Computer Methods in Applied Mechanics and Engineering 425 (2024) 116932

Contents lists available at ScienceDirect

Comput. Methods Appl. Mech. Engrg.

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

Solver-free reduced order homogenization for nonlinear periodic heterogeneous media

Andrew Beel *, Jacob Fish

Department of Civil Engineering and Engineering Mechanics Columbia University, United States of America

ARTICLE INFO

Keywords:
Multiscale
Computational homogenization
Reduced order homogenization
Composites
Finite elements

ABSTRACT

Reduced-order homogenization (ROH) and related methods are important computational tools for simulating the material behavior of composites. These methods generally sacrifice accuracy in exchange for superior computational efficiency, relative to methods such as classical computational homogenization (CCH). In this study, building on the recently developed solver-free CCH, we propose a fine-scale solver-free reduced order homogenization approach that avoids solving the fine-scale equilibrium equations and approximates the phase-average eigenstrains by sampling the fine-scale eigenstrain at a small number of points. The proposed method, which we call solver-free ROH, works by pre-computing history-dependent eigenstrain influence function tensors and sampling point contribution factors based on training data from a small set of CCH simulations. Then, during the online stage of the computation, phase-average eigenstrains are computed from sampling point eigenstrains and used to compute homogenized stresses and strains. Focusing on small-deformation problems, this paper formulates and verifies the solver-free ROH approach for nonlinear periodic heterogeneous media. First, in the formulation, we delineate a sampling point approximation of the phase eigenstrains and describe the use of this approximation within the coarse-scale stress update. Next, we verify the solver-free ROH using loading cases outside the training data set. Finally, we use the proposed method to simulate a multilayer composite plate in three point bending (3pt-bend) and open hole tension (OHT), demonstrating the method's efficiency and accuracy relative to the CCH.

1. Introduction

The wide applicability of composite materials has stimulated significant research interest in computational multiscale methods for simulating their behavior. Generally speaking, multiscale methods may employ constitutive models at multiple length and time scales and bridge these models based on mathematical or physical considerations. A review article by Fish, Wagner, and Keten [1] and one by Matouš, Geers, Kouznetsova, and Gillman [2] describe the following significant milestones in multiscale modeling. The Cauchy–Born Rule [3] is an early principle that relates macroscale deformations to displacements of atoms within chemical bonds. The Voigt model [4], which posits constant strain in each representative volume element (RVE), and the Reuss model [5], which assumes constant stress in each RVE, are other early examples. Books by Bakhvalov and Panasenko [6], Sanchez-Palencia [7], Zohdi and Wriggers [8], and Suquet [9] contain descriptions of significant developments in mathematical homogenization. The transformation field analysis (TFA) [10,11] and related reduced order homogenization [12,13] are examples of methods based on mathematical homogenization. The Hill–Mandel macrohomogeneity condition assumes equality between the internal virtual work

* Correspondence to: Department of Civil Engineering and Engineering Mechanics, Columbia University, 500 W 120 St, New York, NY 10027, United States of America.
E-mail address: andrew.beel@columbia.edu (A. Beel).

https://doi.org/10.1016/j.cma.2024.116932
Received 23 December 2023; Received in revised form 13 March 2024; Accepted 15 March 2024
Available online 27 March 2024

computed from fine-scale quantities and that computed from coarse-scale quantities [14]. Other significant developments include statistical mechanics [15], quantum mechanics [16], archetype-blending continuum theory (ABC) [17], generalized finite element methods (GFEM) [18,19], multiscale finite element methods (MsFEM) [20], the heterogeneous multiscale method (HMM) [21,22], the variational multiscale method (VMM) [23,24], multiscale enrichment based on partition of unity (MEPU) [25], generalized space–time mathematical homogenization (GMH) [26–28], multigrid methods [29], domain decomposition [30], global–local finite elements [31], hierarchical modeling of heterogeneous bodies based on adaptive refinement [32], large hierarchically parallelized multiscale simulation [33], and homogenization-based multigrid [34–38]. More recent exciting developments include mean-field homogenization [39,40], surrogate modeling of the microscale based on deep learning [41,42], an adaptive wavelet method for dynamic high-strain damage problems [43], a self-consistent approach for high strain rate loading of composites [44], a wavelet-enhanced fast Fourier transform-based approach [45], a ROH approach for polycrystalline microstructures with cracks [46], and data-physics-driven reduced order homogenization [47].

As described in greater detail in [1], the two broad categories of multiscale methods are resolved-scale methods and upscaling methods. In resolved-scale methods, a coarse-scale model and a model that resolves fine-scale fields are used in different regions within the domain. Interface or interscale transfer operators are employed to bridge the regional solutions. Examples of resolved-scale methods include domain decomposition and multigrid methods. Upscaling methods, in contrast, homogenize fine-scale fields to compute coarse-scale quantities. Subcategories of upscaling methods include physics-based, data-driven, and mathematics-based methods. The reader may refer to [1] for more details about resolved-scale approaches as well as physics-based and data-driven upscaling approaches.

The method proposed in this study is closely related to mathematics-based methods for upscaling of continua and associated reduced-order methods. Some math-based methods assume separation of scales, such as the asymptotic approaches [6,7,9] and the approach based on the Hill–Mandel macrohomogeneity condition [14]. Other methods account for interscale transitions directly; these methods include multiresolution analysis [48–51] based on wavelets and others (FE^2 [52], MsFEM [20], HMM [21,22], VMM [23,24], and MEPU [25]) based on finite element approximations. These math-based methods and associated reduced-order methods have been applied in the context of metamaterials [53,54], acoustics [53], composite materials [9,12,13,26,27,46,55,56], multiphysics problems [57,58], stochastic methods [59,60], and wave propagation [61].

Classical computational homogenization (CCH) is a math-based upscaling method, in which a system of nonlinear governing equations is solved numerically at the coarse scale and the stress at each Gauss point is determined by solving a separate nonlinear system of equations at the unit cell level [62,63]. Though significantly more efficient than direct numerical simulation, CCH is therefore still impractical for many problems. Many reduced-order methods have been devised to reduce the computational cost of homogenization. Many of these are listed and described in [1], including the Voronoi cell method [64,65], the spectral method [66], the network approximation method [67], methods based on fast Fourier transforms (FFT) [68], the mesh-free reproducing kernel particle method [69], finite-volume direct-averaging micromechanics [70], uniform and nonuniform transformation field analysis [10,11,71–73], methods of cells [74], control theory-based methods such as balanced truncation [75,76], optimal Hankel norm approximation [77], proper orthogonal decomposition [78–80], data-driven reduced-order methods [81–84], reduced order homogenization (ROH) [12,13], and recently data-physics-driven ROH [47]. The FFT, in combination with the Lippmann–Schwinger reformulation of the unit cell problem using an equivalent volume integral equation [68,85] is a particularly exciting recent development.

While these reduced-order methods achieve a significant computational cost savings, they often suffer a great loss of accuracy. For example, the authors of [86] observed inaccurate prediction of the beginning of eigenstrain accumulation in the unit cell and overly stiff behavior of the unit cell in certain loading modes (inclusion locking). Even re-calibration of constituent phase material parameters can fail to make reduced-order approaches reliable for predicting material behavior of composites. More details of reduced-order homogenization's advantages and shortcomings are described in Section 2.1.

This study presents the fine-scale solver-free reduced order homogenization (solver-free ROH hereafter). The proposed approach uses pre-computed eigenstrain influence function tensors that depend on the unit cell material history and are computed based on CCH training simulations, built on the framework of the recent solver-free CCH [87]. The key novel idea of the solver-free ROH that distinguishes it from solver-free CCH is the following: whereas the solver-free CCH constructs the fine-scale eigenstrain field by performing stress updates at every unit cell Gauss point, the solver-free ROH approximates the phase-average eigenstrains by a linear combination of eigenstrains evaluated at a number of sampling points. The weights of the sampling point eigenstrains, as well as the locations of the sampling points themselves, are determined by optimization techniques to be consistent with the same training data that is used to compute the influence function tensors. By considerably reducing the number of unit cell stress updates that need to be performed, the solver-free ROH considerably reduces the computational cost relative to the CCH and to the solver-free CCH.

The sampling point approximation of the phase-average eigenstrain proposed in this study is related to a number of efficient quadrature schemes found in the literature. The most common method of numerical integration over complex geometries is tessellation by simple volume cells such as triangles, quadrilaterals, tetrahedra, or hexahedra, followed by application of well-established Gauss quadrature rules within in each volume cell. Alternatives to this common approach include non-conforming meshing and adaptive quadrature as in [19], Monte Carlo integration [88], integration over general convex polyhedra via mean-value interpolation functions [89], and conversion of volume integrals to contour integrals for self-equilibriating enrichment functions [90], among others. A particularly notable class of efficient quadrature schemes is based on moment fitting equations, including the generalized Gauss quadrature rules developed in [91] for polygons and [92] for polyhedra. A related method, developed by Sudhakar and Wall [93], extended the moment-fitting approach to non-convex polytopes using repeated application

of the Divergence Theorem rather than Laserre's method. These quadrature schemes achieve high accuracy with a reduced number of integration points.

The proposed sampling point approximation is distinguished from these other efficient quadrature schemes by the dependence of the quadrature weights on the unit cell material history and the determination of integration point locations and weights based on observations of the variable being integrated (fine-scale eigenstrain) within a set of training simulations. Typically, quadrature schemes seek a set of point locations and weights such that a set of basis functions is integrated exactly (or very accurately) over a given class of polytope, and are therefore applicable to all domains within this class and all integrands that can be well-approximated by these basis functions. In contrast, while the sampling point procedure proposed in this study is applicable to a variety of integrands and domain geometries, each quadrature rule resulting from this procedure is tailored to the material system and geometry of the unit cell to which it was applied. This allows the number of sampling points to be remarkably small relative to the number of unit cell Gauss points, resulting in marked speedup of the proposed integration scheme relative to full Gauss integration of the fine-scale eigenstrain.

The remainder of this paper is organized as follows: Section 2 begins with a review of relevant ideas from CCH and ROH, followed by a review of the solver-free CCH eigenstrain influence tensors and their proper use during the online stage of the computation, culminating in a detailed formulation of the sampling point approximation of the phase eigenstrains that distinguishes the proposed method from its precursors. Section 3 presents numerical results from four studies: (i) reproduction of the training data following computation of the eigenstrain influence tensors and sampling point contribution factors, (ii) verification by subjecting a single coarse-scale element to non-monotonic mixed tensile and shear loading, and (iii) demonstration of combined accuracy and efficiency of the solver-free ROH compared to CCH for three-point bending and open-hole tension of a multilayer composite plate. Finally, Section 4 summarizes the findings of this study and suggests future research directions.

2. Formulation of the solver-free ROH approach

In the context of nonlinear periodic heterogeneous media, the computational cost of the CCH becomes prohibitive as the number of degrees of freedom in the coarse-scale and fine-scale problems increases. The need to (i) solve the fine-scale equilibrium equations and (ii) perform fine-scale stress updates at every unit cell Gauss point share much of the responsibility for this high computational cost. Building on the solver-free CCH approach described in [87], the solver-free ROH formulated in this section eliminates both of these expensive features of the unit cell computation. In this scheme, the fine-scale strain is expressed in terms of the influence of coarse-scale strain and homogenized eigenstrain, while the phase-average eigenstrain is approximated by a linear combination of samples of the fine-scale eigenstrain field. Section 2.1 briefly reviews CCH and ROH, laying the groundwork for the solver-free ROH approach proposed in this study. Section 2.2 reviews the procedure from [87] for pre-computing snapshots of the solver-free CCH eigenstrain influence tensors, which are used in the proposed method. Section 2.3 formulates the sampling point approximation of the phase eigenstrains and describes how the sampling points are selected. Section 2.4 details how the eigenstrain influence tensors and sampling points are used in the online stage of the computation.

2.1. Review of CCH and ROH

This section contains a review of CCH and ROH adapted from [87] as well as Chapters 2 and 4 of [94]. The following scale-separated small-deformation governing equations for the coarse scale domain Ω and the unit cell Θ, written in terms of coarse- and fine-scale variables, are typical of a CCH scheme. A superscript c is used to denote "coarse-scale" quantities and a superscript f is used to denote "fine-scale" quantities. The coarse-scale governing equations are given by

$$\sigma^{\mathrm{c}}_{ij,j}(\mathbf{x}) + b_i(\mathbf{x}) = 0 \qquad \forall \mathbf{x} \in \Omega \tag{2.1a}$$

$$u^{\mathrm{c}}_i(\mathbf{x}) = \bar{u}_i(\mathbf{x}) \quad \forall \mathbf{x} \in \partial\Omega_u \tag{2.1b}$$

$$\sigma^{\mathrm{c}}_{ij}(\mathbf{x}) n^{\mathrm{c}}_j(\mathbf{x}) = \bar{t}_i(\mathbf{x}) \quad \forall \mathbf{x} \in \partial\Omega_t \tag{2.1c}$$

Here, $\boldsymbol{\sigma}^{\mathrm{c}}$ is the coarse-scale stress, $\mathbf{u}^{\mathrm{c}}$ is the coarse-scale displacement, $\mathbf{b}$ is the body force, $\mathbf{n}^{\mathrm{c}}$ is the normal to the coarse-scale boundary, $\bar{\mathbf{u}}$ is a prescribed displacement, and $\bar{\mathbf{t}}$ is a prescribed traction. The fine-scale equations governing a unit cell at coarse-scale point $\mathbf{x}$ are given by

$$\sigma^{\mathrm{f}}_{ij,y_j}(\mathbf{x},\mathbf{y}) = 0 \qquad \forall \mathbf{y} \in \Theta \tag{2.2a}$$

$$\sigma^{\mathrm{f}}_{ij}(\mathbf{x},\mathbf{y}) = L_{ijkl}(\mathbf{y}) \left(\epsilon^{\mathrm{f}}_{kl}(\mathbf{x},\mathbf{y}) - \mu^{\mathrm{f}}_{kl}(\mathbf{x},\mathbf{y}) \right) \quad \forall \mathbf{y} \in \Theta \tag{2.2b}$$

$$\epsilon^{\mathrm{f}}_{ij}(\mathbf{x},\mathbf{y}) = \epsilon^{\mathrm{c}}_{ij}(\mathbf{x}) + u^{\mathrm{f}}_{(i,y_j)}(\mathbf{x},\mathbf{y}) \qquad \forall \mathbf{y} \in \Theta \tag{2.2c}$$

$$u^{\mathrm{f}}_i(\mathbf{x},\mathbf{y}) \quad \text{periodic on } \Theta \tag{2.2d}$$

Here, $\boldsymbol{\sigma}^{\mathrm{f}}$ is fine-scale stress, $\boldsymbol{\epsilon}^{\mathrm{f}}$ is the fine-scale strain, $\boldsymbol{\mu}^{\mathrm{f}}$ is the fine-scale eigenstrain, $\boldsymbol{\epsilon}^{\mathrm{c}}$ is the coarse-scale strain, and $\mathbf{u}^{\mathrm{f}}$ is the fine-scale displacement correction. The fourth-order tensor $\mathbf{L}$ is the elastic constitutive tensor, which is assumed to depend only on the fine-scale coordinate $\mathbf{y}$. The coarse-scale and fine-scale stress are related via

$$\sigma^{\mathrm{c}}_{ij}(\mathbf{x}) = \frac{1}{|\Theta|} \int_{\Theta} \sigma^{\mathrm{f}}_{ij}(\mathbf{x},\mathbf{y}) \, \mathrm{d}\Theta \tag{2.3}$$

For a composite material with N_{ph} homogeneous individual material phases denoted $\Theta^{(1)}, \Theta^{(2)}, \dots, \Theta^{(N_{\text{ph}})}$, the fine-scale eigenstrain at $(\mathbf{x}, \mathbf{y})$ is found using the constitutive law governing the phase to which $\mathbf{y}$ belongs. For each $\beta = 1, 2, \dots, N_{\text{ph}}$, let $\mathcal{G}^{(\beta)}$ be the constitutive law governing material phase β within every unit cell, which in general takes strain, stress, and a state variable array $\mathbf{s}^{\text{f}}$ as inputs and gives eigenstrain as output. Then

$$\mu_{ij}^{\text{f}}(\mathbf{x}, \mathbf{y}) = \begin{cases} \mathcal{G}_{ij}^{(1)}\left[\boldsymbol{\epsilon}^{\text{f}}(\mathbf{x}, \mathbf{y}), \boldsymbol{\sigma}^{\text{f}}(\mathbf{x}, \mathbf{y}), \mathbf{s}^{\text{f}}(\mathbf{x}, \mathbf{y})\right] & \text{if } \mathbf{y} \in \Theta^{(1)} \\ \mathcal{G}_{ij}^{(2)}\left[\boldsymbol{\epsilon}^{\text{f}}(\mathbf{x}, \mathbf{y}), \boldsymbol{\sigma}^{\text{f}}(\mathbf{x}, \mathbf{y}), \mathbf{s}^{\text{f}}(\mathbf{x}, \mathbf{y})\right] & \text{if } \mathbf{y} \in \Theta^{(2)} \\ \vdots & \\ \mathcal{G}_{ij}^{(N_{\text{ph}})}\left[\boldsymbol{\epsilon}^{\text{f}}(\mathbf{x}, \mathbf{y}), \boldsymbol{\sigma}^{\text{f}}(\mathbf{x}, \mathbf{y}), \mathbf{s}^{\text{f}}(\mathbf{x}, \mathbf{y})\right] & \text{if } \mathbf{y} \in \Theta^{(N_{\text{ph}})} \end{cases} \tag{2.4}$$

Typically, the Galerkin form of the coarse-scale equations (2.1) is solved iteratively in a Newton–Raphson (NR) loop using the finite element method. The fine-scale solver likewise solves the Galerkin form of the fine-scale equations (2.2) in an inner Newton–Raphson loop, with the current iteration of the coarse-scale strain from the coarse-scale solver as input. At each coarse-scale solver iteration, fine-scale stress obtained from the fine-scale solver (2.3) is homogenized to give the coarse-scale stress at each Gauss point. At each coarse-scale Gauss point, for each coarse-scale NR iteration, and for each fine-scale NR iteration, two expensive operations must be performed:

1. One of the phase constitutive equations (2.4) is evaluated at every fine-scale Gauss point.
2. A new fine-scale tangent stiffness matrix is constructed and factorized.

