



**HAL**  
open science

# An unsupervised machine learning approach to reduce nonlinear FE2 multiscale calculations using macro clustering

Souhail Chaouch, Julien Yvonnet

► **To cite this version:**

Souhail Chaouch, Julien Yvonnet. An unsupervised machine learning approach to reduce nonlinear FE2 multiscale calculations using macro clustering. *Finite Elements in Analysis and Design*, 2024, 229, pp.104069. 10.1016/j.finel.2023.104069 . hal-04247386

**HAL Id: hal-04247386**

**<https://univ-eiffel.hal.science/hal-04247386v1>**

Submitted on 18 Oct 2023

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

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

# An unsupervised machine learning approach to reduce nonlinear FE<sup>2</sup> multiscale calculations using macro clustering

Souhail CHAOUCH, Julien YVONNET<sup>a,\*</sup>

<sup>a</sup>Univ Gustave Eiffel, MSME, CNRS UMR 8208, Marne-la-Vallée, F-77454, France

---

## ARTICLE INFO

### Keywords:

Multi-scale modeling  
Unsupervised learning  
k-means FE2  
Finite Elements Method

## ABSTRACT

Solving nonlinear multiscale methods with history-dependent behaviors and fine macroscopic meshes is a well-know challenge. In this work, an unsupervised machine learning-based clustering approach is developed to reduce nonlinear Multilevel Finite Element-FE<sup>2</sup> calculations. In contrast with most available techniques which aim at developing Reduced Order Models (ROM) or AI-based surrogate models for the microscale nonlinear problems, the present technique reduces the problem from the macro scale by creating clusters of macro Gauss points which are assumed to be in close mechanical states. Then, a single micro nonlinear Representative Volume Element (RVE) calculation is performed for each cluster. A linear approximation of the macro stress is used in each cluster. Handling internal variables is carried out by using anelastic macro strains in the clustering vectors in addition to the macro strains components. Finally, some convergence issues related to the use of clusters at the macro scale are addressed through a cluster freezing algorithm. The technique is applied to nonlinear hyperelastic, viscoelastic and elastoplastic composites. In contrast to available ROM or machine-learning -based acceleration techniques, the present method does not require neither preliminary off-line calculations, nor training, nor data base, nor reduced basis at the macro scale, while maintaining typical speed-up factors about 20 as compared to classical FE<sup>2</sup>.

---


## 1. Introduction

Multiscale modeling of nonlinear heterogeneous materials, with time-dependent behaviors is a highly challenging topic in material engineering. In contrast with the linear properties, predicting the homogenized behavior of composites/heterogeneous materials with elasto-plastic, viscoelastic or damageable phases requires advanced numerical multiscale techniques. The most common approach which is nowadays used is the so-called FE<sup>2</sup> method, initiated in [1, 2, 3, 4, 5, 6] (see also more recent review papers and the references therein in [7, 8]). In this technique, a macro (structure) problem is discretized with finite elements. At the micro scale, a Representative Volume Element (RVE) is defined and also usually discretized by finite elements, even though other solvers can be used (Fast Fourier Transform, discrete elements, etc.). At the macro scale, the constitutive law in each element is assumed to be unknown, and obtained numerically by solving a nonlinear RVE problem at each macro Gauss point. The main advantage of this approach is its generality, as no restrictive assumption is required on the type of nonlinearity within the RVE. The obvious drawback is the computational cost. As a nonlinear Finite Element Method (FEM) problem must be solved at each Gauss point of the macro mesh and for all macro Newton iterations, the original version of FE<sup>2</sup> is limited and cannot be applied to large macro (2D or 3D) meshes and thus industrial applications. The interested reader can refer to [8] to have an overview of the impressive number of applications and extensions of this method.

To tackle the computational time and data storage issues in FE<sup>2</sup>, computational acceleration techniques have been proposed, mainly based on Reduced Order Models (ROMs) associated with the nonlinear micro RVE problem: (i) Interpolation-based ROMs; (ii) Proper Orthogonal Decomposition (POD)-based ROMs; (iii) Transformation Field Analysis/Non-uniform Transformation Field Analysis (TFA/NTFA)-based ROMs; (iv) AI-based ROMs. Other techniques using reduction of degrees of freedom in cells can also be mentioned, see e.g. [9]. These different families are briefly reviewed here. A first idea to construct ROMs in FE<sup>2</sup> is to project the nonlinear equations associated with the RVE on a reduced basis. For this purpose, POD approaches have been proposed in [10], in the framework of hyperelastic materials, and extended e.g. in [11, 12, 13, 14]. The displacement modes within the RVE are projected on a reduced basis and the related nonlinear RVE problem is reduced. A second idea is to fully replace the RVE

---

\*Corresponding author

 julien.yvonnet@univ-eiffel.fr (J. YVONNET)

ORCID(s):

nonlinear problem by a "black box" surrogate model, where the input-outputs (e.g. corresponding to macro strain-macro stress relationships) can be efficiently evaluated from preliminary off-line calculations. Early versions of such "Data-driven" surrogate models in FE<sup>2</sup> can be found in [10, 15] using higher-order interpolation schemes. In this approach, the effective strain density energy of a hyperelastic RVE was replaced by a higher-interpolation scheme from discrete values of the effective energy computed via off-line calculations on the RVE. More recent developments of this technique using High-Order Singular Value Decomposition (HOSVD) or radial basis functions can be found e.g. in [16, 17]. Note that such methods have been mainly applied for hyperelastic materials or nonlinearities without loading history-effects. TFA/NTFA approaches [18] were used to reduce the nonlinear viscoplastic multiscale problems in [19], and a POD-based ROM was used to express the local anelastic (elastoplastic) fields with a reduced number of parameters and define macro internal variables. A related approach has been developed recently, named Self-Clustering Analysis (SCA) [20, 21, 22, 23], where the local nonlinear problem is reduced by means of a Fast-Fourier Transform (FFT)-type problem using clustering to parametrize the anelastic strain fields with a reduced number of macro internal variables. Artificial Intelligence (AI)-surrogate models for the nonlinear RVE have been proposed for the first time in FE<sup>2</sup> methods in [24]. The main advantage on other higher-order-interpolation-based surrogate model is that it avoids the use of high-dimensional grids or meshes, and allows the use of widely developed available tools and libraries from the AI community. Initially developed for hyperelastic materials [24], this idea has been extended to other nonlinear behaviors and applications, as well as stochastic problems e.g. in [25, 26]. Incorporation of local morphological features of the microstructure was performed using graph-based auto-encoders for hyperelastic polycrystalline RVEs in [27]. Predicting homogenized stress or apparent permeability using Convolutional Neural Networks (CNN) from images of the microstructures was accomplished in [28, 29]. Comparisons between different surrogate models in FE<sup>2</sup>, including polynomial interpolations, different types of ANN, or hyper-reduction [30, 31, 32] at the micro-level, can be found in [33, 34]. It is worth mentioning that the above AI-based surrogate models are restricted to loading path-independent problems. To handle history-dependent problems, Recurrent Neural networks (RNN) based on Long Short-Term Memory (LSTM) units were used in [35, 36, 37] and showed their capability for cyclic loading of elasto-visco-plastic or elastoplastic heterogeneous materials. Gated Recurrent Unit neural networks (GRU) were used in [38, 39, 40] to model 2D elasto-plastic problem with an ability to generalize non-proportional loading conditions for elasto-plastic problems or large deformation response of anisotropic elasto-plastic 2D solids. In [41], geometric encoders on graphs were used to construct a ROM for the local plastic internal variables within the RVE.

Another recent trend in AI-based surrogate ROMs is the development of hybrid physic-AI based models. In [42], auto-encoders were used to identify a reduced set of latent internal state variables of complex inelastic materials in a Thermodynamics-based Artificial Neural Networks (TANN) frame, and combining that with FEM (FEM×TANN) for multiscale analyses. Recurrent Neural Networks with thermodynamic consistency, i.e. avoiding negative dissipation, was proposed in [43].

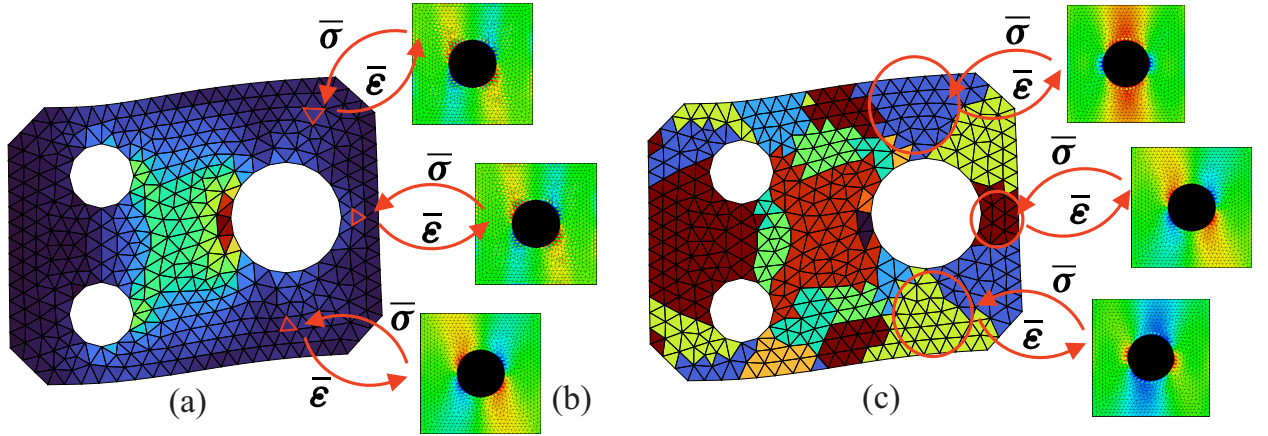
In this paper, an extension of our previous work developed in [44] is proposed, where a new paradigm is introduced: a ROM is proposed for the macro scale FEM problem instead of the micro (RVE) problem. For this purpose, an AI-based unsupervised machine learning technique based on k-means clustering is introduced to avoid redundant calls to nonlinear RVEs at the macro scale within FE<sup>2</sup>. As opposed to Hyper-Reduction techniques mentioned above and to AI-based micro-scale surrogate models, the method does not require neither reduced basis construction nor training, which considerably simplifies the methodology. It should be noted that the present method also fundamentally differs from SCA (see references above) or FEM-Cluster based Analysis (FCA) [45]. Then, the present work contributes to the field of Reduced Order Models in the context of multiscale analysis, but the method we propose operates at the macro scale and not at the micro scale, which makes it complementary with existing micro-scale ROM techniques.

In the present work, several crucial improvements are introduced to the so-called k-means clustering FE<sup>2</sup> (KMFE<sup>2</sup>) developed in [44]. First, the definition of the macro stress within each cluster is extended by using linear interpolation in the strain-stress space. Then, the number of clusters, and then the total number of nonlinear RVE problems to be solved at the macro scale is drastically reduced as compared to the so-called sub-clustering technique developed in [44], by using appropriate clustering vectors containing macro anelastic strain components. Finally, convergence issues related to the use of clusters are addressed.

The paper is organized as follows. Section 2 summarizes the classical FE<sup>2</sup> method. In section 3, the k-means clustering FE<sup>2</sup> method (KMFE<sup>2</sup>) as presented in our first work [44] is first reviewed, then the key novelties introduced in the new version of KMFE<sup>2</sup> are developed. Validation examples and efficiency assessment tests are provided in section 4.

## 2. Review of the FE<sup>2</sup> method

We first review the basics of the so-called FE<sup>2</sup> method [3, 4]. In this approach, two scales are considered. The macro scale is associated with the structure, while the micro scale is defined by a Representative Volume Element (RVE) characterizing the microstructure of the material. A mesh of finite elements is defined for the structure (see Fig. 1 (a)), and referred to as the macro mesh, while each Gauss point is attached to an RVE which is itself discretized by a FEM mesh (see Fig. 1 (b)).



**Figure 1:** Schematic illustration of FE<sup>2</sup> method and k-means clustering FE<sup>2</sup> (KMFE<sup>2</sup>); (a)-(b): classical FE<sup>2</sup> method; (a): macro mesh; (b) micro Representative Volume Elements (RVE) meshes; each Gauss point in the macro mesh is attached to a nonlinear RVE problem; (c) KMFE<sup>2</sup>: only one RVE problem is solved for each cluster of Gauss points, depicted in different colors.

In FE<sup>2</sup> method, a well-identified nonlinear constitutive law is assumed at the micro scale for each phase, while it is unknown for the macro scale. At each Gauss point of the macro mesh, given a macroscopic strain, boundary conditions are then applied to an RVE attached to the Gauss point. After solving the micro nonlinear problem, the macro stress is obtained by numerical averaging of micro stress. The macro equilibrium is usually obtained by an iterative Newton method. In the following, the main equations will be presented for small strains, while an example including finite strains is also presented in the numerical example section 4. The main equations for this case are presented in Appendix A.

### 2.1. Macro scale problem

The macro scale problem is defined in an open domain  $\bar{\Omega} \in \mathbb{R}^D$ , whose boundary is denoted by  $\partial\bar{\Omega}$ , and where  $D$  is the domain dimension. In the absence of body forces, the macro scale problem equations are given by:

$$\nabla \cdot \bar{\sigma}(\bar{\mathbf{x}}) = 0 \text{ in } \bar{\Omega}, \quad (1)$$

where  $\bar{\mathbf{x}}$  is a material point coordinate in the structure and  $\bar{\sigma}$  is the macro Cauchy stress tensor, and  $\nabla \cdot (\bullet)$  denotes the divergence operator. The boundary conditions read:

$$\bar{\mathbf{u}}(\bar{\mathbf{x}}) = \bar{\mathbf{u}}^*(\bar{\mathbf{x}}) \text{ on } \partial\bar{\Omega}_u, \text{ and } \bar{\sigma} \cdot \bar{\mathbf{n}} = \bar{\mathbf{F}}^* \text{ on } \partial\bar{\Omega}_F, \quad (2)$$

where  $\bar{\mathbf{u}}$  denotes displacements at the macro scale,  $\bar{\mathbf{u}}^*$  and  $\bar{\mathbf{F}}^*$  denote prescribed displacements and forces over the macro Dirichlet,  $\partial\bar{\Omega}_u$  and Neumann  $\partial\bar{\Omega}_F$  boundaries, respectively, and  $\bar{\mathbf{n}}$  is the unit normal vector to the external boundary  $\partial\bar{\Omega}$ .

### 2.2. Micro scale problem

We consider an RVE defined in an open domain  $\Omega$ , whose boundary is denoted by  $\partial\Omega$ . Still assuming absence of body forces, the equilibrium equations read:

$$\nabla \cdot \sigma(\mathbf{x}) = 0 \text{ in } \Omega, \quad (3)$$

where  $\boldsymbol{\sigma}$  is the micro Cauchy stress and  $\mathbf{x}$  is a material point coordinate in the RVE. At the micro scale, the constitutive law is an arbitrary nonlinear function in the form

$$\boldsymbol{\sigma}(\mathbf{x}, t) = \mathcal{F}(\boldsymbol{\varepsilon}(\mathbf{x}, t), \boldsymbol{\alpha}(\mathbf{x}, t)), \quad (4)$$

where  $\boldsymbol{\varepsilon} = \frac{1}{2}(\nabla(\mathbf{u}) + \nabla^T(\mathbf{u}))$  is the strain tensor, with  $\nabla(\bullet)$  is the gradient operator, and  $\mathcal{F}$  is a nonlinear operator. In (4),  $\boldsymbol{\alpha}$  denotes a vector of internal variables associated with history-dependent phenomena like e.g. plasticity, damage, etc. The problem is completed with boundary conditions expressed by:

$$\mathbf{u}(\mathbf{x}) = \bar{\boldsymbol{\varepsilon}} \cdot \mathbf{x} + \tilde{\mathbf{u}}(\mathbf{x}) \quad \text{on } \partial\Omega, \quad (5)$$

where  $\tilde{\mathbf{u}}(\mathbf{x})$  is a periodic fluctuation. Equation (5) satisfies

$$\bar{\boldsymbol{\varepsilon}} = \langle \boldsymbol{\varepsilon} \rangle \quad (6)$$

where  $\langle \bullet \rangle = \frac{1}{|\Omega|} \int_{\Omega} (\bullet) d\Omega$  denotes spatial averaging over the RVE. Eq. (6) indicates that the RVE is subjected to a uniform macro strain  $\bar{\boldsymbol{\varepsilon}}$  (see justification e.g. in [46]). In addition, the macro stress is related to the micro stress by spatial averaging as:

$$\bar{\boldsymbol{\sigma}} = \langle \boldsymbol{\sigma}(x) \rangle. \quad (7)$$

Note that (6) and (7) are only valid in the assumption of perfect interfaces [46], which is the assumption adopted in this work. In the following, for the sake of simplicity, we will only consider linear boundary conditions, i.e.  $\tilde{\mathbf{u}}(\mathbf{x}) = 0$  in (5).

