Direct numerical simulation of complex nano-structured composites, considering interface stretching and bending effects, using nano-computational grains

Junbo Wang¹,², Peng Yan¹, Leiting Dong¹*, Satya N. Atluri³

¹School of Aeronautic Science and Engineering, Beihang University, China
²Shenyang Aircraft Design Institute Yangzhou Collaborative Innovation Institute Co., Ltd., China
³Department of Mechanical Engineering, Texas Tech University, USA

Abstract: Driven by the promising applications of nano-composites, the Steigmann-Ogden (S-O) interface stress model is used together with the classical Elasticity theory to model the effective mechanical properties of nano-composites [32], considering both interface stretching and bending effects. However, no literature has been reported on analytical or numerical solutions for composites containing multiple 3D nano-inclusions with S-O interfaces. In order to overcome this difficulty, a new type of computational grain (CG) is developed with an embedded spherical inclusion and S-O matrix/inclusion interface. The stiffness matrix of each CG is computed by a new boundary-type multi-field variational principle together with Papkovich-Neuber potentials. By evaluating and assembling stiffness matrices of CGs along with parallel computations, very efficient direct numerical simulations of complex nano-composites with a large number of inclusions in a Representative Volume Element (RVE) of the nanocomposite are essentially realized. Numerical examples demonstrate the

* Corresponding author: ltdong@buaa.edu.cn (L. Dong). Address (L.D.): School of Aeronautic Science and Engineering, Beihang University, Beijing, 100191, CHINA.

This article has been accepted for publication and undergone full peer review but has not been through the copyediting, typesetting, pagination and proofreading process which may lead to differences between this version and the Version of Record. Please cite this article as doi: 10.1002/nme.6586
validity and the power of the currently developed CGs. Especially, material models with 10000 nano-inclusions are simulated in around 50 minutes on the 16-core workstation. The influence of interface elastic bending parameters and spatial distributions of the nano-inclusions on the overall properties of nano-composites is also investigated in this study.

**Keywords:** computational grains; nano-inclusion; Steigmann-Ogden interface stress model; parallel computation

1. **Introduction**

Composites reinforced with nano-particles demonstrate quite different mechanical properties as compared to composites with micro-sized reinforcements, part of which can be attributed to the interface effects between the nanoparticles and the matrix. Thus, it will be beneficial to develop effective analytical or numerical nano-mechanical models to study the influence of interface effects, for the design and development of nano-composites. In this context, a direct numerical simulation of a representative volume element (RVE) consisting of a large number of randomly distributed nanoparticles is of paramount importance.

Molecular dynamics (MD) simulations are usually employed to investigate the influences of the matrix/inclusion interfaces and inclusion volume fractions on the mechanical properties of nano-composites (e.g. [1] and many others). However, it seems difficult to employ MD to simulate nano-composites containing a large number of randomly distributed nano-inclusions, because of the unbearable computational burden.
A more practical approach to simulate mechanical behaviours of nano-composites is introducing interface-stress models into the classical theory of elasticity [2-12], thus the matrix and the inclusions are modelled by the classical elasticity, while the behaviour of the interface is accounted for by the interface-stress models. Among various interface stress models [7-17], the Gurtin-Murdoch model [13, 14] (hereinafter referred to as the G-M model) has attracted much attention. However, in the G-M model the surface/interface is regarded as a membrane capable only of stretching (no flexural resistance), which leads to the possibility of instability under compressive surface/interface stresses (e.g. wrinkling) [15]. The absence of surface bending energies in the interface stress model also makes the G-M model unable to account for the observed experimental and computational results on the size-dependent surface stresses of nano-materials [12, 13, 16, 17]. In order to overcome these shortcomings, Steigmann and Ogden [18, 19] generalized the G-M model to take both interface stretching and bending resistances into consideration. The Steigmann-Ogden interface stress model (hereinafter referred to as the S-O model) has been used for mechanical analysis of nanobeams [20], nanowires [21], nanoshells [22], polymer brush [15, 16], and half-space material [5, 23], and the S-O model is recognized as an advancement in the field of surface mechanics [24-28].

