Computational Mechanics (2019) 64:351–364
https://doi.org/10.1007/s00466-019-01719-x

ORIGINAL PAPER

Fast calculation of interaction tensors in clustering-based homogenization

Lei Zhang[1] · Shaoqiang Tang[1] · Cheng Yu[2] · Xi Zhu[1] · Wing Kam Liu[2]

Received: 15 February 2019 / Accepted: 30 April 2019 / Published online: 22 May 2019

Abstract

Recently proposed clustering-based methods considerably reduce numerical cost for homogenizing heterogeneous materials, while maintaining the accuracy of mechanical property predictions in an online stage. In such an algorithm, however, the calculation of interaction tensors consumes much of the total computing time. We introduce a new method that expedites the interaction tensors calculation, thereby enhancing the clustering-based methods. We first cast a cubic/rectangular coarse grid over the representative volume element. Using analytical expressions for the integral of the Green's functions, we then calculate interaction tensors on the coarse grid. Finally, the desired interaction tensors on the clusters are approximated based on composition ratios. Moreover, in virtual clustering analysis, we derive the Lippmann–Schwinger equation for finite strain problems. Numerical tests in two and three space dimensions verify the efficiency and accuracy of the proposed method.

Keywords Homogenization · Virtual clustering analysis · Lippmann–Schwinger equation · Green's function · Interaction tensor · Finite strain

1 Introduction

Macroscopic behaviors of heterogeneous materials are dictated by the materials' microscopic fine structures. This lays a basis for material design. The goal of homogenization is to predict quickly and faithfully the macroscopic behavior, i.e., to relate the average stress to the corresponding average strain. Analytical micromechanical methods [1–7] have been proposed over the years. These methods are efficient because of their explicit analytical expressions. However, based on mean-field assumptions and linear relations without fully accounting for the detailed microscopic structure, they do not ensure the accuracy when dealing with complex microstructures and localized nonlinear material behaviors, such as plasticity. On the other hand, although finite element or meshfree methods make use of detailed microstructures and provide accurate results, their computational cost is usually immense.

To balance between cost and accuracy, reduced order models, such as the transformation field analysis (TFA) [8], the nonuniform transformation field analysis (NTFA) [9,10] and proper orthogonal decomposition (POD) [11] have been proposed. More recently, clustering-based homogenization methods [12–18] were developed to further improve efficiency. These methods have two stages: an offline training/learning stage and an online prediction/discovery stage. In the offline stage, material points are grouped into a few clusters based on high-fidelity direct numerical simulation (DNS) or experimental measurements, using the clustering-based data compression method. Then the interaction tensors among the clusters are calculated. The degrees of freedom (DoFs) in a typical application could be reduced from a few hundred million (which is often the case for image based direct numerical simulation) to only a few hundred. The online stage involves a highly reduced mechanical properties

✉ Shaoqiang Tang
maotang@pku.edu.cn

Lei Zhang
zhanglei2012@pku.edu.cn

Cheng Yu
chengyu2015@u.northwestern.edu

Xi Zhu
zhuxi@pku.edu.cn

Wing Kam Liu
w-liu@northwestern.edu

[1] HEDPS and LTCS, College of Engineering, Peking University, Beijing 100871, China

[2] Department of Mechanical Engineering, Northwestern University, 2145 Sheridan Rd., Evanston, IL 60208-3111, USA

prediction by solving the Lippmann–Schwinger equation, which is an integral equation equivalent to the equilibrium problem, assuming all variables (local stress, strain and related state variables) be uniform in each cluster. When nonlinearity presents, the interaction tensors precalculated in the offline stage are frequently used to assemble the Jacobian matrix for the Newton's iterative scheme in the online prediction stage. In virtual clustering analysis (VCA) [13], the calculation of the interaction tensors can be expensive, especially when the DNS mesh is fairly fine and a large number of clusters are used.

In this paper, we propose a new method to approximate interaction tensors with much reduced complexity. First, we cast a coarse grid over the domain under consideration and calculate "coarse-grained" interaction tensors using explicit formulas of Green's functions in physical space. Then we approximate interaction tensors for clusters based on their composition ratios in the coarse grid cells. Moreover, we derive the Lippmann–Schwinger equation for finite strain problems and improve VCA with fast calculated interaction tensors. We apply it to a simple hyper-elastic model for example. The fast method may also be used in Self-consistent Clustering Analysis (SCA) [12].

The rest of the paper is organized as follows. We introduce the problem setup and VCA in Sect. 2.1, and then show the details for calculating interaction tensors in two space dimensions. The fast calculating method is proposed in Sect. 2.3. Section 3 shows numerical results for different nonlinear materials, and discusses the accuracy and efficiency of the method. Some concluding remarks are drawn in Sect. 4.

2 Methodology

In this section, we sketch VCA algorithm first. Then we show the details for calculating interaction tensors in two space dimensions. Finally, we propose a fast method to compute it.

2.1 Virtual clustering analysis

To predict macroscopic properties of materials, high-fidelity representative volume elements (RVE) of the microstructure are commonly adopted in various methods such as finite element or meshfree methods. Although they make accurate predictions of the material behavior, an enormous, sometimes prohibitive amount of computing cost is induced for many practical applications where a stepwise loading process is under consideration. To alleviate the computing load, SCA [12] was proposed and afterwards VCA [13]. They are clustering-based reduced order modeling methods for numerical homogenization. In an offline stage, the RVE is decomposed into a number of clusters. In an online stage, based on an integral equation called as the Lippmann–Schwinger equation, the system can be solved on the reduced order (clustering-based) model. Here, we sketch VCA for illustration.

Consider an elastic material in domain $\Omega \subset \mathbb{R}^d$. The governing system reads

$$\nabla \cdot \boldsymbol{\sigma} = \mathbf{0}, \tag{1}$$

$$\boldsymbol{\varepsilon} = \frac{1}{2}\left(\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^{\mathrm{T}}\right), \tag{2}$$

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{\varepsilon}; \boldsymbol{x}, q), \tag{3}$$

where $\boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{\varepsilon}; \boldsymbol{x}, q)$ is a function of the strain ε, the position x and state variables q (e.g. the effective plastic strain). They define respectively the equilibrium Eq. (1), strain-displacement Eq. (2), and constitutive Eq. (3).

The goal of homogenization is to relate the average stress

$$\boldsymbol{\sigma}^{Macro} = \frac{1}{|\Omega|}\int_{\Omega} \boldsymbol{\sigma}(\boldsymbol{x})\mathrm{d}\boldsymbol{x} \tag{4}$$

with the corresponding average strain

$$\boldsymbol{\varepsilon}^{Macro} = \frac{1}{|\Omega|}\int_{\Omega} \boldsymbol{\varepsilon}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}. \tag{5}$$

Instead of using the periodic boundary conditions, we introduce a fictitious homogeneous isotropic material, also termed as comparison material, with stiffness tensor $\boldsymbol{C}^0$, surrounding the original domain. We load it with a uniform strain $\boldsymbol{\varepsilon}^0$ at infinity. That is to say,

$$\boldsymbol{\sigma} = \begin{cases} \boldsymbol{\sigma}(\boldsymbol{\varepsilon}; \boldsymbol{x}), & \boldsymbol{x} \in \Omega; \\ \boldsymbol{C}^0 : \boldsymbol{\varepsilon}, & \boldsymbol{x} \notin \Omega, \end{cases} \tag{6}$$

$$\lim_{\boldsymbol{x}\to\infty} \boldsymbol{\varepsilon}(\boldsymbol{x}) = \boldsymbol{\varepsilon}^0. \tag{7}$$

Due to homogeneity, the resulting stress at infinity is

$$\lim_{\boldsymbol{x}\to\infty} \boldsymbol{\sigma}(\boldsymbol{x}) = \boldsymbol{\sigma}^0 = \boldsymbol{C}^0 : \boldsymbol{\varepsilon}^0. \tag{8}$$

We rewrite Eq. (1) as

$$\nabla \cdot \left(\boldsymbol{C}^0 : \left(\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^0\right)\right) + \nabla \cdot \boldsymbol{\tau} = \mathbf{0}. \tag{9}$$

Here the polarization tensor $\boldsymbol{\tau} = \boldsymbol{\sigma} - \boldsymbol{C}^0 : \boldsymbol{\varepsilon}$ vanishes outside of Ω. A body force $\nabla \cdot \boldsymbol{\tau}$ acts on the comparison material within the domain Ω only.

By virtue of the Green's function and Betti's reciprocal work theorem over Ω, it can be shown that Eq. (9) leads to a Lippmann–Schwinger (integral) equation for $\boldsymbol{x} \in \Omega$ [13]

$$\boldsymbol{\varepsilon}(\boldsymbol{x}) - \boldsymbol{\varepsilon}^0 + \boldsymbol{\Phi} * \left(\boldsymbol{\sigma} - \boldsymbol{C}^0 : \boldsymbol{\varepsilon}\right)$$
$$= -\oint_{\partial\Omega} \boldsymbol{\Psi}(\boldsymbol{x} - \tilde{\boldsymbol{x}}) \cdot (\boldsymbol{n} \cdot (\boldsymbol{\sigma}(\tilde{\boldsymbol{x}}) - \boldsymbol{C}^0 : \boldsymbol{\varepsilon}^0))\mathrm{d}S$$

$$-\oint_{\partial\Omega}((\boldsymbol{\Phi}(\boldsymbol{x}-\tilde{\boldsymbol{x}}):\boldsymbol{C}^0)\cdot\boldsymbol{n})\cdot(\boldsymbol{u}(\tilde{\boldsymbol{x}})-\boldsymbol{u}^0(\tilde{\boldsymbol{x}}))\mathrm{d}S, \tag{10}$$

where $*$ denotes convolution by $\boldsymbol{f} * \boldsymbol{g} = \int_\Omega \boldsymbol{f}(\boldsymbol{x}-\tilde{\boldsymbol{x}}) : \boldsymbol{g}(\tilde{\boldsymbol{x}})\mathrm{d}\tilde{\boldsymbol{x}}$. Here the kernel functions are

$$\Phi_{ijkl} = \frac{1}{2}\left(\frac{\partial}{\partial x_j}\varepsilon_{kl}^{(i)} + \frac{\partial}{\partial x_i}\varepsilon_{kl}^{(j)}\right), \tag{11}$$

$$\Psi_{ijk} = \frac{1}{2}\left(\frac{\partial}{\partial x_j}u_k^{(i)} + \frac{\partial}{\partial x_i}u_k^{(j)}\right), \tag{12}$$

where $\boldsymbol{\varepsilon}^{(i)}$ and $\boldsymbol{u}^{(i)}$ are the fundamental solutions under a concentrated unit force in x^i direction. Their detailed expressions in two space dimensions will be given in the next subsection. The Lippmann–Schwinger equation (10) is equivalent to the original governing system (1)–(3).

