A three dimensional adaptive multiscale method for crack growth in Silicon

A three dimensional adaptive multiscale method for crack growth in Silicon

P.R. Budarapu[a], J. Reinoso[a,b,*], B. Javvaji[c], M. Paggi[a,*], T. Rabczuk[d]

[a]*Multi-scale Analysis of Materials Research Unit, IMT School for Advanced Studies Lucca, Piazza San Francesco 19, 55100 Lucca, Italy*
[b]*Elasticity and Strength of Materials Group, School of Engineering, University of Seville, Camino de los Descubrimientos s/n, 41092, Seville, Spain*
[c]*Institut für Kontinuumsmechanik, Leibniz Universität, 30167 Hannover, Germany*
[d]*Institute of Structural Mechanics, Bauhaus University of Weimar, 99423 Weimar, Germany*

Abstract

A three dimensional concurrently coupled adaptive multiscale method is herein proposed to simulate crack growth patterns by combining several numerical techniques across the length scales. The material of the coarse scale is modeled based on the virtual atom cluster model. A three dimensional version of the phantom node method is employed to model the strong kinematic discontinuities in the bulk. The discontinuous continuum element is concurrently coupled with a molecular statics model placed around the crack tip. The coupling between the coarse scale and the fine scale is realized through the use of ghost atoms, whose positions are interpolated from the coarse scale solution and their displacements are enforced as boundary conditions to the fine scale model. In the proposed numerical scheme, the fine scale region is adaptively enlarged as the crack propagates and the region behind the crack tip is adaptively coarsened to optimize the computation costs. An energy criterion is used to detect the crack tip location. All the atomistic simulations are carried out using the LAMMPS software. A computational framework has been developed in MATLAB to trigger the LAMMPS through `system` command. The developed framework is assessed through several numerical examples, focused on the study of crack growth in Silicon. Therefore, the diamond cubic lattice structure of Silicon is used in the fine scale, where the atom-atom interactions are modeled based on the Tersoff potential function. These applications show the versatility and robustness of the proposed methodology.

Keywords: Multiscale analysis; Three dimensional fracture; Atomistic simulations; Silicon; Adaptivity.

1. Introduction

Investigating the material behaviour at the smallest scale provides valuable information regarding the physical behavior of materials, especially in detecting material failures [1]. However, there are other mechanical aspects of vital importance in heterogeneous materials, such as: the comprehensive understanding of the influence of multiphasic interfaces and hierarchical organizations across length scales on macroscale properties and the prospective presence of defects, to name a few. The exploitation of this information will provide an excellent potential for developing a materials-by-design approach for novel engineering materials and cyberphysical material systems [2]. In this context, although molecular dynamics (MD) simulations promise to reveal the fundamental mechanics of material failure at nano scales by modeling the atom-to-atom interactions; due to their small dimensions of the order of angstroms (Å), they are still prohibitively expensive to be employed in industrial applications [3, 4].

Given the present limitations, the advent of multi-scale modeling procedures opened a new scenario. Among the different multi-scale methods, a possible alternative to reduce the computational demand envisages the coupling between the continuum scale with the discrete scale using continuum-atomistic approaches. In such paradigms, defects are explicitly modeled at the sub-scales, whilst a self-consistent continuum model is defined over the entire domain under analysis. Adopting this computational scheme, several numerical models dealing with multiple spatial and temporal length scales have been proposed in the last two decades [5, 6, 7, 8, 9, 10, 11, 12, 13, 14]. Most of the coupling methods and simulations have been focussing on models of intact materials (without cracks). However, the information transfer through different length scales for problems involving material failure and finite temperatures remains a challenging task. In this regard, multiscale methods can be categorized into hierarchical [15, 16, 17, 18, 19], semi-concurrent [20, 21, 22, 23, 24, 25] and concurrent methods [6, 26, 8, 27, 12, 28, 29, 30, 14]. More specifically, numerous concurrent multiscale methods have been developed, which can be referred as 'Interface' coupling methods and 'Handshake' coupling methods. In particular, on the one hand, the coupling is performed along a defined interface in case of Interface

*Corresponding author. Tel: +39-0583-4326-561, Fax: +39-0583-4326-565
Email addresses: `jreinoso@us.es` (J. Reinoso), `marco.paggi@imtlucca.it` (M. Paggi)

coupling methods [26], whereas, on the other hand, a definite region of coupling exists for the 'handshake' coupling approaches [9]. From the numerical point of view, 'interface' coupling methods seem to be less effective for dynamic applications due to the fact that the complete avoidance of spurious wave reflections at the 'artificial' interface seems to be very problematic and heuristic. This is because of the abrupt change in the stiffness around the boundary of the atomistic domain, due to the rigid boundary conditions along the interface. Conversely, adaptive adjustment of the fine scale region as defects propagate is cumbersome in 'handshake' methods.

Some of the the existing concurrent multiscale methods have been subsequently extended to model fracture events [30, 31]. Talebi *et al*., [32] developed a concurrent coupling scheme coupling molecular dynamics to extended finite element method through the bridging domain method, to model three dimensional cracks and dislocations at the atomistic level. This technique particularly employs the material behavior in the continuum scale to estimate the corresponding properties based on the Cauchy-Born method. Recently, Budarapu *et al*., [33] proposed a solid shell-based adaptive atomistic-continuum numerical method by coupling a hybrid solid shell formulation in the continuum with atomistic domain to simulate complex crack growth patterns in thin-walled structures. The solid-shell formulation in the continuum domain envisages the combined use of the enhanced assumed strain (EAS) and the assumed natural strain (ANS) methods to efficiently model the material in thin structures. In addition to the previous considerations, it is worth mentioning that one of the main difficulties in multiscale methods for fracture is to upscale fracture-related information from the fine-scale to the coarse-scale, in particular for complex crack patterns. Most of the above mentioned approaches therefore were applied to examples with comparatively few macroscopic cracks. Recently, Greco *et al*., [34] have proposed a concurrent multiscale method to overcome the existing limitations on homogenization in the presence of strain localization in masonry structures. They adopted a multilevel domain decomposition approach equipped with an adaptive zooming-in criterion for detecting the zones affected by strain localizations. Many of the above mentioned multiscale methods do not adaptively adjust the fine scale domain as the defects propagate.

In this paper, we propose an effective multiscale method to study the propagation of cracks in a wide range of problems. Therefore, a computational frame work in three dimensions has been herein developed, which can be extended to many atom-to-atom potential functions and continuum based discretization. The continuum is modeled based on the virtual atom cluster (VAC) approach, which is developed considering the underlying atomistic lattice structure. One of the advantages of the VAC model is the fact that when VAC assumes the underlying lattice and consequently the continuum will be equivalent to the full scale atomistic model [35, 14]. During the implementation, the generality of the computational and algorithmic methods is thoroughly treated in order to be applicable for real physical conditions and computational sizes beyond just simple academic examples. This is achieved by coupling the VAC based phantom node method (PNM) code to the Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS) [36] molecular simulator. This coupling enables the use of vast capabilities of LAMMPS in an affordable and straightforward manner. The present three dimensional method allows to accurately simulate the complex fractures, which is not possible in two dimensional simulations. The developed methodology is applied to study crack growth in Silicon.

This article is an extension of the techniques proposed in [14]. However, several improvements are included in the present work to the original research. First, the developed method is applied to study crack growth within a three dimensional setting. Second, the proposed framework is developed in an in-house MATLAB code to trigger the LAMMPS software, which is being activated to solve for the fine scale solution. An efficient neighbor search method is proposed to locate atoms in the finite element region. Third, the adaptive refinement and coarse graining schemes are extended to three dimensions in order to adaptively adjust the fine scale region as the crack grows. Finally, a simple but effective method is proposed to locate the crack tip approaching the boundary that finally allows the atomistic region to be adaptively adjusted.

The organization of the article is as follows: The proposed three dimensional multiscale method is discussed in Section 2. The specific coupling conditions are discussed in Section 2.1. An overview of the mathematical formulation of the coarse scale model along with the details of incorporating the crack in the continuum through the phantom node method are presented in Section 3. The adaptivity scheme and the computer implementation algorithms are discussed in Sections 4 and 5, respectively. The present multiscale method is assessed through two numerical examples, presented in Section 6. As a first example, we studied the crack opening in the coarse domain, developed using the 8-node brick element, the solid-shell element and a VAC model based coarse regions. The second example is dedicated to study the crack growth in a double edge notched specimen. In the last example, we considered the growth of a straight edge crack. The key contributions are summarized in Section 7.

2. Three dimensional Adaptive Multiscale Method: fine scale model and coupling conditions

In this section, we present the fundamental aspects of the present three dimensional (3D) multiscale method for the adaptive simulation of quasi-static crack growth. In particular, the methodology is applied to simulate

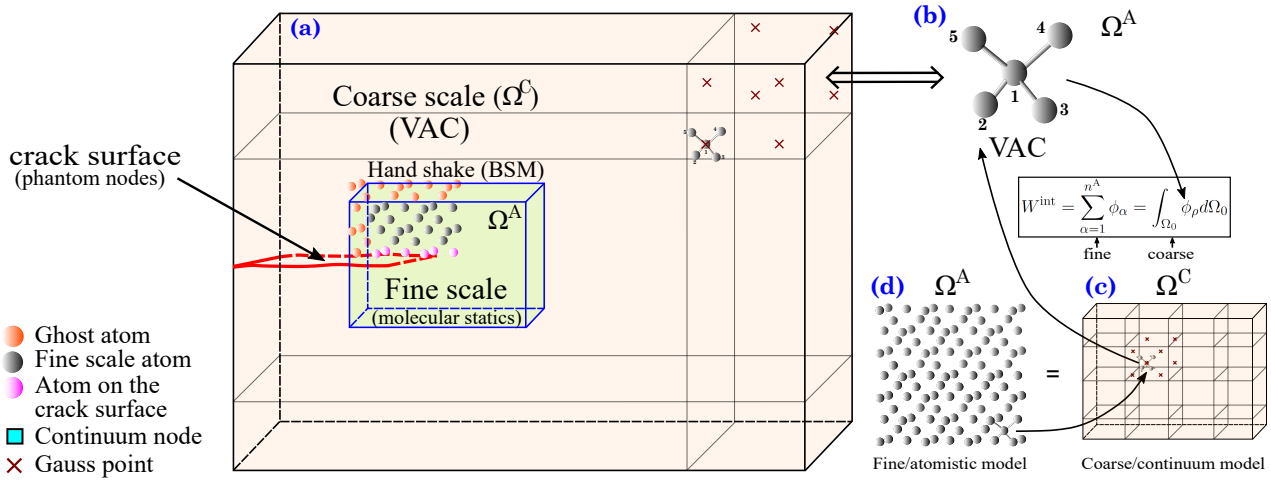

Figure 1: (a) Schematic of three dimensional multiscale method showing the coarse and fine scale domains along with the crack surface. (b) Virtual atom cluster of Silicon employed to model the material of the coarse scale region. (c) Virtual atom cluster placed at a Gauss point in the coarse scale region. (d) Cubic diamond lattice structure of Silicon used to model the fine scale region.

crack growth in Silicon systems, although the basic characteristics can be extended to different materials.

