.

3.3.5   Application of the Bayesian POD-greedy algorithm

We now proceed to apply the POD-greedy Algorithms 2,3 and 4 on the RVE problem described in section 2. We define the target tolerance $\epsilon = 10^{-3}$, $\gamma_Q^{\text{R}} = \frac{1}{2}, \gamma_\nu^{\text{R}} = \frac{1}{2}$ and $\gamma_\nu^{\text{HR}} = 0.9$. We proceed to build a reduced-order model achieving tolerance ϵ on the successive pseudo parameter spaces $\widehat{\mathcal{P}}$ of dimensions 2, 5 and 8. The very initial parameter value is the proportional loading of equal value in all directions (that is in ϵ_{xx}, ϵ_{yy} and ϵ_{xy}). Results are displayed in Figure 17.

After achieving the tolerance for snapshots in the initial pseudo parameter space of dimension 2, the pessimistic value of the error (up to one standard error) $\widehat{\mathcal{Q}^+}$ increases slightly when moving on to the space of dimension 5. This is not surprising since the reduced-order model was constructed to achieve the tolerance on the space of dimension 2 and does not represent as well the space of dimension 5. However, this error increase is small and remains underneath the target tolerance ϵ. Moving on to the space of dimension 8 leads to the same analysis. When considering the space of dimension 11, we can see that the error decreases. This means that despite the last computed solution belongs to a space of larger dimension (and is the least well represented one) than the space used to build the ROM, it is correctly approximated. One can then argue that the current ROM is accurate enough to represent the solutions issued from parameter spaces of any dimension. Hence, there is no need to consider any finer spaces and the procedure can stop there.

Remark: *For dimension 8 and 11, we used a quasi-random Latin-Hypercube sampling [56] rather that the Gaussian process regression described in section 3.3.3. Indeed, in higher dimensions, the Gaussian process regression requires proportionally more data to make sensible predictions.*
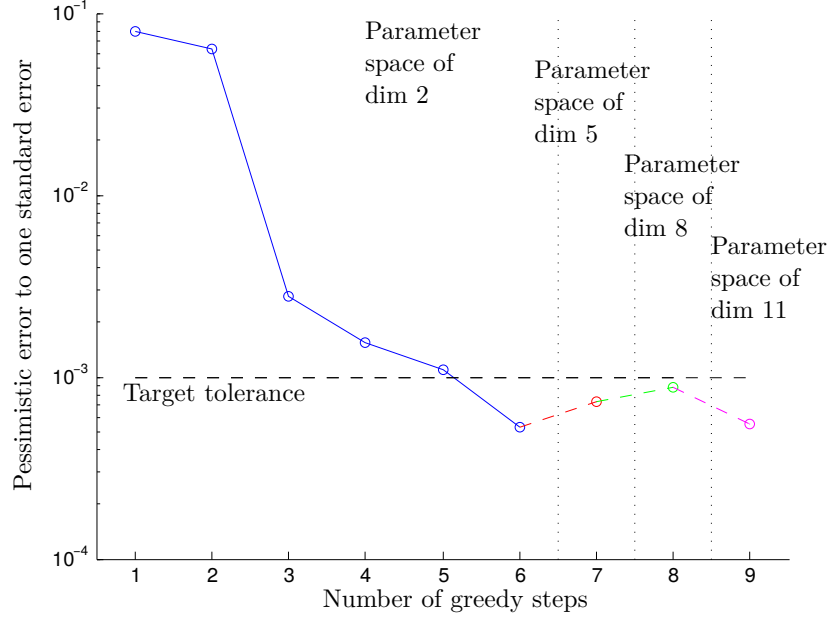
Figure 17: Evolution of the pessimistic error $\widehat{\mathcal{Q}^+}$ (defined in equation (53)) inside and across pseudo-parameter spaces. The error is computed from the error indicator using the Gaussian process predictor plus one standard error as explained in section 3.3.2. The reduced-order model is first build based on solutions from pseudo parameter space $\widehat{\mathcal{P}}^0$ that is of dimension 2 up to reaching the tolerance $\epsilon = 10^{-3}$. Once the tolerance is reached, pseudo parameter space $\widehat{\mathcal{P}}^1$ of dimension 5 is considered. Because the first evaluation of the error is already achieving the tolerance ϵ, the procedure moves on to space $\widehat{\mathcal{P}}^2$ of dimension 8 which also achieves this tolerance straight away. Moving onto the space $\widehat{\mathcal{P}}^3$ of dimension 11, the error decreases: one can consider that convergence has been achieved for parameter spaces of any size.