If many finite elements are required to resolve the unit cell geometry, it is easy to see that the CCH procedure can be extremely computationally intensive.

Reduced-order computational homogenization methods, such as reduced order homogenization (ROH), circumvent these costs by assuming a reduced basis for the fine-scale displacement correction field that automatically satisfies (2.2). The displacement correction is assumed to take the form

$$u_i^{\text{f}}(\mathbf{x}, \mathbf{y}) = H_i^{kl}(\mathbf{y})\epsilon_{kl}^{\text{c}}(\mathbf{x}) + \int_{\Theta} \tilde{h}_i^{kl}(\mathbf{y}, \tilde{\mathbf{y}})\mu_{kl}^{\text{f}}(\mathbf{x}, \tilde{\mathbf{y}})\, \mathrm{d}\tilde{\Theta} \tag{2.5}$$

and, from (2.2c), the fine-scale strain is correspondingly given by

$$\begin{aligned} \epsilon_{ij}^{\text{f}}(\mathbf{x}, \mathbf{y}) &= E_{ij}^{kl}(\mathbf{y})\epsilon_{kl}^{\text{c}}(\mathbf{x}) + \int_{\Theta} \tilde{h}_{(i,j)}^{kl}(\mathbf{y}, \tilde{\mathbf{y}})\mu_{kl}^{\text{f}}(\mathbf{x}, \tilde{\mathbf{y}})\, \mathrm{d}\tilde{\Theta} \\ E_{ij}^{kl}(\mathbf{y}) &= I_{ijkl} + H_{(i,j)}^{kl}(\mathbf{y}) \end{aligned} \tag{2.6}$$

Next, influence functions $\mathbf{H}$ and $\tilde{\mathbf{h}}$ are pre-computed in the offline stage so as to satisfy (2.2) for arbitrary coarse-scale strain and fine-scale eigenstrain distributions. Taking $\boldsymbol{\mu}^{\text{f}} = 0$ yields the $\mathbf{H}$ tensor, since the fine-scale equilibrium equations become

$$\begin{aligned} &\left[L_{ijkl}\left[I_{klmn} + H_{(k,l)}^{mn}(\mathbf{y})\right]\right]_{,j} = 0 \\ &\mathbf{H}^{mn}(\mathbf{y}) \quad \text{periodic on } \Theta \end{aligned} \tag{2.7}$$

It is less straightforward to characterize $\tilde{\mathbf{h}}$ for an arbitrary fine-scale eigenstrain distribution. In the literature, the unit cell is frequently divided into disjoint partitions $\Theta^{(\kappa)}, \kappa = 1, 2, \dots, N_{\text{part}}$, each consisting of one or more finite elements. It is then assumed that the eigenstrain may be decomposed as

$$\mu_{ij}^{\text{f}}(\mathbf{x}, \mathbf{y}) = \sum_{\kappa=1}^{N_{\text{part}}} N_{ij}^{kl(\kappa)}(\mathbf{y})\hat{\mu}_{kl}^{(\kappa)}(\mathbf{x}) \tag{2.8}$$

The eigenstrain need only be C^{-1} continuous on Θ, so the simplest choice for eigenstrain shape functions $\mathbf{N}$ is

$$N_{ij}^{kl(\kappa)}(\mathbf{y}) = I_{ijkl}^{(\kappa)}(\mathbf{y}) = \begin{cases} I_{ijkl} & \text{if } \mathbf{y} \in \Theta^{(\kappa)} \\ 0 & \text{otherwise} \end{cases} \tag{2.9}$$

For such a piecewise-uniform eigenstrain distribution, partition eigenstrain influence functions are naturally defined by

$$\begin{aligned} \tilde{h}_i^{kl(\kappa)}(\mathbf{y}) &= \int_{\Theta^{(\kappa)}} \tilde{h}_i^{kl}(\mathbf{y}, \tilde{\mathbf{y}})\, \mathrm{d}\tilde{\Theta} \\ \tilde{P}_{ij}^{kl(\kappa)}(\mathbf{y}) &= \tilde{h}_{(i,j)}^{kl(\kappa)}(\mathbf{y}) \end{aligned} \tag{2.10}$$

Substituting (2.9) and (2.8) into (2.2c) and choosing $\boldsymbol{\epsilon}^{\text{c}} = 0$ yields

$$\begin{aligned} &\left[L_{ijkl}\left[\tilde{h}_{(k,l)}^{mn(\kappa)}(\mathbf{y}) - I_{klmn}^{(\kappa)}\right]\right]_{,j} = 0 \\ &\tilde{\mathbf{h}}^{mn(\kappa)}(\mathbf{y}) \quad \text{periodic on } \Theta \end{aligned} \tag{2.11}$$

The one-partition-per-element approach that appears in the literature [86,95] is one extreme example of this piecewise-uniform eigenstrain framework. This approach preserves much of the accuracy of the CCH but is nearly as computationally intensive: The fine-scale eigenstrain must be computed in each unit cell element, the fine-scale strain at a single point in the unit cell must be

assembled from the influence of each element's eigenstrain on that point, and this process must be performed iteratively because the updated strain field yields an updated eigenstrain field from the nonlinear constitutive equations.

The opposite extreme case of the piecewise-uniform framework is the one-partition-per-material-phase approach. This approach lends itself to a reduced-order unit cell computation in the online stage, wherein the full CCH unit cell equations (2.2) are replaced by analogous reduced-order equations with greatly reduced dimension. Reduced-order methods, while computationally inexpensive, are plagued by inaccuracies. The main sources of these inaccuracies relative to the CCH are

1. Failure to account for non-uniform strain and eigenstrain in the unit cell
2. Failure to account for variation of the shape of the fine-scale eigenstrain distribution with unit cell material history
3. Erroneous computation of phase-average eigenstrains based on evaluation of constitutive models at phase-average strain, stress, and state variables

Extensive calibration of phase material parameters is required to preserve some accuracy of the ROH compared to the CCH. However, the calibrated material parameters of the composite are often unphysical or differ enormously from values provided by the manufacturer. The following are other examples of related reduced-order approaches that seek to preserve accuracy while reducing computational cost:

1. Nonuniform eigenstrain shape functions [96]
2. Hybrid compatible–incompatible eigenstrains [86]
3. Adaptive partitioning [12]
4. Tangent-based updating of influence functions [97]
5. Data-physics-driven reduced order homogenization [47]

2.2. Pre-computed eigenstrain influence tensors

Recently, a solver-free CCH approach has been proposed, which pre-computes eigenstrain influence tensors that are assumed to be dependent on the unit cell material history [87]. These tensors are utilized during the unit cell computation in such a way as to eliminate the need to solve the unit cell equilibrium equations. The solver-free CCH thereby proves less computationally intensive than CCH, while retaining considerable accuracy. The procedure for pre-computing the eigenstrain influence tensors of the solver-free CCH, first described in [87] by the authors of this manuscript, lays the groundwork for the solver-free ROH method. This section is mostly reproduced from this source. To formulate the eigenstrain influence tensors, we begin with the expression for fine-scale displacement correction (2.5) used as the basis for the ROH. The influence tensors $\mathbf{H}$ and $\mathbf{E}$ associated with the coarse-scale strain remain the same as for ROH, but a different approach is taken to decompose the fine-scale eigenstrain field.

Observe the following about the nature of an increment of the fine-scale eigenstrain field. As illustrated by (2.4), two different initial strain fields, two different strain field increments, two different initial stress fields, two different stress field increments, or two different material histories will, generally speaking, all result in two different distributions of the incremental fine-scale eigenstrain field. It is no wonder, then, that assumptions such as "uniform eigenstrain in each material phase" would inadequately capture the fine-scale constitutive behavior. On the other hand, some simplifying assumptions must be made about the eigenstrain distribution in order to reduce the computational complexity of the CCH.

Rather than assuming uniform eigenstrain in each material phase, we make the following assumptions about the fine-scale eigenstrain increment:

1. The increment of each component mn of the fine-scale eigenstrain field has a shape dependent only on the unit cell material history.
2. This material history dependence is only through a scalar variable $|{}_0\boldsymbol{\mu}|$ representing the norm of the homogenized unit cell eigenstrain accumulated prior to the current load increment. This variable is defined by

$$
\begin{aligned}
|{}_0\boldsymbol{\mu}|(\mathbf{x}) &= \sqrt{\sum_{mn} \left({}_0\mu^{\mathrm{c}}_{mn}(\mathbf{x})\right)^2} \\
{}_0\mu^{\mathrm{c}}_{ij}(\mathbf{x}) &= {}_0\epsilon^{\mathrm{c}}_{ij}(\mathbf{x}) - C^{\mathrm{c}}_{ijkl}(\mathbf{x})\ {}_0\sigma^{\mathrm{c}}_{kl}(\mathbf{x})
\end{aligned}
\tag{2.12}
$$

   Here, ${}_0\boldsymbol{\epsilon}^{\mathrm{c}}$ is the coarse-scale strain from the previous increment, ${}_0\boldsymbol{\sigma}^{\mathrm{c}}$ is the coarse-scale stress from the previous increment, and $\mathbf{C}^{\mathrm{c}}$ is the homogenized elastic compliance tensor.

Under these assumptions, the eigenstrain may be decomposed as

$$
\Delta\mu^{\mathrm{f}}_{mn}(\mathbf{x},\mathbf{y}) = \mathcal{N}^{\underline{mn}}\left(|{}_0\boldsymbol{\mu}|(\mathbf{x}),\mathbf{y}\right)\Delta\mu^{\mathrm{c}}_{\underline{mn}}(\mathbf{x}) \tag{2.13}
$$

Here, the underlined indices indicate no summation. Inserting (2.13) into the fine-scale strain yields

$$
\begin{aligned}
\Delta\epsilon^{\mathrm{f}}_{kl}(\mathbf{x},\mathbf{y}) &= E^{mn}_{kl}(\mathbf{y})\Delta\epsilon^{\mathrm{c}}_{mn}(\mathbf{x}) + P^{mn}_{kl}(|{}_0\boldsymbol{\mu}|(\mathbf{x}),\mathbf{y})\Delta\mu^{\mathrm{c}}_{mn}(\mathbf{x}) \\
P^{mn}_{kl}(|{}_0\boldsymbol{\mu}|(\mathbf{x}),\mathbf{y}) &\equiv \int_{\Theta} \tilde{h}^{\underline{mn}}_{(k,y_l)}(\mathbf{y},\tilde{\mathbf{y}})\mathcal{N}^{\underline{mn}}(\mathbf{x},\tilde{\mathbf{y}})\ \mathrm{d}\tilde{\Theta}
\end{aligned}
\tag{2.14}
$$

We emphasize here that the $\mathbf{P}(|_0\boldsymbol{\mu}|(\mathbf{x}), \mathbf{y})$ tensor fields described here represent the influence of the *homogenized* eigenstrain on the fine-scale strain at each point in the unit cell. This is in contrast to the one-partition-per-element approach described in Section 2.1, in which *element* influence tensor fields $\mathbf{P}^{(I)}(\mathbf{y})$ must be constructed for all unit cell elements $I = 1, \dots, N_{\text{el}}$. The next order of business is to identify suitable components of $\mathbf{P}$ for a given array of $|_0\boldsymbol{\mu}|$. Because we wish to evaluate the fine-scale strain at each unit cell Gauss point $\mathbf{y}^{(\kappa)}, \kappa = 1, 2, \dots, N_{\text{gp}}$, we will pre-compute each eigenstrain influence tensor at each $\mathbf{y}^{(\kappa)}$. Roughly speaking, this will be accomplished by conducting a small number of inexpensive uni-directional load tests simulated by CCH, taking snapshots of the resulting data, and setting the fine-scale strain increment from each of these snapshots equal to our influence function-based expression for the fine-scale strain (2.14). Suppose we have obtained the fine-scale strain and eigenstrain over the course of six CCH simulations $mn = 11, 22, 33, 23, 13, 12$, each corresponding to incremental loading of a single macroscale element that produces a stress primarily in direction mn. As long as eigenstrain has been allowed to develop in each of the CCH tests, then the data in each test will exhibit a range of the variable $|_0\boldsymbol{\mu}|$. Choose a list of representative values $|_0\boldsymbol{\mu}|_J, J = 1, 2, \dots, N_{\text{snap}}$ (snapshots) from the intersection of these ranges. Let $\Delta\epsilon^{\text{c},mn}_{Jkl}$ and $\Delta\mu^{\text{c},mn}_{Jkl}$ represent the kl component of the coarse-scale strain and eigenstrain increment, respectively, from CCH test mn at snapshot J, for which $|_0\boldsymbol{\mu}| = |_0\boldsymbol{\mu}|_J$. Likewise, let $\Delta\epsilon^{(\kappa)mn}_{Jkl}$ represent the kl component of the fine-scale strain increment at Gauss point $\mathbf{y}^{(\kappa)}$ from CCH test mn at snapshot J. If, as we assume above, the fine-scale eigenstrain increment at $\mathbf{y}^{(\kappa)}$ depends only on $|_0\boldsymbol{\mu}|$, it is also true that the $\mathbf{P}$ tensor at $\mathbf{y}^{(\kappa)}$ depends only on $|_0\boldsymbol{\mu}|$. Thus, for each snapshot J and Gauss point $\mathbf{y}^{(\kappa)}$, (2.6) leads to the simultaneous equations

$$\begin{cases} E^{ab}_{kl}(\mathbf{y}^{(\kappa)})\Delta\epsilon^{\text{c},11}_{Jab} + P^{ab}_{kl}(|_0\boldsymbol{\mu}|_J, \mathbf{y}^{(\kappa)})\Delta\mu^{\text{c},11}_{Jab} = \Delta\epsilon^{(\kappa)11}_{Jkl} \\ E^{ab}_{kl}(\mathbf{y}^{(\kappa)})\Delta\epsilon^{\text{c},22}_{Jab} + P^{ab}_{kl}(|_0\boldsymbol{\mu}|_J, \mathbf{y}^{(\kappa)})\Delta\mu^{\text{c},22}_{Jab} = \Delta\epsilon^{(\kappa)22}_{Jkl} \\ \vdots \\ E^{ab}_{kl}(\mathbf{y}^{(\kappa)})\Delta\epsilon^{\text{c},12}_{Jab} + P^{ab}_{kl}(|_0\boldsymbol{\mu}|_J, \mathbf{y}^{(\kappa)})\Delta\mu^{\text{c},12}_{Jab} = \Delta\epsilon^{(\kappa)12}_{Jkl} \end{cases} \tag{2.15}$$

In matrix form, the system for a given snapshot can be written as follows (no sum over J):

$$\mathbf{M}_J\,\mathbf{p}_J(\mathbf{y}^{(\kappa)}) = \mathbf{e}_J(\mathbf{y}^{(\kappa)}) \tag{2.16}$$

Here, the matrix components are given by

$$\mathbf{M}_J = \begin{bmatrix} \Delta\mu^{\text{c},11}_{J11} & \Delta\mu^{\text{c},11}_{J22} & \dots & \Delta\mu^{\text{c},11}_{J12} \\ \Delta\mu^{\text{c},22}_{J11} & \Delta\mu^{\text{c},22}_{J22} & \dots & \Delta\mu^{\text{c},22}_{J12} \\ \vdots & \vdots & \ddots & \vdots \\ \Delta\mu^{\text{c},12}_{J11} & \Delta\mu^{\text{c},12}_{J22} & \dots & \Delta\mu^{\text{c},12}_{J12} \end{bmatrix} \tag{2.17}$$

$$\mathbf{p}_J(\mathbf{y}^{(\kappa)}) = \begin{bmatrix} P^{11}_{11}(|_0\boldsymbol{\mu}|_J, \mathbf{y}^{(\kappa)}) & P^{11}_{22}(|_0\boldsymbol{\mu}|_J, \mathbf{y}^{(\kappa)}) & \dots & P^{11}_{12}(|_0\boldsymbol{\mu}|_J, \mathbf{y}^{(\kappa)}) \\ P^{22}_{11}(|_0\boldsymbol{\mu}|_J, \mathbf{y}^{(\kappa)}) & P^{22}_{22}(|_0\boldsymbol{\mu}|_J, \mathbf{y}^{(\kappa)}) & \dots & P^{22}_{12}(|_0\boldsymbol{\mu}|_J, \mathbf{y}^{(\kappa)}) \\ \vdots & \vdots & \ddots & \vdots \\ P^{12}_{11}(|_0\boldsymbol{\mu}|_J, \mathbf{y}^{(\kappa)}) & P^{12}_{22}(|_0\boldsymbol{\mu}|_J, \mathbf{y}^{(\kappa)}) & \dots & P^{12}_{12}(|_0\boldsymbol{\mu}|_J, \mathbf{y}^{(\kappa)}) \end{bmatrix} \tag{2.18}$$

$$\mathbf{e}_J(\mathbf{y}^{(\kappa)}) = \begin{bmatrix} \Delta\epsilon^{(\kappa)11}_{J11} - E^{ab}_{11}(\mathbf{y}^{(\kappa)})\Delta\epsilon^{\text{c},11}_{Jab} & \dots & \Delta\epsilon^{(\kappa)11}_{J12} - E^{ab}_{12}(\mathbf{y}^{(\kappa)})\Delta\epsilon^{\text{c},11}_{Jab} \\ \vdots & \ddots & \vdots \\ \Delta\epsilon^{(\kappa)12}_{J11} - E^{ab}_{11}(\mathbf{y}^{(\kappa)})\Delta\epsilon^{\text{c},12}_{Jab} & \dots & \Delta\epsilon^{(\kappa)12}_{J12} - E^{ab}_{12}(\mathbf{y}^{(\kappa)})\Delta\epsilon^{\text{c},12}_{Jab} \end{bmatrix} \tag{2.19}$$

To summarize, a system of equations of the form (2.16) is solved for each snapshot J to determine the components of the $\mathbf{P}$ tensor for that snapshot at a given unit cell Gauss point $\mathbf{y}^{(\kappa)}$. In this system, the matrices $\mathbf{M}_J$, $\mathbf{p}_J(\mathbf{y}^{(\kappa)})$, and $\mathbf{e}_J(\mathbf{y}^{(\kappa)})$ each have dimensions 6×6. Note that the matrix $\mathbf{M}_J$ is the same for all Gauss points. Thus, it needs to be factorized only once; the components of $\mathbf{P}$ at each Gauss point are then found by back-substitution.