### 2.3. Numerical solving procedure

At the macro scale, the weak form associated with (1)-(2) is given by:

$$\mathbf{R}(\bar{\mathbf{u}}, \delta\bar{\mathbf{u}}) = \int_{\bar{\Omega}} \bar{\boldsymbol{\sigma}}(\bar{\boldsymbol{\varepsilon}}(\bar{\mathbf{u}})) : \bar{\boldsymbol{\varepsilon}}(\delta\bar{\mathbf{u}}) d\bar{\Omega} - \int_{\partial\bar{\Omega}_F} \bar{\mathbf{F}}^* \cdot \delta\bar{\mathbf{u}} d\Gamma = 0 \quad \forall \delta\bar{\mathbf{u}} \in \mathcal{U}_0, \forall \bar{\mathbf{u}} \in \mathcal{U}, \quad (8)$$

with  $\mathcal{U} = \left\{ \mathbf{v} \mid \mathbf{v} = \bar{\mathbf{u}}^* \text{ on } \partial\bar{\Omega}_u, \mathbf{v} \text{ sufficiently regular on } \bar{\Omega} \right\}$ ,  $\mathcal{U}_0 = \left\{ \mathbf{v} \mid \mathbf{v} = 0 \text{ on } \partial\bar{\Omega}_u, \mathbf{v} \text{ sufficiently regular on } \bar{\Omega} \right\}$ . Eq. (8) can be solved using an iterative Newton method. The linearization of (8) around a known solution at iteration ( $n$ ) gives:

$$\int_{\bar{\Omega}} \bar{\mathbf{C}}_{tan}(\bar{\boldsymbol{\varepsilon}}(\bar{\mathbf{u}}^n)) : \bar{\boldsymbol{\varepsilon}}(\Delta\bar{\mathbf{u}}) : \bar{\boldsymbol{\varepsilon}}(\delta\bar{\mathbf{u}}) d\bar{\Omega} = - \int_{\bar{\Omega}} \bar{\boldsymbol{\sigma}}(\bar{\boldsymbol{\varepsilon}}(\bar{\mathbf{u}}^n)) : \bar{\boldsymbol{\varepsilon}}(\delta\bar{\mathbf{u}}) d\bar{\Omega} + \int_{\partial\bar{\Omega}_F} \bar{\mathbf{F}}^* \cdot \delta\bar{\mathbf{u}} d\Gamma. \quad (9)$$

By introducing a FEM discretization into equation (9), we obtain the linear system of equations:

$$\bar{\mathbf{K}}_{tan} \Delta\bar{\mathbf{U}} = -\bar{\mathbf{R}}, \quad (10)$$

where  $\bar{\mathbf{R}}$  and  $\bar{\mathbf{K}}_{tan}$  are assembled from the elementary residual vectors  $\bar{\mathbf{R}}^e$  and matrices  $\bar{\mathbf{K}}_{tan}^e$  in a macro element  $\bar{\Omega}^e$ , respectively, expressed by:

$$\bar{\mathbf{R}}^e = \int_{\bar{\Omega}^e} \mathbf{B}^T \bar{\boldsymbol{\sigma}}(\bar{\mathbf{u}}^n) d\bar{\Omega} - \int_{\partial\bar{\Omega}_F^e} \mathbf{N}^T \bar{\mathbf{F}}^* d\Gamma, \quad (11)$$

and

$$\bar{\mathbf{K}}_{tan}^e = \int_{\bar{\Omega}^e} \mathbf{B}^T \bar{\mathbf{C}}_{tan}(\bar{\mathbf{u}}^n) \mathbf{B} d\bar{\Omega}. \quad (12)$$

The shape function matrices  $\mathbf{B}$  and  $\mathbf{N}$  are defined such that  $\bar{\boldsymbol{\varepsilon}}(\delta\bar{\mathbf{u}}) = \mathbf{B}(\bar{\mathbf{x}}) \delta\bar{\mathbf{u}}^e$ ,  $\bar{\boldsymbol{\varepsilon}}(\Delta\bar{\mathbf{u}}) = \mathbf{B}(\bar{\mathbf{x}}) \Delta\bar{\mathbf{u}}^e$  and  $\delta\bar{\mathbf{u}} = \mathbf{N}(\bar{\mathbf{x}}) \delta\bar{\mathbf{u}}^e$ . Finally, we obtain the correction  $\Delta\bar{\mathbf{U}}$  to update the displacement  $\bar{\mathbf{U}}^{n+1} = \bar{\mathbf{U}}^n + \Delta\bar{\mathbf{U}}$ , where  $\bar{\mathbf{U}}$  denotes the global vector of displacements.

Thus,  $\bar{\sigma}$  and  $\bar{\mathbb{C}}_{tan}$  in Eq. (9) are obtained by solving the nonlinear RVE problem (3)-(5) at each Gauss point of the macro mesh. The macro stress  $\bar{\sigma}$  is obtained by (7) and  $\bar{\mathbb{C}}_{tan}$  is only evaluated by perturbation using several additional RVE calculations for each perturbed configuration:

$$\left(\bar{\mathbb{C}}_{tan}^j\right)_{abcd} \simeq \frac{\bar{\sigma}_{ab}(\bar{\epsilon}^j + \delta_{cd}) - \bar{\sigma}_{ab}(\bar{\epsilon}^j)}{\epsilon}, \quad (13)$$

where

$$\delta_{cd} = \epsilon \frac{1}{2} (\mathbf{e}_c \otimes \mathbf{e}_d + \mathbf{e}_d \otimes \mathbf{e}_c), \quad (14)$$

$\epsilon$  being a small numerical parameter, and  $\mathbf{e}_i$  denoting orthogonal vectors. Note that in FE<sup>2</sup>, there is no practical way to evaluate the tangent tensor  $\bar{\mathbb{C}}_{tan}$  analytically. Then, the large number of nonlinear problems to be solved at each macro Newton iteration induces dramatic computational costs, especially for large 2D or 3D meshes. In the next section, we present an approach to reduce these computational costs.

### 3. An enhanced k-means clustering FE<sup>2</sup> method

In this section, we present a method called k-means clustering FE<sup>2</sup> (KMFE<sup>2</sup>), which is based on our previous work [44] and where several new contributions are developed in this paper. The basics ideas are first presented, then the improved version is developed to address some limitations of the initial method.

#### 3.1. Basics of k-means clustering FE<sup>2</sup>

In this section, we briefly recall the basics of KMFE<sup>2</sup> as proposed in [44]. The key idea is to reduce the amount of nonlinear RVE problems by using an unsupervised machine learning technique based on clustering to select the Gauss points in the macro structure mesh that have close mechanical states. Then, instead of solving one nonlinear RVE problem for each Gauss point, a single nonlinear problem is solved in each cluster (see Fig. 1 (c)). This process dramatically reduces the number of nonlinear problems, without the need for a training stage, as no surrogate model is required. This is in contrast with approaches based on supervised machine learning (see e.g. [23, 24, 21, 39, 41], among many others), where an off-line stage is required for the learning step. Such learning step has two drawbacks: (i) the accuracy of the surrogate model critically depends on the size and completeness of the data base: for history-dependent problems, the number of loading trajectories might be high and difficult to define; (ii) the cost of the off-line calculations is usually enormous and the database must be re-constructed for each new microstructure (topology and local behavior).

In KMFE<sup>2</sup> as proposed in [44], at each iteration of the Newton procedure, the Gauss points are classified into clusters using the k-means clustering algorithm (see e.g. [47]). Taking the components of the macro strain tensors into vectors  $\mathbf{v}_i$ ,  $i = 1, 2, \dots, N_G$ , with  $N_G$  the number of Gauss points, a number of  $K$  clusters is defined by the user. Afterwards, the vectors  $\mathbf{v}_i$  are grouped into clusters  $S^k$ ,  $k = 1, 2, \dots, K$  using the algorithm described in Table 1.

Finally, a single micro problem (3)-(5) is solved in each cluster  $k$  with boundary conditions (5), where  $\bar{\epsilon} = \bar{\epsilon}^k$ ,  $\bar{\epsilon}^k$  being defined as the weighted averaging of the macro strains within the cluster  $k$ . As a result,  $\bar{\sigma}(\bar{\epsilon}^k)$ , and  $\bar{\mathbb{C}}_{tan}(\bar{\epsilon}^k)$  can be evaluated. The average strain is defined for each cluster using:

$$\bar{\epsilon}^k = \frac{1}{V^k} \int_{\Omega^k} \bar{\epsilon}(\bar{\mathbf{x}}) d\Omega, \quad (15)$$

where  $V^k = \int_{\Omega^k} d\Omega$ , and  $\Omega^k$  is the domain defined as the union of all macro elements belonging to the cluster  $S^k$ . In [44], a constant value of  $\bar{\sigma}(\bar{\epsilon}^k)$  and  $\bar{\mathbb{C}}_{tan}(\bar{\epsilon}^k)$  was assigned to all integrations points belonging to the same cluster  $k$  (see Fig. 1 (c)). Moreover, to take into account the effects of the internal variables in the RVEs, a heuristic procedure based on the intersection between old and new clusters was adopted (see [44] for more details). This procedure has the drawback of creating many new clusters, called sub-clusters, whose number cannot be controlled by the user and which can be high as compared to the target number of clusters  $K$  based solely on the macro strains. The method can then be summarized into the following steps at each iteration of the macro Newton procedure: (i) perform clustering of macro

**Table 1**

---

**Algorithm 1:** k-means clustering

---

**Input:** A set of  $N$  vectors  $S = \{\mathbf{v}_i, i = 1, 2, \dots, N\}$ , and a user defined number of clusters  $K \ll N$ .

1. Initialize the centroids  $\bar{\mathbf{v}}^k = \mathbf{v}_m^k$ , where  $\mathbf{v}_m^k$  are randomly chosen vectors in  $S$ .
2. Initialise the optimisation function  $J^{(0)} = 0$ .
3. While  $\Delta J \neq 0$ 
  - (a) Assign each vector  $\mathbf{v}_i$  to the cluster  $S^k$  that has the closest mean vector  $\bar{\mathbf{v}}_k$   $S^{k(r)} = \{\mathbf{v}_i \mid \|\mathbf{v}_i - \bar{\mathbf{v}}_k\| \leq \|\mathbf{v}_i - \bar{\mathbf{v}}_m\|, \forall m, m \neq k\}$ .
  - (b) Compute the means  $\bar{\mathbf{v}}_k$  of  $S^{k(r)}$ ,  $k = 1 \dots K$  by  $\bar{\mathbf{v}}_k = \frac{\sum_{i \in S^k} \mathbf{v}_i}{N_k}$ ,  $N_k$  is the number of vectors belonging to  $S^k$ .
  - (c) Compute the objective function  $J^{(r)}$  through  $J^{(r)} = \sum_{k=1}^K \sum_{i \in S^{k(r)}} \|\mathbf{v}_i - \bar{\mathbf{v}}_k\|^2$
  - (d) If  $|\Delta J| := |J^{(r)} - J^{(r-1)}| = 0$ , then  $S^k = S^{k(r)}$ ,  $k = 1 \dots K$  go to (4)
  - (e) Else:  $r = r + 1$  and go to (a)
4. End

**Output:** Clusters  $S^k$ ,  $k = 1 \dots K$  and their representative centroids  $\bar{\mathbf{v}}_k$ ,  $k = 1, 2, \dots, K$

---

Gauss points based on macro stains; (ii) handle internal variable dependence: subdivide clusters into sub-clusters as intersection between old and new clusters (see [44] for more details); (iii) perform one RVE simulation per cluster  $k$  to obtain the macro stress and the macro tangent operator; (iv) assign the same stress and macro tangent operator to all integration points of the cluster.

### 3.2. An improved version of KMFE<sup>2</sup>

In this work, we propose several improvements of the initial version of the method [44], and more specifically related to the above points (ii) and (iv).

#### 3.2.1. Avoiding sub-clusters

To avoid the creation of sub-clusters mentioned above and to reduce the computational times, we propose in this work to add new information into the clustering classification vectors  $\mathbf{v}_i$ . It is important to note that including the whole set of internal variables  $\boldsymbol{\alpha}(\mathbf{x})$  of each RVE in the vectors  $\mathbf{v}_i$  may induce a difficulty: the k-means clustering algorithm may not be efficient and accurate if the input vectors are of too high dimension.

In this paper, we restrict our analysis to anelastic materials without damage. These behaviors include e.g. plasticity, visco-plasticity and viscoelasticity. In this situation, the microscopic strain field within the RVE can be expressed as:

$$\boldsymbol{\varepsilon}(\mathbf{x}) = \boldsymbol{\varepsilon}^e(\mathbf{x}) + \boldsymbol{\varepsilon}^{an}(\mathbf{x}), \quad (16)$$

where  $\boldsymbol{\varepsilon}^{an}(\mathbf{x})$  is an anelastic strains tensor depending on the local internal variables vector  $\boldsymbol{\alpha}(\mathbf{x})$ . We assume here that in absence of damage, the local behavior can be expressed as:

$$\boldsymbol{\sigma}(\mathbf{x}) = \mathbb{C}(\mathbf{x}) : \boldsymbol{\varepsilon}^e(\mathbf{x}) = \mathbb{C}(\mathbf{x}) : (\boldsymbol{\varepsilon}(\mathbf{x}) - \boldsymbol{\varepsilon}^{an}(\boldsymbol{\alpha}(\mathbf{x}))) = \mathbb{C}(\mathbf{x}) : \boldsymbol{\varepsilon}(\mathbf{x}) + \boldsymbol{\sigma}^{an}(\boldsymbol{\alpha}(\mathbf{x})). \quad (17)$$

Then, by analogy with the relation shown in [19], the macroscopic behavior can be re-expressed into the form :

$$\bar{\boldsymbol{\sigma}} = \bar{\mathbb{C}} : \bar{\boldsymbol{\varepsilon}} = \bar{\mathbb{C}} : \left( \bar{\boldsymbol{\varepsilon}} - \bar{\boldsymbol{\varepsilon}}^{an}(\boldsymbol{\alpha}(\mathbf{x})) \right) = \bar{\mathbb{C}} : \bar{\boldsymbol{\varepsilon}} + \bar{\boldsymbol{\sigma}}^{an}(\boldsymbol{\alpha}(\mathbf{x})), \quad (18)$$

where  $\bar{\boldsymbol{\varepsilon}}^{an}(\boldsymbol{\alpha}(\mathbf{x})) = \langle \boldsymbol{\varepsilon}^{an}(\mathbf{x}) \rangle$  and  $\bar{\boldsymbol{\sigma}}^{an}(\boldsymbol{\alpha}(\mathbf{x})) = \langle \boldsymbol{\sigma}^{an}(\mathbf{x}) \rangle$ . We note that Eq. (18) is only true in the case of small strains, due to the superposition principle. However, it is not true in the finite strains context. However, this equation only provides a guideline to construct the clustering vector, but is not eventually used to define the constitutive relationship. From (18), it can be noted that the only pertinent information to recover the macro stress  $\bar{\boldsymbol{\sigma}}$  is  $\bar{\boldsymbol{\varepsilon}}$  and  $\bar{\boldsymbol{\varepsilon}}^{an}$ , or  $\bar{\boldsymbol{\varepsilon}}$  and  $\bar{\boldsymbol{\sigma}}^{an}$ . Thus, we propose to extend the classification of Gauss points using either one of the following vectors for a 2D problem:

$$\mathbf{v}_i = \left\{ \bar{\varepsilon}_{11}^i, \bar{\varepsilon}_{22}^i, 2\bar{\varepsilon}_{12}^i, \bar{\varepsilon}_{11}^{an^i}, \bar{\varepsilon}_{22}^{an^i}, 2\bar{\varepsilon}_{12}^{an^i} \right\}, \forall i \in S^k, \quad (19)$$

or

$$\mathbf{v}_i = \left\{ \bar{\epsilon}_{11}^i, \bar{\epsilon}_{22}^i, 2\bar{\epsilon}_{12}^i, \overline{\sigma}_{11}^{an}, \overline{\sigma}_{22}^{an}, \overline{\sigma}_{12}^{an} \right\}, \forall i \in S^k. \quad (20)$$

This last case might be useful in situations where the local anelastic strain  $\boldsymbol{\epsilon}^{an}(\mathbf{x})$  is not available directly in the numerical calculations (see e.g. section 4.2). Then, as the vector  $\mathbf{v}_i$  contains values with different units, a normalization is required, like e.g.:

$$\mathbf{v}_i = \left\{ \frac{\bar{\epsilon}_{11}^i}{\max(|\bar{\epsilon}|)}, \frac{\bar{\epsilon}_{22}^i}{\max(|\bar{\epsilon}|)}, \frac{\bar{\epsilon}_{12}^i}{\max(|\bar{\epsilon}|)}, \frac{\overline{\sigma}_{11}^{an}}{\max(|\overline{\sigma}^{an}|)}, \frac{\overline{\sigma}_{22}^{an}}{\max(|\overline{\sigma}^{an}|)}, \frac{\overline{\sigma}_{12}^{an}}{\max(|\overline{\sigma}^{an}|)} \right\}, \forall i \in S^k, \quad (21)$$

where  $\max(|\bar{\epsilon}|)$  and  $\max(|\overline{\sigma}^{an}|)$  denote the maximum values of the macro strains and anelastic stresses for all Gauss points in the macro mesh. Thus, this new procedure does not induce the creation of additional clusters (sub-clusters) as in [44], and yet takes into account the influence of internal variables at the micro scale.