4   Conclusion and perspectives

In this paper, we have developed an Offline/Online Reduced Basis strategy to reduce the computational cost of solving the RVE boundary value problem involved in nested computational homogenisation (i.e. FE^2). Our strategy has been illustrated in the context of elastic-damageable particulate composites. Such problems can be parametrised by the history of the far-fields that are applied as boundary conditions to the RVE. The main challenge comes from the very high-dimensionality of the parameter domain which consists of all possible macroscopic load histories that preserve material stability. This makes the sampling of the parameter space, which is necessary to train projection-based reduced models, a complicated task. We have proposed two strategies to solve this problem:

- The sampling is done randomly, in a brute force manner, whilst enforcing that a certain increment of energy dissipation occurs at each timstep of the discrete load history. The reduced space is found by using the Proper Orthogonal Decomposition.

- The problem is solved using a POD-greedy reduced-basis method. To reduce the dimensionality of the RVE problem to tractable levels, the parameter space is substituted by a hierarchy of approximate spaces of small and increasing dimensions. A reduced-order model is computed for each of these approximate spaces, using a POD-Greedy training algorithm, in conjunction with a Bayesian-optimisation-based a posteriori error estimate.

The brute force strategy can be very expensive when requiring a high accuracy of the ROM. Indeed, despite the minimum dissipation constraint, the exhaustiveness of the snapshot requires a large number of random solutions. This approach can be appealing nevertheless if the accuracy required is relatively low, due to its low computational cost and ease of implementation. On the other hand, the hierarchical reduced basis strategy is exploring the parameter space in a robust and quasi-optimal manner, at the cost of a certain algorithmic complexity.

Coming back to the context of multiscale modelling, we can question the necessity of computing reduced models of RVE problems in the space of arbitrary far-field loads. Indeed, in practical applications, only specific loadings may actually be applied to the RVE, making the pursuit of an exhaustive snapshot irrelevant. We are currently investigating the possibility of integrating some knowledge about potential macroscopic solutions in order to restrict the size of the parameter domain a priori.

Acknowledgements

References

[1] E. Sanchez-Palencia. Non homogeneous media and vibration theory. *Lecture Notes in Physics*, 127, 1980.

[2] P. Suquet. *Homogenization techniques for composite media*, chapter Elements of Homogenization for Inelastic Solid Mechanics. E. Sanchez-Palencia and A. Zaoui, Springer Verlag, 1987.

[3] Sia Nemat-Nasser and Muneo Hori. *Micromechanics: overall properties of heterogeneous materials*, volume 2. Elsevier Amsterdam, 1999.

[4] Graeme W Milton. *The theory of composites*, volume 6. Cambridge University Press, 2002.

[5] Jacob Fish and Wen Chen. Higher-order homogenization of initial/boundary-value problem. *Journal of engineering mechanics*, 127(12):1223–1230, 2001.

[6] S. Forest, F. Pradel, and K. Sab. Asymptotic analysis of heterogeneous cosserat media. *International Journal of Solids and Structures*, 38(26-27):4585–4608, 2001.

[7] G. Allaire. Homogenisation and two-scale convergence. *SIAM J. Math. Anal.*, 23(6):1482–1518, 1992.

[8] V. Buryachenko. *Micromechanics of Heterogeneous Materials*. Springer, 2007.

[9] T Mori and K Tanaka. Average stress in matrix and average elastic energy of materials with misfitting inclusions. *Acta metallurgica*, 21(5):571–574, 1973.

[10] JR Willis. Bounds and self-consistent estimates for the overall properties of anisotropic composites. *Journal of the Mechanics and Physics of Solids*, 25(3):185–202, 1977.

[11] T Zohdi, M Feucht, D Gross, and P Wriggers. A description of macroscopic damage through microstructural relaxation. *International journal for numerical methods in engineering*, 43(3):493–506, 1998.

[12] F. Feyel and J.-L. Chaboche. FE^2 multiscale approach for modelling the elastoviscoplastic behaviour of long fibre SiC/Ti composite materials. *Computer Methods in Applied Mechanics and Engineering*, 183(3-4):309–330, 2000.