If we replace Ω by a suitably enlarged domain $\tilde{\Omega} \supset \Omega$, Saint-Venant's principle shows that the effect of the boundaries of the enlarged domain on stress and strain response in Ω can be neglected. Since the polarization tensor $\boldsymbol{\tau}$ equals to zero in $\tilde{\Omega}\backslash\Omega$, Eq. (10) reduces to a boundary-term-free Lippmann–Schwinger equation and the convolution is performed only over Ω:

$$\boldsymbol{\varepsilon}(\boldsymbol{x}) - \boldsymbol{\varepsilon}^0 + \boldsymbol{\Phi} * \left(\boldsymbol{\sigma} - \boldsymbol{C}^0 : \boldsymbol{\varepsilon}\right) = \boldsymbol{0}. \tag{13}$$

In the offline stage, the high-fidelity RVE is represented by n voxels or elements denoted as $\omega_i, i = 1, \cdots, n$. These voxels are grouped into k clusters according to their responses under several selected loadings. These responses are obtained from DNS results. The clustering is performed by a machine learning technique such as k-means or Self-Organizing Map (SOM). The clustering results are collected in index sets $\mathcal{I}(\Omega^I) = \{i|\omega_i \subset \Omega^I, i = 1, 2, \cdots, n\}$ for $I = 1, 2, \cdots, k$.

In the online stage, we assume the response (strain) at material points of the same cluster to be the same and denote the strain in cluster Ω^I as $\boldsymbol{\varepsilon}^I$. The corresponding average stress in this domain can be calculated from

$$\boldsymbol{\sigma}^I = \frac{1}{\left|\Omega^I\right|}\int_{\Omega^I}\boldsymbol{\sigma}(\boldsymbol{\varepsilon}^I;\boldsymbol{x})d\boldsymbol{x}, \quad I = 1, 2, \ldots, k, \tag{14}$$

where $|\cdot|$ denotes the volume. Approximating the stress by its average, we integrate (13) over Ω^I and obtain the discrete form

$$\boldsymbol{\varepsilon}^I - \boldsymbol{\varepsilon}^0 + \sum_{J=1}^{k}\boldsymbol{D}^{IJ} : \left(\boldsymbol{\sigma}^J - \boldsymbol{C}^0 : \boldsymbol{\varepsilon}^J\right) = \boldsymbol{0}, \quad I = 1, 2, \ldots, k, \tag{15}$$

with the interaction tensor

$$\boldsymbol{D}^{IJ} = \frac{1}{\left|\Omega^I\right|}\int_{\Omega^I}\int_{\Omega^J}\boldsymbol{\Phi}(\boldsymbol{x}-\tilde{\boldsymbol{x}})d\tilde{\boldsymbol{x}}d\boldsymbol{x}. \tag{16}$$

We remark that $\boldsymbol{D}^{IJ}$ depends only on the clusters for given reference material. Hence, $\boldsymbol{D}^{IJ}$ can be precomputed in the offline stage.

If the microscopic constitutive relation is linear, (15) is a linear algebraic system for $\boldsymbol{\varepsilon}^I$. In general, a nonlinear constitutive relation presents, such as in plasticity. The algebraic system is then nonlinear, and may be solved by the Newton-Raphson method.

After obtaining stress and strain in each cluster, we compute the averages.

$$\boldsymbol{\sigma}^{Macro} = \frac{1}{|\Omega|}\sum_{I=1}^{k}\left|\Omega^I\right|\boldsymbol{\sigma}^I, \tag{17}$$

$$\boldsymbol{\varepsilon}^{Macro} = \frac{1}{|\Omega|}\sum_{I=1}^{k}\left|\Omega^I\right|\boldsymbol{\varepsilon}^I. \tag{18}$$

2.2 Calculation of $\boldsymbol{D}^{IJ}$ in two and three space dimensions

This subsection presents the analytical expressions and numerical evaluation of $\boldsymbol{D}^{IJ}$ in two space dimensions. Three dimensional results are shown in "Appendix B".

The fundamental solution to

$$\nabla\cdot(\boldsymbol{C}^0 : \boldsymbol{\varepsilon}) = \begin{pmatrix}\delta(\boldsymbol{x})\\ 0\end{pmatrix} \tag{19}$$

is $\boldsymbol{u}^{(1)}$ for displacement, and $\boldsymbol{\varepsilon}^{(1)}$ for corresponding strain. That to

$$\nabla\cdot(\boldsymbol{C}^0 : \boldsymbol{\varepsilon}) = \begin{pmatrix}0\\ \delta(\boldsymbol{x})\end{pmatrix} \tag{20}$$

is $\boldsymbol{u}^{(2)}$ for displacement, and $\boldsymbol{\varepsilon}^{(2)}$ for corresponding strain.

Taking Fourier transform $(x_1, x_2) \to (\xi, \eta)$, we solve (19) and (20), and obtain

$$\mathcal{F}\boldsymbol{u}^{(1)} = \frac{-\phi_0}{(\xi^2+\eta^2)^2}\begin{bmatrix}\mu_0\xi^2 + (\lambda_0+2\mu_0)\eta^2\\ -(\lambda_0+\mu_0)\xi\eta\end{bmatrix}, \tag{21}$$

$$\mathcal{F}\boldsymbol{u}^{(2)} = \frac{-\phi_0}{(\xi^2+\eta^2)^2}\begin{bmatrix}-(\lambda_0+\mu_0)\xi\eta\\ (\lambda_0+2\mu_0)\xi^2 + \mu_0\eta^2\end{bmatrix} \tag{22}$$

for displacement, and

$$\mathcal{F}\boldsymbol{\varepsilon}^{(1)} = \frac{-i\phi_0}{(\xi^2+\eta^2)^2}$$
$$\begin{bmatrix} \xi(\mu_0\xi^2 + (\lambda_0+2\mu_0)\eta^2) & \frac{\eta}{2}(-\lambda_0\xi^2 + (\lambda_0+2\mu_0)\eta^2) \\ \frac{\eta}{2}(-\lambda_0\xi^2 + (\lambda_0+2\mu_0)\eta^2) & -(\lambda_0+\mu_0)\xi\eta^2 \end{bmatrix}, \tag{23}$$

$$\mathcal{F}\boldsymbol{\varepsilon}^{(2)} = \frac{-i\phi_0}{(\xi^2+\eta^2)^2}$$
$$\begin{bmatrix} -(\lambda_0+\mu_0)\xi^2\eta & \frac{\xi}{2}(-\lambda_0\eta^2 + (\lambda_0+2\mu_0)\xi^2) \\ \frac{\xi}{2}(-\lambda_0\eta^2 + (\lambda_0+2\mu_0)\xi^2) & \eta((\lambda_0+2\mu_0)\xi^2 + \mu_0\eta^2) \end{bmatrix} \tag{24}$$

for strain. Here λ_0, μ_0 are the Lamé coefficients of the reference stiffness $\boldsymbol{C}^0$, and $\phi_0 = \frac{1}{(\lambda_0+2\mu_0)\mu_0}$.

We notice that the fundamental solution to the biharmonic equation

$$f = \frac{1}{16\pi}(x_1^2+x_2^2)\ln(x_1^2+x_2^2) \tag{25}$$

takes a Fourier transform $\frac{1}{(\xi^2+\eta^2)^2}$. It is used to express

$$\boldsymbol{u}^{(1)} = -\phi_0 \begin{bmatrix} \mu_0 f_{11} + (\lambda_0+2\mu_0)f_{22} \\ -(\lambda_0+\mu_0)f_{12} \end{bmatrix}, \tag{26}$$

$$\boldsymbol{u}^{(2)} = -\phi_0 \begin{bmatrix} -(\lambda_0+\mu_0)f_{12} \\ (\lambda_0+2\mu_0)f_{11} + \mu_0 f_{22} \end{bmatrix} \tag{27}$$

by its derivatives $f_{ij}, i, j = 1, 2$. As a matter of fact, they are the Kelvin's solutions. Thus entries in the tensor $\boldsymbol{\Phi}$ are

$$\boldsymbol{\Phi}_{11} = \phi_0 \begin{bmatrix} \mu_0 f_{1111} + (\lambda_0+2\mu_0)f_{1122} & \frac{1}{2}(-\lambda_0 f_{1112} + (\lambda_0+2\mu_0)f_{1222}) \\ \frac{1}{2}(-\lambda_0 f_{1112} + (\lambda_0+2\mu_0)f_{1222}) & -(\lambda_0+\mu_0)f_{1122} \end{bmatrix}, \tag{28}$$

$$\boldsymbol{\Phi}_{22} = \phi_0 \begin{bmatrix} -(\lambda_0+\mu_0)f_{1122} & \frac{1}{2}(-\lambda_0 f_{1222} + (\lambda_0+2\mu_0)f_{1112}) \\ \frac{1}{2}(-\lambda_0 f_{1222} + (\lambda_0+2\mu_0)f_{1112}) & (\lambda_0+2\mu_0)f_{1122} + \mu_0 f_{2222} \end{bmatrix}, \tag{29}$$

$$\boldsymbol{\Phi}_{12} = \boldsymbol{\Phi}_{21} = \frac{\phi_0}{2} \begin{bmatrix} -\lambda_0 f_{1112} + (\lambda_0+2\mu_0)f_{1222} & -\lambda_0 f_{1122} + \frac{1}{2}(\lambda_0+2\mu_0)(f_{1111}+f_{2222}) \\ -\lambda_0 f_{1122} + \frac{1}{2}(\lambda_0+2\mu_0)(f_{1111}+f_{2222}) & -\lambda_0 f_{1222} + (\lambda_0+2\mu_0)f_{1112}) \end{bmatrix}. \tag{30}$$