Consider a representative region of the multiscale model consisting of the coarse and fine scale regions along with the crack surface in three dimensions as shown in Fig. 1(a). The material of the coarse scale region is modeled based on the VAC model as shown in Fig. 1(b). In the diamond cubic lattice structure of Silicon shown in Fig. 1(d), each atom possess four nearest neighbours. Therefore, an atom with its four nearest neighbours (see Fig. 1(b)) is taken as the representative lattice structure of Silicon. Fig. 1(c) shows the modeling details of the coarse scale region by placing the virtual atom cluster of Silicon at a Gauss point. The energy equivalence of the VAC based coarse scale model in Fig. 1(c) and the fine scale model in Fig. 1(d) is shown in Fig. 1(b). The phantom node method [37, 38, 39], explained in Section 3.2, is used to model the crack surfaces in the coarse scale region.

An initial crack is assumed in the coarse scale, where the corresponding crack tip is captured in the fine scale region, see Fig. 1(a). The fine scale region is formed by the atomistic model of Silicon, as shown in Fig. 1(d). Energy minimization in the fine scale domain is carried out using the LAMMPS software at the lowest scale of analysis [36]. The atom-atom interactions of Silicon in the fine scale are modeled through the postulation of the Tersoff potential function [40]. In order to model the existence of the initial crack in the fine scale region, a set of the bonds between atoms on the crack surface are deleted and the corresponding neighbor list is accordingly updated. In the LAMMPS software, this is achieved by creating several regions and excluding the interactions between the regions containing the crack. By doing so, the neighbour list is generated based on a radius of influence, which is updated after every load step along the solution process. In line with the method developed in [41]. The 'crack tip' in an atomistic domain is identified as the intersection of atoms on the crack surface on either side of the crack. Atoms on the crack surface possess higher potential energies compared to other atoms in the lattice structure. This allows the identification of the atoms on the crack surface based on an energy criteria [41, 42].

Ghost atoms are located in the coarse region but within the cutoff radius of the atoms in the fine region, see Fig. 1(a). The corresponding positions of the ghost atoms are interpolated from the coarse scale solution and enforced as the boundary conditions for the fine scale solution. This computational process has been automated through the development of an interface in MATLAB which activates the LAMMPS in each load/time step. Consequently, the LAMMPS input file(s) is suitably updated with the positions of atoms in the latest deformed configuration with the minimal intervention along the solution procedure. Thus, the fine scale region is adaptively enlarged as the crack propagates and the model behind the crack tip is coarse grained to reduce the size of the fine scale model [39, 41, 42]. The above steps led to an adaptive continuum-atomistic multiscale method in the framework of enhanced bridging scale method (BSM) for crack growth in three dimensions, wherein the use of the phantom node method to model crack in continuum based on VAC model is envisaged.

Adopting the simplest kinematic assumptions, in the two scale model, the total displacement field $\mathbf{u}_\alpha$ of an atom α can decomposed into coarse and fine scale components:

$$\mathbf{u}_\alpha = \mathbf{u}_\alpha^{\mathrm{C}} + \mathbf{u}_\alpha^{\mathrm{A}} \tag{1}$$

where $\mathbf{u}_\alpha^{\mathrm{C}}$ is the coarse scale component and $\mathbf{u}_\alpha^{\mathrm{A}}$ is the fine scale component, whose projection onto the coarse scale is zero [10]. The coarse scale displacement $\mathbf{u}_\alpha^{\mathrm{C}}$ of an atom α can be represented within the FEM setting by

means of a set of basis functions defined over a set of n^{C} nodal points,

$$\mathbf{u}^{\mathrm{C}}_{\alpha} = \sum_{I=1}^{n^{\mathrm{C}}} N_I(\mathbf{X}_{\alpha})\mathbf{u}^{\mathrm{C}}_{I} \tag{2}$$

where $N_I(\mathbf{X}_{\alpha})$ identifies the shape function defined at node I for the kinematic interpolation, which is estimated at the α^{th} atom with the material coordinate $\mathbf{X}_{\alpha}$, and $\mathbf{u}^{\mathrm{C}}_{I}$ is the continuum displacement vector at node I.

2.1. *Coupling the coarse and fine scales*

The Tersoff potential function is employed in this study, to simulate the atom-to-atom interactions among the Silicon atoms in the fine scale region. The mathematical expression of the Tersoff potential function is given by [40]:

$$V(r_{\alpha\beta}) = f_c(r_{\alpha\beta})[f_R(r_{\alpha\beta}) + b_{\alpha\beta}f_A(r_{\alpha\beta})] \qquad f_R > f_c, \;\; f_A > b_{\alpha\beta} \tag{3}$$

where $r_{\alpha\beta}$ is the distance between the atoms α and β. The bond energy in the Tersoff framework is a combination of attractive (f_A) and repulsive (f_R) energy functions, which are expressed in the form of the exponential Morse like functions; f_c is a smooth spherical cut-off function around atom α based upon the distance to the first nearest neighbor cell. Fundamental details of the Tersoff potential function, the fine scale model and the estimation of the internal forces are outlined in AppendixA, AppendixB and AppendixC, respectively.

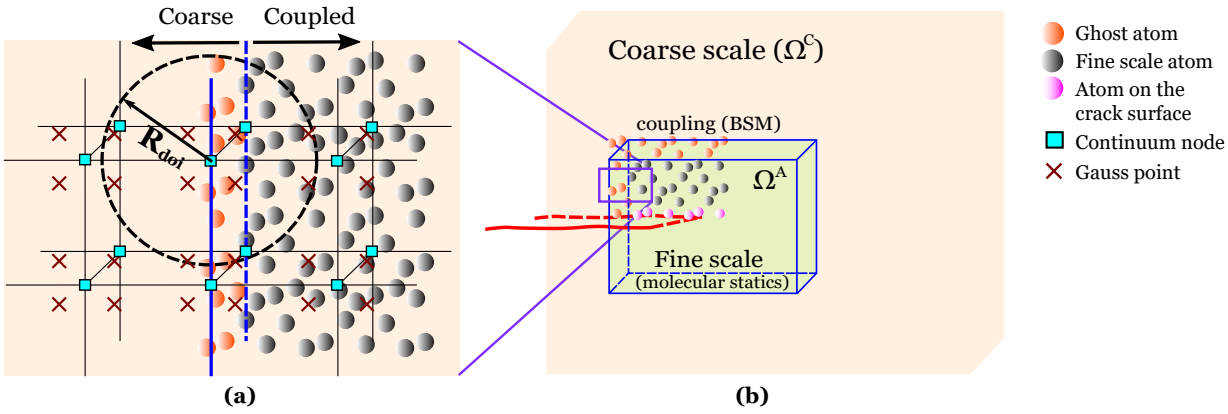

Figure 2: Schematic showing a close up of the region along the coupling boundary.

In the bridging scale method, the coupling between the coarse and the fine scales is realized by enforcing the displacement solutions corresponding to the delimiting boundaries of the fine scale on the ghost atoms, see Fig. 2. The positions of the ghost atoms are interpolated from the coarse scale solution. Thus, let β to be the index of the ghost atoms, the corresponding displacements are given by:

$$\mathbf{u}^{\mathrm{C}}_{\beta} = \sum_{I=1}^{n^{\mathrm{C}}} N_I(\mathbf{X}_{\beta})\mathbf{u}^{\mathrm{C}}_{I} \tag{4}$$

where $N_I(\mathbf{X}_{\beta})$ are the shape function defined at node I, which is estimated at the β^{th} atom with the material coordinates $\mathbf{X}_{\beta}$.

3. Coarse scale model

3.1. *Constitutive model at the coarse scale*

The interest of the current numerical method relies on Silicon systems in engineering applications. Concerning the constitutive formulation at the coarse scale, a periodic diamond crystal structure of Silicon is considered. Figures 3(a) and 3(b) depict an isometric and front views of the periodic diamond crystal structure of Silicon, respectively, whilst the space lattice of the Silicon is shown in Fig. 3(c).

The geometric parameters of the diamond crystal lattice are expressed using the lattice constant a, as shown in Fig. 3(a). Because of the symmetry of the crystal structure, a cluster of atoms can be taken as a representative model of the whole lattice structure [35]. As a result, all the calculations can be performed with reference to the representative cluster, which improves the computational efficiency. Since the locations of atoms in such cluster do not necessarily coincide with the exact locations of the atoms in the atomistic model, it is usually

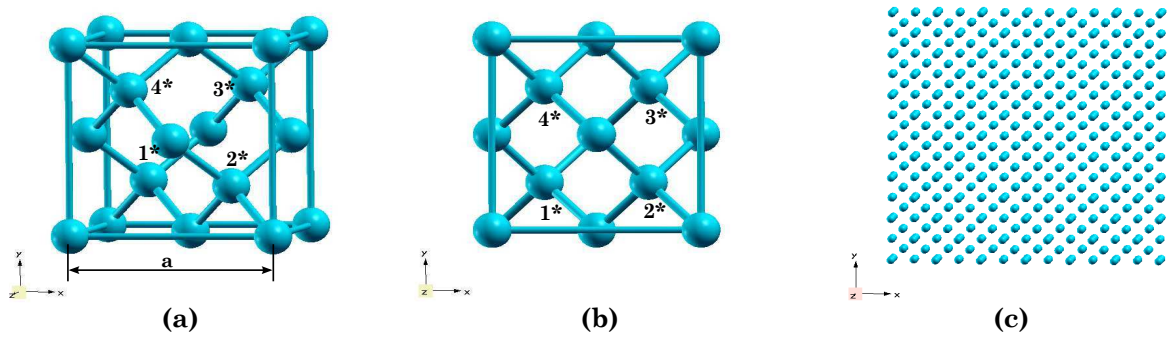

Figure 3: Silicon unit lattice in (a) an isometric view and (b) front view, highlighting the quarter atoms, 1* to 4*. (c) The front view of the space lattice.