[13] J. Fish, K. Shek, M. Pandheeradi, and M. S. Shephard. Computational plasticity for composite structures based on mathematical homogenization: Theory and practice. *Computer Methods in Applied Mechanics and Engineering*, 148:53–73, 1997.

[14] Christian Miehe. Strain-driven homogenization of inelastic microstructures and composites based on an incremental variational formulation. *International Journal for Numerical Methods in Engineering*, 55(11):1285–1322, 2002.

[15] Tarek J. Zohdi and Peter Wriggers. *Introduction to Computational Micromechanics*, volume 20 of *lecture notes in applied and computational mechanics*. Springer, 2005.

[16] M.G.D. Geers, V.G. Kouznetsova, and W.A.M. Brekelmans. Multi-scale computational homogenization: Trends and challenges. *J. Computational Applied Mathematics*, 234(7):2175–2182, 2010.

[17] George J Dvorak. Transformation field analysis of inelastic composite materials. *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, 437(1900):311–327, 1992.

[18] Jean-Claude Michel and Pierre Suquet. Nonuniform transformation field analysis. *International journal of solids and structures*, 40(25):6937–6955, 2003.

[19] Felix Fritzen and T Böhlke. Three-dimensional finite element implementation of the nonuniform transformation field analysis. *International Journal for Numerical Methods in Engineering*, 84(7):803–829, 2010.

[20] Caglar Oskay and Jacob Fish. Eigendeformation-based reduced order homogenization for failure analysis of heterogeneous materials. *Computer Methods in Applied Mechanics and Engineering*, 196(7):1216–1243, 2007.

[21] Jacob Fish, Vasilina Filonova, and Zheng Yuan. Hybrid impotent–incompatible eigenstrain based homogenization. *International Journal for Numerical Methods in Engineering*, 2013.

[22] Somnath Ghosh. *Micromechanical analysis and multi-scale modeling using the Voronoi cell finite element method*. Taylor & Francis US, 2011.

[23] J Yvonnet and Q.C. He. The reduced model multiscale method (R3M) for the non-linear homogenization of hyperelastic media at finite strains. *Journal of Computational Physics*, 223(1):341–368, 2007.

[24] E Monteiro, J Yvonnet, and QC He. Computational homogenization for nonlinear conduction in heterogeneous materials using model reduction. *Computational Materials Science*, 42(4):704–712, 2008.

[25] P. Breitkopf L. Xia. A reduced multiscale model for nonlinear structural topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 280:117–134, 2014.

[26] L. Sirovich. Turbulence and the dynamics of coherent structures. part I: coherent structures. *Quarterly of Applied Mathematics*, 45:561–571, 1987.

[27] Gal Berkooz, Philip Holmes, and John L Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual review of fluid mechanics*, 25(1):539–575, 1993.

[28] P.A. LeGresley and J.J. Alonso. Investigation of non-linear projection for pod based reduced order models for aerodynamics. *AIAA paper*, 926:2001, 2001.

[29] M. Meyer and H.G. Matthies. Efficient model reduction in non-linear dynamics using the Karhunen-Loeve expansion and dual-weighted-residual methods. *Computational Mechanics*, 31(1):179–191, 2003.

[30] K. Kunisch and S. Volkwein. Galerkin Proper Orthogonal Decomposition Methods for a General Equation in Fluid Dynamics. *SIAM Journal on Numerical analysis*, 40(2):492–515, 2003.

[31] P. Kerfriden, P. Gosselet, S. Adhikari, and S. Bordas. Bridging proper orthogonal decomposition methods and augmented Newton-Krylov algorithms: an adaptive model order reduction for highly nonlinear mechanical problems. *Computer Methods in Applied Mechanics and Engineering*, 200(5-8):850–866, 2011.

[32] K. Carlberg, C. Bou-Mosleh, and C. Farhat. Efficient non-linear model reduction via a least-squares petrov–galerkin projection and compressive tensor approximations. *International Journal for Numerical Methods in Engineering*, 86(2):155–181, 2011.

[33] Pierre Kerfriden, Olivier Goury, Timon Rabczuk, and Stephane Pierre-Alain Bordas. A partitioned model order reduction approach to rationalise computational expenses in nonlinear fracture mechanics. *Computer methods in applied mechanics and engineering*, 2012.

[34] A. Radermacher and S. Reese. Model reduction in elastoplasticity: proper orthogonal decomposition combined with adaptive sub-structuring. *Computational Mechanics*, 54(3):677–687, 2014.

