Hierarchically parallel coupled finite strain multiscale solver for modeling heterogeneous layers

Matthew Mosby and Karel Matouš*,†

Department of Aerospace and Mechanical Engineering, University of Notre Dame, Notre Dame, IN 46556, USA

SUMMARY

We develop a three-dimensional, hierarchically parallel, finite strain multiscale solver capable of computing nonlinear multiscale solutions with over 1 billion finite elements and over 574 million nonlinear equations on 1552 computing cores. In the vein of FE^2, we use the nested iterative procedure and devote the solver to multiscale cohesive modeling of heterogeneous hyperelastic layers. The hierarchically parallel multiscale solver takes advantage of a client-server non-blocking communication matrix that limits latency, starvation, and overhead by overlaying computations at different scales. We perform simulations of real-scale engineering devices and bridge $\mathscr{O}(10^6)$ in length-scales, spanning from $\mathscr{O}(10^1)$ mm to $\mathscr{O}(10^1)$ nm in spatial resolution. Verification of the hierarchically parallel solver is provided together with a mesh convergence study. Moreover, we report on the scaling performance. Copyright © 2014 John Wiley & Sons, Ltd.

1. INTRODUCTION

Multiscale phenomena are ubiquitous in engineering science and are increasingly important in design of structures and materials. For instance, current structural designs frequently make use of adhesive joints in favor of more traditional fastening techniques for ease of manufacturing and reduction of stress concentrations. Most modern adhesives are highly heterogeneous, containing a wide range of sizes, shapes, and material properties of reinforcing constituents. Examples include rubber particles added for toughening effects [1], microcapsules of healing polymers for strength recovery [2], or silver flakes for increased electrical conductivity [3]. Other examples of heterogeneous layers in engineering are polymeric sealants, such as 3M VHB tape consisting of an epoxy-impregnated foam. Unfortunately, these adhesive joints and/or layers are often the weak link in the load-bearing capacity of structures and detailed understanding of the complex multiscale behavior is essential.

One of the techniques that simultaneously captures all continuum spatial and temporal scales is direct numerical modeling (DNM) (commonly called DNS in the fluid mechanics community). However, even with recent increases in performance and availability of super-computing environments, DNM often remains prohibitively expensive. The large DNM simulations demand access to tens of thousands of computing cores and require long computing times [4, 5]. Furthermore, many methods for solving the large systems of equations arising from the discretized DNM problem do not scale well beyond a few thousand cores without specialized and often problem-specific or hardware-specific considerations [5–7]. Even generating the computational

---

*Correspondence to: Karel Matouš, Department of Aerospace and Mechanical, Engineering, University of Notre Dame, 367 Fitzpatrick Hall of Engineering, Notre Dame, IN 46556, USA.

†E-mail: kmatous@nd.edu

domains (meshes) for complex DNM simulations is difficult and time consuming. Therefore, there is a great need for computational tools capable of spanning the large range of spatial and temporal scales to aid in design of novel material formulations and safety assessments of components.

In this work, we develop a highly scalable, hierarchically parallel, 3D finite strain multiscale solver for simulating heterogeneous hyperelastic layers. In particular, we focus on the parallel implementation, performance, and potential for engineering application. The solver is capable of computing multiscale problems with more than 1 billion finite elements and over 574 million non-linear degrees of freedom (DOFs) on less than 1600 computing cores. The solver is based on the multiscale cohesive modeling formulation, originally proposed by Matouš *et al.* [8] and extended by many others thereafter [9–12]. This theory collapses the macroscopic layer to a surface and attaches a representative unit cell (RUC) with thickness l_c, equal to the thickness of the layer, at each macroscopic point. The macroscale and microscale are linked through Hill's principle of energy equivalence [13] extended to interfaces in [8]. The coupled multiscale problem is solved by a nested iterative scheme [9, 10, 14]. This nested solution method, typically referred to as FE^2, has also been extensively used in multiscale modeling of bulk materials [15–18].

In the nested solution process, each microstructure can be solved independently of both the macroscale and other microstructures, allowing for effective parallelization. There have been a few parallel implementations of the nested solution process [14, 17, 19–22], but they often require that the macroscale and individual microscale domains be computed on a single computing core or set of shared-memory processors. In this work, we present a method for efficiently computing both the macroscale and microscale solutions in a hierarchically parallel fashion that is scalable to many thousands of distributed computing cores.

FE^2 is known to be highly accurate, but it is often imprecisely regarded as too expensive to be practical [23–25]. While FE^2 may be too expensive for deployment in full component design and optimization codes, we show that it is possible to accurately simulate engineering scale domains including the full resolution of the microscale with reasonable computational resources. We show that this framework can bridge $\mathscr{O}(10^6)$ in length scales, providing sufficient detail to simulate a $\mathscr{O}(10^1)$ mm large device with numerical resolution of $\mathscr{O}(10^1)$ nm. Such a device can easily be experimentally tested to provide validation data for our hierarchically parallel FE^2 solver. Once validated, such a tool can prove invaluable in generating a rich database, that is, *Virtual Materials Testing*, to aid in development of reduced order models [23, 25] employed in light-weight modeling and simulation tools. Furthermore, such a tool can aid in discovery of previously unknown or not easily captured (by computations) phenomena.

The remainder of the manuscript is organized as follows. In the next section, we summarize the multiscale cohesive modeling formulation in the 3D finite strain setting for completeness of the presentation. Then, in Section 3, we describe the novel computational implementation including the hierarchically parallel communication structure and solution procedure. Finally, in Section 4, we present a simple verification and convergence study, followed by a strong scaling test, and a large simulation of mixed-mode loading conditions.

2. GOVERNING EQUATIONS

Consider a macroscale body $\Omega_0 \subset \mathbb{R}^3$ consisting of material points $X \in \Omega_0$ with the boundary $\partial\Omega_0$ decomposed such that $\partial\Omega_0 = \partial\Omega_0^u \cup \partial\Omega_0^t$ and $\partial\Omega_0^u \cap \partial\Omega_0^t = \emptyset$. Next, let $\partial\Omega_0^u$ and $\partial\Omega_0^t$ represent the boundaries of applied displacement $\bar{u}$ and traction $\bar{t}$, respectively. Let the heterogeneous interface with thickness $l_c > 0$ be described by an oriented manifold $\Gamma_0 \in \mathbb{R}^2$ with the unit normal $^0N(X)$ in the reference configuration (Figure 1). For the remainder of the presentation, let $^0(\bullet)$ and $^1(\bullet)$ represent quantities at the macroscale and microscale, respectively. The interface decomposes Ω_0 into two bodies (adherends) denoted by $\Omega_0^\pm$. The deformation of the adherends is described by the deformation map $^0\varphi(X) = X + {}^0u(X) \ \forall X \in \Omega_0^\pm$ and deformation gradient, $^0F(X) = \nabla_X {}^0\varphi = 1 + \nabla_X {}^0u \ \forall X \in \Omega_0^\pm$, where ^0u is the macroscopic displacement. The deformation of the interface, Γ_0, is represented by the average deformation gradient
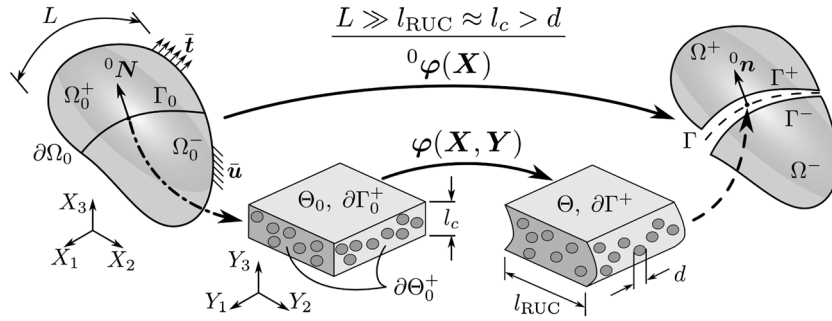
Figure 1. Multiscale modeling of heterogeneous interfaces with widely varying length-scales. In this figure, L is the dimension of the macroscale interface, l_{RUC} is the size of the RUC, and l_c is the thickness of the interface. The RUC contains heterogeneities with characteristic size d.

$$^0\boldsymbol{F}(\boldsymbol{X}) = \mathbf{1} + \frac{1}{l_c}[\![^0\boldsymbol{u}]\!](\boldsymbol{X}) \otimes {}^0\boldsymbol{N}(\boldsymbol{X}) \quad \forall \boldsymbol{X} \in \Gamma_0, \tag{1}$$

where $[\![^0\boldsymbol{u}]\!] = {}^0\varphi^+ - {}^0\varphi^- = {}^0\boldsymbol{u}^+ - {}^0\boldsymbol{u}^-$ is the displacement jump across the interface.

Neglecting inertia and body forces, the quasi-steady equilibrium of a body with an interface is governed by

$$\begin{aligned}
\nabla_{\boldsymbol{X}} \cdot {}^0(\boldsymbol{F}\boldsymbol{S}) &= \boldsymbol{0} && \in \Omega_0^\pm, \\
{}^0(\boldsymbol{F}\boldsymbol{S}) \cdot {}^0\boldsymbol{N} &= \bar{\boldsymbol{t}} && \text{on } \partial\Omega_0^t, \\
{}^0\boldsymbol{u} &= \bar{\boldsymbol{u}} && \text{on } \partial\Omega_0^u, \\
{}^0\boldsymbol{t}^+ + {}^0\boldsymbol{t}^- &= \boldsymbol{0} && \text{on } \Gamma_0,
\end{aligned} \tag{2}$$

where ${}^0\boldsymbol{S} = 2\partial\,{}^0W/\partial\,{}^0\boldsymbol{C}$ is the macroscopic second Piola-Kirchhoff stress tensor, ${}^0\boldsymbol{C} = {}^0\boldsymbol{F}^{\mathrm{T}}\,{}^0\boldsymbol{F}$ is the macroscopic right Cauchy-Green deformation tensor, ${}^0\boldsymbol{t}^\pm$ represents the macroscopic traction vector across the interface, and ${}^0W$ is the macroscopic strain energy density function that defines the behavior of the adherends. Applying the standard variational procedure leads to a typical macroscopic variational problem: Find ${}^0\boldsymbol{u}(\boldsymbol{X}) \in \mathscr{C}_{\Omega_0}\{{}^0\boldsymbol{u} : \Omega_0 \to \mathbb{R}^3|\det({}^0\boldsymbol{F}) > 0 \text{ in } \Omega_0 \text{ and } {}^0\boldsymbol{u}|_{\partial\Omega_0^u} = \bar{\boldsymbol{u}}\}$ such that

$$\mathscr{R}_{0\boldsymbol{u}} := \int_{\Omega_0^\pm} {}^0\boldsymbol{S} : [{}^0\boldsymbol{F}^{\mathrm{T}}\nabla_{\boldsymbol{X}}\delta^0\boldsymbol{u}]^{\mathrm{sym}} \, \mathrm{d}V + \int_{\Gamma_0} {}^0\boldsymbol{t} \cdot [\![\delta^0\boldsymbol{u}]\!] \, \mathrm{d}A - \int_{\partial\Omega_0^t} \bar{\boldsymbol{t}} \cdot \delta^0\boldsymbol{u} \, \mathrm{d}A = 0, \tag{3}$$

holds for variations $\delta^0\boldsymbol{u}(\boldsymbol{X}) \in \mathcal{V}_{\Omega_0}\{\delta^0\boldsymbol{u} : \Omega_0 \to \mathbb{R}^3 | \delta^0\boldsymbol{u}|_{\partial\Omega_0^u} = \boldsymbol{0}\}$. We note that in this work, ${}^0\boldsymbol{t}$ is constructed numerically using microscale solutions.