The involved fourth-order derivatives are

$$f_{1111} = \frac{-x_1^4 - 6x_1^2x_2^2 + 3x_2^4}{4\pi(x_1^2+x_2^2)^3}, \tag{31}$$

$$f_{1112} = \frac{x_1^3x_2 - 3x_1x_2^3}{2\pi(x_1^2+x_2^2)^3}, \tag{32}$$

$$f_{1122} = \frac{-x_1^4 + 6x_1^2x_2^2 - x_2^4}{4\pi(x_1^2+x_2^2)^3}, \tag{33}$$

$$f_{1222} = \frac{-3x_1^3x_2 + x_1x_2^3}{2\pi(x_1^2+x_2^2)^3}, \tag{34}$$

$$f_{2222} = \frac{3x_1^4 - 6x_1^2x_2^2 - x_2^4}{4\pi(x_1^2+x_2^2)^3}. \tag{35}$$

They are homogeneous functions, singular at the origin. We have indefinite integrals

$$\begin{aligned} &\iint f_{1111}(x_1-\tilde{x}_1, x_2-\tilde{x}_2)d\boldsymbol{x}d\tilde{\boldsymbol{x}} \\ &\quad= \frac{1}{16\pi}\Big((x_2-\tilde{x}_2)^2\ln((x_1-\tilde{x}_1)^2+(x_2-\tilde{x}_2)^2) \\ &\quad -3(x_1-\tilde{x}_1)^2\ln((x_1-\tilde{x}_1)^2+(x_2-\tilde{x}_2)^2) \\ &\quad +8(x_1-\tilde{x}_1)(x_2-\tilde{x}_2)\arctan\left(\frac{x_2-\tilde{x}_2}{x_1-\tilde{x}_1}\right)\Big), \end{aligned} \tag{36}$$

$$\begin{aligned} &\iint f_{1112}(x_1-\tilde{x}_1, x_2-\tilde{x}_2)d\boldsymbol{x}d\tilde{\boldsymbol{x}} \\ &\quad= \frac{1}{8\pi}((x_1-\tilde{x}_1)(x_2-\tilde{x}_2)\ln((x_1-\tilde{x}_1)^2+(x_2-\tilde{x}_2)^2) \\ &\quad -2(x_1-\tilde{x}_1)^2\arctan\left(\frac{-x_2+\tilde{x}_2}{x_1-\tilde{x}_1}\right) \\ &\quad -(x_1-\tilde{x}_1)(x_2-\tilde{x}_2)), \end{aligned} \tag{37}$$

$$\begin{aligned} &\iint f_{1122}(x_1-\tilde{x}_1, x_2-\tilde{x}_2)d\boldsymbol{x}d\tilde{\boldsymbol{x}} \\ &\quad= \frac{1}{16\pi}((x_1-\tilde{x}_1)^2+(x_2-\tilde{x}_2)^2)\ln((x_1-\tilde{x}_1)^2 \\ &\quad +(x_2-\tilde{x}_2)^2). \end{aligned} \tag{38}$$

Other terms are obtained by rotating x_1 and x_2. It is noticed that the indefinite integrals (36)–(38) are singular for $x_1 = \tilde{x}_1$ or $x_2 = \tilde{x}_2$. When calculating interaction tensors, we take a numerical limit along (1,1) direction.

With these, we are ready to compute all terms in the interaction tensors $\boldsymbol{D}^{IJ}$, provided the clusters are meshed with voxels/pixels.

For independent subdomains A and B of Ω, we denote

$$\mathcal{S}(A, B) = \int_A \int_B \boldsymbol{\Phi}(\boldsymbol{x}-\tilde{\boldsymbol{x}})\mathrm{d}\tilde{\boldsymbol{x}}\mathrm{d}\boldsymbol{x}. \tag{39}$$

The interaction tensors are then

$$
\begin{aligned}
\boldsymbol{D}^{IJ} &= \frac{1}{|\Omega^I|}\mathcal{S}(\Omega^I, \Omega^J)\\
&= \frac{1}{|\Omega^I|}\sum_{i\in\mathcal{I}(\Omega^I)}\sum_{j\in\mathcal{I}(\Omega^J)}\mathcal{S}(\omega^i, \omega^j), \quad I, J = 1, 2, \cdots, k.
\end{aligned}
\tag{40}
$$

That is, we sum up all the interaction terms between a voxel/pixel in Ω^I and a voxel/pixel in Ω^J.

2.3 Fast calculation of $\boldsymbol{D}^{IJ}$

We propose the following technique to compute interaction tensors $\boldsymbol{D}^{IJ}$ more efficiently.

First, we introduce a cubic/rectangular domain Q to cover the RVE Ω. See Fig. 1. Q is partitioned into m small cubes/rectangles Q_p, $p = 1, 2, \cdots, m$. They form a coarse grid. Here, m is usually chosen much smaller than the number of voxels/pixels n.

Second, we calculate $\mathcal{S}(Q_p, Q_q)$ with the indefinite integrals given in 2.2. Then we use $\mathcal{S}(Q_p, Q_q)$ to approximate $\mathcal{S}(\omega_i, \omega_j)$ for $\omega_i \subset Q_p, \omega_j \subset Q_q$. That is,

$$
\mathcal{S}(\omega_i, \omega_j) \approx \begin{cases} \frac{|\omega_i|}{|Q_p|}\frac{|\omega_j|}{|Q_q|}\mathcal{S}(Q_p, Q_q), & \omega_i \subset Q_p, \omega_j \subset Q_q, p \neq q;\\ \frac{|\omega_i|}{|Q_p|}\delta_{ij}\mathcal{S}(Q_p, Q_p), & \omega_i, \omega_j \subset Q_p, \end{cases}
\tag{41}
$$

where δ_{ij} is the Kronecker delta. The approximation is motivated as follows. If ω_i and ω_j belong to different coarse grid cells, the double integral is approximately proportional to the product of $|\omega_i|$ and $|\omega_j|$, because the kernel function $\boldsymbol{\Phi}(\boldsymbol{x} - \tilde{\boldsymbol{x}})$ is approximately constant over both coarse grid cells. Otherwise, for ω_i and ω_j both belonging to Q_p, singularity of the integrand at the origin $\boldsymbol{x} - \tilde{\boldsymbol{x}} = 0$ requires more delicate treatment. Since the integrand is a homogeneous function, the integral is proportional to the volume rather than the square of the volume. Thus in this case, we use $\frac{|\omega_i|}{|Q_p|}\mathcal{S}(Q_p, Q_p)$ to approximate $\mathcal{S}(\omega_i, \omega_i)$, and enforce $\mathcal{S}(\omega_i, \omega_j) = 0$ for $i \neq j$ to conserve the relation

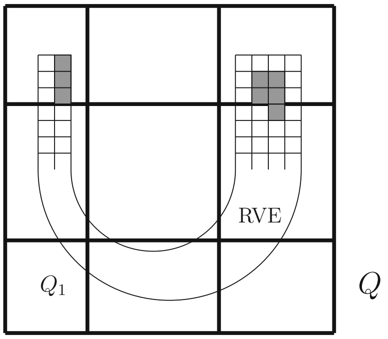


Fig. 1 A schematic plot of pixels, clusters, coarse grid and RVE in two space dimensions. The RVE Ω is represented by an arbitrarily-shaped domain (here U-shaped for illustration). It is covered by a coarse grid over a rectangle Q with cells Q_p (rectangles with edges shown by heavy lines). The RVE is also split into pixels (small rectangles with edge shown by light lines). For clarity, we only show part of the pixels. The pixels are categorized into clusters, where the gray ones belong to Ω^1, as an example

$$
\sum_{\omega_i\subset Q_p}\sum_{\omega_j\subset Q_p}\mathcal{S}(\omega_i, \omega_j) = \mathcal{S}(Q_p\bigcap\Omega, Q_p\bigcap\Omega). \tag{42}
$$

According to Eqs. (40) and (41), we obtain

$$
\begin{aligned}
\boldsymbol{D}^{IJ} \approx\ & \frac{1}{|\Omega^I|}\sum_{p=1, q\neq p}^{m}\sum_{q=1}^{m}\alpha_p^I\alpha_q^J\mathcal{S}(Q_p, Q_q)\\
&+\frac{1}{|\Omega^I|}\sum_{p=1}^{m}\alpha_p^I\delta_{IJ}\mathcal{S}(Q_p, Q_p).
\end{aligned}
\tag{43}
$$

Here, we use

$$
\alpha_p^I = \sum_{\substack{i\in\mathcal{I}(\Omega^I)\\ \omega_i\subset Q_p}}\frac{|\omega_i|}{|Q_p|} = \frac{|\Omega^I\bigcap Q_p|}{|Q_p|} \tag{44}
$$

to stand for the volume fraction of the cluster Ω^I in the coarse grid cell Q_p, which is called as composition ratio.

It is noticed that even if a voxel/pixel ω_i is not fully contained in a coarse grid cell Q_p, its contribution to $\boldsymbol{D}^{IJ}$ still counts according to the volume ratio. Therefore the above approximation (43) is still valid. This approximation only depends on the integral $\mathcal{S}(Q_p, Q_q)$ over the coarse grid and the volume fraction α_p^I in each cell.