Furthermore, it is important to note that after a new clustering step at iteration  $j + 1$ , one cluster associated with a domain  $\{\Omega_k\}_{j+1}$  may contain several Gauss points associated with previous clusters at iteration  $j$ , and then different sets of internal variables. A schematic illustration can be found in Fig.2b. A new domain associated with a cluster can be decomposed as:

$$\{\Omega_k\}_{j+1} = \bigcup_{p=1}^K \Omega_{pk}, \quad (22)$$

where  $\Omega_{pk} = \{\Omega_k\}_{j+1} \cap \{\Omega_p\}_j$ . If we consider a new domain associated with a cluster  $k$ , each of its sub-domain  $\Omega_{pk} \subset \{\Omega_k\}_{j+1}$  is attached to a different internal variable vector  $\boldsymbol{\alpha}^p$ . Defining the new set of internal variables  $\boldsymbol{\alpha}^k$  as a weighted average of previous internal variable vectors would lead to inconsistent mechanical states. Instead, we choose  $\boldsymbol{\alpha}^k$  for the new cluster with respect to previous internal variables vectors as follows:

i) Compute the weighted average of the macroscopic anelastic strains:

$$\overline{\boldsymbol{\epsilon}}_k^{an,m} = \frac{\sum_{p=1}^K V_{pk} \overline{\boldsymbol{\epsilon}}_p^{an}}{\sum_{p=1}^K V_{pk}}, \quad \forall p \in \mathcal{F}^k, \quad (23)$$

where  $\mathcal{F}^k$  is the set of indices associated to sub-domain composing the previous domain  $\{\Omega_p\}_j$  and  $V_{pk}$  is the volume related to the sub-domain  $\Omega_{pk}$ .

ii) Compute the distance to each of the macroscopic anelastic strain  $\overline{\boldsymbol{\epsilon}}_p^{an}$  associated with the subdomain  $\Omega_{pk} \subset \{\Omega_k\}_{j+1}$ :

$$d_p = \|\overline{\boldsymbol{\epsilon}}_k^{an,m} - \overline{\boldsymbol{\epsilon}}_p^{an}\|, \quad p \in \mathcal{F}^k. \quad (24)$$

iii) Define  $\boldsymbol{\alpha}^k$  for the new cluster  $k$  such that:

$$k = \underset{p \in \mathcal{F}^k}{\operatorname{argmin}}(d_p). \quad (25)$$

As an illustrative example, we consider the domain  $\{\Omega_2\}_j$  of the cluster number 2 in Fig. 2a (The green domain). After defining new clusters, the domain becomes  $\{\Omega_2\}_{j+1} = \Omega_{12} \cup \Omega_{22} \cup \Omega_{32}$  as shown in Fig. 2b. Thus, This domain is associated with three previous internal variables vectors  $\boldsymbol{\alpha}^1$ ,  $\boldsymbol{\alpha}^2$  and  $\boldsymbol{\alpha}^3$ . Then here,  $\overline{\boldsymbol{\epsilon}}_2^{an,m} = \frac{V_{12}\overline{\boldsymbol{\epsilon}}_1^{an} + V_{22}\overline{\boldsymbol{\epsilon}}_2^{an} + V_{32}\overline{\boldsymbol{\epsilon}}_3^{an}}{V_{12} + V_{22} + V_{32}}$ , and  $\boldsymbol{\alpha}^k$  that satisfies the minimum distance among the three distances  $\min(d_1, d_2, d_3)$  is chosen to represent the cluster 2. This procedure is summarized in **Algorithm 2**.

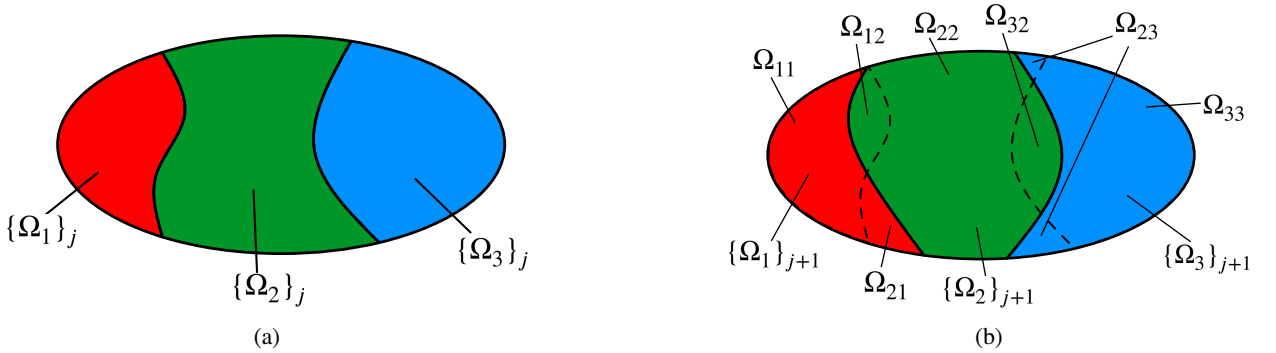
As a result, the total number of clusters defined by the user matches the actual number of RVE problems to be solved during the two-scale calculation. This drastically reduces the number of RVE problems in FE<sup>2</sup> method from  $N_{total} = (1 + \beta) \times N_G \times N_e \times N_{iter} \times N_{step}$  to  $N_{total} = (1 + \beta) \times N_{clusters} \times N_{iter} \times N_{step}$ , where  $\beta = 3$  for 2D and  $\beta = 6$  for 3D,  $N_G$  is the number of Gauss points per element,  $N_e$  is the number of elements,  $N_{clusters}$  is the number of clusters, and  $N_{step}$  is the number of loading steps. Then a speed-up factor proportional to  $N_e/N_{clusters}$  can be expected.



**Table 2**

**Algorithm 2:** Algorithm for assigning internal variables to each cluster after a new clustering step.

- 1) Loop over clusters (For  $k = 1 \dots K$ ).
  - a) Compute the weighted average of macroscopic anelastic strain  $\overline{\epsilon}_k^{an,m}$  using Eq. (23).
  - b) Compute the distance to each of the macroscopic anelastic strains vectors  $\overline{\epsilon}_p^{an}$  associated with sub-domains  $\Omega_{pk}$  using equation (24).
  - c) Define  $\alpha^k$  using Eq. (25).
- 2) End



**Figure 2:** Evolution of clusters during iterations: sub-domains used to transfer the internal variables vectors to new clusters: (a) step  $j$ , (b) new clustering at step  $j + 1$ .

### 3.2.2. linear approximation of the stress in clusters

The constant approximation of stress used in [44] is coarse and might lead to non-convergence of the macro Newton scheme. To improve the approximation of the macro stress in each cluster, a linear approximation is proposed in this work for all Gauss points with index  $i$  belonging to a cluster  $k$ :

$$\overline{\sigma}(\overline{\epsilon}^i) = \overline{\sigma}(\overline{\epsilon}^k) + \overline{C}_{tan}^k(\overline{\epsilon}^k) : (\overline{\epsilon}^i - \overline{\epsilon}^k), \quad (26)$$

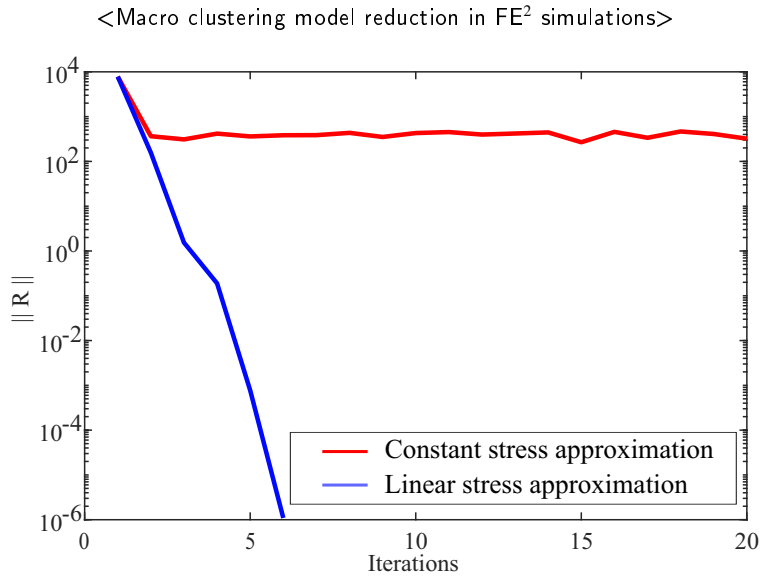
where  $\overline{\epsilon}^k$  is the spatial average of the macro strains within the cluster  $k$ . As a result, different stress values are assigned to each macroscopic Gauss point within the cluster  $k$  without any additional costs (no additional nonlinear RVE problems to be solved). It is worth noting that the linear approximation is performed in the strain-stress space, and not in the coordinate space. Then, clusters do not need to be continuous within the macro mesh.

As an illustration, the evolution of the residual for a macro Newton iterations is plotted in Fig. 3, for constant macro stress and linear approximation in each cluster. This convergence curve is taken from the example in section 4.1 (hyperelastic composite structure). While the constant stress approximation (red curve) clearly leads to non-convergence and large residual errors, the linear approximation (blue curve) leads to a fast decrease of the residual and allows convergence in a few iterations.

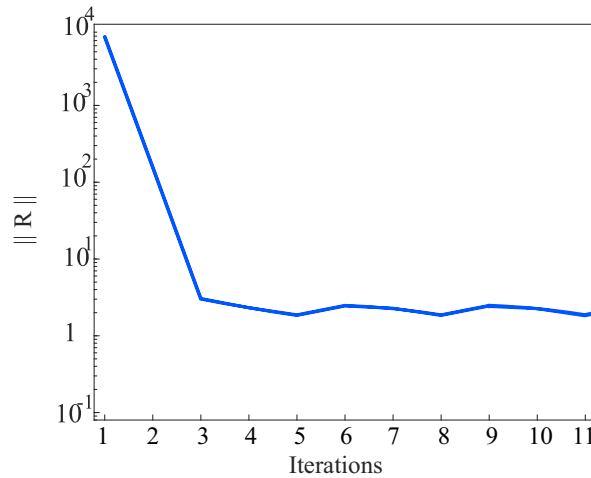
### 3.2.3. Addressing non-convergence issues due to cluster changes: a cluster freezing algorithm

In some cases, even when using the linear approximation of stress, the macro residual might not converge because of perpetual changes of the clusters distributions during the macro Newton iterations. As a result, the macro residual evolution exhibits a plateau with fluctuations and large errors. This phenomenon is illustrated in Fig. 4.

To circumvent this issue, we introduce a *cluster freezing* algorithm. The idea is that when the macro Newton convergence is not achieved within a loading increment due to changes of clusters after a given number of iterations, the distribution of clusters is maintained fixed (frozen) until the next loading increment.



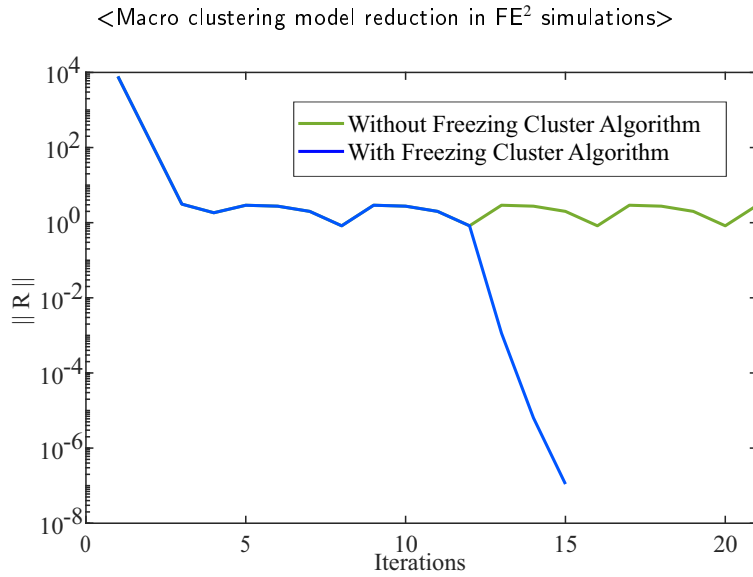
**Figure 3:** Evolution of the residual for one macro Newton-Raphson step: in red, non-convergence using constant stress approximation within the clusters; in blue, convergence when using linear approximation of the stress within the clusters.



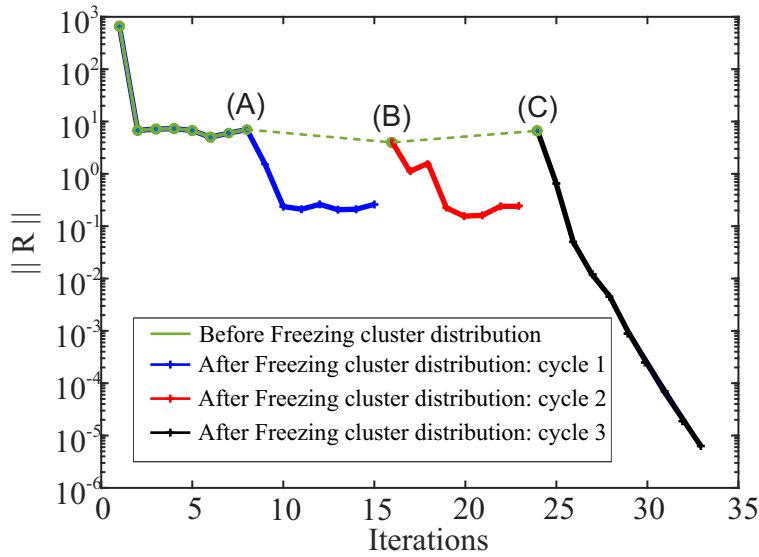
**Figure 4:** Non-convergence of the residual due to the constant changes in the clusters distribution.

This treatment effectively removes this convergence issue, as shown in Fig. 5. Again, the evolution of the macro residual during one Newton iteration is plotted. The green curve corresponds to the macro residual obtained using the linear stress approximation in each cluster, without treatment. The blue curve corresponds to the macro residual obtained using the freezing cluster algorithm. We can observe that maintaining the clustering fixed during the Newton iterations effectively allows convergences in this case.

However, we have reported that in some situations, the convergence was not achieved even when maintaining the clusters fixed. The proposed algorithm is then further elaborated by recording the displacement solution corresponding to the last cluster freezing. Then, in case of non-convergence, the Newton iterations are continued using a new cluster distribution and this previously saved displacement solution. This operation, called here a cycle, is repeated until convergence. An illustration is provided in Fig. 6. Here, the blue, red and black curves correspond to 3 cycles. At the beginning of each of these cycles, the displacement solution saved at iteration (A) is re-used at iterations (B) and (C), for which new cluster distributions are used. We can note that in this case, iteration was reached after 3 cycles (black curve). The algorithm is described in Table 3.



**Figure 5:** Evolution of the macro residual during one macro loading increment; in green: non-convergence using linear stress approximation within clusters, but perpetual changes in the clusters induce oscillations; in blue, convergence achieved using the freezing clustering algorithm.



**Figure 6:** Evolution of the macro residual during one macro loading increment; in green: convergence is not achieved due to perpetual clustering changes; (A) freezing the cluster distribution and non convergence (Cycle 1); (B) starting from the last configuration with another cluster distribution, the convergence is still non achieved (Cycle 2); (C) repeating the procedure with another cluster distribution, the convergence is finally achieved (Cycle 3).

### 3.2.4. Summary of the improved KMFE<sup>2</sup>

A summary of the improved KMFE<sup>2</sup> developed in this work is described in table 4 (**Algorithm 4**).

**Table 3**

---

**Algorithm 3:** Cluster freezing algorithm.

---

**Input:** The last two Newton iterations residuals norms of the current step  $\|\bar{R}\|_{j-1}$  and  $\|\bar{R}\|_j$ ; the counter ( $ct$ );  $f_{old}$ ;  $\bar{U}_{save}$  and  $\bar{U}_j^n$ .

- 1) If  $((ct = 0) \text{ and } (\|\bar{R}\|_j < \|\bar{R}\|_{j-1}))$ , then:  $f_{new} = f_{old}$ .
- 2) Else  $((ct \neq 0) \text{ or } (\|\bar{R}\|_j > \|\bar{R}\|_{j-1}))$ :
  - a) If  $(ct = N_{max})$ , then:
    - i.  $(ct \leftarrow 0)$
    - ii.  $f_{new} = f_{old} + 1$
  - b) else:  $f_{new} = f_{old}$  and update  $ct \leftarrow ct + 1$ .
- 3) End
- 4)
  - If  $(f_{new} > f_{old} \ \& \ f_{old} = 0)$ : **Save**  $\bar{U}_{save} = \bar{U}_j^n$ , and **freeze** clustering,
  - else if  $(f_{new} > f_{old} \ \& \ f_{old} > 0)$ :  $\bar{U}_j^n = \bar{U}_{save}$  and **Evaluate** a new clustering
  - else if  $(f_{new} = f_{old} \ \& \ f_{old} > 0)$ : **Freeze** clustering.
  - else  $(f_{new} = f_{old} = 0)$ : **Evaluate** a new clustering.

**Output:** Block cluster distribution or not; the counter ( $ct$ );  $f_{new}$ ;  $\bar{U}_{save}$  and  $\bar{U}_j^n$ .

**Remark:**  $ct$ ,  $f_{old} = 0$  and  $f_{new} = 0$  are variables that were introduced to ensure the algorithm.

---

## 4. Numerical examples

### 4.1. Hyperelastic composite structure

In this first example, a composite hyperelastic structure is considered. Even though this problem does not involve any internal variable, we present it to analyze the accuracy of the KMFE<sup>2</sup> solution using the linear approximation of the stress in each cluster as developed in section 3.2.2. The geometry and boundary conditions of the macrostructure are described in Fig. 7a, with  $L = 40$  cm,  $H = 10$  cm. A displacement  $\bar{u} = 25$  cm is incrementally prescribed on the corner ( $x = H$ ,  $y = L$ ) and 10 increments are used to reach this value in the simulation. The RVE is periodic, and its geometry is described in Fig. 8a. It consists into a circular fiber centered in a square domain, with  $d = 0.4 \times l$  and  $l = 1$  mm. A Neo-Hookean compressible model is considered for the matrix, and a Saint-Venant-Kirchhoff compressible model is considered for the fiber. The numerical parameters are chosen as  $E^m = 2$  GPa,  $\nu^m = 0.25$ ,  $E^f = 210$  GPa,  $\nu^f = 0.3$ , where  $E^m$ ,  $E^f$ ,  $\nu^m$  and  $\nu^f$  denote Young's moduli of matrix and fiber, respectively, and Poisson's moduli of matrix and fiber, respectively. The numerical formulation for the hyperelastic model is reviewed in Appendix A. The macro mesh is composed of 871 elements, and the micro (RVE) mesh is composed of 1084 elements. In all examples treated in this paper 2D linear triangular elements will be employed for both macro and micro meshes.

To validate the results, the FE<sup>2</sup> solution, i.e. solving the RVE problems in all macro elements, is taken as the reference. In Fig. 9, the KMFE<sup>2</sup> solution is compared with the reference solution for different number of clusters. The component of the stress  $\bar{\sigma}_{11}$  is evaluated in two elements of the mesh, denoted by elements A and B (see Fig. 7b). We can observe the convergence of the KMFE<sup>2</sup> solution with respect to the number of clusters. The convergence is approximatively reached for 7 clusters, while 14 clusters provides a very accurate solution as compared to the reference.

An illustration of the cluster distribution at the final load for  $K = 2, 4$  and  $K = 14$  is provided in Fig. 10.