Let each macroscpic point $\boldsymbol{X} \in \Gamma_0$ consist of a RUC denoted by $\Theta_0 \subset \mathbb{R}^3$. Next, let each RUC $\Theta_0 \subset \mathbb{R}^3$ consist of material points $\boldsymbol{Y} \in \Theta_0$. We assume that the microscale deformation is a function of both macro-variables and micro-variables as

$$\begin{aligned}
\boldsymbol{\varphi}(\boldsymbol{X}, \boldsymbol{Y}) &= {}^0\boldsymbol{F}(\boldsymbol{X})\boldsymbol{Y} + {}^1\boldsymbol{u}(\boldsymbol{Y}) && \forall \boldsymbol{Y} \in \Theta_0, \\
\boldsymbol{F}(\boldsymbol{X}, \boldsymbol{Y}) &= {}^0\boldsymbol{F}(\boldsymbol{X}) + \nabla_{\boldsymbol{Y}}{}^1\boldsymbol{u}(\boldsymbol{Y}) && \forall \boldsymbol{Y} \in \Theta_0,
\end{aligned} \tag{4}$$

where ${}^1\boldsymbol{u}(\boldsymbol{Y})$ is the microscale displacement fluctuation vector. The boundary of the RUC, $\partial\Theta_0$, is decomposed into non-overlapping sections $\partial\Theta_0^\pm$ and $\partial\Gamma_0^\pm$ corresponding to the $Y_{1,2}^\pm$ and $Y_3^\pm$ faces respectively (Figure 1). In this work, Θ_0 is $Y_{1,2}$-periodic. Note that different admissible boundary conditions can also be used [9]. We comment on the field and material periodicity when we solve particular examples. Neglecting body and inertia forces, the equilibrium problem at the microscale reads

$$\begin{aligned}
\nabla_{\boldsymbol{Y}} \cdot (\boldsymbol{F}\,{}^1\boldsymbol{S}) &= \boldsymbol{0} && \in \Theta_0, \\
{}^1\boldsymbol{u} &= \boldsymbol{0} && \text{on } \partial\Gamma_0^\pm, \\
{}^1\boldsymbol{u}^+ - {}^1\boldsymbol{u}^- &= \boldsymbol{0} && \text{on } \partial\Theta_0^\pm, \\
{}^1\boldsymbol{t}^+ + {}^1\boldsymbol{t}^- &= \boldsymbol{0} && \text{on } \partial\Theta_0^\pm.
\end{aligned} \tag{5}$$

In Equation 5, $^1\boldsymbol{S} = 2\partial\,^1W/\partial\boldsymbol{C}$ is the microscale second Piola-Kirchhoff stress tensor, $\boldsymbol{C} = \boldsymbol{F}^{\mathrm{T}}\boldsymbol{F}$ represents the microscale right Cauchy-Green deformation tensor, and ^1W is the microscale strain energy density potential that defines material behavior of the layer.

The macroscale and microscale are linked through the Hill-Mandel condition for interfaces originaly proposed by Matouš *et al.*, [8] which reads

$$\inf_{[\![^0\boldsymbol{u}]\!]}{}^0\psi([\![^0\boldsymbol{u}]\!]) = \inf_{[\![^0\boldsymbol{u}]\!]}\inf_{^1\boldsymbol{u}}\frac{l_c}{|\Theta_0|}\int_{\Theta_0}{}^1W\left({}^0\boldsymbol{F} + \nabla_Y\delta^1\boldsymbol{u}\right)\mathrm{d}V, \tag{6}$$

and minimizes the potential energy at both scales. First, taking variations of Equation 6 with respect to $^1\boldsymbol{u}$ yields the variational problem: Find $^1\boldsymbol{u}(Y) \in \mathscr{C}_{\Theta_0}\{^1\boldsymbol{u}\,:\,\Theta_0\,\rightarrow\,\mathbb{R}^3\,|\det(\boldsymbol{F})\,>\,0$ in Θ_0, $^1\boldsymbol{u}|_{\partial\Gamma_0^{\pm}} = \boldsymbol{0}$ and $^1\boldsymbol{u}$ is $Y_{1,2} -$ periodic on $\partial\Theta_0^{\pm}\}$ such that

$$\mathscr{R}_{1\boldsymbol{u}} := \frac{l_c}{|\Theta_0|}\int_{\Theta_0}{}^1\boldsymbol{S} : \left[\boldsymbol{F}^{\mathrm{T}}\nabla_Y\delta^1\boldsymbol{u}\right]^{\mathrm{sym}}\mathrm{d}V = 0, \tag{7}$$

holds for all variations $\delta^1\boldsymbol{u}(Y) \in \mathscr{V}_{\Theta_0}\{\delta^1\boldsymbol{u} : \Theta_0 \rightarrow \mathbb{R}^3\,\big|\;\delta^1\boldsymbol{u}|_{\partial\Gamma_0^{\pm}} = \boldsymbol{0}$ and $\delta^1\boldsymbol{u}$ is $Y_{1,2}-$periodic on $\partial\Theta_0^{\pm}\}$. Equation 7 is a weak form of the microscale equilibrium (Equation (5)). Second, taking variations of Equation 6 with respect to $[\![^0\boldsymbol{u}]\!]$ yields

$$^0\boldsymbol{t} = \frac{1}{|\Theta_0|}\left[\int_{\Theta_0}\boldsymbol{F}\,^1\boldsymbol{S}\,\mathrm{d}V\right]\cdot{}^0\boldsymbol{N}, \tag{8}$$

which is the closure equation for the macroscale traction in Equation 3.

2.1. Finite-element discretization

The macroscale and microscale are discretized by standard linear tetrahedron finite elements. Moreover, the interface is discretized by linear surface/cohesive elements. Therefore, each macroscopic cohesive element contains one microscale RUC. The displacement field is approximated in the macroscale adherend, interface, and microscale elements (Ω^e, Γ^e and Θ^e) as

$$^0\boldsymbol{u}^h = \sum_a^{nn\in\Omega^e}{}^0\tilde{N}_a{}^0\boldsymbol{u}_a, \quad [\![^0\boldsymbol{u}]\!]^h = \sum_a^{nn\in\Gamma^e}{}^\Gamma\tilde{N}_a{}^0\boldsymbol{u}_a, \text{ and } {}^1\boldsymbol{u}^h = \sum_a^{nn\in\Theta^e}{}^1\tilde{N}_a{}^1\boldsymbol{u}_a.$$

Here, nn is the number of nodes on the element, and $^0\tilde{N}$, $^\Gamma\tilde{N}$, and $^1\tilde{N}$ are finite element shape function operators. The gradients within the volumetric elements are approximated by

$$\nabla_X{}^0\boldsymbol{u}^h = \sum_a^{nn\in\Omega^e}{}^0\tilde{B}_a{}^0\boldsymbol{u}_a, \quad \nabla_Y{}^1\boldsymbol{u}^h = \sum_a^{nn\in\Theta^e}{}^1\tilde{B}_a{}^1\boldsymbol{u}_a,$$

where $^0\tilde{B}$ and $^1\tilde{B}$ are the finite element gradient operators at the different scales.

The discrete weak forms of the equilibrium at both scales (Equations (3) and (7)) are given by

$$\boldsymbol{R}_0 := \underbrace{\overset{\Omega_0^{\pm}}{\underset{e}{\mathbf{A}}}\left[\int_{\Omega_0^{\pm,e}}{}^0\boldsymbol{S} : \left[{}^0\boldsymbol{F}^{\mathrm{T}0}\tilde{B}\right]^{\mathrm{sym}}\mathrm{d}V - \int_{\partial\Omega_0^{t,e}}\bar{\boldsymbol{t}}\cdot{}^0\tilde{N}\,\mathrm{d}A\right]}_{\boldsymbol{R}_0|_{\Omega_0^{\pm}}} + \underbrace{\overset{\Gamma_0}{\underset{e}{\mathbf{A}}}\left[\int_{\partial\Gamma_0^e}{}^0\boldsymbol{t}\cdot{}^\Gamma\tilde{N}\,\mathrm{d}A\right]}_{\boldsymbol{R}_0|_{\Gamma_0}} = \boldsymbol{0}, \tag{9}$$

$$\boldsymbol{R}_1 := \overset{\Theta_0}{\underset{e}{\mathbf{A}}}\frac{l_c}{|\Theta_0|}\int_{\Theta_0^e}{}^1\boldsymbol{S} : \left[\boldsymbol{F}^{\mathrm{T}1}\tilde{B}\right]^{\mathrm{sym}}\mathrm{d}V = \boldsymbol{0}, \tag{10}$$

where $\mathbf{A}$ is the assembly operator. The discrete form of the traction vector (Equation 8) reads

$$^0\boldsymbol{t} = \overset{\Theta_0}{\underset{e}{\mathbf{A}}}\frac{1}{|\Theta_0|}\left[\int_{\Theta_0^e}\boldsymbol{F}\,^1\boldsymbol{S}\,\mathrm{d}V\right]\cdot{}^0\boldsymbol{N}. \tag{11}$$