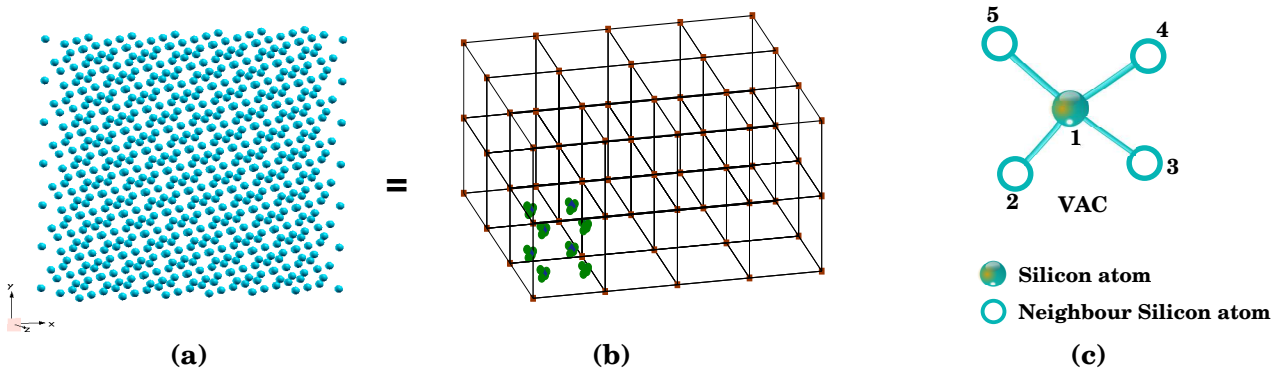

Figure 4: Silicon space lattice in (a) isometric view, (b) equivalent VAC based continuum model, showing the VAC placed at the Gauss points in the lowest left element. Due to the periodicity of the lattice structure (refer Fig. (3), any single atom can be considered as VAC. (c) The considered single atom VAC (atom 1* renamed as atom 1) along with its four nearest neighbors.

denominated as virtual atom cluster (VAC) in the related literature. The same inter-atomic potential as that corresponding to the full scale molecular statics (MS) model is used in the VAC model. Accordingly, the coarse scale displacement $\mathbf{u}_\alpha^{\mathrm{C}}$ of an atom α in the VAC can be estimated using Eq. (2).

The total potential energy of the fine scale system is given by the sum of all bond potentials ϕ_α, estimated as:

$$\phi_\alpha = \frac{1}{2}\sum_{\beta\neq\alpha}^{n^{\mathrm{A}}} V(r_{\alpha\beta}) \tag{5}$$

where $V(r_{\alpha\beta})$ is the potential function and $r_{\alpha\beta}$ is the distance between the atoms α and β.

In order to set up the constitutive model at the coarse scale, consider an equivalent coarse scale model based on the VAC as illustrated in Fig. 4. Fig. 4(a) shows the atomistic model, the equivalent VAC (see Fig. 4(c)) based continuum model is shown in Fig. 4(b). In order to achieve an equivalence between the fine scale and coarse scale models, their corresponding potential energy should be equal. This is achieved by defining a distributed energy density function, ϕ_ρ. Therefore, the discrete summation of the potential energy defined in the original molecular structure, can now be replaced with an integral based on the VAC model, i.e.:

$$W^{\mathrm{int}} = \sum_{\alpha=1}^{n^{\mathrm{A}}} \phi_\alpha = \int_{\Omega_0} \phi_\rho d\Omega_0 \approx \sum_{G} w_{\mathrm{G}} \phi_\rho^{\mathrm{G}}. \tag{6}$$

Due to the periodic nature of the lattice, ϕ_ρ can be defined as the potential energy of a VAC divided by the volume of the VAC. The volume of the VAC for a diamond lattice is $\mathrm{V}_0 = \frac{a^3}{8}$. When only nearest neighbour interactions are considered, each atom in the diamond lattice interacts with four neighbouring atoms, see Figs. 1 and 3. Thus, the energy attributed to a central atom involves the summation of the contributions from each bond with their corresponding neighbouring atoms, where the energy of the central atom is the energy of the VAC.

Using the numbering scheme shown in Fig. 1(b), the potential energy of the VAC is given by:

$$\phi_{\text{VAC}} = \frac{1}{2}\sum_{\beta=2}^{5} V(r_{1\beta}). \tag{7}$$

Therefore, the distributed energy density function ϕ_ρ can be defined as

$$\phi_\rho = \frac{\phi_{\text{VAC}}}{\text{V}_0} = \frac{1}{2}\sum_{\beta=2}^{5}\frac{V(r_{1\beta})}{a^3/8} = \frac{1}{2}\sum_{\beta=2}^{5}\phi_{1\beta}, \tag{8}$$

where,

$$\phi_{1\beta} = \frac{V(r_{1\beta})}{a^3/8}. \tag{9}$$

Hence, the energy density function ϕ_ρ becomes:

$$\phi_\rho = \frac{1}{2}(\phi_{12} + \phi_{13} + \phi_{14} + \phi_{15}) \tag{10}$$

where $\phi_{1\beta}$ are the bonding energy densities, w_{G} denote the quadrature weight and ϕ_ρ^{G} is the corresponding energy density function evaluated at a quadrature point.

The internal forces on node 'I' in the coarse scale region, can be estimated by taking the partial derivative of Eq. (6) with respect to the nodal displacements $\mathbf{u}_I^{\text{C}}$:

$$\mathbf{F}_I^{\text{int}} = -\frac{\partial W^{\text{int}}}{\partial \mathbf{u}_I^{\text{C}}} = \int_{\Omega_0} -\frac{\partial \phi_\rho}{\partial \mathbf{u}_I^{\text{C}}} d\Omega_0 = \int_{\Omega_0} -\frac{\partial \phi_\rho}{\partial \mathbf{u}}\frac{\partial \mathbf{u}}{\partial \mathbf{u}_I^{\text{C}}} d\Omega_0 \approx -\sum_{\text{G}} w_{\text{G}} \frac{\partial \phi_\rho^{\text{G}}}{\partial \mathbf{u}}\frac{\partial \mathbf{u}}{\partial \mathbf{u}_I^{\text{C}}}. \tag{11}$$

As shown in Fig. 1(b), there are 5 atoms in the VAC in total, and therefore, the energy density function can be expressed as: $\phi_\rho^{\text{G}} = \phi_\rho^{\text{G}}(\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3, \mathbf{r}_4, \mathbf{r}_5)$. Moreover, since the VAC model is used in the coarse scale region, $\mathbf{u}_\alpha = \mathbf{u}_\alpha^{\text{C}}$, $\alpha = 1-5$; also considering Eq. (2), the internal forces are computed as follows:

$$\mathbf{F}_I^{\text{int}} \approx -\sum_{\text{G}} w_{\text{G}} \frac{\partial \phi_\rho^{\text{G}}}{\partial \mathbf{u}}\frac{\partial \mathbf{u}}{\partial \mathbf{u}_I^{\text{C}}} \approx -\sum_{\text{G}} w_{\text{G}} \sum_{\alpha=1}^{5} \frac{\partial \phi_\rho^{\text{G}}}{\partial \mathbf{u}_\alpha^{\text{C}}}\frac{\partial \mathbf{u}_\alpha^{\text{C}}}{\partial \mathbf{u}_I^{\text{C}}} \approx -\sum_{\text{G}=1}^{n^{\text{G}}} w_{\text{G}} \sum_{\alpha=1}^{5} \frac{\partial \phi_\rho^{\text{G}}}{\partial \mathbf{u}_\alpha^{\text{C}}} N_I(\mathbf{X}_\alpha). \tag{12}$$

The term $\frac{\partial \phi_\rho}{\partial \mathbf{u}_\alpha^{\text{C}}}$ in Eq. (12) can be evaluated for each α as given below:
$\alpha = 1$

$$\frac{\partial \phi_\rho}{\partial u_{1i}^{\text{C}}} = \frac{\partial \phi_{12}}{\partial r_{12}}\frac{r_{12i}}{r_{12}} + \frac{\partial \phi_{13}}{\partial r_{13}}\frac{r_{13i}}{r_{13}} + \frac{\partial \phi_{14}}{\partial r_{14}}\frac{r_{14i}}{r_{14}} + \frac{\partial \phi_{15}}{\partial r_{15}}\frac{r_{15i}}{r_{15}} \tag{13}$$

$\alpha = 2$ to 5

$$\frac{\partial \phi_\rho}{\partial u_{\alpha i}^{\text{C}}} = -\frac{\partial \phi_{1\alpha}}{\partial r_{1\alpha}}\frac{r_{1\alpha i}}{r_{1\alpha}} \tag{14}$$

where i is the index of the coordinate axes and $r_{\alpha\beta i}$ is the component of $r_{\alpha\beta}$ in the i^{th} direction, which is given by: $r_{\alpha\beta i} = r_{\alpha i} - r_{\beta i}$. The detailed derivation of the term $\frac{\partial \phi_\rho}{\partial \mathbf{u}_\alpha^{\text{C}}}$ is explained in appendix of [14], for the arrangement of atoms as on the (111) plane of an face centered cubic lattice (fcc) structure and therefore it is omitted here for the sake of brevity. Finally, it is worth mentioning that a similar methodology can be extended to the single atom VAC of a diamond cubic lattice structure of Silicon considered in the present work. Therefore, equations (13)-(14) are substituted into Eq. (12) to calculate the internal nodal forces in the coarse scale. The minimization problem can be solved for the coarse scale solution by minimizing the potential energy for the given boundary conditions.

3.2. Phantom node method for three dimensional fracture

Uncracked elements at the coarse scale correspond to standard first-order fully integrated solid elements, whose main attributes are standard and therefore their variational formulation and the subsequent FE discretization is omitted for brevity. In the case of cracked elements, which are completely cut by a crack, a discontinuous kinematic description is accounted for through the use of phantom node method. In particular, this approach endows a discontinuous kinematic description by superimposing two separate displacement

fields which are continuous on each part of the cracked element [37, 38, 39, 43, 44]. One of the most appealing aspects of the PNM is the fact that only a small modification in the existing finite element codes is required, since, the discontinuous element is replaced by two overlapping elements with the additional phantom nodes. Therefore, standard shape functions of an intact element can be used for a cracked element as well. Moreover, the elements adjacent to the cracked elements do not require any further modification.

The fundamental aspects of the PNM are summarized in the sequel. Let to consider an arbitrary continuum body with a surface of discontinuity Γ_c. According to the PNM, the kinematics of a cracked element can be described by of superimposing two separate displacement fields, which are active only in a selected region of the domain. Characteristic representations of three dimensional finite elements containing straight and an angled edge crack are shown in Fig. 5(a) and (b), respectively.

