ORIGINAL ARTICLE

Journal Section

A Multiscale Virtual Element Method for the analysis of heterogeneous media

Abhilash Sreekumar[1]* | **Savvas P. Triantafyllou**[2]* | **François-Xavier Bécot**[3]† | **Fabien Chevillotte**[3]

[1]Centre for Structural Engineering and Informatics, Faculty of Engineering, The University of Nottingham, UK

[2]Institute for Structural Analysis and Aseismic Research, School of Civil Engineering, National Technical University of Athens, Greece

[3]Matelys - Research Lab 7 rue des Maraîchers Bâtiment B 69120 Vaulx-en-Velin France

Correspondence
Savvas P. Triantafyllou, Institute for Structural Analysis and Aseismic Research, School of Civil Engineering, National Technical University of Athens, 9, Heroon Polytechniou Str., Zografou Campus, GR-15780, Greece
Email: savtri@mail.ntua.gr

Funding information
H2020 Marie Skłodowska-Curie Actions, 765472

We introduce a novel heterogeneous multiscale method for the elastic analysis of two-dimensional domains with a complex micro-structure. To this end, the multiscale finite element method is revisited and originally upgraded by introducing virtual element discretizations at the micro-scale hence allowing for generalised polygonal and non-convex elements. The micro-scale is upscaled through the numerical evaluation of a set of multiscale basis functions. The solution of the equilibrium equations is performed at the coarse-scale at a reduced computational cost. We discuss the computation of the multiscale basis functions and corresponding virtual projection operators. The performance of the method in terms of accuracy and computational efficiency is evaluated through a set of numerical examples.

KEYWORDS
multiscale, virtual elements, heterogeneous domains

1 | INTRODUCTION

Advances in automated manufacturing and in particular additive manufacturing have led to the wide-spread application of components possessing complex and fit-for-purpose material layouts in the construction, aerospace and automotive industries [1]. Additively manufactured functionally graded composites and foams can be tailored to increased mechanical properties when compared to traditional layered composites or metals, e.g., higher strength to weight ratios and higher damping to weight ratios [2]. However, the corresponding manufacturing processes can be

*Equally contributing authors.

extensive, are prone to errors and necessitate several design iterations before a desirable layout is finally produced. This motivates the development of computational methods that can lead to augmenting desirable mechanical traits while reducing undesirable ones while still at the design stage.

Yet, the flexibility provided by manufacturing poses a series of challenges vis-a-vis the numerical simulation of structural components with exotic material layouts. The distribution of material heterogeneities at the micro- or meso-scale significantly affects the overall mechanical response of the component. Hence, both from a physical and a computational perspective, the problem of assessing the mechanical performance of a component becomes a multiscale one. Numerical simulation of physics across multiple length scales can, in principle, be done with the standard finite-element approach [3]. However, this would necessitate the use of extremely fine mesh discretizations to resolve the corresponding heterogeneities, hence leading to high computational costs [4].

Computational homogenisation techniques have been developed to efficiently address this issue [see, e.g. 5, 6, 7, 8]. These are based on the definition of a representative volume element (RVE) [9] that forms the basis for evaluating the effective constitutive behaviour of a heterogeneous domain [10, 11]. The structural problem is then solved at a macroscopic scale using standard discretization methods, e.g., finite elements. However, computational homogenisation methods rely on the assumptions of periodicity and scale separation which is often not the case in highly heterogeneous domains. Increasing the size of the RVE [12, 13] to accommodate more information about the heterogeneities will lead to increased computational costs, effectively negating the advantages of upscaling.

Multiscale Finite Element Methods (MsFEM) [14] offer a very interesting alternative to account for such heterogeneous problems. MsFEM use the information about the morphology at the micro or meso-scopic scale to construct equivalent numerical problems subject to certain kinematical constraints. Solutions to these equivalent problems yield multiscale basis functions, which are used to map mesoscale information to the macroscopic scale. These basis functions are dependent on the geometry and material properties of the underlying constituents. An inherent drawback of this method, that prevents its application to structural engineering problems, is its inability to account for the Poisson's effect. The Enhanced Multiscale Finite Element Method (EMsFEM) [15] has been developed to overcome this problem by introducing additional coupling terms when computing the multiscale basis functions. However, the EMsFEM can only treat quadrilateral meshes.

Accounting for complex morphologies of the heterogeneities encompassed by an RVE using traditional element geometries like quadrilateral and triangular elements may necessitate quite fine mesoscopic mesh discretizations, thus driving up the cost of computing multiscale basis functions. Optimization of the underlying mesh could thus prove critical to improving the performance of the method. Mesh optimization requires, as a pre-requisite, numerical methods that allow for more flexible mesh generation capabilities.

Polygonal and polyhedral finite element methods (PFEM) [16, 17, 18, 19, 20, 21, 22, 23] are increasingly being applied to several fields of computational mechanics and and are well suited in modelling complex microstructure morphologies. To this point, these formulations have been applied to topology optimization [24, 25], computational fracture and damage modelling [26, 27, 28, 29], contact problems [30, 31] and fluid mechanics [32]. However, PFEM are limited by the availability of suitable basis functions. The typically employed analytical Wachspress shape functions are valid only over simple and convex polyhedra [33]. Hence, for the case of complex and in general non-convex polygons and polyhedrons numerical approaches have to be employed that however increase the computational toll of the method [34] as in the case of, e.g., numerically evaluated harmonic [17, 35] and maximum-entropy shape functions [36, 37, 38]. This issue is further augmented in non-linear problems [39, 40]. The virtual element method (VEM) emerged as a versatile alternative for solving partial differential equations using flexible element geometries to address the aforementioned issues [41, 42, 43, 44, 45, 46, 47].

The development of the VEM can chronologically be traced back to Finite Volume Methods [48, 49] and more

recently the Mimetic Finite Difference method (MFD) [50, 51, 52, 53, 54, 55]. More specifically, the VEM emerged as a variational reformulation of the low-order MFD method [52] and its higher-order generalization [56]. MFDs, when extended to Finite Element Methods (FEM), were able to model traditional trial/test function spaces over generalised element geometries without having explicit representations for these basis functions inside the element interior. The aforementioned approaches aim to improve the accuracy of the standard FEM, when applied to polyhedral meshes, by enriching the space of trial functions with possibly non-polynomial expressions. Contrary to the MFD, the VEM attempts to preserve the polynomial accuracy over simplexes [41]. This allows the use of complex polyhedral and non-convex element shapes with more general continuity requirements such as $\mathcal{H}(\mathrm{div})$ conformity [57].

In this work, we derive a novel heterogeneous multiscale method, termed the Enhanced Multiscale Virtual Element method (EMsVEM) for the analysis of highly heterogeneous domains across multiple length scales. To achieve this, we employ the VEM to resolve heterogeneities at the fine-scale and derive appropriate multiscale basis functions to project the VEM onto a coarse finite element mesh. The solution of the governing equations is then performed at the coarse mesh at a significantly reduced computational cost. We proceed to examine and discuss the advantages and drawbacks that arise from employing virtual elements at the fine-scale in terms of accuracy and computational efficiency.

The remainder of the manuscript is organized as follows: in Section 2, the formulation of the VEM for linear elasticity is discussed. The derivations pertaining to the EMsVEM are provided in Section 3. The method is validated against the standard FEM and analytical solutions in Section 4 and the merits and bottlenecks of the EMsVEM are discussed. Concluding remarks are provided in Section 5.

2 | PRELIMINARIES

2.1 | Problem Statement

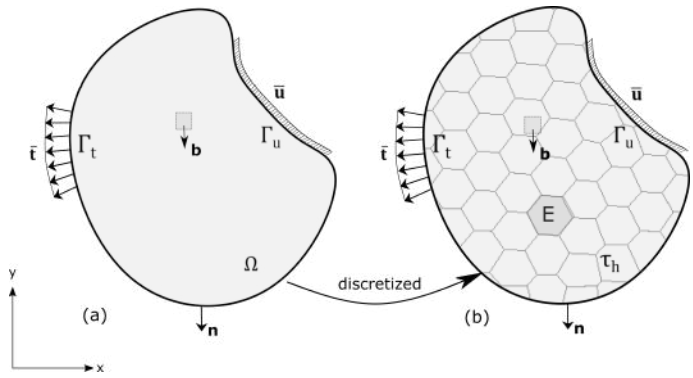


FIGURE 1 Schematic representation of a two dimensional domain Ω with boundary $\partial\Omega$. (a) Essential and natural boundaries $\bar{\mathbf{u}}$ and $\bar{\mathbf{t}}$ are prescribed on Γ_u and Γ_t respectively. (b) The discretized domain $\mathcal{T}_h$ is decomposed into polygonal elements.

The case of a domain $\Omega \subset \mathbb{R}^2$ is considered herein, subjected to plane stress or plane strain conditions. The domain is subjected to a body force $\mathbf{b}$, and is prescribed a displacement and traction $\bar{\mathbf{u}}$ and $\bar{\mathbf{t}}$ over the boundaries Γ_u and Γ_t respectively. The displacement and traction boundaries are defined to be non-intersecting, i.e., $\Gamma_u \cap \Gamma_t = \varnothing$. The

corresponding equilibrium, constitutive, and compatibility equations are defined in Eqs. (1a), (1b), and (1c), respectively as:

$$\mathrm{div}(\boldsymbol{\sigma}) + \mathbf{b} = 0 \quad \text{in } \Omega \tag{1a}$$

$$\boldsymbol{\sigma} = \mathbb{C}\boldsymbol{\varepsilon} \tag{1b}$$

$$\boldsymbol{\varepsilon} = \frac{1}{2}(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) = \nabla^S(\mathbf{u}) \equiv \boldsymbol{\varepsilon}(\mathbf{u}), \tag{1c}$$

subject to generalized inhomogeneous displacement and traction boundaries defined as follows:

$$\mathbf{u} = \bar{\mathbf{u}} \quad \text{on } \Gamma_u \tag{2a}$$

$$\boldsymbol{\sigma} \cdot \mathbf{n} = \bar{\mathbf{t}} \quad \text{on } \Gamma_t. \tag{2b}$$

Here, $\mathbb{C}$ is the material constitutive tensor, $\boldsymbol{\sigma}$ is the stress tensor, and $\boldsymbol{\varepsilon}$ is the infinitesimal strain tensor. The displacement field $\mathbf{u}$ is being solved for. The symbols, ∇, ∇^S and $\mathrm{div}(\cdot)$ denote the gradient, symmetric gradient and divergence operators, respectively.

Eqs. (1) are recast into their corresponding weak form considering the following function spaces

$$\mathcal{V}_0 = \{\mathbf{w} \in [\mathcal{H}^1(\Omega)]^2 \quad | \ \mathbf{w} = \mathbf{0} \text{ on } \Gamma_u\} \tag{3a}$$

$$\mathcal{V} = \{\mathbf{w} \in [\mathcal{H}^1(\Omega)]^2 \quad | \ \mathbf{w} = \bar{\mathbf{u}} \text{ on } \Gamma_u\}, \tag{3b}$$

where $[\mathcal{H}^1(\Omega)]^2$ represent the standard Hilbert function space of dimension 2. Defining an appropriate weighting function $\mathbf{v} \in \mathcal{V}_0$, the weak formulation of the governing equations defined in Eqs. (1a), (1b) and (1c) is expressed in the following compact form

$$\begin{cases} \text{Find } \mathbf{u} \in \mathcal{V} \text{ such that} \\ a(\mathbf{u}, \mathbf{v}) = F(\mathbf{v}) \quad \forall\, \mathbf{v} \in \mathcal{V}_0, \end{cases} \tag{4}$$

where $a(\cdot,\cdot)$ and $F(\cdot)$ are bilinear and linear functionals, respectively defined as

$$a(\mathbf{u}, \mathbf{v}) = \int_\Omega \boldsymbol{\varepsilon}(\mathbf{u})\, \mathbb{C}\, \boldsymbol{\varepsilon}(\mathbf{v})\, d\Omega \tag{5a}$$

$$F(\mathbf{v}) = \int_{\Gamma_t} \bar{\mathbf{t}}\, \mathbf{v}\, d\Gamma - \int_\Omega \mathbf{b}\, \mathbf{v}\, d\Omega. \tag{5b}$$

In this work, the virtual element approach is employed to discretize Eqs. (5).

2.2 | Virtual Element discretization

The aim of the Virtual Element Method is to generalize the standard finite element domain decomposition to polygonal (or polyhedral in 3D) elements with any number of edges (not restricted to merely quadrilateral, triangular, hexahedral or tetrahedral meshes). This includes also the case of non-convex elements. Let Ω be decomposed as shown in

Fig. 1b into n_{el} non-overlapping two dimensional polygonal elements contained in a set $\mathcal{T}_h$[1]. The corner vertices of each element $\mathcal{K}_i$ for $i = 1 \dots n_{el}$ are represented by v_j^i for $j = 1, 2, \dots N_v^i$ where N_v^i denotes the number of corner vertices. The edges are denoted by e_j^i for $j = 1, 2, \dots N_e^i$ where N_e^i=N_v^i is the total number of edges. Each edge e_j^i connects vertices v_j^i and v_{j+1}^i. For a k^{th} order VEM, the edge e_j^i is assumed to contain $k-1$ internal edge-points. The case of a second order polygonal element is shown in Fig. 2. In the following, $\mathcal{K}_i$ will be written as $\mathcal{K}$ for brevity.

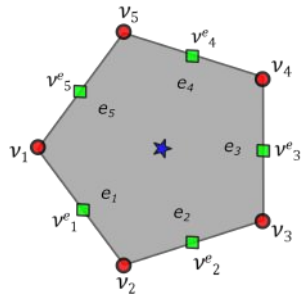


FIGURE 2 Conventions adopted for computing barycentric coordinates over a polygonal element, The element index i is omitted here for simplicity of presentation

Similar to the standard FEM, the approximations $\mathbf{u}_h$ and $\mathbf{v}_h$ to the actual displacement and weighting field, respectively, are defined as

$$\mathbf{u}_h \in \mathcal{V}_h \subset \mathcal{V} \tag{6}$$

$$\mathbf{v}_h \in \mathcal{V}_{h0} \subset \mathcal{V}_0 \tag{7}$$

where $\mathcal{V}_h$ and $\mathcal{V}_{h0}$ are termed "virtual spaces".

Hence, Eq. (4) can be established in the following discretized form

$$\begin{cases} \text{Find } \mathbf{u}_h \in \mathcal{V}_h \text{ such that} \\ a(\mathbf{u}_h, \mathbf{v}_h) = F(\mathbf{v}_h) \quad \forall\, \mathbf{v}_h \in \mathcal{V}_{h0}, \end{cases} \tag{8}$$

where the bilinear forms $a(\cdot,\cdot)$ and $F(\cdot)$ in Eq. (8) are evaluated from the piece-wise element bilinear operators $a^{\mathcal{K}}(\cdot,\cdot)$ and $F^{\mathcal{K}}(\cdot)$, respectively as

$$a(\mathbf{u}_h, \mathbf{v}_h) = \sum_{\mathcal{K}\in\mathcal{T}_h} a^{\mathcal{K}}(\mathbf{u}_h, \mathbf{v}_h), \quad \forall\, (\mathbf{u}_h, \mathbf{v}_h) \in \mathcal{V}_h^{\mathcal{K}} \times \mathcal{V}_{h0}^{\mathcal{K}} \subset \mathcal{V}_h \times \mathcal{V}_{h0}, \tag{9a}$$

$$F(\mathbf{v}_h) = \sum_{\mathcal{K}\in\mathcal{T}_h} F^{\mathcal{K}}(\mathbf{v}_h), \quad \forall\, \mathbf{v}_h \in \mathcal{V}_{h0}^{\mathcal{K}} \subset \mathcal{V}_{h0}. \tag{9b}$$

The global virtual spaces $\mathcal{V}_h$ and $\mathcal{V}_{h0}$ and local element virtual spaces $\mathcal{V}_h^{\mathcal{K}}$ and $\mathcal{V}_{h0}^{\mathcal{K}}$ are part of standard virtual element literature [see, e.g., 58]. Explicit definitions are provided in Appendix A for the sake of completeness.

Remark 1 *The spaces employed within the VEM are examples of generalized finite element spaces that allow the trial and*

[1] The parameter h is interpreted as the maximum diameter of all elements contained in $\mathcal{T}_h$.

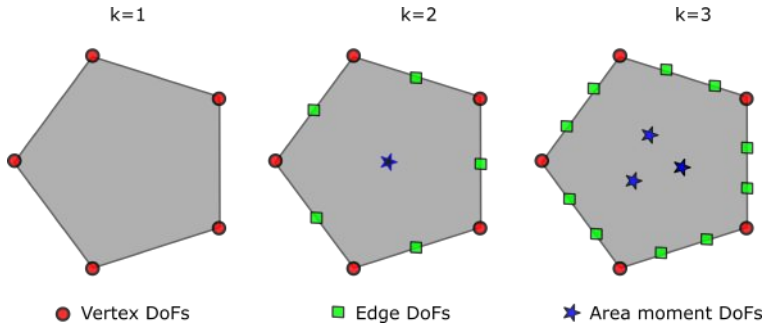


FIGURE 3 Degrees of freedom for three polynomial orders

test fields to assume non-polynomial forms over an element. This is seen as a weaker construction of the more conventional finite element space where the field variables are restricted to only members of a polynomial space.