We solve the discrete nonlinear equations, Equations (9) and (10), using implicit Newton's method. The multiscale consistent tangent matrix is obtained through typical linearization:

$$
\begin{bmatrix} \boldsymbol{K}_{00}|_{\Omega_0^\pm} + \boldsymbol{K}_{00}|_{\Gamma_0} & \boldsymbol{K}_{01} \\ \boldsymbol{K}_{10} & \boldsymbol{K}_{11} \end{bmatrix} \begin{Bmatrix} \Delta^0\boldsymbol{u} \\ \Delta^1\boldsymbol{u} \end{Bmatrix} = - \begin{Bmatrix} \boldsymbol{R}_0|_{\Omega_0^\pm} + \boldsymbol{R}_0|_{\Gamma_0} \\ \boldsymbol{R}_1 \end{Bmatrix}.
\tag{12}
$$

Here, $\boldsymbol{K}_{00}|_{\Omega_0^\pm} = \partial \boldsymbol{R}_0|_{\Omega_0^\pm}/\partial^0\boldsymbol{u}$, $\boldsymbol{K}_{00}|_{\Gamma_0} = \partial \boldsymbol{R}_0|_{\Gamma_0}/\partial^0\boldsymbol{u}$, $\boldsymbol{K}_{01} = \partial \boldsymbol{R}_0|_{\Gamma_0}/\partial^1\boldsymbol{u}$, $\boldsymbol{K}_{10} = \partial\boldsymbol{R}_1/\partial^0\boldsymbol{u}$, and $\boldsymbol{K}_{11} = \partial\boldsymbol{R}_1/\partial^1\boldsymbol{u}$. In this work, the nonlinear system of equations is solved using a staggered approach. The discrete tangents are derived in Appendix A.

2.2. Nested iterative solution procedure

Monolithic solution of Equation 12 will achieve a quadratic rate of convergence, but becomes prohibitively expensive for detailed numerical simulations. Therefore, we solve the multiscale system of equations using a nested iterative procedure as suggested in [9, 10].

Algorithm 1 Microscale solution procedure.

Set $j \leftarrow 0$, ${}^1\boldsymbol{u}_{n+1}^{(j)} \leftarrow {}^1\boldsymbol{u}_n$, and $\Delta\,{}^1\boldsymbol{u}_{n+1}^{(j)} = \boldsymbol{0}$

Evaluate $\boldsymbol{R}_1({}^0\boldsymbol{u}_{n+1}^{(i)}, {}^1\boldsymbol{u}_{n+1}^{(j)})$

Evaluate $\boldsymbol{K}_{11}({}^0\boldsymbol{u}_{n+1}^{(i)}, {}^1\boldsymbol{u}_{n+1}^{(j)})$ by Equation (10)

while $\|\boldsymbol{R}_1^{(j)}\|/\|\boldsymbol{R}_1^{(0)}\| > \mathrm{tol}$ **do**

    Solve $\boldsymbol{K}_{11}^{(j)} \Delta\Delta\,{}^1\boldsymbol{u}_{n+1}^{(j+1)} = -\boldsymbol{R}_1^{(j)}$

    $\Delta\,{}^1\boldsymbol{u}_{n+1}^{(j+1)} = \Delta\,{}^1\boldsymbol{u}_{n+1}^{(j)} + \Delta\Delta\,{}^1\boldsymbol{u}_{n+1}^{(j+1)}$

    ${}^1\boldsymbol{u}_{n+1}^{(j+1)} \leftarrow {}^1\boldsymbol{u}_{n+1}^{(j)} + \Delta\,{}^1\boldsymbol{u}_{n+1}^{(j+1)}$

    Evaluate $\boldsymbol{R}_1({}^0\boldsymbol{u}_{n+1}^{(i)}, {}^1\boldsymbol{u}_{n+1}^{(j+1)})$ by Equation (10)

    $j \leftarrow j + 1$

end while

Evaluate $\boldsymbol{K}_{00}|_{\Gamma_0}$, $\boldsymbol{K}_{01}$, $\boldsymbol{K}_{10}$, $\boldsymbol{K}_{11}$, and $\boldsymbol{R}_0|_{\Gamma_0}$ for $({}^0\boldsymbol{u}_{n+1}^{(i)}, {}^1\boldsymbol{u}_{n+1})$

Solve $\boldsymbol{K}_{11}\mathscr{A} = \boldsymbol{K}_{10}$

Compute $\mathscr{K}|_{\Gamma_0} = \boldsymbol{K}_{00}|_{\Gamma_0} - \boldsymbol{K}_{01}\mathscr{A}$

Return $\mathscr{K}|_{\Gamma_0}$ and $\boldsymbol{R}_0|_{\Gamma_0}$ to the macroscale

---

Assume that all macro-variables and micro-variables at loading step n are known and equilibrated. Holding all macro-variables constant at iteration i, we first prescribe $[\![{}^0\boldsymbol{u}]\!]_{n+1}^{(i)}$ at the microscale. The microscale equilibrium is reached by Newton's algorithm as

$$
\boldsymbol{K}_{11}\left({}^0\boldsymbol{u}^{(i)}, {}^1\boldsymbol{u}^{(j)}\right) \Delta\Delta\,{}^1\boldsymbol{u}^{(j+1)} = -\boldsymbol{R}_1\left({}^0\boldsymbol{u}^{(i)}, {}^1\boldsymbol{u}^{(j)}\right),
\tag{13}
$$

where $n+1$ subscripts are implied. Algorithm 1 provides the detailed microscale solution procedure. Given the converged solution of the microscale problem in Equation 13, we proceed to compute the macroscale equilibrium

$$
\mathscr{K}^{(i)} \Delta\Delta\,{}^0\boldsymbol{u}^{(i+1)} = -\boldsymbol{R}_0\left({}^0\boldsymbol{u}^{(i)}, {}^1\boldsymbol{u}\right) + \underbrace{\boldsymbol{K}_{01}\boldsymbol{K}_{11}^{-1}\boldsymbol{R}_1\left({}^0\boldsymbol{u}^{(i)}, {}^1\boldsymbol{u}\right)}_{=\boldsymbol{0}},
\tag{14}
$$

where $n+1$ subscripts are implied and ${}^1\boldsymbol{u}$ is the equilibrated microscale solution. The Schur complement evaluated for $\left({}^0\boldsymbol{u}^{(i)}, {}^1\boldsymbol{u}\right)$ reads

$$
\mathscr{K} = \boldsymbol{K}_{00}|_{\Omega_0^\pm} + \underbrace{\boldsymbol{K}_{00}|_{\Gamma_0} - \boldsymbol{K}_{01}\boldsymbol{K}_{11}^{-1}\boldsymbol{K}_{10}}_{\mathscr{K}|_{\Gamma_0}}.
\tag{15}
$$

Note that the Schur complement involves matrix operations with large mixed-scale matrices. Furthermore, the direct factors of $\boldsymbol{K}_{11}^{-1}$ are too expensive to compute and store. Thus, we first solve

the matrix system of equations $\boldsymbol{K}_{11}\mathscr{A} = \boldsymbol{K}_{10}$ by successive calls to the sparse iterative linear solver (we use HYPRE [26]). Then, we compute $\mathscr{K}|_{\Gamma_0} = \boldsymbol{K}_{00}|_{\Gamma_0} - \boldsymbol{K}_{01}\mathscr{A}$ by a simple matrix product and subtraction. Both $\boldsymbol{K}_{00}|_{\Gamma_0}$ and $\boldsymbol{R}_0|_{\Gamma_0}$ are constructed at the microscale along with $\boldsymbol{K}_{01}$ and $\boldsymbol{K}_{10}$. Algorithm 2 gives the nested iterative solution procedure.

Algorithm 2 Nested iterative solution procedure.

Initialization
   Assign a RUC to each macroscopic integration point in Γ_0
   Compute $\mathscr{K}|_{\Gamma_0}$ and $\boldsymbol{R}_0|_{\Gamma_0}$ under no load (Algorithm 1)
   Compute element tangents $\boldsymbol{K}_{00}|_{\Omega_0^\pm}$

Set $i \leftarrow 0$, ${}^0\boldsymbol{u}_{n+1}^{(i)} \leftarrow {}^0\boldsymbol{u}_n$, and $\Delta\, {}^0\boldsymbol{u}_{n+1}^{(i)} = \boldsymbol{0}$
Increment load
Evaluate $\boldsymbol{R}_0|_{\Omega_0^\pm}\,({}^0\boldsymbol{u}_{n+1}^{(i)}, {}^1\boldsymbol{u}_{n+1})$ by Equation (9)
while $\|\boldsymbol{R}_0^{(i)}\|/\|\boldsymbol{R}_0^{(0)}\| > \mathrm{TOL}$ **do**
   Assemble full tangent $\mathscr{K}_{n+1}^{(i)}$
   Solve $\mathscr{K}_{n+1}^{(i)} \Delta\Delta\, {}^0\boldsymbol{u}_{n+1}^{(i+1)} = -\boldsymbol{R}_0^{(i)}$
   $\Delta\, {}^0\boldsymbol{u}_{n+1}^{(i+1)} = \Delta\, {}^0\boldsymbol{u}_{n+1}^{(i)} + \Delta\Delta\, {}^0\boldsymbol{u}_{n+1}^{(i+1)}$
   ${}^0\boldsymbol{u}_{n+1}^{(i+1)} \leftarrow {}^0\boldsymbol{u}_{n+1}^{(i)} + \Delta\, {}^0\boldsymbol{u}_{n+1}^{(i+1)}$
   Compute $\mathscr{K}|_{\Gamma_0}$ and $\boldsymbol{R}_0|_{\Gamma_0}$ due to ${}^0\boldsymbol{u}_{n+1}^{(i+1)}$ (Algorithm 1)
   Evaluate $\boldsymbol{R}_0^{(i+1)} = \boldsymbol{R}_0^{(i+1)}|_{\Omega_0^\pm} + \boldsymbol{R}_0|_{\Gamma_0}$ (Equation (9))
   $i \leftarrow i + 1$
end while
${}^0\boldsymbol{u}_n \leftarrow {}^0\boldsymbol{u}_{n+1}$
For each cohesive integration point (each RUC), ${}^1\boldsymbol{u}_n \leftarrow {}^1\boldsymbol{u}_{n+1}$

---

3. PARALLEL COMPUTATIONAL IMPLEMENTATION

In this section, we describe our new hierarchically parallel implementation that is capable of computing large multiscale problems. We use our in-house highly scalable finite element library *PGFem3D*. The hierarchically parallel multiscale solver is based on a non-blocking client-server communication structure that allows overlay of computations at the different scales. Both *PGFem3D* and the hierarchically parallel multiscale solver are implemented using the Message Passing Interface (MPI).