Remark: It is possible for matrix $\mathbf{M}_J$ to be singular due to relationships between components of the homogenized eigenstrain imposed by the phase material properties and the unit cell geometry. In practice, we have found $\mathbf{M}_J$ to be nonsingular. However, if the matrix were found to be singular, the special techniques described in the Appendix of [87] could be used to find a minimum-norm least squares solution to (2.16).

2.3. Sampling point approximation of phase eigenstrains

The primary limitation of the solver-free CCH is the need to perform stress updates at every unit cell Gauss point. The solver-free ROH method proposed in this study seeks to reduce the number of fine-scale stress updates by reconstructing phase average eigenstrain increments from a linear combination of a small number of sampling point eigenstrain increments.

To develop our sampling point approximation, we begin by choosing a number of sampling points $N^{(\beta)}_{\text{sampl}}$ in phase β. Let $\{\mathbf{y}^{(\beta)}_k\} : k = 1, \dots, N^{(\beta)}_{\text{sampl}}$ be a set of sampling Gauss points within phase β and let $\boldsymbol{\kappa}^{(\beta)}$ denote the vector of indices of these Gauss points. We approximate the phase-average eigenstrain $\Delta\boldsymbol{\mu}^{(\beta)}$ by

$$\Delta\mu^{(\beta)}_{ij}(\mathbf{x}) = \sum_{k=1}^{N^{(\beta)}_{\text{sampl}}} \xi^{(\beta)}_k(|_0\boldsymbol{\mu}|(\mathbf{x}))\ \Delta\mu^{\text{f}}_{ij}(\mathbf{y}^{(\beta)}_k) \tag{2.20}$$

The coefficients $\xi_k^{(\beta)}$ multiplying the sampling point eigenstrain increments, which we call the sampling point contribution factors, are assumed like the influence tensors to depend on the unit cell material history only through the previous-increment eigenstrain norm. Also like the $\mathbf{P}$ tensors, the contribution factors will be interpolated during the online stage of the computation from a set of pre-computed snapshots. Each sampling point eigenstrain increment $\Delta\boldsymbol{\mu}_{ij}^{\mathrm{f}}(\mathbf{y}_k^{(\beta)})$ is computed by stress update using the constitutive model corresponding to the material phase containing the sampling point. It remains to specify (i) how the sampling points should be selected and (ii) how the contribution factors should be computed for a given set of sampling points.

2.3.1. Choosing sampling points

A natural choice for the set of sampling points would be that which minimizes the error between the phase eigenstrain increments observed in the training set of CCH simulations and the best possible approximation to these phase eigenstrain increments that can be achieved by a linear combination (2.20) of the eigenstrains observed at these sampling points in the training data. This can be formalized as follows. Let $\{\hat{\kappa}_k\} : k = 1, \dots, N_{\mathrm{sampl}}^{(\beta)}$ be a set of numbers in ascending order drawn without replacement from the set of indices of Gauss points in phase β, and assemble the elements of this set into a vector $\hat{\boldsymbol{\kappa}}$. Let $\{\hat{\mathbf{y}}_k\} : k = 1, \dots, N_{\mathrm{sampl}}^{(\beta)}$ be the set of Gauss points corresponding to the indices in $\hat{\boldsymbol{\kappa}}$. Recasting the sampling point eigenstrain increments in terms of $\hat{\boldsymbol{\kappa}}$, let $\Delta\boldsymbol{\mu}_J^{(\hat{\kappa}_k)mn} \equiv \Delta\boldsymbol{\mu}_J^{\mathrm{f},mn}(\hat{\mathbf{y}}_k)$. The optimal set of sampling point indices $\boldsymbol{\kappa}^{(\beta)}$ in the least squares sense would satisfy

$$\begin{aligned}
\boldsymbol{\kappa}^{(\beta)} &= \underset{\hat{\boldsymbol{\kappa}} \in \mathcal{K}^{(\beta)}}{\operatorname{argmin}}\, C(\hat{\boldsymbol{\kappa}}) \\
C(\hat{\boldsymbol{\kappa}}) &= \sum_J \min_{\hat{\boldsymbol{\xi}}} \sum_{mn} \sum_{ij} \left[\Delta\mu_{Jij}^{(\beta)mn} - \sum_{k=1}^{N_{\mathrm{sampl}}^{(\beta)}} \hat{\xi}_{J\hat{\kappa}_k} \Delta\mu_{Jij}^{(\hat{\kappa}_k)mn} \right]^2
\end{aligned} \tag{2.21}$$

The objective function can be rewritten as

$$\begin{aligned}
C(\hat{\boldsymbol{\kappa}}) &= \sum_J \sum_{mn} \sum_{ij} \left(r_{Jij}^{*(\beta)mn}(\hat{\boldsymbol{\kappa}}) \right)^2 \\
\mathbf{r}_J^{*(\beta)mn}(\hat{\boldsymbol{\kappa}}) &= \mathbf{c}_J^{(\beta)mn} - \mathbf{A}_J^{(\beta)mn}(\hat{\boldsymbol{\kappa}}) \left[\sum_{pq} \left(\mathbf{A}_J^{(\beta)pq}(\hat{\boldsymbol{\kappa}}) \right)^{\mathrm{T}} \mathbf{A}_J^{(\beta)pq}(\hat{\boldsymbol{\kappa}}) \right]^{-1} \left[\sum_{ab} \left(\mathbf{A}_J^{(\beta)ab}(\hat{\boldsymbol{\kappa}}) \right)^{\mathrm{T}} \mathbf{c}_J^{(\beta)ab} \right]
\end{aligned} \tag{2.22}$$

Here, the matrices $\mathbf{A}_J^{(\beta)mn}(\hat{\boldsymbol{\kappa}})$ and $\mathbf{c}_J^{(\beta)mn}$ are defined by

$$\begin{aligned}
\mathbf{A}_J^{(\beta)mn} &= \begin{bmatrix} \Delta\mu_{J11}^{\mathrm{f},mn}(\mathbf{y}_1^{(\beta)}) & \dots & \Delta\mu_{J11}^{\mathrm{f},mn}(\mathbf{y}_{N_{\mathrm{sampl}}^{(\beta)}}^{(\beta)}) \\ \vdots & \ddots & \vdots \\ \Delta\mu_{J12}^{\mathrm{f},mn}(\mathbf{y}_1^{(\beta)}) & \dots & \Delta\mu_{J12}^{\mathrm{f},mn}(\mathbf{y}_{N_{\mathrm{sampl}}^{(\beta)}}^{(\beta)}) \end{bmatrix} \\
\mathbf{c}_J^{(\beta)mn} &= \begin{bmatrix} \Delta\mu_{J11}^{(\beta)mn} \\ \vdots \\ \Delta\mu_{J12}^{(\beta)mn} \end{bmatrix}
\end{aligned} \tag{2.23}$$

Here, $\mathbf{A}_J^{(\beta)mn}$ has dimensions $6 \times N_{\mathrm{sampl}}^{(\beta)}$ and contains components of the sampling point eigenstrain increments at snapshot J of training simulation mn in phase β. The vector $\mathbf{c}_J^{(\beta)mn}$ has six components and contains the phase-average eigenstrain increment in phase β observed at snapshot J of training simulation mn. Observe that $\mathbf{A}_J^{(\beta)pq}(\hat{\boldsymbol{\kappa}})$ depends on the set of sampling points chosen since the entry at row ij and column k of this matrix is $A_{Jijk}^{(\beta)pq}(\hat{\boldsymbol{\kappa}}) = \Delta\mu_{Jij}^{(\hat{\kappa}_k)mn}$.

The size of the sample space $\mathcal{K}^{(\beta)}$ of vectors of Gauss point indices is the binomial coefficient $\mathrm{Ch}(N_{\mathrm{gp}}^{(\beta)}, N_{\mathrm{sampl}}^{(\beta)})$ elements:

$$\mathrm{Ch}(N_{\mathrm{gp}}^{(\beta)}, N_{\mathrm{sampl}}^{(\beta)}) = \frac{N_{\mathrm{gp}}^{(\beta)}!}{N_{\mathrm{sampl}}^{(\beta)}! \left(N_{\mathrm{gp}}^{(\beta)} - N_{\mathrm{sampl}}^{(\beta)} \right)!} \tag{2.24}$$

In general, this is an enormous number. For example, the number of possible sets of 10 sampling points in a unit cell phase with 5000 Gauss points is on the order of 10^{30}. Moreover, gradient-based methods are not an option for a discrete optimization problem such as this. For the purposes of finding a set of sampling points to approximate the phase eigenstrain by (2.20), it is sufficient to find a set of sampling points which achieves a value of the objective function $C(\hat{\boldsymbol{\kappa}})$ below a user-specified tolerance, or else the best set that can be found within a user-specified maximum number of iterations. The specific locations of the sampling points have little importance outside of their success in minimizing the objective function. Based on these considerations, a heuristic approach to optimization is most suitable for our purposes. We employ a simple genetic algorithm, described as follows:

Algorithm 1:

Given the following hyperparameters:

- `np`: number of parents, taken from sample set $\mathcal{K}^{(\beta)}$
- `nc`: number of children, taken from sample set $\mathcal{K}^{(\beta)}$

- pimm: probability of immigration
- pmut: probability of mutation
- ngen_max: maximum number of generations
- tol: maximum acceptable value of objective function

1. Generate an initial population of parents $\{\boldsymbol{\kappa}^{\mathtt{ip}}\} : \mathtt{ip} = 1, \dots, \mathtt{np}$ via the immigration operation.
2. Evaluate the objective function (2.22) for each of these parents and sort the parents on this basis.
3. Let igen be the current generation number. Let tar_min_igen be the minimum value of the objective function achieved by any of the current parents. While igen < ngen_max and tar_min_igen > tol, do the following:

    3.1 Generate a population of children $\{\boldsymbol{\kappa}^{\mathtt{ic}}\} : \mathtt{ic} = 1, \dots, \mathtt{nc}$ based on the set of parents. A child is generated by the immigration operation (Algorithm 2) with probability pimm and by the mutation operation (Algorithm 3) with probability pmut.
    3.2 Evaluate the objective function for each of these children and sort the children on this basis.
    3.3 From among all np + nc parents and children, choose the np members that achieve the lowest values of the objective function. These members survive to the next generation as the new parents.

It should be noted that a crossover operation is commonly included in genetic algorithms. However, crossover has been omitted here because it was not found to speed up convergence or otherwise improve the solution during the implementation of this algorithm. The algorithm for the immigration operation is as follows:

Algorithm 2:
Output: Child $\boldsymbol{\kappa}^{\mathtt{ic}}$.

1. Draw an equal-probability sample from $\mathcal{K}^{(\beta)}$. In other words, randomly choose without replacement $N_{\text{sampl}}^{(\beta)}$ from among the $N_{\text{gp}}^{(\beta)}$ Gauss points within phase β. List their indices in a vector.
2. Sort these indices in ascending order and assign this list to $\boldsymbol{\kappa}^{\mathtt{ic}}$.

Conceptually, the immigration operation can be thought of as importing a new genetic sequence with no relation to the existing population. The algorithm for the mutation operation is as follows:

Algorithm 3:
Output: Child $\boldsymbol{\kappa}^{\mathtt{ic}}$.

1. Draw an integer ip from $1, \dots, \mathtt{np}$ with uniform probability.
2. Draw an integer isamp from $1, \dots, N_{\text{sampl}}^{(\beta)}$ with uniform probability.
3. Let $\kappa_{\mathtt{jsamp}}^{\mathtt{ip}}$ be the Gauss point index at position jsamp of parent $\boldsymbol{\kappa}^{\mathtt{ip}}$. Draw a Gauss point index igp within phase β in the interval $(\kappa_{\mathtt{isamp}-1}^{\mathtt{ip}}, \kappa_{\mathtt{isamp}+1}^{\mathtt{ip}})$, with uniform probability.
4. Perform the following assignment operations:

$$\begin{aligned} \boldsymbol{\kappa}^{\mathtt{ic}} &\leftarrow \boldsymbol{\kappa}^{\mathtt{ip}} \\ \kappa_{\mathtt{isamp}}^{\mathtt{ic}} &\leftarrow \mathtt{igp} \end{aligned} \tag{2.25}$$

The mutation operation can be thought of as asexual reproduction of a member of the parent population, in which one link in the parent's genetic sequence is changed to a new value. As a whole, genetic algorithms are meant to strike a balance at each iteration between randomly probing an enormous sample space and perturbing the best candidate solutions so far. In this way, with the right choices of hyperparameters, the algorithm is likely to find a variety of locally-best solutions, from which a good-quality global solution can be chosen. A more detailed treatment of genetic algorithms may be found in [98].

2.3.2. Sampling point contribution factors

Restricting our attention to material phase β, now suppose we have chosen sampling points $\{\mathbf{y}_k^{(\beta)}\} : k = 1, \dots, N_{\text{sampl}}^{(\beta)}$ according to the procedure outlined in the previous section and we are given a candidate set of contribution factor snapshots $\{\hat{\xi}_k\} : k = 1, \dots, N_{\text{sampl}}^{(\beta)}$ arranged in a vector $\hat{\boldsymbol{\xi}}$. Using the same training data set as in Section 2.2, let $\Delta\boldsymbol{\mu}_J^{(\beta)mn}$ denote the average eigenstrain increment in phase β at snapshot J of CCH training simulation mn. Similarly, let $\Delta\boldsymbol{\mu}_J^{\mathrm{f},mn}(\mathbf{y})$ denote the fine-scale eigenstrain increment at fine-scale coordinate $\mathbf{y}$ at snapshot J of CCH training simulation mn. For a given snapshot J of training simulation mn, we can define the residual between the observed phase eigenstrain increment and the sampling point approximation as follows:

$$\mathbf{r}_J^{(\beta)mn}(\hat{\boldsymbol{\xi}}) = \Delta\boldsymbol{\mu}_J^{(\beta)mn} - \sum_{k=1}^{N_{\text{sampl}}^{(\beta)}} \hat{\xi}_k \Delta\boldsymbol{\mu}_J^{\mathrm{f},mn}(\mathbf{y}_k^{(\beta)}) \tag{2.26}$$

The optimal set of contribution factors (arranged in a vector $\boldsymbol{\xi}_J^{(\beta)}$) at snapshot J in the least squares sense would be that which minimizes the sum of squared residual components over all training simulations:

$$\boldsymbol{\xi}_J^{(\beta)} = \underset{\hat{\boldsymbol{\xi}}}{\operatorname{argmin}} \sum_{mn} \sum_{ij} \left(r_{Jij}^{(\beta)mn}(\hat{\boldsymbol{\xi}}) \right)^2 \tag{2.27}$$

Equivalently, the vector of contribution factors is the solution of the following least squares system of equations, with $\mathbf{A}_J^{(\beta)mn}$ and $\mathbf{c}_J^{(\beta)mn}$ defined in (2.23):

$$\left[\sum_{mn}\left(\mathbf{A}_J^{(\beta)mn}\right)^{\mathrm{T}}\mathbf{A}_J^{(\beta)mn}\right]\boldsymbol{\xi}_J^{(\beta)}=\sum_{mn}\left(\mathbf{A}_J^{(\beta)mn}\right)^{\mathrm{T}}\mathbf{c}_J^{(\beta)mn} \tag{2.28}$$

The least squares system in (2.28), which has dimensions $N_{\mathrm{sampl}}^{(\beta)}\times N_{\mathrm{sampl}}^{(\beta)}$, is solved for each unit cell material phase and each snapshot.