In Fig. 11 the local von Mises stress fields obtained by the KMFE<sup>2</sup> and the reference solution are compared. It is worth noting that a relatively small number of clusters (around 14) is sufficient to accurately reproduce the reference solution. We notice that surprisingly, even 2 clusters are sufficient to capture the main features of the local field distribution, even details like the stress concentration at the top-right corner. This is due to the linear approximation of stress adopted in the clusters.

The simulations were conducted on a Matlab code on 4 processors with 2.3 GHz and 252 GB of RAM. The full FE<sup>2</sup> computations took 4h55min. Using  $K = 14$  clusters, the computations are reduced to 19.6 min leading to a speed-up ratio of 15. Using  $K = 7$  clusters, the calculations drop to 13,2 min, corresponding to a speed-up ratio of 22.

**Table 4**

**Algorithm 4:** Summary of the improved KMFE<sup>2</sup> method

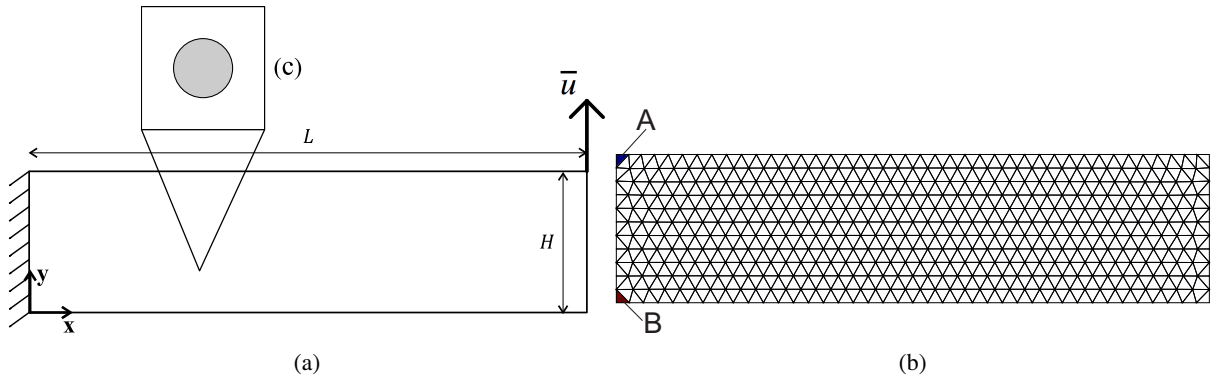
1. Initialize  $\bar{\mathbf{U}}^0 = \bar{\mathbf{U}}(\bar{\mathbf{x}}^0)$ ,  $\alpha_0(\mathbf{x}) := \{\alpha^k(\mathbf{x}^0) \mid k = 1 \dots K\}$ .
2. **Loop** over all time steps: for  $n = 1 \dots N_{step}$ 
  - (a) Initialize iteration  $j = 0$ , a counter  $ct = 0$ ,  $\bar{\mathbf{U}}_{save} = \mathbf{0}$ ,  $f_{old} = 0$  and  $f_{new} = 0$ .
  - (b)  $\bar{\mathbf{U}}_j^n = \bar{\mathbf{U}}^{n-1}$
  - (c) **While**  $Err > \epsilon_{tol}$ 
    - c.1 Update iteration:  $j = j + 1$
    - c.2 If  $j > 1$ , Given  $ct$ ,  $\bar{\mathbf{U}}_{save}$ ,  $f_{old}$ ,  $\|\bar{\mathbf{R}}\|_j$  and  $\|\bar{\mathbf{R}}\|_{j-1}$ , apply **Algorithm 3**.
    - c.3 Compute strain vectors for all macro integration points  $\{\bar{\boldsymbol{\epsilon}}^i(\bar{\mathbf{U}}_j^n), i = 1 \dots N_G\}$ .
      - If the clustering is blocked: Compute  $\{\bar{\boldsymbol{\epsilon}}^k \mid k = 1 \dots K\}$  using equation (15), then **Go** to (c.6).
      - Else: **Continue** to (c.4).
    - c.4 Evaluate a new clustering through **Algorithm 1**. **Output:**  $\{\bar{\boldsymbol{\epsilon}}^k \mid k = 1 \dots K\}$ , and  $S^k$ .
    - c.5 Given  $\alpha_{n-1}(\mathbf{x})$ , define the representative states of fields  $\{\alpha^k(\mathbf{x}) \mid k = 1 \dots K\}$  using **Algorithm 2**.
    - c.6 **Loop** over all clusters (For  $k = 1 \dots K$ )
      - Solve microscopic problem (3)-(5). **Output:**  $\bar{\boldsymbol{\sigma}}^k$ ,  $\bar{\mathbf{C}}_{tan}^k$  and  $\alpha^k(\mathbf{x})$
      - Compute stress  $\bar{\boldsymbol{\sigma}}^i$ ,  $\forall i \in S^k$  using equation (26).
      - Assign the computed stress  $\bar{\boldsymbol{\sigma}}^i$  and the tangent tensor  $\bar{\mathbf{C}}_{tan}^i = \bar{\mathbf{C}}_{tan}^k$ ,  $\forall i \in S^k$  using (26).
    - c.7 Compute  $\bar{\mathbf{R}}^e$  and  $\bar{\mathbf{K}}_{tan}^e$ .
    - c.8 Assemble  $\bar{\mathbf{R}}$  and  $\bar{\mathbf{K}}_{tan}$ .
    - c.9 Compute  $\Delta \bar{\mathbf{U}}$  by solving equation (10).
    - c.10 Update  $\bar{\mathbf{U}}_{j+1}^n = \bar{\mathbf{U}}_j^n + \Delta \bar{\mathbf{U}}$ .
    - c.11 if ( $f_{new} > f_{old}$  &  $f_{new} \neq 1$ ):  $\bar{\mathbf{U}}_{save} = \bar{\mathbf{U}}_{j+1}^n$ . **Continue** to (c.12)
    - c.12  $Err = \|\bar{\mathbf{R}}\|$ , and store  $\|\bar{\mathbf{R}}\|_j = \|\bar{\mathbf{R}}\|$
    - c.13  $f_{old} = f_{new}$ .
  - (d) End
3. Update  $\bar{\mathbf{U}}^n = \bar{\mathbf{U}}_{j+1}^n$  and  $\alpha_n(\mathbf{x}) = \{\alpha^k(\mathbf{x}) \mid k = 1 \dots K\}$
4. Update step:  $n = n + 1$ .
5. End

## 4.2. Viscoelastic composite structure

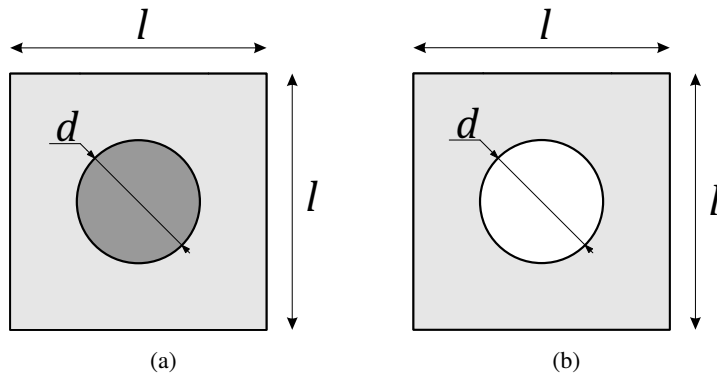
### 4.2.1. Convergence and accuracy analysis

The objective of this new example is to apply the proposed KMFE<sup>2</sup> method for a material involving micro internal variables and history-dependent behavior, and investigate the procedure described in section 3.2.1. A plate with a centered hole is considered, as depicted in Fig. 13. Due to the problem symmetry, only a quarter of the plate is modeled. The RVE of the material composing the plate is made of a viscoelastic matrix and linear stiff inclusions. The dimensions of the plate are  $L = H = 80$  cm and the radius of the central hole is  $R = 15$  cm (see Fig. 8b). A displacement  $\bar{u}(t)$  is applied according to Fig. 12 on the upper end of the plate, with maximum value 0.5 mm. Two macro meshes were used: one with 1027 elements (medium mesh, Fig. 13b), and one with 3217 elements (fine mesh, Fig. 13c). The matrix in the RVE is made of a viscoelastic material following the generalized Maxwell model using two Maxwell branches. The assumption of small strains is adopted. A linear stiff elastic model is considered within the fiber with a diameter  $d = 0.48 \times l$  and  $l = 1$  mm. The model and its integration algorithm are described in appendices C and D. The numerical parameters of the RVE materials are given in Table 5.

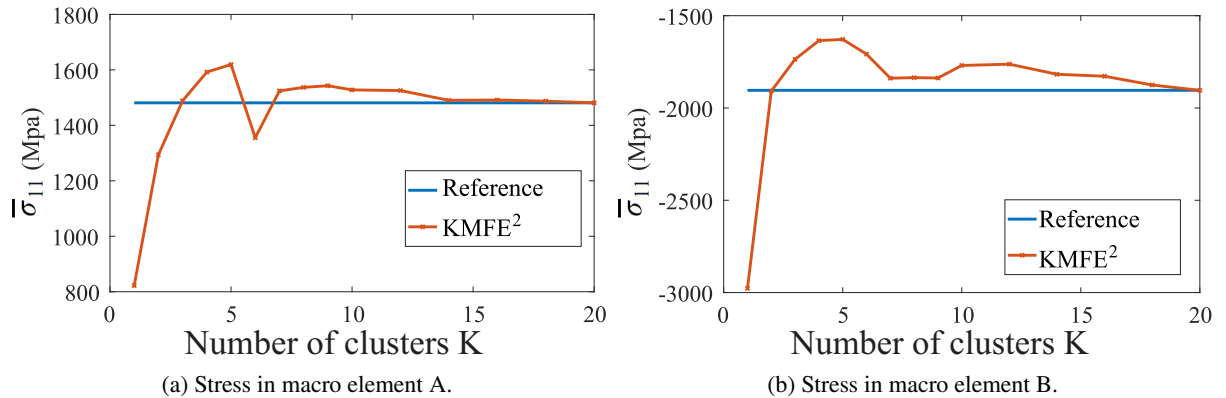
To test the accuracy of the KMFE<sup>2</sup> solution, several simulations are performed for different numbers of clusters. The results are then compared as in the previous example with the FE<sup>2</sup> reference solution. Fig. 14a and Fig. 14b show



**Figure 7:** Hyperelastic composite structure:(a) geometry of the macro scale structure and boundary conditions; (b) macro mesh; (c) RVE.



**Figure 8:** RVEs for: (a) hyperelastic and viscoelastic problems, (b) for the elastoplastic problem.

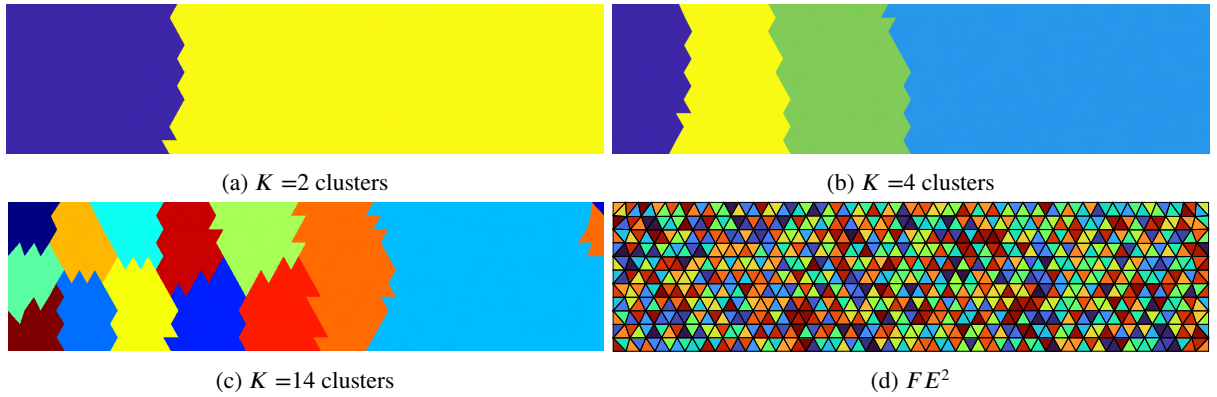


**Figure 9:** Evolution of  $\bar{\sigma}_{11}$  stress in macro elements A and B (see Fig. 7 (b)).

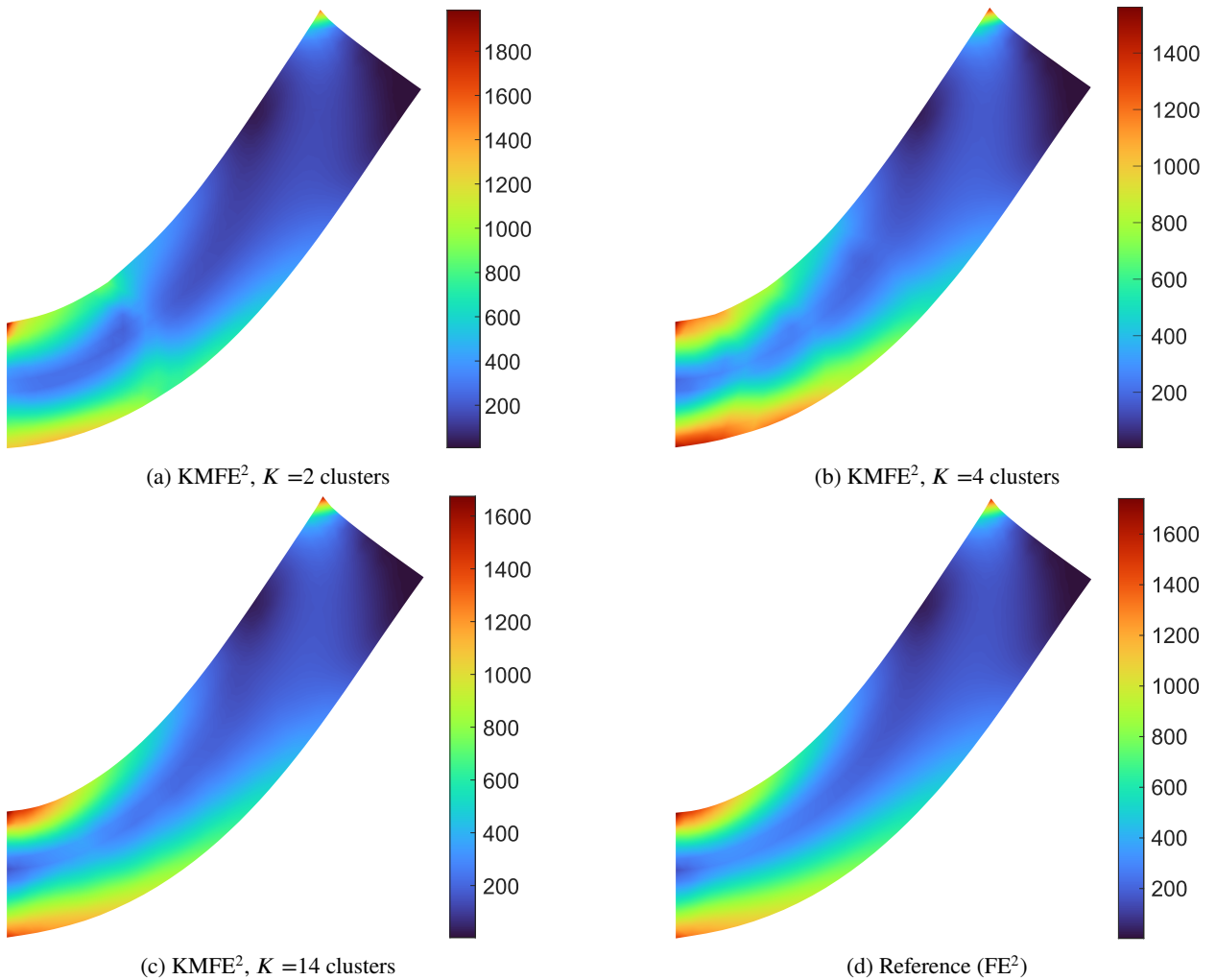
a comparison of the maximum von Mises stress in the macro structure for different numbers of clusters for the medium mesh and the fine mesh, respectively. We can note that only 25 and 50 clusters are sufficient to accurately match the full FE<sup>2</sup> for the medium and fine meshes, respectively. We can also appreciate the convergence of the response with respect to the number of clusters to the reference solution.

Figs. 15 - (a),(c),(e),(g),(i),(k) show distributions of clusters at loading peak (see Fig. 14a) and at the end of relaxation, for different numbers of clusters  $K$ . Figs. 15 - (b),(d),(f),(h),(j),(l) show the local von Mises stress

<Macro clustering model reduction in FE<sup>2</sup> simulations>



**Figure 10:** Distribution of clusters in the hyperelastic structure at final load for different number of clusters.



**Figure 11:** Comparison of von Mises stress field for different number of clusters with the reference ( $FE^2$ ) solution and deformed configuration (not magnified).

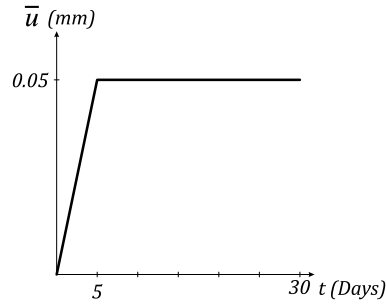


Figure 12: Evolution of the prescribed displacement  $\bar{u}$  for the viscoelastic problem.