3.1. Parallel finite element library, PGFem3D

The finite element library *PGFem3D* is based on a traditional Domain Decomposition (DD) concept [7]. The finite element domain is decomposed into N_p parts using the parallel graph partitioning algorithm ParMETIS [27]. The decomposition minimizes the number of shared nodes while maintaining a nearly equal number of elements in each domain. A sparse non-blocking point-to-point communication structure based on the decomposition is then constructed for efficient computation and assembly of the distributed global system of equations. The global system of equations is solved using the HYPRE library [26]. We use the GMRES solver paired with the Euclid preconditioner [28]. Figure 2 shows weak scaling of the *PGFem3D* parallel finite element library for a single scale simulation. On each core, we use a block with a spherical void loaded by uniform void growth. The 1024 voids/cores domain contains 11.97 million nodes, 61.6 million elements, and 28.9 million DOFs. Each simulation computes four nonlinear loading steps with a total of 12 linear solves. Thus, we perform one linear finite element solve (including assembly, etc.) consisting of 28.9 million equations in $\sim 8.69\,\mathrm{s}$ on 1024 cores. Figure 2 shows only a 26% increase in execution time over a 64-fold increase in computing cores. This small increase in execution time over the ideal scaling is partially due to the GMRES solver.

3.2. Hierarchically parallel multiscale solver

The hierarchically parallel multiscale solver uses the client-server communication structure outlined schematically in Figure 3 to link each scale. Both the macroscale and microscale are solved using the parallel finite element library *PGFem3D*. As shown in Figure 3(a), the macroscale (adherends) are decomposed into N_p domains. Each domain is computed on a single computing core or macroscale 'client' (Figure 3(a)). The microscale solutions are computed on N_s groups of computing cores or 'servers' (Figure 3(b)). Each microscale server is composed of n_p cores and evaluates the response from $c_{n,s}$ RUCs that are assigned by the macroscale clients. Similarly to the macroscale DD, the RUC is decomposed to n_p domains that are executed on a particular server using n_p cores as described earlier. This allows us to exploit the hierarchical parallelism stemming from the multiscale theory derived in section 2.

We assign RUCs to the servers using round-robin scheduling to balance the load between microscale servers and to reduce data starvation at the macroscale. Figure 3(c) shows the communication pattern between the clients and servers. As shown, each server consists of n_p cores,

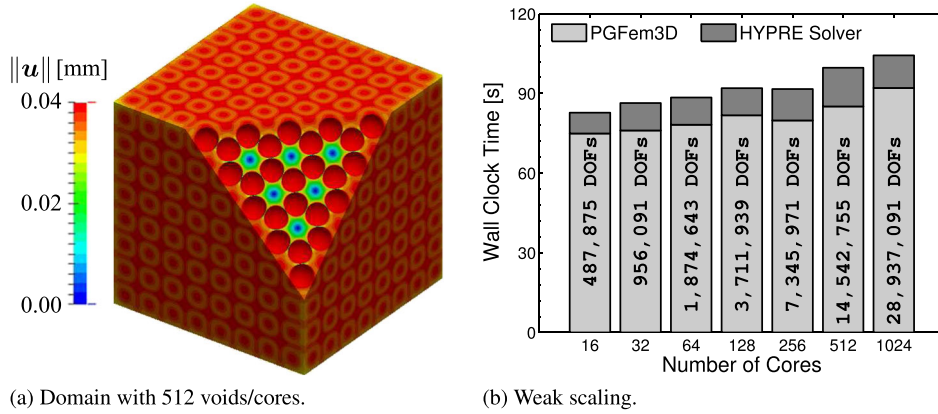(a) Domain with 512 voids/cores.　　　　(b) Weak scaling.

Figure 2. Weak scaling of the parallel finite element library *PGFem3D*. The number of finite elements per computing core is held constant for all simulations. (a) shows the computational domain containing 512 voids loaded by uniform void growth computed on 512 cores. (b) shows the weak scaling up to 1024 cores.
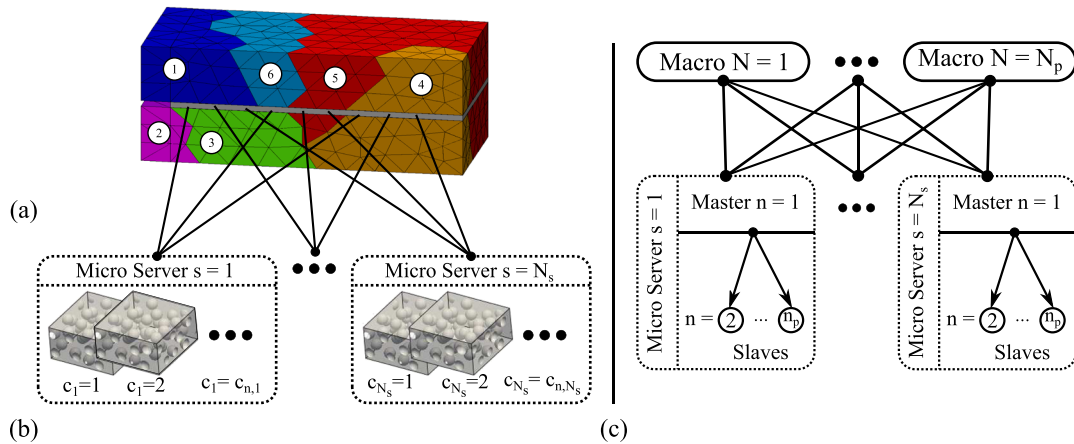

Figure 3. The hierarchically parallel communication pattern. (a) shows the macroscale adherends decomposed into $N_p = 6$ domains (clients) that are computed on six cores. (b) shows the N_s microscale servers s that compute the microscale response from $c_{n,s}$ RUCs each. Each RUC is decomposed into n_p domains and computed using all n_p cores in the microscale server s. (c) shows the hierarchically parallel communication between N_p macroscale processors and the N_s master cores of each server s containing n_p cores each.
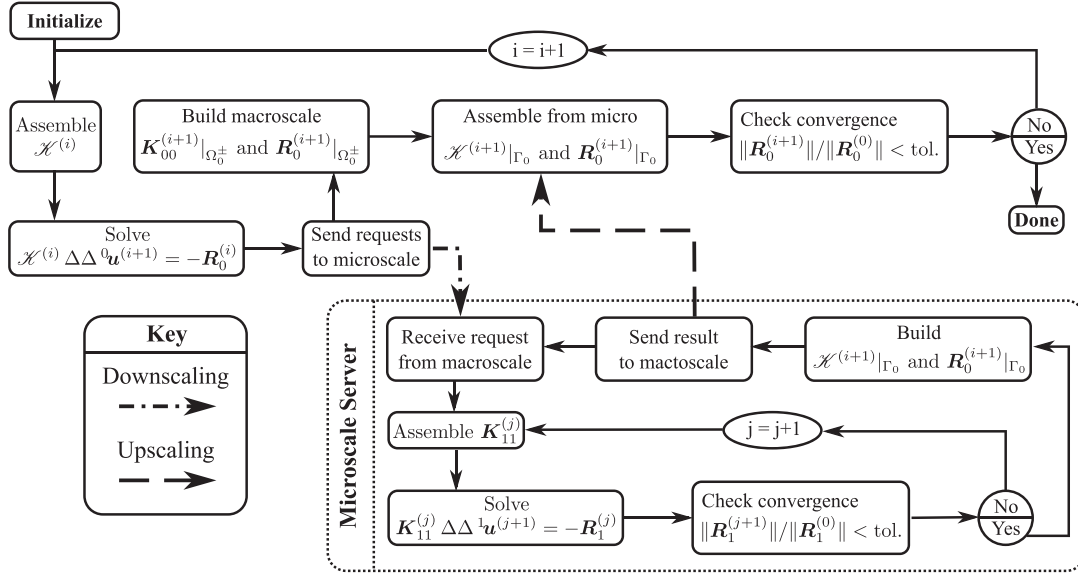
Figure 4. The hierarchically parallel implementation of the nested iterative solution procedure given in Algorithm 2. Macroscale and microscale computations are overlaid via non-blocking point-to-point communication patterns. Note that there are N_s microscale servers that simultaneously compute microscale contributions for the N_p macroscale clients.

with $n = 1$ designated the 'master' and the remaining $n_p - 1$ cores designated as 'slaves'. The clients and servers communicate only through the master cores, but all clients can communicate with all servers and vice versa. Communication between clients and servers is performed using point-to-point non-blocking send and receives. This non-blocking communication pattern enables us to overlay computation and communication at both scales and thereby reduce latency and overhead. Figure 4 depicts a flowchart of the hierarchically parallel implementation of the nested iterative solution procedure using the client-server communication structure. After solving the linearized macroscale iteration, the macroscale client loops over all cohesive elements and sends requests to the assigned servers to compute the microscale response from $\llbracket^0 u \rrbracket$. After sending all of the requests, the macroscale clients compute the element tangent and residuals for the adherends ($K_{00}|_{\Omega_0^\pm}$ and $R_0|_{\Omega_0^\pm}$). The master on each server processes requests from the clients as they arrive. When a request is received, the master broadcasts the information to the waiting slaves. Simultaneously with the macroscale, the microscale servers compute the response from $\llbracket^0 u \rrbracket$ on their lists of $c_{n,s}$ RUCs as described in Algorithm 1. Upon equilibrium, the contributions to the macroscale tangent and residual ($\mathscr{K}|_{\Gamma_0}$ and $R_0|_{\Gamma_0}$) are computed and condensed on the master. The master sends a response to the client with the computed results and moves on to process the next request. When the macroscale clients finish computing the tangent and residual for the adherends, they process the responses from the microscale servers as they arrive. After assembling the contributions from the microscale servers, the relative norm of the residual is checked for convergence.