It is remarked that we usually make an equidistant partition in each space dimension. The kernel function $\boldsymbol{\Phi}(\boldsymbol{x} - \tilde{\boldsymbol{x}})$ depends on the relative position between x and $\tilde{x}$. Hence, the integral $\mathcal{S}(Q_p, Q_q)$ also only depends on the relative position of Q_p and Q_q. Thus we can precompute the integral $\mathcal{S}(Q_p, Q_q)$ and restore it in the set $\mathbf{M} = \{\mathcal{S}(Q_p, Q_q)|p, q = 1, 2, \cdots, m\}$. The number of non-repetitive elements in the set $\mathbf{M}$ is less than $2^d m$, so the computational complexity of $\mathbf{M}$ is only of order $O(m)$.

The calculation of interaction tensors $\boldsymbol{D}^{IJ}$ is summarized in the following chart:

```
1. Draw a coarse grid over the RVE $\Omega$ with $m$ coarse grid cells $Q_p$, $p = 1, \cdots, m$.
2. Compute the volume proportion $\alpha^I_p$ of the cluster $\Omega^I$ in the coarse grid cell $Q_p$.
3. Compute $\boldsymbol{D}^{IJ}$ according to (43):
   (a) Set $\boldsymbol{D}^{IJ} = 0$;
   (b) Loop $p$ and $q$ from 1 to $m$:
       a. if $p \neq q$, add $\alpha^I_p \alpha^J_q \mathcal{S}(Q_p, Q_q)$ to $\boldsymbol{D}^{IJ}$;
       b. if $p = q$, add $\alpha^I_p \delta_{IJ} \mathcal{S}(Q_p, Q_p)$ to $\boldsymbol{D}^{IJ}$.
```

2.4 A comparison of the complexity for different method

In 2.2, the calculation of $\boldsymbol{D}^{IJ}$ based on voxels/pixels leads to numerical cost of the order $O(n^2)$. In 2.3, we construct a coarse grid to replace the voxels/pixels and then propose the fast calculating method based on composition ratios. With this approximation, the cost for calculating $\boldsymbol{D}^{IJ}$ is only of the order $O(k^2m^2)$, which is far less than the original method. Furthermore, The complexity of the computation decreases when we choose bigger cells for the coarse grid. Moreover, it is remarked that the complexity of the fast Fourier transform method (FFT) used in SCA is of the order $O(k^2 n \log(n))$. The comparison of the complexity for different methods is summarized in Table. 1.

3 Numerical tests

In this section, we apply VCA improved with the fast method for interaction tensors calculation to both elasto-plastic materials and hyperelastic materials.

3.1 Elasto-plastic materials

We apply VCA to 2D plane strain and 3D heterogeneous materials under different loading conditions to illustrate the efficiency of fast calculation of $\boldsymbol{D}^{IJ}$.

For both 2D and 3D problems considered, the RVE Ω is a two-phase material consisting of inclusion (phase 2) and matrix (phase 1). To predict the plastic behavior, we use a simple material law: J_2 *plasticity* with piece-wise linear isotropic hardening. The plasticity law for the matrix material (phase 1) considers a von Mises yield surface

$$\bar{\sigma} - \sigma_Y(\bar{\varepsilon}) = 0, \tag{45}$$

where $\bar{\sigma}$ is the von Mises equivalent stress and $\bar{\varepsilon}$ is the equivalent plastic strain.

3.1.1 Two dimensional elasto-plastic materials

In the 2-D plane strain test, the volume fraction of the fibers is 50% as shown in Fig. 2. The Young's moduli and Poisson's

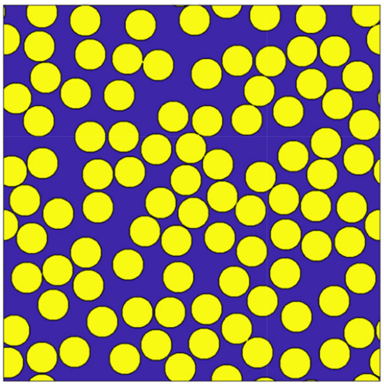


Fig. 2 Geometry for two-phase composite material in the 2D plane strain example, where the color blue represents the matrix and the color yellow represents the inclusion. (Color figure online)

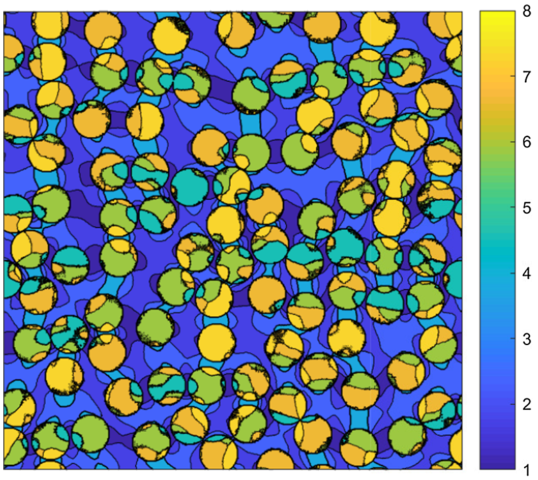


Fig. 3 Clustering distribution for 4 clusters in the matrix and 4 clusters in the inclusion

Table 1 A comparison of the complexity for different methods

Method	FFT	The original method	The fast method
Complexity	$O(k^2 n \log n)$	$O(n^2)$	$O(k^2 m^2)$

n is the number of voxels/pixels. m is the number of coarse grid cells. k is the number of clusters ratios of the matrix (phase 1) and inclusion (phase 2) are:

$$\begin{aligned} &E_1 = 100 \text{ MPa}, \nu_1 = 0.3, && \text{in phase 1;} \\ &E_2 = 500 \text{ MPa}, \nu_2 = 0.19, && \text{in phase 2.} \end{aligned}$$

The yield stress σ_Y is given by a piece-wise linear hardening law that depends on $\bar{\varepsilon}$

$$\sigma_Y(\bar{\varepsilon}) = \begin{cases} 0.5 + 5\bar{\varepsilon}, & \bar{\varepsilon} \in [0, 0.04); \\ 0.62 + 2\bar{\varepsilon}, & \bar{\varepsilon} \in [0.04, \infty). \end{cases} \tag{46}$$

The RVE Ω is represented by 600×600 elements (pixels) with finite element method. A fine scale DNS is performed for the RVE under a far-field strain from 0 to 0.05 in x_1 direction, while the strain is enforced to zero in other directions. The response of each pixel at load 0.001 is restored as the database. In an offline stage, the DNS mesh is compressed into $k_1 (k_1 = 4, 8, 16)$ clusters in the matrix and k_2 clusters in the inclusion by SOM. We take $k_2 = k_1$ as the volume fraction is 50%. The cluster distribution for $k_1 = 4$ is shown in Fig. 3.

We apply SOM in each phase based on two reasons. First, we get a discrete form of the Lippmann–Schwinger equation by approximating the stress with its average, and the stress fluctuations need to be small for better approximation. Second, the plasticity is only considered for the matrix material in elasto-plastic examples, thus phases are needed to be identified in the computational process.

In fact, clustering method can also work without consideration of the phases in general, because the strain responses in each phase perform differently according to different material properties. In [13], we consider a field-character (or checkboard) problem and make clustering without considering the phases. The boundaries of clusters fit almost those of phases.

Once the clustering is performed, the interaction tensors $\boldsymbol{D}^{IJ}$ are computed. The computational time for different coarse grid sizes is listed in Table 2. The domain is evenly partitioned into m coarse grid cells.

The selection of the reference material influences the predictions [13], which will be further explored in a future work. Here, we choose the reference material with the Young's modulus $E_0 = 20$ MPa and Poisson's ratio $\nu_0 = 0.4$.

The predictions of the online $\bar{\sigma}^{Macro} - \varepsilon_{11}^{Macro}$ for $m = 200^2$ in Fig. 4 show the nonlinearity in our numerical examples. The $\sigma_{11}^{Macro} - \varepsilon_{11}^{Macro}$ curves for $m = 200^2$ are shown in Fig. 5. The solid line shows the DNS results for comparison.

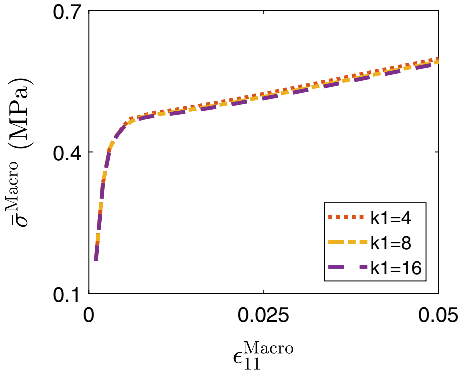


Fig. 4 $\bar{\sigma}^{Macro} - \varepsilon_{11}^{Macro}$ curves for $m = 200^2$. $\bar{\sigma}^{Macro}$ is the von Mises equivalent stress. These curves show strong nonlinearity of the material

The numerical error of VCA compared to DNS is reduced when increasing the number of clusters both in elastic region and plastic region. The error under the loading $\varepsilon_{11} = 0.05$ is less than 1.5% when 16 clusters are used in the matrix.

The $\sigma_{11}^{Macro} - \varepsilon_{11}^{Macro}$ curves for different coarse grid sizes $m = 200^2, 120^2, 60^2, 30^2$ are plotted in Fig. 6 to illustrate the accuracy of our method in the calculation of $\boldsymbol{D}^{IJ}$ when taking 8 clusters in the matrix. The fast method makes numerical responses harder and the error increases when taking less coarse grid cells. Table 2 manifests the efficiency. We use an Intel Xeon e5-2650 v2 processor and parallel pool (16 workers) in MATLAB. For $m = 200^2$, it takes about 3354 s for $k = 8$ in the calculation of $\boldsymbol{D}^{IJ}$, while for $m = 30^2$, it only takes about 3 s. The results for $m = 30^2$ are satisfactory, for the error maitains less than 2% with high efficiency. Figure 7 shows that the computational time is roughly proportional to the square of the coarse grid cell number m.