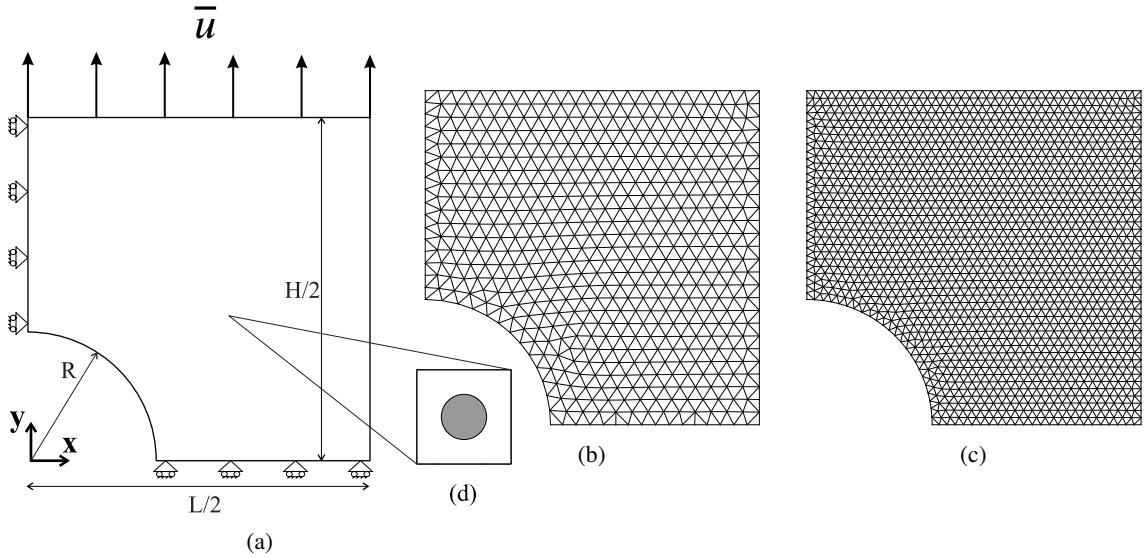


Figure 13: Viscoelastic composite structure: (a) geometry and boundary conditions; (b) coarse mesh; (c) fine mesh; (d) RVE.

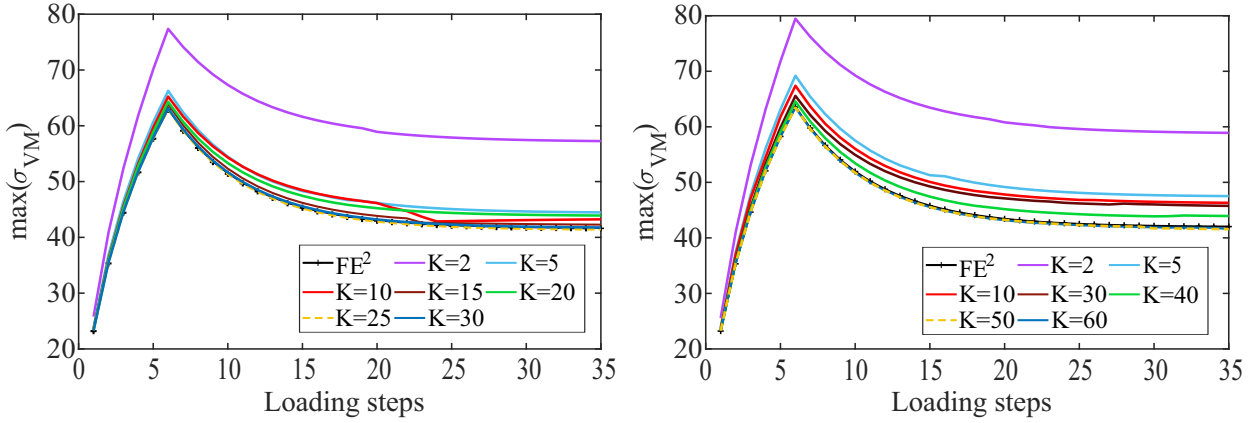
Table 5

Material parameters of the phases in the viscoelastic RVE [48].

Matrix		Inclusion	
$E_{(i=1,2)}^{(m)}$ (MPa)	4.25, 6.33	$E^{(i)}$ (MPa)	2398.4
$\tau_{(i=1,2)}^{(m)}$ (days)	4.7341, 4.0363		
$\Delta T$ (days)	0.833		
$E_{\infty}$ (MPa)	10.909		
$\nu^{(m)}$	0.256	$\nu^{(f)}$	0.28
$\sigma_0^{(m)}$ (MPa)	1	$\sigma_0^{(f)}$ (MPa)	1000
$\epsilon_0^{(m)}$	1	$\epsilon_0^{(f)}$	1
$m^{(m)}$	0.5	$m^{(f)}$	1

distribution obtained with KMFE<sup>2</sup>, which can be compared with the reference solution in Figs. 15 (m) - (n). We note the presence of some discontinuities in the local von Mises stress as compared with the reference (see e.g. Fig. 15 (l)). We note that in some situations, oscillations in the stress field obtained with KMFE<sup>2</sup> remain (see e.g. Fig. 15 (l)). This is due to the linear approximation of the stress in each cluster. Then, there is still room for improvement, which can be investigated in future works.





(a) Convergence of KMFE<sup>2</sup> w.r. to the number of clusters  $K$ , (b) Convergence of KMFE<sup>2</sup> w.r. to the number of clusters  $K$ , fine medium mesh.

**Figure 14:** Maximum von Mises for different numbers of clusters  $K$  in the viscoelastic composite structure.

**Remark:** The k-means algorithm is well-known to be sensitive to the initialization of the clusters centroids [49]. To alleviate this point, we use the initialization algorithm proposed in [50] at the beginning of the simulation. Then, for each new clustering (at each iteration and at each load increment), we use the same initial centroids. Other strategies for initialization and their impact on the efficiency may be investigated in future works.

#### 4.2.2. Comparison with sub-clustering

In our previous work [44], a sub-clustering technique was proposed to ensure that within one cluster, all RVEs were associated with close macro strain and internal variables (see section 3.1). Here, we show the gains induced by the new version of the method developed in this paper as compared to the previous one in [44], referred to as "sub-clustering". The same example than in the previous section is considered (same geometry, same material parameters). Fig. 16 shows the evolution of the number of sub-clusters with the previous version of the method throughout the simulation when choosing  $K = 2, 5, 10, 20, 30, 40, 50$  clusters.

Table 6 provides the computational times of the simulations for the fine mesh with 3217 elements (see Fig. 13c). These simulations were done for  $K = 10, 30, 40, 50$  clusters using the enhanced approach, and for  $K = 30$  clusters using the previous approach (sub-clustering). In the latter case, the total number of generated sub-clusters reached 290.

The simulations were all run on a Matlab code using 8 workers on an Intel® Xeon® Platinum 8268 CPU @2.90 GHz with 1.5TB of RAM. From Table 6, we can note that with the new version of KMFE<sup>2</sup>, a speed-up factor of 30 can be obtained for  $K = 40$  and 26.6 for  $K = 50$ , which corresponds to a better converged solution (see Fig. 14b). For this last case, it corresponds to a reduction from about 18h to 40 min. We can also note that the speed-up factor as compared to the previous version of KMFE<sup>2</sup> using sub-clustering is  $26.7/3.2 = 8.34$ , which drastically reduces the computational times. Note also that this speed-up is obtained without losing accuracy and without additional computational times associated with training or data-base construction.

### 4.3. Elastoplastic composite structures

#### 4.3.1. Rectangular sample

In this next example, a composite elastoplastic structure is considered. The geometry of the macro structure, of the RVE, and the loading are described in Fig. 17. The assumption of small strains is adopted. The macro structure is a rectangular bar whose dimensions are  $L = 20$  cm and  $H = 40$  cm. A cyclic load is prescribed on its upper end, whose evolution is provided in Fig. 17a. The macro mesh is composed of 575 linear triangular elements. The RVE is composed of a square with a centered hole as shown in fig 8b, with diameter is  $d = 0.4 \times l$ ,  $l = 1$  mm. The behaviour of the considered material within the RVE follows an elasto-plastic model with linear kinematic hardening. Its material parameters are given by:  $E = 2$  GPa,  $\nu = 0.3$ , hardening modulus  $H_p = 80$  MPa,  $\sigma_y = 24$  MPa. The model is described in Appendix E.

**Table 6**

Comparison of computational times between the new clustering approach, previous version [44] and classical FE<sup>2</sup>, viscoelastic problem, fine mesh.

Number of clusters $K$	Total number of RVE problems	Computational time (8 workers)	Speed-up factor
10	8924	18 min	60
30	25924	27 min	40
40	35044	35 min 25 s	30
50	43524	40 min	26.7
30 Sub-clustering method [44]	301160	5 h 35 min	3.2
FE <sup>2</sup> (3217)	1248196	17 h 48 min	

Fig. 20 shows stress-strain relationship in the elements A of the macro mesh (see Fig. 17c) throughout the simulation using  $K = 5, 8, 12, 16, 22$  clusters, which are compared with the reference (FE<sup>2</sup>) solution. We can note the convergence to the reference solution with respect to the number of clusters.

The local von Mises stress fields within the bar are plotted in Fig. 19 and compared with the reference solution for load steps 30 and 50 which correspond to the peak of maximum compression and second peak of traction, respectively.

Fig. 18 shows the distribution of clusters for different number of clusters at the end of the simulation.

#### 4.3.2. Elasto-plastic bracket

Finally, an application involving a more complex macro structural geometry is considered. The material and the associated RVE are similar as in the previous example. The geometry of the bracket structure is described in Fig. 21 (a). The dimensions are  $L = 80$  cm,  $W = 60$  cm,  $e = 8$  cm,  $R1 = R2 = 7$  cm,  $R3 = 12$  cm,  $d11 = 20$  cm,  $d12 = 40$  cm,  $d21 = 15$  cm and  $d22 = 30$  cm. The macro mesh contains 8482 elements. A cyclic load is prescribed on the boundary of the right-end hole according to Fig. 22. The displacements along the boundary of the left-end holes are fixed. The macro mesh has 8482 elements.

We plot the maximum von Mises stress according to the number of clusters for KMFE<sup>2</sup> in Fig. 23 in two elements of the mesh, denoted by elements A and B in Fig. 21b. These elements have been chosen as reaching the maximum absolute values for the components  $\bar{\sigma}_{12}$  and  $\bar{\sigma}_{22}$ , respectively. In this example, the reference solution is not computed due to an unreasonable estimated computational time. However, we can appreciate the convergence of the KMFE<sup>2</sup> method in both cases with respect to the number of clusters in Fig. 23. The convergence is reached for about 110 clusters here.

In Figs. 24-25, we compare the local von Mises stress for different numbers of clusters at two loading steps, which correspond to the loading maximum and minimum. We can appreciate the convergence in both cases with respect to the number of clusters, and that with 110 clusters, a good accuracy is obtained with respect to the converged solution (120 clusters).

Table 7 highlights the speed-up factors obtained for this example, where the computational time of the reference (FE<sup>2</sup>) solution is estimated. The estimation is made by taking a typical time related to solving one RVE problem and estimating the total number of RVE problems to be solved, based on the total number of elements in the macro mesh. The times were estimated using 8 workers for  $K = 80, 110, 120$  clusters as well as for FE<sup>2</sup>. We can note that in this case, using  $K = 110$  clusters, which corresponds to a fairly converged solution (see Fig. 20), the computational times are reduced from about one month to 1 day and 15 hours, corresponding to a speed-up factor of 19.

## 5. Conclusions

In this work, a method was proposed to reduce computational times in history-dependent, nonlinear multiscale problems. In contrast to most existing Reduced Order Models (ROM)-based methods in the context of multiscale analysis, where ROMs usually operate at the micro scale, like e.g. Proper Orthogonal Decomposition or machine-learning ROMs, the present technique reduces the macro scale model directly by selecting macro Gauss points in close mechanical states using an unsupervised clustering machine learning technique. The method has been initially developed in [44]. In the present paper, several important improvements were proposed, including: (i) a linear stress approximation in each cluster to avoid macro convergence issues; (ii) a new definition of the clustering vectors based

**Table 7**

Comparison of computation times between KMFE<sup>2</sup>, and FE<sup>2</sup> solutions for the bracket elastoplastic problem, using 8 workers.

Number of clusters $K$	Total number of RVE problems	Computational times	Speed-up factor
80	418880	1 d 8 h	22.87
110	504000	1 d 14 h	19.26
120	559680	1 d 16 h 33 min	18
FE <sup>2</sup> (Estimated, 8482)	$\approx 10178400$	30 d 12 h	

on macro anelastic strains, allowing to drastically reduce the computational times and (iii) an algorithm to circumvent spurious cluster effects-convergence issues. As a result, no learning stage, data base construction, or preliminary off-line calculations are required, while maintaining typical speed-up factors around 20 as compared to classical FE<sup>2</sup>. The technique has been applied successfully to hyperelastic, viscoelastic and elastoplastic composites. Future developments will include micro-scale damage, and coupling the technique with micro ROM-based approaches to further reduce the computational times in nonlinear multiscale simulations. Then, the present work contributes to the field of ROMs in the context of multiscale analysis, but the method we propose here operates at the macro scale and not at the micro scale, which makes it complementary with existing micro-scale ROM techniques.

## 6. Acknowledgement

The authors gratefully acknowledge the financial support from the Bosch Research Foundation.

### A. Appendix: FE<sup>2</sup> method at finite strains

#### A.1. Macro scale problem

The macro scale problem is defined in an open domain  $\bar{\Omega} \in \mathbb{R}^D$ , whose boundary is denoted by  $\partial\bar{\Omega}$  in its reference configuration, and where  $D$  is the domain dimension. In the absence of body forces, the macro scale problem equations are given by:

$$\nabla \cdot \bar{\mathbf{P}}(\bar{\mathbf{X}}) = 0 \quad \text{in } \bar{\Omega}, \quad (27)$$

where  $\bar{\mathbf{X}}$  is a material point coordinate in the structure,  $\bar{\mathbf{P}}$  is the macro first Piola-Kirchhoff stress tensor. The boundary conditions read:

$$\bar{\mathbf{u}}(\bar{\mathbf{X}}) = \bar{\mathbf{u}}^*(\bar{\mathbf{X}}) \quad \text{on } \partial\bar{\Omega}_u, \quad \text{and} \quad \bar{\mathbf{P}} \cdot \bar{\mathbf{N}} = \bar{\mathbf{F}}^* \quad \text{on } \partial\bar{\Omega}_F, \quad (28)$$

where  $\bar{\mathbf{u}}$  denotes displacements at the macro scale,  $\bar{\mathbf{u}}^*$  and  $\bar{\mathbf{F}}^*$  denote prescribed displacements and forces over the macro Dirichlet,  $\partial\bar{\Omega}_u$  and Neumann  $\partial\bar{\Omega}_F$  boundaries, respectively, and  $\bar{\mathbf{N}}$  is the unit normal vector to the external boundary  $\partial\bar{\Omega}$  in the reference configuration.

#### A.2. Micro scale problem

We consider an RVE defined in an open domain  $\Omega$ , whose boundary is denoted by  $\partial\Omega$ . Assuming absence of body forces, the equilibrium equations read:

$$\nabla \cdot \mathbf{P}(\mathbf{X}) = 0 \quad \text{in } \Omega, \quad (29)$$

where  $\mathbf{P}$  is the micro first Piola-Kirchhoff tensor and  $\mathbf{X}$  is a material point coordinate in the reference configuration. At the micro scale, the constitutive law is an arbitrary nonlinear function in the form

$$\mathbf{P}(\mathbf{X}, t) = \mathcal{F}(\mathbf{F}(\mathbf{X}, t), \boldsymbol{\alpha}(\mathbf{X}, t)), \quad (30)$$

where  $\mathbf{F} = \mathbf{1} + \nabla(\mathbf{u})$  is the deformation gradient tensor and  $t$  denotes a pseudo-time (quasi-static evolution parameter). In (30),  $\boldsymbol{\alpha}$  denotes a vector of internal variables associated with history-dependent phenomena like e.g. plasticity, damage, etc. The problem is completed with boundary conditions expressed by:

$$\mathbf{u} = \left( \bar{\mathbf{F}} - \mathbf{1} \right) \mathbf{X} + \tilde{\mathbf{u}}(\mathbf{X}) \text{ on } \partial\Omega, \quad (31)$$

where  $\tilde{\mathbf{u}}(\mathbf{X})$  is a periodic fluctuation. Equation (31) satisfies

$$\bar{\mathbf{F}} = \langle \mathbf{F}(\mathbf{X}) \rangle. \quad (32)$$

In addition, the macro stress is related to the micro stress by spatial averaging as:

$$\bar{\mathbf{P}} = \langle \mathbf{P}(\mathbf{X}) \rangle. \quad (33)$$

### A.3. Linear approximation in clusters for finite strains

The linear approximation of the first Piola Kirchhoff stress for finite strains for all Gauss points with index  $i$  belonging to a cluster  $k$  is expressed as follows:

$$\bar{\mathbf{P}} \left( \bar{\mathbf{F}}^i \right) = \bar{\mathbf{P}} \left( \bar{\mathbf{F}}^k \right) + \bar{\mathbb{L}}_{tan}^k \left( \bar{\mathbf{F}}^k \right) : \left( \bar{\mathbf{F}}^i - \bar{\mathbf{F}}^k \right), \quad (34)$$

where  $\bar{\mathbf{F}}^k$  is the spatial average of the macro deformation gradients within the cluster  $k$ , and  $\bar{\mathbb{L}}_{tan}^k \left( \bar{\mathbf{F}}^k \right)$  is evaluated by perturbation et element  $i$ :

$$\left( \bar{\mathbb{L}}_{tan}^k \right)_{abcd} \simeq \frac{\bar{\mathbf{P}}_{ab} \left( \bar{\mathbf{F}}^k + \boldsymbol{\delta}_{cd} \right) - \bar{\mathbf{P}}_{ab} \left( \bar{\mathbf{F}}^k \right)}{\epsilon}, \quad (35)$$

where

$$\boldsymbol{\delta}_{cd} = \epsilon \left( \mathbf{e}_c \otimes \mathbf{e}_d \right), \quad (36)$$

$\epsilon$  being a small numerical parameter, and  $\mathbf{e}_i$  denoting orthogonal vectors.