4. NUMERICAL EXAMPLES

In this section, we present three numerical examples using the hierarchically parallel multiscale solver for modeling nonlinear response of a heterogeneous hyperelastic layer. The first example is a simple verification problem and convergence study. The second example shows the strong scaling performance of the hierarchically parallel multiscale solver, using up to 2048 processing cores. The last example shows the solver's ability to perform large and detailed engineering simulations that would unlikely be possible by DNM with the same computational resources. In particular, we perform a multiscale simulation with ~ 1.1 billion elements and ~ 575 million nonlinear equations on only 1552 computing cores. Similarly sized simulations presented in the literature [4] are typically

computed on several thousand computing cores. In all examples, we use hyperelastic material potentials for both scales given by

$$^0W \equiv {}^1W = \frac{G}{2}\left(\mathrm{tr}\hat{\boldsymbol{C}} - 3\right) + \frac{E}{6(1-2\nu)}\left[\exp(J-1) - \ln(J) - 1\right], \qquad (16)$$

where E is Young's modulus, G is the shear modulus, and ν is Poisson's ratio. The Jacobian of the deformation is given by $J = \det(\boldsymbol{C})^{1/2}$, and $\hat{\boldsymbol{C}} = J^{-2/3}\boldsymbol{C}$ is the deviatoric right Cauchy-Green deformation tensor. Note that the appropriate macroscale and microscale deformation tensors ($^0\boldsymbol{C}$ and $\boldsymbol{C}$, respectively) are used in Equation 16 to evaluate the macroscale and microscale potential functions, ^0W and ^1W, respectively. The material properties for the adherends and interface are listed in Table I.

We note that the theory presented in Section 2 relies on the assumption of both field and material $Y_{1,2}$-periodicity. This assumption is often unsatisfied due to the macroscopic boundary data, and boundary correction terms are required at the microscale [29]. However, the boundary correction terms are usually neglected in engineering practice because of their exponential decay away from the boundary [30]. In this work, we follow this common simplification and neglect the boundary correction terms. The effect of this simplification on fully coupled 2D multiscale cohesive simulations has been shown in our previous work [10].

4.1. Verification study

For the verification study, we compute the force-displacement response of the domain shown in Figure 5 by DNM and FE^2. This example is presented purely as a numerical test to verify the correctness of the FE^2 implementation, consistent with practices presented in [31]. The problem consists of two unit cube adherends ($L = 1$ mm) connected by a heterogeneous layer with thickness $l_c = 0.0625$ mm (Figure 5(a)). The top adherend is loaded in the vertical direction by an incremental displacement, δ, and all other surfaces are constrained in the normal directions. In the FE^2 simulation, the layer with thickness l_c is collapsed to a surface.

To maintain a reasonable computational size for the finest resolution DNM simulation, the heterogeneous layer (Figure 5(b)) is constructed by a tessellation of 16 copies of the simple computational cell shown in Figure 5(c). The computational cell contains ten 40 μm diameter voids ($c_v = 8.58\%$). The void centers relative to the cell center are listed in Table II. The overall computational cell dimensions are $l_c = 0.0625$ mm and $l_{RUC} = 0.25$ mm. The macroscale domain for FE^2 is discretized with 16 cohesive elements. Thus, the overall heterogeneous layer for FE^2 modeling also contains 16 computational cells and is geometrically equivalent to the DNM heterogeneous layer shown in Figure 5(b).

We perform the verification study using three different mesh refinements. For DNM, the mesh refinement is focused inside the layer. However, the adherends are also refined to maintain mesh continuity. For FE^2, only the discretization of the computational cell is refined. The FE^2 mesh of the adherends remains constant for all levels of refinement and contains 1038 nodes, 4478 elements, and 2524 DOFs. Table III shows the mesh size for one cell at each level of refinement in the FE^2 study. Table IV shows the total mesh size for the three levels of refinement. The total number of elements in the FE^2 and DNM microstructures are held to within 3% of each other. Figure 6 shows the force-displacement response computed by DNM and FE^2 for the three mesh refinements described in Table IV. There is good agreement between the DNM and FE^2 response for all levels of refinement, and the solutions converge together as the mesh is refined within the layer. Notice

Table I. Material properties for all numerical examples.

	Young's modulus E (MPa)	Shear modulus G (MPa)	Poisson's ratio ν (−)
Adherends	$15 \cdot 10^4$	$6.000 \cdot 10^4$	0.25
Interface	$5 \cdot 10^3$	$1.866 \cdot 10^3$	0.34

Displacement loading, δ

(a) DNM dimensions and boundary conditions.
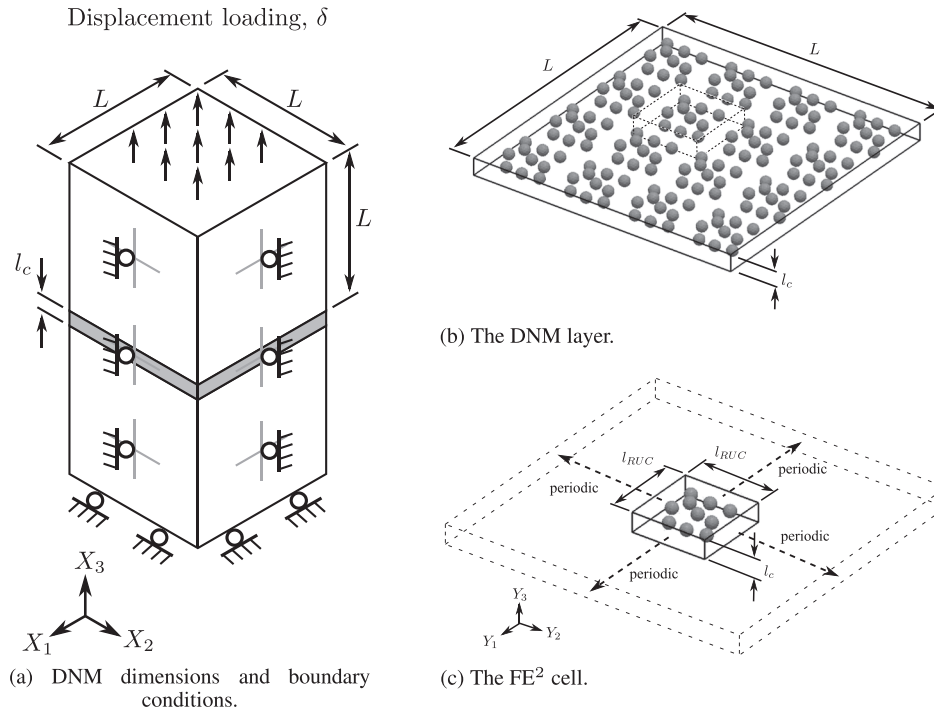
(b) The DNM layer.

(c) The FE^2 cell.

Figure 5. Verification problem computed by DNM and FE^2. (a) shows adherends with $L = 1$ mm connected by a heterogeneous layer of thickness $l_c = 0.0625$ mm. (b) shows the microscale layer for the DNM simulation based on a regular tessellation of the microstructure identified by the dotted box. (c) shows the cell used in the FE^2 simulation with $l_{RUC} = 0.25$ mm containing 10 voids with 40 μm diameter. Note that the FE^2 simulation uses 16 computational cells, one for each macroscale cohesive element.

Table II. Coordinates of the void centers relative to the cell geometric center in μm (Figure 5(c)).

	Void Number									
	1	2	3	4	5	6	7	8	9	10
Y_1	65.718	62.821	−96.900	−10.177	−90.417	61.285	−15.104	−6.424	−92.506	−53.979
Y_2	18.430	−44.074	−86.026	−87.556	−15.196	84.004	−2.400	51.745	51.484	−58.404
Y_3	−2.262	−2.815	−3.408	−5.185	−3.795	0.808	1.981	−1.770	−6.055	−0.081

Table III. Mesh size for each refinement of one computational cell in the FE^2 verification study.

	Nodes	Elements
Coarse	7298	36,688
Medium	49,672	273,013
Fine	375,275	2,166,525

the highly nonlinear character of the solution. Figure 7 compares the effective Eulerian/Almansi strain $\|e\| = \|1/2 \left[\mathbf{1} - (\mathbf{F}\,\mathbf{F}^{\mathrm{T}})^{-1} \right]\|$ in an interior region of the DNM layer (highlighted by the dotted box in Figure 5(b)) and a FE^2 computational cell at the end of the load history. The deformed microstructures are clipped at the mid-plane to view the interior strains. The deformation within the interior of the DNM layer is very similar to that of the computational cell used in the FE^2 simulation. Notice the very large strains ($\sim 300\%$) in highly refined ligaments between voids, confirming the requirement of finite strain simulations. We also note that neglecting the boundary correction terms in the FE^2 simulation (i.e., neglecting field and material periodicity at the microscale boundary) does not introduce a noticeable error in both the macroscopic and microscopic response because

Table IV. Mesh sizes for three levels of refinement of the verification problem in Figure 5.

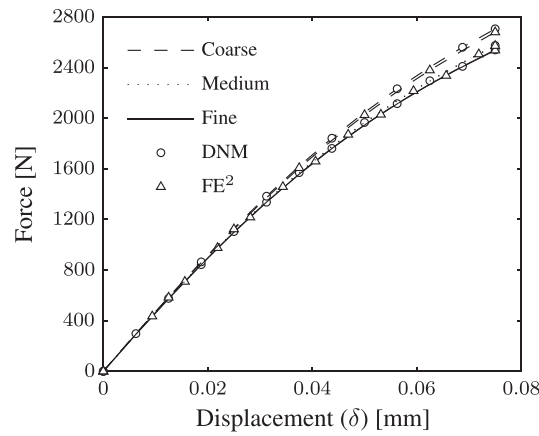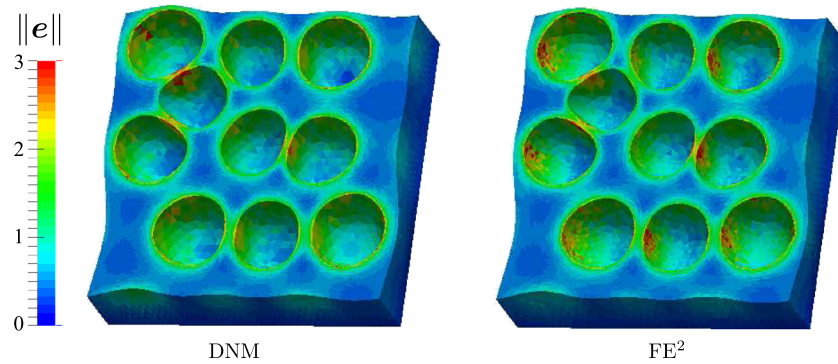|	Total (Macroscale + Microscale) size			Microscale size	
	Nodes	Elements	DOFs	Nodes	Elements
Coarse					
DNM	225,008	1,276,095	666,498	125,623	579,382
FE^2	117,806	591,486	288,988	116,768	587,008
Medium					
DNM	1,245,614	7,175,565	3,715,116	913,672	4,489,367
FE^2	795,790	4,372,686	2,225,644	794,752	4,368,208
Fine					
DNM	6,892,081	39,929,638	20,616,407	6,771,950	34,545,812
FE^2	6,005,438	34,668,878	17,451,964	6,004,400	34,664,400

Figure 6. Comparison of force-displacement response between DNM and FE^2 for the meshes in Table IV.