3.1.2 Three dimensional elasto-plastic materials

Similar to 2D plane strain test, our method applies to 3D nonlinear elasto-plastic materials. The volume fraction of the inclusion particle phase is about 15.58% as shown in Fig. 8. The Young's moduli and Poisson's ratios of the matrix (phase 1) and inclusion (phase 2) are:

$$\begin{aligned} &E_1 = 3800 \text{ MPa}, \nu_1 = 0.387, && \text{in phase 1;} \\ &E_2 = 38000 \text{ MPa}, \nu_2 = 0.387, && \text{in phase 2.} \end{aligned}$$

The matrix has a von Mises yield surface and a piece-wise hardening law depending on the effective plastic strain $\bar{\varepsilon}$

Table 2 Computational time for interaction tensors and deviations of reference solution under the loading $\varepsilon_{11} = 0.05$ for $k_1 = 8$

m	200^2	120^2	60^2	30^2
CPU time (s)	3353.72	367.97	28.83	3.12
Speed-up	–	9.11	116.33	1074.91
Deviations from DNS results (%)	1.59	1.74	1.95	1.98

Fig. 5 $\sigma_{11}^{Macro} - \varepsilon_{11}^{Macro}$ curves for $m = 200^2$: The reference solution plotted by solid line

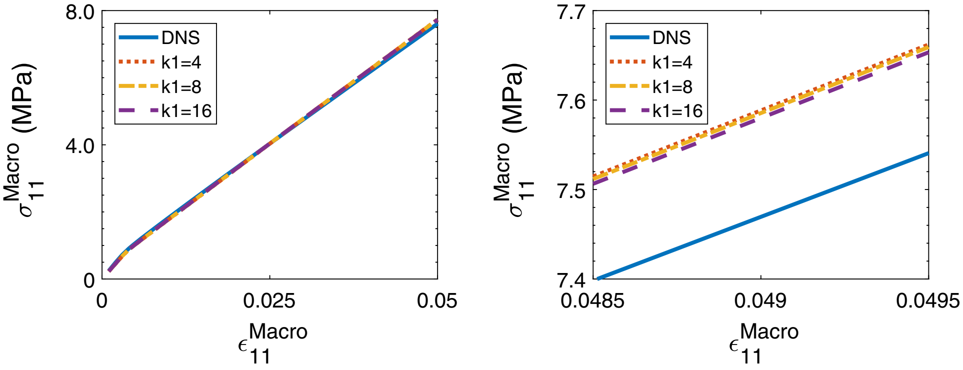
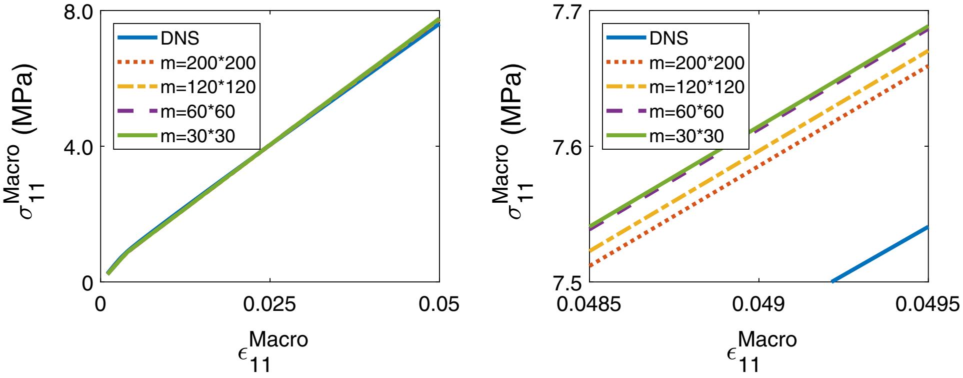


Fig. 6 $\sigma_{11}^{Macro} - \varepsilon_{11}^{Macro}$ curves for different coarse grid cell number m when taking $k_1 = 8$



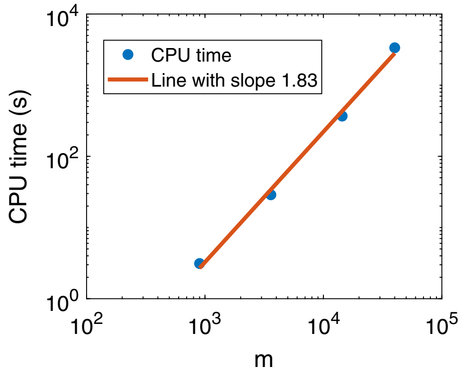


Fig. 7 The computational time for calculating interaction tensors versus the number of coarse grid cells m for $k_1 = 8$. The slope of the line is about 1.83

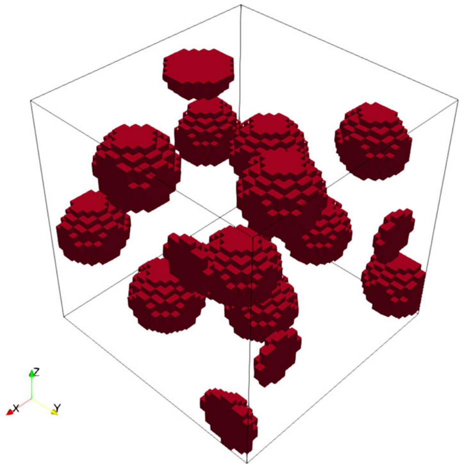


Fig. 8 The $41 \times 41 \times 41$ mesh of the 3D material displayed in the inclusion phase (phase 2)

$$\sigma_Y(\bar{\varepsilon}) = \begin{cases} 50 + 250\bar{\varepsilon}, & \bar{\varepsilon} \in [0, 0.04); \\ 56 + 100\bar{\varepsilon}, & \bar{\varepsilon} \in [0.04, \infty). \end{cases} \tag{47}$$

We take $k_1 = 4, 8, 16, k_2 = k_1/4$. Reference solution is obtained from [19] with a $41 \times 41 \times 41$ mesh. In the online stage, the reference material takes Young's modulus 1000 MPa and the Poisson's ratio ν 0.387.

The $\sigma_{11}^{Macro} - \varepsilon_{11}^{Macro}$ curves predicted by VCA with various number of clusters are depicted in Fig. 9 when taking $m = 21^3$. The predictions converge to the reference solution when the number of clusters k increases. The deviations from the DNS results and the computational time when computing $\boldsymbol{D}^{IJ}$ are shown in Table 3. The reduction of the computational cost is significant when the coarse grid cell number m decreases. The coarse grid for $m = 60^2, k_1 = 8$ in previous 2D example takes a CPU time about 28.83 s. The coarse grid for $m = 14^3, k_1 = 8$ in 3D case takes a CPU time about 137.32 s. Both the number of the coarse grid cells and the number of clusters in 3D are smaller than those in 2D, but the computational time is larger, because there involve more fourth-order derivatives, and more complex form of the indefinite integrals of these derivatives in three space dimensions.

Fig. 9 $\sigma_{11}^{Macro} - \varepsilon_{11}^{Macro}$ curves for $m = 21^3$: The reference solution plotted by solid line

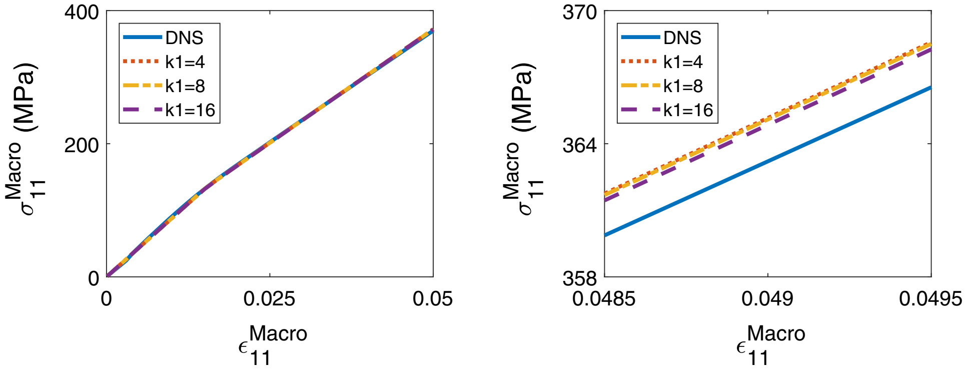


Table 3 Computational time for interaction tensors and deviations of reference solution under the loading $\varepsilon_{11} = 0.05$ for $k_1 = 8$

m	21^3	14^3	7^3	3^3
CPU time (s)	829.31	137.32	12.89	0.92
Speed-up	–	6.04	64.33	905.66
Deviations from DNS results (%)	0.55	0.64	0.79	0.87

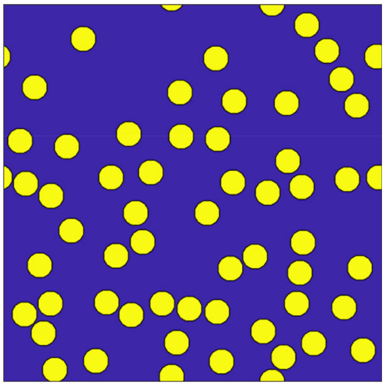


Fig. 10 Geometry for two-phase composite material in the 2D finite strain example, where the color blue represents the matrix and the color yellow represents the inclusion. (Color figure online)

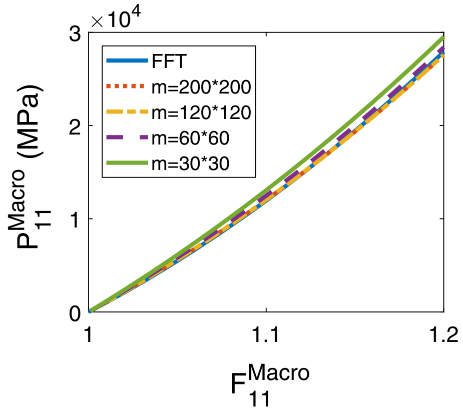


Fig. 12 $P_{11}^{Macro} - F_{11}^{Macro}$ curves for different coarse grid cell number m when taking $k_1 = 8$