2.4. Online stage of the solver-free ROH

During the online stage of the computation, unit cell phase stresses and strains are computed with the help of the eigenstrain influence tensors described in Section 2.2 and the eigenstrain sampling procedure described in Section 2.3. From the offline stage of the computation, we have a set of sampling point indices $\{\kappa_k^{(\beta)}\}$ and corresponding Gauss points coordinates $\{\mathbf{y}_k^{(\beta)}\}$, sampling point coarse-scale strain influence tensors $\{\mathbf{E}^{(\kappa_k^{(\beta)})}\}$, snapshots of sampling point eigenstrain influence tensors $\{\mathbf{P}_J^{(\kappa_k^{(\beta)})}\}$ and snapshots of sampling point contribution factors $\{\xi_{Jk}^{(\beta)}\}$, with $k=1,\dots,N_{\mathrm{sampl}}^{(\beta)}$; $\beta=1,\dots,N_{\mathrm{ph}}$; $J=1,\dots,N_{\mathrm{snap}}$. At the beginning of each load increment, the appropriate $\mathbf{P}$ tensors and sampling point contribution factors are interpolated from the precomputed values using the following procedure:

1. The value of $|_0\boldsymbol{\mu}|$ at the current increment is situated between snapshots $|_0\boldsymbol{\mu}|_L$ and $|_0\boldsymbol{\mu}|_{L+1}$, corresponding to eigenstrain influence tensor snapshots $\mathbf{P}_L^{(\kappa_k^{(\beta)})}$ and $\mathbf{P}_{L+1}^{(\kappa_k^{(\beta)})}$ and sampling point contribution factors $\xi_{Lk}^{(\beta)}$ and $\xi_{(L+1)k}^{(\beta)}$.
2. For each sampling point $\kappa_k^{(\beta)}$, the interpolated eigenstrain influence tensor is computed as

$$\mathbf{P}^{(\kappa_k^{(\beta)})}=\mathbf{P}_L^{(\kappa_k^{(\beta)})}+\frac{|_0\boldsymbol{\mu}|-|_0\boldsymbol{\mu}|_L}{|_0\boldsymbol{\mu}|_{L+1}-|_0\boldsymbol{\mu}|_L}\left(\mathbf{P}_{L+1}^{(\kappa_k^{(\beta)})}-\mathbf{P}_L^{(\kappa_k^{(\beta)})}\right) \tag{2.29}$$

Each phase-average eigenstrain influence tensor $\mathbf{P}^{(\beta)}$ is computed in the same fashion. Similarly, the interpolated sampling point contribution factors are given by

$$\xi_k^{(\beta)}=\xi_{Lk}^{(\beta)}+\frac{|_0\boldsymbol{\mu}|-|_0\boldsymbol{\mu}|_L}{|_0\boldsymbol{\mu}|_{L+1}-|_0\boldsymbol{\mu}|_L}\left(\xi_{(L+1)k}^{(\beta)}-\xi_{Lk}^{(\beta)}\right) \tag{2.30}$$

It may be noted that this interpolation procedure is based only on the norm of homogenized eigenstrain, despite the fact that different individual components of the eigenstrain may be activated. Recall that the fourth-order eigenstrain influence tensors are computed based on CCH training simulations in which different eigenstrain components are activated. Each column mn of the eigenstrain influence tensor $\mathbf{P}_J$ represents the increment of fine-scale strain due to a unit increment of homogenized eigenstrain in the mn-direction during a unit cell material history for which $|_0\boldsymbol{\mu}|=|_0\boldsymbol{\mu}|_J$. Thus, the influence of different eigenstrain components on the fine-scale strain are captured by the operation of the influence function tensor on the homogenized eigenstrain tensor (2.14). Now that the $\mathbf{P}$ tensors and sampling point contribution factors are established for the current load increment, the coarse-scale stress is computed using the following fixed-point iteration scheme:

$$\begin{aligned}{}^{q+1}\sigma_{ij}^{\mathrm{c}}&=\sum_{\beta}\phi^{(\beta)}\,\sigma_{ij}^{(\beta)}({}^{q}\boldsymbol{\sigma}^{\mathrm{c}})\\ {}^{0}\boldsymbol{\sigma}^{\mathrm{c}}&={}_0\boldsymbol{\sigma}^{\mathrm{c}}+\mathbf{C}^{\mathrm{c}}:\Delta\boldsymbol{\epsilon}^{\mathrm{c}}\qquad\text{(Initial condition)}\end{aligned} \tag{2.31}$$

Each phase stress is given in terms of the phase strain increment and phase eigenstrain increment by

$$\sigma_{ij}^{(\beta)}({}^{q}\boldsymbol{\sigma}^{\mathrm{c}})={}_0\sigma_{ij}^{(\beta)}+L_{ijab}^{(\beta)}\left(\Delta\epsilon_{ab}^{(\beta)}({}^{q}\boldsymbol{\sigma}^{\mathrm{c}})-\Delta\mu_{ab}^{(\beta)}({}^{q}\boldsymbol{\sigma}^{\mathrm{c}})\right) \tag{2.32}$$

The phase strain increment is computed using the phase-average influence function tensors:

$$\Delta\epsilon_{ij}^{(\beta)}({}^{q}\boldsymbol{\sigma}^{\mathrm{c}})=E_{ij}^{ab(\beta)}\Delta\epsilon_{ab}^{\mathrm{c}}+P_{ij}^{ab(\beta)}\Delta\mu_{ab}^{\mathrm{c}}({}^{q}\boldsymbol{\sigma}^{\mathrm{c}}) \tag{2.33}$$

In this expression, the increment of homogenized eigenstrain is computed from the previous *iteration* of the coarse-scale stress in the current increment, according to

$$\Delta\mu_{ij}^{\mathrm{c}}({}^{q}\boldsymbol{\sigma}^{\mathrm{c}})=\Delta\epsilon_{ij}^{\mathrm{c}}-C_{ijab}^{\mathrm{c}}\left({}^{q}\sigma_{ab}^{\mathrm{c}}-{}_0\sigma_{ab}^{\mathrm{c}}\right) \tag{2.34}$$

As described in Section 2.3, the phase eigenstrains are given by

$$\Delta\mu_{ij}^{(\beta)}({}^{q}\boldsymbol{\sigma}^{\mathrm{c}})=\sum_{k=1}^{N_{\mathrm{sampl}}^{(\beta)}}\xi_k^{(\beta)}\,\Delta\mu_{kij}^{\mathrm{f}(\beta)}({}^{q}\boldsymbol{\sigma}^{\mathrm{c}}) \tag{2.35}$$

Each sampling point eigenstrain increment is computed from the sampling point stress and strain using the phase constitutive model, according to

$$\Delta\mu_{kij}^{\mathrm{f}(\beta)}({}^{q}\boldsymbol{\sigma}^{\mathrm{c}})=\Delta\epsilon_{kij}^{\mathrm{f}(\beta)}({}^{q}\boldsymbol{\sigma}^{\mathrm{c}})-C_{ijab}^{(\beta)}\left[\hat{\sigma}_{ab}^{(\beta)}\left({}_0\epsilon_k^{\mathrm{f}(\beta)}+\Delta\epsilon_k^{\mathrm{f}(\beta)}({}^{q}\boldsymbol{\sigma}^{\mathrm{c}}),{}_0\mathbf{s}_k^{\mathrm{f}(\beta)}\right)-{}_0\sigma_{kab}^{\mathrm{f}(\beta)}\right] \tag{2.36}$$

Here, the sampling point strain increments are found using the sampling point influence function tensors, via

$$\Delta\epsilon^{\mathrm{f}(\beta)}_{kij}({}^{q}\boldsymbol{\sigma}^{\mathrm{c}}) = E^{ab(\kappa^{(\beta)}_k)}_{ij}\Delta\epsilon^{\mathrm{c}}_{ab} + P^{ab(\kappa^{(\beta)}_k)}_{ij}\Delta\mu^{\mathrm{c}}_{ab}({}^{q}\boldsymbol{\sigma}^{\mathrm{c}}) \tag{2.37}$$

The variables ${}_0\boldsymbol{\sigma}^{\mathrm{f}(\beta)}_k$, ${}_0\boldsymbol{\epsilon}^{\mathrm{f}(\beta)}_k$ and ${}_0\mathbf{s}^{\mathrm{f}(\beta)}_k$ are tracked as state variables for the sampling point $\kappa^{(\beta)}_k$. The increment of homogenized eigenstrain in (2.37) is again given by (2.34).

2.4.1. Coarse-scale tangent modulus

The following is a formulation of an algorithmically consistent coarse-scale tangent modulus for use in the Newton–Raphson loop of the coarse-scale solver. Applying the chain rule to (2.31) and (2.32) yields

$$\begin{aligned}\frac{\partial\Delta\sigma^{\mathrm{c}}_{ij}}{\partial\Delta\epsilon^{\mathrm{c}}_{mn}} &= \sum_{\beta}\phi^{(\beta)}\frac{\partial\Delta\sigma^{(\beta)}_{ij}}{\partial\Delta\epsilon^{\mathrm{c}}_{mn}}\\ &= \sum_{\beta}\phi^{(\beta)}L^{(\beta)}_{ijab}\left(\frac{\partial\Delta\epsilon^{(\beta)}_{ab}}{\partial\Delta\epsilon^{\mathrm{c}}_{mn}} - \frac{\partial\Delta\mu^{(\beta)}_{ab}}{\partial\Delta\epsilon^{\mathrm{c}}_{mn}}\right)\end{aligned} \tag{2.38}$$

Differentiating (2.33) with respect to coarse-scale strain yields

$$\frac{\partial\Delta\epsilon^{(\beta)}_{ij}}{\partial\Delta\epsilon^{\mathrm{c}}_{mn}} = E^{mn(\beta)}_{ij} + P^{ab(\beta)}_{ij}\frac{\partial\Delta\mu^{\mathrm{c}}_{ab}}{\partial\Delta\epsilon^{\mathrm{c}}_{mn}} \tag{2.39}$$

Using the chain rule to differentiate (2.35) with respect to coarse-scale strain, then inserting (2.36) and (2.37), yields

$$\begin{aligned}\frac{\partial\Delta\mu^{(\kappa)}_{ij}}{\partial\Delta\epsilon^{\mathrm{c}}_{mn}} &= \sum_{k=1}^{N^{(\beta)}_{\mathrm{sampl}}}\xi^{(\beta)}_k\frac{\partial\Delta\mu^{\mathrm{f}(\beta)}_{kij}}{\partial\Delta\epsilon^{\mathrm{f}(\beta)}_{kab}}\frac{\partial\Delta\epsilon^{\mathrm{f}(\beta)}_{kab}}{\partial\Delta\epsilon^{\mathrm{c}}_{mn}}\\ &= \sum_{k=1}^{N^{(\beta)}_{\mathrm{sampl}}}\xi^{(\beta)}_k\left(I_{ijab} - C^{(\beta)}_{ijpq}L^{\mathrm{ep}\ (\kappa^{(\beta)}_k)}_{pqab}\right)\left(E^{mn(\kappa^{(\beta)}_k)}_{ab} + P^{cd(\kappa^{(\beta)}_k)}_{ab}\frac{\partial\Delta\mu^{\mathrm{c}}_{cd}}{\partial\Delta\epsilon^{\mathrm{c}}_{mn}}\right)\end{aligned} \tag{2.40}$$

The gradient of homogenized eigenstrain with respect to coarse-scale strain, which appears in both (2.39) and (2.40), is found in terms of the coarse-scale tangent modulus by differentiating (2.34):

$$\frac{\partial\Delta\mu^{\mathrm{c}}_{ij}}{\partial\Delta\epsilon^{\mathrm{c}}_{mn}} = I_{ijmn} - C^{\mathrm{c}}_{ijab}\frac{\partial\Delta\sigma^{\mathrm{c}}_{ab}}{\partial\Delta\epsilon^{\mathrm{c}}_{mn}} \tag{2.41}$$

Combining (2.38)–(2.41) yields the following system of equations for the coarse-scale tangent modulus:

$$\begin{aligned}&\left(I_{ijab} + \mathcal{P}_{ijcd}C^{\mathrm{c}}_{cdab}\right)\frac{\partial\Delta\sigma^{\mathrm{c}}_{ab}}{\partial\Delta\epsilon^{\mathrm{c}}_{mn}} = \mathcal{E}_{ijmn} + \mathcal{P}_{ijmn}\\ &\mathcal{E}_{ijmn} = \sum_{\beta}\phi^{(\beta)}L^{(\beta)}_{ijab}\left[E^{mn(\beta)}_{ab} + \sum_{k=1}^{N^{(\beta)}_{\mathrm{sampl}}}\xi^{(\beta)}_k\left(I_{abcd} - C^{(\beta)}_{abpq}L^{\mathrm{ep}\ (\kappa^{(\beta)}_k)}_{pqcd}\right)E^{mn(\kappa^{(\beta)}_k)}_{cd}\right]\\ &\mathcal{P}_{ijmn} = \sum_{\beta}\phi^{(\beta)}L^{(\beta)}_{ijab}\left[P^{mn(\beta)}_{ab} + \sum_{k=1}^{N^{(\beta)}_{\mathrm{sampl}}}\xi^{(\beta)}_k\left(I_{abcd} - C^{(\beta)}_{abpq}L^{\mathrm{ep}\ (\kappa^{(\beta)}_k)}_{pqcd}\right)P^{mn(\kappa^{(\beta)}_k)}_{cd}\right]\end{aligned} \tag{2.42}$$

3. Numerical results

In this section, four numerical examples are used to demonstrate the capabilities of the solver-free ROH approach outlined in Section 2. First, the geometry, discretization, and material properties of the unit cell used for the four examples are described in Section 3.1. Next, in Section 3.2, the eigenstrain influence function tensors and eigenstrain sampling point contribution factors are computed and used to reproduce the coarse-scale stress–strain curves from the CCH training data set. Section 3.3 verifies the solver-free ROH for a single coarse-scale element subjected to combined non-monotonic loading in tension and shear. Section 3.4 contains results from three-point bending of an eight-layer composite coupon, which demonstrate the combined accuracy and computational efficiency of the proposed method. Finally, Section 3.5 presents coarse-scale stress–strain results from tensile loading of a three-layer composite plate with a center hole.

3.1. Unit cell description

A continuous fiber unit cell with geometry shown in Fig. 1 is used in each of the numerical examples presented here. Its overall shape is a cube with edges of unit length. This unit cell is commonly used to model a carbon fiber-reinforced polymer composite. In this study, a fiber volume fraction 26% has been chosen, giving a fiber radius of 0.29. A $J2$-elastoplastic constitutive model is chosen

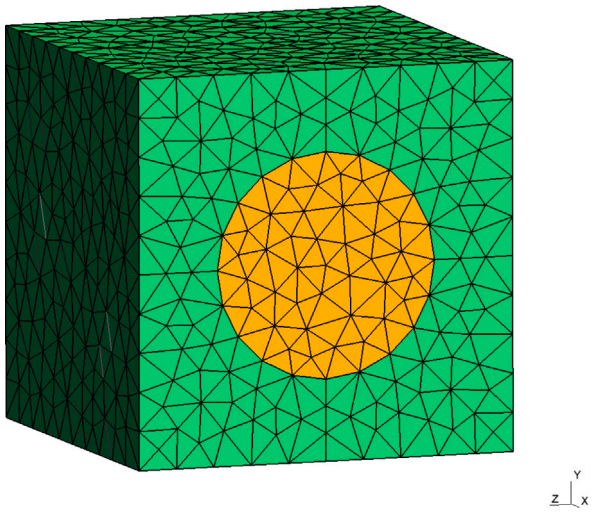


Fig. 1. Continuous fiber unit cell.

Table 1
Material properties of the unit cell matrix and fiber.

Property		Matrix ($\beta = 1$)	Fiber ($\beta = 2$)
E	Young's modulus, MPa	3800	252 000
ν	Poisson's ratio	0.32	0.02
σ_1	Yield strength, MPa	28.0	3400
σ_2	Ultimate strength, MPa	28.0	3800
δ	Hardening exponent	1.0	600
H	Hardening modulus, MPa	380.0	2000
	Hardening type	kinematic	kinematic
$\phi^{(\beta)}$	Volume fraction	0.74	0.26

for the fiber, with high stiffness and high yield stress relative to the matrix material. A more realistic model of carbon fiber would be a damage model. However, damage models present convergence and mesh-dependency challenges (even for the CCH reference solutions). Thus, more realistic constitutive models for the fiber will be considered in a future study. The matrix material is also modeled as a $J2$-elastoplastic material. The hardening law has a linear and an exponential term:

$$\hat{H}(a) = (\sigma_2 - \sigma_1)(1 - \exp(-\delta a)) + Ha \tag{3.1}$$

The specific matrix elastoplastic properties and fiber elastic properties chosen, given in Table 1, approximate those of epoxy resin and carbon fiber respectively [87,95]. The finite element mesh used for the majority of the examples here has 1500 nodes and 6593 linear tetrahedron elements, as shown in Fig. 1.

3.2. Reproduction of CCH training data

As described in Section 2, eigenstrain influence tensors and sampling point contribution factors are computed based on a small set of training data generated with the CCH. This section details how the training data is generated and evaluates how well the solver-free ROH is able to reproduce the coarse-scale stress–strain curves from the training data.

In this study, as in [87], training data is generated by six uni-directional loadings of a single coarse-scale 8-Gauss point brick finite element occupying the volume $[0,1]\times[0,1]\times[0,1]$ in $\mathbb{R}^3$. In each test mn, a minimum of constraints are applied so that stress in directions other than mn is close to zero. The specific boundary conditions used for the CCH tests are described in Tables 2 and 3 in terms of a control parameter η^{mn}. The parameter η^{mn} corresponds to the magnitude of strain in the loading direction and may be chosen by the user to cover a desired range of values of accumulated eigenstrain.

From these training data, 10 snapshots of the eigenstrain influence tensors are computed according to the procedure described in Section 2.2. Fig. 2 shows the variation of the Euclidean norm of four columns of the phase-average influence tensors over a range of values of the homogenized eigenstrain norm. As shown, these influence tensors vary gradually as the homogenized eigenstrain norm increases. For the unit cell presented here, the most marked variation is in the shear components of the influence tensors for values of eigenstrain norm close to zero, i.e., near the onset of accumulation of eigenstrain in the unit cell. Capturing this variation is key to accurately capturing the stress response near the onset of eigenstrain accumulation in the unit cell.

Table 2
Boundary conditions for unidirectional tensile tests ($mn = 11, 22, 33$).

Location on boundary	BC		
$x_1 = 0,\ x_2 = 0,\ x_3 = 0$	$u_1 = 0$	$u_2 = 0$	$u_3 = 0$
$x_1 = 0$	$u_1 = 0$		
$x_2 = 0$		$u_2 = 0$	
$x_3 = 0$			$u_3 = 0$
$x_m = 1$		$u_m = \eta^{mn}$	

Table 3
Boundary conditions for unidirectional shear tests ($mn = 23, 13, 12$).