Figure 7. Effective Almansi strain in the mid-plane of the deformed heterogeneous layer. The cell-labeled DNM is cropped from the full layer as shown in Figure 5(b).

of the homogeneous macroscopic boundary conditions. Finally, we mention that there is savings in terms of the number of DOFs for similar accuracy in the solution when using the FE^2 method over DNM (Table IV).

4.2. Scaling performance

As alluded to in Section 3.2, the hierarchically parallel client-server execution model takes full advantage of the parallelism inherent in the multiscale theory derived in Section 2. This leads to
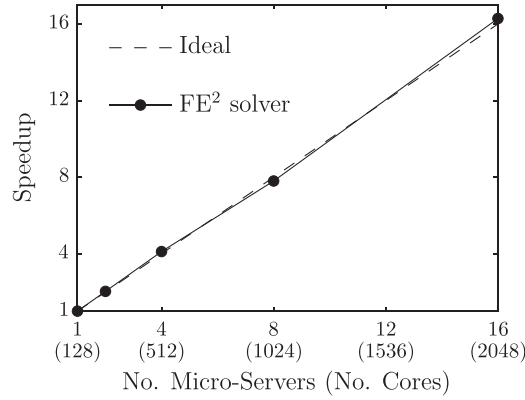
Figure 8. Speedup of the hierarchically parallel fully coupled multiscale solver with respect to the number of microscale servers.
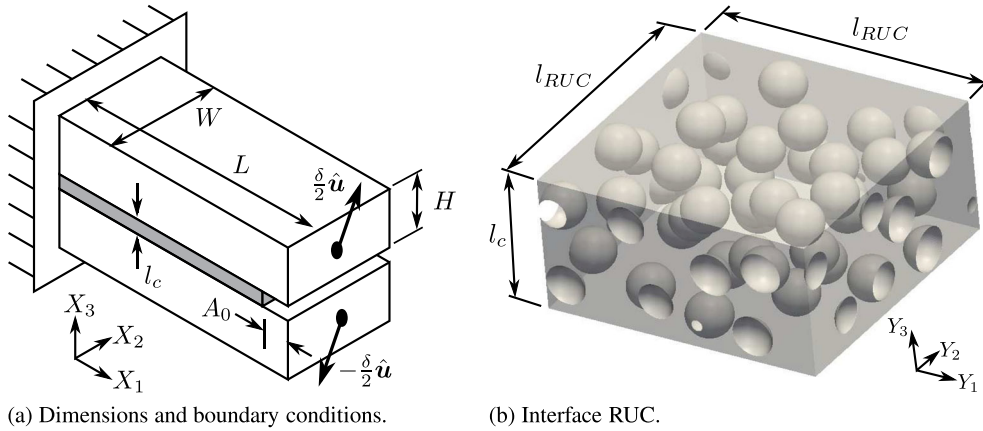

(a) Dimensions and boundary conditions.

(b) Interface RUC.

Figure 9. Dimensions, loading conditions, and microscale morphology for the 1.1 billion element FE^2 simulation. (a) shows the dimensions and loading conditions of the macroscale adherends. (b) shows the RUC for the heterogeneous layer containing 40 arbitrarily located 40 μm diameter voids ($c_v = 17.17\%$).

significant gains in computational performance, particularly in terms of strong scaling speedup. We measure the speedup of our implementation by timing the first loading step of the finest FE^2 simulation described in Section 4.1. Traditionally, a single processing core is used as the unit of computational resources for speedup analysis. Here, we use a microscale server as the unit of resources, with each server consisting of 128 processing cores. An additional 16 cores are used to solve the macroscale problem for all cases. Figure 8 shows that the hierarchically parallel multiscale solver exhibits ideal speedup to 2048 cores (16 servers computing one cell each). The computations were performed on up to 129 IBM nodes, each with dual eight-core Intel Xeon 2.60 GHz processors and 32 GB of RAM, connected by Mellanox FDR InfiniBand.

4.3. 1.1 Billion element FE^2 simulation

We now present the capability of the hierarchically parallel FE^2 solver by simulating a complex multiscale nonlinear example devoted to hyperelastic porous interfaces. Figure 9(a) shows the dual cantilever beam (DCB) specimen loaded by mixed-mode opening displacement. Table V provides the macroscale dimensions and loading direction. The RUC (Figure 9(b)) is $l_c = 0.125$ mm thick, $l_{RUC} = 0.25$ mm wide and contains 40 arbitrarily placed 40 μm diameter voids ($c_v = 17.17\%$). The microstructure is generated using a random sequential addition algorithm (see, e.g., [32]). More complex microstructures may be generated using the packing algorithm described in [33], for example. Figure 10 shows the in-plane isotropic two-point probability functions, S_{rs}, computed

Table V. Dimensions and the loading direction for the DCB adherends shown in Figure 9(b).

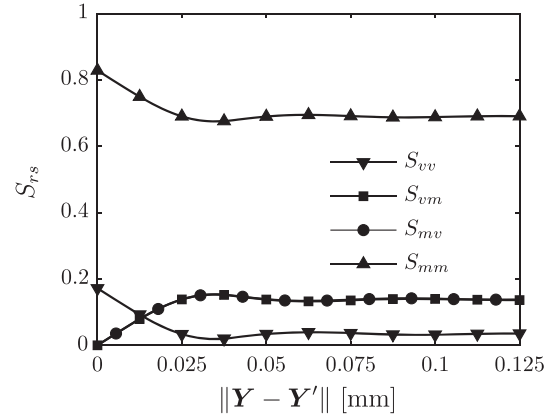| L (mm)	W (mm)	H (mm)	A_0 (mm)	l_c (mm)	Loading direction, $\hat{\boldsymbol{u}}$
22.0	10.0	5.0	2.0	0.125	$\{1/\sqrt{3},\ 1/\sqrt{3},\ 1/\sqrt{3}\}^{\mathrm{T}}$

Figure 10. The in-plane isotropic two-point probability functions for the RUC shown in Figure 9(b). The subscripts m and v denote the matrix and void phases, respectively.

Table VI. Number of nodes, elements, and DOFs for the DCB FE^2 example. Each RUC in the simulation has 1.6 million nodes, 9.2 million elements, and 4.8 million DOFs.

	Nodes	Elements	DOFs
Macroscale	731	2684	1878
Microscale	193,873,920	1,098,283,920	574,612,560
TOTAL	193,874,651	1,098,286,604	574,614,438

using our statistical sampling tool *Stat3D* [34]. The saturation of the probability functions shows that the RUC is sufficiently large to capture the geometric characteristics of the material and that the microstructure has no long-range order. Because the focus of this paper is on the development of the numerical method, we do not investigate material representativeness of this cell [35].

The size ratio between macroscale and microscale domains is $L/l_c = 160$ and $L/l_{RUC} = 80$, respectively. The ratio between macrosize and void size is $L/d = 500$. The macroscale is discretized such that the effective area of the cohesive element is $\sim 27\times$ larger than that of the RUC, l_{RUC}^2. The length of the cohesive element is $\sim 5\times$ larger than the size of RUC, l_{RUC}. This provides a good separation of scales, yet the coupled simulation still resolves the physics of the complete multiscale problem since the macroscopic discretization is sufficiently refined (the cohesive element size is ~ 1.96 mm). The resulting interface discretization contains 120 cohesive elements, corresponding to 120 microstructures. The RUC is discretized with a similar resolution as the finest mesh in the verification study of Section 4.1. Within the RUC, the nominal element size ranges from $h_{\min} = 0.05\ \mu$m to $h_{\max} = 2.922\ \mu$m, with $h_{\mathrm{mean}} = 1.713\ \mu$m. Thus, in this simulation, we resolve $L/h_{\min} = \mathcal{O}(10^6)$ in length-scales. The mesh characteristics for the macroscale and microscale domains are given in Table VI. Note that we again neglect the boundary correction terms at the microscale boundary due to the non-periodic fields arising from non-periodic boundary macro-data.

The macroscale problem is computed on 16 processing cores, while the microscale solutions are computed on 12 servers responsible for 10 RUCs each. Each microscale server employs $n_p = 128$ processing cores. Therefore, we use $12 \cdot 128 + 16 = 1552$ total processing cores to simulate the fully coupled multiscale problem. We are effectively computing the nonlinear solution of $\sim 370,241$

(a) The deformed adherends (magnified $10\times$).
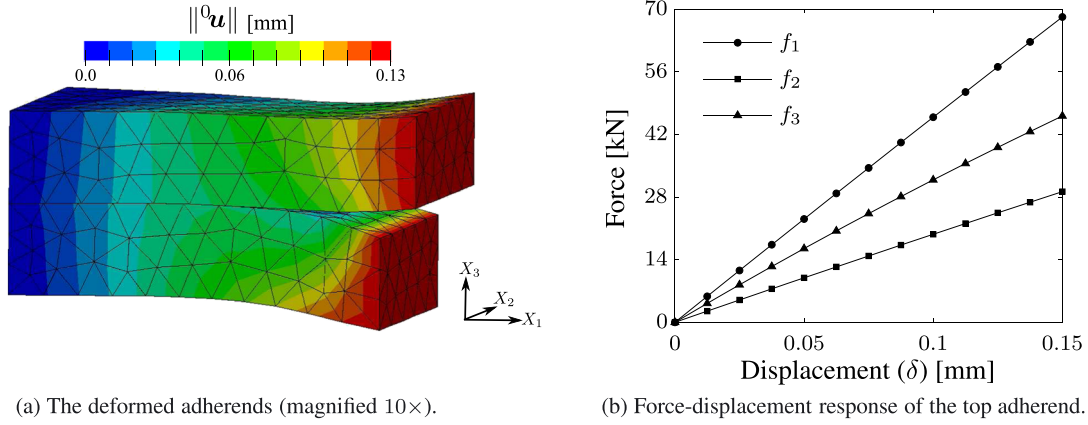

(b) Force-displacement response of the top adherend.

Figure 11. Multiscale solution of the DCB problem with a heterogeneous interface. (a) shows contours of displacement magnitude on the deformed adherends at the end of the load history. The deformation is magnified $10\times$. (b) shows the nonlinear force-displacement response of the free end of the top adherend.