The displacement approximation, i..e, $\mathbf{u}_h$ is implicitly interpolated at the virtual element degrees of freedom (DoFs) that include

1. $2N_v$ values at the vertices of $\mathcal{K}$, $j = 1, 2, \ldots N_v$
2. $2N_v(k-1)$ at the $k-1$ internal boundary points on each edge of $\mathcal{K}$, $j = 1, \ldots, k-1$ for each edge
3. $2\frac{k(k-1)}{2}$ second moments of $\mathbf{u}_h$ defined on the element domain interior.

The values of the degrees of freedom are denoted via the operator $dof(\cdot)$ that assumes the following expression

$$\text{dof}(\mathbf{u}_h) = \begin{cases} \mathbf{u}_h(\boldsymbol{\nu}_j) & \text{for } j = 1, 2, \ldots N_v^i \\ \mathbf{u}_h(\boldsymbol{\nu}_j^e) & \text{for } j = 1, 2, \ldots N_e^i \\ \frac{1}{|\mathcal{K}|}\int_{\mathcal{K}} \mathbf{u}_h \cdot \mathbf{p}\, d\mathcal{K} \quad \forall \mathbf{p} \in [\mathbb{M}_{k-2}(\mathcal{K})]^2 & \text{element domain interior} \end{cases}, \tag{10}$$

where $[\mathbb{M}_{k-2}(\mathcal{K})]^2$ denotes a space of $k-2$ order vector monomials of dimension 2. The weighting function $\mathbf{v}_h$ is also interpolated at the same DoFs.

Explicit definitions for these vector monomial spaces are provided in Appendix B for the sake of completeness. The DoFs of the space have been illustrated over an arbitrary polygonal element for $k = 1, 2, 3$ in Fig. 3. Hence, for the case of a two-dimensional domain where the displacements assume two DoFs per node, the dimension of the virtual element space is

$$n = 2N_v + 2N_v(k-1) + 2\frac{k(k-1)}{2} = 2N_v k + k(k-1) \tag{11}$$

Since the VEM functions do not have an explicit expression over the domain, essential operations like $\nabla(\cdot)$ and consequently $\boldsymbol{\varepsilon}(\cdot)$ (Eq. (1c)) cannot directly be performed. Conversely, a strain specific projection operator $\Pi_k^{\boldsymbol{\varepsilon}} : \mathcal{V}_h^{\mathcal{K}}(\mathcal{K}) \rightarrow$

$[\mathbb{P}_k(\mathcal{K})]^2$ is defined according to the following optimality criterion

$$a^{\mathcal{K}}(\mathbf{u}_h - \Pi_k^{\boldsymbol{\varepsilon}}\mathbf{u}_h, \mathbf{p}) = 0, \quad \forall \mathbf{p} \in [\mathbb{P}_k(\mathcal{K})]^2, \tag{12}$$

where $[\mathbb{P}_k(\mathcal{K})]^2$ is a space of k^{th} order two-dimensional vector polynomials. This criterion ensures that the error arising from the polynomial approximation is *energetically orthogonal* to the approximating subspace $[\mathbb{P}_k(\mathcal{K})]^2$. It follows that the strain energy is computed exactly, despite the simplifying assumption introduced by the projection. In VEM literature, this property is called *k-consistency*.

Remark 2 *Introducing the strain specific $\Pi_k^{\boldsymbol{\varepsilon}}$ projection onto a polynomial space is a deviation from the classical Galerkin framework. Approximating a non-polynomial function using a polynomial basis falls into a class of "variational crimes" [41] and introduces additional error into the formulation. This error is a result of performing numerical integration for non-polynomial functions using polynomial quadrature rules. However, one can derive a-priori bounds and estimators to quantify and control this error [see, e.g., 59].*

The approximating subspace $[\mathbb{P}_k(\mathcal{K})]^2$ is spanned by vector monomials belonging to $[\mathbb{M}_k(\mathcal{K})]^2$. However, these monomials also contain zero deformation modes, i.e., rigid body modes that contribute zero strain energy to the formulation described in Eq. (12). To avoid spurious results arising from ill-conditioned or singular matrices, such modes are excluded when computing $\Pi_k^{\boldsymbol{\varepsilon}}$. Instead, they are treated in a stability term. This is discussed in Section 3.2.

Following this reasoning, Eq. (12) is eventually established in the following form

$$a^{\mathcal{K}}(\mathbf{u}_h - \Pi_k^{\boldsymbol{\varepsilon}}\mathbf{u}_h, \mathbf{m}_j) = 0, \quad \forall \mathbf{m}_j \in [\mathbb{M}_k(\mathcal{K})]^2 \setminus \mathbb{K}^{\boldsymbol{\varepsilon}}(\mathcal{K}), \tag{13}$$

where $\mathbb{K}^{\boldsymbol{\varepsilon}}(\mathcal{K})$ belongs to the kernel of rigid body motions and $[\mathbb{M}_k(\mathcal{K})]^2 \setminus \mathbb{K}^{\boldsymbol{\varepsilon}}(\mathcal{K})$ is the kernel of the strain operator $\boldsymbol{\varepsilon}(\cdot)$. The contents of these sets can be determined using the additive kinematic decomposition rules mentioned in [60]. The monomial spaces used for the VEM formulation are provided in Appendix B for the sake of completeness. The exact procedure followed for computing $\Pi_k^{\boldsymbol{\varepsilon}}$ and the associated local stiffness matrix $a^{\mathcal{K}}(\mathbf{u}_h, \mathbf{v}_h)$ is discussed within a multiscale context in Section 3.2.

3 | ENHANCED MULTISCALE VIRTUAL ELEMENT METHOD

3.1 | Overview

To this point, the EMsFEM has been developed to treat regular heterogeneous domains as shown in Fig. 4a, where each rectangular coarse element clusters its own representative portion of the underlying fine mesh. The fine mesh is designed to resolve all mesoscale heterogeneities. These are upscaled to the coarse level, where the solution is obtained at reduced computational cost. In this work, we treat the most general case of complex geometrical domains at the micro structure as shown in Fig. 4b.

To achieve this, we employ the virtual element method to accurately and efficiently resolve the heterogeneities at the fine scale. Similar to the EMsFEM, the heterogeneous domain is coarsely discretized into $\mathcal{K}_{M(\alpha)},\ \alpha = 1 \ldots n_{M_{el}}$, coarse elements, where $n_{M_{el}}$ is the number of coarse elements. Each coarse element clusters its own set of $\mathcal{K}_{m(i)},\ i = 1 \ldots n_{m_{el}}$ micro-elements where $n_{m_{el}}$ is the number of micro-elements in the $\mathcal{K}_{M(\alpha)}$.

For each type of coarse element, a set of multiscale basis functions is evaluated using the standard VEM. These basis functions are then employed to map the fine scale onto the coarse scale where the solution of the governing

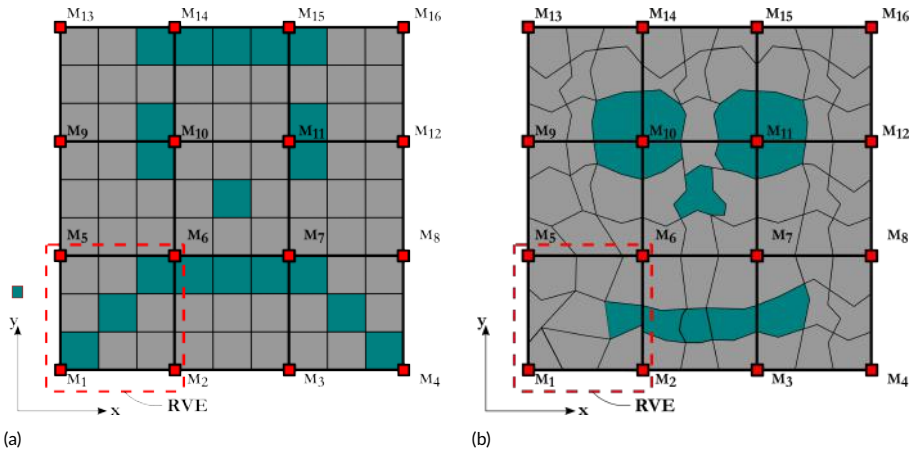


FIGURE 4 Multiscale mesh with (a) 9 coarse quadrilateral elements and 81 quadrilateral fine-elements and (b) 9 coarse quadrilateral elements and 81 polygonal (possibly non-convex) fine-elements

equations is performed. The EMsVEM procedure is schematically depicted in Fig. 5.

The multiscale basis functions required for the upscaling procedure are evaluated through the solution of a homogeneous version of Eq. (8) over the α^{th} coarse element domain $\mathcal{K}_{M(\alpha)}$, i.e.,

$$\begin{cases} \text{Find } \mathbf{u}_h \in \mathcal{V}_h(\mathcal{K}_{M(\alpha)}) \text{ such that} \\ a(\mathbf{u}_h, \mathbf{v}_h) = 0 \quad \forall\, \mathbf{v}_h \in \mathcal{V}_{h0}(\mathcal{K}_{M(\alpha)}). \end{cases} \tag{14}$$

These homogeneous equations are subjected to kinematical constraints that account for heterogeneities and the deformability of the boundary. They are imposed over the RVE in the form of linear or periodic boundaries. The procedure for such enforcements are discussed in [2] and will not be detailed here. The choice of the coarse element boundary conditions plays an important role vis-a-vis the accuracy of the method. Resonance errors may occur when the coarse-element length scale approaches the fine-element scale, i.e., a large number of coarse elements are employed to mesh the domain. Such errors can be overcome by an oversampling strategy [61].

3.2 | Virtual fine-scale stiffness matrices

The bilinear operator used in the coarse element sub-problem, defined in Eq. (14), is decomposed in a manner analogous to Eq. (9):

$$a(\mathbf{u}_h, \mathbf{v}_h) = \sum_{i=1}^{n_{m_{el}}} a^{\mathcal{K}_m}(\mathbf{u}_h, \mathbf{v}_h), \quad \forall (\mathbf{u}_h, \mathbf{v}_h) \in \mathcal{V}_h^{\mathcal{K}}(\mathcal{K}_{m(i)}) \times \mathcal{V}_{h0}^{\mathcal{K}}(\mathcal{K}_{m(i)}) \subset \mathcal{V}_h(\mathcal{K}_{M(\alpha)}) \times \mathcal{V}_{h0}(\mathcal{K}_{M(\alpha)}) \tag{15}$$

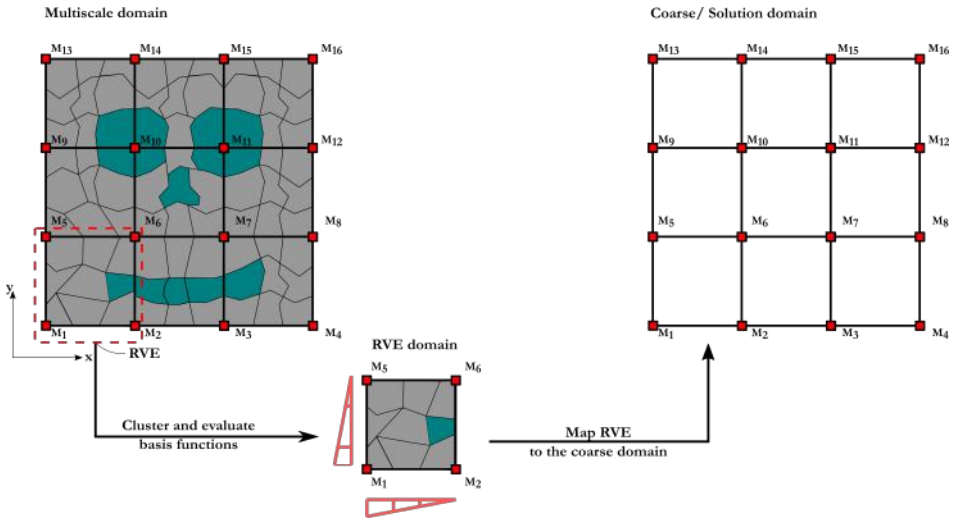


FIGURE 5 Schematic of the EMsVEM upscaling procedure. The case of linear boundary conditions for the evaluation of multiscale basis functions is considered.

Contrary to the standard EMsFEM, in this work we employ virtual elements at the fine scale. Consequently, we adopt a VEM discretization for the element-wise bilinear functional shown in Eq. (15):

$$a^{\mathcal{K}_m}(\mathbf{u}_h, \mathbf{v}_h) \equiv a^{\mathcal{K}_m}\Big((\mathbf{u}_h - \Pi_k^{\boldsymbol{\varepsilon}_m}\mathbf{u}_h) + \Pi_k^{\boldsymbol{\varepsilon}_m}\mathbf{u}_h, (\mathbf{v}_h - \Pi_k^{\boldsymbol{\varepsilon}_m}\mathbf{v}_h) + \Pi_k^{\boldsymbol{\varepsilon}_m}\mathbf{v}_h\Big), \tag{16}$$

where $\Pi_k^{\boldsymbol{\varepsilon}_m}$ denotes the projector discussed in Section 2.2.

Remark 3 *To account for the potential non-polynomial expressions, standard Lagrange polynomials are not a suitable basis. Rather, a canonical basis is utilised that is implicitly approximated in terms of a linear combination of conventional scaled vector monomials. This is convenient because such monomials have simple analytic expressions. The problem of giving an explicit form to the canonical basis is now reduced to computing the coefficients involved in the linear combination. To accomplish this, functions belonging to virtual spaces need to be projected onto a polynomial space* $[\mathbb{P}_k(\mathcal{K}_m)]^2$ *spanned by the scaled vector monomial basis* $[\mathbb{M}_k(\mathcal{K}_m)]^2$.

Expanding Eq. (16) and exploiting the symmetry of the bilinear functional and the energetic orthogonality condition in Eq. (13), the following relation is obtained

$$a^{\mathcal{K}_m}(\mathbf{u}_h, \mathbf{v}_h) = a^{\mathcal{K}_m}(\Pi_k^{\boldsymbol{\varepsilon}_m}\mathbf{u}_h, \Pi_k^{\boldsymbol{\varepsilon}_m}\mathbf{v}_h) + a^{\mathcal{K}_m}(\mathbf{u}_h - \Pi_k^{\boldsymbol{\varepsilon}_m}\mathbf{u}_h, \mathbf{v}_h - \Pi_k^{\boldsymbol{\varepsilon}_m}\mathbf{v}_h), \tag{17}$$

where the first term consists entirely of polynomial expressions and can be exactly computed analytically (k = 1 methods) or numerically (k ≥ 2 methods) through the degrees of freedom defined in Eq. (10).. This is called the *consistency term* and involves the elliptic projection of the virtual element functions,i.e., $\Pi_k^{\boldsymbol{\varepsilon}_m}\mathbf{u}_h$ and $\Pi_k^{\boldsymbol{\varepsilon}_m}\mathbf{v}_h$, respectively. The second term on the right hand side of the expression, contains unknown non-polynomial integrands $\mathbf{u}_h$ and $\mathbf{v}_h$. These terms are not computable as elements of a virtual element space lack an explicit definition over the element interior.

The non-computable term is replaced with a computable bilinear form $\mathcal{S}^{\mathcal{K}_m}(\cdot,\cdot)$ that can account for strain energy

associated with higher order deformation modes while ensuring coercivity; this is called the *stability* term. Hence, the final VEM approximation of the strain energy is expressed as:

$$a^{\mathcal{K}_m}(\mathbf{u}_h, \mathbf{v}_h) \approx a^{\mathcal{K}_m}(\Pi_k^{\boldsymbol{\varepsilon}^m}\mathbf{u}_h, \Pi_k^{\boldsymbol{\varepsilon}^m}\mathbf{v}_h) + \mathcal{S}^{\mathcal{K}_m}(\mathbf{u}_h - \Pi_k^{\boldsymbol{\varepsilon}^m}\mathbf{u}_h, \mathbf{v}_h - \Pi_k^{\boldsymbol{\varepsilon}^m}\mathbf{v}_h). \tag{18}$$

There exists several ways to define the approximated stability operator $\mathcal{S}^{\mathcal{K}_m}(\cdot,\cdot)$. One is referred to [62, 63] for a more comprehensive discussion on the stability term. In this work, we adopt the D-recipe stabilization [64] that also ensures correct scaling in relation to the consistency term. that is discussed in Appendix E for the sake of completeness.

Remark 4 *The consistency term on its own is rank deficient. The stability term is included to restore coercivity. In the context of quadrilateral or hexahedral elements, the stability term can be interpreted as the hourglass stiffness. The term can assume any bilinear form that satisfies the fundamental properties of coercivity and stability and reduces to zero over polynomial subspaces. The easy-to-compute stability term introduces additional error into the formulation. However, this approximation is chosen such that the error convergence rates are still optimal. It is for this reason that one should not expect the VEM formulation to reduce to the familiar FEM approach in the case of, e.g., quadrilateral elements. In the select case of the three noded triangular element, the polynomial space used is complete to degree one. In this case the stability term reduces to zero over polynomial subspaces and the consistency term coincides with the classical three-noded FEM stiffness matrix.*

The fine scale stiffness matrix $\mathbf{K}^{el,\alpha}_{m(i)}$ is now defined on the basis of Eq. (18):

$$\begin{aligned}\mathbf{K}^{el,\alpha}_{m(i)} = a^{\mathcal{K}_m}(\mathbf{u}_h, \mathbf{v}_h) = \underbrace{\int_{\mathcal{K}_{m(i)}} \boldsymbol{\varepsilon}\left(\Pi_k^{\boldsymbol{\varepsilon}^m}\mathbf{u}_h\right)^T \boldsymbol{\sigma}\left(\Pi_k^{\boldsymbol{\varepsilon}^m}\mathbf{v}_h\right) d\mathcal{K}}_{\mathbf{K}_C} + \underbrace{\mathcal{S}^{\mathcal{K}_m}(\mathbf{u}_h - \Pi_k^{\boldsymbol{\varepsilon}^m}\mathbf{u}_h, \mathbf{v}_h - \Pi_k^{\boldsymbol{\varepsilon}^m}\mathbf{v}_h)}_{\mathbf{K}_S},\\ \forall(\mathbf{u}_h, \mathbf{v}_h) \in \mathcal{V}_h^{\mathcal{K}}(\mathcal{K}_{m(i)}) \times \mathcal{V}_{h0}^{\mathcal{K}}(\mathcal{K}_{m(i)}),\end{aligned} \tag{19}$$

where, $\mathbf{K}_C$ is the consistency stiffness matrix and $\mathbf{K}_S$ is the stability stiffness matrix whose expressions are provided in Appendices D and E, respectively.