**Remark:** In the present work, we do not consider elastoplasticity for the finite strains example, this case will be reported to future studies. Then, the clustering vector for the 2D hyperelastic problem is defined as:

$$\mathbf{v}_i = \left\{ \bar{F}_{11}^i, \bar{F}_{12}^i, \bar{F}_{21}^i, \bar{F}_{22}^i \right\}, \forall i \in S^k. \quad (37)$$

## B. Appendix: Hyperelastic model: Neo-Hookean

The compressible Neo-Hookean model that is considered in this article is described by the following potential function:

$$\psi(\mathbf{C}) = \frac{1}{2} \lambda \{ \log(J) \}^2 - \mu \log(J) + \frac{1}{2} \mu [Tr(\mathbf{C}) - 3], \quad (38)$$

where  $\mathbf{C} = \mathbf{F}^T \mathbf{F}$  is the right Cauchy-Green strain tensor,  $J = \det(\mathbf{F})$  is the jacobien,  $\lambda = \frac{E\nu}{[(1+\nu)(1-2\nu)]}$  and  $\mu = \frac{E}{[2(1+\nu)]}$  are Lamé's coefficients. The constitutive law is given in the Lagrangien description by:

$$\mathbf{S}(\mathbf{C}) := 2 \frac{\psi(\mathbf{C})}{\mathbf{C}} = \lambda \log(J) \mathbf{C}^{-1} + \mu (\mathbf{I} - \mathbf{C}^{-1}), \quad (39)$$

where  $\mathbf{S}$  is the second Piola-Kirchhoff stress tensor related to Cauchy stress by  $\mathbf{S} = J \mathbf{F}^{-1} \boldsymbol{\sigma} \mathbf{F}^{-T}$ , and to the first Piola-Kirchhoff stress  $\mathbf{P}$  by  $\mathbf{S} = \mathbf{F}^{-1} \mathbf{P}$ .

### C. Appendix: Viscoelastic model: Generalized relaxation model

The viscoelastic model that is used in this paper is the generalized Maxwell relaxation model (Fig. 26). We give here a description of the model restricted to the small strains assumption. For more details, see [51]. The model problem is formulated by the following equations:

$$\boldsymbol{\sigma} = \frac{\partial}{\partial \boldsymbol{\varepsilon}} W^0(\boldsymbol{\varepsilon}) - \sum_{i=1}^N \mathbf{q}_i, \quad \dot{\mathbf{q}}_i + \frac{1}{\tau_i} \mathbf{q}_i = \frac{\gamma_i}{\tau_i} \frac{\partial}{\partial \boldsymbol{\varepsilon}} W^0, \quad \lim_{t \rightarrow -\infty} \mathbf{q}_i(t) = 0, \quad (40)$$

where  $W^0(\boldsymbol{\varepsilon})$  is the (initial) stored-energy function,  $\mathbf{q}_i$  is the stress-like set of internal variables that replaces the visous strains  $\boldsymbol{\alpha}_i$

$$\mathbf{q}_i := E_i \boldsymbol{\alpha}_i, \quad i = 1, 2, \dots, N, \quad (41)$$

and,  $0 \leq \gamma_i \leq 1$  are the non-dimensional (relative) moduli defined by:

$$\begin{cases} \gamma_i := E_i/E_0, & i = 1, 2, \dots, N, \\ \gamma_\infty := E_\infty/E_0, \end{cases} \quad (42)$$

where  $E_i$  and  $E_\infty$  are stiffness constants,  $E_0$  is the initial modulus and  $\tau_i$  are the relaxation times defined as:

$$\begin{cases} E_0 := E_\infty + \sum_{i=1}^N E_i, \\ \tau_i := \eta_i/E_i, & i = 1, 2, \dots, N. \end{cases} \quad (43)$$

Next, we present an extension of the model (40) in 3D. Since any arbitrary convex function  $W^0$  of  $\boldsymbol{\varepsilon}$  can be considered, we choose herein the following isotropic nonlinear elastic potential:

$$W^0(\boldsymbol{\varepsilon}) = \frac{9}{2} \kappa \varepsilon_m^2 + \frac{\varepsilon_0 \sigma_0}{1+m} \left( \frac{\varepsilon_{eq}}{\varepsilon_0} \right)^{1+m} \quad (44)$$

where  $\varepsilon_m := Tr(\boldsymbol{\varepsilon})/3$  is the hydrostatic strain,  $\varepsilon_{eq} = \sqrt{\frac{2}{3} \mathbf{e} : \mathbf{e}}$  is the equivalent strain and  $\mathbf{e} = \boldsymbol{\varepsilon} - \varepsilon_m \mathbf{1}$  is the deviatoric strain. We use the notation  $\boldsymbol{\sigma}^0 := \frac{\partial W^0(\boldsymbol{\varepsilon})}{\partial \boldsymbol{\varepsilon}}$ . From Eq. (40), the stress response for 3D model is given by:

$$\boldsymbol{\sigma}(t) = \boldsymbol{\sigma}^0(t) - \sum_{i=1}^N \mathbf{q}_i(t), \quad (45)$$

where  $\{\mathbf{q}_1, \dots, \mathbf{q}_N\}$  is a set of internal variables such as :

$$\mathbf{q}_i = \frac{\gamma_i}{\tau_i} \int_{-\infty}^t \exp[-(t-s)/\tau_i] \{\partial_{\boldsymbol{\varepsilon}} W^0[\boldsymbol{\varepsilon}(s)]\} ds. \quad (46)$$

Substituting  $\mathbf{q}_i$  in equation (45) yields:

$$\boldsymbol{\sigma}(t) = \frac{\partial}{\partial \boldsymbol{\varepsilon}} W^0 - \sum_{i=1}^N \frac{\gamma_i}{\tau_i} \int_{-\infty}^t \exp[-(t-s)/\tau_i] \{\partial_{\boldsymbol{\varepsilon}} W^0[\boldsymbol{\varepsilon}(s)]\} ds$$

which can be re-written as

$$\boldsymbol{\sigma}(t) = \frac{\partial}{\partial \boldsymbol{\varepsilon}} W^0 - \sum_{i=1}^N \gamma_i \int_{-\infty}^t \frac{d(\exp[-(t-s)/\tau_i])}{ds} \{\partial_{\boldsymbol{\varepsilon}} W^0[\boldsymbol{\varepsilon}(s)]\} ds$$

Then, by applying an integration by parts, and by using the initial condition in eq. (40), we obtain :

$$\boldsymbol{\sigma}(t) = \frac{\partial}{\partial \boldsymbol{\varepsilon}} W^0 - \sum_{i=1}^N \gamma_i \frac{\partial}{\partial \boldsymbol{\varepsilon}} W^0 + \sum_{i=1}^N \gamma_i \int_{-\infty}^t \exp[-(t-s)/\tau_i] \frac{d}{ds} \{ \partial_{\boldsymbol{\varepsilon}} W^0 [\boldsymbol{\varepsilon}(s)] \} ds. \quad (47)$$

Combining Eq. (42) and Eq.(43) leads to the expression for the relative moduli  $\sum_{i=1}^N \gamma_i + \gamma_{\infty} = 1$ . Thus, the constitutive equation can be written as the following convolution integral:

$$\boldsymbol{\sigma}(t) = \int_{-\infty}^t g(t-s) \frac{d}{ds} \{ \partial_{\boldsymbol{\varepsilon}} W^0 [\boldsymbol{\varepsilon}(s)] \} ds,$$

where :  $g(t) := \gamma_{\infty} + \sum_{i=1}^N \gamma_i \exp[-t/\tau_i]$ , which finally gives

$$\boldsymbol{\sigma}(t) = \gamma_{\infty} \frac{\partial}{\partial \boldsymbol{\varepsilon}} W^0 + \sum_{i=1}^N \gamma_i \mathbf{h}^{(i)}(t), \quad (48)$$

where

$$\mathbf{h}^{(i)}(t) = \int_{-\infty}^t \exp[-(t-s)/\tau_i] \frac{d}{ds} \{ \partial_{\boldsymbol{\varepsilon}} W^0 [\boldsymbol{\varepsilon}(s)] \} ds. \quad (49)$$

Note that the initial stored energy  $W^0(\boldsymbol{\varepsilon})$  can be written as a split between an elastic volumetric response  $U^0(Tr(\boldsymbol{\varepsilon}))$  and a deviatoric response  $\overline{W}^0(\mathbf{e})$  as in [51]. In this work we did not consider this separation into account.

## D. Appendix: Integration algorithm for the viscoelastic problem

Let  $[T_0, T] \subset \mathbb{R}$ , with  $T > 0$  and  $T > T_0$ , be the time of interest. Further, we consider the following partition of time interval:

$$[T_0, T] = \bigcup_{n \in \mathbb{I}} [t_n, t_{n+1}], \quad t_{n+1} = t_n + \Delta t_n.$$

The stress-strain relationship at the Gauss points  $x_l$  with  $l = 1, 2, \dots, n_{Gauss}$  is defined through the convolution equation:

$$\boldsymbol{\sigma}_l(t) = \int_{T_0}^t g(t-s) \frac{d}{ds} \{ \partial_{\boldsymbol{\varepsilon}} W^0 [\boldsymbol{\varepsilon}(s)] \} ds. \quad (50)$$

We define the set of  $N$  internal variables as:

$$\mathbf{h}^{(i)}(t) = \int_{T_0}^t \exp[-(t-s)/\tau_i] \frac{d}{ds} \{ \partial_{\boldsymbol{\varepsilon}} W^0 [\boldsymbol{\varepsilon}(s)] \} ds, \quad i = 1 \dots N. \quad (51)$$

We consider algorithmic internal variables stored at Gauss points in a finite-element mesh, and we consider a strain-driven algorithm:

i. Given the "initial stress"  $\mathbf{S}_n^0$  and the algorithmic internal variables  $\mathbf{h}_n^{(i)}$  be given at time  $t_n \in [T_0, T]$ :

$$\{ \mathbf{S}_n^0, \mathbf{h}_n^{(i)}, \quad i = 1, 2, \dots, N \}$$

while:

$$\mathbf{S}_n^0 := \partial_{\boldsymbol{\varepsilon}} W^0 [\boldsymbol{\varepsilon}(t_n)] = \kappa Tr(\boldsymbol{\varepsilon}) \mathbf{I} + \frac{2}{3} \frac{\varepsilon_0}{\sigma_0} \left( \frac{\boldsymbol{\varepsilon}_{eq}}{\varepsilon_0} \right)^{m-1} \boldsymbol{\varepsilon}_d \quad (52)$$

ii. Let  $\boldsymbol{\varepsilon}_{n+1} = \boldsymbol{\varepsilon}_n + \Delta \boldsymbol{\varepsilon}_n$ , where  $\Delta \boldsymbol{\varepsilon}_n$  is strain increment at one Gauss point.

iii. The objectif is to compute the stress  $\boldsymbol{\sigma}_{n+1}$  at time  $t_{n+1} = t_n + \Delta t_n$ .

The key in this integration algorithm is semigroup property. The standard semigroup property holds for the exponential function  $\exp[(t + \Delta t)/a] = \exp(\Delta t/a)\exp(t/a)$ , for any constants  $\Delta t$ , and  $a$  in  $\mathbb{R}$ . Thus,  $\mathbf{h}^{(i)}(t_{n+1})$  is determined by the integration of (51) in terms of  $\mathbf{h}^{(i)}(t_n)$  and an integral over the time step  $[t_n, t_{n+1}]$ :

$$\mathbf{h}_{n+1}^{(i)} = \int_{T_0}^{t_{n+1}} \exp[-(t_n + \Delta t_n - s)/\tau_i] \frac{d}{ds} \mathbf{S}_s^0 ds = \int_{T_0}^{t_{n+1}} \exp[-(\Delta t_n)/\tau_i] \exp[-(t_n - s)/\tau_i] \frac{d}{ds} \mathbf{S}_s^0 ds \quad (53)$$

$$\mathbf{h}_{n+1}^{(i)} = \int_{T_0}^{t_n} \exp[-(\Delta t_n)/\tau_i] \exp[-(t_n - s)/\tau_i] \frac{d}{ds} \mathbf{S}_s^0 ds + \int_{t_n}^{t_{n+1}} \exp[-(t_n + \Delta t_n - s)/\tau_i] \frac{d}{ds} \mathbf{S}_s^0 ds \quad (54)$$

$$\mathbf{h}_{n+1}^{(i)} = \exp(-\Delta t_n/\tau_i) \mathbf{h}_n^{(i)} + \int_{t_n}^{t_{n+1}} \exp[-(t_n + \Delta t_n - s)/\tau_i] \frac{d}{ds} \mathbf{S}_s^0 ds, \quad (55)$$

then using the midpoint rule for the integral part in (55), we obtain the approximation of  $\mathbf{h}_{n+1}^{(i)}$  as follows:

$$\mathbf{h}_{n+1}^{(i)} = \exp(-\Delta t_n/\tau_i) \mathbf{h}_n^{(i)} + \exp(-\Delta t_n/2\tau_i) (\mathbf{S}_{n+1}^0 - \mathbf{S}_n^0). \quad (56)$$

Thus, the expression of the stress tensor becomes:

$$\boldsymbol{\sigma}_{n+1} = \gamma_\infty \mathbf{S}_{n+1}^0 + \sum_{i=1}^N \gamma_i \mathbf{h}_{n+1}^{(i)}. \quad (57)$$

The tangent modulus tensor is given by:

$$\mathbf{C}_{n+1} := \partial_{\boldsymbol{\varepsilon}_{n+1}} \boldsymbol{\sigma}_{n+1}, \quad (58)$$

where the elastic modulus tensor is given by:

$$\mathbf{C}_{n+1}^0 := \partial_{\boldsymbol{\varepsilon}_{n+1}} \mathbf{S}_{n+1}^0. \quad (59)$$

Finally, (58) can be written as follows:

$$\mathbf{C}_{n+1} := (\gamma_\infty + \sum_{i=1}^N \gamma_i \exp[-\delta t_n/2\tau_i]) \mathbf{C}_{n+1}^0. \quad (60)$$

## E. Appendix: Elastoplastic model

In this work, an elastoplastic von Mises model with linear kinematic hardening is considered: The free Gibbs energy is given by:

$$2\rho\psi = (\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^p) : \mathbb{C} : (\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^p) + \frac{2}{3} H \boldsymbol{\varepsilon}^p : \boldsymbol{\varepsilon}^p, \quad (61)$$

with  $\mathbb{C} = 3K\mathbb{P}^H + 2\mu\mathbb{P}^D$ ,  $K$  and  $\mu$  are the Lamé's coefficients,  $\mathbb{P}^H = \frac{1}{3}\mathbf{1} \otimes \mathbf{1}$ ,  $\mathbb{P}^D = \mathbf{1} - \mathbb{P}^H$ ,  $\boldsymbol{\varepsilon}^p$  is the plastic deformation and  $H$  is the hardening coefficient. The stress and the thermodynamic force associated to the plastic deformation is obtained as:

$$\boldsymbol{\sigma} = \rho \frac{\partial \psi}{\partial \boldsymbol{\varepsilon}} = (3K\mathbb{P}^H + 2\mu\mathbb{P}^D) : (\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^p), \quad (62)$$

$$\boldsymbol{\sigma}^D - \mathbf{X} = -\rho \frac{\partial \psi}{\partial \boldsymbol{\varepsilon}^p}, \quad \mathbf{X} = \frac{2}{3} H \boldsymbol{\varepsilon}^p. \quad (63)$$

The second-order tensor  $\mathbf{X}$  describes the translation of the elastic domain. The yield condition is written as:

$$f = \sqrt{\frac{2}{3}} \|\boldsymbol{\sigma}^D - \mathbf{X}\| - \sigma_0 \leq 0, \quad (64)$$

where  $\boldsymbol{\sigma}^D$  is the deviatoric part of  $\boldsymbol{\sigma}$  and  $\sigma_y$  is the yield stress. The plastic evolution is given by

$$f = 0, \quad \frac{\partial f}{\partial \boldsymbol{\sigma}} : \partial \boldsymbol{\sigma} > 0. \quad (65)$$

Then,

$$d\boldsymbol{\varepsilon}^p = \frac{\partial f}{\partial(\boldsymbol{\sigma}^D - \mathbf{X})} d\lambda = \sqrt{\frac{3}{2}} \frac{\boldsymbol{\sigma}^D - \mathbf{X}}{\|\boldsymbol{\sigma}^D - \mathbf{X}\|} d\lambda, \quad d\lambda > 0, \quad (66)$$

where  $d\lambda$  is a consistency parameter. The cumulative plasticity is defined by  $dp := \sqrt{\frac{2}{3}} \|d\boldsymbol{\varepsilon}^p\|$ , thus  $d\lambda = dp$ . Then,  $dp$  is defined by using the consistency equation  $df = \frac{\partial f}{\partial(\boldsymbol{\sigma}^D - \mathbf{X})} = 0$ :

$$\frac{(\boldsymbol{\sigma}^D - \mathbf{X})}{\|\boldsymbol{\sigma}^D - \mathbf{X}\|} : d\mathbf{X} = \frac{(\boldsymbol{\sigma}^D - \mathbf{X})}{\|\boldsymbol{\sigma}^D - \mathbf{X}\|} : d\boldsymbol{\sigma}, \quad (67)$$

and by using (62), (63) and (66):

$$\sqrt{\frac{2}{3}} H dp = \frac{(\boldsymbol{\sigma}^D - \mathbf{X})}{\|\boldsymbol{\sigma}^D - \mathbf{X}\|} : \mathbb{C} : (d\boldsymbol{\varepsilon} - d\boldsymbol{\varepsilon}^p), \quad (68)$$

which becomes:

$$dp = \frac{\frac{(\boldsymbol{\sigma}^D - \mathbf{X})}{\|\boldsymbol{\sigma}^D - \mathbf{X}\|} : \mathbb{C} : d\boldsymbol{\varepsilon}}{\left(\sqrt{\frac{2}{3}} H + \frac{(\boldsymbol{\sigma}^D - \mathbf{X})}{\|\boldsymbol{\sigma}^D - \mathbf{X}\|} : \mathbb{C} : \frac{(\boldsymbol{\sigma}^D - \mathbf{X})}{\|\boldsymbol{\sigma}^D - \mathbf{X}\|}\right)}, \quad (69)$$

$$dp = \frac{2\mu}{\left(\sqrt{\frac{2}{3}} H + 2\mu\right)} \frac{(\boldsymbol{\sigma}^D - \mathbf{X})}{\|\boldsymbol{\sigma}^D - \mathbf{X}\|} : d\boldsymbol{\varepsilon}. \quad (70)$$