In contrast to the large number of studies on the mechanics of composites with the Gurtin-Murdoch interface model (e.g. [18, 28-33] and many others), the literature on nano-particle reinforced composites with Steigmann–Ogden interfaces is rather limited. The only works we are aware of are [34-39]. Most of these papers focus on
analytical solutions for an infinite matrix containing one circular/spherical nano-inhomogeneity (e.g. [28, 31, 40, 41]) and analytical micromechanical methods for overall properties of nano-composites (e.g. [39, 42]). These studies show that the interface bending parameters can affect the local stress distributions as well as effective properties of nano-composites, and thus should not be neglected.

The analytical solution for an infinite matrix containing one spherical nano-inclusion under general far-field loading is already rather complex [40, 41], let alone the problem of a matrix containing multiple nano-inclusions. No literature on analytical solutions for multiple 3D nano-inhomogeneities with S-O interfaces has been reported heretofore. In addition, no numerical simulations (e.g. by the finite element method or the boundary element method, etc.) has been reported for nano-composites with S-O interfaces, to our best knowledge. Thus, developing a numerical tool to simulate nano-composites with S-O interfaces, which can be quite useful for designing and developing nano-composites, is very desirable.

In this study, a new type of computational grains (CGs) are developing for directly simulating nano-composites considering both interface stretching and bending effects. Each three-dimensional CG, which is a virtual or mathematically defined finite-sized geometrical domain of a polyhedral shape, can include a spherical elastic nano-inclusion. A newly developed boundary-type multi-field boundary variational principle as well as Papkovich-Neuber potentials are employed to construct the element stiffness matrix. Parallel computations are implemented in the developed in-house code to further accelerate the simulation of a representative volume element
containing a large number of nano-inclusions. Following the philosophy of previous works for CGs [28, 31, 42], the currently developed nanoscale computational grains can be employed for direct numerical simulations of complex structures of nano-composites, and thus predict the effective mechanical properties as well as interface stress distributions, which will help understand the interfacial damage precursors. Numerical examples for problems of a single, multiple, and a large number of nanoscale inhomogeneities are given to demonstrate the validity and the power of the currently developed CG model for nanomechanics.

The rest of this paper is organized as follows: In Section 2, we gave a brief introduction to the newly derived multi-field boundary variational principle for nano-composites with S-O interfaces. In Section 3, algorithmic implementations of computational grains for nano-composites are given in detail. In Section 4, numerical examples are presented to demonstrate the advantages of the proposed method. In Section 5, we complete this paper with some concluding remarks.

2. A new multi-field boundary variational principle for nano-composites considering with S–O interfaces

Consider a linear elastic nano-composite material containing a large number of randomly distributed nano-inclusions with S-O interfaces, a typical representative material or volume element (RVE) is shown in Fig. 1a. In the CG method, the RVE is discretized into polyhedral CGs by tessellation methods presented in [28], as illustrated in Fig. 1b. In each polyhedral CG, a spherical elastic inclusion is included.
Fig. 1c is a polyhedral CG with a spherical inhomogeneity. In Fig. 1c, \( \Omega_m \) and \( \Omega_c \) denote the matrix and the inclusion in each CG, respectively. \( \partial\Omega_c \) is the boundary of the inclusion, i.e. the matrix/inclusion interface. \( \partial\Omega^e \) is the outer boundary of the element, which can be divided into three parts: the displacement boundary \( S^e_u \), the traction boundary \( S^e_t \), and the inter-element boundary \( \rho^e \), i.e. \( \partial\Omega^e = S^e_u + S^e_t + \rho^e \).

To develop the stiffness matrix of each Trefftz-type CG, independent displacement fields are assumed in the matrix and in the inclusion respectively. This displacement fields are expressed as a linear combination of Trefftz functions presented in the next section. Thus, they satisfy the Navier’s equations in the matrix/inclusion individually:

\[
\mu^e k \nabla^2 \mathbf{u}^e_k + (\lambda^e_k + \mu^e_k) \nabla (\nabla \cdot \mathbf{u}^e_k) = 0, \quad k = m \text{ or } c \tag{1}
\]

where the superscript \( k = m \) denotes the matrix material, and \( k = c \) denotes the inclusion, respectively. \( \nabla \) is the gradient operators. \( \lambda^e_k = \frac{v_k E_k}{(1 - 2v_k)(1 + v_k)} \) and \( \mu^e_k = \frac{E_k}{2(1 + v_k)} \) are Lamé constants for matrix/inclusions, where \( E_k \) and \( v_k \) are the Young's modulus and Poisson's ratio.