(a) Macroscale cohesive/interface mesh.

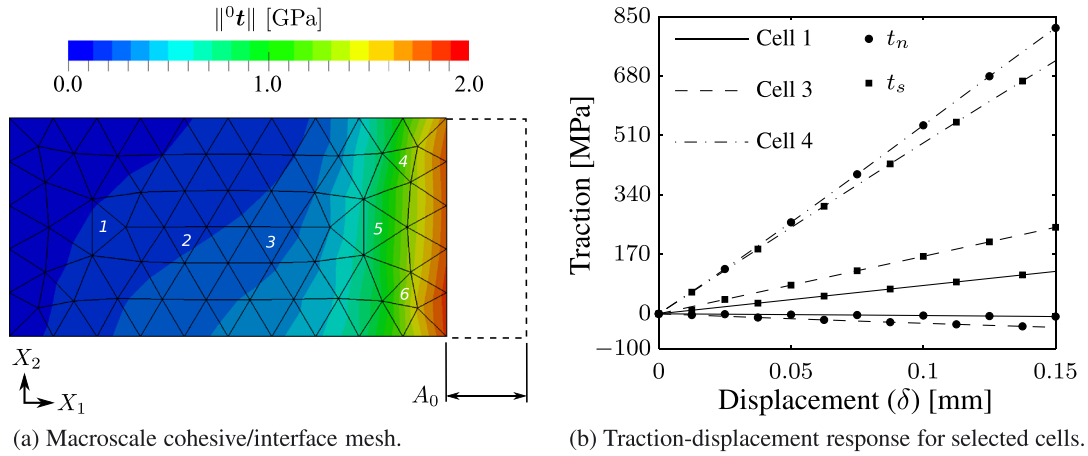(b) Traction-displacement response for selected cells.

Figure 12. Macroscopic response of the interface. (a) shows the magnitude of traction vector with the locations of microstructures used in the analysis. The distance A_0 displays the pre-crack. (b) shows the normal and shear components of the traction vector for the cohesive elements 1, 3, and 4.

DOFs per core using our hierarchically parallel multiscale solver. The solution was computed in less than 48 hours using the same hardware as in Section 4.2. We note that similar computations usually require several thousand processing cores [4, 6]. Figure 11 shows the deformed adherends at the end of the load history and the nonlinear force-displacement response of the free end of the top adherend. Figure 11(b) shows that the shear responses, f_1 and f_2, are not equal. This is due to the complex mixed-mode loading conditions (Figure 9(a)) resulting in compressive loading of the bottom adherend and tensile loading of the top adherend in the X_1-direction (Figure 11(a)).

Figure 12 shows the macroscopic traction-displacement (δ) response of the interface. Figure 12(a) shows the non-uniform distribution of the magnitude of traction on the interface at the end of the loading history. Figure 12(b) shows normal and shear components of the macroscopic traction-displacement response of selected microstructures. The normal traction is given by $t_n = {}^0t_3$, and the shear traction is evaluated as $t_s = \sqrt{{}^0t_1^2 + {}^0t_2^2}$. Figure 12(b) shows that microstructures away from the applied opening (cells 1 and 3) experience compressive loading ($t_n < 0$), while cells near the applied opening experience large tensile loading. Figure 12(b) also shows that the compressive normal traction in cell 3 is higher than in cell 1. This is expected as the compressive normal traction tends to zero at the far end from the opening. Figure 13 shows the effective Eulerian/Almansi strain, $\|e\|$, in the microstructures corresponding to the six marked interface elements in Figure 12(a).
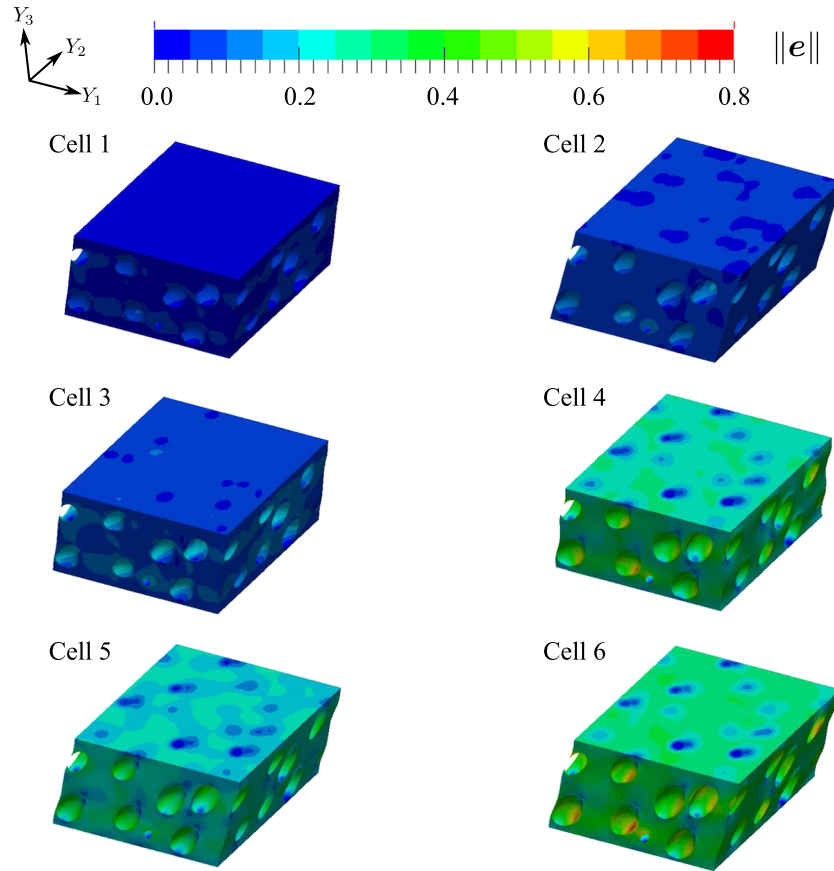
Figure 13. Effective Eulerian/Almansi strain in the microstructures at marked points on the macroscale interface.

The high levels of strain detected (80%) show the importance of finite strain modeling. We also note that strains in cells 4 and 6 are not symmetric because of the mixed-mode loading. As expected, the strains substantially increase for cells closer to the leading edge where opening displacement is applied (Figures 9(a) and 12(a)).

5. CONCLUSIONS

We have developed a fully coupled, multiscale cohesive solver for modeling heterogeneous layers. The hierarchically parallel multiscale solver is based on a point-to-point non-blocking client-server communication scheme and is scalable to many thousands of distributed computing cores. We show that the well established FE^2 method can be practical to accurately simulate engineering scale domains with resolution from $\mathcal{O}(10^1)$ nm to $\mathcal{O}(10^1)$ mm ($\mathcal{O}(10^6)$ in spatial scales). Engineering devices of this size can easily be tested experimentally to provide validation data for the new hierarchically parallel FE^2 multiscale framework. Such a validated framework lends itself to predictive scientific studies and *Virtual Materials Testing* in order to develop reduced order models for application in industry-ready modeling and simulation tools. A rigorous verification and mesh refinement study is performed and shows excellent agreement between DNM and FE^2 solutions. Using the verified multiscale framework, we solve an example with over 1 billion finite elements and over 574 million nonlinear equations on only 1552 computing cores. The solver provides the nonlinear solution to $\sim 370,241$ unknowns per processing core. Additionally, the proposed high-performance solver shows excellent strong scaling performance. With this parallel multiscale solver in hand, future studies can now be performed to investigate the influence of boundary conditions as well as microstructure morphology on the overall behavior of bonded systems.

APPENDIX A: MULTISCALE DISCRETE TANGENTS

The discrete multiscale tangents are derived by linearizing Equations (9) and (10) with respect to the macroscale and microscale variables. The macroscale tangent

$$
\boldsymbol{K}_{00} = \overset{\Omega_0^\pm}{\underset{e}{\mathbf{A}}} \left\{ \int_{\Omega_0^{\pm,e}} {}^0\boldsymbol{S} : \left[{}^0\tilde{\boldsymbol{B}}^{\mathrm{T}0}\tilde{\boldsymbol{B}} \right]^{\mathrm{sym}} \mathrm{d}V + \int_{\Omega_0^{\pm,e}} \left[{}^0\boldsymbol{F}^{\mathrm{T}0}\tilde{\boldsymbol{B}} \right]^{\mathrm{sym}} : {}^0\mathbb{L} : \left[{}^0\boldsymbol{F}^{\mathrm{T}0}\tilde{\boldsymbol{B}} \right]^{\mathrm{sym}} \mathrm{d}V \right\}
$$
$$
+ \overset{\Gamma_0}{\underset{e}{\mathbf{A}}} \int_{\Gamma_0^e} {}^\Gamma\tilde{\boldsymbol{N}}^{\mathrm{T}} \frac{\partial {}^0\boldsymbol{t}}{\partial [\![{}^0\boldsymbol{u}]\!]} {}^\Gamma\tilde{\boldsymbol{N}} \mathrm{d}A,
$$
(A.1)

is obtained by differentiating Equation 9 with respect to the macroscale displacements. In Equation A.1, ${}^0\mathbb{L} = 2\partial {}^0\boldsymbol{S}/\partial {}^0\boldsymbol{C}$ is the material tangent stiffness at the macroscale and

$$
\frac{\partial {}^0\boldsymbol{t}}{\partial [\![{}^0\boldsymbol{u}]\!]} {}^\Gamma\tilde{\boldsymbol{N}} = \frac{1}{l_c |\Theta_0|} \overset{\Theta_0}{\underset{e}{\mathbf{A}}} \left\{ \int_{\Theta_0^e} \left[{}^\Gamma\tilde{\boldsymbol{N}} \otimes {}^0\boldsymbol{N} \right] {}^1\boldsymbol{S} \, \mathrm{d}V + \int_{\Theta_0^e} \boldsymbol{F} {}^1\mathbb{L} : \left[\boldsymbol{F}^{\mathrm{T}} \left({}^\Gamma\tilde{\boldsymbol{N}} \otimes {}^0\boldsymbol{N} \right) \right]^{\mathrm{sym}} \mathrm{d}V \right\} \cdot {}^0\boldsymbol{N}
$$
(A.2)

is obtained by differentiation of Equation 11, where ${}^1\mathbb{L} = 2\partial {}^1\boldsymbol{S}/\partial \boldsymbol{C}$ is the material tangent stiffness at the microscale. The macro-to-micro tangent

$$
\boldsymbol{K}_{01} = \overset{\Gamma_0}{\underset{e}{\mathbf{A}}} \int_{\Gamma_0^e} {}^\Gamma\tilde{\boldsymbol{N}}^{\mathrm{T}} \frac{\partial {}^0\boldsymbol{t}}{\partial {}^1\boldsymbol{u}} {}^1\tilde{\boldsymbol{N}} \mathrm{d}A,
$$
(A.3)

is derived by differentiating Equation 9 with respect to the microvariables, where

$$
\frac{\partial {}^0\boldsymbol{t}}{\partial {}^1\boldsymbol{u}} {}^1\tilde{\boldsymbol{N}} = \frac{1}{|\Theta_0|} \overset{\Theta_0}{\underset{e}{\mathbf{A}}} \left\{ \int_{\Theta_0^e} {}^1\tilde{\boldsymbol{B}} {}^1\boldsymbol{S} \, \mathrm{d}V + \int_{\Theta_0^e} \boldsymbol{F} {}^1\mathbb{L} : \left[\boldsymbol{F}^{\mathrm{T}1}\tilde{\boldsymbol{B}} \right]^{\mathrm{sym}} \mathrm{d}V \right\} \cdot {}^0\boldsymbol{N}
$$
(A.4)

is obtained by differentiation of Equation 11.