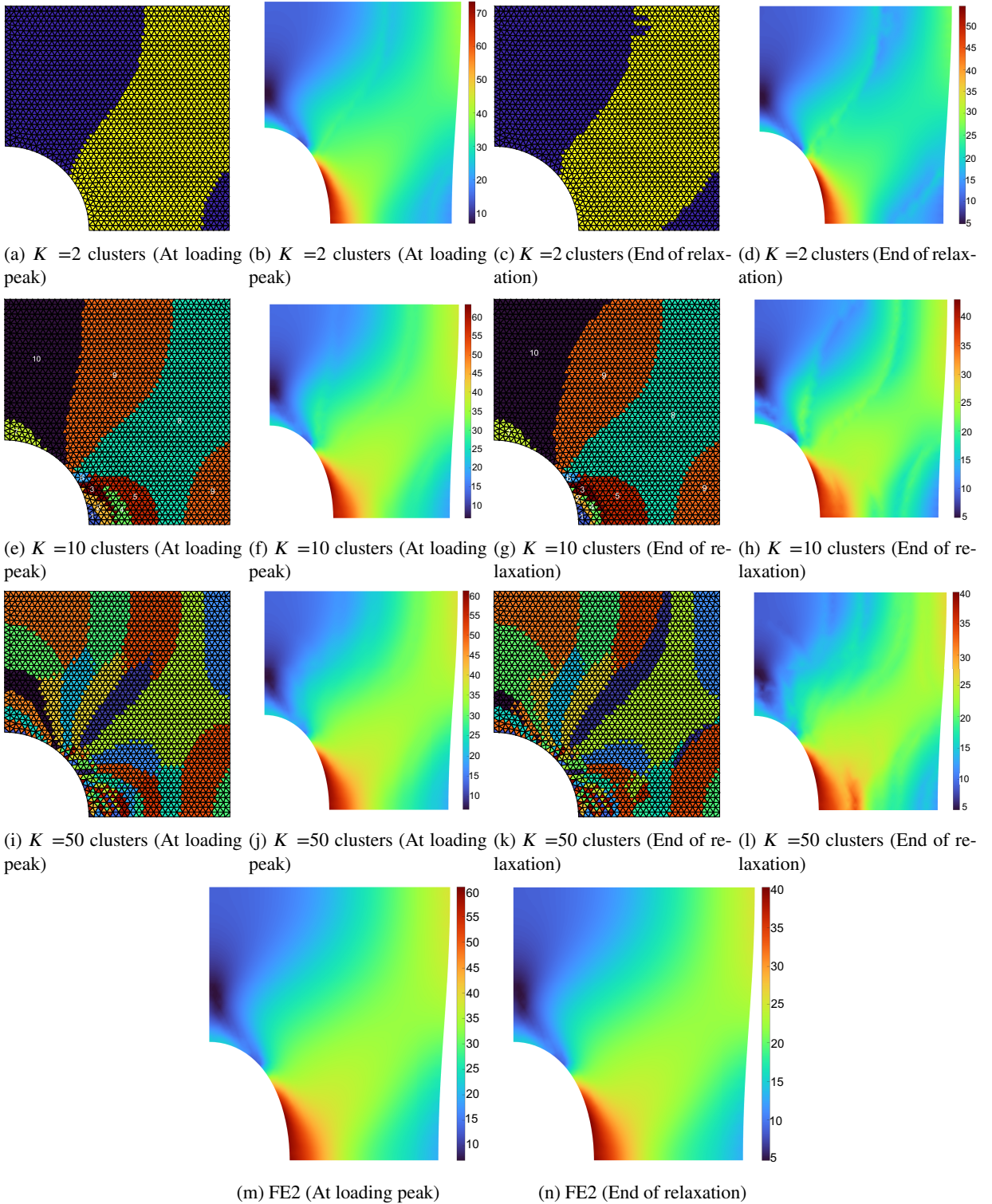
## References

- [1] S. Ghosh, K. Lee, and S. Moorthy. Two scale analysis of heterogeneous elastic-plastic materials with asymptotic homogenization and voronoi cell finite element model. *Computer methods in applied mechanics and engineering*, 132(1-2):63–116, 1996.
- [2] R.J.M. Smit, W.A.M. Brekelmans, and H.E.H. Meijer. Prediction of the mechanical behavior of nonlinear heterogeneous systems by multi-level finite element modeling. *Computer methods in applied mechanics and engineering*, 155(1-2):181–192, 1998.
- [3] F. Feyel. Multiscale FE2 elastoviscoplastic analysis of composite structures. *Computational Materials Science*, 16(1):344–354, December 1999.
- [4] F. Feyel and J-L. Chaboche. FE2 multiscale approach for modelling the elastoviscoplastic behaviour of long fibre SiC/Ti composite materials. *Computer Methods in Applied Mechanics and Engineering*, 183(3):309–330, March 2000.
- [5] K. Terada and N. Kikuchi. A class of general algorithms for multi-scale analyses of heterogeneous media. *Computer methods in applied mechanics and engineering*, 190(40-41):5427–5464, 2001.
- [6] V. Kouznetsova, M. G. D. Geers, and W. A. M. Brekelmans. Multi-scale constitutive modelling of heterogeneous materials with a gradient-enhanced computational homogenization scheme. *International Journal for Numerical Methods in Engineering*, 54(8):1235–1260, 2002. [\\_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.541](https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.541).
- [7] M.G.D. Geers and J. Yvonnet. Multiscale modeling of microstructure–property relations. *MRS Bulletin*, 41(8):610–616, 2016.
- [8] K. Raju, T-E. Tay, and V. B. C. Tan. A review of the fe 2 method for composites. *Multiscale and Multidisciplinary Modeling, Experiments and Design*, 4:1–24, 2021.

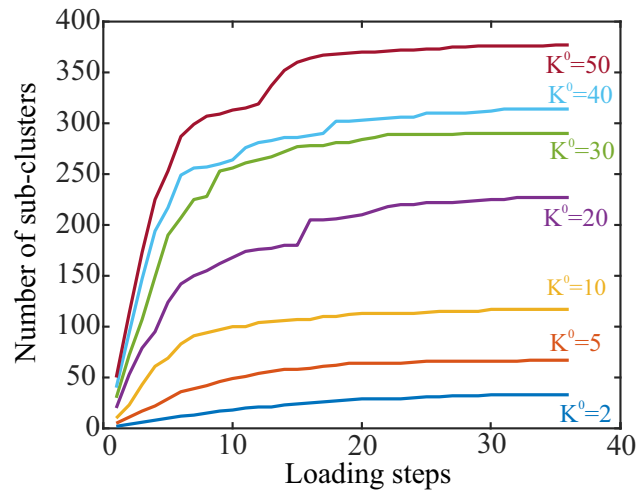


- [9] Daniel A White, Jun Kudo, Kenneth Swartz, Daniel A Tortorelli, and Seth Watts. A reduced order model approach for finite element analysis of cellular structures. *Finite Elements in Analysis and Design*, 214:103855, 2023.
- [10] J. Yvonnet and Q-C. He. The reduced model multiscale method (R3M) for the non-linear homogenization of hyperelastic media at finite strains. *Journal of Computational Physics*, 223(1):341–368, April 2007.
- [11] E. Monteiro, J. Yvonnet, and Q-C. He. Computational homogenization for nonlinear conduction in heterogeneous materials using model reduction. *Computational Materials Science*, 42(4):704–712, 2008.
- [12] J. A. Hernández, J. Oliver, A. E. Huespe, M. A. Caicedo, and J. C. Cante. High-performance model reduction techniques in computational multiscale homogenization. *Computer Methods in Applied Mechanics and Engineering*, 276:149–189, July 2014.
- [13] J. Oliver, M. Caicedo, A.E. Huespe, J.A. Hernández, and E. Roubin. Reduced order modeling strategies for computational multiscale fracture. *Computer Methods in Applied Mechanics and Engineering*, 313:560–595, January 2017.
- [14] AM Couto Carneiro, A Francisca Carvalho Alves, RP Cardoso Coelho, Jaime S Cardoso, and FM Andrade Pires. A simple machine learning-based framework for faster multi-scale simulations of path-independent materials at large strains. *Finite Elements in Analysis and Design*, 222:103956, 2023.
- [15] I. Temizer and P. Wriggers. An adaptive method for homogenization in orthotropic nonlinear elasticity. *Computer Methods in Applied Mechanics and Engineering*, 196(35-36):3409–3423, 2007.
- [16] J. Yvonnet, E. Monteiro, and Q-C. He. COMPUTATIONAL HOMOGENIZATION METHOD AND REDUCED DATABASE MODEL FOR HYPERELASTIC HETEROGENEOUS STRUCTURES. *International Journal for Multiscale Computational Engineering*, 11(3), 2013. Publisher: Begel House Inc.
- [17] F. Fritzen and O. Kunc. Two-stage data-driven homogenization for nonlinear solids using a reduced order model. *European Journal of Mechanics - A/Solids*, 69:201–220, May 2018.
- [18] J. C. Michel and P. Suquet. Nonuniform transformation field analysis. *International Journal of Solids and Structures*, 40(25):6937–6955, December 2003.
- [19] S. Roussette, J-C. Michel, and P. Suquet. Nonuniform transformation field analysis of elastic–viscoplastic composites. *Composites Science and Technology*, 69(1):22–27, 2009.
- [20] Z. Liu, M. A. Bessa, and W. K. Liu. Self-consistent clustering analysis: An efficient multi-scale scheme for inelastic heterogeneous materials. *Computer Methods in Applied Mechanics and Engineering*, 306:319–341, July 2016.
- [21] Z. Liu, M. Fleming, and W. K. Liu. Microstructural material database for self-consistent clustering analysis of elastoplastic strain softening materials. *Computer Methods in Applied Mechanics and Engineering*, 330:547–577, March 2018.
- [22] X. Han, J. Gao, M. Fleming, C. Xu, W. Xie, S. Meng, and W. K. Liu. Efficient multiscale modeling for woven composites based on self-consistent clustering analysis. *Computer Methods in Applied Mechanics and Engineering*, 364:112929, June 2020.
- [23] Y. Feng, H. Yong, and Y. Zhou. A concurrent multiscale framework based on self-consistent clustering analysis for cylinder structure under uniaxial loading condition. *Composite Structures*, 266:113827, June 2021.
- [24] B. A. Le, J. Yvonnet, and Q-C. He. Computational homogenization of nonlinear elastic materials using neural networks. *International Journal for Numerical Methods in Engineering*, 104(12):1061–1084, 2015. \_eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.4953>.
- [25] X. Lu, D. G. Giovanis, J. Yvonnet, V. Papadopoulos, F. Detrez, and J. Bai. A data-driven computational homogenization method based on neural networks for the nonlinear anisotropic electrical response of graphene/polymer nanocomposites. *Computational Mechanics*, 64(2):307–321, August 2019.
- [26] X. Lu, J. Yvonnet, L. Papadopoulos, I. Kalogeris, and V. Papadopoulos. A stochastic fe2 data-driven method for nonlinear multiscale modeling. *Materials*, 14(11):2875, 2021.
- [27] N. N. Vlassis, R. Ma, and W. Sun. Geometric deep learning for computational mechanics Part I: anisotropic hyperelasticity. *Computer Methods in Applied Mechanics and Engineering*, 371:113299, November 2020.
- [28] F. Aldakheel, E. S. Elsayed, T. I. Zohdi, and P. Wriggers. Efficient multiscale modeling of heterogeneous materials using deep neural networks. *Computational Mechanics*, April 2023.
- [29] F. Aldakheel, C. Soyarslan, S.H. Palanisamy, and E.S. Elsayed. Machine Learning Aided Multiscale Magnetostatics, January 2023. arXiv:2301.12782 [cond-mat].
- [30] D. Ryckelynck. A priori hyperreduction method: an adaptive approach. *Journal of Computational Physics*, 202(1):346–366, January 2005.
- [31] D. Ryckelynck. Hyper-reduction of mechanical models involving internal variables. *International Journal for Numerical Methods in Engineering*, 77(1):75–89, 2009. \_eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.2406>.
- [32] C. Farhat, T. Chapman, and P. Avery. Structure-preserving, stability, and accuracy properties of the energy-conserving sampling and weighting method for the hyper reduction of nonlinear finite element dynamic models. *International journal for numerical methods in engineering*, 102(5):1077–1110, 2015.
- [33] P. Avery, D. Z. Huang, W. He, J. Ehlers, A. Derkevorkian, and C. Farhat. A computationally tractable framework for nonlinear dynamic multiscale modeling of membrane woven fabrics. *International Journal for Numerical Methods in Engineering*, 122(10):2598–2625, 2021.
- [34] I.B.C.M. Rocha, P. Kerfriden, and F.P. van der Meer. Micromechanics-based surrogate models for the response of composites: A critical comparison between a classical mesoscale constitutive model, hyper-reduction and neural networks. *European Journal of Mechanics-A/Solids*, 82:103995, 2020.
- [35] K. Wang and W. Sun. A multiscale multi-permeability poroplasticity model linked by recursive homogenizations and deep learning. *Computer Methods in Applied Mechanics and Engineering*, 334:337–380, June 2018.
- [36] M. Mozaffar, R. Bostanabad, W. Chen, K. Ehmann, J. Cao, and M.A. Bessa. Deep learning predicts path-dependent plasticity. *Proceedings of the National Academy of Sciences*, 116(52):26414–26420, 2019.
- [37] F. Ghavamian and A. Simone. Accelerating multiscale finite element simulations of history-dependent materials using a recurrent neural network. *Computer Methods in Applied Mechanics and Engineering*, 357:112594, December 2019.

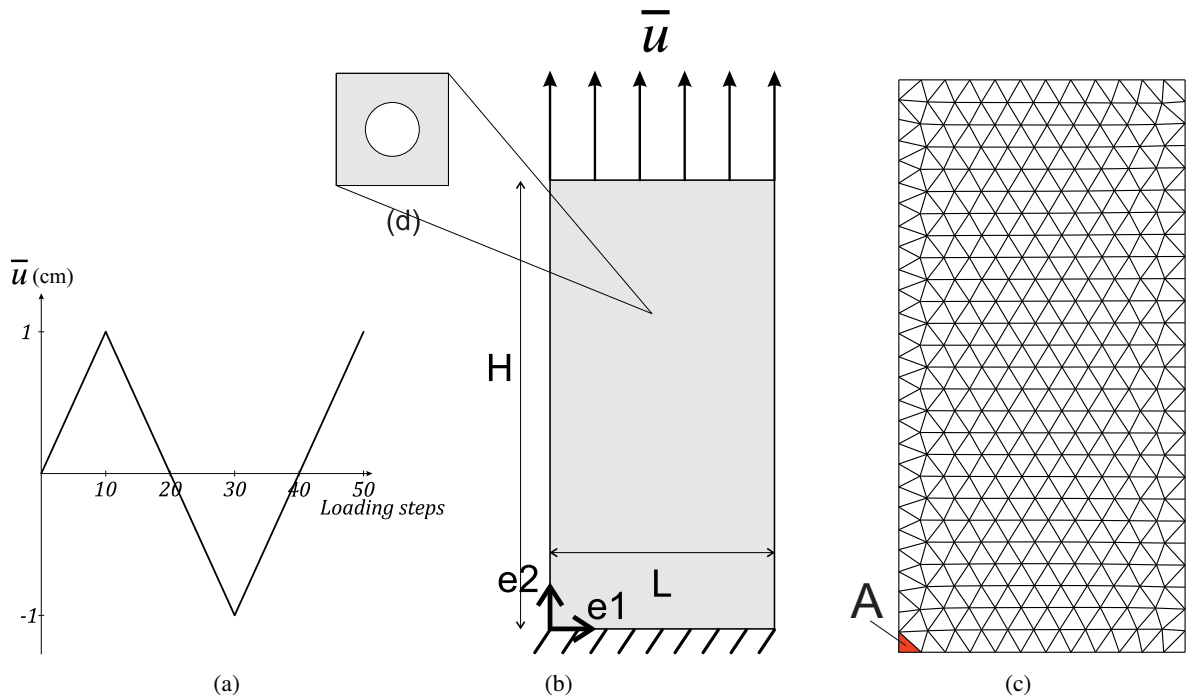
- [38] B. Li and X. Zhuang. Multiscale computation on feedforward neural network and recurrent neural network. Frontiers of Structural and Civil Engineering, 14(6):1285–1298, December 2020.
- [39] L. Wu, V. D. Nguyen, N. G. Kilingar, and L. Noels. A recurrent neural network-accelerated multi-scale model for elasto-plastic heterogeneous materials subjected to random cyclic and non-proportional loading paths. Computer Methods in Applied Mechanics and Engineering, 369:113234, September 2020.
- [40] M. B. Gorji, M. Mozaffar, J. N. Heidenreich, J. Cao, and D. Mohr. On the potential of recurrent neural networks for modeling path dependent plasticity. Journal of the Mechanics and Physics of Solids, 143:103972, October 2020.
- [41] N. N. Vlassis and W. Sun. Geometric learning for computational mechanics Part II: Graph embedding for interpretable multiscale plasticity. Computer Methods in Applied Mechanics and Engineering, 404:115768, February 2023.
- [42] F. Masi and I. Stefanou. Multiscale modeling of inelastic materials with Thermodynamics-based Artificial Neural Networks (TANN). Computer Methods in Applied Mechanics and Engineering, 398:115190, August 2022.
- [43] A. Danoun, E. Prulière, and Y. Chemisky. Thermodynamically consistent Recurrent Neural Networks to predict non linear behaviors of dissipative materials subjected to non-proportional loading paths. Mechanics of Materials, 173:104436, October 2022.
- [44] M. A. Benaimeche, J. Yvonnet, B. Bary, and Q-C. He. A k-means clustering machine learning-based multiscale method for anelastic heterogeneous structures with internal variables. International Journal for Numerical Methods in Engineering, 123(9):2012–2041, 2022. [\\_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.6925](https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.6925).
- [45] G. Cheng, X. Li, Y. Nie, and H. Li. Fem-cluster based reduction method for efficient numerical prediction of effective properties of heterogeneous material in nonlinear range. Computer Methods in Applied Mechanics and Engineering, 348:157–184, 2019.
- [46] J. Yvonnet. Nonlinear Computational Homogenization. In J. Yvonnet, editor, Computational Homogenization of Heterogeneous Materials with Finite Elements, Solid Mechanics and Its Applications, pages 161–209. Springer International Publishing, Cham, 2019.
- [47] J. MacQueen. Classification and analysis of multivariate observations. In 5th Berkeley Symp. Math. Statist. Probability, pages 281–297, 1967.
- [48] A.B. Tran, J. Yvonnet, Q-C. He, C. Toulemonde, and J. Sanahuja. A simple computational homogenization method for structures made of linear heterogeneous viscoelastic materials. Computer Methods in Applied Mechanics and Engineering, 200(45-46):2956–2970, 2011.
- [49] U. Fayyad, C. Reina, and P. S. Bradley. Initialization of iterative refinement clustering algorithms. In Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining, KDD'98, pages 194–198, New York, NY, August 1998. AAAI Press.
- [50] Y. Li and H. Wu. A Clustering Method Based on K-Means Algorithm. Physics Procedia, 25:1104–1109, January 2012.
- [51] J. C. Simo and T. J. R. Hughes. Computational Inelasticity. Springer Science & Business Media, May 2006. Google-Books-ID: EILbBwAAQBAJ.