Another set of inter-CG compatible displacement field \( \mathbf{u}_m \) is introduced, which satisfies the essential boundary conditions a-priori:

\[
\mathbf{u}_m = \mathbf{u} \quad \text{at} \quad S^e_u \tag{2}
\]

where \( \mathbf{u} \) is the prescribed displacements at the displacement boundary \( S^e_u \). Then the inter-CG traction reciprocity as well as the matrix/inclusion interface conditions can be satisfied in a weak form by stationarity of the boundary functional (note that \( \Pi \) is a boundary-only functional, since the Navier’s equations in the matrix/inclusion are
all satisfied by the Trefftz functions \( u_k, k = m \) or \( c \):

\[
\Pi(u_m, u_c, \tilde{u}_m) = \sum_{k} \left( - \int_{\partial \Omega} \frac{1}{2} \mathbf{t}_m \cdot \mathbf{u}_m \, dS + \int_{\partial \Omega} \mathbf{t}_m \cdot \mathbf{u}_m \, dS - \int_{\partial \Omega} \tilde{\mathbf{f}} \cdot \tilde{\mathbf{u}}_m \, dS + \int_{\partial \Omega} \mathbf{t}_c \cdot \mathbf{u}_c \, dS + \int_{\partial \Omega} U_s \, dS \right)
\]

where \( \tilde{\mathbf{f}} \) is the prescribed traction at the traction boundary \( S^T \). \( \mathbf{t}_m \) and \( \mathbf{t}_c \) are the traction fields on the boundary derived by the Trefftz displacement fields \( u_m \) and \( u_c \) respectively. \( U_s \) is the surface strain energy density defined as [43]:

\[
U_s = \mu_s \varepsilon_s : \varepsilon_s + \frac{1}{2} \lambda_s \text{tr}(\varepsilon_s)^2 + \chi_s : \kappa_s + \frac{1}{2} \zeta_s \text{tr}(\kappa_s)^2
\]

\[
\varepsilon_s = \frac{1}{2} \left[ \nabla_s \mathbf{u}_s \cdot \mathbf{I}_s + \mathbf{I}_s \cdot (\nabla_s \mathbf{u}_s)^T \right]
\]

\[
\kappa_s = \frac{1}{2} \left[ \nabla_s \vartheta \cdot \mathbf{I}_s + \mathbf{I}_s \cdot (\nabla_s \vartheta)^T \right]
\]

\[
\vartheta = \nabla_s (\mathbf{n} \cdot \mathbf{u}_s) - (\nabla_s \mathbf{n}) \cdot \mathbf{u}_s
\]

where \( \mathbf{u}_s, \varepsilon_s \), and \( \kappa_s \) are the interface displacement, strain and curvature, respectively. It should be pointed out that in the boundary functional \( \Pi \) (Eqs.(3-7)) \( \mathbf{u}_s = \mathbf{u}_c \) are enforced a-priori, and \( u_m = u_c \) can be obtained in a weak-form after a variation. \( \lambda_s \) and \( \mu_s \) are the stiffness parameters characterizing the interface stretching effect. \( \chi_s \) and \( \zeta_s \) are the stiffness parameters characterizing the interface bending effect. \( \mathbf{I}_s \) is the unit tangent tensor defined on the interface and \( \mathbf{I}_s = \mathbf{e}_o \otimes \mathbf{e}_o + \mathbf{e}_o \otimes \mathbf{e}_o \) in spherical coordinates. \( \nabla_s = (\mathbf{I}_s - \mathbf{n} \mathbf{n}) \cdot \nabla \) is the gradient operator defined on the interface, where \( \mathbf{n} \) is the unit outer-normal vector of the interface, and \( \mathbf{I}_s \) is the 3D unit tensor.

After setting the first variation of the functional in Eq.(3), the obtained Euler-Lagrange equations of Eq.(3) are:

\[
\mathbf{t}_m = \tilde{\mathbf{f}} \text{ at } S^T \text{ [traction b.c. on a CG]}
\]
\( \mathbf{u}_m = \mathbf{u}_{m} \) at \( \partial \Omega \) [inter CG compatibility]

(9)

\( \mathbf{t}_m^n + \mathbf{t}_m^c = 0 \) at \( \rho^c \) [inter CG traction reciprocity]

(10)

\( \mathbf{u}_m = \mathbf{u}_c \) at \( \partial \Omega_c^e \) [matrix/inclusion interface compatibility]