Location on boundary	BC	
$x_p = 0$	$u_p = 0$	(for $p \neq m$ and $p \neq n$)
$x_n = 0$	$u_m = -\eta^{mn}/2$	
$x_m = 0$	$u_n = -\eta^{mn}/2$	
$x_n = 1$	$u_m = \eta^{mn}/2$	
$x_m = 1$	$u_n = \eta^{mn}/2$	

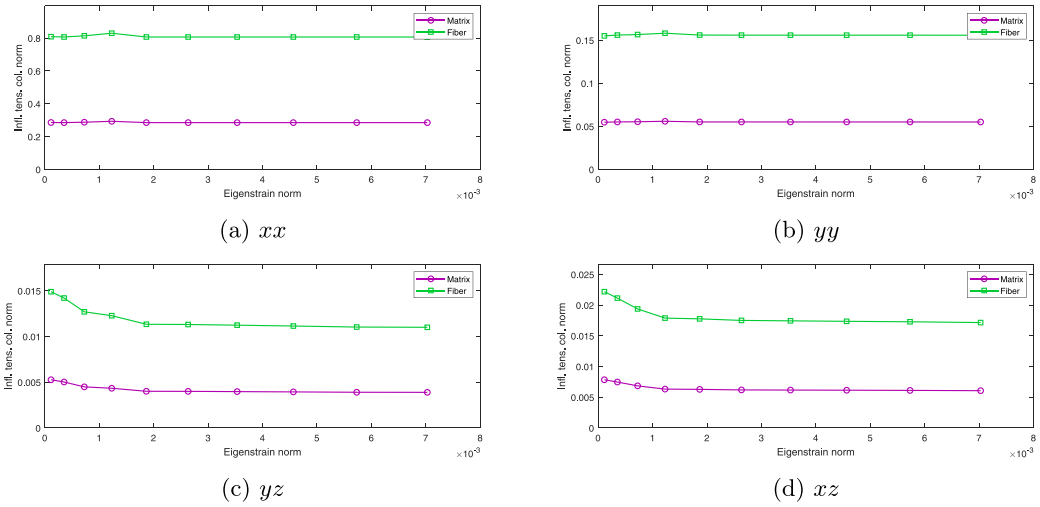


Fig. 2. Influence of individual components of eigenstrain on magnitude of phase-average strains: variation with homogenized eigenstrain norm.

Next, the locations of sampling points are chosen using the genetic algorithm described in Section 2.3.1. Because evaluation of the objective function used by the genetic algorithm is relatively inexpensive, we can use the following procedure to select the number of sampling points in each material phase:

1. Run the genetic algorithm with `ngen_max` = 100 (or some other small number) for various trial values of $N^{(\beta)}_{\text{sampl}}$, recording the minimum value of the objective function $C_{N^{(\beta)}_{\text{sampl}}}$ given in (2.22) achieved for each trial.
2. Define the 100-generation relative error δ for each $N^{(\beta)}_{\text{sampl}}$ by

$$\delta = \frac{C_{N^{(\beta)}_{\text{sampl}}}}{C_1} \tag{3.2}$$

The minimum value of C achievable in 100 generations of the genetic algorithm by a one-sampling-point approximation, C_1, is a useful worst-case reference value for evaluating the quality of an $N^{(\beta)}_{\text{sampl}}$-point approximation. For chosen relative threshold $\delta_{\max}$ and absolute threshold $C_{\max}$, choose the minimum number of sampling points that achieves $\delta < \delta_{\max}$ and/or $C < C_{\max}$.

Fig. 3 shows the value of the objective function vs. the number of sampling points resulting from the above procedure for the matrix phase. The y-value of each data point in this plot is $C_{N^{(1)}_{\text{sampl}}}$, the minimum value of the objective function returned by the genetic algorithm after 100 generations. The 48 points were generated in 114 s on a machine with an Intel(R) Core(TM) i7-9850H CPU @ 2.60 GHz. From this data, we choose $N^{(1)}_{\text{sampl}} = 10$ in the matrix phase and $N^{(2)}_{\text{sampl}} = 6$ in the fiber phase for the numerical examples in this study because they achieve $\delta < 1\times10^{-1}$ and $C < 2\times10^{-4}$. We have found this threshold to be sufficient to reproduce the training data. Below this threshold, the error due to the influence function approximation of fine-scale strain begins to dominate

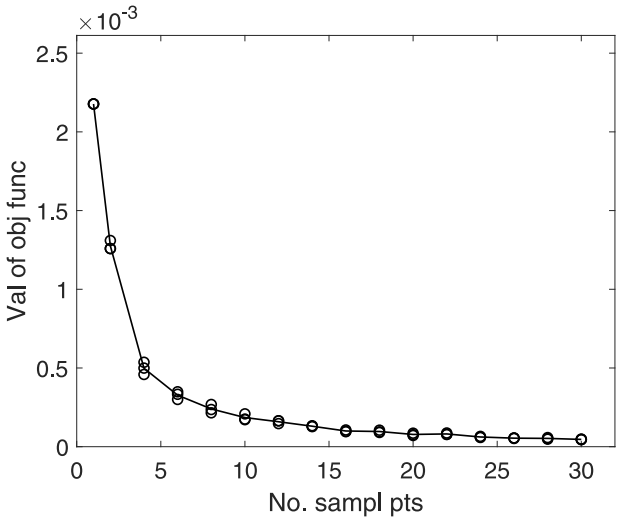


Fig. 3. 100-generation objective function in genetic algorithm vs. number of matrix sampling points.

Table 4
Choices of hyperparameters for the genetic algorithm.

Parameter		Value
`np`	Number of parents	100
`nc`	Number of children	100
`pimm`	Immigration probability	10%
`pmut`	Mutation probability	90%
`ngen_max`	Maximum number of generations	5000
`tol`	Stopping threshold for obj func	1.0e−4

the error due to the sampling point approximation of eigenstrain. Next, the location of the 10 sampling points are determined from one last application of the genetic algorithm, with chosen hyperparameters given in Table 4. Finally, the sampling point contribution factors are computed using the procedure described in Section 2.3.2.

Remark. Suitable hyperparameters for the genetic algorithm such as `pmut`, some of which affect the rate of convergence, may be selected using the 100-generation procedure described above. From Fig. 4, for example, we see that the range $0.6 \leq \texttt{pmut} = 1 - \texttt{pimm} \leq 1.0$ achieves the smallest minimum values of the objective function on average within 100 generations, with $\texttt{pmut} = 0.9$ achieving the minimum overall. A value of $\texttt{pmut} = 0.9$ has therefore been chosen for the numerical examples in this study.

In Fig. 5a–f, the coarse-scale stress–strain curves from the CCH training data are compared with solver-free ROH reproductions of these training data. The solver-free ROH results agree with the load test data, as expected. In practice, the assumption that the influence tensors and sampling point contribution factors depend on the unit cell material history only through a scalar variable $|_0\boldsymbol{\mu}|$ is imperfect. The manner in which these tensors and factors are pre-computed seeks to balance the discrepancy between the solver-free ROH solution and the CCH solution across training simulations. Thus, the small discrepancies between the solver-free ROH results and the CCH results shown in Fig. 5a–f are to be expected.

3.3. Multiaxial loading of a single coarse-scale element

In this section, we verify whether the solver-free ROH approach is sufficiently robust to capture the behavior of this composite in mixed-loading cases. For this purpose, we apply non-monotonic loading to the coarse-scale element from the previous section in two combinations of tension and shear. The prescribed displacements of the vertices of the coarse-scale element have been chosen to cause a significant amount of both types of strain simultaneously. The non-monotonic loading is achieved by applying the boundary conditions incrementally according to the following amplitude function of pseudo-time given by

$$\text{amp}(t) = \begin{cases} t & 0 \leq t < 1 \\ 2 - t & 1 \leq t < 2 \\ t - 2 & 2 \leq t \leq 3 \end{cases} \tag{3.3}$$

Here, pseudo-time $t = 3$ corresponds to the end of the simulation. The boundary conditions are described in more detail in Tables 5 and 6. The same unit cell mesh and material properties from the previous example are used in this example. The eigenstrain influence

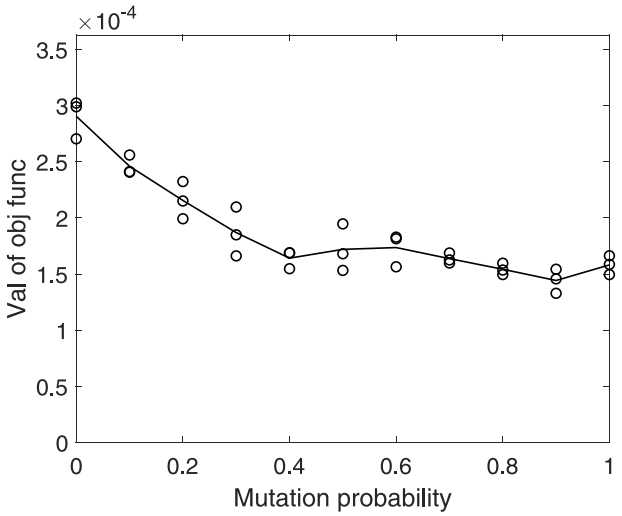


Fig. 4. 100-generation objective function in genetic algorithm vs. probability of mutation.

Table 5
Boundary conditions at conclusion of zz-xz loading.

Location on boundary	BC	$\bar{\epsilon}_{33} = 0.007$	$\bar{\epsilon}_{13} = 0.0045$
$x_2 = 0$		$u_2 = 0$	
$x_1 = 0,\ x_3 = 0$	$u_1 = -\bar{\epsilon}_{13}/2$		$u_3 = -\bar{\epsilon}_{13}/2$
$x_1 = 0,\ x_3 = 1$	$u_1 = \bar{\epsilon}_{13}/2$		$u_3 = -\bar{\epsilon}_{13}/2 + \bar{\epsilon}_{33}$
$x_1 = 1,\ x_3 = 0$	$u_1 = -\bar{\epsilon}_{13}/2$		$u_3 = \bar{\epsilon}_{13}/2$
$x_1 = 1,\ x_3 = 1$	$u_1 = \bar{\epsilon}_{13}/2$		$u_3 = \bar{\epsilon}_{13}/2 + \bar{\epsilon}_{33}$

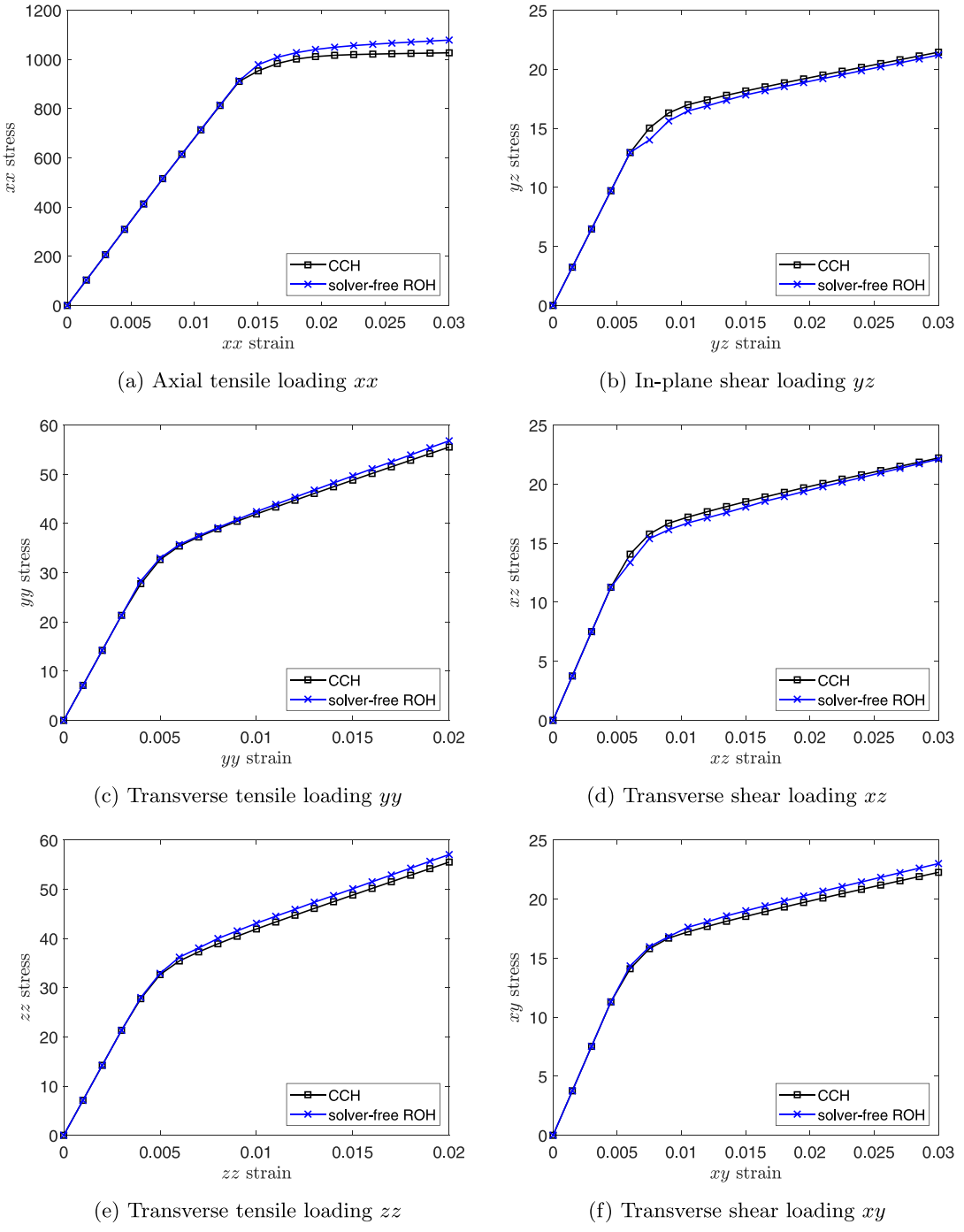
Table 6
Boundary conditions at conclusion of zz-yz loading.

Location on boundary	BC	$\bar{\epsilon}_{33} = 0.004$	$\bar{\epsilon}_{23} = 0.006$
$x_1 = 0$	$u_1 = 0$		
$x_2 = 0,\ x_3 = 0$		$u_2 = -\bar{\epsilon}_{23}/2$	$u_3 = -\bar{\epsilon}_{23}/2$
$x_2 = 0,\ x_3 = 1$		$u_2 = \bar{\epsilon}_{23}/2$	$u_3 = -\bar{\epsilon}_{23}/2 + \bar{\epsilon}_{33}$
$x_2 = 1,\ x_3 = 0$		$u_2 = -\bar{\epsilon}_{23}/2$	$u_3 = \bar{\epsilon}_{23}/2$
$x_2 = 1,\ x_3 = 1$		$u_2 = \bar{\epsilon}_{23}/2$	$u_3 = \bar{\epsilon}_{23}/2 + \bar{\epsilon}_{33}$

tensors and sampling point contribution factors from the previous example are re-used as well, since they are pre-computed for the same unit cell mesh and material properties.

As shown in Fig. 6, there is agreement between the stress–strain curves from the proposed approach and the CCH reference solutions for the mixed-loading cases given. As with the uni-directional results, the discrepancy between the CCH and solver-free ROH results for zz-xz loading stems from the assumption about equivalent unit cell states. In this case, complex interactions between tensile and shear loading are not fully captured by the unidirectional test data. Despite this assumption and despite pre-computation of the eigenstrain influence tensors and sampling point contribution factors using only uni-directional load test results, the solver-free ROH is sufficiently robust that it can capture the behavior of the composite under mixed loading conditions with reasonable accuracy. This is particularly evident when the solver-free ROH results are compared with results from the un-calibrated hybrid compatible–incompatible eigenstrain ROH (labeled ROH in the plot legend). Also notably, while the solver-free ROH does not resolve fine-scale fields like the solver-free CCH is able to do, the solver-free ROH still retains accuracy in the coarse-scale fields relative to the CCH.

In Fig. 7, relevant components of phase-average stress are plotted against the corresponding components of phase-average strain in the matrix phase for both multi-axial loading examples. Since plastic strain does not develop in the fiber phase in these examples, the plots of fiber stress vs. strain have been omitted for brevity. As with the coarse-scale stress–strain results, the solver-free ROH matrix stress–strain curves agree with the CCH results. The improvement in accuracy of the proposed method over the standard ROH method is pronounced near the onset of eigenstrain accumulation in each of the plots. Fig. 8 contains phase-average plots of plastic strain vs. strain in the matrix phase. When the eigenstrain in the unit cell is well-developed (e.g. during the unloading), deviation in plastic strain between the solver-free ROH and CCH is evident. However, the solver-free ROH still retains significantly greater accuracy than the standard ROH in most of the plots. Moreover, in the transverse tension/transverse shear loading example, the solver-free ROH captures the continued development of eigenstrain during unloading and reloading, whereas the standard ROH predicts no change in plastic strain during this stage.



(a) Axial tensile loading xx

(b) In-plane shear loading yz

(c) Transverse tensile loading yy

(d) Transverse shear loading xz

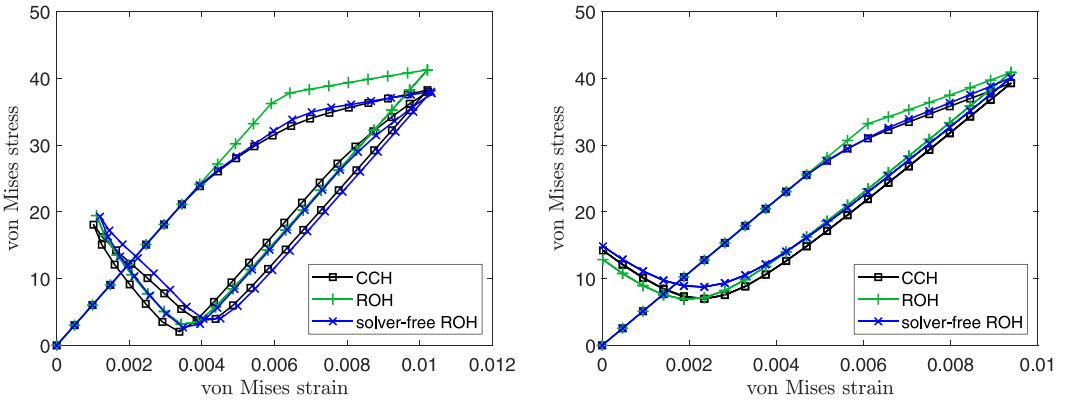
(e) Transverse tensile loading zz

(f) Transverse shear loading xy

Fig. 5. Uni-directional load tests: coarse-scale stress vs strain in primary loading direction.