Fig. 11 $P_{11}^{Macro} - F_{11}^{Macro}$ curves: $m = 200^2$ and the reference solution plotted by solid line

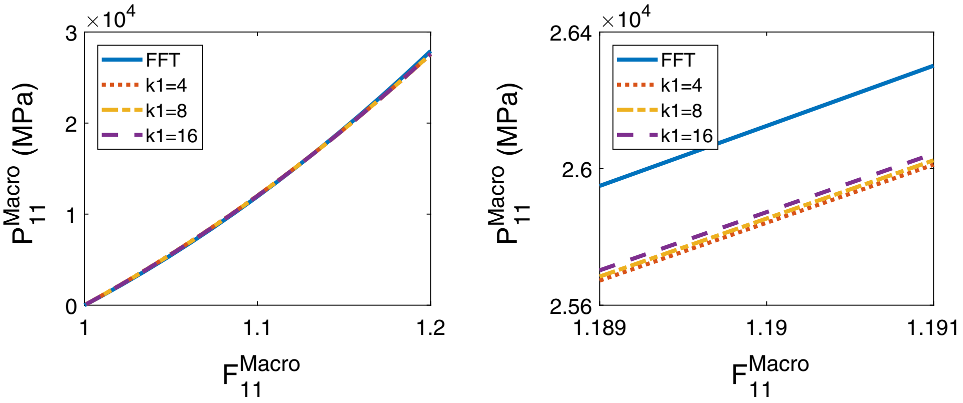


Table 4 Computational time for interaction tensors and deviations of the results from DNS under the loading $F_{11} = 1.2$ for $k_1 = 8$

m	200^2	120^2	60^2	30^2
CPU time (s)	2995.24	361.24	25.18	2.76
Speed-up	–	8.29	118.97	1084.15
deviations from DNS results (%)	−1.18	−1.20	1.71	5.71

3.2 Hyper-elastic materials

In this section, we deal with the finite strain problem. The setup and the Lippmann–Schwinger equation in this problem are shown in "Appendix C".

One of the simplest constitutive model in this class of problem is a hyper-elastic model. In the undeformed configuration, we assume a linear relation between the second Piola–Kirchhoff stress $\boldsymbol{S}$ and the Green strain $\boldsymbol{E}$

$$\boldsymbol{S} = \boldsymbol{C} : \boldsymbol{E}, \tag{48}$$

where $\boldsymbol{C}$ is the standard fourth-order isotropic stiffness tensor.

By VCA, we simulate the 2D plane strain problem under uniaxial tension. In this example, we consider a two-phases material with some fibers (phase 2) embedded in the matrix (phase 1) where the volume fraction of the inclusions is 20% as shown in Fig. 10. The Young's moduli and Poisson's ratios of phases 1 and 2 are:

$$E_1 = 7 \times 10^4 \text{ MPa}, \nu_1 = 0.33 \quad \text{in phase 1;}$$
$$E_2 = 4 \times 10^5 \text{ MPa}, \nu_2 = 0.2 \quad \text{in phase 2.}$$

The mesh size for the high-fidelity RVE model is 600 × 600. Here, the reference result is obtained with the FFT-based method with the basic scheme [18,19]. The results under the unixial tension loading $\varepsilon_{11} = 0.001$ are used as data-base of clustering. As the volume fraction of the inclusion phase is 20%, we take $k_2 = k_1/4$ in the clustering process.

The reference material takes Young's modulus 1.4×10^5 MPa and the Poisson's ratio ν 0.33.

The response curves $P_{11}^{Macro} - F_{11}^{Macro}$ predicted by VCA and the reference results are shown in Figs. 11 and 12. The reference solution is plotted as solid lines. The loading ranges from 0 to 0.2, much beyond the limit of small deformation. Although the response performs differently compared to the small strain case, the accuracy of the prediction improves when the number of clusters increases. VCA results is close to the reference solution, but is slightly softer. Figure 12 shows the response curves for different coarse grid cell number m when 8 clusters in the matrix. The response performs harder when m decreases. The corresponding CPU time and errors are shown in Table 4. CPU time increases rapidly while gaining higher accuracy. In consideration of efficiency and accuracy, the results for $m = 60^2$ are satisfactory.

4 Conclusion

In this paper, we propose a fast method to calculate interaction tensors based on two major components. First, we give explicit formulas for integrating terms in the Green's functions, in both two and three dimensions. Second, we propose an efficient approximation technique based on interaction tensors over coarse grid cells and composition ratios. In the numerical tests, we improve VCA by the fast method for calculating interaction tensors and apply it to a simple hyper-elastic model for illustration. Advantages of the proposed method are summarized as follows.

1. Efficiency: The computational time is reduced considerably when the number of coarse grid cells decreases.
2. Convergence: The numerical results converge to the DNS results when voxels/pixels and coarse grid cells are all cubes/rectangles and coincide.
3. Applicability: Our method applies to any shape RVE and voxels/pixels.

In the calculation of interaction tensors, there are some issues to be further studied. First, when unstructured mesh and coarse grid are used, the convergence needs to be explored. Secondly, the singularity of the Green's functions needs delicate and effective treatment if the kernel functions are not homogeneous functions. Finally, the method requires the analytical forms of interaction tensors.

Acknowledgements Lei Zhang, Xi Zhu and Shaoqiang Tang were supported partially by NSFC under Grant Numbers 11832001, 11521202, and 11890681. Cheng Yu and Wing Kam Liu were supported by award 70NANB14H012 from US Department of Commerce, National Institute of Standards and Technology as part of the Center for Hierarchical Materials Design (CHiMaD).

Appendix A: The Lippmann–Schwinger equation in the Fourier space for two space dimensions

We solve the governing system (1–3) in the Fourier space for two space dimensions to show the differences of interaction tensors between VCA and SCA.

First, (2) can be replaced by the strain compatibility equation

$$\nabla \times \boldsymbol{\varepsilon} \times \nabla = 0, \tag{49}$$

and we rewrite the governing system as

$$\nabla \times \tilde{\boldsymbol{\varepsilon}} \times \nabla = 0, \tag{50}$$

$$\nabla \cdot (\boldsymbol{C}^0 : \tilde{\boldsymbol{\varepsilon}}) + \nabla \cdot \boldsymbol{\tau} = 0 \tag{51}$$

in terms of the difference $\tilde{\boldsymbol{\varepsilon}} = \boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^0$.

In VCA, the RVE is surrounded by the reference material with constant stiffness $\boldsymbol{C}^0$. Therefore it is remarked that $\tilde{\boldsymbol{\varepsilon}}$ equals to zero at infinity. Taking Fourier transform $(x_1, x_2) \to (\xi, \eta)$, the equation is expressed in the Fourier space as

$$\eta^2 \mathcal{F}\tilde{\varepsilon}_{11} + \xi^2 \mathcal{F}\tilde{\varepsilon}_{22} - 2\xi\eta\mathcal{F}\tilde{\varepsilon}_{12} = 0, \tag{52}$$

$$\begin{aligned}(\lambda_0 + 2\mu_0)\xi\mathcal{F}\tilde{\varepsilon}_{11} + \lambda_0\xi\mathcal{F}\tilde{\varepsilon}_{22} + 2\mu_0\eta\mathcal{F}\varepsilon_{\tilde{1}2}\\ + \xi\mathcal{F}\tau_{11} + \eta\mathcal{F}\tau_{12} = 0,\end{aligned} \tag{53}$$

$$\begin{aligned}2\mu_0\xi\mathcal{F}\tilde{\varepsilon}_{12} + \lambda_0\eta\mathcal{F}\tilde{\varepsilon}_{11} + (\lambda_0 + 2\mu_0)\eta\mathcal{F}\tilde{\varepsilon}_{22}\\ + \xi\mathcal{F}\tau_{12} + \eta\mathcal{F}\tau_{22} = 0.\end{aligned} \tag{54}$$

Thus we have

$$\begin{aligned}&\mathcal{F}\begin{bmatrix}\tilde{\varepsilon}_{11}\\ \tilde{\varepsilon}_{22}\\ \tilde{\varepsilon}_{12}\end{bmatrix} + \begin{bmatrix}\eta^2 & \xi^2 & -2\xi\eta\\ (\lambda_0+2\mu_0)\xi & \lambda_0\xi & 2\mu_0\eta\\ \lambda_0\eta & (\lambda_0+2\mu_0)\eta & 2\mu_0\xi\end{bmatrix}^{-1}\\ &\quad\begin{bmatrix}0 & 0 & 0\\ \xi & 0 & \eta\\ 0 & \eta & \xi\end{bmatrix}\mathcal{F}\begin{bmatrix}\tau_{11}\\ \tau_{22}\\ \tau_{12}\end{bmatrix}\\ &\quad = \mathcal{F}\begin{bmatrix}\tilde{\varepsilon}_{11}\\ \tilde{\varepsilon}_{22}\\ \tilde{\varepsilon}_{12}\end{bmatrix} + H\mathcal{F}\begin{bmatrix}\tau_{11}\\ \tau_{22}\\ \tau_{12}\end{bmatrix} = 0,\end{aligned} \tag{55}$$

where we denote

$$\begin{aligned}H = &\frac{1}{\mu_0(\lambda_0+2\mu_0)(\xi^2+\eta^2)^2}\\ &\begin{bmatrix}\mu_0\xi^4 + (\lambda_0+2\mu_0)\xi^2\eta^2 & -(\lambda_0+\mu_0)\xi^2\eta^2 & -\lambda_0\xi^3\eta + (\lambda_0+2\mu_0)\xi\eta^3\\ -(\lambda_0+\mu_0)\xi^2\eta^2 & \mu_0\eta^4 + (\lambda_0+2\mu_0)\xi^2\eta^2 & -\lambda_0\xi\eta^3 + (\lambda_0+2\mu_0)\xi^3\eta\\ \frac{1}{2}(-\lambda_0\xi^3\eta + (\lambda_0+2\mu_0)\xi\eta^3) & \frac{1}{2}(-\lambda_0\xi\eta^3 + (\lambda_0+2\mu_0)\xi^3\eta) & -\lambda_0\xi^2\eta^2 + \frac{1}{2}(\lambda_0+2\mu_0)(\xi^4+\eta^4)\end{bmatrix}.\end{aligned} \tag{56}$$

(55) is expressed in the form of tensor as

$$\mathcal{F}\tilde{\boldsymbol{\varepsilon}} + \hat{\boldsymbol{\Phi}} : \mathcal{F}\boldsymbol{\tau} = 0. \tag{57}$$

The entries in Φ are from entries in the matrix H. The tensor $\hat{\boldsymbol{\Phi}}$ is the Fourier transform of the kernel functions Φ actually and the equation above (57) is the Lippmann–Schwinger in the Fourier space.