(11)

\[ \mathbf{u}_c = \mathbf{u}_c \]

\[ \mathbf{u}_c = \mathbf{u}_c \]

\[ \mathbf{u}_c = \mathbf{u}_c \]

\[ \mathbf{u}_c = \mathbf{u}_c \]

\[ \mathbf{u}_c = \mathbf{u}_c \]

where \( \tau_c = 2\mu_c \varepsilon_c + \lambda_c \text{tr}(\varepsilon_c) \mathbf{I}_c \) and \( \mathbf{m}_c = 2\chi_c \kappa_c + \zeta_c \text{tr}(\kappa_c) \mathbf{I}_c \) are the interface stress and bending moment, respectively. Eqs.(11-12) are the governing equations for the S–O interface stress model. Details of the S–O interface stress model can be found in [44].

3. Algorithmic Implementation of computational grains for nano-composites

3.1. Trefftz Trial Functions Scaled by Characteristic Lengths

In this sub-section, we give a brief introduction to the Trefftz trial functions used in the CG method. For composites containing spherical nano-inclusions, the Trefftz trial functions are expressed in terms of the Papkovich-Neuber solution, in which spherical harmonics are employed. Detailed discussions of the completeness of the Papkovich-Neuber solution can be found in [45].

For the inclusion, as shown in Fig.1c, the displacement field \( \mathbf{u}_c \) can be derived by substituting the non-singular harmonics:

\[ \mathbf{B}_c = \sum_{n=0}^{\infty} \left( \frac{R}{R_c} \right)^n \left\{ a_n YC_0^0(\theta, \varphi) + \sum_{l=1}^{n} \left[ a_n^l YC_l^0(\theta, \varphi) + b_n^l YS_l^0(\theta, \varphi) \right] \right\} \]

(13)

into:

\[ \mathbf{u}_c = \mathbf{u}_c = \left[ 4(1 - \nu_c) \mathbf{B}_c + \mathbf{R} \cdot \nabla \mathbf{B}_c - \mathbf{R} \cdot \mathbf{B}_c \right] / 2\mu_c \]

(14)

where the superscript \( c \) denotes the inclusion, and \( a_n^l, b_n^l \) are the unknown coefficients in \( \mathbf{B}_c \) (Eq.(22)). \( YC_0^0(\theta, \varphi) \) and \( YS_l^0(\theta, \varphi) \) are the spherical harmonics.
defined as:

\[ YC_n^l(\theta, \varphi) = \frac{2n+1}{4\pi} \frac{(n-l)!}{(n+l)!} P_n^l(\cos(\theta)) \cos(l \varphi) \]

\[ YS_n^l(\theta, \varphi) = \frac{2n+1}{4\pi} \frac{(n-l)!}{(n+l)!} P_n^l(\cos(\theta)) \sin(l \varphi) \]  

\[ P_n^l(x) = \left(1 - x^2\right)^{\frac{l}{2}} \frac{d^{n+l}}{dx^{n+l}} \left(x^2 - 1\right)^n \]  

The displacement field \( \mathbf{u}_m \) in the matrix is the summation of \( \mathbf{u}_m^p \) (the non-singular part) and \( \mathbf{u}_m^k \) (the singular part, with the singularity located at the center of the inclusion). \( \mathbf{u}_m^p \) can be derived by substituting:

\[ \mathbf{B}_m^p = \sum_{n=0}^{k_m} \left( \frac{R_m}{R_c} \right)^n \left\{ c_0 YC_n^0(\theta, \varphi) + \sum_{l=1}^{n} \left[ c_l^p YC_n^l(\theta, \varphi) + d_l^p YS_n^l(\theta, \varphi) \right] \right\} \]  

into:

\[ \mathbf{u}_m^p = \left[ 4(1-\nu)\mathbf{B}_m^p + \mathbf{R} \cdot \nabla \mathbf{B}_m^p - \mathbf{R} \cdot \mathbf{B}_m^p \right] / 2\mu_m \]  

and \( \mathbf{u}_m^k \) can be derived by substituting

\[ \mathbf{B}_m^k = \sum_{n=0}^{k_m} \left( \frac{R_m}{R_c} \right)^{n+l+1} \left\{ c_0 YC_n^0(\theta, \varphi) + \sum_{l=1}^{n} \left[ c_l^k YC_n^l(\theta, \varphi) + f_l^k YS_n^l(\theta, \varphi) \right] \right\} \]  

into:

\[ \mathbf{u}_m^k = \left[ 4(1-\nu)\mathbf{B}_m^k - \nabla \mathbf{R} \cdot \mathbf{B}_m^k \right] / 2\mu_m \]  