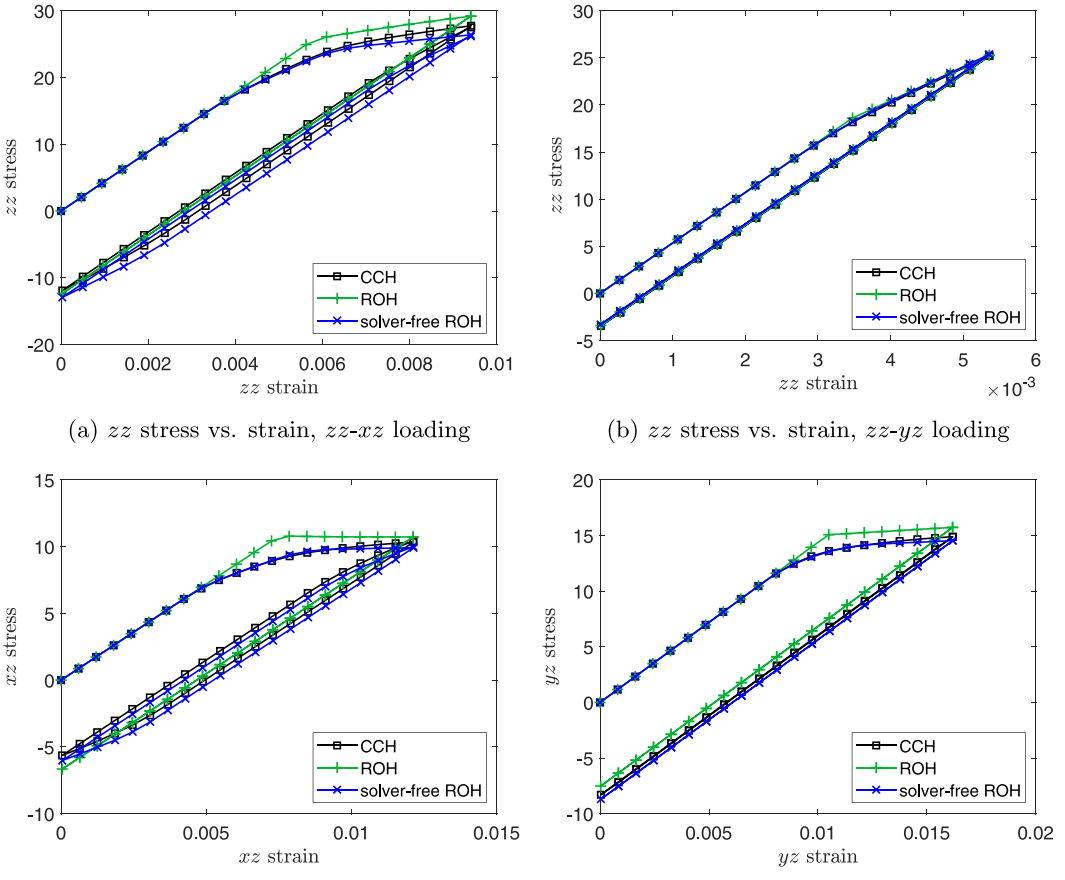
3.4. Computational cost savings: Three-point bending

To demonstrate the combination of accuracy and computational efficiency of the proposed solver-free ROH, we examine the results from a simulated three-point bending test of a composite coupon. The geometry of the bending coupon is shown in Fig. 9. The coupon is a thin plate with large-face dimensions 40 mm × 8 mm. The total thickness is 1.04 mm, divided into eight composite layers. The same continuous-fiber unit cell used in the previous examples is used to model all layers of the plate, with the unit cells oriented differently in each layer. The unit cell orientations in each layer, expressed as angle between fiber direction and coarse-scale



(a) zz-xz loading (b) zz-yz loading

Fig. 6. Multi-axial loading: coarse-scale von Mises stress vs strain.

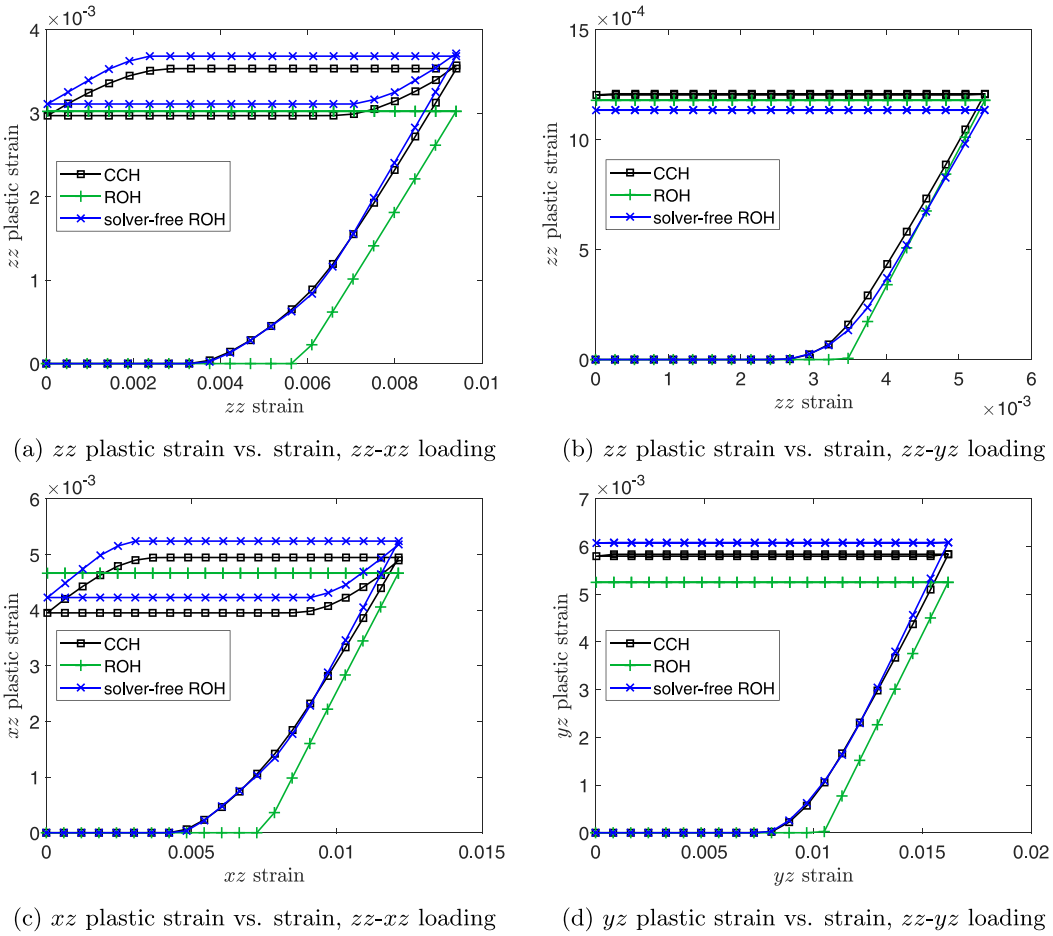


(a) zz stress vs. strain, zz-xz loading (b) zz stress vs. strain, zz-yz loading

(c) xz stress vs. strain, zz-xz loading (d) yz stress vs. strain, zz-yz loading

Fig. 7. Multi-axial loading: components of stress vs. strain in the matrix phase.

x axis, are provided in Table 7. The mesh used for this coarse-scale geometry contains 160 8-node brick elements. Such a coarse mesh has been chosen so that the CPU time required to compute the solution would be dominated by the unit cell computations rather than the coarse-scale solver. Moreover, using a coarse macro-scale mesh allowed us to minimize the high computational cost of generating CCH reference solutions for this example while still obtaining a meaningful comparison of the computational efficiency of the two methods.



(a) zz plastic strain vs. strain, zz-xz loading

(b) zz plastic strain vs. strain, zz-yz loading

(c) xz plastic strain vs. strain, zz-xz loading

(d) yz plastic strain vs. strain, zz-yz loading

Fig. 8. Multi-axial loading: components of plastic strain vs. strain in the matrix phase.

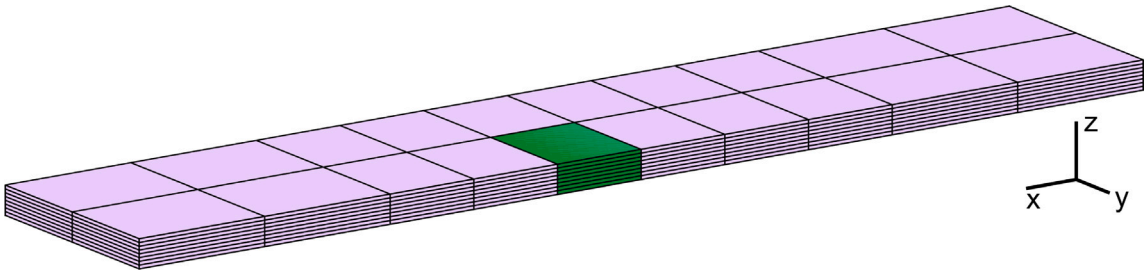


Fig. 9. Three-point bending: coarse-scale geometry and mesh. Critical elements highlighted.

Table 7
Unit cell orientations in layers of bending coupon.

Layer (Bottom to top)	1	2	3	4	5	6	7	8
Orientation	−45°	0°	+45°	90°	−45°	0°	+45°	90°

The coarse-scale boundary conditions used in the three-point bending are provided in Table 8. Fig. 10a-h show the stress–strain curves in the column of critical elements highlighted in Fig. 9, according to both the CCH reference solution and the solver-free ROH solution. The solver-free ROH stress–strain curves closely match those from CCH.

In Fig. 11, the von Mises stress distribution in the bottom layer of the coupon is shown for both CCH and solver-free ROH simulations. Fig. 11 shows that there is agreement between the CCH and solver-free ROH coarse-scale stress fields as a whole.

Table 8
Boundary conditions for three-point bending example.

Location on boundary	x BC	y BC	z BC	
$x = 0,\ y = 0,\ z = 0$	$u_x = 0$	$u_y = 0$	$u_z = 0$	(fixed corner)
$x = 0,\ z = 0$	$u_x = 0$		$u_z = 0$	(pinned end)
$x = 40$ mm, $z = 0$			$u_z = 0$	(roller end)
$x = 20$ mm, $z = 1.04$ mm			$u_z = -1.8$ mm	(displ. at center)

Table 9
Breakdown of computational cost for solver-free ROH: 15,000-element unit cell.

Offline stage		1.01E+03 s
Generating CCH data		6.79E+02 s
Pre-computing influence tensors		2.74e+01 s
Genetic algorithm		2.95e+02 s
Online stage	160 CS GP, 20 load incr	3.30E+01 s
All CS stress updates	per load incr	5.89E−01 s
CS solver	per load incr, per NR iter	4.65E−03 s
Avg number of NR iterations	per load incr	2.7

Table 10
Boundary conditions for plate with hole example.

Location on boundary	x BC	y BC	z BC	
$x = 0$	$u_x = 0$	$u_y = 0$	$u_z = 0$	(fixed left end)
$x = 40$ mm	$u_x = 0.32$ mm			(prescribed displ.)

We study the computational efficiency of the solver-free ROH approach relative to the CCH by repeating the bending simulation for four increasingly dense unit cell meshes. The CPU time required for the online stage of the computation is recorded for each mesh in both the CCH simulation and the solver-free ROH simulation. As shown in Fig. 12, the computational cost required for the online stage of the solver-free ROH method hardly changes as the number of elements in the unit cell mesh increases. In contrast, the computational cost of the CCH method is nearly a factor of 10 higher for the finest mesh than for the coarsest mesh. For our implementation, the online stage of the solver-free ROH approach achieved a speedup of 3300 for the most dense unit cell mesh of about 22,000 elements. For reference, the overall speedup reported for the solver-free CCH in [87] in a similar example was on the order of 10. This comparison indicates that the reduction in the number of fine-scale stress updates results in a significant additional reduction in computational time compared to both the solver-free CCH and the CCH. The computational costs of various components of the solver-free ROH algorithm are broken down in Table 9 for a unit cell mesh with approximately 15,000 elements. As expected, the coarse-scale solver accounts for very little computational cost relative to the unit cell computation in this example; this is true for both the CCH and solver-free ROH. It stands to reason that the solver-free ROH would achieve the most significant speedup over CCH in problems with complex unit cell geometries requiring many finite elements to resolve.

3.5. Open-hole tension of a composite plate

The final example considered in this study, open-hole tension of a thin composite plate, further demonstrates the accuracy achievable by the solver-free ROH. Following the example of [87], we consider a rectangular plate with dimensions 40 mm × 24 mm and total thickness 0.4088 mm, with a 8 mm-diameter circular hole at the center. The thickness (z) dimension comprises three layers, each represented at the fine scale by the unit cell used in the previous examples, with varying fiber orientations. The orientations from bottom to top layer, expressed as angle between fiber direction and coarse-scale x axis, are $-45°, +45°, 90°$. The mesh used to discretize the coarse-scale domain consists of 14,562 linear tetrahedral elements.

To simulate tensile loading, the left end of the plate is fixed in every direction and a prescribed x displacement is applied to the right end. Table 10 contains the details of the boundary conditions applied.

Plots of coarse-scale von Mises stress and strain are given in Fig. 13 for the maximum-stress and maximum-strain elements appearing in Figs. 14 and 15. Both the overall stress and strain fields and the critical element stress–strain curves show agreement between the CCH and solver-free ROH solutions. Figs. 14 and 15 compare the von Mises stress and strain distributions respectively in each layer of the plate that result from the CCH and solver-free ROH simulations.

4. Conclusion

The solver-free ROH approach devised in this study considerably reduces the computational cost of homogenization of heterogeneous media relative to the CCH, while preserving a great deal of accuracy. Like the ROH, the solver-free ROH relies on a set of pre-computed eigenstrain influence tensors. Unlike the ROH, however, snapshots of the solver-free ROH eigenstrain influence tensor are allowed to depend on the unit cell material history and computed based on a set of low-cost CCH training simulations,

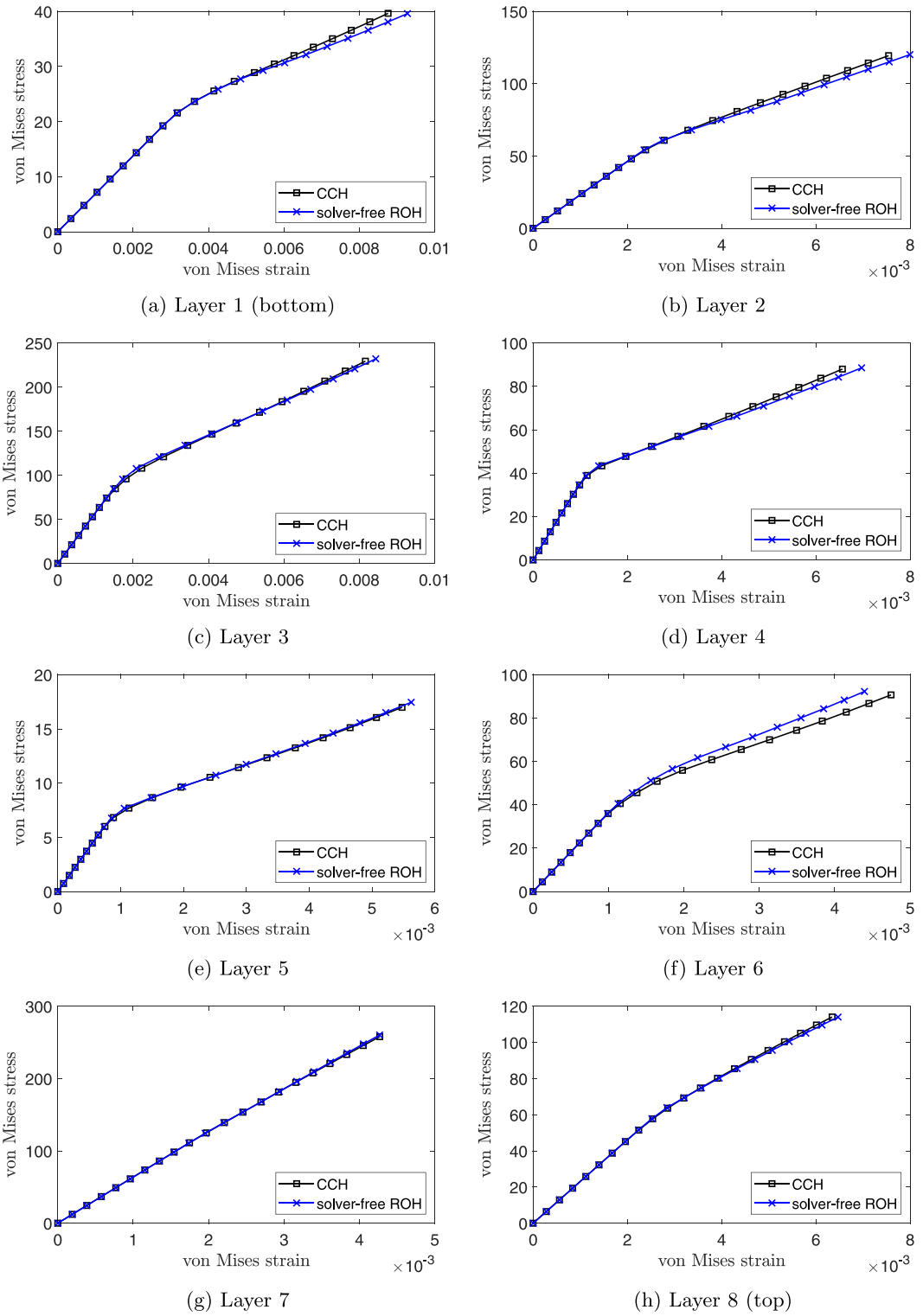
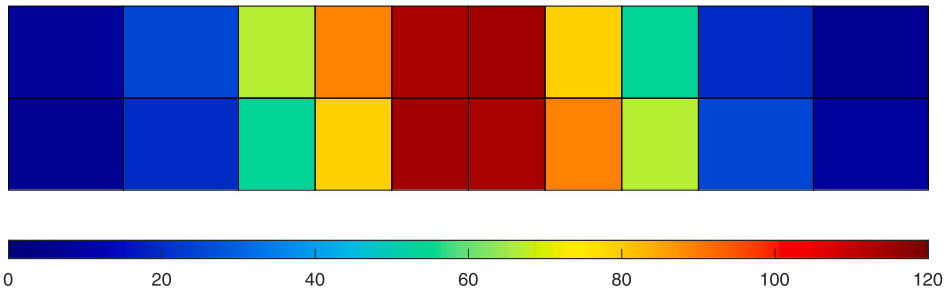
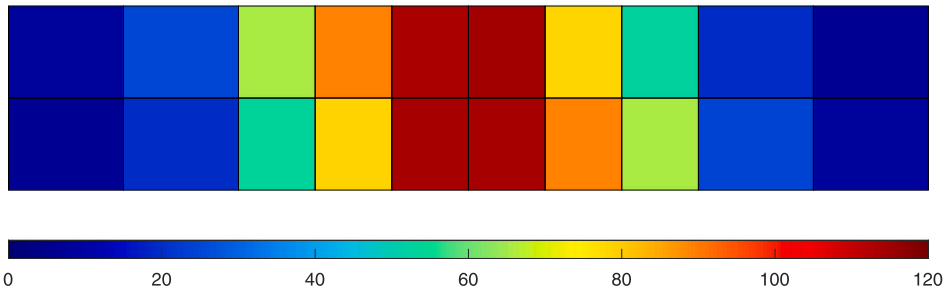


Fig. 10. Three-point bending: coarse-scale von Mises stress vs strain in critical elements.

as first described in [87] for the solver-free CCH. Unlike both the ROH and solver-free CCH, the solver-free ROH approximates the phase-average eigenstrains using a linear combination of fine-scale eigenstrains computed at pre-chosen sampling points. The



(a) CCH



(b) solver-free ROH

Fig. 11. Three-point bending: coarse-scale von Mises stress in bottom layer.