In SCA, periodic boundary conditions are applied to the RVE problem and FFT is used to calculate interaction tensors. Hence, the problem is solved based on the Fourier series rather than the Fourier transform over the whole space. We denote

$$\boldsymbol{\varepsilon} \sim \sum_{p=-\infty}^{+\infty}\sum_{q=-\infty}^{+\infty} \hat{\varepsilon}_{pq} \exp(2ip\pi/T + 2iq\pi/T), \tag{58}$$

where T is the period. The system for each pair $(\xi_p, \eta_q) = (2p\pi/T, 2q\pi/T)$ is expressed by

$$\eta_q^2 \mathcal{F}\tilde{\varepsilon}_{11} + \xi_p^2 \mathcal{F}\tilde{\varepsilon}_{22} - 2\xi_p\eta_q\mathcal{F}\tilde{\varepsilon}_{12} = 0, \tag{59}$$

$$\begin{aligned}(\lambda_0 + 2\mu_0)\xi_p\mathcal{F}\tilde{\varepsilon}_{11} + \lambda_0\xi_p\mathcal{F}\tilde{\varepsilon}_{22} + 2\mu_0\eta_q\mathcal{F}\varepsilon_{\tilde{1}2}\\ + \xi_p\mathcal{F}\tau_{11} + \eta_q\mathcal{F}\tau_{12} = 0,\end{aligned} \tag{60}$$

$$\begin{aligned}2\mu_0\xi_p\mathcal{F}\tilde{\varepsilon}_{12} + \lambda_0\eta_q\mathcal{F}\tilde{\varepsilon}_{11} + (\lambda_0 + 2\mu_0)\eta_q\mathcal{F}\tilde{\varepsilon}_{22}\\ + \xi_p\mathcal{F}\tau_{12} + \eta_q\mathcal{F}\tau_{22} = 0.\end{aligned} \tag{61}$$

Similar to the process above, we can also derive the equation in the form of tensor

$$\mathcal{F}\tilde{\boldsymbol{\varepsilon}} + \hat{\boldsymbol{\Phi}}_{pq} : \mathcal{F}\boldsymbol{\tau} = 0. \tag{62}$$

The form of entries in the tensor $\hat{\boldsymbol{\Phi}}_{pq}$ is same as that in $\hat{\boldsymbol{\Phi}}$ but $\hat{\boldsymbol{\Phi}}_k$ depends on (ξ_p, η_q).

VCA uses Green's functions with zero strain boundary condition at infinity, while SCA uses Fourier series of them to express interaction tensors.

Appendix B: Calculation of interaction tensors in three space dimensions

In the same way as for two space dimensions, we show the details of the calculation of $\boldsymbol{D}^{IJ}$ in three space dimensions. We also denote $u^{(i)}, \varepsilon^{(i)}$ as the fundamental solution to

$$\nabla \cdot (\boldsymbol{C}^0 : \varepsilon) = \delta(\boldsymbol{x})\boldsymbol{e}_i, \tag{63}$$

where $\boldsymbol{e}_i$ is the unit vector at x_i direction.

Taking Fourier transform $(x_1, x_2, x_3) \to (\xi, \eta, \zeta)$, we get the corresponding solution

$$\mathcal{F}\boldsymbol{u}^{(1)} = -\frac{\phi_0}{(\xi^2+\eta^2+\zeta^2)^2}\begin{bmatrix}\xi^2 + (s+1)(\eta^2+\zeta^2)\\ -s\xi\eta\\ -s\xi\zeta\end{bmatrix}, \tag{64}$$

$$\mathcal{F}\boldsymbol{u}^{(2)} = -\frac{\phi_0}{(\xi^2+\eta^2+\zeta^2)^2}\begin{bmatrix} -s\xi\eta \\ \eta^2+(s+1)(\xi^2+\zeta^2) \\ -s\eta\zeta \end{bmatrix}, \tag{65}$$

$$\mathcal{F}\boldsymbol{u}^{(3)} = -\frac{\phi_0}{(\xi^2+\eta^2+\zeta^2)^2}\begin{bmatrix} -s\xi\zeta \\ -s\eta\zeta \\ \zeta^2+(s+1)(\xi^2+\eta^2) \end{bmatrix} \tag{66}$$

again with λ_0, μ_0 as the Lamé coefficients of the reference material. We also denote $s = \frac{\lambda_0+\mu_0}{\mu_0}$ and let $\phi_0 = \frac{1}{\mu_0(1+s)}$.

In terms of strain, we have

$$\mathcal{F}\varepsilon^{(1)} = -\frac{i\phi_0}{(\xi^2+\eta^2+\zeta^2)^2}$$
$$\begin{bmatrix} \xi[\xi^2+(1+s)(\eta^2+\zeta^2)] & *** & *** \\ \frac{\eta}{2}[(1-s)\xi^2+(1+s)(\eta^2+\zeta^2)] & -s\xi\eta^2 & *** \\ \frac{\zeta}{2}[(1-s)\xi^2+(1+s)(\eta^2+\zeta^2)] & -s\xi\eta\zeta & -s\xi\zeta^2 \end{bmatrix}, \tag{67}$$

$$\mathcal{F}\varepsilon^{(2)} = -\frac{i\phi_0}{(\xi^2+\eta^2+\zeta^2)^2}$$
$$\begin{bmatrix} -s\xi^2\eta & \frac{\xi}{2}[(1-s)\eta^2+(1+s)(\xi^2+\zeta^2)] & -s\xi\eta\zeta \\ *** & \eta[\eta^2+(1+s)(\xi^2+\zeta^2)] & *** \\ *** & \frac{\zeta}{2}[(1-s)\eta^2+(1+s)(\xi^2+\zeta^2)] & -s\eta\zeta^2 \end{bmatrix}, \tag{68}$$

$$\mathcal{F}\varepsilon^{(3)} = -\frac{i\phi_0}{(\xi^2+\eta^2+\zeta^2)^2}$$
$$\begin{bmatrix} -s\xi^2\zeta & *** & \frac{\xi}{2}[(1-s)\zeta^2+(1+s)(\xi^2+\eta^2)] \\ -s\xi\eta\zeta & -s\eta^2\zeta & \frac{\eta}{2}[(1-s)\zeta^2+(1+s)(\xi^2+\eta^2)] \\ *** & *** & \zeta[\zeta^2+(1+s)(\xi^2+\eta^2)] \end{bmatrix}. \tag{69}$$

Here $***$ denotes an entry determined by symmetry. We use the fundamental solution to the biharmonic equation

$$f = -\frac{1}{8\pi}\sqrt{x_1^2+x_2^2+x_3^2} \tag{70}$$

to express the kernel function Φ as

$$\Phi_{11} = \frac{\phi_0}{8\pi}\begin{bmatrix} f_{1111}+(1+s)(f_{1122}+f_{1133}) & *** & *** \\ (1-s)f_{1112}+(1+s)(f_{2221}+f_{3312}) & -sf_{1122} & -2sf_{1123} \\ (1-s)f_{1113}+(1+s)(f_{2213}+f_{3331}) & *** & -sf_{1133} \end{bmatrix}, \tag{71}$$

$$\Phi_{22} = \frac{\phi_0}{8\pi}\begin{bmatrix} -sf_{1122} & (1-s)f_{2221}+(1+s)(f_{1112}+f_{3312}) & -2sf_{2213} \\ *** & f_{2222}+(1+s)(f_{1122}+f_{2233}) & *** \\ *** & (1-s)f_{2223}+(1+s)(f_{1123}+f_{3332}) & -sf_{2233} \end{bmatrix}, \tag{72}$$

$$\Phi_{33} = \frac{\phi_0}{8\pi}\begin{bmatrix} -sf_{1133} & -2sf_{3312} & (1-s)f_{3331}+(1+s)(f_{1113}+f_{2213}) \\ *** & -sf_{2233} & (1-s)f_{3332}+(1+s)(f_{1123}+f_{2223}) \\ *** & *** & f_{3333}+(1+s)(f_{1133}+f_{2233}) \end{bmatrix}, \tag{73}$$

$$\Phi_{23} = \frac{\phi_0}{8\pi}\begin{bmatrix} -sf_{1123} & \frac{1}{2}((1-3s)f_{2213}+(1+s)(f_{1113}+f_{3331})) & \frac{1}{2}((1-3s)f_{3312}+(1+s)(f_{1112}+f_{2221})) \\ *** & \frac{1}{2}((1-s)f_{2223}+(1+s)(f_{1123}+f_{3332})) & (1-s)f_{2233}+\frac{1}{2}(1+s)(f_{1133}+f_{1122}+f_{3333}+f_{2222}) \\ *** & *** & \frac{1}{2}((1-s)f_{3332}+(1+s)(f_{1123}+f_{2223})) \end{bmatrix}, \tag{74}$$

$$\Phi_{13} = \frac{\phi_0}{8\pi}\begin{bmatrix} \frac{1}{2}((1-s)f_{1113}+(1+s)(f_{2213}+f_{3331})) & *** & (1-s)f_{1133}+\frac{1}{2}(1+s)(f_{2233}+f_{3333}+f_{1111}+f_{1122}) \\ \frac{1}{2}((1-3s)f_{1123}+(1+s)(f_{2223}+f_{3332})) & -sf_{2213} & \frac{1}{2}((1-3s)f_{3312}+(1+s)(f_{1112}+f_{2221})) \\ *** & *** & \frac{1}{2}((1-s)f_{3331}+(1+s)(f_{1113}+f_{2213})) \end{bmatrix}, \tag{75}$$

$$\Phi_{12} = \frac{\phi_0}{8\pi}\begin{bmatrix} \frac{1}{2}((1-s)f_{1112}+(1+s)(f_{2221}+f_{3312})) & (1-s)f_{1122}+\frac{1}{2}(1+s)(f_{2222}+f_{2233}+f_{1111}+f_{1133}) & *** \\ *** & \frac{1}{2}((1-s)f_{2221}+(1+s)(f_{1112}+f_{3312})) & *** \\ \frac{1}{2}((1-3s)f_{1123}+(1+s)(f_{2223}+f_{3332})) & \frac{1}{2}((1-3s)f_{2213}+(1+s)(f_{1113}+f_{3331})) & -sf_{3312} \end{bmatrix}. \tag{76}$$