Then the displacement field \( \mathbf{u}_m \) in the matrix can be expressed as:

\[ \mathbf{u}_m = \mathbf{u}_m^p + \mathbf{u}_m^k \]  

where \( c_n^l, d_n^l, e_n^l, f_n^l \) are the unknown coefficients in \( \alpha^\epsilon \) (Eq.(21)). \( R_m \) and \( R_c \) are the characteristic lengths introduced to scale the Trefftz trial functions. For this problem, \( R_m \) is the maximum radial distance of points in the matrix, and \( R_c \) is the
radius of the inclusion, as shown in Fig. 2. Therefore, \( \left( \frac{R}{R_w} \right)^n \) and \( \left( \frac{R}{R_c} \right)^{-(n+1)} \) are confined between 0 and 1 for any point in the matrix, and \( \left( \frac{R}{R_c} \right)^n \) is confined between 0 and 1 for any point in the inclusion. Thereby the difficulty of solving ill-conditioned systems of equations can be avoided.

3.2. Stiffness Matrix for nano-composite computational grains

For convenience, we use a matrix and vector notation to express the displacement fields:

\[
\begin{align*}
\mathbf{u}_m &= \mathbf{N}_m \mathbf{a}^e \quad \text{in } \Omega^e_m \\
\mathbf{u}_c &= \mathbf{N}_c \mathbf{b}^e \quad \text{in } \Omega^c
\end{align*}
\]  

(21)

(22)

where \( \mathbf{N}_j \) is the matrix in which each column is a Papkovich-Neuber potential derived from Eqs.(13-20). The elastic strain fields \( \mathbf{\varepsilon}_m = \mathbf{L}_m \mathbf{a}^e \) and \( \mathbf{\varepsilon}_c = \mathbf{L}_c \mathbf{b}^e \) can be obtained by submitting Eqs.(21-22) into displacement-strain equations. The traction fields at the CG boundary and at the matrix/inclusion interface can be obtained by:

\[
\begin{align*}
\mathbf{t}_m &= \mathbf{n}_m \mathbf{C}_m \mathbf{\varepsilon}_m = \mathbf{n}_m \mathbf{C}_m \mathbf{L}_m \mathbf{a}^e = \mathbf{R}_m \mathbf{a}^e \quad \text{at } \partial \Omega^e_m, \partial \Omega^c_m \\
\mathbf{t}_c &= \mathbf{n}_c \mathbf{C}_c \mathbf{\varepsilon}_c = \mathbf{n}_c \mathbf{C}_c \mathbf{L}_c \mathbf{b}^e = \mathbf{R}_c \mathbf{b}^e \quad \text{at } \partial \Omega^c_c
\end{align*}
\]  

(23)

(24)

where \( \mathbf{n}_m, \mathbf{n}_c \) are the unit outer-normal vectors and \( \mathbf{C}_m, \mathbf{C}_c \) are the elasticity matrix of the matrix/inclusion. Similarly, the interface strain fields \( \mathbf{\varepsilon}_i \) and the interface curvature fields \( \mathbf{\kappa}_i \) can be obtained by submitting Eq.(22) into Eq.(5) and Eq.(6), respectively. Thus, we have:
\[ \mathbf{u}_m = \hat{N}_m q' \quad \text{at } \partial \Omega' \]
\[ \mathbf{u}_m = N_m \alpha' \quad \text{in } \Omega_m' \]
\[ \mathbf{t}_m = R_m \alpha' \quad \text{at } \partial \Omega', \partial \Omega'_c \]
\[ \mathbf{u}_c = N_c \beta' \quad \text{in } \Omega_c' \]
\[ \mathbf{t}_c = R_c \beta' \quad \text{at } \partial \Omega'_c \]
\[ \mathbf{e}_c = L_c \beta' \quad \text{at } \partial \Omega'_c \]
\[ \mathbf{\kappa}_c = M_c \beta' \quad \text{at } \partial \Omega'_c \] (25)