As discussed in Section 2.2, the strains $\boldsymbol{\varepsilon}(\cdot)$ cannot be directly evaluated on $\mathbf{u}_h$ and $\mathbf{v}_h$ as these are implicitly defined. The projector $\Pi_k^{\boldsymbol{\varepsilon}^m}$ provides the functions with an explicit form by approximating them with a monomial expansion. The projector $\Pi_k^{\boldsymbol{\varepsilon}^m}$ is evaluated using the following orthogonality condition at the micro-scale

$$a^{\mathcal{K}_m}(\mathbf{u}_h - \Pi_k^{\boldsymbol{\varepsilon}^m}\mathbf{u}_h, \mathbf{m}_j) = 0, \quad \forall \mathbf{m}_j \in [\mathbb{M}_k(\mathcal{K}_m)]^2 \setminus \mathbb{K}^{\boldsymbol{\varepsilon}^m}(\mathcal{K}_m), \tag{20}$$

The aforementioned monomial expansion for the term $\Pi_k^{\boldsymbol{\varepsilon}^m}\mathbf{u}_h$ is expressed as follows:

$$\Pi_k^{\boldsymbol{\varepsilon}^m}\mathbf{u}_h = \sum_{i=1}^{n_k-3} s_i \mathbf{m}_i, \tag{21}$$

where $\mathbf{m}_i \in [\mathbb{M}_k(\mathcal{K}_m)]^2 \setminus \mathbb{K}_k^{\boldsymbol{\varepsilon}^m}(\mathcal{K}_m)$. The quantity n_k is

$$n_k = (k+1)(k+2) \tag{22}$$

and denotes the number of vector valued monomials in $[\mathbb{M}_k(\mathcal{K}_m)]^2$.

Substituting Eq. (21) into Eq. (20) and simplifying, the following expression is derived

$$\sum_{i=1}^{n_k-3} s_i \underbrace{a^{\mathcal{K}_m}(\mathbf{m}_i, \mathbf{m}_j)}_{\tilde{G}_{ij}} = \underbrace{a^{\mathcal{K}_m}(\mathbf{u}_h, \mathbf{m}_j)}_{B_{ij}}. \tag{23}$$

The procedure to evaluate the fine scale matrices $\tilde{\mathbf{G}}$ and $\mathbf{B}$ is provided in Appendix C.

Eq. (23) is conveniently recast in matrix form as

$$\tilde{\mathbf{G}}\boldsymbol{\Pi}_k^{\boldsymbol{\varepsilon}_m} = \mathbf{B}. \tag{24}$$

where $\boldsymbol{\Pi}_k^{\boldsymbol{\varepsilon}_m}$ is a vector comprising the coefficients s_i. Hence, the fine-scale projection operator is eventually derived as

$$\boldsymbol{\Pi}_k^{\boldsymbol{\varepsilon}_m} = \tilde{\mathbf{G}}^{-1}\mathbf{B}. \tag{25}$$

The micro-element stiffness matrix in Eq. (19) is used to assemble the coarse element specific RVE stiffness matrix $\mathbf{K}_m^{\alpha}$ using a standard direct stiffness approach, i.e.,

$$\mathbf{K}_m^{\alpha} = \mathop{\mathsf{A}}_{i=1}^{n_{m_{el}}} \mathbf{K}_{m(i)}^{el,\alpha}, \tag{26}$$

where A is the assembly operator. The coarse element specific RVE load vector $\mathbf{f}_m^{\alpha}$ is also similarly assembled from it's micro-element contributions as

$$\mathbf{f}_m^{\alpha} = \mathop{\mathsf{A}}_{i=1}^{n_{m_{el}}} \mathbf{f}_{m(i)}^{el,\alpha}, \tag{27}$$

where $\mathbf{f}_{m(i)}^{el,\alpha}$ is the nodal force vector defined at the fine scale. This is established on the on the basis of Eqs. (5b) and (9b) as

$$\mathbf{f}_{m(i)}^{el,\alpha} = F^{\mathcal{K}_m}(\mathbf{v}_h) = \int_{\Gamma_t \in \partial\mathcal{K}_{m(i)}} \mathbf{v}_h^T \bar{\mathbf{t}}\, d\Gamma - \int_{\mathcal{K}_{m(i)}} \mathbf{v}_h^T \mathbf{b}\, d\Omega, \tag{28}$$

where Γ_t denotes the traction boundaries associated with the element under consideration. The body force and traction terms can be computed by defining appropriate nodal quadrature rules over the micro-element interior and edge, respectively as detailed in [65].

3.3 | Constructing multiscale basis functions

In the multiscale finite element framework utilized herein, the micro-displacement components of the fine mesh are mapped to the macro-displacement nodal components of the corresponding coarse-element according to Eq. (29).

$$\begin{aligned} u_{mx,i} &= \sum_{J=1}^{n_M} N_{iJxx} u_{Mx,J} + \sum_{J=1}^{n_M} N_{iJxy} u_{My,J} \\ u_{my,i} &= \sum_{J=1}^{n_M} N_{iJyx} u_{Mx,J} + \sum_{J=1}^{n_M} N_{iJyy} u_{My,J} \end{aligned} \tag{29}$$

where $u_{mx,i}$ and $u_{my,i}$, $i = 1, \ldots, n_m$ are the displacement components of the i_{th} micro-node, n_m is the number of micro-nodes within the coarse-element and $n_M = 4$ is the number of macro-nodes of the coarse-element.

Moreover, $u_{Mx,J}$ and $u_{My,J}$ are the macro-displacement components defined at the J_{th} macro-node of the coarse-element; N_{iJxx}, N_{iJyy}, N_{iJxy} and N_{iJyx} correspond to the micro-basis interpolation functions. Eqs. (29) hold if and only if the micro-basis interpolation functions satisfy the following conditions

$$\begin{matrix} \sum_{J=1}^{n_M} N_{IJxx} = 1 & \sum_{J=1}^{n_M} N_{IJxy} = 0 \\ \sum_{J=1}^{n_M} N_{IJyx} = 0 & \sum_{J=1}^{n_M} N_{IJyy} = 1 \end{matrix} \quad , \; I = 1 \ldots n_M \tag{30}$$

At the fine-element scale, this mapping assumes the following form

$$\mathbf{u}^{\alpha}_{m(i)} = \mathbf{N}_{m(i)} \mathbf{u}_{M(\alpha)}, \tag{31}$$

where $\mathbf{u}^{\alpha}_{m(i)}$ is the displacement vector for the i^{th} fine-element in the α^{th} RVE. The array $\mathbf{N}_{m(i)}$ denotes the multiscale basis functions mapping the associated coarse-nodal displacements in $\mathbf{u}_{M(\alpha)}$ to the current fine-element.

Collecting Eqs. (31) for all micro-elements within the coarse-element, the following equation is established

$$\mathbf{u}^{\alpha}_{m} = \mathbf{N}_{m} \mathbf{u}_{M(\alpha)}, \tag{32}$$

where $\mathbf{u}^{\alpha}_{m}$ comprises the nodal displacement of the fine-mesh contained within coarse-element α.

A set of interpolation functions satisfying Eqs. (30) (hence Eqs. (29) also) can be established by solving the following boundary value problem

$$\begin{cases} \mathbf{K}^{\alpha}_{m} \mathbf{u}^{\alpha}_{m} = \mathbf{0} & , \text{on } \mathcal{K}_{M(\alpha)} \\ \mathbf{u}_S = \bar{\mathbf{u}}_{IJ} & , \text{on } \partial\mathcal{K}_{M(\alpha)} \end{cases} , I = 1 \ldots n_M, J = 1, 2 \tag{33}$$

where $\mathbf{K}^{\alpha}_{m}$ is the RVE stiffness matrix that is assembled according to Eq. (26), $\mathbf{u}_S$ is the vector of boundary micro-nodal displacements and $\bar{\mathbf{u}}$ are the the prescribed displacements obtained from imposing linear, periodic or oversampling boundary conditions over the coarse-element/RVE boundary [see, e.g., 61, 2]. Each solution derived for a prescribed set of boundary conditions $\bar{\mathbf{u}}_{IJ}$ corresponds to a column of the multiscale basis function matrix $\mathbf{N}_m$. Algorithm 1 summarizes the procedural steps required for the evaluation of the micro-basis functions.

Remark 5 *To retain consistency with the literature, each coarse-element is termed herein an RVE. This however should not be confused with the typical definition of the RVE as employed within a scale separated homogenisation theory.*

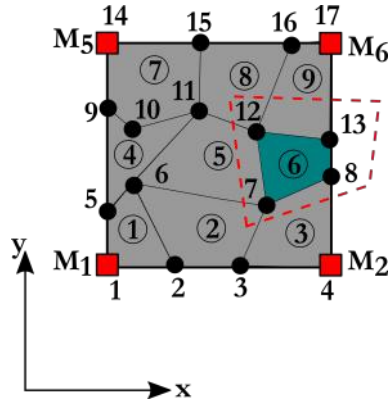


FIGURE 6 RVE with 9 generalized (possibly non-convex) polygonal elements. The number of fine-elements and fine-nodes in the RVE are $n_{m_{el}} = 9$ and $n_m = 17$ respectively.

As an example, Eq. (31) assumes the following form for the case of the RVE shown in Fig. 6 and the micro-element i = 6,

$$\mathbf{u}^1_{m(6)} = \mathbf{N}_{m(6)}\mathbf{u}_{M(1)}, \tag{34}$$

where the vector of nodal displacements for fine-element #6 is

$$\mathbf{u}^1_{m(6)} = \begin{bmatrix} u_{mx,7} & u_{my,7} & u_{mx,8} & u_{my,8} & u_{mx,13} & u_{my,13} & u_{mx,12} & u_{my,12} \end{bmatrix}^T, \tag{35}$$

the coarse-element nodal displacement vector is

$$\mathbf{u}_{M(1)} = \begin{bmatrix} u_{Mx,1} & u_{My,1} & u_{Mx,2} & u_{My,2} & u_{Mx,6} & u_{My,6} & u_{Mx,5} & u_{My,5} \end{bmatrix}^T, \tag{36}$$

and the corresponding multiscale basis function matrix is expressed as

$$\mathbf{N}_{m(6)} = \begin{bmatrix} N_{xx7,1} & N_{xy7,1} & N_{xx8,1} & N_{xy8,1} & N_{xx13,1} & N_{xy13,1} & N_{xx12,1} & N_{xy12,1} \\ N_{xy7,1} & N_{yy7,1} & N_{xy8,1} & N_{yy8,1} & N_{xy13,1} & N_{yy13,1} & N_{xy12,1} & N_{yy12,1} \\ N_{xx7,2} & N_{xy7,2} & N_{xx8,2} & N_{xy8,2} & N_{xx13,2} & N_{xy13,2} & N_{xx12,2} & N_{xy12,2} \\ N_{xy7,2} & N_{yy7,2} & N_{xy8,2} & N_{yy8,2} & N_{xy13,2} & N_{yy13,2} & N_{xy12,2} & N_{yy12,2} \\ N_{xx7,3} & N_{xy7,3} & N_{xx8,3} & N_{xy8,3} & N_{xx13,3} & N_{xy13,3} & N_{xx12,3} & N_{xy12,3} \\ N_{xy7,3} & N_{yy7,3} & N_{xy8,3} & N_{yy8,3} & N_{xy13,3} & N_{yy13,3} & N_{xy12,3} & N_{yy12,3} \\ N_{xx7,4} & N_{xy7,4} & N_{xx8,4} & N_{xy8,4} & N_{xx13,4} & N_{xy13,4} & N_{xx12,4} & N_{xy12,4} \\ N_{xy7,4} & N_{yy7,4} & N_{xy8,4} & N_{yy8,4} & N_{xy13,4} & N_{yy13,4} & N_{xy12,4} & N_{yy12,4} \end{bmatrix}, \tag{37}$$

respectively.

For the RVE shown in Fig. 6, $\mathbf{u}^{\alpha}_m$ is a 34×1 vector. Furthermore, $\mathbf{N}_m$ is the coarse-element matrix of multiscale basis functions; for the case of Fig. 6 this is a 34×8 matrix. Each column j of $\mathbf{N}_m$ corresponds to the deformed configuration

of the RVE when the jth coarse degree of freedom is equal to 1 whereas all other coarse degrees of freedom are equal to zero.

Algorithm 1: multiscale basis function evaluation schema

Data: Define coarse mesh and fine mesh/micromesh and material properties

foreach *coarse element* α **do**

  foreach *micro element* i **do**

    Compute: $\mathbf{\Pi}_{k}^{\boldsymbol{\varepsilon}_m}$ (see Eq. (25)) ;

    Compute: $\mathbf{K}_C$, $\mathbf{K}_S$, and $\mathbf{K}_{m(i)}^{el,\alpha}$ (see Eq. (19)) ;

    Assemble to $\mathbf{K}_m^{\alpha}$;

  end

  foreach *macro-node* $I = 1, \ldots 4$ **do**

    foreach *macro degree of freedom* $J = 1, 2$ **do**

      Define: $\bar{\mathbf{u}}_{IJ}$;

      Solve: $\begin{cases} \mathbf{K}_m^{\alpha}\mathbf{u}_m^{\alpha} = \mathbf{0} \\ \mathbf{u}_S = \bar{\mathbf{u}}_{IJ} \end{cases}$;

    end

  end

end

3.4 | Governing multiscale equilibrium equations

The element-wise equilibrium equations at the fine-scale of the RVE under consideration are

$$\mathbf{K}_{m(i)}^{el,\alpha}\mathbf{u}_{m(i)}^{\alpha} = \mathbf{f}_{m(i)}^{el,\alpha}, \tag{38}$$

where $\mathbf{K}_{m(i)}^{el,\alpha}$ and $\mathbf{f}_{m(i)}^{el,\alpha}$ denote the fine-element stiffness matrix and load vector defined in Eqs. (19) and (28), respectively. Substituting Eq. (31) into Eq. (38) results in the following expression:

$$\mathbf{K}_{m(i)}^{el,\alpha}\mathbf{N}_{m(i)}\mathbf{u}_{M(\alpha)} = \mathbf{f}_{m(i)}^{el,\alpha}.$$

Premultiplying this by $\mathbf{N}_{m(i)}^{T}$, the following relation is obtained:

$$\mathbf{K}_{M(\alpha),m(i)}^{el}\mathbf{u}_{M(\alpha)} = \mathbf{f}_{M(\alpha),m(i)}^{el}, \tag{39}$$

where $\mathbf{K}_{M(\alpha),m(i)}^{el}$ corresponds to the fine-element stiffness matrix that is however, defined at the coarse nodes

$$\mathbf{K}_{M(\alpha),m(i)}^{el} = \mathbf{N}_{m(i)}^{T}\mathbf{K}_{m(i)}^{el,\alpha}\mathbf{N}_{m(i)}, \tag{40}$$

and $\mathbf{f}^{\text{el}}_{\text{M}(\alpha),\text{m(i)}}$ to the corresponding vector of nodal forces

$$\mathbf{f}^{\text{el}}_{\text{M}(\alpha),\text{m(i)}} = \mathbf{N}^{T}_{\text{m(i)}} \mathbf{f}^{\text{el},\alpha}_{\text{m(i)}}, \tag{41}$$

respectively.

The reduced order coarse element equilibrium equation can be established in the following form

$$\mathbf{K}^{\text{el}}_{\text{M}(\alpha)} \mathbf{u}_{\text{M}(\alpha)} = \mathbf{f}^{\text{el}}_{\text{M}(\alpha)}, \tag{42}$$

where $\mathbf{K}^{\text{el}}_{\text{M}(\alpha)}$ and $\mathbf{f}^{\text{el}}_{\text{M}(\alpha)}$ are the coarse element stiffness matrix and load vector respectively. These quantities are a priori unknown but can be derived on the basis of strain energy equivalence between the coarse element and its underlying fine scale mesh.

The overall strain energy E_{int} of the coarse element can be, in principle, established as

$$\text{E}_{\text{int}} = \int_{\mathcal{K}_{\text{M}(\alpha)}} \boldsymbol{\varepsilon}^{T}_{\text{M}} \boldsymbol{\sigma}_{\text{M}} \, \text{d}\mathcal{K} = \mathbf{u}^{T}_{\text{M}(\alpha)} \mathbf{K}^{\text{el}}_{\text{M}(\alpha)} \mathbf{u}_{\text{M}(\alpha)} \tag{43}$$

where $\boldsymbol{\varepsilon}$ and $\boldsymbol{\sigma}$ correspond to the strain and stress fields defined over the RVE.