Here $f_{ijkl}, i, j, k, l = 1, 2, 3$ are the fourth-order derivatives of f. Before computing $\boldsymbol{D}^{IJ}$ for three dimentions, the indefinite integrals of these derivatives are needed. Because of the complex forms, we just show the integral for f_{1112} to illustrate as below.

$$\begin{aligned}
&8\pi \iint f_{1112}(x, y, z)\mathrm{d}x\mathrm{d}y\mathrm{d}z\mathrm{d}x\mathrm{d}y\mathrm{d}z \\
&\quad = 1/72(-((15x^3y)/\sqrt{(x^2+y^2+z^2)}) \\
&\qquad -(6xy^3)/\sqrt{x^2+y^2+z^2} + (9xyz^2)/\sqrt{x^2+y^2+z^2} \\
&\qquad -30xy\sqrt{x^2+y^2+z^2} \\
&\qquad -(24x^3z(-((yz)/(x^2+y^2+z^2)^{3/2}) \\
&\qquad -(yz)/(x^2\sqrt{x^2+y^2+z^2})))/(1 \\
&\qquad +(y^2z^2)/(x^2(x^2+y^2+z^2))) \\
&\qquad -(9x^5)/(\sqrt{x^2+y^2+z^2}(y+\sqrt{x^2+y^2+z^2})) \\
&\qquad +(18x^3z^2)/(\sqrt{x^2+y^2+z^2}(y+\sqrt{x^2+y^2+z^2})) \\
&\qquad +(3xz^4)/(\sqrt{x^2+y^2+z^2}(y+\sqrt{x^2+y^2+z^2})) \\
&\qquad +(36x^3yz)/(\sqrt{x^2+y^2+z^2}(z+\sqrt{x^2+y^2+z^2})) \\
&\qquad +(12xy^3z)/(\sqrt{x^2+y^2+z^2}(z+\sqrt{x^2+y^2+z^2})) \\
&\qquad -72x^2z\arctan((yz)/(x\sqrt{x^2+y^2+z^2})) \\
&\qquad -36x^3\mathrm{In}\,(y+\sqrt{x^2+y^2+z^2}) \\
&\qquad +36xz^2\mathrm{In}\,(y+\sqrt{x^2+y^2+z^2}) \\
&\qquad +72xyz\mathrm{In}\,(z+\sqrt{x^2+y^2+z^2})),
\end{aligned} \tag{77}$$

For the sake of clarify, here we present by replacing (x_1, x_2, x_3) with (x, y, z).

Appendix C: Finite strain formulation

The finite strain problem is different from the case with small deformation, which is governed by the following equations in the undeformed configuration:

$$\nabla \cdot \boldsymbol{P}^T(\boldsymbol{F}) = \boldsymbol{0}, \tag{78}$$

$$\boldsymbol{F} = \nabla_X(\boldsymbol{x}) = \boldsymbol{I}_2 + \nabla \boldsymbol{u}(\boldsymbol{X}), \quad \forall \boldsymbol{X} \in \Omega, \tag{79}$$

where $\boldsymbol{P}$ is the first Piola–Kirchhoff stress, $\boldsymbol{F}$ is the deformation gradient at each material point $\boldsymbol{X}$.

The loading is also applied at infinity through the reference material. The homogeneous isotropic reference material is described by a constitutive relation:

$$\boldsymbol{P}(\boldsymbol{F}) = \boldsymbol{C}^0 : (\frac{1}{2}(\boldsymbol{F}^T + \boldsymbol{F}) - \boldsymbol{I}_2) = \boldsymbol{C}^0 : (\boldsymbol{F}(\boldsymbol{X}) - \boldsymbol{I}_2). \tag{80}$$

$\boldsymbol{C}^0$ is the standard fourth-order isotropic elastic stiffness tensor.

Similar to Eq. (13), the finite strain system can also be reformulated by the Lippmann–Schwinger equation

$$\boldsymbol{F}(\boldsymbol{X}) - \boldsymbol{F}^0 + \tilde{\boldsymbol{\Phi}} * (\boldsymbol{P}(\boldsymbol{X}) - \boldsymbol{C}^0 : (\boldsymbol{F}(\boldsymbol{X}) - \boldsymbol{I}_2)) = 0. \tag{81}$$

Here, we neglect the boundary term. $\tilde{\boldsymbol{\Phi}}$ is the fourth-order tensor defined by

$$\tilde{\Phi}_{ijkl} = \frac{\partial}{\partial X^j} u_{k,l}^{(i)}. \tag{82}$$

After clustering the material points based on DNS results, we can also obtain the discrete Lippmann–Schwinger equation

$$\boldsymbol{F}^I - \boldsymbol{F}^0 + \sum_{J=1}^{k} \boldsymbol{D}_{finitestrain}^{IJ} : (\boldsymbol{P}^J - \boldsymbol{C}^0 : (\boldsymbol{F}^J - \boldsymbol{I}_2)) = 0, \tag{83}$$

where $\boldsymbol{D}_{finitestrain}^{IJ}$ is the interaction tensor for the finite strain problem given by

$$\boldsymbol{D}_{finitestrain}^{IJ} = \frac{1}{|\Omega_I|} \int_{\Omega^I} \left(\int_{\Omega^J} \boldsymbol{\Phi}(\boldsymbol{X} - \tilde{\boldsymbol{X}}) d\tilde{\boldsymbol{X}} \right) d\boldsymbol{X}. \tag{84}$$

References

1. Voigt W (1889) On the relation between the elasticity constants of isotropic bodies. Ann Phys Chem 274:573–587
2. Eshelby JD (1957) The determination of the elastic field of an ellipsoidal inclusion, and related problems. Proc R Soc Lond A 241(1226):376–396
3. Hashin Z, Shtrikman S (1963) A variational approach to the theory of the elastic behaviour of multiphase materials. J Mech Phys Solids 11(2):127–140
4. Hill R (1965) A self-consistent mechanics of composite materials. J Mech Phys Solids 13(4):213–222
5. Budiansky B (1965) On the elastic moduli of some heterogeneous materials. J Mech Phys Solids 13(4):223–227
6. Mori T, Tanaka K (1973) Average stress in matrix and average elastic energy of materials with misfitting inclusions. Acta Metall 21(5):571–574
7. Liu Z, Moore JA, Aldousari SM, Hedia HS, Asiri SA, Liu WK (2015) A statistical descriptor based volume-integral micromechanics model of heterogeneous material with arbitrary inclusion shape. Comput Mech 55(5):963–981
8. Dvorak GJ (1992) Transformation field analysis of inelastic composite materials. Proc R Soc A: Math Phys Eng Sci 437(1900):311–327. https://doi.org/10.1098/rspa.1992.0063

9. Michel JC, Suquet P (2003) Nonuniform transformation field analysis. Int J Solids Struct 40(25):6937–6955
10. Roussette S, Michel JC, Suquet P (2009) Nonuniform transformation field analysis of elastic-viscoplastic composites. Compos Sci Technol 69(1):22–27
11. Berkooz G, Holmes P, Lumley JL (1993) The proper orthogonal decomposition in the analysis of turbulent flows. Annu Rev Fluid Mech 25(1):539–575
12. Liu Z, Bessa M, Liu WK (2016) Self-consistent clustering analysis: an efficient multi-scale scheme for inelastic heterogeneous materials. Comput Methods Appl Mech Eng 306:319–341
13. Tang S, Zhang L, Liu WK (2018) From virtual clustering analysis to self-consistent clustering analysis: a mathematical study. Comput Mech 1–18
14. Liu Z, Fleming M, Liu WK (2018) Microstructural material database for self-consistent clustering analysis of elastoplastic strain softening materials. Comput Methods Appl Mech Eng 330:547–577
15. Liu Z, Kafka OL, Yu C, Liu WK (2018) Data-driven self-consistent clustering analysis of heterogeneous materials with crystal plasticity. In: Oñate E, Peric D, de Souza Neto E, Chiumenti M (eds) Advances in computational plasticity, vol. 46 of computational methods in applied sciences. Springer, London, pp 221–242. https://doi.org/10.1007/978-3-319-60885-3_11
16. Shakoor M, Kafka OL, Yu C, Liu WK (2018) Data science for finite strain mechanical science of ductile materials. Comput Mech 1–13. https://doi.org/10.1007/s00466-018-1655-9
17. Kafka OL, Yu C, Shakoor M, Liu Z, Wagner GJ, Liu WK (2018) Data-driven mechanistic modeling of influence of microstructure on high-cycle fatigue life of nickel titanium, JOM 1–5
18. Yu C, Kafka OL, Liu WK (2019) Self-consistent clustering analysis for multiscale modeling at finite strains. Comput Methods Appl Mech Eng 349:339–359
19. Moulinec H, Suquet P (1998) A numerical method for computing the overall response of nonlinear composites with complex microstructure. Comput Methods Appl Mech Eng 157(1–2):69–94

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.