Substitute Eq. (25) into Eq. (3), we obtain:

\[ \mathcal{\Delta} = \sum \delta \left( \frac{1}{2} \alpha'^T H_{aa} \alpha' + \alpha'^T G_{aq} q' + \alpha'^T G_{ag} \beta' + \frac{1}{2} \beta'^T H_{bb} \beta' + \frac{1}{2} \beta'^T H_{ab} \alpha' - q'^T Q' \right) \]
\[ = \sum \delta \alpha'^T \left( -H_{aa} \alpha' + G_{aq} q' + G_{ag} \beta' \right) + \delta q'^T \left( G_{aq} \alpha' - Q' \right) + \delta \beta'^T \left( G_{ag} \alpha' + H_{ab} \beta' + H_{aa} \beta' \right) \] (26)

where

\[ G_{aq} = \int R_m^T N_c dS \]
\[ G_{ag} = \int R_m^T N_m dS \]
\[ H_{aa} = \int R_m^T N_m dS \]
\[ H_{ab} = \int R_m^T N_m dS \]
\[ H_{bb} = \int R_m^T N_m dS \]
\[ H_{ab} = \int R_m^T N_m dS \]
\[ Q' = \int \hat{N}^T \tilde{N} dS \]
\[ H_{ab} = \int \frac{1}{2} L_c^T C_c L_c + \int \frac{1}{2} M_c^T C_c M_c dS \] (27)

where \( C_c \) and \( C_k \) are the interface elasticity matrix characterizing the interface stretching and bending, respectively:

\[ C_c = \begin{pmatrix}
\lambda_s + 2\mu_s & \lambda_s & \lambda_s & 0 & 0 & 0 \\
\lambda_s & \lambda_s + 2\mu_s & \lambda_s & 0 & 0 & 0 \\
\lambda_s & \lambda_s & \lambda_s + 2\mu_s & 0 & 0 & 0 \\
0 & 0 & 0 & 4\mu_s & 0 & 0 \\
0 & 0 & 0 & 0 & 4\mu_s & 0 \\
0 & 0 & 0 & 0 & 0 & 4\mu_s 
\end{pmatrix} \] (28)
This leads to finite element equations:

$$\sum_{\epsilon} k^\epsilon q^\epsilon = \sum Q^\epsilon$$

(30)

where

$$k_{11} = G_{aq}^{eT} \left( H_{aa}^e \right)^{-1} G_{aq}^e$$

$$k_{12} = G_{aq}^{eT} \left( H'_{aa}^e \right)^{-1} G_{aq}^e$$

$$k_{22} = G_{aq}^{eT} \left( H'_{aa}^e \right)^{-1} G_{aq}^e + \left( H'_{pp} + H'_{ss} \right)$$

$$k^\epsilon = k_{11} - k_{12} k_{22}^{-1} k_{12}$$

(31)

We call $k^\epsilon$ the stiffness matrix of the computational grain. Thus the global stiffness matrix can be obtained by the usual assembly procedure of FEM. And boundary conditions can be enforced using the same methods of applying loads and constraints for FEM. This makes the currently developed CGs quite suitable for numerical implementation.

3.3. Parallel computation when modelling a large number of nano-inclusions

In the previous works for CGs [29, 46, 47], the stiffness matrix of each CG is computed one by one using sequential computation. Although CGs have shown high efficiency in modelling a RVE containing a small number of inclusions[43-45, 48], parallel computation may further reduce computational time when simulating a RVE containing a large number of inclusions. In this study, in order to accelerate the analysis, parallel computation is implemented with the help of Matlab Parallel...
Computing Toolbox. The flow chart of the parallel algorithm is illustrated in Fig.3. The CG analysis starts with input and initialization of the data structure with given CGs, material properties, constraints and loads. The next step is to determine the number of parallel threads. Suppose there are n CGs after tessellation and k parallel threads for computation. Each of first n-1 parallel threads should compute and assemble the stiffness matrices of \( \lceil n/k \rceil \) CGs (\( \lceil x \rceil \) rounds x to the nearest integer). The last parallel thread should compute and assemble the stiffness matrices of the rest of the CGs. Finally, we can solve the system of equations with the assembled global stiffness matrix to obtain the nodal displacements.

4. Numerical Examples

In this section, we present some numerical examples to demonstrate the validity and efficiency of the CGs developed in this study for modelling nano-composites considering both interface stretching and bending effects. The in-house code of the CG method with parallel computation is run on a workstation with 2 Intel Core E5-2640 processors and 32g RAM.