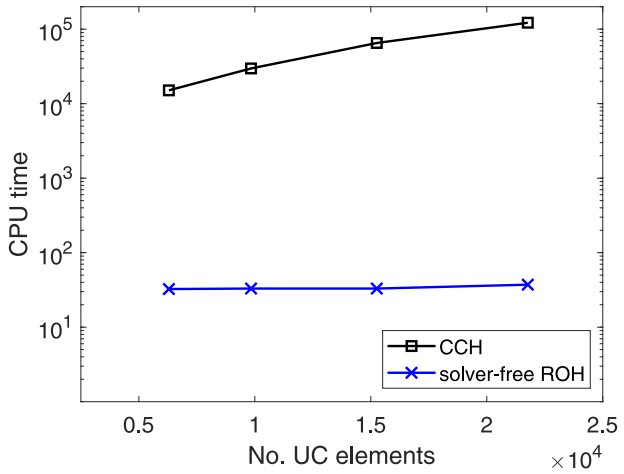
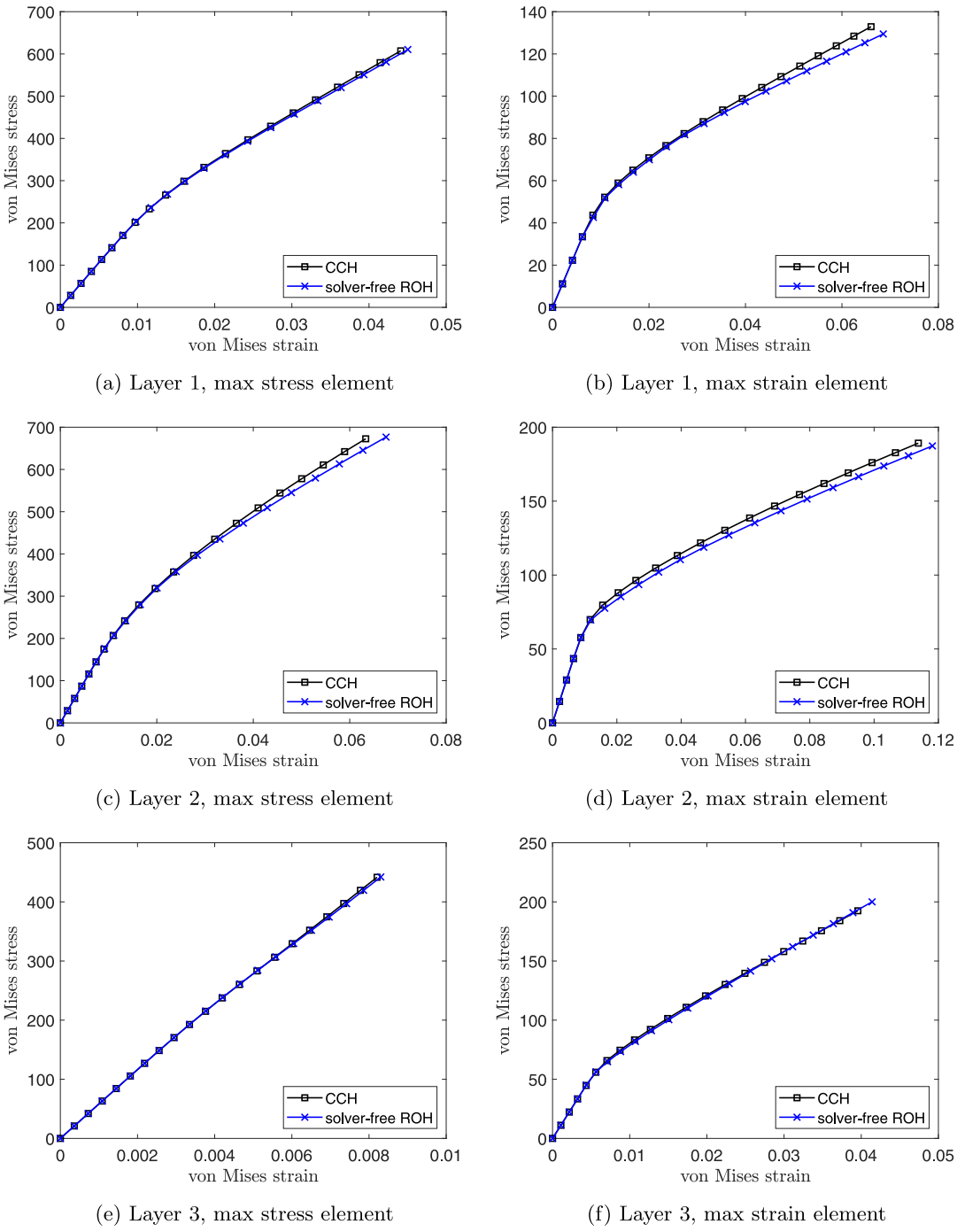


Fig. 12. Three-point bending: CPU time required for the online computation vs number of unit cell elements.

sampling points and snapshots of their weights in this linear combination are pre-computed using the same training data as has been used for the influence tensors.

The proposed approach has first been verified by reproducing the training set of stress–strain curves from uni-directional one-coarse-scale-element CCH simulations. Following this verification, the consistency of the solver-free ROH approach with CCH results has been further demonstrated in mixed-mode loading outside the training set, as well as bending coupon and open-hole plate tension examples. In addition to producing results consistent with CCH, the solver-free ROH approach has been shown to achieve a remarkable computational cost savings, which rises steeply as the density of the unit cell mesh is increased.

Part of this speedup is due to the removal of the fine-scale solver, as first described in [87]. This reduces the cost from approximately $\mathcal{O}(N_{\text{dof}}^3)$ to $\mathcal{O}(N_{\text{gp}})$, where N_{dof} is the number of unit cell degrees of freedom and N_{gp} is the number of unit cell



(a) Layer 1, max stress element

(b) Layer 1, max strain element

(c) Layer 2, max stress element

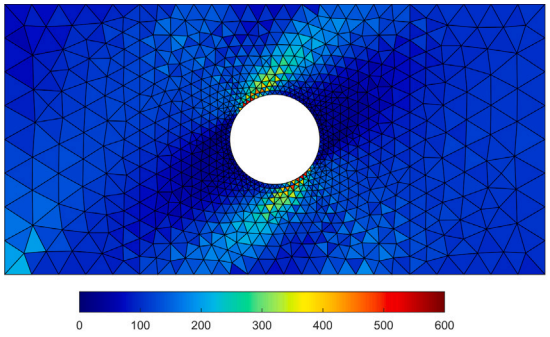
(d) Layer 2, max strain element

(e) Layer 3, max stress element

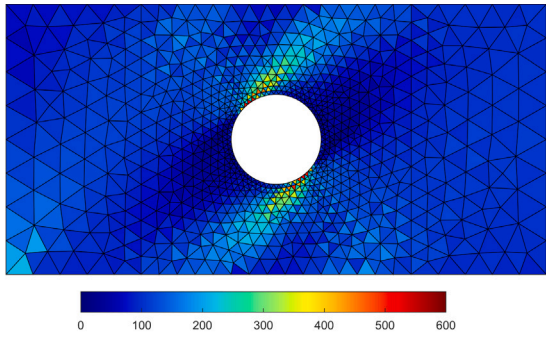
(f) Layer 3, max strain element

Fig. 13. Plate with hole: coarse-scale von Mises stress vs strain.

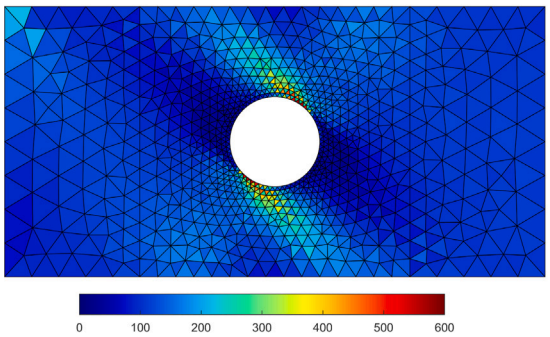
Gauss points, which is dramatic as unit cell density grows very large. Comparing the solver-free ROH to the solver-free CCH, however, we find the more significant cause of the speedup: the introduction of the sampling point approximation of the phase-average eigenstrains. This approximation reduces the cost of each coarse-scale stress update from $\mathcal{O}(N_{\text{gp}})$ to $\mathcal{O}(N_{\text{sampl}})$, where N_{sampl} is the user-specified number of unit cell sampling points. For a fixed number of sampling points relative to the density of the unit cell mesh, the solver-free ROH coarse-scale stress update is approximately constant-time, a characteristic it shares with the one-partition-per-phase ROH.



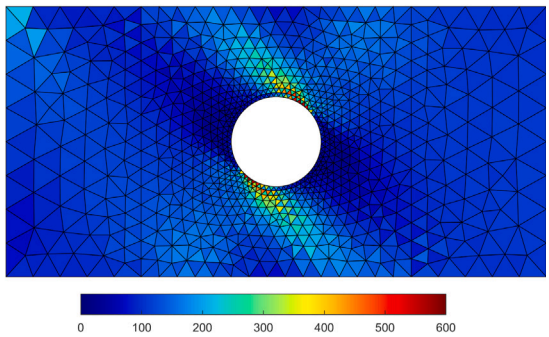
(a) Layer 1, CCH



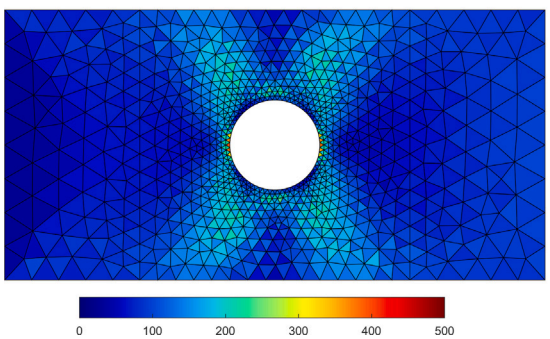
(b) Layer 1, solver-free ROH



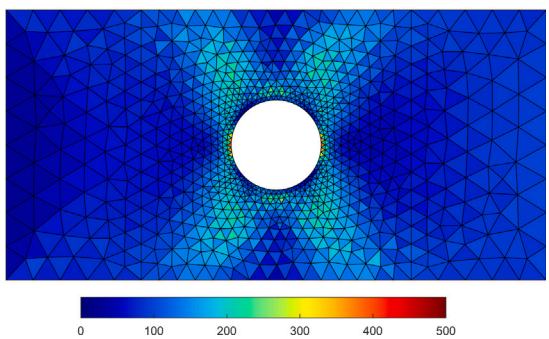
(c) Layer 2, CCH



(d) Layer 2, solver-free ROH



(e) Layer 3, CCH

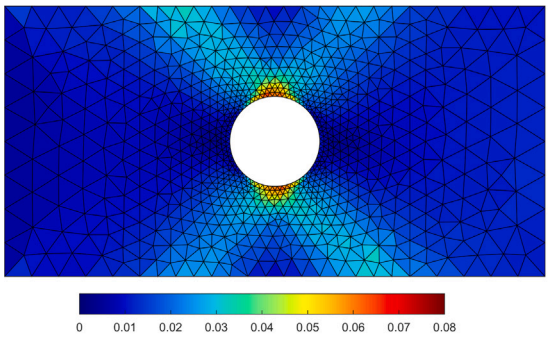


(f) Layer 3, solver-free ROH

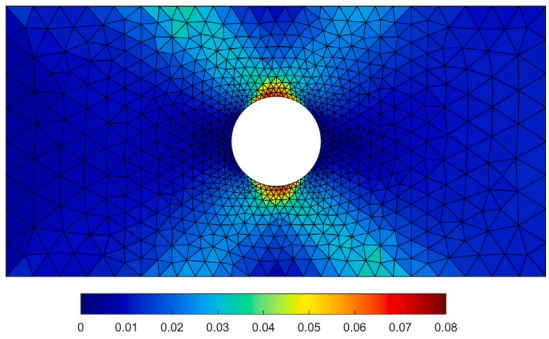
Fig. 14. Plate with hole: coarse-scale von Mises stress.

In the offline stage of the computation, a genetic algorithm is employed to identify suitable eigenstrain sampling points in the unit cell. While this approach works well for our purposes, it is cumbersome for large numbers of snapshots, sampling points, or training simulations. Moreover, it is a heuristic approach to optimization, so it lacks a guarantee that a globally optimum solution has been reached. In a future study, it would be worthwhile to devise a physically-based approach to identifying suitable sampling points, utilizing insight about the unit cell geometry and material properties.

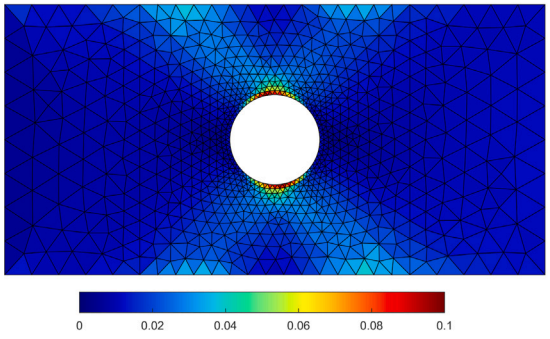
Another possible area for improvement would be in the choice of proxy variable (or variables) for the unit cell material history, perhaps using a data-driven approach. In this way, appropriate snapshots of eigenstrain influence tensors and sampling point contribution factors could be taken over a wide representative range of unit cell material histories.



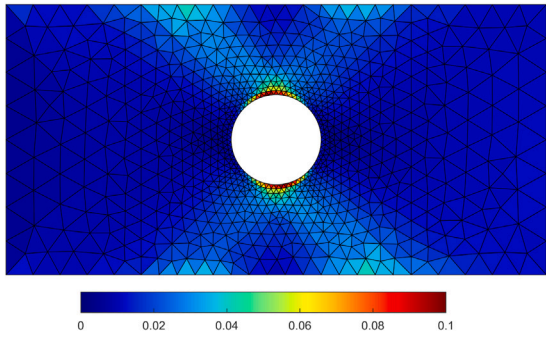
(a) Layer 1, CCH



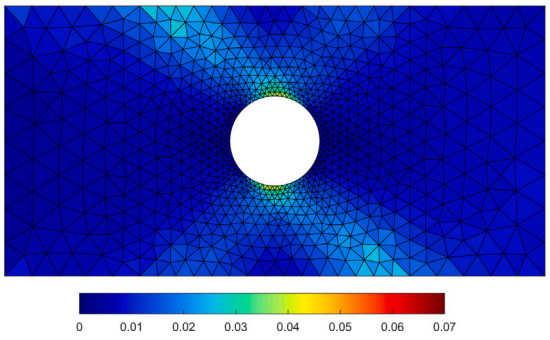
(b) Layer 1, solver-free ROH



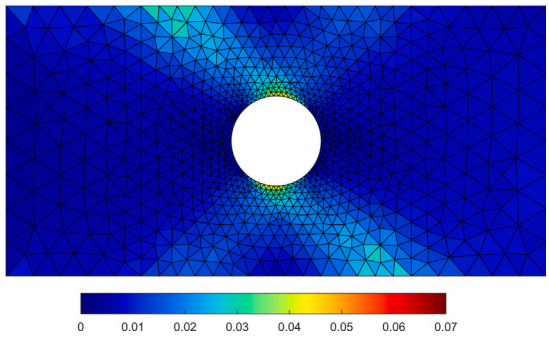
(c) Layer 2, CCH



(d) Layer 2, solver-free ROH



(e) Layer 3, CCH



(f) Layer 3, solver-free ROH

Fig. 15. Plate with hole: coarse-scale von Mises strain.

CRediT authorship contribution statement

Andrew Beel: Writing – original draft, Visualization, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Jacob Fish:** Writing – review & editing, Supervision, Resources, Project administration, Funding acquisition, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Research data are not shared.