Let Ω_0 to be the real domain, which is to be cut by the crack, whereas Ω_{ph} identifies the phantom domain, such that $\Omega_{\text{ph}} = \Omega_{\text{ph}}^{+} \bigcup \Omega_{\text{ph}}^{-}$. The part of the cracked elements which belong to the real domain Ω_0 are extended over the phantom domain Ω_{ph} identifies the phantom domain, such that $\Omega_{\text{ph}} = \Omega_{\text{p}}$, so that the interpolation basis is complete at the element level. Therefore, the displacement in the real domain Ω_0 can be interpolated using the degrees of freedom for the nodes in the phantom domain Ω_{ph}. The nodes in the phantom domain are called the phantom nodes, denoted by empty circles in Fig. 5(a) and (b). In other words, the approximation of the cracked element is the superposition of the two approximations ($\Omega_0^{+} \bigcup \Omega_{\text{ph}}^{-}$ and $\Omega B_0^{-} \bigcup \Omega B_{\text{ph}}^{+}$) over the cracked elements, as shown in Fig. 5(a) and (b). Through the definition of f as the signed distance measured from the crack, W_0^{+}, W_{ph}^{-}, W_0^{-} and W_{ph}^{+} denote the nodes belonging to Ω_0^{+}, Ω_{ph}^{-}, Ω_0^{-} and Ω_{ph}^{+}, respectively. With these definitions at hand, the kinematic field can be expressed as:

$$\mathbf{u}(\mathbf{X},t) = \sum_{I \in \{W_0^{+}, W_{\text{ph}}^{-}\}} \mathbf{u}_I(t) N_I(\mathbf{X}) H(f(\mathbf{X})) + \sum_{J \in \{W_0^{-}, W_{\text{ph}}^{+}\}} \mathbf{u}_J(t) N_J(\mathbf{X}) H(-f(\mathbf{X})) \tag{15}$$

where H stands for the the Heaviside function.

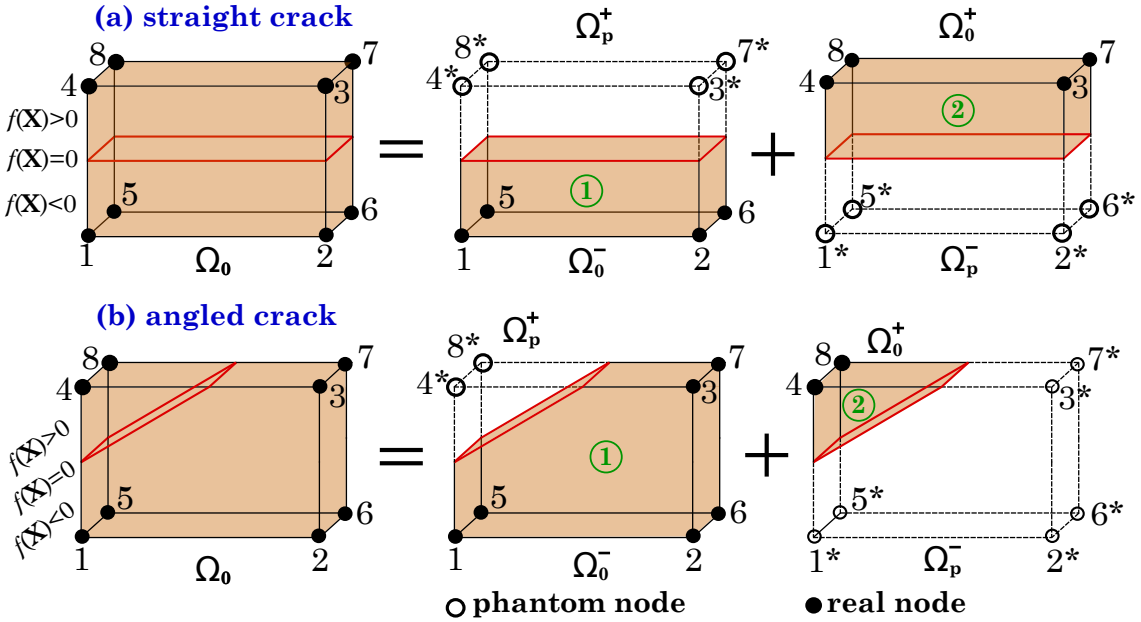

Figure 5: Schematic showing the Phantom node representation of a cracked element with (a) straight and (b) angled cracks.

Relying on the previous description, it is worth noting that cracked elements have both real nodes and phantom nodes as shown in Fig. 5(a) and (b). Consequently, the discontinuity in the displacement field is realized by simply integrating only over the volume from the side of the real nodes up to the crack, i.e. the shaded areas in element 1 and element 2, Ω_0^{-} and Ω_0^{+}, respectively.

The initial phantom nodes are created on the completely cracked elements. The crack tip location is captured at every load step, from the fine scale model described above. Based on the location of the crack tip, the elements are checked for complete fracture. Thus, if an element is completely cracked, the crack is propagated in the coarse scale domain. To do so, the new phantom nodes are created on the newly cracked element, and their positions are initialized by interpolation from the coarse scale solution. The nodal connectivity table is updated with the phantom nodes, for the next load step.

4. Adaptivity

In order to provide a high level of versatility of the current multi-scale procedure, we adopt the adaptivity scheme proposed by the authors in [14]. In line with this procedure, the crack tip is contained within the fine scale region Ω^A, while the fine scale regions are coarse-grained when the crack is predicted to propagate. Therefore, the adaptivity scheme consists of the following steps:

1. Identification of the coarse region Ω^C to be refined. Refine the model by expanding the atomistic region Ω^A, i.e by converting the estimated coarse scale region into a fine scale region.
2. Determination of the fine region Ω^A to be coarse grained. Coarsen the model by shrinking the atomistic region Ω^A, i.e by converting the estimated fine scale region into a coarse scale region.

The details of refinement and coarsening algorithms are discussed in [14] in detail, and therefore are not included here.

It is also mentionable that the initial size of the fine scale domain is chosen in such a way that all the mechanics of crack growth particularly around the crack tip can be captured. Therefore, it is important that the initial domain size surrounds sufficient region ahead and behind the crack tip. Furthermore, a large initial domain can lead to higher computational costs and the crack tip may jump out of small fine scale domains. Some of parameters that can influence size of the fine scale region include: type of problem (static/dynamic), geometry and boundary conditions, type and rate of loading and the type fracture (brittle/ductile). Hence, we followed the guide lines mentioned in [14] and tested several sizes before finally arriving at the domain sizes used in the present work.

The adaptive refinement process explained in Fig. 6, consists of the following steps:

1. Identify the elements to be refined.
2. Create and initialize the atoms in the elements to be refined.
3. Create and initialize the phantom nodes on the newly cracked elements.
4. Update the fine and coarse scale regions.

Figure 6(a) shows the elements flagged for refinement and the enlarged atomistic region after the refinement operation is shown in Fig. 6(b).

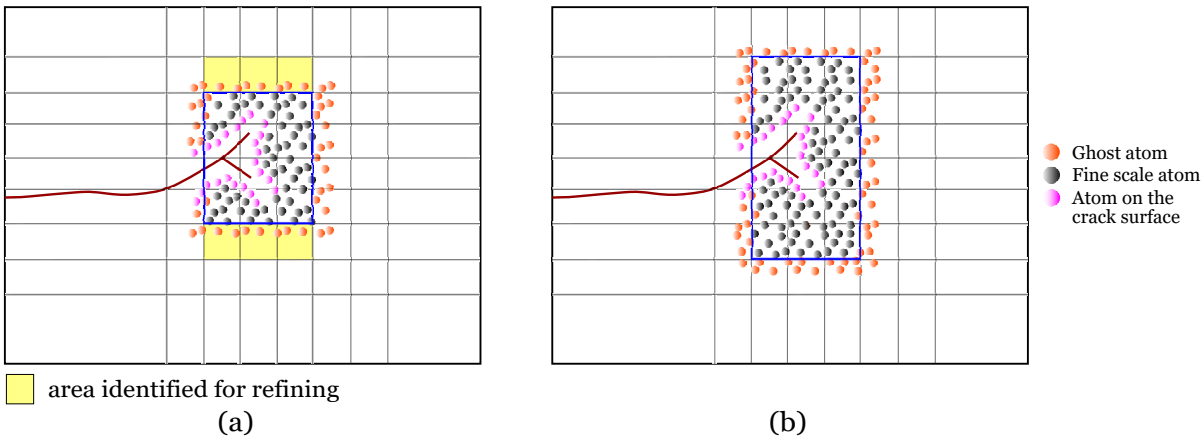

Figure 6: Schematic showing the atomistic domain (a) before and (b) after a refinement operation.

In the current study, the crack is to be propagated in both the coarse scale and the fine scale regions. Crack propagation in the fine scale region depends on the number of broken bonds, which are identified based on the bond distance. The newly created atoms in the elements identified for refinement operation, are initialized using Eq. (2). Now the atomistic domain is enlarged to handle the elongated crack. To improve the computational efficiency, the atomistic elements behind the crack tip can be coarse grained.

The adaptive coarsening operation involves the following major steps:

1. Identify the elements to be coarse grained.
2. Delete the atoms in the elements to be coarse grained.
3. Update the fine and coarse scale regions.

The process of the adaptive coarse graining operation is explained in Fig. 7. Figure 7(a) shows the elements containing low energy inactive atoms being flagged and the elements after coarse graining are shown in Fig. 7(b). The crack path is a continuous surface when the enriched methods, such as phantom node method is used on the coarse scale. Hence, the crack in the coarse grained model is approximated by joining the crack paths in each of the discretized region containing the crack. Detailed implementation of the above methodology for various methods in two and three dimensions is explained in [41].

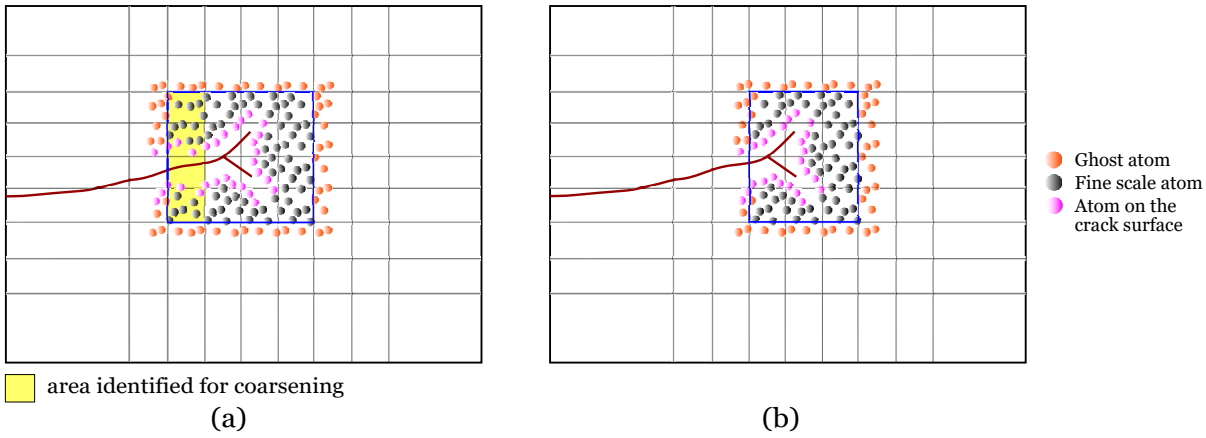

Figure 7: Schematic showing the atomistic domain (a) before and (b) after a coarse graining operation.