Due to the lack of experimental data of the interface parameters in the S–O model, here hypothetical parameters are used just to test the performances of the CG method proposed in this study. In the following numerical examples, unless specified otherwise, the material and interface properties are \( E_m = 10 \text{GPa} \), \( v_m = 0.25 \), \( E_c = 50 \text{GPa} \), \( v_c = 0.3 \), \( \lambda_s = 5 \text{N/m} \), \( \mu' = 5 \text{N/m} \), \( \chi_s = 5 \text{nN·nm} \) and \( \zeta_s = 5 \text{nN·nm} \).
4.1. Numerical validation with a single CG

The first example is a spherical nano-inclusion embedded in an infinite matrix, which is subjected to remote tension \( P=100 \text{ Mpa} \) in \( z \) direction. The exact solution can be found in [46]. The radius of the nano-inclusion is 5nm in this example. For numerical implementation, the infinite medium is truncated to a finite cube, as illustrated in Fig. 4. The length of each side of the truncated cube is equal to \( l=1000\text{nm} \). For this problem, only one CG is used to simulate the RVE. Traction boundary conditions are applied to the outer-boundary of the RVE. For the CG used in this example, the displacements of each node on the lower surface are constrained to be the same as the exact solutions.

We compare the computed \( \sigma_{zz} \) along the positive \( x \)-axis and \( \sigma_{xx} \) along the positive \( z \)-axis to those of the exact solution, as illustrated in Fig. 5. Fig. 6 shows the stress distributions of computed stresses on the \( y=0 \) plane and the \( z=0 \) plane. The stress concentration factor at point \( (0,0,R) \) is listed in Table. 1, and the relative error is \( 9.5744 \times 10^{-5} \). From Fig.5 and Table. 1, it is found that computational grains give very accurate computed stresses, even though only one CG is used, which demonstrates the validity of the proposed CG method.

4.2. A large number of interacting spherical nano-inclusions with parallel computation

This example considers an RVE containing \( n \) randomly distributed
nano-inclusions with different radius. Uniform traction $P = 100\text{MPa}$ is applied in the $z$-direction at the lower boundary and the upper boundary. For this problem, $n$ CGs are used to model the RVE, as shown in Fig. 7. The time for modelling 100, 1000 and 10000 spherical nano-inclusions with/without parallel computation are listed in Table. 2. Fig. 8 shows the computed stress distributions of $\sigma_{zz}$ for these three cases. As seen from Table. 2, the CPU time needed for the simulation has been significantly reduced after employing parallel computation. Using the CG method with parallel computation, an RVE with 10000 inclusions can be simulated within 50 minutes. Considering the procedure is also entirely automatic, where in the RVE construction, Voronoi tessellation, and CG computation are efficiently executed, a direct numerical simulation of the nano-composite material is indeed achieved. This is probably impossible when the traditional FEM is used, where tens of thousands of simple finite elements are needed to model a single CG [49].

4.3. The influence of interface bending parameters and spatial distributions of the nano-inclusions

In this section, we will show the effect of the spatial distribution of the nano-inclusions on the overall properties of nano-composites, and investigate whether the interface bending resistance parameters have influence on the effective bulk modulus and shear modulus.

In the first example, the influence of spatial distributions of the nano-inclusions on the overall properties is investigated. The material properties for the inclusion are:
$E_i = 410 \text{ GPa}$ and $\nu_i = 0.14$, and the material properties for the matrix are: $E_m = 71 \text{ GPa}$ and $\nu_m = 0.35$, which are the properties of SiC and Al, respectively. The interface elastic constants are: $\lambda_s = 3.4939 \text{ N/m}$, $\mu_s = -5.4251 \text{ N/m}$, $\chi_s = 5\text{nN} \cdot \text{nm}$ and $\zeta_s = 5\text{nN} \cdot \text{nm}$. 3 typical spatial distributions of the nano-inclusions: (a) primitive cubic (PC), (b) body-centered cubic (BCC), (c) face-centered cubic (FCC) and (d) random distributed packing structures are investigated in this example, as shown in Fig. 9. For random distributed packing structures, 10 different randomly generated RVEs are used the error bars of the calculated effective moduli are plotted. Periodic boundary conditions are applied to the RVE to calculate the effective properties. Detailed discussions of periodic boundary conditions for CGs can be found in [46]. In this example, the length of each side of the RVE is $60\text{nm}$. Fig. 10 shows the variation of the effective shear modulus with the volume fraction of the nano-inclusions. It is observed from Fig. 10 that the effective shear modulus of the nano-composites is affected by the spatial distributions of nano-inclusions, while the influence of spatial distributions on the effective bulk modulus seems to be negligible.