The microscale tangent

$$
\boldsymbol{K}_{11} = \frac{l_c}{|\Theta_0|} \overset{\Theta_0}{\underset{e}{\mathbf{A}}} \int_{\Theta_0^e} {}^1\boldsymbol{S} : \left[{}^1\tilde{\boldsymbol{B}}^{\mathrm{T}1}\tilde{\boldsymbol{B}} \right]^{\mathrm{sym}} \mathrm{d}V + \int_{\Theta_0^e} \left[\boldsymbol{F}^{\mathrm{T}1}\tilde{\boldsymbol{B}} \right]^{\mathrm{sym}} : {}^1\mathbb{L} : \left[\boldsymbol{F}^{\mathrm{T}1}\tilde{\boldsymbol{B}} \right]^{\mathrm{sym}} \mathrm{d}V,
$$
(A.5)

is computed by differentiating Equation 10 with respect to the microscale displacements. Finally, the micro-to-macro tangent yields

$$
\boldsymbol{K}_{10} = \frac{1}{|\Theta_0|} \overset{\Theta_0}{\underset{e}{\mathbf{A}}} \left\{ \int_{\Theta_0^e} {}^1\boldsymbol{S} : \left[\left({}^\Gamma\tilde{\boldsymbol{N}} \otimes {}^0\boldsymbol{N} \right) {}^1\tilde{\boldsymbol{B}} \right]^{\mathrm{sym}} \mathrm{d}V \right.
$$
$$
\left. + \int_{\Theta_0^e} \left[\boldsymbol{F}^{\mathrm{T}1}\tilde{\boldsymbol{B}} \right]^{\mathrm{sym}} : {}^1\mathbb{L} : \left[\boldsymbol{F}^{\mathrm{T}} \left({}^\Gamma\tilde{\boldsymbol{N}} \otimes {}^0\boldsymbol{N} \right) \right]^{\mathrm{sym}} \mathrm{d}V \right\},
$$
(A.6)

where we have differentiated Equation 10 with respect to $[\![{}^0\boldsymbol{u}]\!]$.

REFERENCES

1. Pearson RA, Yee AF. Influence of particle size and particle size distribution on toughening mechanisms in rubber-modified epoxies. *Journal of Materials Science* 1991; **26**(14):3828–3844.
2. White SR, Sottos N, Geubelle P, Moore J, Kessler M, Sriram S, Brown E, Viswanathan S. Autonomic healing of polymer composites. *Nature* 2001; **409**(6822):794–797.
3. Lee HH, Chou KS, Shih ZW. Effect of nano-sized silver particles on the resistivity of polymeric conductive adhesives. *International Journal of Adhesion and Adhesives* 2005; **25**(5):437–441.
4. Sahni O, Zhou M, Shephard MS, Jansen KE. Scalable implicit finite element solver for massively parallel processing with demonstration to 160k cores. *High Performance Computing Networking, Storage and Analysis, Proceedings of the Conference on*, IEEE, Portland, OR, 2009; 1–12.
5. Evans TM, Davidson GG, Slaybaugh RN, CLARNO K. Three-dimensional full core power calculations for pressurized water reactors. *Journal of Physics: Conference Series, SciDAC*, Vol. 68, Chattanooga, TN, 2010; 367–379.
6. Baker AH, Gamblin T, Schulz M, Yang UM. Challenges of scaling algebraic multigrid across modern multicore architectures. *Parallel & Distributed Processing Symposium (IPDPS), 2011 IEEE International*, IEEE, Anchorage, AK, 2011; 275–286.
7. Adams MF. Parallel multigrid solvers for 3D unstructured finite element problems in large deformation elasticity and plasticity. *International Journal for Numerical Methods in Engineering* 2000; **48**(8):1241–1262.
8. Matouš K, Kulkarni MG, Geubelle PH. Multiscale cohesive failure modeling of heterogeneous adhesives. *Journal of the Mechanics and Physics of Solids* 2008; **56**(4):1511–1533.
9. Hirschberger C, Ricker S, Steinmann P, Sukumar N. Computational multiscale modelling of heterogeneous material layers. *Engineering Fracture Mechanics* 2009; **76**(6):793–812.
10. Kulkarni M, Matouš K, Geubelle P. Coupled multi-scale cohesive modeling of failure in heterogeneous adhesives. *International Journal for Numerical Methods in Engineering* 2010; **84**(8):916–946.
11. McBride A, Mergheim J, Javili A, Steinmann P, Bargmann S. Micro-to-macro transitions for heterogeneous material layers accounting for in-plane stretch. *Journal of the Mechanics and Physics of Solids* 2012; **60**(6):1221–1239.
12. Coenen E, Kouznetsova V, Geers M. Multi-scale continuous–discontinuous framework for computational-homogenization–localization. *Journal of the Mechanics and Physics of Solids* 2012; **60**(8):1486–1507.
13. Hill R. On constitutive macro-variables for heterogeneous solids at finite strain. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences* 1972; **326**(1565):131–147.
14. Nguyen VP, Stroeven M, Sluys LJ. Multiscale failure modeling of concrete: micromechanical modeling, discontinuous homogenization and parallel computations. *Computer Methods in Applied Mechanics and Engineering* 2012; **201**:139 –156.
15. Miehe C, Schröder J, Schotte J. Computational homogenization analysis in finite plasticity simulation of texture development in polycrystalline materials. *Computer Methods in Applied Mechanics and Engineering* 1999; **171**(3):387–418.
16. Özdemir I, Brekelmans WAM, Geers MGD. Computational homogenization for heat conduction in heterogeneous solids. *International Journal for Numerical Methods in Engineering* 2008; **73**(2):185–204.
17. Souza FV, Allen DH. Multiscale modeling of impact on heterogeneous viscoelastic solids containing evolving microcracks. *International Journal for Numerical Methods in Engineering* 2010; **82**(4):464–504.
18. Geers M, Kouznetsova V, Brekelmans W. Multi-scale computational homogenization: trends and challenges. *Journal of Computational and Applied Mathematics* 2010; **234**(7):2175–2182.
19. Feyel F, Chaboche JL. Fe^2 multiscale approach for modelling the elastoviscoplastic behaviour of long fibre SiC/Ti composite materials. *Computer Methods in Applied Mechanics and Engineering* 2000; **183**(3):309–330.
20. Belytschko T, Song JH. Coarse-graining of multiscale crack propagation. *International Journal for Numerical Methods in Engineering* 2010; **81**(5):537–563.
21. Okada Ji, Washio T, Hisada T. Study of efficient homogenization algorithms for nonlinear problems. *Computational Mechanics* 2010; **46**(2):247–258.
22. Talebi H, Silani M, Bordas SP, Kerfriden P, Rabczuk T. A computational library for multiscale modeling of material failure. *Computational Mechanics* 2013:1–25.
23. Yvonnet J, He QC. The reduced model multiscale method (R3M) for the non-linear homogenization of hyperelastic media at finite strains. *Journal of Computational Physics* 2007; **223**(1):341–368.
24. Monteiro E, Yvonnet J, He Q. Computational homogenization for nonlinear conduction in heterogeneous materials using model reduction. *Computational Materials Science* 2008; **42**(4):704–712.
25. Oskay C, Fish J. Eigendeformation-based reduced order homogenization for failure analysis of heterogeneous materials. *Computer Methods in Applied Mechanics and Engineering* 2007; **196**(7):1216–1243.
26. Center for Applied Scientific Computing. *Hypre User's Manual*. Lawrence Livermore National Laboratory: Livermore, CA, 2012. (Available from: http://computation.llnl.gov/casc/hypre/software.html).
27. Karypis G, Kumar V. Parallel multilevel series k-way partitioning scheme for irregular graphs. *Siam Review* 1999; **41**(2):278–300.
28. Hysom D, Pothen A. A scalable parallel algorithm for incomplete factor preconditioning. *SIAM Journal of Scientific Computing* 2001; **22**(6):2194–2215.
29. Bensoussan A, Lions JL, Papanicolaou G. Asymptotic analysis for periodic structures. *American Mathematical Society* 2011; **374**.

30. Fish J, Yu Q. Two-scale damage modeling of brittle composites. *Composites Science and Technology* 2001; **61**(15):2215–2222.
31. Roache PJ. *Verification and Validation in Computational Science and Engineering*. Hermosa Albuquerque: NM, 1998.
32. Segurado J, Llorca J. A numerical approximation to the elastic properties of sphere-reinforced composites. *Journal of the Mechanics and Physics of Solids* 2002; **50**(10):2107–2121.
33. Matouš K, Inglis H, Gu X, Rypl D, Jackson T, Geubelle PH. Multiscale modeling of solid propellants: from particle packing to failure. *Composites Science and Technology* 2007; **67**(7):1694–1708.
34. Lee H, Brandyberry M, Tudor A, Matouš K. Three-dimensional reconstruction of statistically optimal unit cells of polydisperse particulate composites from microtomography. *Physical Review E* 2009; **80**(6):061301.
35. Khisaeva Z, Ostoja-Starzewski M. On the size of RVE in finite elasticity of random composites. *Journal of elasticity* 2006; **85**(2):153–173.