However, the RVE strain energy can also be considered to be additively decomposed into the contributions of the associated fine-scale elements, i.e.,

$$\text{E}_{\text{int}} = \sum_{\text{i}=1}^{\text{n}_{\text{m}_{\text{el}}}} \int_{\mathcal{K}_{\text{m(i)}}} \boldsymbol{\varepsilon}^{\alpha\,T}_{\text{m(i)}} \boldsymbol{\sigma}^{\alpha}_{\text{m(i)}} \, \text{d}\mathcal{K} = \sum_{\text{i}=1}^{\text{n}_{\text{m}_{\text{el}}}} \mathbf{u}^{\alpha\,T}_{\text{m(i)}} \mathbf{K}^{\text{el},\alpha}_{\text{m(i)}} \mathbf{u}^{\alpha}_{\text{m(i)}}, \tag{44}$$

Comparing Eq. (43) to Eq. (44), the following expression is established

$$\mathbf{u}^{T}_{\text{M}(\alpha)} \mathbf{K}^{\text{el}}_{\text{M}(\alpha)} \mathbf{u}_{\text{M}(\alpha)} = \sum_{\text{i}=1}^{\text{n}_{\text{m}_{\text{el}}}} \mathbf{u}^{\alpha\,T}_{\text{m(i)}} \mathbf{K}^{\text{el},\alpha}_{\text{m(i)}} \mathbf{u}^{\alpha}_{\text{m(i)}}. \tag{45}$$

Substituting Eq. (31) into Eq. (45) results in the following expression:

$$\mathbf{u}^{T}_{\text{M}(\alpha)} \mathbf{K}^{\text{el}}_{\text{M}(\alpha)} \mathbf{u}_{\text{M}(\alpha)} = \mathbf{u}^{T}_{\text{M}(\alpha)} \sum_{\text{i}=1}^{\text{n}_{\text{m}_{\text{el}}}} \left(\mathbf{N}^{T}_{\text{m(i)}} \mathbf{K}^{\text{el},\alpha}_{\text{m(i)}} \mathbf{N}_{\text{m(i)}} \right) \mathbf{u}_{\text{M}(\alpha)} \tag{46}$$

Eq. (46) holds if and only if

$$\mathbf{K}^{\text{el}}_{\text{M}(\alpha)} = \sum_{\text{i}=1}^{\text{n}_{\text{m}_{\text{el}}}} \mathbf{K}^{\text{el}}_{\text{M}(\alpha),\text{m(i)}} \tag{47}$$

Similarly, the following expression must hold for the RVE reduced order nodal load vector, i.e.,

$$\mathbf{f}^{\text{el}}_{\text{M}(\alpha)} = \sum_{\text{i}=1}^{\text{n}_{\text{m}_{\text{el}}}} \mathbf{f}^{\text{el}}_{\text{M}(\alpha),\text{m(i)}}. \tag{48}$$

The reduced order RVE stiffness and nodal load matrices defined in Eqs. (47) and (48) can be assembled using a direct stiffness method to eventually derive the reduced order global equilibrium equation

$$\mathbf{K}_M \mathbf{u}_M = \mathbf{f}_M, \tag{49}$$

where

$$\mathbf{K}_M = \mathop{\mathsf{A}}_{\alpha=1}^{n_{M_{el}}} \mathbf{K}^{el}_{M(\alpha)}, \qquad \mathbf{f}_M = \mathop{\mathsf{A}}_{\alpha=1}^{n_{M_{el}}} \mathbf{f}_{M(\alpha)}, \tag{50}$$

and A denotes a standard direct stiffness assembly operator.

3.5 | Solution at the coarse scale

Having computed $\mathbf{K}^{el,\alpha}_{m(i)}$, the multiscale basis functions can be evaluated by assembling $\mathbf{K}^{\alpha}_{m}$ and solving Eq. (33). The global system of equations at the coarse-scale has already been established in Eq. (49):

$$\mathbf{K}_M \mathbf{u}_M = \mathbf{f}_M, \tag{51}$$

where $\mathbf{K}_M$ and $\mathbf{f}_M$ denote the $2n_M \times 2n_M$ global coarse stiffness matrix and $2n_M \times 1$ global coarse load vector respectively. The global coarse stiffness is assembled from its local coarse stiffness contributions as shown in Eq. (50). The global coarse load vector is evaluated from the local coarse element load contributions. Finally, the EMsVEM solution is performed for the n_M coarse-scale nodes $\mathbf{u}_M$.

The micro-displacements can be evaluated from the solution of the reduced order solution of Eq. (51) using the following down-scaling procedure. The coarse element-wise displacements are first extracted from $\mathbf{u}_M$ and stored in the vector of macro-element nodal displacements $\mathbf{u}_{M(\alpha)}$, $\alpha = 1 \ldots n_{M_{el}}$. The displacements associated with the i^{th} fine-element in the α^{th} coarse-element / RVE is computed using Eq. (31). The strains and stresses associated with these fine-scale displacements can be computed as follows:

$$\boldsymbol{\varepsilon}^{\alpha}_{m(i)} = \mathbf{B}\mathbf{u}^{\alpha}_{m(i)} \qquad \boldsymbol{\sigma}^{\alpha}_{m(i)} = \mathbb{C}\boldsymbol{\varepsilon}^{\alpha}_{m(i)} \tag{52}$$

where $\mathbf{B}$ is the term provided in Eq. (23). These strains and stresses are uniform over the i^{th} fine-element domain. To allow for compatibility with traditional post-processing routines, the element is decomposed into sub-triangles. The evaluated stresses and strains are then associated with desired quadrature integration points.

The process flow of the EMsVEM is graphically shown in Fig. 7.

4 | APPLICATIONS

The EMsVEM is verified by comparing against the standard VEM and analytical solutions. A first order VEM (k=1), is used in all cases. In the following, the four different element-types illustrated in Fig. 8 are used for the verification. The Centroidal Voronoi Tessellations (CVT) shown in Fig. 8b were generated by Lloyd's algorithm [66] with the generated seeds being forced to coincide with the associated centroid of each polygon. The Random Voronoi Elements (RAND) shown in Fig. 8d were created by a random set of seeds. The polygonal meshes were generated using PolyMesher

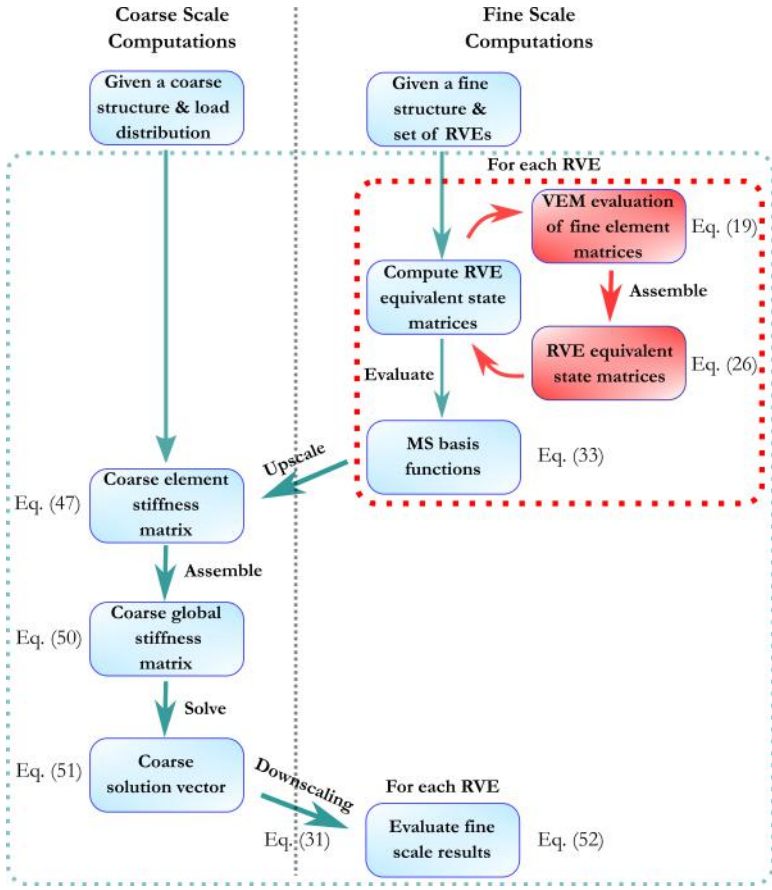


FIGURE 7 Process flow of the Enhanced Multiscale Virtual Element procedure.

[67].

To investigate the fidelity of the proposed method, the $\mathcal{L}_2$ norm and the $\mathcal{H}_1$ semi-norm are employed for the displacement and stress/strain approximations, respectively.

$$||\mathbf{u}_{h,M} - \mathbf{u}_{ex,M}||_{\mathcal{L}_2} = \sqrt{\frac{1}{n_{M_{el}}} \sum_{i=1}^{n_{M_{el}}} \frac{\langle \mathbf{u}_{M(i)} - \mathbf{u}_{M(i),ex}, \mathbf{u}_{M(i)} - \mathbf{u}_{M(i),ex} \rangle}{\langle \mathbf{u}_{M(i),ex}, \mathbf{u}_{M(i),ex} \rangle}} \tag{53a}$$

$$|\mathbf{u}_{h,M} - \mathbf{u}_{ex,M}|_{\mathcal{H}_1} = \sqrt{\frac{1}{n_{M_{el}}} \sum_{i=1}^{n_{M_{el}}} \frac{\langle \boldsymbol{\varepsilon}(\mathbf{u}_{M(i)} - \mathbf{u}_{M(i),ex}), \boldsymbol{\sigma}(\mathbf{u}_{M(i)} - \mathbf{u}_{M(i),ex}) \rangle}{\langle \boldsymbol{\varepsilon}(\mathbf{u}_{M(i)),ex}), \boldsymbol{\sigma}(\mathbf{u}_{M(i),ex}) \rangle}} \tag{53b}$$

where $\langle \cdot, \cdot \rangle$, $\mathbf{u}_{h,M}$ and $\mathbf{u}_{ex,M}$ denotes the scalar product, the numerically evaluated coarse-nodal displacements and the reference solution at the coarse-nodes, respectively.

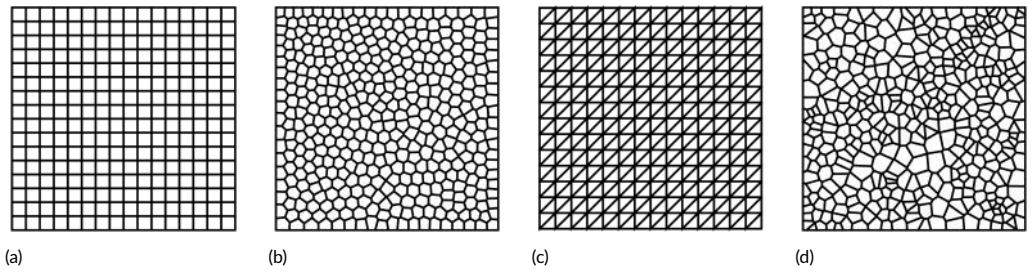


FIGURE 8 Element types (a) Quadrilateral (b) Centroidal Voronoi Tessellations (c) Triangle (d) Random Voronoi.

4.1 | Square plate under tension

The case of the homogeneous square plate shown in Fig. 9a is considered herein. The plate is fully clamped at the bottom and subjected to a traction vector $\mathbf{t} = \begin{bmatrix} 0 & 10 \end{bmatrix}^T$ at the top edge as shown in Fig. 9. The vertical sides are left unconstrained. The material has a Young modulus $E = 10^7$ N/m^2 and a Poisson's ratio $\nu = 0.3$. A mesh

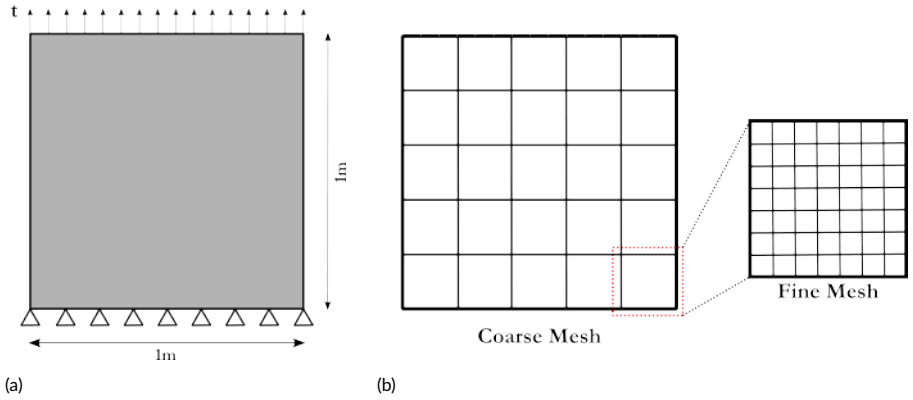


FIGURE 9 (a) Geometry and boundary conditions (b) Multiscale mesh

discretization of 40×40 quadrilateral plane stress elements with full integration is employed for the FEM and VEM as shown. The EMsFEM and EMsVEM solutions are derived considering a coarse mesh consisting of 5×5 quadrilateral coarse-elements. This is illustrated by the coarse-grid in Fig. 9b. Each coarse-element contains 8×8 quadrilateral fine-elements. The case of rectangular elements only is considered in this case for comparisons between the virtual element and finite element based methods to be meaningful. Linear boundary conditions are used to derive the multiscale basis functions for this example.

The $\mathcal{L}_2$-norm of the errors between the FEM and the EMsFEM is 5.7×10^{-3}. The corresponding norm for the VEM to EMsVEM comparison is practically the same, i.e., 5.5×10^{-3}. A convergence study is also performed by retaining the same number of coarse-elements as shown in Fig. 9b, and increasing the number of fine-scale elements.

The evolution of the $\mathcal{L}_2$-norm as a function of the number of elements is provided in Fig. 10, where the EMsFEM and EMsVEM behave in an identical fashion. A reference slope is also shown for comparison of error convergence

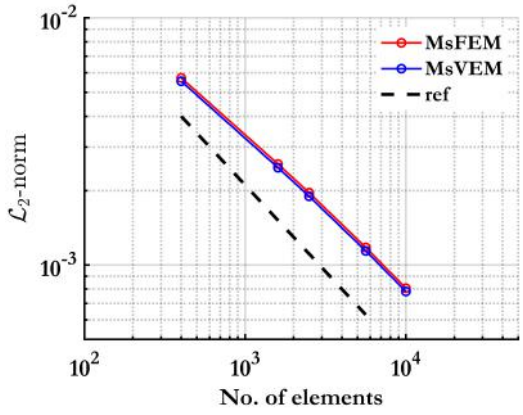


FIGURE 10 Relative error plots: EMsFEM and EMsVEM evaluated displacements compared against FEM and VEM displacements respectively, at coarse-nodes. A reference slope is provided for comparison.

rates. Near optimal rates are observed for both methods.

4.2 | Cantilever beam subjected to parabolic traction

The cantilever structure shown in Fig. 11 is fully clamped at the left and is subjected to a parabolic traction at its free end. The domain has a length L = 8 m, height D = 4 m and thickness t = 1m. The material has a Young's modulus E = 10^7 N/m^2 and a Poisson's ratio ν = 0.3. The parabolic traction at the free end assumes the following form

$$t_x = 0, \qquad t_y = \frac{p(D^2/4 - y^2)}{2I}, \tag{54}$$

where p = −1000 N is the total load applied and $I = tD^3/12$.

The displacements are evaluated analytically on the basis of plane strain assumptions [68] according to Eqs. (55a) and (55b) for the horizontal and vertical component, respectively

$$u_x = -\frac{py}{6\bar{E}I}\left((6L - 3x)x + (2 + \bar{\nu})y^2 - \frac{3D^2}{2}(1 + \bar{\nu})\right) \tag{55a}$$

$$u_y = \frac{p}{6\bar{E}I}\left(3\bar{\nu}y^2(L - x) + (3L - x)x^2\right), \tag{55b}$$

where $\bar{E} = \frac{E}{1-\nu^2}$ and $\bar{\nu} = \frac{\nu}{1-\nu}$.

Three different geometries of micro-structure elements are considered; these are summarized in Table 1. The problem is also solved with the EMsFEM for the case of quadrilateral micro-elements.

The convergence behaviour of the EMsFEM and the EMsVEM with respect to different element-types specified in Table 1 is investigated in the form of relative $\mathcal{L}_2$ and $\mathcal{H}_1$ error convergence plots shown in Fig. 12 and Fig. 13. The evolution of the errors are studied as a function of the number of coarse-elements, for 5 micro-structure configurations. At low discretizations (Figs. 12 a, 12 b, 13 a and 13 b), the EMsFEM solution provides the best accuracy. However, for finer micro-structure configurations (Figs. 12 e and 13 e), the CVT , RAND and QUAD element meshes used by the EMsVEM offer accuracies approaching the QUAD EMsFEM method, over all coarse-element mesh discretizations.