5. Solution algorithm of the multiscale model

In order to facilitate the automatization of the proposed multi-scale method, enabling versatility and robustness, the developed methodology has been implemented into an in-house code running within MATLAB environment. This allows the use of a large library of functionalities and libraries. In particular, the combination of the FE code in MATLAB with the MD tool LAMMPS is carried out through the '`system`' command, according to the following sentence:

$$\text{system('lmp\_mpi -in .../.../input\_file\_name -log log.ini');} \tag{16}$$

where 'lmp_mpi' indicate the LAMMPS executable, which is generated by the compilation of the parallel version of the LAMMPS code. Moreover, the command '.../.../input_file_name' is used to identify the exact location of the input file, whereas log file: *log.ini*, review the command history used by LAMMPS.

The main stages of the solutions scheme of the multiscale model are explained in Algorithm 1, and described in the sequel. First, the geometry of the coarse scale is discretized to create the 3D mesh. In the initial configuration, the elements containing the crack namely: split and tip elements, are identified based on the geometry of the existing crack(s), using the level set functions. Phantom nodes are created on the completely cracked elements. Extra Gauss points are created around the crack region. Later on the VAC model is placed at each Gauss point to estimate the potential energy and the nodal forces in the coarse domain.

Second, a LAMMPS input file is developed e.g.: '*test_pre.in*', in order to create an atomistic region around and including the crack tip. Some of the key pre-processing steps in LAMMPS are summarized in Algorithm 2. According to Algorithm 2, the diamond cubic lattice structure of the Silicon with a lattice constant of 5.431 is created using the `lattice` command. The fine scale domain dimensions are specified through the `region` command, where the 'min' and the 'max' indicate the minimum and maximum values of the domain in a given direction. As mentioned in the `create_box` and `create_atoms` commands, 5 different Si atom types are created in total. Accordingly, five Si atom types are selected in the 'tersoff' potential function as mentioned in the `pair_style` and `pair_coeff` commands. This will facilitate to separate the atoms into five different regions/groups based on their type. The atoms are grouped into different types using the `group` command, so that the ghost atoms in the entire simulation can be called by their type. Crack in the fine scale region is created by excluding the neighbour interactions between the groups of atoms lying on either side of the crack surface through the `neigh_modify` command. The CSP and per atom potential energies are estimated by employing the `compute` operations. Subsequently, '`system`' command is triggered to initiate the LAMMPS pre-processing steps. Note that the pre-processing steps are required to be executed only once, i.e., in the beginning of the simulation, e.g.: through the file 'test_pre.in'.

In the third step, a `for` loop is initiated to apply the external boundary conditions on the coarse scale, see step 6 in Algorithm 1. The total displacement load is divided into several specified steps. In this stage, the ghost atom positions are interpolated from the coarse scale solution using Eq. (4). The updated ghost atom positions after the interpolation are stored in an auxiliary atom data file, for example: atom_data$_{i(ls-1)}$, where '*ls*' corresponds to the current load step and the index '*i*' indicate an intermediate step between the previous and current load steps. In this study, the deformed configurations during the intermediate steps are stored in a separate folder. The updated atom positions are read into a separate input file, e.g.: 'test_min.in' through the `read_data` command, see Algorithm 2. The LAMMPS is now triggered again to minimize the potential energy of atomistic domain by fixing the updated ghost atom positions. In this study, this is achieved by applying the interpolated displacements of ghost atoms in several steps. The energy minimization in LAMMPS is carried

1. Create the intact and the initially cracked elements at the coarse scale in the initial configuration.
2. Create the phantom nodes on the initially cracked elements.
3. Create the Gauss points in each element. Note that extra Gauss points are created in the completely cracked elements.
4. Place the VAC at each Gauss point and estimate the potential energy and nodal forces of the coarse domain.
5. (a) Develop a LAMMPS input file, e.g.: *test_pre.in*, see Algorithm 2..
   (b) Trigger LAMMPS through `system` command to execute the file *test_pre.in* and dump the details of the atomistic model(s) in the initial configuration to a separate file
6. **for** *i (load_step) = 1 : number of steps* **do**
   7. Solve for the solution of multiscale model using the minimum energy criteria.
   (a) Apply the boundary conditions in continuum and solve for the coarse scale solution $\mathbf{u}_I^C$.
   (b) Interpolate the ghost atom positions based on Eq. (4)
   (c) Write the updated atom positions to another file, e.g.: atom_data$_{i(ls)}$
   (d) Trigger LAMMPS to minimize the potential energy of atomistic domain through the input file *test_min.in*,
   (e) Store the latest atom positions along with their energy and centro symmetry parameter (CSP)
   (f) Estimate the location of the crack tip based on the energy/CSP criteria.
   (g) Evaluate:
      if (crack tip is close to the boundary) **then**
         Activate the adaptivity scheme (refinement & coarsening)
      else
         Check and update the crack tip location in the coarse scale
         (Phantom nodes & connectivity)
      end if
end

Algorithm 1: Solution algorithm of the multiscale model.

```
# Key pre-processing steps, part of the input file 'test_pre.in'
# To create the diamond cubic lattice structure of Silicon
lattice        diamond   5.431

# Creating the fine scale region
region box block ${xmin} ${xmax} ${ymin} ${ymax} ${zmin} ${zmax} units box
create_box     5 box
create_atoms   5 region box

# Potential function information
pair_style  tersoff
pair_coeff  * *  SiC.tersoff  Si  Si  Si  Si  Si

# To create the crack
neigh_modify  exclude  group  lower  upper

# To compute the centro symmetry parameter and the per atom potential energy
compute  csp  all  centro/atom  4
compute  pe   all  pe/atom

# Below key steps are part of the separate input file for energy minimization, i.e., 'test_min.in'
# To read the interpolated ghost atom positions in the current load step
variable  ls  index  1
variable  rs  equal  "v_ls-1"
read_data  .../atom_data_${rs}

# Energy minimization
minimize  1.0e-9  1.0e-10  10000  100000
```

Algorithm 2: Key steps of energy minimization using the LAMMPS software. The methodology is applied to simulate the crack growth in Silicon using the Tersoff potential function.

out through the `minimize` command. Therefore, the latest atom positions at the end of energy minimization along with their energy and centro symmetry parameter (CSP) are saved into a different folder with different output file name, e.g.: 'dump_Si$_{ls}$.

The updated crack tip location at any particular load step can be identified using either the energy or the CSP criteria. Without loss of generality, we use the energy criteria to identify the atoms on the crack surface and hence, the location of the crack tip. Based on this criterion, two possible scenarios can be encountered depending on the location of the crack tip: (i) if the crack tip location is close to the boundary of the atomistic domain, the adaptivity scheme is activated, and (ii) the crack tip location can be directly updated in the coarse scale since it is far enough from the delimiting region of the fine scale. Finally, it is worth mentioning that updating the crack tip location in the coarse scale may require the introduction of new phantom nodes and the corresponding modification of the connectivity of the FE mesh [14]. The above steps within the `for` loop are repeated until either the atomistic regions reach the boundaries of the coarse scale or the end of the simulation.

6. Numerical examples

In this section,three numerical examples are presented to validate the proposed methodology and show its versatility. In the first example, crack opening in Mode I with an initial edge crack, is simulated based on the VAC coarse scale model. The nodal positions from the VAC based coarse model are compared to the nodal positions estimated using a 8-node brick element and a 8-node solid shell element [33]. The second example is dedicated to study the crack growth in a double edge notched specimen, using both molecular dynamics and the multiscale method. In the final example, Mode I crack growth of an edge notched specimen is studied based on the multiscale method.

6.1. Example 1: crack opening in the coarse region

In this example, modeling crack in the continuum region (Ω_C) based on the phantom node method using the 8-node brick element, the solid-shell element and the VAC model are compared. Consider a three dimensional continuum domain of dimensions 234 Å$\times$ 178 Å$\times$ 23 Å with a through-the-thickness initial edge crack of length 38 Å located at 89 Å from the bottom edge along the y direction, see Fig. 8(a). The top and bottom edges of the panel are subjected to tensile load, such that the crack opens in mode I. To achieve the objective, a displacement load of 8.345 Å has been prescribed on the top and bottom rows of nodes in 16 steps. The left and right edge nodes are allowed to move only along the y direction.

In the present work, the coarse region in all the examples is developed using the VAC model. Figure 8 shows the modeling details of the crack in the continuum based on the VAC model. The panel is discretized considering one element along the thickness direction. All the plots in Fig. 8 are shown in the in-plane direction. Figure 8(a) shows the distribution of the Gauss points in the entire domain. Note the extra Gauss points created in the completely cracked (split) elements on either side of the crack surface. More Gauss points around the crack zone ensures the accurate estimation of the nodal displacements in the split elements. A closeup of the domain around the crack in Fig. 8(a) is shown in Fig. 8(b). Figure 8(c) shows the distribution of atoms after placing the VAC at the corresponding Gauss points in Fig. 8(b). A closeup around the crack origin is shown in the inset picture in Fig. 8(c). Atoms in the VAC are separated by the lattice constant of 5.431 Å, as in the original diamond cubic lattice structure of Silicon.

The internal energy of the coarse region developed based on the VAC model is estimated through the summation of the energies of the VAC in each element according to Eq. (6). In every load step, the displacement load per step (total load/number of steps) is prescribed on the top and bottom row of nodes. The updated positions of the VAC atoms are estimated through interpolation using the latest nodal positions. The updated nodal and atoms positions are used to estimate the potential energy of the coarse region. The nodal positions in the current configuration are estimated based on the minimum potential energy criteria. In this study, the minimization is carried out using the `fmincon` function, using the `sqp` algorithm, in MATLAB software. Note that the linearization of the internal forces is not required and hence the stiffness matrix is not required to be evaluated. Nodal solution in the current load step is set as the initial solution for the next load step. The process is repeated until the end of the simulation. Solution at the end of the last load step indicate the deformed configuration at the end of the simulation. Steps of potential energy minimization in MATLAB are explained in Algorithm 3.

Step 3(c) is the key step in Algorithm 3. The `optimoptions` function in step 3c(i) allows to select the type of optimization, along with the algorithm and the optimality tolerance. The selected options are further used in the step 3c(ii) to activate the `fmincon` function. The potential energy of the VAC atoms is estimated using the `potene_c` function and the boundary conditions are specified in `congrad_c`. Hence, the details of the Tersoff potential function used to simulation the atom-to-atom interactions in the VAC are embedded in the function `potene_c`. Therefore, the nodal solution in the current load step `rn` is arrived based on the minimum potential energy criteria, using `r0n` as the initial guess. The gradient of the energy in the current configuration, indicating the internal nodal forces is stored in the variable `grad_c` in step 3c(ii).