Acknowledgment

This material is based upon work supported by the National Science Foundation under Grant No. 2225290.

References

[1] J. Fish, G.J. Wagner, S. Keten, Mesoscopic and multiscale modelling in materials, Nature Mater. 20 (2021) 774–786.
[2] K. Matouš, et al., A review of predictive nonlinear theories for multiscale modeling of heterogeneous materials, J. Comput. Phys. 330 (2017) 192–220.
[3] J.L. Ericksen, On the Cauchy–Born rule, Math. Mech. Solids 13 (2008) 199–220.
[4] W. Voigt, Ueber die Beziehung zwischen den beiden Elasticitätsconstanten isotroper Körper, Ann. Phys. 274 (12) (1889) 573–587.
[5] A. Reuss, Berechnung der Fliessgrenze von Mischkristallen auf Grund der Plastizitätsbedingung für Einkristalle, Z. Angew. Math. Mech. 9 (1929) 49–58.
[6] N.S. Bakhvalov, G.P. Panasenko, Homogenisation: Averaging Processes in Periodic Media: Mathematical Problems in the Mechanics of Composite Materials, Kluwer, 1989.
[7] E. Sanchez-Palencia, Non-Homogenous Media and Vibration Theory, Springer, 1980.
[8] T.I. Zohdi, P. Wriggers, An Introduction to Computational Micromechanics, Springer Science and Business Media, 2004.
[9] P.M. Suquet, Homogenization techniques for composite media, in: E. Sanchez-Palencia, A. Zaoui (Eds.), in: Lecture Notes in Physics, Vol. 272, Springer, 1985, pp. 193–279.
[10] G.J. Dvorak, Y. Benveniste, On transformation strains and uniform-fields in multiphase elastic media, Proc. R. Soc. Lond., Ser. A 437 (1992) 291–310.
[11] G.J. Dvorak, Transformation field analysis of inelastic composite materials, Proc. R. Soc. Lond., Ser. A 437 (1992) 311–327.
[12] C. Oskay, J. Fish, Eigendeformation-based reduced order homogenization, Comput. Methods Appl. Mech. Engrg. 196 (2007) 1216–1243.
[13] Z. Yuan, J. Fish, Multiple scale eigendeformation-based reduced order homogenization, Comput. Methods Appl. Mech. Engrg. 198 (2009) 2016–2038.
[14] R. Hill, Elastic properties of reinforced solids: some theoretical principles, J. Mech. Phys. Solids 11 (1963) 357–372.
[15] R.C. Tolman, The Principles of Statistical Mechanics, Clarendon, 1938.
[16] P.A.M. Dirac, The Principles of Quantum Mechanics, fourth ed., Clarendon, 1958.
[17] K.I. Elkhodary, M.S. Greene, S. Tang, T. Belytschko, W.K. Liu, Archetype-blending continuum theory, Comput. Methods Appl. Mech. Engrg. 254 (2013) 309–333.
[18] C.A. Duarte, I. Babuska, J.T. Oden, Generalized finite element methods for three-dimensional structural mechanics problems, Comput. Struct. 77 (2000) 215–232.
[19] T. Strouboulis, K. Copps, I. Babuska, The generalized finite element method, Comput. Methods Appl. Mech. Engrg. 190 (2001) 4081–4193.
[20] T.Y. Hou, X.H. Wu, A multiscale finite element method for elliptic problems in composite materials and porous media, J. Comput. Phys. 134 (1997) 169–189.
[21] W. E., B. Engquist, Z.Y. Huang, Heterogeneous multiscale method: a general methodology for multiscale modeling, Phys. Rev. B 67 (2003).
[22] W. E., B. Engquist, The heterogeneous multi-scale methods, Commun. Math. Sci. 1 (2003) 87–133.
[23] T.J.R. Hughes, Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods, Comput. Methods Appl. Mech. Engrg. 127 (1995) 387–401.
[24] T.J.R. Hughes, G.R. Feijóo, L. Mazzei, J.B. Quincy, The variational multiscale method—a paradigm for computational mechanics, Comput. Methods Appl. Mech. Engrg. 166 (1) (1998) 3–24.
[25] J. Fish, Z. Yuan, Multiscale enrichment based on partition of unity, Internat. J. Numer. Methods Engrg. 62 (2004) 1341–1359.
[26] W. Chen, J. Fish, A generalized space-time mathematical homogenization theory for bridging atomistic and continuum scales, Internat. J. Numer. Methods Engrg. 67 (2006) 253–271.
[27] J. Fish, W. Chen, R.G. Li, Generalized mathematical homogenization of atomistic media at finite temperatures in three dimensions, Comput. Methods Appl. Mech. Engrg. 196 (2007) 908–922.
[28] A.Q. Li, R.G. Li, J. Fish, Generalized mathematical homogenization: from theory to practice, Comput. Methods Appl. Mech. Engrg. 197 (2008) 3225–3248.
[29] R.P. Fedorenko, A relaxation method for solving elliptic difference equations, USSR Comput. Math. Phys. 1 (1962) 1092–1096.
[30] H.A. Schwarz, Ueber einige abbildungsaufgaben, J. Reine Angew. Math. (1869).
[31] C.D. Mote, Global-local finite element, Internat. J. Numer. Methods Engrg. 3 (1971) 565–574.
[32] T.I. Zohdi, J.T. Oden, G.J. Rodin, Hierachical modeling of heterogeneous bodies, Comput. Methods Appl. Mech. Engrg. 138 (1996) 273–298.
[33] M. Mosby, K. Matouš, Computational homogenization at extreme scales, Extreme Mech. Lett. 6 (2016) 68–74.
[34] J. Fish, V. Belsky, Multigrid method for periodic heterogeneous media, part I: convergence studies for one-dimensional case, Compute. Methods Appl. Mech. Eng. 126 (1995) 1–16.
[35] J. Fish, V. Belsky, Multigrid method for periodic heterogeneous media, part II: Multiscale modeling and quality control in multidimensional case, Compute. Methods Appl. Mech. Eng. 126 (1995) 17–38.
[36] C. Miehe, C.G. Bayreuther, On multiscale FE analyses of heterogeneous structures: from homogenization to multigrid solvers, Internat. J. Numer. Methods Engrg. 71 (2007) 1135–1180.
[37] S. Knapek, Matrix-dependent multigrid homogenization for diffusion problems, SIAM J. Sci. Comput. 20 (1998) 515–533.
[38] J.D. Moulton, J.E. Dendy, J.M. Hyman, The black box multigrid numerical homogenization algorithm, J. Comput. Phys. 142 (1998) 80–108.
[39] M.I. Idiart, N. Lahellec, P. Suquet, Model reduction by mean-field homogenization in viscoelastic composites. I. Primal theory, Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci. 476 (2020).
[40] M.I. Idiart, N. Lahellec, P. Suquet, Model reduction by mean-field homogenization in viscoelastic composites. II. Application to rigidly reinforced solids, Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci. 476 (2020).
[41] M. Yin, E. Zhang, Y. Yu, G.E. Karniadakis, Interfacing finite elements with deep neural operators for fast multiscale modeling of mechanics problems, Comput. Methods Appl. Mech. Engrg. 402 (2022).
[42] F. Aldakheel, E.S. Elsayed, T.I. Zohdi, P. Wriggers, Efficient multiscale modeling of heterogeneous materials using deep neural networks, Comput. Mech. (2023) 1–17.
[43] C. Harnish, L. Dalessandro, K. Matouš, D. Livescu, An adaptive wavelet method for nonlinear partial differential equations with applications to dynamic damage modeling, J. Comput. Phys. 479 (2023).
[44] C. Prakash, S. Ghosh, Self-consistent homogenization-based parametrically upscaled continuum damage mechanics model for composites subjected to high strain-rate loading, J. Compos. Mater. 57 (2023) 545–563.

[45] T. Kaiser, T. Raasch, J.J.C. Remmers, M.G.D. Geers, A wavelet-enhanced adaptive hierarchical FFT-based approach for the efficient solution of microscale boundary value problems, Comput. Methods Appl. Mech. Engrg. 409 (2023).
[46] D. Xia, C. Oskay, Reduced order mathematical homogenization method for polycrystalline microstructure with microstructurally small cracks, Internat. J. Numer. Methods Engrg. 124 (2023) 3166–3190.
[47] J. Fish, Y. Yu, Data-physics driven reduced order homogenization, Internat. J. Numer. Methods Engrg. (2022) 1–26.
[48] S.G. Mallat, A theory for multiresolution signal decomposition—the wavelet representation, IEEE Trans. Pattern Anal. Mach. Intell. 11 (1989) 674–693.
[49] G. Beylkin, N. Coult, A multiresolution strategy for reduction of elliptic PDEs and eigenvalue problems, Appl. Comput. Harmon. Anal. 5 (1998) 129–155.
[50] A.C. Gilbert, A comparison fo multiresolution and classical one-dimensional homogenization schemes, Appl. Comput. Harmon. Anal. 5 (1998) 1–35.
[51] S. Mahraeen, J.S. Chen, Wavelet Galerkin method in multi-scale homogenization of heterogeneous media, Internat. J. Numer. Methods Engrg. 66 (2006) 381–403.
[52] F. Feyel, J.L. Chaboche, FE2 multiscale approach for modelling the elastoviscoplastic behavior of long fiber SiC/Ti composite materials, Comput. Methods Appl. Mech. Engrg. 183 (2000) 309–330.
[53] A. Sridhar, V.G. Kouznetsova, M.G.D. Geers, Homogenization of locally resonant acoustic metamaterials towards an emergent enriched continuum, Comput. Mech. 57 (2016) 423–435.
[54] D.M. Kochmann, J.B. Hopkins, L. Valdevit, Multiscale modeling and optimization of the mechanics of hierarchical metamaterials, MRS Bull. 44 (2019) 773–781.
[55] R.D. Crouch, S.B. Clay, C. Oskay, Experimental and computational investigation of progressive damage accumulation in CFRP composites, Composites B 48 (2013) 59–67.
[56] S. Klinge, K. Hackl, Application of the multiscale FEM to the modeling of nonlinear composites with a random microstructure, Int. J. Multiscale Comput. Eng. 10 (2012) 213–227.
[57] I. Özdemir, W.A.M. Brekelmans, M.G.D. Geers, Computational homogenization for heat conduction in heterogeneous solids, Internat. J. Numer. Methods Engrg. 73 (2008) 185–204.
[58] Q. Yu, J. Fish, Multiscale asymptotic homogenization for multiphysics problems with multiple spatial and temporal scales: a coupled thermo-viscoelastic example problem, Int. J. Solids Struct. 39 (2002) 6429–6452.
[59] J. Fish, W. Wu, A nonintrusive stochastic multiscale solver, Internat. J. Numer. Methods Engrg. 88 (2011) 862–879.
[60] M. Kaminski, M. Kleiber, Perturbation based stochastic finite element method for homogenization of two-phase elastic composites, Comput. Struct. 78 (2000) 811–826.
[61] W. Chen, J. Fish, A dispersive model for wave propagation in periodic heterogeneous media based on homogenization with multiple spatial and temporal scales, J. Appl. Mech. 68 (2000) 153–161.
[62] V. Kouznetsova, W.A.M. Brekelmans, F.P.T. Baaijens, An approach to micro-macro modeling of heterogeneous materials, Comput. Mech. 27 (2001) 37–48.
[63] V. Kouznetsova, M.G.D. Geers, W.A.M. Brekelmans, Multi-scale constitutive modelling of heterogeneous materials with a gradient-enhanced computational homogenization scheme, Internat. J. Numer. Methods Engrg. 54 (2002) 1235–1260.
[64] S. Ghosh, S. Moorthy, Elastic-plastic analysis of arbitrary heterogeneous materials with the voronoi-cell finite element method, Comput. Methods Appl. Mech. Engrg. 121 (1995) 373–409.
[65] S. Ghosh, K. Lee, S. Moorthy, Two scale analysis of heterogeneous elasticplastic materials with asymptotic upscaling and Voronoi cell finite element model, Comput. Methods Appl. Mech. Engrg. 132 (1996) 63–116.
[66] J. Aboudi, A continuum theory for fiber-reinforced elastic-viscoplastic composites, Internat. J. Engrg. Sci. 20 (1982) 605–621.
[67] L. Berlyand, A. Kolpakov, Network approximation in the limit of small interparticle distance of the effective properties of a high-contrast random dispersed composite, Arch. Ration. Mech. Anal. 159 (2001) 179–227.
[68] H. Moulinec, P. Suquet, A numerical method for computing the overall response of nonlinear composites with complex microstructure, Comput. Methods Appl. Mech. Engrg. 157 (1998) 69–94.
[69] J.S. Chen, C. Pan, C.T. Wu, W.K. Liu, Reproducing kernel particle methods for large deformation analysis of nonlinear structures, Comput. Methods Appl. Mech. Engrg. 139 (1996) 195–229.
[70] M.A.A. Cavalcante, H. Khatam, M.J. Pindera, Homogenization of elastic-plastic periodic materials by FVDAM and FEM approaches, Composites B 42 (2011) 1713–1730.
[71] G.J. Dvorak, On uniform fields in heterogeneous media, Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci. 431 (1990) 89–110.
[72] J.C. Michel, P. Suquet, Computational analysis of nonlinear composite structures using the nonuniform transformation field analysis, Comput. Methods Appl. Mech. Engrg. 193 (2004) 5477–5502.
[73] F. Fritzen, T. Bohlke, Nonuniform transformation field analysis of materials with morphological anisotropy, Compos. Sci. Technol. 71 (2011) 433–442.
[74] M. Paley, J. Aboudi, Micromechanical analysis of composites by the generalized cells model, Mech. Mater. 14 (1992) 127–139.
[75] B.C. Moore, Principal component analysis in linear-systems—controllability, observability, and model-reduction, IEEE Trans. Automat. Control 26 (1981) 17–32.
[76] M. Green, A relative error bound for balanced stochastic truncation, IEEE Trans. Automat. Control 33 (1988) 961–965.
[77] K. Glover, All optimal Hankel-norm approximations of linear-multivariable systems and their L_∞ error bounds, Internat. J. Control 39 (1984) 1115–1193.
[78] P. Krysl, S. Lall, J.E. Marsden, Dimensional model reduction in non-linear finite element dynamics of solids and structures, Internat. J. Numer. Methods Engrg. 51 (2001) 479–504.
[79] J. Yvonnet, Q.C. He, The reduced model multiscale method (R3M) for the non-linear homogenization of hyperelastic media at finite strains, J. Comput. Phys. 223 (2007) 341–368.
[80] A. Radermacher, B.A. Bednarcyk, B. Stier, J. Simon, L. Zhou, S. Reese, Displacement-based multiscale modeling of fiber-reinforced composites by means of proper orthogonal decomposition, Adv. Model. Simul. Eng. Sci. 3 (2016).
[81] B.A. Le, J. Yvonnet, Q.C. He, Computational homogenization of nonlinear elastic materials using neural networks, Internat. J. Numer. Methods Engrg. 104 (2015) 1061–1084.
[82] S. Bhattacharjee, K. Matouš, A nonlinear manifold-based reduced order model for multiscale analysis of heterogeneous hyperelastic materials, J. Comput. Phys. 313 (2016) 635–653.
[83] J. Fish, Z.F. Yuan, R. Kumar, Computational certification under limited experiments, Internat. J. Numer. Methods Engrg. 114 (2018) 172–195.
[84] Y. Wang, S.W. Cheung, E.T. Chung, Y. Efendiev, M. Wang, Deep multiscale model learning, J. Comp. Phys. 406 (2020) 479–504.
[85] L. Dalcin, M. Mortensen, D.E. Keyes, Fast parallel multidimensional FFT using advanced MPI, J. Parallel Distrib. Comput. 128 (2019) 137–150.
[86] J. Fish, V. Filonova, Z. Yuan, Hybrid impotent-incompatible eigenstrain based homogenization, Internat. J. Numer. Methods Engrg. 95 (2013) 1–32.
[87] A. Beel, J. Fish, Solver-free classical computational homogenization for nonlinear periodic heterogeneous media, Internat. J. Numer. Methods Engrg. (2023) http://dx.doi.org/10.1002/nme.7390.
[88] R.E. Caflisch, Monte carlo and quasi-monte carlo methods, Acta Numer. 7 (1998) 1–49.
[89] M. Wicke, M. Botcsh, M. Gross, A finite element method on convex polyhedra, Eurographics 26 (2007) 1–10.
[90] G. Ventura, R. Gracie, T. Belytschko, Fast integration and weight function blending in the extended finite element method, Internat. J. Numer. Methods Engrg. 77 (2009) 1–29.
[91] S.E. Mousavi, H. Xiao, N. Sukumar, Generalized Gaussian quadrature rules on arbitrary polygons, Internat. J. Numer. Methods Engrg. 82 (2010) 99–113.

[92] S.E. Mousavi, N. Sukumar, Numerical integration of polynomials and discontinuous functions on irregular convex polygons and polyhedrons, Comput. Mech. 47 (2011) 535–554.
[93] Y. Sudhakar, W.A. Wall, Quadrature schemes for arbitrary convex/concave volumes and integration of weak form in enriched partition of unity methods, Comput. Methods Appl. Mech. Engrg. 258 (2013) 39–54.
[94] J. Fish, Practical Multiscaling, Wiley, 2013.
[95] Z. Yuan, R. Crouch, J. Wollschlager, A. Shojaei, J. Fish, Assessment of altair multiscale designer for damage tolerant design principles (DTDP) of advanced composite aircraft structures, J. Compos. Mater. 51 (10) (2016) 1379–1391.
[96] J.C. Michel, P. Suquet, Nonuniform transformation field analysis, Int. J. Solids Struct. 40 (2003) 6937–6955.
[97] J. Chaboche, S. Kruch, J. Maire, T. Pottier, Towards a micromechanics based inelastic and damage modeling of composites, Int. J. Plast. 17 (2001) 411–439.
[98] M.D. Vose, The Simple Genetic Algorithm: Foundations and Theory, MIT Press, 1999.