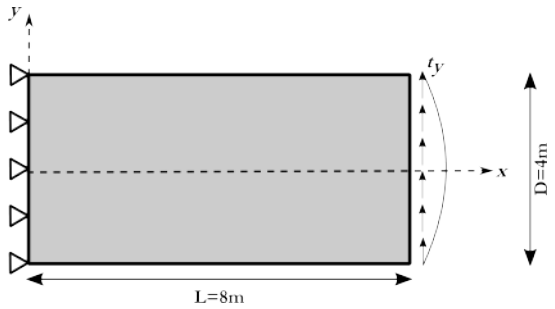


FIGURE 11 Schematic diagram of a cantilever beam subject to parabolic tractions

Element Type	Abbreviation	EMsFEM	EMsVEM
Quadrilateral	QUAD	□	□
Centroidal Voronoi Tessellations	CVT	-	★
Random	RAND	-	*

TABLE 1 Convergence behaviour of EMsFEM and EMsVEM are studied with respect to different element types

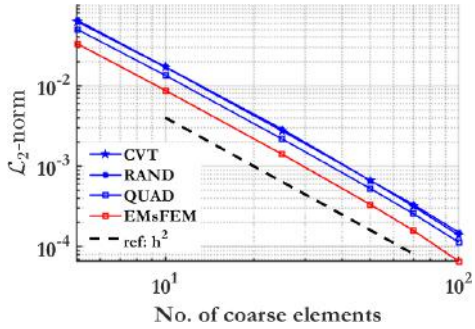
The convergence rates are provided in Table 2 and Table 3. These are found to nearly coincide with the expected theoretical slopes of -2 in the $\mathcal{L}_2$ and -1 in the $\mathcal{H}_1$ relative error norms, respectively. The theoretical convergence rates are derived in [69]. One can conclude that near-optimal convergence rates are obtained by the method over all coarse-element discretizations. This suggests that the EMsVEM is a viable alternative to the EMsFEM when a flexible mesh generation is required to account for fine-scale heterogeneities.

No. of micro-elements	CVT	RAND	QUAD	EMsFEM
4	-1.8868	-1.9303	-1.9012	-1.9244
16	-1.9547	-1.8829	-1.9255	-1.9325
100	-1.9635	-1.9200	-1.9343	-1.9355
625	-1.8931	-1.9358	-1.9358	-1.9359
2500	-1.9799	-1.9359	-1.9359	-1.9360

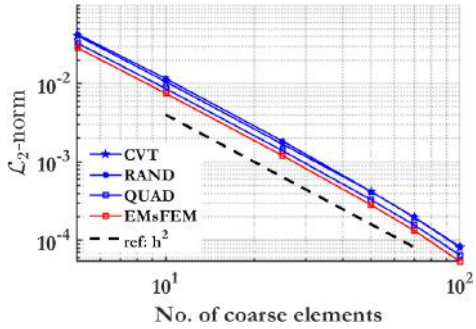
TABLE 2 Convergence rates of the $\mathcal{L}_2$ relative error norm

The primary advantage offered by the EMsVEM over the EMsFEM lies in its ability to handle any kind of micro-structure configuration. This is illustrated by computing displacements with the EMsVEM for a 10×10 coarse-element discretization with an arbitrarily chosen micro-structure definition described in Table 4. The contour plots of the resulting total displacements are shown in Fig. 14.

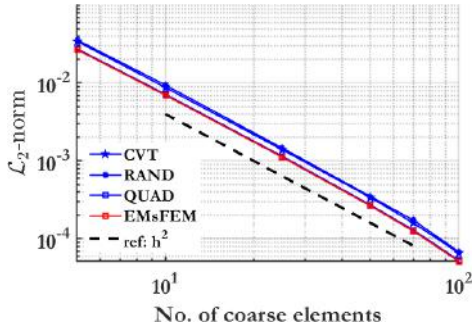
Figs. 14a and 14b illustrate the method's ability to sufficiently handle widely varying micro-structural configura-



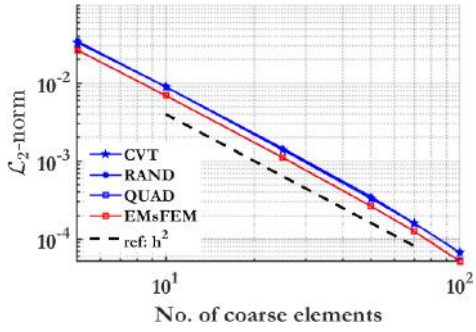
(a) 4 micro-elements



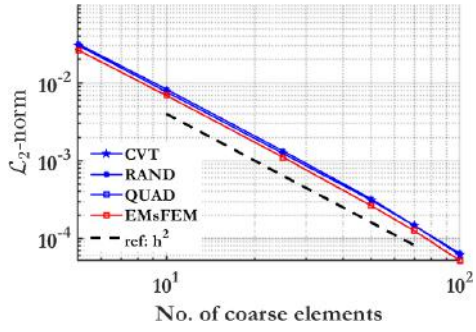
(b) 16 micro-elements



(c) 100 micro-elements



(d) 625 micro-elements



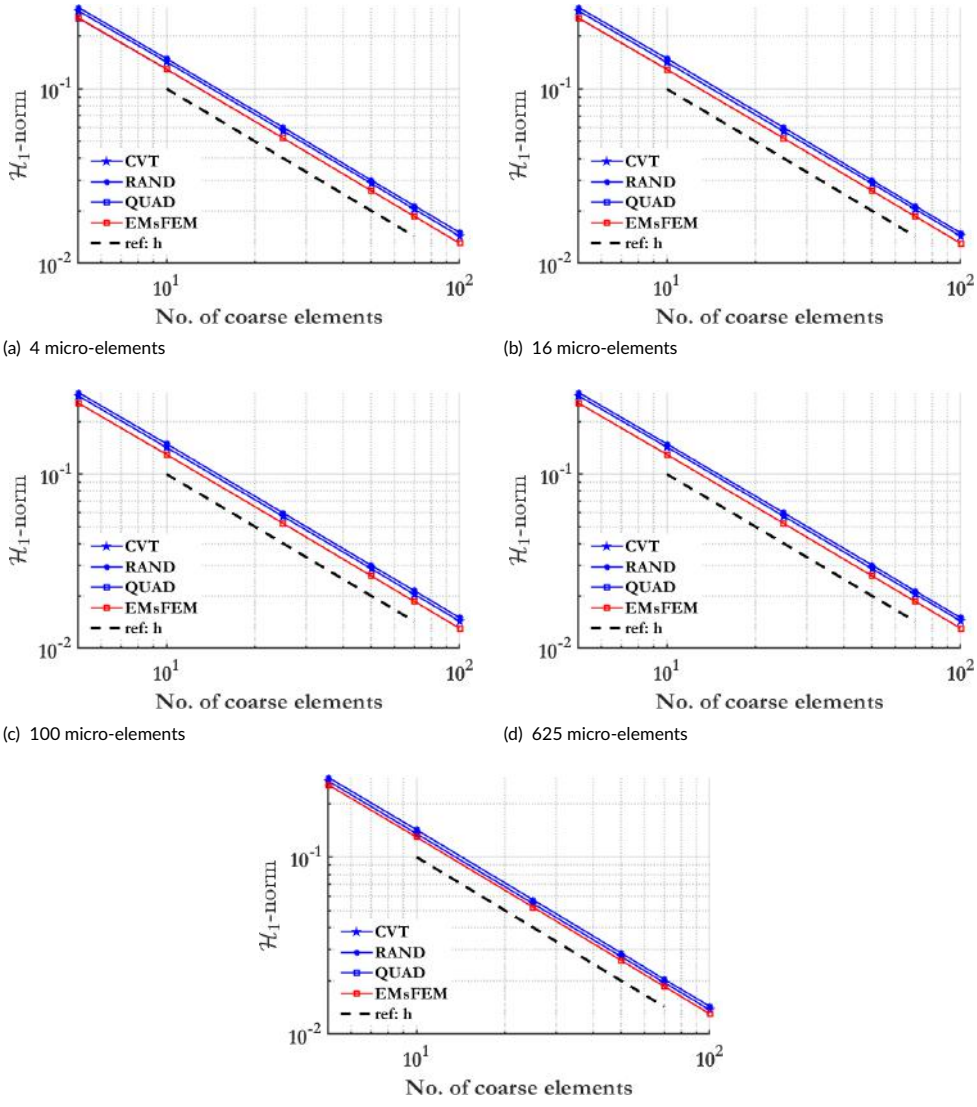
(e) 2500 micro-elements

FIGURE 12 Relative $\mathcal{L}_2$ error convergence plots for 5 micro-structure configurations

tions within a single problem. The corresponding y-displacements obtained for the free end of the neutral axis of the structure are compared against the analytical solution in Table 5.

4.3 | Cantilever beam with a periodic microstructure

In this example, a cantilever beam with periodically repeating circular inclusions is considered. A 30×6 coarse grid is created over this domain as shown in Fig. 15a. The micro-structure enclosed within each coarse element contains a



(a) 4 micro-elements

(b) 16 micro-elements

(c) 100 micro-elements

(d) 625 micro-elements

(e) 2500 micro-elements

FIGURE 13 Relative $\mathcal{H}_1$ error convergence plots for 5 micro-structure configurations

circular inclusion as illustrated in Fig. 15b. The material properties considered are $E_m = 1$GPa, $\nu_m = 0.3$ and $E_i = 10$GPa, $\nu_i = 0.3$ for the matrix m and the inclusion i, respectively.

This example is provided to establish the EMsVEM as a useful tool in the analysis of composites for driving down computational costs. To prove this claim, we draw attention to the relaxed conformity requirements on polygonal meshes when compared with conventional quadrilateral finite element meshes. This flexibility is exploited to minimize the number of nodes involved in the micro-structural discretization while still retaining satisfactory accuracy. In

No. of micro-elements	CVT	RAND	QUAD	EMsFEM
4	-0.9681	-0.9724	-0.9695	-0.9750
16	-0.9781	-0.9709	-0.9752	-0.9773
100	-0.9809	-0.9765	-0.9780	-0.9784
625	-0.9742	-0.9785	-0.9785	-0.9786
2500	-0.9829	-0.9786	-0.9786	-0.9786

TABLE 3 Convergence rates of the $\mathcal{H}_1$ relative error norm

No.	Element Type	No. of Elements
1	TRI	64
2	QUAD	25
3	RAND	36
4	CVT	100

TABLE 4 Arbitrarily chosen micro-structural definition assigned periodically to each unit-cell

Numerical solution	Analytical solution
-2.91×10^{-3} m	-2.96×10^{-3} m

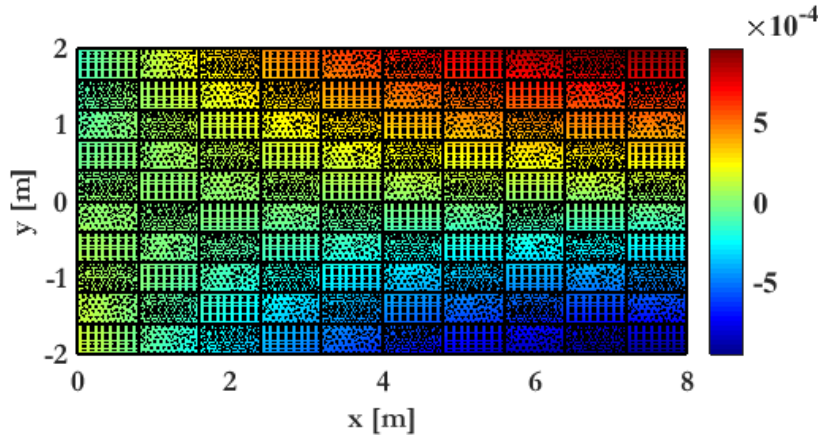
TABLE 5 Neutral axis free end displacements in the y-direction computed using EMsVEM and analytical solution.

particular, the microstructure is discretized using three approaches, i.e, uniform quadrilateral elements, uniform polygonal CVT elements, and an adaptively refined mesh as shown in Fig. 16. Mesh 1, schematically depicted in Fig. 16a, comprises 10,116 uniform quadrilateral elements and is treated as a reference solution. Mesh 2, contains 5000 uniform CVT elements and is shown in Fig. 16b. Mesh 3 (Fig. 16c) is adapted from [70], wherein efficient polygonal discretizations are exploited for performing a non-linear analysis on fiber composites. The properties of each mesh are summarized in Table 6. Periodic boundary conditions are used to derive the multiscale basis functions, for all cases.

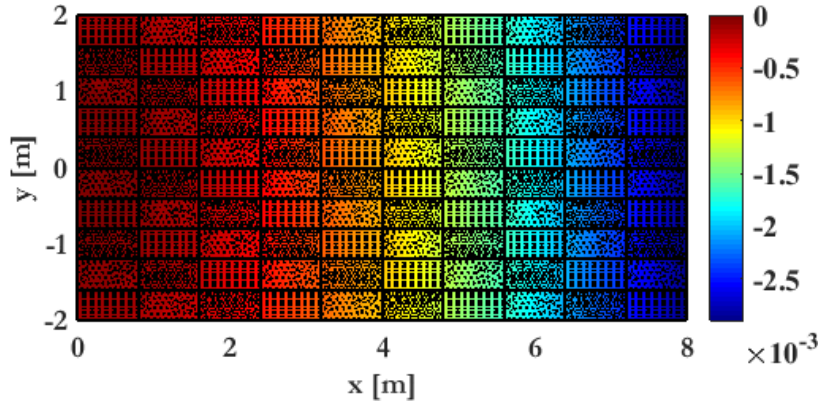
Label	Element-Type	No. of Elements	No. of Nodes	Description
Mesh 1	QUAD	10116	10299	Reference Solution
Mesh 2	CVT	5000	7278	Uniform mesh
Mesh 3	POLY	1441	2452	Adaptive mesh

TABLE 6 Meshes used to discretize the micro-structure

The $\mathcal{L}_2$ and $\mathcal{H}_1$ error norms obtained by Mesh 2 and Mesh 3 are summarized in Table 7. It is observed that both



(a) $\mathbf{u}_x$: Displacements in x-direction



(b) $\mathbf{u}_y$: Displacements in y-direction

FIGURE 14 Total displacement contours for the arbitrarily chosen micro-structure defined in Table 4 - Units are in m.

meshes achieve comparable accuracy despite the fact that Mesh 2 has a considerably larger number of DoFs. The computational time for all cases, averaged over 5 runs is shown in Table 7. This result illustrates appreciable benefits attainable through the EMsVEM by using more flexible mesh generation capabilities.

	$\mathcal{L}_2$ **error norm**	$\mathcal{H}_1$ **error norm**	Analysis time [sec]
Mesh 1	-	-	3870
Mesh 2	0.0062	0.047	1015
Mesh 3	0.0056	0.050	215

TABLE 7 $\mathcal{L}_2$ and $\mathcal{H}_1$ error norms for Mesh 2 and Mesh 3 using Mesh 1 as a reference solution and computational times.

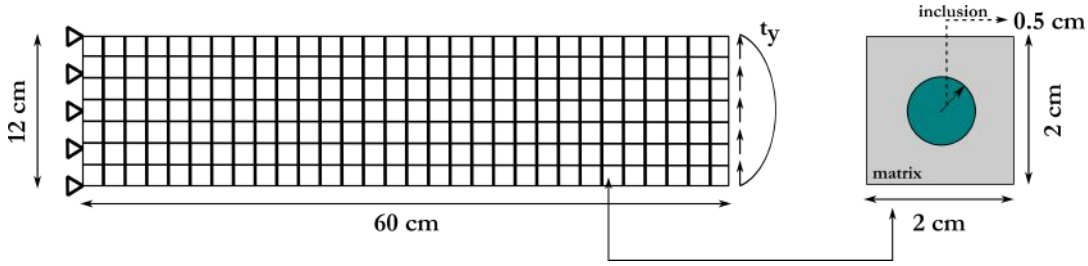


FIGURE 15 Cantilever with a periodic micro-structure: Geometry and boundary conditions

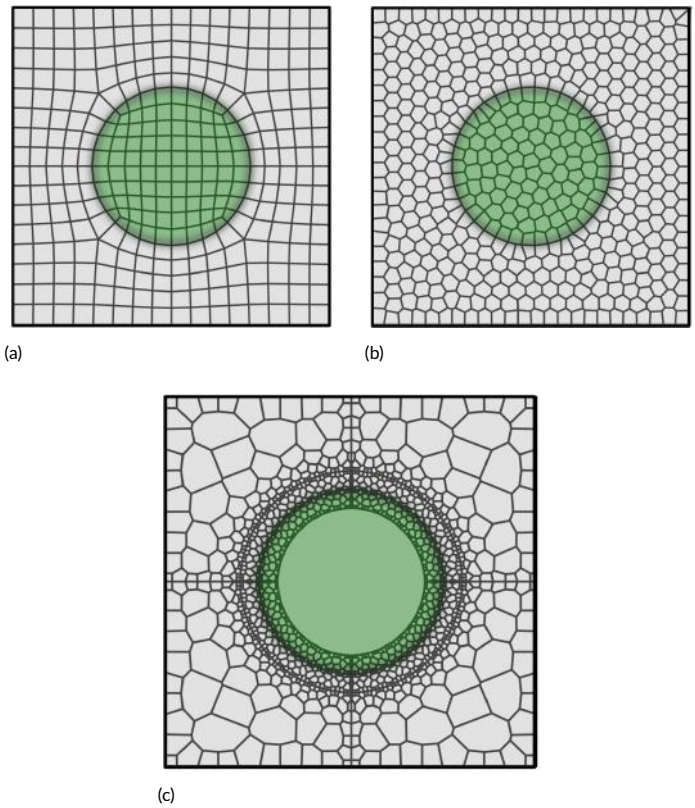
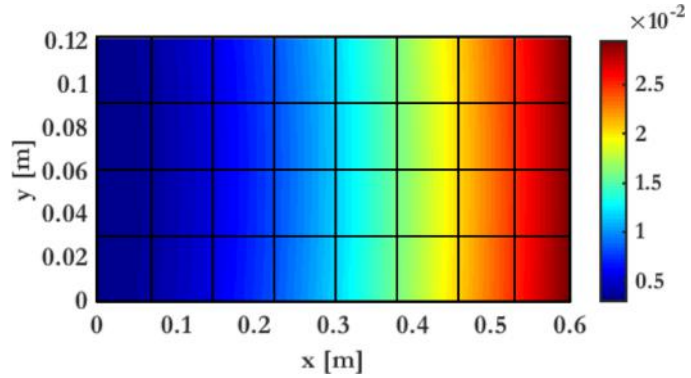
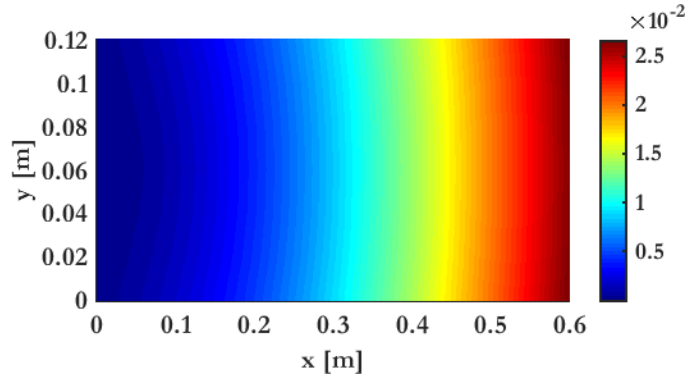


FIGURE 16 Element types (a) Uniform Quadrilateral, (b) Centroidal Voronoi Tesselations (c) Specially designed polygonal mesh

Contour plots of the total displacements are shown in Fig. 17a and Fig. 17b for the EMsVEM-P model with Mesh 3 and the standard VEM, respectively. The corresponding von-Mises stresses $\boldsymbol{\sigma}_{\text{VM}}$ are provided in Fig. 18a and Fig. 18b for the EMsVEM and the VEM, respectively. Both methods provide practically identical results, as also manifested by the $\mathcal{L}_2$ and $\mathcal{H}_1$ norms shown in Table 7.