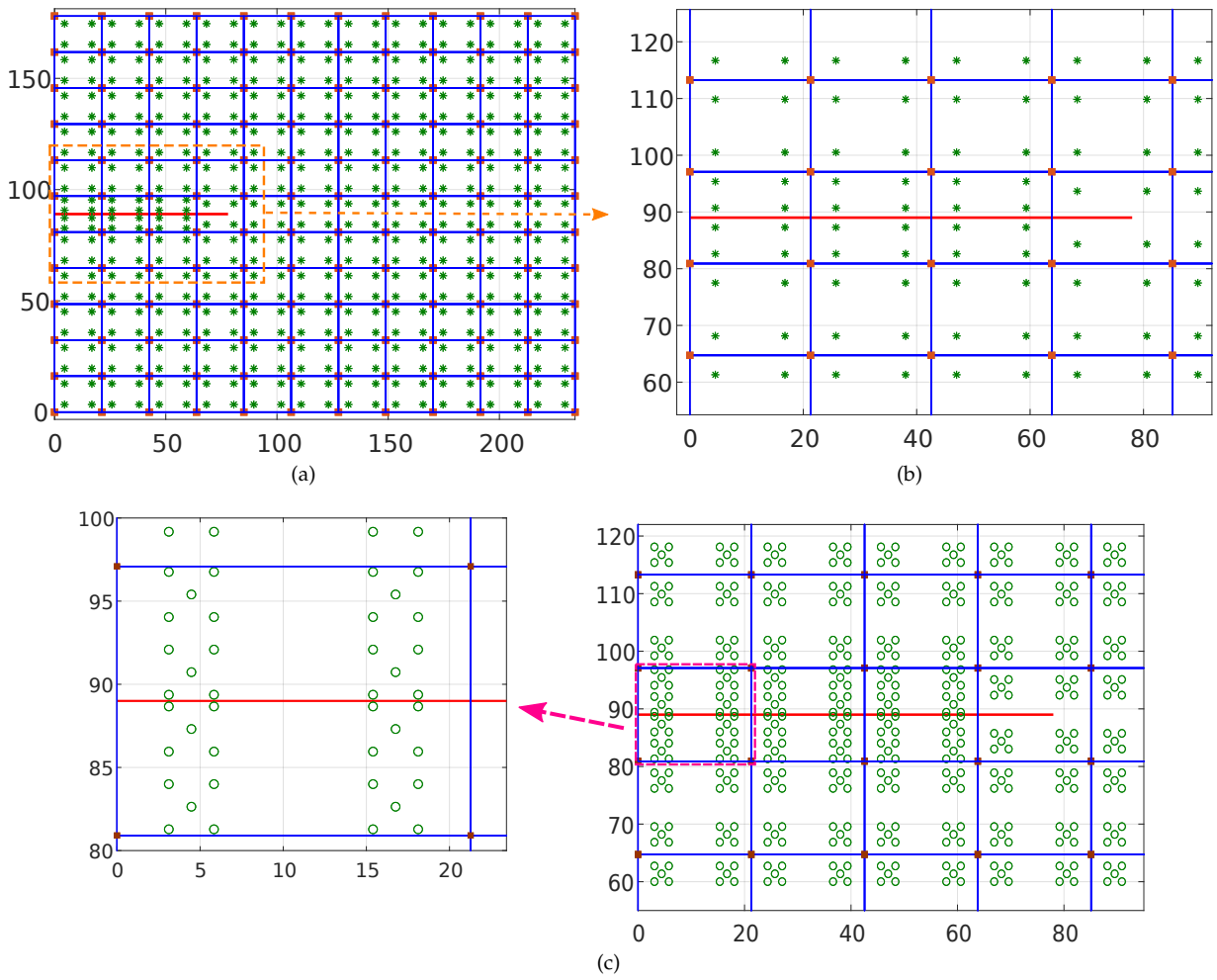

Figure 8: Modeling crack in the continuum using the VAC model. (a) Distribution of Gauss points in the entire domain. Note the extra Gauss points created in the split elements on either side of the crack surface. (b) A closeup of (a) around the crack surface. (c) Distribution of atoms in (b) after placing the VAC at the corresponding Gauss points, where the inset picture shows a closeup at the crack origin.

The initial and deformed configurations of the coarse region estimated using the 8-node solid brick element, the solid-shell element and the VAC models are shown in Fig. 9. The element is broken along the crack surface while modeling the crack using the 8-node brick element, by creating the extra nodes on the crack surface, see Figs. 9(a-c). On the other hand, the Phantom node method has been employed to simulate the crack in the coarse region modeled using the solid-shell and the VAC models, see Figs. 9(d-i).

The initial configurations of the coarse region modeled using 8-node solid brick element, the solid-shell element and the VAC models are shown in Figs. 9(a),(d) and (g), respectively. Deformed configurations in the middle (after 8 load steps) and the end of simulation (after 16 load steps) are plotted in the second and third columns of Fig. 9, respectively. The movement of the broken 8-node brick elements with load are shown in Figs. 9(b) and (c). The positions of the VAC atoms are shown in the deformed configurations Figs. 9(h) and (i). Figure 10(a) shows a comparison of the nodal positions estimated considering the 8-node brick, the solid-shell and the VAC model based coarse regions, at the end of simulation. Based on Fig. 10(a) the nodal positions are in excellent agreement among the three models. The load-displacement diagram as the crack opens, considering the VAC model based coarse domain is provided in Fig. 10(b). According to Fig. 10(b), a continuous increase of load with displacement is observed, indicating the crack opening.

6.2. Example 2: Double edge crack growth simulation

The second example is dedicated to discuss the simulations of double edge crack growth. Consider a three dimensional continuum domain of dimensions 256 Å$\times$ 256 Å$\times$ 20.5 Å with two through-the-thickness initial edge cracks of length 40 Å located at 66 Å and 190.1 Å from the bottom edge, respectively, along the y direction, see Fig. 11. Both molecular dynamics and multiscale simulations are carried out to compare the results. The top and bottom edges of the panel are subjected to tensile load, such that the cracks open in mode I. In this example, two different cases are studied. In the first case (see Fig. 11(a)), roller constraints are specified

1. Estimate the intact and cracked elements in the initial configuration.
2. Create the phantom nodes and extra Gauss points in the completely cracked elements.
3. Place the VAC model at each Gauss point.
4. Calculate the load per step: total load/number of steps.
5. **for** *i (load_step) = 1 : number of steps* **do**
   (a) Apply the boundary conditions on the boundary nodes and estimate the updated positions of the VAC atoms through interpolation.
   (b) Set the updated nodal and atom positions as the initial guess (`r0n`) for the current load step.
   (c) Activate the `fmincon` function with options as explained below:
      (i) `options = optimoptions('fmincon','Algorithm','sqp', ... 'OptimalityTolerance', 1e-3);`
      (ii) `[rn, fval_c, exitflag_c, output_c, lambda_c, grad_c] = ... fmincon(@potene_c, r0n, [], [], [], [], [], [], @congrad_c, options);`
   (d) Extract the current configuration from the variable `rn` and set as the initial configuration for the next load step.

   end
6. Solution at the end of the last load step indicate the deformed configuration at the end of the simulation.

Algorithm 3: Solution steps of energy minimization in the coarse scale.

along the left and right edges such that the corresponding atoms/nodes can move only along the y-direction. Whereas, in the second case no roller constraints are specified (see Fig. 11(b)), allowing the free deformation of atoms. Loading rates of 0.04 Å and 0.4 Å per step are considered in the atomistic and multiscale simulations, respectively. The simulations are continued until the cracks grow. The geometry of the cracks, the loading and boundary conditions are selected such that the cracks grow in opposite directions.

Figure 12 shows the crack growth simulations of double edge notched specimen based on molecular dynamics, considering the first case, described in Fig. 11(a). The initial configuration in Fig. 12(a) consists of a total of 71444 atoms, where the groups top, bottom, left and right contains 1136 atoms each. Cracks in the initial configuration are created by deleting the interactions between the atoms on either sides of the crack surfaces. The neighbour are calculated based on a force cutoff distance of 3.9 Å. Furthermore, the neighbours are updated after every load step. Displacement load is specified on the top and bottom rows of atoms such that the cracks open in mode I. Roller constraints are specified on the left and right edge atoms using `fix` command in LAMMPS. Continuous increase in load leads to crack opening by bond stretching until 800 steps as shown in Figs. 12(b) and (c). The system potential energy continues to increase as a result of bond stretching. Variation of the potential energy with displacement along the y-direction for cases 1 and 2 is plotted in Fig. 13(a). A comparison of the load-displacement diagram for both cases is shown in Fig. 13(b). The forces and displacements in Fig. 13(b) are estimated along the y-direction until the breakage of first bonds. Based on Fig. 13, the first bonds are observed to be broken around a displacement of 30 Å and 33 Å in cases 1 and 2, respectively. This corresponds to first breakage of bonds after 820 steps in case 1, as observed in Fig. 12(d). Further increase in load leads to further breakage of bonds and hence further growth of cracks. The cracks continue to grow as depicted in the deformed configuration plots after 1000 and 1400 steps in Figs. 12(e) and (f), respectively.

Crack growth simulations of double edge notched specimen based on molecular dynamics, considering the second case, described in Fig. 11(b) are demonstrated in Fig. 14. The initial configuration in Fig. 14(a) also consists of a total of 71444 atoms, where the groups top and bottom contains 1136 atoms each. The neighbour are calculated based on a force cutoff distance of 3.9 Å, where the neighbours are updated after every load step. Displacement load is specified on the top and bottom rows of atoms such that the cracks open in mode I. Continuous increase in load leads to crack opening by bond stretching until 800 steps as shown in Figs. 12(b) and (c). Lack of roller constraints on the left and right edge atoms results in poor stiffness compared to case 1 in Fig. 12. The system potential energy continues to increase as a result of bond stretching. However, breakage of the first bonds are delayed as compared to the case with roller constraints. Based on the variation of the potential energy with displacement plotted in Fig. 13(a), unconstrained case required more energy to break the first bonds. This is further confirmed by the load-displacement diagram shown in Fig. 13(b), where higher loads are noticed for the case without roller constraints, at the time of failure of the first bonds. The first breakage of bonds after 834 load steps in case 2, is shown in Fig. 14(d). Further increase in load leads to further breakage of bonds and hence further growth of cracks. The cracks continue to grow as depicted in the deformed configuration plots after 1000 and 1400 steps in Figs. 14(e) and (f), respectively. However, comparing the crack length at a given instant, rate of crack growth is observed to be small in case 2. Therefore, the crack