[35] P. Kerfriden, J. J. Ródenas, and S. P.-A. Bordas. Certification of projection-based reduced order modelling in computational homogenisation by the constitutive relation error. *International Journal for Numerical Methods in Engineering*, 97(6):395–422, 2014.

[36] Olivier Goury. *Computational time savings in multiscale fracture mechanics using model order reduction*. PhD thesis, Cardiff University, 2015.

[37] C. Prud'homme, D. V. Rovas, K. Veroy, L. Machiels, Y. Maday, A. T. Patera, and G. Turinici. Reliable Real-Time Solution of Parametrized Partial Differential Equations: Reduced-Basis Output Bound Methods. *Journal of Fluids Engineering*, 124(1):70–80, 2002.

[38] M. Barrault, Y. Maday, N.C. Nguyen, and A.T. Patera. An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus de Mathématiques*, 339(9):667–672, 2004.

[39] Tan Bui-Thanh, Karen Willcox, and Omar Ghattas. Model reduction for large-scale systems with high-dimensional parametric input space. *SIAM Journal on Scientific Computing*, 30(6):3270–3288, 2008.

[40] A Quarteroni, G Rozza, and A Manzoni. Certified reduced basis approximation for parametrized partial differential equations and applications. *Journal of Mathematics in Industry*, 1(3):1–44, 2011.

[41] P. G. Constantine and Q. Wang. Residual minimizing model interpolation for parameterized nonlinear dynamical systems. *SIAM Journal on Scientific Computing*, 34:118–144, 2012.

[42] Donald R Jones. A taxonomy of global optimization methods based on response surfaces. *Journal of global optimization*, 21(4):345–383, 2001.

[43] A Paul-Dubois-Taine and D Amsallem. An adaptive and efficient greedy procedure for the optimal training of parametric reduced-order models. *International Journal for Numerical Methods in Engineering*, 102(5):1262–1292, 2015.

[44] P. Astrid, S. Weiland, K. Willcox, and A.C.P.M. Backx. Missing point estimation in models described by proper orthogonal decomposition. *IEEE Transactions on Automatic Control*, 53(10):2237–2251, 2008.

[45] D. Ryckelynck. Hyper-reduction of mechanical models involving internal variables. *International Journal for Numerical Methods in Engineering*, 77(1):75 – 89, 2008.

[46] S. Chaturantabut and D.C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal on Scientific Computing*, 32:2737–2764, 2010.

[47] Charbel Farhat, Philip Avery, Todd Chapman, and Julien Cortial. Dimensional reduction of nonlinear finite element dynamic models with finite rotations and energy-based mesh sampling and weighting for computational efficiency. *International Journal for Numerical Methods in Engineering*, 98(9):625–662, 2014.

[48] A. Arslan, R. Ince, and B. L. Karihaloo. Improved lattice model for concrete fracture. *Journal of Engineering Mechanics*, 128(157-65), 2002.

[49] Pierre Kerfriden, Karl Michael Schmidt, Timon Rabczuk, and Stephane Pierre-Alain Bordas. Statistical extraction of process zones and representative subspaces in fracture of random composites. *International Journal for Multiscale Computational Engineering*, 11(3), 2013.

[50] J-C. Jouhaud T. Braconnier, M. Ferrier and P. Sagaut. Towards an adaptive pod/svd surrogate model for aeronautic design. *Computers and Fluids*, 40(1):195–209, 2011.

[51] S. Chaturantabut and D.C. Sorensen. Discrete empirical interpolation for nonlinear model reduction. In *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, pages 4316–4321. Ieee, 2004.

[52] Richard Everson and Lawrence Sirovich. Karhunen–loeve procedure for gappy data. *J. Opt. Soc. Am. A*, 12(8):1657–1664, 1995.

[53] Carl Edward Rasmussen. Gaussian processes for machine learning. 2006.

[54] D Wirtz, DC Sorensen, and B Haasdonk. A posteriori error estimation for deim reduced nonlinear dynamical systems. *SIAM Journal on Scientific Computing*, 36(2):A311–A338, 2014.

[55] Martin Drohmann and Kevin Carlberg. The romes method for statistical modeling of reduced-order-model error. *SIAM/ASA Journal on Uncertainty Quantification*, 3(1):116–145, 2015.

[56] Michael Stein. Large sample properties of simulations using latin hypercube sampling. *Technometrics*, 29(2):143–151, 1987.