(a) Evaluated using the EMsVEM-P



(b) Evaluated using VEM

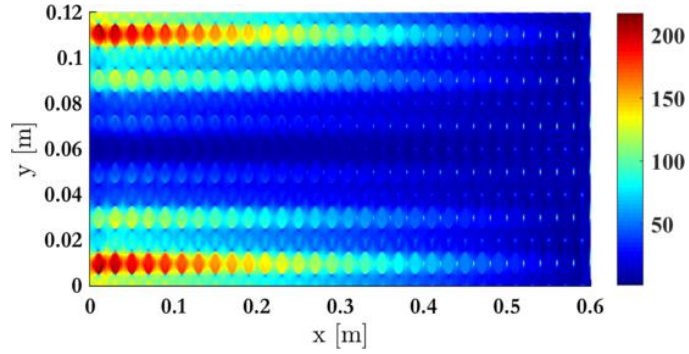
FIGURE 17 Total displacement contours - Units are in m

4.4 | Cantilever beam with a highly heterogeneous material distribution subjected to parabolic traction

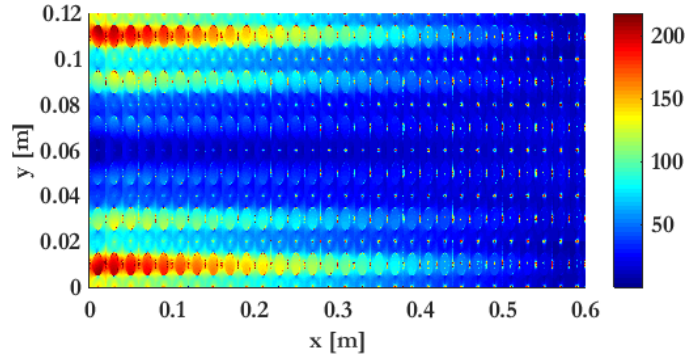
The case of the cantilever beam examined in Section 4.2 and shown in Fig. 11 is considered here also. In this case, we investigate the effect of the material heterogeneity on the performance of the proposed EMsVEM both in terms of accuracy and computational efficiency. The results obtained from the EMsVEM are compared against the standard VEM. Comparisons are also provided against the standard FEM and the EMsFEM for the case of quadrilateral elements. To investigate the effect of the assumed boundary conditions for the evaluation of the multiscale basis functions, two variants are considered, i.e., linear and periodic boundary conditions. The abbreviations of the multiscale methods used in this example are shown in Table 8.

Three cases are considered vis-a-vis the geometry of the fine-scale, i.e., with heterogeneities having (a) QUAD, (b) CVT and (c) RAND shapes. Furthermore, two discretizations are examined per case, to assess the the effect of the scale separation on the accuracy of the EMsVEM. The models employed along with their corresponding coarse and fine-scale discretizations are summarized in Table 9.

Each micro-element in both discretization schemes is randomly assigned a Young's modulus generated by a



(a) Evaluated using the EMsVEM-P



(b) Evaluated using VEM

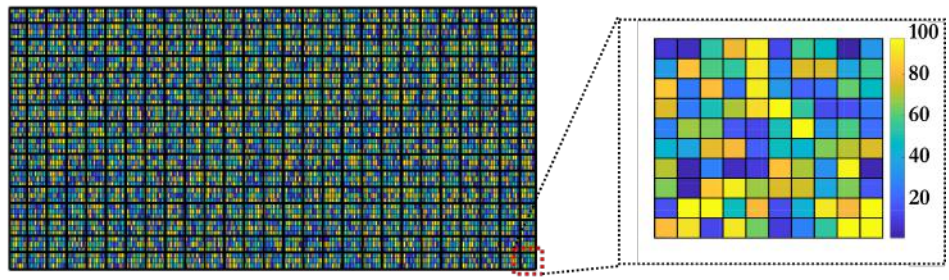
FIGURE 18 Von Mises stress contours - Units are in MPa

Abbreviation	Method/ Boundary conditions
EMsFEM-L	Multiscale Finite Element Method/ Linear
EMsFEM-P	Multiscale Finite Element Method/ Periodic
EMsVEM-L	Multiscale Virtual Element Method/ Linear
EMsVEM-P	Multiscale Virtual Element Method/ Periodic

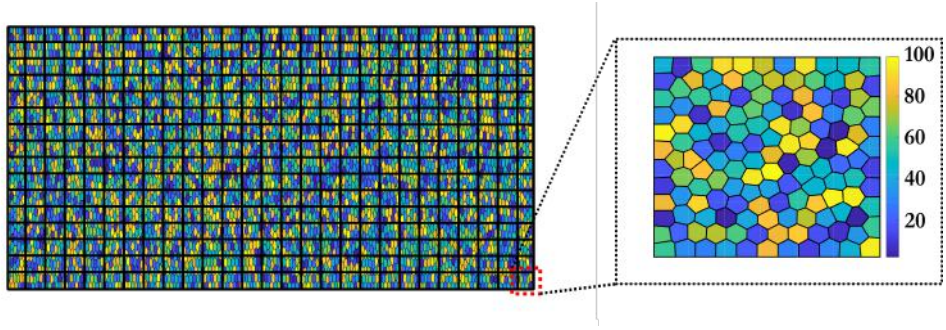
TABLE 8 Multiscale method labeling based on the boundary conditions used to evaluate the multiscale basis functions.

uniform distribution. The lower and upper bounds of the Young's modulus are considered to be $E_l = 1$ GPa and $E_u = 100$ GPa, respectively. The material distribution associated with discretization scheme B is illustrated in Fig. 19. This material distribution within the coarse-element is assumed to repeat periodically over the entire domain. The Poisson's ratio is 0.3 in all cases.

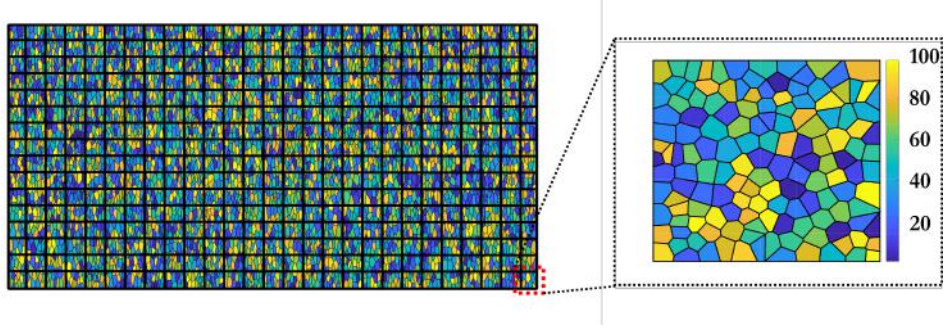
380 The profile of the vertical displacements along the neutral axis (y = 0) of the structure are shown for all the cases



(a)



(b)



(c)

FIGURE 19 Snapshots of randomly distributed Young's modulus in a typical coarse-element with (a) 144 QUAD fine-elements (b) 144 CVT fine-elements (c) 144 RAND fine-elements - Units are in GPa

in Fig. 20. For discretization scheme A, shown in Figs. 20a, 20c, and 20e, a significant deviation in the displacements obtained through FEM and VEM and the multiscale solutions is observed. The methods using periodic boundaries, i.e., EMsFEM-P and EMsVEM-P are found to approximate the complete solutions better than EMsFEM-L and EMsVEM-L.

On the other hand, in the case of the finer discretization scheme B, as shown in Figs. 20b, 20d and 20f, all methods provide practically identical results. The relative error of the EMsVEM to the VEM solution for linear and

	Discretization A			Discretization B		
	Full Mesh	Multiscale Mesh		Full Mesh	Multiscale Mesh	
	[-]	Macro	Micro	[-]	Macro	Micro
Quad	720	15x3	16	72000	50x12	144
CVT	720	15x3	16	72000	50x12	144
RAND	720	15x3	16	72000	50x12	144

TABLE 9 Number of elements considered in each run. The discretization scheme B is illustrated in Fig. 19.

periodic boundary conditions is shown in Table 10. These are practically identical to the relative errors of the EMsFEM to FEM solution which are also shown in Table 10 for completeness.

The horizontal displacements ($\mathbf{u}_x$) obtained for the micro-structure using RAND elements and the discretization scheme B are indicatively shown in Fig. 21a and Fig. 21b for the VEM and EMsVEM-P, respectively. The Von-Mises stress distribution ($\boldsymbol{\sigma}_{\text{VM}}$) obtained for all cases using the discretization scheme B, are shown in Fig. 22.

	Discretization A			Discretization B		
	Quad	CVT	RAND	Quad	CVT	RAND
EMsFEM-L	0.2246	-	-	0.0562	-	-
EMsFEM-P	0.1402	-	-	0.0072	-	-
EMsVEM-L	0.2256	0.1953	0.242	0.061	0.0807	0.12
EMsVEM-P	0.1440	0.1111	0.1618	0.0076	0.0144	0.0233

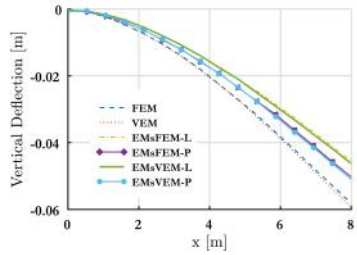
TABLE 10 Relative $\mathcal{L}_2$ errors between standard solutions and multiscale solutions computed at coarse-nodes.

4.4.1 | Discussion on computational gains

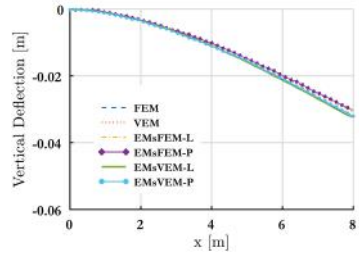
To assess the computational effectiveness of the EMsVEM, we evaluate the time required for the assembly and inversion of the global state matrices for all the methods employed. The average times over five runs are shown in Fig. 23. There is an appreciable reduction in the time taken for assembly and matrix inversion when using multiscale methods, as evidenced by Fig. 23a and Fig. 23b. This is to be expected as the number of nodes involved in the multiscale assembly and solution procedures are significantly reduced when compared to the standard FEM and VEM.

Furthermore, the time required in the EMsVEM is significantly lower when compared to the EMsFEM. This is attributed to the fact that there are no iterative evaluations over quadrature integration points done in the case of first order EMsVEM as opposed to the EMsFEM. In Fig. 23c, post-processing times are also compared for all methods. This is to account for the effect potential overheads might have on the computational efficiency of the methods, especially within an incremental/ iterative solution scheme. The time required for the multiscale methods to downscale the coarse-solution is also captured here.

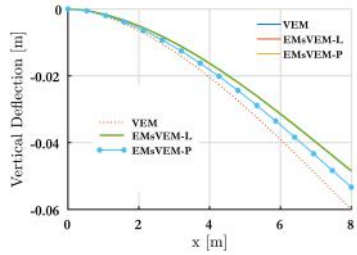
While it is clear from Fig. 23d that post-processing is a significant factor for the multiscale methods, Fig. 23c reveals that this is indeed lower than in the corresponding fine scale implementations. This is due to the fact that down-



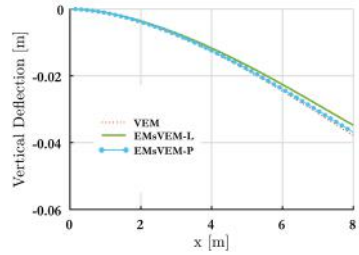
(a) Discretization Scheme A with QUAD elements



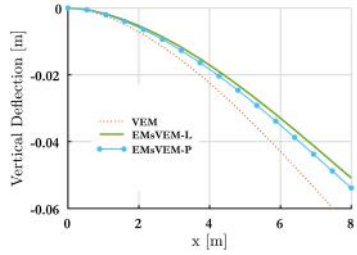
(b) Discretization Scheme B with QUAD elements



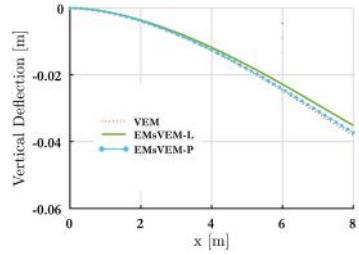
(c) Discretization Scheme A with CVT elements



(d) Discretization Scheme B with CVT elements



(e) Discretization Scheme with RAND elements



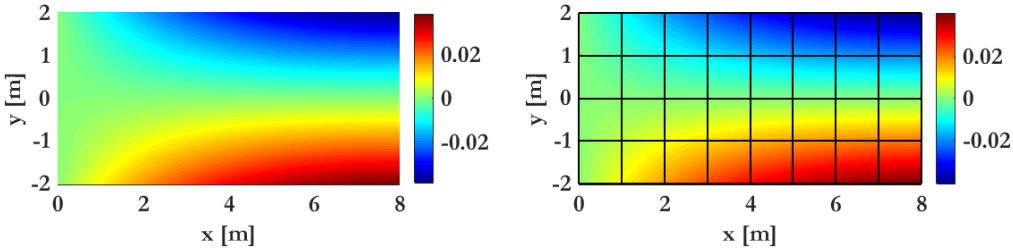
(f) Discretization Scheme B with RAND elements

FIGURE 20 Vertical displacement along the neutral axis of the cantilever

scaling is performed per coarse element, hence implicitly vectorizing the corresponding strain and stress computation loops.

5 | CONCLUSIONS

A novel Enhanced Multiscale Virtual Element Method, termed the EMsVEM, is developed for the analysis of heterogeneous elastic structures. The novelty of the EMsVEM rests on the utilisation of the Virtual Element Method at the fine scale to enable flexible meshing that can efficiently capture any type of domain heterogeneity. Numerically computed multiscale basis functions upscale the heterogeneities allowing the solution procedure to be performed at the coarse scale. Subsequently, the solution is downscaled to capture local fine scale variations.



(a) Evaluated using the VEM with RAND elements

(b) Evaluated using EMsVEM-P with RAND elements

FIGURE 21 Visualization of $\mathbf{u}_x$ (x-displacement) distributions for RAND elements with discretization scheme B.

The EMsVEM efficiently models complex geometries at the micro-scale while retaining the accuracy of the standard Finite Element Method and its EMsFEM variant. The proposed method is validated against analytical solutions and the standard FEM and VEM. Our convergence studies demonstrate that the EMsVEM is well behaved even in cases of significant heterogeneities in the fine-scale.

Based on the computational advantages of the VEM, stemming from its flexible element geometries and the relaxed requirement on domain integration, the EMsVEM is shown to significantly reduce the complexity of the computational problem compared to the standard FEM and the EMsFEM. The computational upscaling procedure proposed is shown to significantly reduce the computational costs compared to the standard VEM.

The ability of the method to handle adaptively refined virtual element meshes at the fine scale is also demonstrated. According to our observations, intelligently chosen polygonal fine scale discretizations drive down computational costs without significantly sacrificing accuracy.

Similar to the standard EMsFEM, the boundary conditions used to derive the multiscale basis functions play a critical role in the accuracy of the method. Periodic boundary conditions are shown to offer a better account of the heterogeneous behaviour when significant heterogeneities are present.