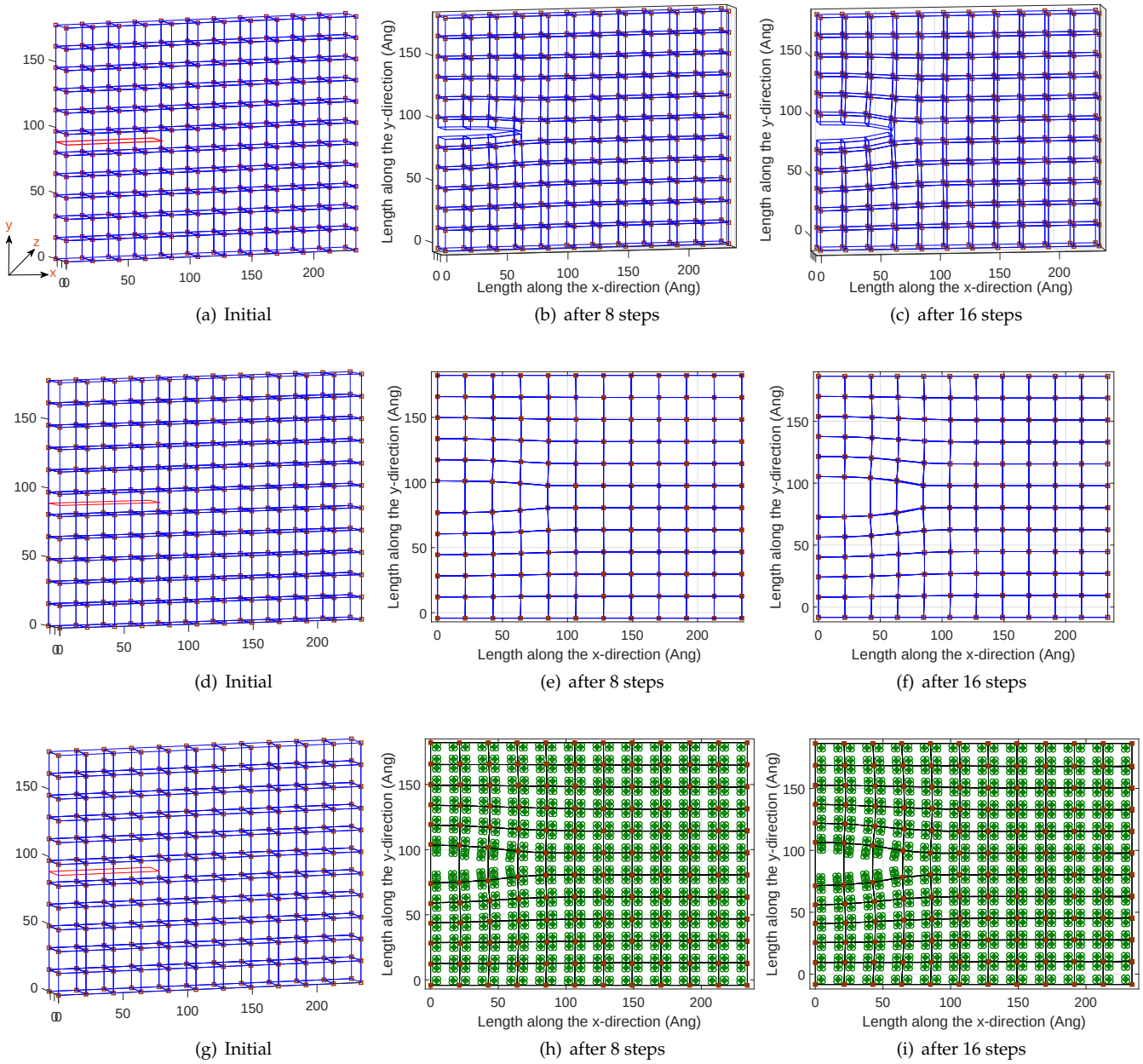

Figure 9: Comparison of the deformed configurations of the coarse region estimated using the 8-node solid brick element, the solid-shell element and the VAC model, plotted in the first (a-c), second (d-f) and third rows (g-i), respectively. The initial configurations are shown in the first column. Deformed configurations in the middle (after 8 load steps) and the end of simulation (after 16 load steps) are shown in the second and third columns, respectively. The element is broken along the crack surface in case of the 8-node brick model, whereas, the crack surface is modeled using the Phantom node method in the solid-shell and VAC based continuum regions. The VAC atoms are shown in the deformed configurations (h) and (i).

lengths after 1000 and 1400 steps in Figs. 14(e) and (f) are small, compared to the crack lengths in Figs. 12(e) and (f).

Crack growth simulations of the double edge notched specimen using the multiscale model are demonstrated in Fig. 15. In order capture the two crack tips, two atomistic regions are considered around the two crack tips. Displacement boundary conditions are specified on the nodes belonging to the top and bottom sides of the plate in the continuum region. On the other hand, roller constraints are prescribed on the nodes lying on the left and right edges of the domain. The continuum region is discretized using $34\times34\times2$ nodes along the x, y and z directions, respectively. Phantom nodes are created on the completely cracked elements. In the current initial model there are 10 completely cracked elements and 2 tip elements with 48 phantom nodes in total. Two fine scale regions measuring each one 79.3 Å$\times$79.6 Å$\times$ 20.5 Åwith 14592 active atoms and 2644 ghost atoms have been created as shown in Fig. 15(a). A close up view of the regions around the crack tips in Fig. 15(a) are shown in Fig. 15(b) and (c), respectively. A uniform loading rate of 0.4 Å per step is applied on the nodes on the top and bottom boundaries. Both the top and bottom edge nodes are restrained in all directions, whereas the left and right edge nodes are restrained in the x and z directions only.

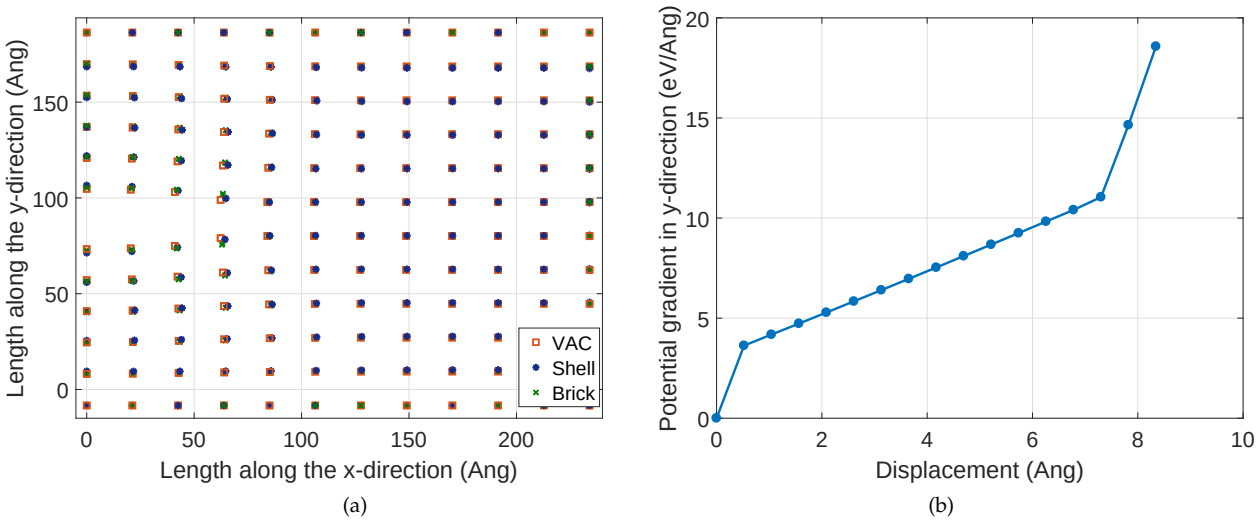

Figure 10: (a) Comparison of the nodal positions estimated considering the 8-node brick, the solid-shell and the VAC model based coarse regions, at the end of simulation. (b) Load-displacement diagram as the crack opens, considering the VAC model based coarse region.

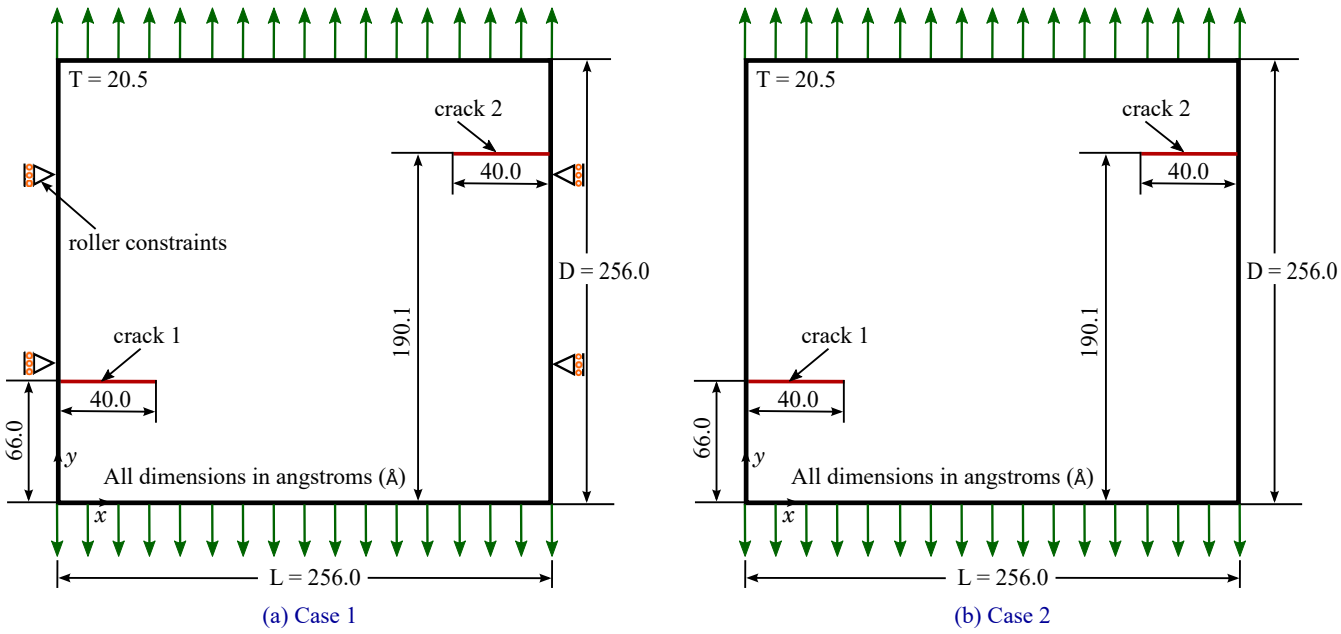

Figure 11: Schematic of the double edge crack simulations, showing the geometry, domain dimensions along with the loads and boundary conditions. (a) Roller constraints along the left and right edges are specified in case 1, such that the nodes will move only along the y-direction. Whereas, (b) no roller constraints are prescribed in case 2.