In the second example, we investigate whether the interface bending resistance parameters affect the effective bulk modulus and shear modulus. In this example, a RVE with 172 nano-inclusions of FCC type packing structure are investigated, as shown in Fig. 9(c). The volume fraction of nano-inclusions is $0.3$. The interface bending resistance parameters $\chi_s$ and $\zeta_s$ vary from $-100\text{nN} \cdot \text{nm}$ to $100\text{nN} \cdot \text{nm}$, and other material properties and interface properties keep the same with the previous example. Periodic boundary conditions are applied to the RVE to calculate the
effective properties. Fig. 11 shows the variation of the effective bulk and shear modulus with the interface bending resistance parameters. It is clearly seen that the interface bending resistance parameters have significant influence on the effective shear modulus, but little effect is found on the effective bulk modulus, which agrees well with the conclusion given in by the semi-analytical micromechanical model developed in our previous work [43].

In the last example, the influence of sizes of nano-inclusions on the effective bulk modulus and the shear modulus is analysed. In this example, the RVE used is the same as that in the previous example. In order to show the size effect clearly, we set \( \lambda_i = 100 \text{ N/m} \), \( \mu_i = 100 \text{ N/m} \), \( \chi_i = 100 \text{nN nm} \) and \( \zeta_i = 100 \text{nN nm} \). The radius of the nano-inclusions varies from 1nm to 10nm. Fig. 12 shows the variation of the effective bulk and shear modulus with sizes of the nano-inclusions. It is clearly seen that the interface effect caused by interface bending stiffness parameters is size dependent. The smaller the nano-inclusion is, the more significant interface effects are.

5. Conclusions

In this study, a new type of computational grains is developed for direct numerical modelling of composites with nanoscale inclusions considering both interface stretching and bending effects, using a large number of CGs in a in a representative volume element. The CGs developed in this study are by far the first and the only
numerical tool for direct numerical modelling nano-composites with a large number of inclusions with S-O matrix/inclusion interfaces. By using a new boundary-type multi-field variational principle together with Papkovich-Neuber potentials, the stiffness matrices of Trefftz-type CGs can be directly evaluated and assembled. Together with the parallel algorithms implemented in the developed in-house code, it is found that very efficient simulations of nanocomposites can be realized, e.g. an RVE containing 10000 nano-inclusions only takes 50 minutes on a 16-core computer.

The influence of spatial distributions of the nano-inclusions on the overall properties of nano-composites is also investigated in this study. We also study the influence of interface bending resistance parameters on the effective modulus of nano-composites. Numerical results show that interface bending resistance parameters affect the shear modulus of nano-composites, but their effect on the bulk modulus is negligible.

Acknowledgement

The authors (excluding SN Atluri) thankfully acknowledge the support from the National Key Research and Development Program of China (No. 2017YFA0207800) and the National Natural Science Foundation of China (grant No. 12072011 and 11872008).

References


[20] Yun G, Park HS. Surface stress effects on the bending properties of fcc metal


Bound Elem. 2018;94:172-83.


[47] Rycroft C. Voro++: A three-dimensional Voronoi cell library in C++. Lawrence Berkeley National Lab.(LBNL), Berkeley, CA (United States); 2009.


### Tables

<table>
<thead>
<tr>
<th></th>
<th>Analytical solution</th>
<th>CG</th>
</tr>
</thead>
<tbody>
<tr>
<td>stress concentration factor</td>
<td>1.636787636606527</td>
<td>1.636630923426181</td>
</tr>
</tbody>
</table>

Table. 1. The stress concentration factor at point $(0,0,R)$ computed by the analytical solution and one CG.
<table>
<thead>
<tr>
<th></th>
<th>100</th>
<th>1000</th>
<th>10000</th>
</tr>
</thead>
<tbody>
<tr>
<td>without parallel algorithm</td>
<td>360.28s</td>
<td>2886.23s</td>
<td>30819.18s</td>
</tr>
<tr>
<td>with parallel algorithm</td>
<td>38.17s</td>
<td>254.32s</td>
<td>2857.74s</td>
</tr>
</tbody>
</table>

Table. 2. Time needed for the CG method