A limitation of the proposed method is that it is confined to the utilisation of quadrilateral coarse element geometries. Extending the method to account for the most general case of polygonal coarse element representations is currently under development.

acknowledgements

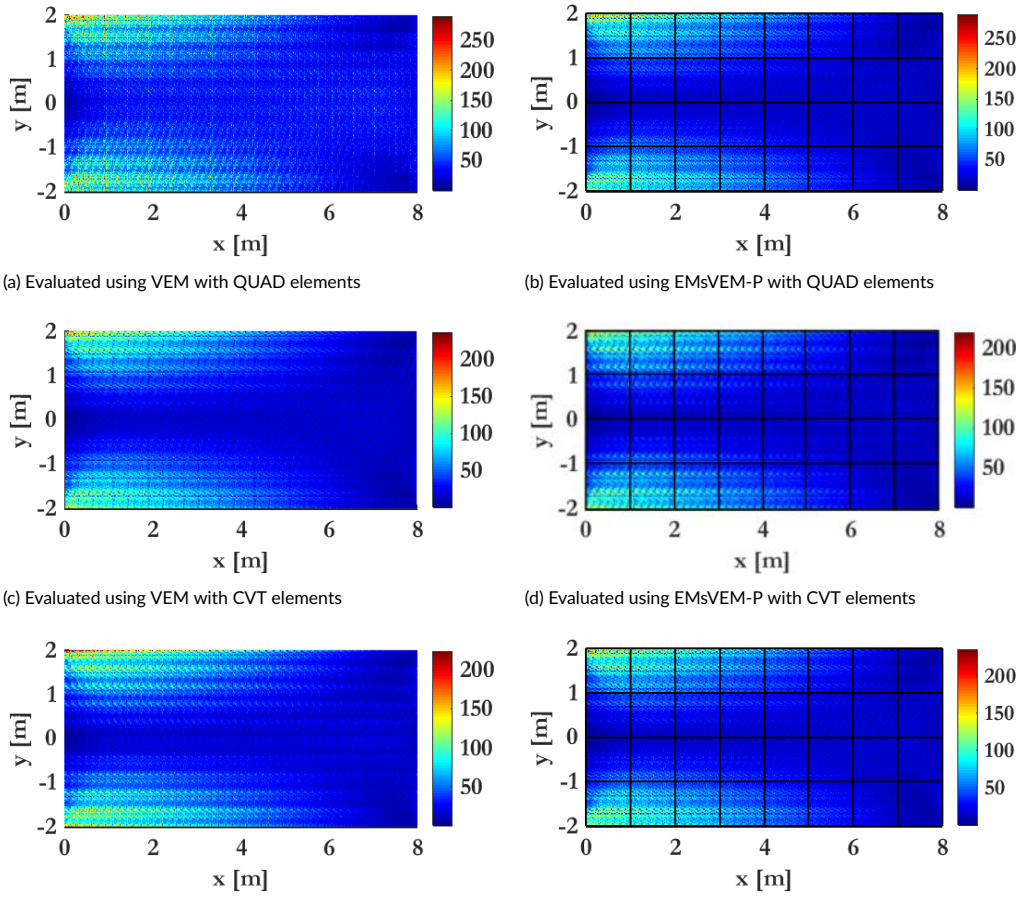
This work has been carried out under the auspices of the grant: "European industrial doctorate for advanced, lightweight and silent, multifunctional composite structures — N2N". The N2N project is funded under the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie Actions Grant: 765472.

references

[1] Herakovich CT. Mechanics of composites: a historical review. Mechanics Research Communications 2012;41:1–20.

[2] Triantafyllou SP, Chatzi EN. A hysteretic multiscale formulation for nonlinear dynamic analysis of composite materials. Computational Mechanics 2014;54:763–787.

[3] Zienkiewicz OC, Taylor RL, Zhu JZ. The finite element method: its basis and fundamentals. Elsevier; 2005.



(a) Evaluated using VEM with QUAD elements

(b) Evaluated using EMsVEM-P with QUAD elements

(c) Evaluated using VEM with CVT elements

(d) Evaluated using EMsVEM-P with CVT elements

(e) Evaluated using VEM with RAND elements

(f) Evaluated using EMsVEM-P with RAND elements

FIGURE 22 Visualization of σ_{VM} (Von-Mises stress) distributions for all element cases with discretization scheme B.

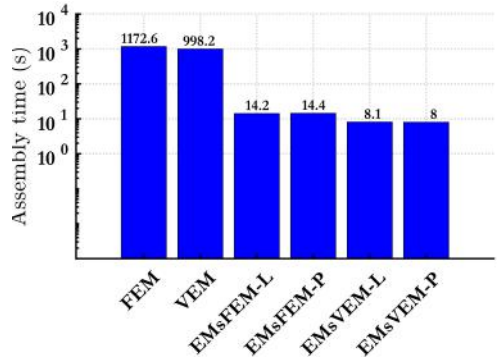
[4] Zohdi TI, Wriggers P. An introduction to computational micromechanics. Springer Science & Business Media; 2008.

[5] Andrade JE, Tu X. Multiscale framework for behavior prediction in granular media. Mechanics of Materials 2009;41(6):652–669.

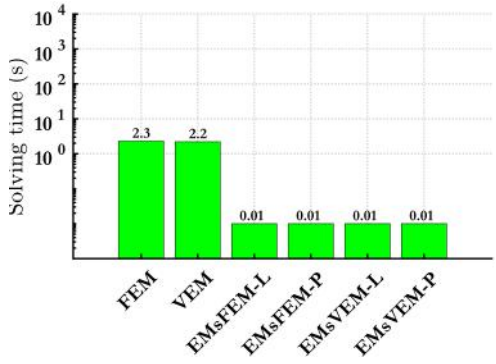
[6] Kanouté P, Boso D, Chaboche J, Schrefler B. Multiscale methods for composites: a review. Archives of Computational Methods in Engineering 2009;16(1):31–75.

[7] Markov K, Preziosi L. Heterogeneous media: micromechanics modeling methods and simulations. Springer Science & Business Media; 2012.

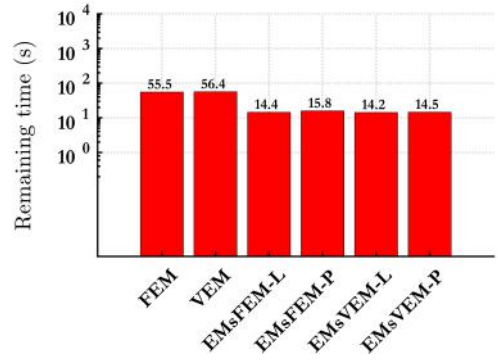
[8] Tootkaboni M, Graham-Brady L. A multi-scale spectral stochastic method for homogenization of multi-phase periodic composites with random material properties. International journal for numerical methods in engineering 2010;83(1):59–90.



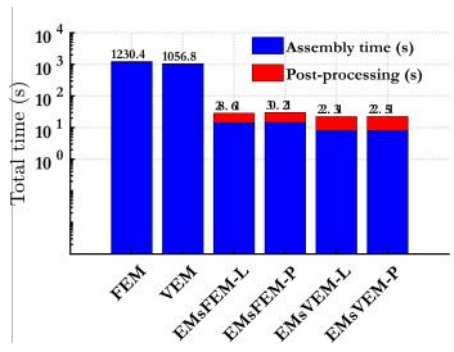
(a) Time required to assemble global stiffness matrix



(b) Time required to solve the linear system of equations



(c) Time required for post-processing



(d) Total analysis time

FIGURE 23 Comparing the time taken at critical points during solving.

[9] Babuška I. Homogenization approach in engineering. In: Computing methods in applied sciences and engineering Springer; 1976.p. 137–153.

[10] Aghdam M, Pavier M, Smith D. Micro-mechanics of off-axis loading of metal matrix composites using finite element analysis. International journal of solids and structures 2001;38(22-23):3905–3925.

[11] Taliercio A. Macroscopic strength estimates for metal matrix composites embedding a ductile interphase. International Journal of Solids and Structures 2007;44(22-23):7213–7238.

[12] Azizi R, Niordson CF, Legarth BN. Size-effects on yield surfaces for micro reinforced composites. International Journal of Plasticity 2011;27(11):1817–1832.

[13] Terada K, Hori M, Kyoya T, Kikuchi N. Simulation of the multi-scale convergence in computational homogenization approaches. International Journal of Solids and Structures 2000;37(16):2285–2311.

[14] Hou T, Efendiev Y. Multiscale Finite Element Methods; 2009.

[15] Ren M, Cong J, Wang B, Guo X. Extended multiscale finite element method for small-deflection analysis of thin composite plates with aperiodic microstructure characteristics. Composite Structures 2017 jan;160:422 – 434. `http://www.sciencedirect.com/science/article/pii/S0263822316312715`.

[16] Tabarraei A, Sukumar N. Conforming polygonal finite elements. Int J NumerMethods Engrg 2004;61:2045–2066.

[17] Bishop JE. A displacement-based finite element formulation for general polyhedra using harmonic shape functions. International Journal for Numerical Methods in Engineering 2014;97(1):1–31.

[18] Floater MS, Hormann K, Kós G. A general construction of barycentric coordinates over convex polygons. advances in computational mathematics 2006;24(1-4):311–331.

[19] Kuznetsov Y, Repin S. New mixed finite element method on polygonal and polyhedral meshes. Russian Journal of Numerical Analysis and Mathematical Modelling rnam 2003;18(3):261–278.

[20] Mousavi S, Sukumar N. Numerical integration of polynomials and discontinuous functions on irregular convex polygons and polyhedrons. Computational Mechanics 2011;47(5):535–554.

[21] Natarajan S, Bordas S, Roy Mahapatra D. Numerical integration over arbitrary polygonal domains based on Schwarz–Christoffel conformal mapping. International Journal for Numerical Methods in Engineering 2009;80(1):103–134.

[22] Sukumar N, Tabarraei A. Conforming polygonal finite elements. International Journal for Numerical Methods in Engineering 2004;61(12):2045–2066.

[23] Manzini G, Russo A, Sukumar N. New perspectives on polygonal and polyhedral finite element methods. Mathematical Models and Methods in Applied Sciences 2014;24(08):1665–1699.

[24] Talischi C, Paulino GH, Pereira A, Menezes IF. Polygonal finite elements for topology optimization: A unifying paradigm. International journal for numerical methods in engineering 2010;82(6):671–698.

[25] Talischi C, Paulino GH, Pereira A, Menezes IF. PolyTop: a Matlab implementation of a general topology optimization framework using unstructured polygonal finite element meshes. Structural and Multidisciplinary Optimization 2012;45(3):329–357.

[26] Spring DW, Leon SE, Paulino GH. Unstructured polygonal meshes with adaptive refinement for the numerical simulation of dynamic cohesive fracture. International Journal of Fracture 2014;189(1):33–57.

[27] Sukumar N, Bolander J. Voronoi-based interpolants for fracture modelling. Tessellations in the Sciences 2009;.

[28] Leon S, Spring D, Paulino G. Reduction in mesh bias for dynamic fracture using adaptive splitting of polygonal finite elements. International Journal for Numerical Methods in Engineering 2014;100(8):555–576.

[29] Bishop JE. Simulating the pervasive fracture of materials and structures using randomly close packed Voronoi tessellations. Computational Mechanics 2009;44(4):455–471.

[30] Biabanaki S, Khoei A, Wriggers P. Polygonal finite element methods for contact-impact problems on non-conformal meshes. Computer Methods in Applied Mechanics and Engineering 2014;269:198–221.

[31] Biabanaki S, Khoei A. A polygonal finite element method for modeling arbitrary interfaces in large deformation problems. Computational Mechanics 2012;50(1):19–33.

[32] Talischi C, Pereira A, Paulino GH, Menezes IF, Carvalho MS. Polygonal finite elements for incompressible fluid flow. International Journal for Numerical Methods in Fluids 2014;74(2):134–151.

[33] Floater M, Gillette A, Sukumar N. Gradient bounds for Wachspress coordinates on polytopes. SIAM Journal on Numerical Analysis 2014;52(1):515–532.

[34] Joshi P, Meyer M, DeRose T, Green B, Sanocki T. Harmonic coordinates for character articulation. ACM Transactions on Graphics (TOG) 2007;26(3):71.

[35] Martin S, Kaufmann P, Botsch M, Wicke M, Gross M. Polyhedral finite elements using harmonic basis functions. In: Computer Graphics Forum, vol. 27 Wiley Online Library; 2008. p. 1521–1529.

[36] Arroyo M, Ortiz M. Local maximum-entropy approximation schemes: a seamless bridge between finite elements and meshfree methods. International journal for numerical methods in engineering 2006;65(13):2167–2202.

[37] Hormann K, Sukumar N. Maximum entropy coordinates for arbitrary polytopes. In: Computer Graphics Forum, vol. 27 Wiley Online Library; 2008. p. 1513–1520.

[38] Sukumar N. Quadratic maximum-entropy serendipity shape functions for arbitrary planar polygons. Computer Methods in Applied Mechanics and Engineering 2013;263:27–41.

[39] Talischi C, Paulino GH. Addressing integration error for polygonal finite elements through polynomial projections: a patch test connection. Mathematical Models and Methods in Applied Sciences 2014;24(08):1701–1727.

[40] Talischi C, Pereira A, Menezes IF, Paulino GH. Gradient correction for polygonal and polyhedral finite elements. International Journal for Numerical Methods in Engineering 2015;102(3-4):728–747.

[41] Beirão da Veiga L, Brezzi F, Cangiani A, Manzini G, Marini LD, Russo A. Basic principles of Virtual Element Methods. Mathematical Models and Methods in Applied Sciences 2013;23(1):199–214.

[42] Ahmad B, Alsaedi A, Brezzi F, Marini LD, Russo A. Equivalent projectors for virtual element methods. Computers & Mathematics with Applications 2013;66(3):376–391.

[43] Brezzi F, Falk RS, Marini LD. Basic principles of mixed virtual element methods. ESAIM: Mathematical Modelling and Numerical Analysis 2014;48(4):1227–1240.

[44] Lipnikov K, Manzini G, Shashkov M. Mimetic finite difference method. Journal of Computational Physics 2014;257:1163–1227.

[45] Bonelle J, Ern A. Analysis of compatible discrete operator schemes for elliptic problems on polyhedral meshes. ESAIM: Mathematical Modelling and Numerical Analysis 2014;48(2):553–581.

[46] Vacca G, Beirão da Veiga L. Virtual element methods for parabolic problems on polygonal meshes. Numerical Methods for Partial Differential Equations 2015;31(6):2110–2134.

[47] Ayuso de Dios B, Lipnikov K, Manzini G. The nonconforming virtual element method. ESAIM: Mathematical Modelling and Numerical Analysis 2016;50(3):879–904.

[48] Chow P, Cross M, Pericleous K. A natural extension of the conventional finite volume method into polygonal unstructured meshes for CFD application. Applied Mathematical Modelling 1996;20(2):170–183.

[49] Droniou J, Eymard R, Gallouët T, Herbin R. A unified approach to mimetic finite difference, hybrid finite volume and mixed finite volume methods. Mathematical Models and Methods in Applied Sciences 2010;20(02):265–295.

[50] Beirão da Veiga L, Lipnikov K, Manzini G. The mimetic finite difference method for elliptic problems, vol. 11. Springer; 2014.

[51] Brezzi F, Lipnikov K, Simoncini V. A family of mimetic finite difference methods on polygonal and polyhedral meshes. Math Models Methods Appl Sci 2005;15:1533–1553.

[52] Brezzi F, Buffa A, Lipnikov K. Mimetic finite differences for elliptic problems. ESAIM: Mathematical Modelling and Numerical Analysis 2009;43(2):277–295.

[53] Brezzi F, Lipnikov K, Shashkov M, Simoncini V. A new discretization methodology for diffusion problems on generalized polyhedral meshes. Computer Methods in Applied Mechanics and Engineering 2007;196(37-40):3682–3692.

[54] Beirão da Veiga L. A mimetic finite difference method for linear elasticity. M2AN: Math Model Numer Anal 2010;44(2):231–250.

[55] Bochev PB, Hyman JM. Principles of mimetic discretizations of differential operators. In: Compatible spatial discretizations Springer; 2006.p. 89–119.

[56] Beirão da Veiga L, Lipnikov K, Manzini G. Arbitrary-order nodal mimetic discretizations of elliptic problems on polygonal meshes. SIAM Journal on Numerical Analysis 2011;49(5):1737–1760.

[57] Beirão da Veiga L, Brezzi F, Marini LD, Russo A. The Hitchhiker's guide to the Virtual Element Method. Mathematical Models and Methods in Applied Sciences 2014;24(8):1541–1573.

[58] Mengolini M, Benedetto MF, Aragon AM. An engineering perspective to the virtual element method and its interplay with the standard finite element method. Computer Methods in Applied Mechanics and Engineering, 2019;350(6):995–1023, doi =https://doi.org/10.1016/j.cma.2019.02.043,.

[59] Beirão da Veiga L, Brezzi F, Marini L. Virtual Elements for Linear Elasticity Problems. SIAM Journal on Numerical Analysis 2013;51(2):794–812. https://doi.org/10.1137/120874746.

[60] Dassi F, Vacca G. Bricks for the mixed high-order virtual element method: projectors and differential operators;.

[61] Zhang HW, Wu JK, Lü J, Fu ZD. Extended multiscale finite element method for mechanical analysis of heterogeneous materials. Acta Mechanica Sinica/Lixue Xuebao 2010 12;26(6):899–920.

[62] Beirão da Veiga L, Lovadina C, Russo A. Stability analysis for the virtual element method. Mathematical Models and Methods in Applied Sciences 2017;27(13):2557–2594.