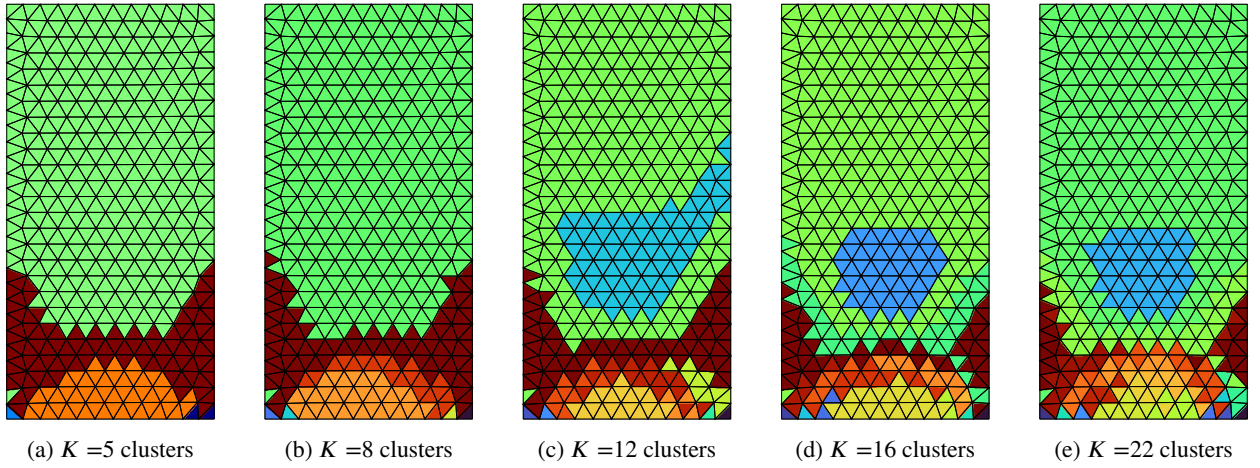
**Figure 15:** Evolution of the distribution of the clusters and their corresponding von Mises stress.



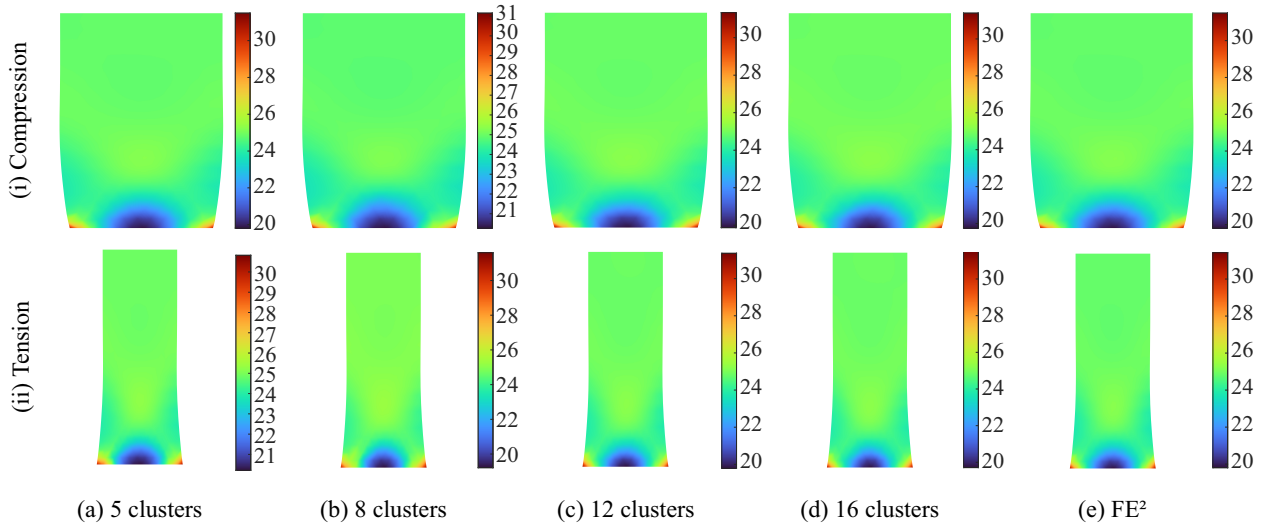
**Figure 16:** Evolution of the number of sub clusters with the previous approach developed in [44]. The number  $K^0$  indicates the initial number of clusters at the beginning of the simulation.



**Figure 17:** Elastoplastic composite structure: (a) evolution of the load; (b) geometry of the macro structure and boundary conditions; (c) macro mesh; (d) RVE.



**Figure 18:** Evolution of the cluster distribution at the end of the loading (increment 50).



**Figure 19:** von Mises stress using KMFE<sup>2</sup> for different clusters, compared with FE<sup>2</sup>: (i): At the peak of compression, (ii) at the second peak of traction.

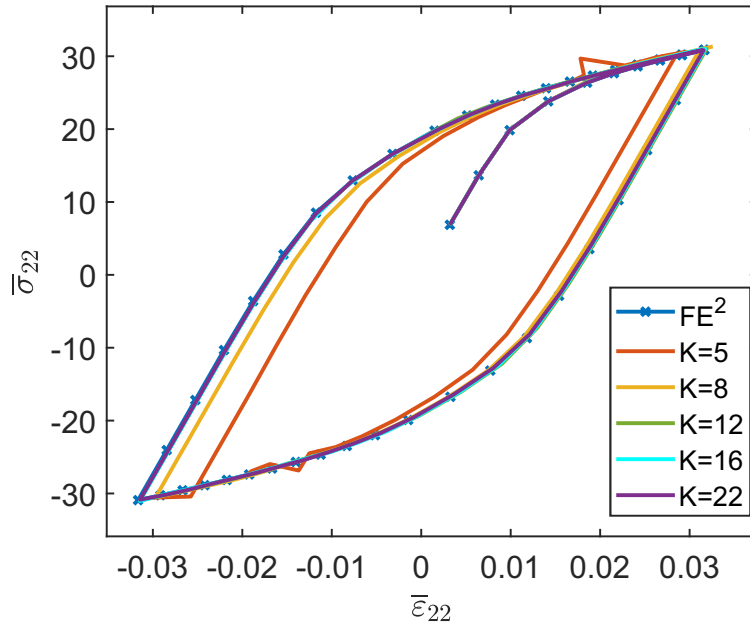


Figure 20: Stress-strain relationship along the  $y$ - direction for element A (see Fig. 17c.)

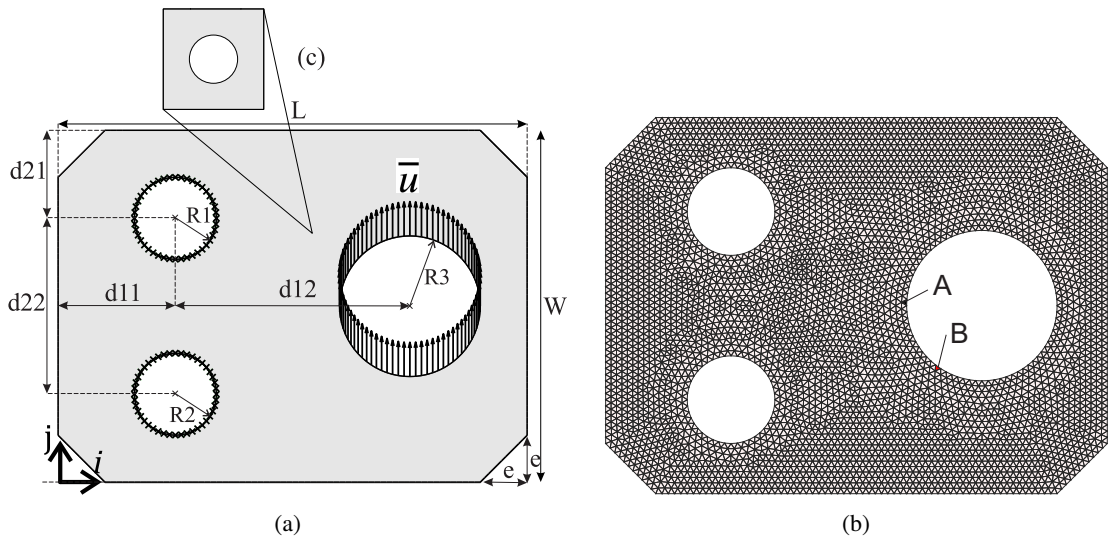


Figure 21: Bracket structure: (a) macroscopic structure, (b) Mesh, (c) RVE.

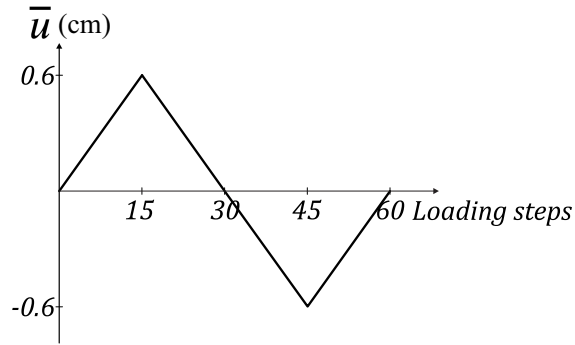


Figure 22: Loading for the elasto-plastic bracket.

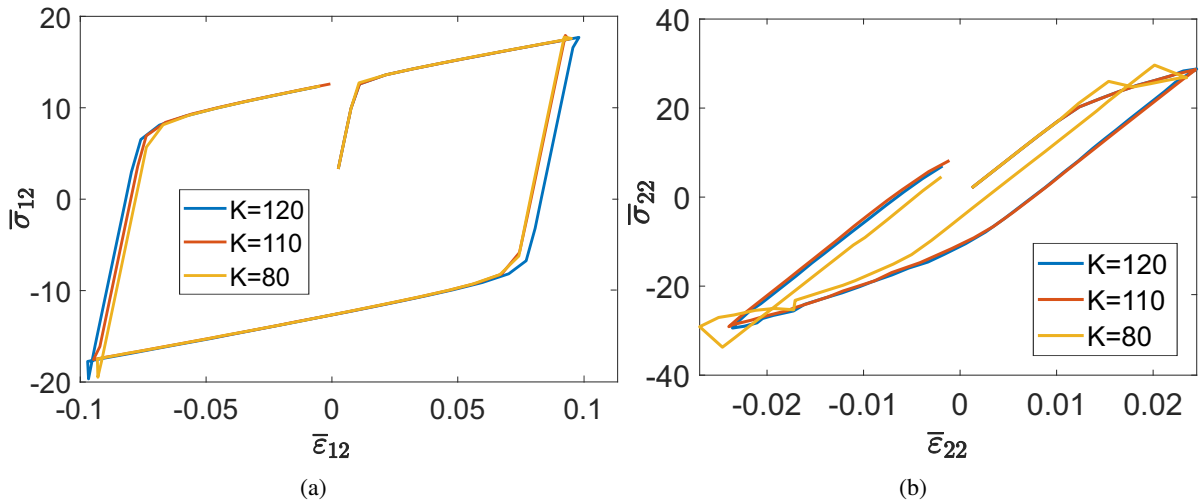


Figure 23: Stress-strain relationships in elements A and B (see Figure 21b).

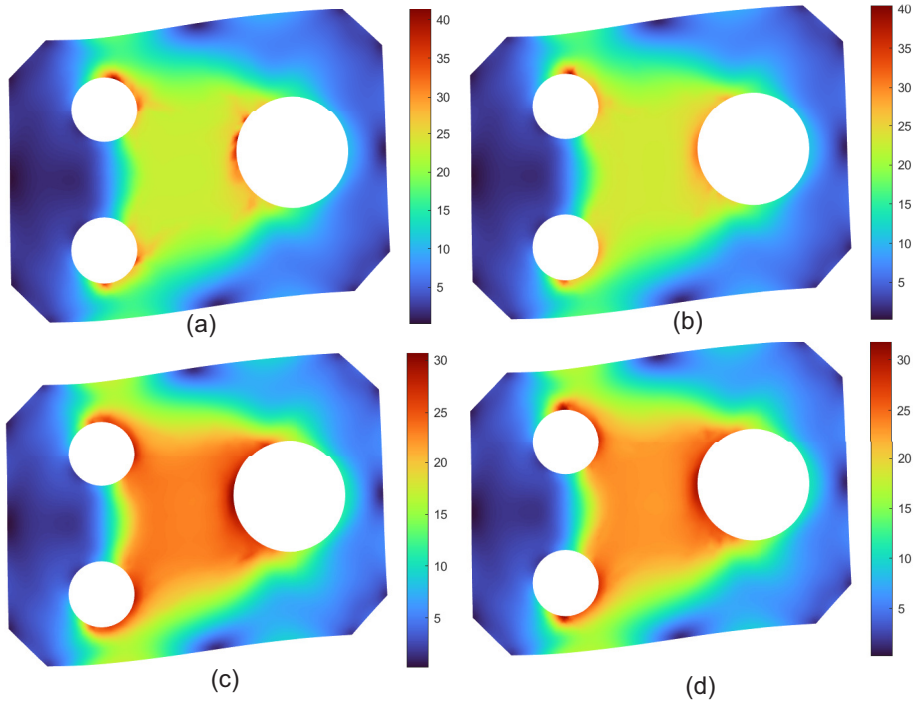


Figure 24: von Mises stress for (a)  $K=20$  clusters, (b)  $K=80$  clusters (c)  $K=100$  clusters (d)  $K=120$  clusters.

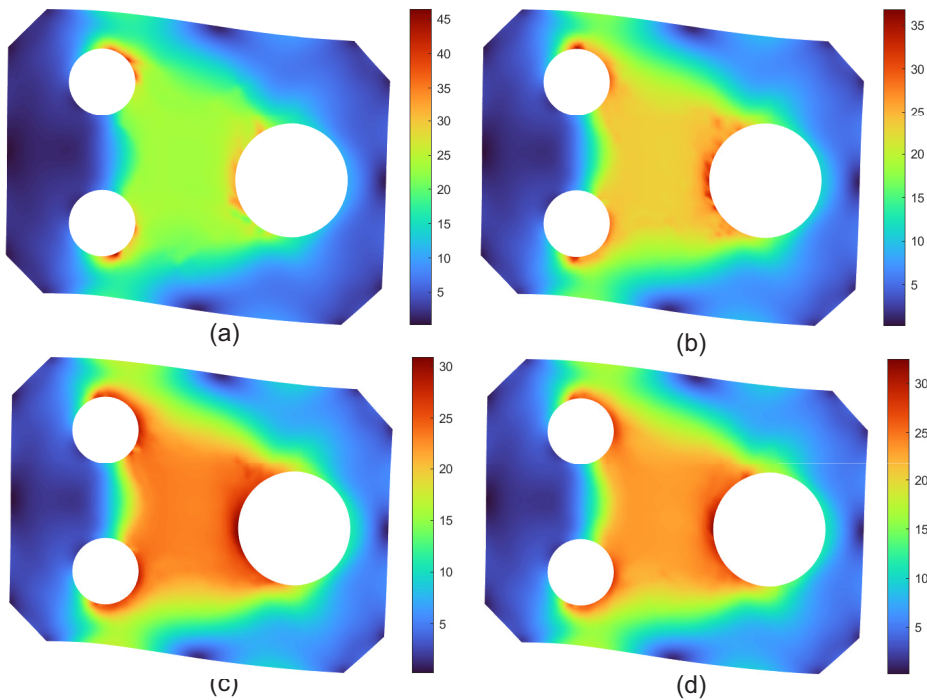
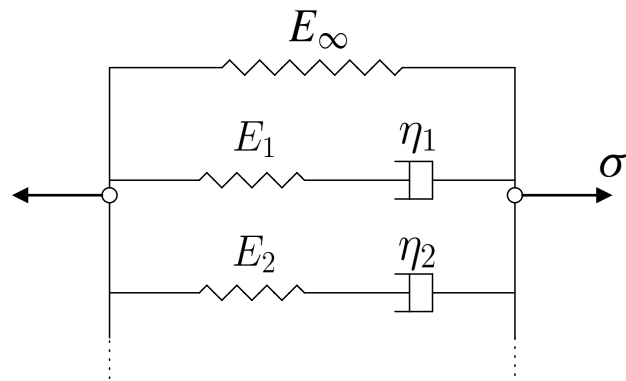


Figure 25: von Mises stress for (a)  $K=20$  clusters, (b)  $K=80$  clusters (c)  $K=100$  clusters (d)  $K=120$  clusters.





**Figure 26:** Generalized relaxation model