[63] Cangiani A, Manzini G, Russo A, Sukumar N. Hourglass stabilization and the virtual element method. International Journal for Numerical Methods in Engineering 2015;102(3-4):404–436.

[64] Dassi F, Mascotto L. Exploring high-order three dimensional virtual elements: bases and stabilizations. Computers & Mathematics with Applications 2018;75(9):3379–3401.

[65] Gain A, Talischi C, H Paulino G. On the Virtual Element Method for Three-Dimensional Elasticity Problems on Arbitrary Polyhedral Meshes. Computer Methods in Applied Mechanics and Engineering 2013 11;282:132–160.

[66] Du Q, Emelianenko M, Ju L. Convergence of the Lloyd algorithm for computing centroidal Voronoi tessellations. SIAM journal on numerical analysis 2006;44(1):102–119.

[67] Talischi C, Paulino GH, Pereira A, Menezes IFM. PolyMesher: a general-purpose mesh generator for polygonal elements written in Matlab. Structural and Multidisciplinary Optimization 2012 Mar;45(3):309–328. https://doi.org/10.1007/s00158-011-0706-z.

[68] Augarde CE, Deeks AJ. The use of Timoshenko's exact solution for a cantilever beam in adaptive analysis. Finite Elements in Analysis and Design 2008 jun;44(9):595 – 601. http://www.sciencedirect.com/science/article/pii/S0168874X08000140.

[69] Beirão da Veiga L, Dassi F, Russo A. High-order virtual element method on polyhedral meshes. Computers & Mathematics with Applications 2017;74(5):1110–1122.

[70] Artioli E, Marfia S, Sacco E. High-order virtual element method for the homogenization of long fiber nonlinear composites. Computer Methods in Applied Mechanics and Engineering 2018;341:571–585.

A | VIRTUAL ELEMENT SPACES

The global virtual space of k^{th} order $\mathcal{V}_h$ is defined as

$$\mathcal{V}_h := \{\mathbf{v}_h \in [\mathcal{H}^1(\Omega) \cap \mathcal{C}^0(\mathcal{T}_h)]^2 : \mathbf{v}_h|_{\mathcal{K}} \in \mathcal{V}_h^{\mathcal{K}}(\mathcal{K}), \forall \mathcal{K} \in \mathcal{T}_h\}, \tag{56}$$

where the local virtual space $\mathcal{V}_h^{\mathcal{K}}$ is defined over an element $\mathcal{K}$ as

$$\mathcal{V}_h^{\mathcal{K}}(\mathcal{K}) := \{\mathbf{v}_h \in [\mathcal{H}^1(\mathcal{K}) \cap \mathcal{C}^0(\mathcal{K})]^2 : \mathbf{v}_{h,i}|_e \in \mathbb{P}_k(e)\, \forall e \in \partial\mathcal{K};\ \Delta\mathbf{v}_{h,i}|_{\mathcal{K}} \in \mathbb{P}_{k-2}(\mathcal{K}) \text{ for } i = 1, 2\}. \tag{57}$$

This space is spanned by a basis implicitly defined through the associated degrees of freedom (DOFs) defined in Eq. (10).

B | MONOMIAL SPACES

The monomial space can be defined iteratively as follows,

$$[\mathbb{M}_0(\mathcal{K}_m)]^2 = \left\{\begin{pmatrix}1\\0\end{pmatrix}, \begin{pmatrix}0\\1\end{pmatrix}\right\}$$

$$[\mathbb{M}_k(\mathcal{K}_m)]^2 = \left\{[\mathbb{M}_{k-1}(\mathcal{K}_m)]^2, \begin{pmatrix}\xi^k\\0\end{pmatrix}, \begin{pmatrix}0\\\xi^k\end{pmatrix}, \begin{pmatrix}\xi^{k-1}\eta\\0\end{pmatrix}, \begin{pmatrix}0\\\xi^{k-1}\eta\end{pmatrix}, \cdots \begin{pmatrix}\eta^k\\0\end{pmatrix}, \begin{pmatrix}0\\\eta^k\end{pmatrix}\right\},$$

where ξ and η denote the scaled scalar monomials $\frac{x-x_{\mathcal{K}_m}}{h_{\mathcal{K}_m}}$ and $\frac{y-y_{\mathcal{K}_m}}{h_{\mathcal{K}_m}}$, respectively. The centroid and diameter of the element are contained in $(x_{\mathcal{K}_m}, y_{\mathcal{K}_m})^T$ and $h_{\mathcal{K}_m}$ respectively. The number of vector valued monomials in $[\mathbb{M}_k(\mathcal{K}_m)]^2$ is generalizable to $n_k = (k+1)(k+2)$. The bases specific to rigid body motion are contained in $\mathbb{K}^{\boldsymbol{\varepsilon}_m}(\mathcal{K}_m)$; two for translations and one for rotation, i.e.,

$$\mathbb{K}^{\boldsymbol{\varepsilon}_m}(\mathcal{K}_m) = \text{span}\left\{\begin{pmatrix}1\\0\end{pmatrix}, \begin{pmatrix}0\\1\end{pmatrix}, \begin{pmatrix}\eta\\-\xi\end{pmatrix}\right\}. \tag{58}$$

C | EVALUATION OF THE VEM MATRICES $\tilde{G}$ AND B

Only the expressions necessary for the derivations provided in the main manuscript are provided herein. For a more extensive discussion [see, e.g. 58]. The weak form defined in Eq. (23) is re-written here for convenience:

$$\sum_{i=1}^{n_k-3} s_i \underbrace{a^{\mathcal{K}_m}(\mathbf{m}_i, \mathbf{m}_j)}_{\tilde{G}_{ij}} = \underbrace{a^{\mathcal{K}_m}(\mathbf{u}_h, \mathbf{m}_j)}_{B_{ij}} \tag{59}$$

This is written in a more explicit form as follows:

$$\sum_{i=1}^{n_k-3} s_i \underbrace{\int_{\mathcal{K}_m} \boldsymbol{\varepsilon}_m(\mathbf{m}_i)^T \, \boldsymbol{\sigma}_m(\mathbf{m}_j) \, d\mathcal{K}}_{\tilde{\mathbf{G}}} = \underbrace{\int_{\mathcal{K}_m} \boldsymbol{\varepsilon}_m(\mathbf{w}_h)^T \, \boldsymbol{\sigma}_m(\mathbf{m}_j) \, d\mathcal{K}}_{\mathbf{B}}. \tag{60}$$

In this, the left hand side term $\tilde{\mathbf{G}}$ can be evaluated in a straightforward manner since it contains only vector valued monomials. The right hand side term is expanded according to Eq. (61) as

$$\underbrace{\int_{\mathcal{K}_m} \boldsymbol{\varepsilon}_m(\mathbf{u}_h)^T \, \boldsymbol{\sigma}_m(\mathbf{m}_j) \, d\mathcal{K}}_{\mathbf{B}} = \underbrace{-\int_{\mathcal{K}_m} \mathbf{u}_h \cdot (\mathrm{div}(\boldsymbol{\sigma}_m(\mathbf{m}_j))) \, d\mathcal{K}}_{\mathbf{B}_i} + \underbrace{\sum_{e \in \partial\mathcal{K}_m} \int_e \mathbf{u}_h \cdot (\boldsymbol{\sigma}_m(\mathbf{m}_j)\mathbf{n}_e) \, de}_{\mathbf{B}_b}. \tag{61}$$

The domain integral $\mathbf{B}_i$ in Eq. (61) vanishes for $k = 1$. When $k \geq 2$, further manipulation of the integral is necessary. The term $\mathrm{div}(\boldsymbol{\sigma}_m(\mathbf{m}_j))$ is expanded out as follows:

$$\mathrm{div}(\boldsymbol{\sigma}_m(\mathbf{m}_j)) = \sum_{\beta=1}^{n_{k-2}} d_{j\beta} \mathbf{m}_\beta, \tag{62}$$

where $\mathbf{m}_\beta \in [\mathbb{M}_{k-2}(\mathcal{K}_m)]^2$. The coefficients of the linear expansion $d_{j\beta}$ directly depend on the material properties and can be obtained through inspection ([see, e.g. 58]). Substituting Eq. (62) into the expression for $\mathbf{B}_i$ in Eq. (61) yields the following form

$$\mathbf{B}_i = -|\mathcal{K}_m| \sum_{\beta=1}^{n_{k-2}} d_{j\beta} \mathrm{dof}_{2kN_v+\beta}(\mathbf{u}_h) = -|\mathcal{K}_m| \mathbf{d}. \tag{63}$$

where dof_i in Eq. (63) is the value of $\mathbf{u}_h$ at the i^{th} degree of freedom. Here, the definition of the *dof* operator at interior nodes (see Eq. (10)) is exploited.

The boundary integral $\mathbf{B}_b$ in Eq. (61) can be evaluated directly using a Gauss-Lobatto quadrature rule according to Eq. (64)

$$\mathbf{B}_b = \sum_{i=1}^{N_v} \sum_{j=1}^{k+1} \frac{1}{2} (\mathbf{N}_e \mathbf{N}^P)^T \mathbf{N}_V(\mathbf{x}_j) l_i w_j, \tag{64}$$

where l_i is the length of the edge under consideration, w_j is the Gauss-Lobatto weight of the boundary node located on that edge, $\mathbf{N}_V(\mathbf{x}_j)$ is the matrix of the canonical basis functions evaluated at the edge node j, $\mathbf{N}^P$ is the matrix of $\mathbb{C}\boldsymbol{\varepsilon}(\mathbf{m}_j)$, and $\mathbf{N}_e$ denotes the outwards looking normal edge vector written in matrix form as

$$\mathbf{N}_e = \begin{bmatrix} n_x & 0 & n_y \\ 0 & n_y & n_x \end{bmatrix}. \tag{65}$$

Hence, matrix $\mathbf{B}$ in Eq. (61) eventually becomes

$$\mathbf{B} = \left[\mathbf{B}_b \ , \ \mathbf{B}_i\right]. \tag{66}$$

and Eq. (60) can be expressed in the convenient matrix representation

$$\tilde{\mathbf{G}}\mathbf{\Pi}_k^{\boldsymbol{\varepsilon}_m} = \mathbf{B}, \tag{67}$$

where $\tilde{\mathbf{G}}$ is the left hand term in Eq. (60) and can be readily computed. For $k = 1$ methods, this can be fairly trivial as the integrand is comprised of constants. For higher order methods, numerical integration over sub-triangulated domains might be necessary. It can also be verified using the relation $\tilde{\mathbf{G}} = \mathbf{BD}$ as shown in [60]. $\mathbf{\Pi}_k^{\boldsymbol{\varepsilon}_m}$ is the vector representation of the projection operator. One solves for this:

$$\mathbf{\Pi}_k^{\boldsymbol{\varepsilon}_m} = \tilde{\mathbf{G}}^{-1}\mathbf{B}. \tag{68}$$

D | CONSISTENCY TERM

The consistency term $\mathbf{K}_C$ shown in Eq. (19) is defined as

$$\mathbf{K}_C = \mathbf{\Pi}_k^{\boldsymbol{\varepsilon}_m\,T}\underbrace{\left[\int_{\mathcal{K}_m} \boldsymbol{\varepsilon}_m(\mathbf{m}_i)^T\,\boldsymbol{\sigma}_m(\mathbf{m}_j)\,d\mathcal{K}\right]}_{\tilde{\mathbf{G}}}\mathbf{\Pi}_k^{\boldsymbol{\varepsilon}_m} \quad i,j = 1,\ldots,n_k. \tag{69}$$

where the matrix $\tilde{\mathbf{G}}$ and $\mathbf{\Pi}_k^{\boldsymbol{\varepsilon}_m}$ are provided in Appendix C and n_k is the number of vector valued monomials defined in Eq. (22).

E | STABILIZATION TERM

To avoid hourglass modes and to ensure the coercivity of the bilinear approximation and positive definiteness of the stiffness matrix in Eq. (19), an additional contribution is required that stems from the so-called stabilisation term. The stability term classically employed in several works was originally proposed in [41, 57]. The associated stability stiffness matrix contribution $\mathbf{K}_S^{(1)}$ is evaluated according to Eq. (70) as

$$\mathbf{K}_S^{(1)} = (\mathbf{I} - \mathbf{\Pi}_{tot}^{\boldsymbol{\varepsilon}_m})^T\,|\mathcal{K}_m|(\mathbf{I} - \mathbf{\Pi}_{tot}^{\boldsymbol{\varepsilon}_m}), \tag{70}$$

where the factor $|\mathcal{K}_m|$ has been included to ensure scaling of the stability energy with the element size. However, it has been reported in [69] that this choice of the stability term may lead to non-optimal error convergence rates in the case of three dimensional higher-order methods.

An alternate strategy, the *D-recipe stabilization*, was proposed in [64]. This has been adapted into an engineering context in [65] and reads as follows:

$$\mathbf{K}_S^{(2)} = (\mathbf{I} - \mathbf{\Pi}_{tot}^{\boldsymbol{\varepsilon}_m})^T\,\beta(\mathbf{I} - \mathbf{\Pi}_{tot}^{\boldsymbol{\varepsilon}_m}), \tag{71}$$

where β is a modified stability parameter, i.e.,

$$\beta = \gamma \beta *, \tag{72}$$

where $\gamma = 1$ and

$$\beta^* = |\mathcal{K}_\mathrm{m}| \frac{\mathrm{tr}(\mathbb{C})}{\mathrm{tr}(\mathbf{D}^T \mathbf{D})}. \tag{73}$$

This choice of the stability parameter is preferable to the one used in Eq. (70). In cases where the energy associated with the consistency term is very high, the stabilisation effects provided by $\mathbf{K}_\mathrm{S}^{(1)}$ tend to be negligible. Conversely, the choice $\mathbf{K}_\mathrm{S}^{(2)}$ ensures correct scaling in relation to the consistency term. In this work, we employ the D-recipe stabilization in Eq. (71).

The projector $\mathbf{\Pi}_\mathrm{tot}^{\boldsymbol{\varepsilon}_\mathrm{m}}$ in Eqs. (70) and (71) is defined as

$$\mathbf{\Pi}_\mathrm{tot}^{\boldsymbol{\varepsilon}_\mathrm{m}} = \mathbf{D}\mathbf{\Pi}_\mathrm{k}^{\boldsymbol{\varepsilon}_\mathrm{m}} + \mathbf{D}_\mathrm{rb}\mathbf{\Pi}_\mathrm{rb}^{\boldsymbol{\varepsilon}_\mathrm{m}}, \tag{74}$$

where the matrix $\mathbf{D}$ contains the monomials evaluated at each DoF and is defined as

$$\mathbf{D} = \begin{bmatrix} \mathrm{dof}_1(\mathbf{m}_1) & \dots & \mathrm{dof}_1(\mathbf{m}_{\mathrm{n_k}-3}) \\ \vdots & \dots & \vdots \\ \mathrm{dof}_\mathrm{n}(\mathbf{m}_1) & \dots & \mathrm{dof}_n(\mathbf{m}_{\mathrm{n_k}-3}) \end{bmatrix}, \tag{75}$$

where $\mathrm{dof_i}(\mathbf{m_j})$, $\mathrm{i} = 1, 2, \dots, 2\mathrm{kN_v}$ denotes the value of the $\mathrm{j^{th}}$ monomial at the $\mathrm{i^{th}}$ DoF. The internal DoFs, i.e., $\mathrm{dof_i}(\mathbf{m_j})$, $\mathrm{i} = 2\mathrm{kN_v} + 1, \dots, \mathrm{n}$ are evaluated according to the second moments definition of the degrees of freedom in Section 2.2, using numerical integration, i.e.,

$$\mathrm{dof}_{2\mathrm{kN_v}+\beta}(\mathbf{m_j}) = \frac{1}{|\mathcal{K}_\mathrm{m}|} \int_{\mathcal{K}_\mathrm{m}} \mathbf{m_j} \cdot \mathbf{m}_\beta \, \mathrm{d}\mathcal{K} \quad \forall \mathbf{m}_\beta \in [\mathbb{M}_{\mathrm{k}-2}(\mathcal{K}_\mathrm{m})]^2. \tag{76}$$

Furthermore, in Eq. (74), $\mathbf{\Pi}_\mathrm{k}^{\boldsymbol{\varepsilon}_\mathrm{m}}$ is the projector computed in Eq. (68) and $\mathbf{\Pi}_\mathrm{rb}^{\boldsymbol{\varepsilon}_\mathrm{m}}$ is defined as

$$\mathbf{\Pi}_\mathrm{rb}^{\boldsymbol{\varepsilon}_\mathrm{m}} = \tilde{\mathbf{G}}_\mathrm{rb}^{-1} \mathbf{B}_\mathrm{rb}, \tag{77}$$

where $\tilde{\mathbf{G}}_\mathrm{rb} = \mathbf{B}_\mathrm{rb}\mathbf{D}_\mathrm{rb}$, and $\mathbf{D}_\mathrm{rb}$ contains the monomials associated with rigid body motion evaluated at each DoF in a similar way to Eq. (75).

Finally, the matrix $\mathbf{B}_\mathrm{rb}$ assumes the following special form

$$\mathbf{B}_\mathrm{rb} = \begin{bmatrix} 1/\mathrm{N_v} & 0 & 1/\mathrm{N_v} & 0 & \dots \\ 0 & 1/\mathrm{N_v} & 0 & 1/\mathrm{N_v} & \dots \\ \eta(x_1) & -\xi(x_1) & \eta(x_2) & -\xi(x_2) & \dots \end{bmatrix}. \tag{78}$$