Figure 15(d) shows the deformed configuration of the multiscale model after 45 load steps. A close up view of the regions around the crack tips of the deformed configuration in Fig. 15(d) are shown in Fig. 15(e) and (f), respectively. The potential energy of the atoms around the crack tip is observed to be higher compared to their neighbors. From Fig. 15(e) and (f), the bonds of the atoms around the crack tip are ready to be broken. A deformed configuration after 47 load steps is plotted in Fig. 16(g), where a close up view of the regions around the crack tips in Fig. 15(g) are shown in Fig. 15(h) and (i), respectively. Comparing Fig. 15(e), (f) and Fig. 15(h), (i), few bonds around the crack tip are broken and hence the distance between the atoms has been increased, marking the onset of crack growth. Correspondingly, a small drop in the potential energy is noticed Fig. 13(a). Further drops in the potential energy indicates subsequent instances of crack growth. Furthermore, since the atoms on the crack tip in Fig.15(h) and (i) have completely crossed the element boundaries, two new split elements are added to the existing set of split elements. The total number of phantom nodes is accordingly increased, and hence the connectivity table is updated accordingly. Deformed configuration after 52 load steps is shown in Fig. 15(j-l). Therefore, cracks started to propagate in the opposite directions after 47 load steps. Continued application of the load leads to growth of cracks, finally reaching the vertical edges of the fine scale

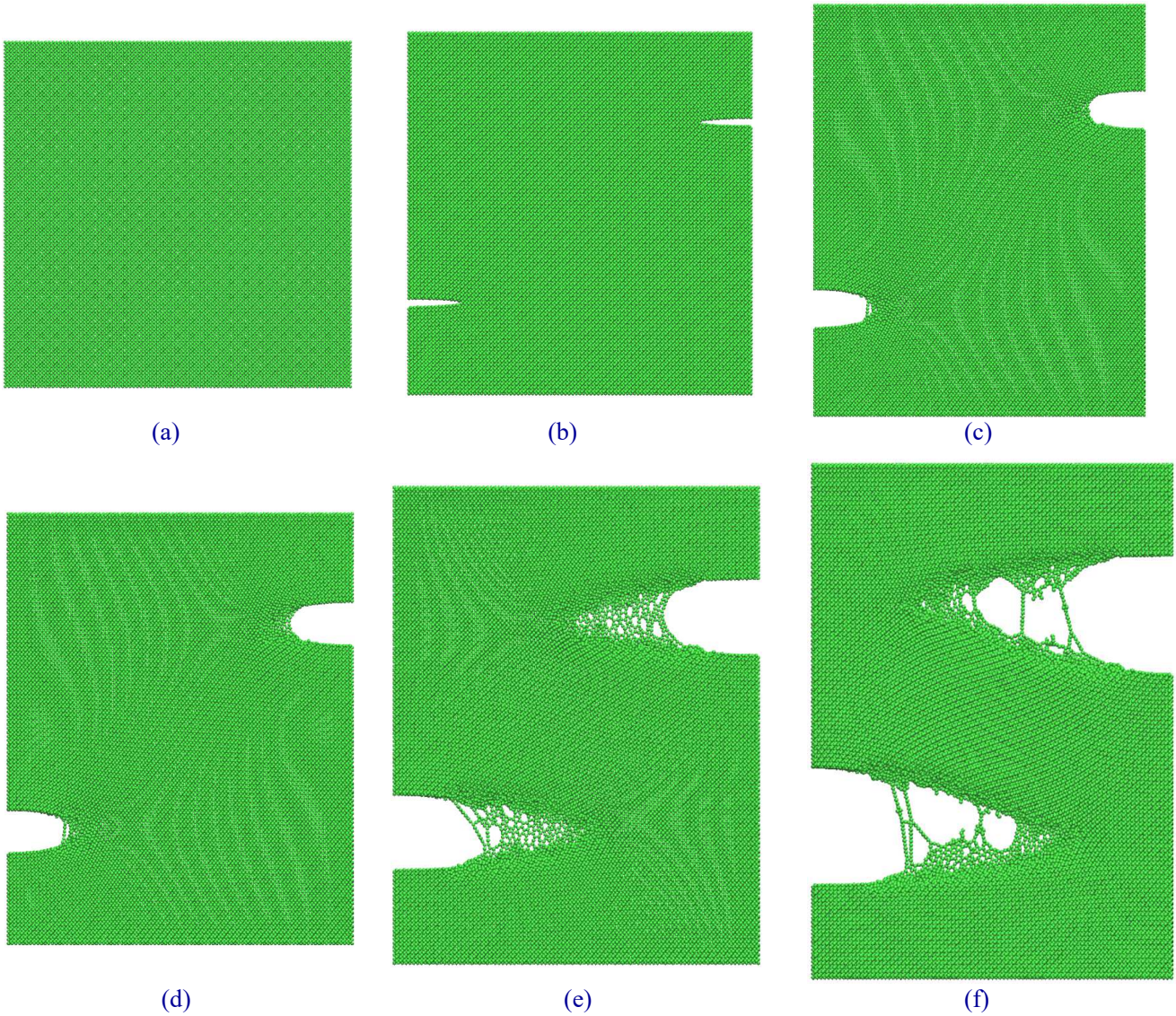

Figure 12: Crack growth simulations of a double edge notched specimen with roller constraints, case 1 as shown Fig. 11(a). (a) Initial and deformed configurations after (b) 200 steps, (c) 800 steps, (d) 820 steps, (e) 1000 steps and (f) 1400 steps. Cracks starts to open until 800 steps, where the first bonds are observed to be broken after 820 steps. Continuous loading leads to further in opposite directions. The roller constraints on the left and right edges provide extra stiffness leading to fast crack growth.

region. Since crack growth in Silicon is a brittle phenomenon, in order to avoid sudden fracture and hence the crack tips crossing out of the fine scale regions, adaptivity schemes are activated at this point. Adaptivity scheme explained in [14, 33] is adopted in this study. Deformed configurations of the coarse model after 72, 75 and 90 load steps are shown in Fig. 16, where a significant crack growth has been observed.

6.3. Example 3: Edge crack growth simulations

In this example, Mode I crack growth of a through-the-thickness edge crack, located in the middle of the plate along the y direction has been studied. Displacement boundary conditions are specified on the top and bottom row nodes of the plate in the continuum region, Ω_C. To estimate the location of the crack tip and hence the orientation of the crack growth, an atomistic region, Ω_A is considered around the crack tip. In every load step, the information on the location of the crack tip and the orientation crack surface estimated in the fine scale region, is sent back to the coarse region. The process is repeated until the end of simulation.

Consider a three dimensional coarse scale domain with dimensions 180.0 Å$\times$180.0 Å$\times$18.6 Å. Initial edge notch of length 30 Å along the x direction, located at 90.0 Å in the middle of the domain along the y direction, are created on the left and right edges in the coarse scale model, see Fig. 17(a). The continuum region is discretized using 40$\times$40 nodes along the x and y directions, respectively. The crack in the coarse scale domain is modeled by the phantom node method. Therefore, phantom nodes are created on the completely cracked elements.

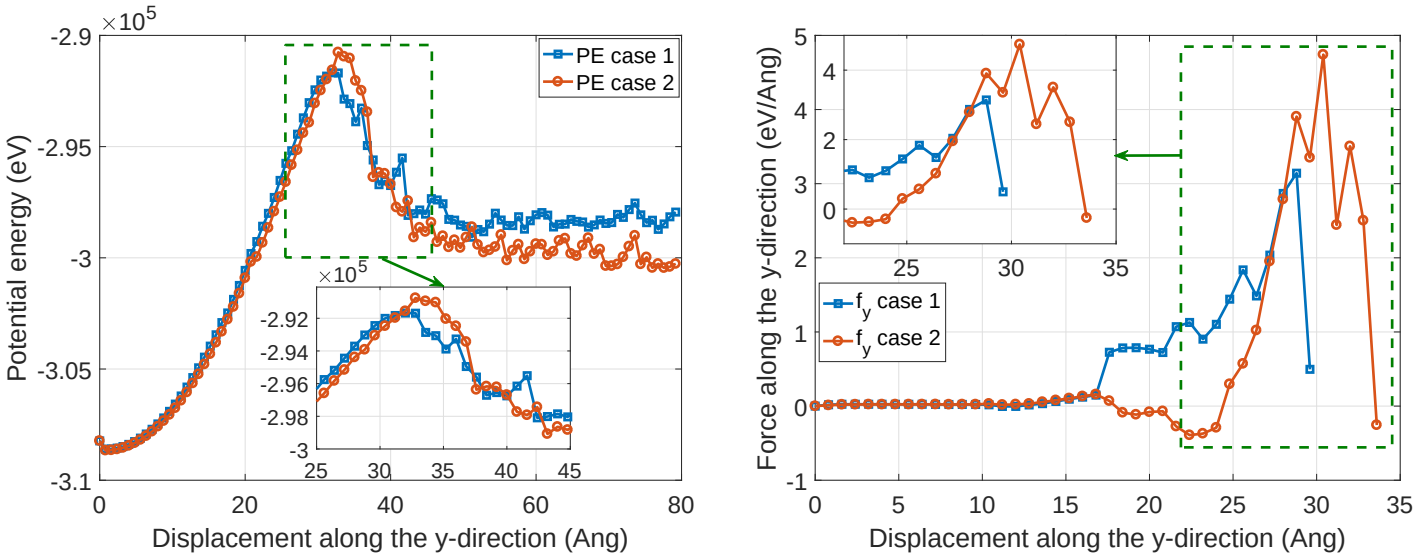

Figure 13: (a) Variation of the potential energy with displacement along the y-direction for cases 1 and 2, see Fig. 11. A closeup around the peak is shown in the inset. (b) Distribution of the force along the y-direction with the y-displacement, until the breakage of first bonds, i.e. up to 850 load steps. The inset picture shows a comparison of the cases 1 and 2 at the time of first bond breaking.

A fine scale region measuring 75.69 Å×73.84 Å×18.6 Å with 4243 active atoms and 998 ghost atoms has been created as shown in Fig. 17(a). An isometric view of Fig. 17(a), showing the discretization along the thickness direction is plotted in Fig. 8(d). A uniform displacement of 22 Å is specified on the top and bottom boundaries of the coarse scale model, in 60 pseudo-time steps. Both the top and bottom edge nodes are restrained in all directions, whereas the left and right edge nodes are restrained in the x and z directions only.

Figure 17(b) shows the deformed configuration of the multiscale model after 25 load steps. A close up of the region around the crack tip of the deformed configuration in Fig. 17(b) is shown in Fig. 17(c). Note that the potential energy of the atoms around the crack tip is the higher compared to their neighbors, indicating the onset of crack growth. From Fig. 17(c), the bond lengths of the atoms around the crack tip are about to increase and hence the first bonds are ready to be broken. A deformed configuration after 39 and 42 load steps is plotted in Fig. 17(e) and Fig. 17(f) respectively. Comparing Fig. 17(e) and (f), we noticed that the crack propagation is brittle after the onset of crack growth.

The distribution of the potential energy of the fine scale region with the strain (ϵ_y) along the y direction in each load step is plotted in Fig. 18. Strain in the current step is estimated as the ratio of the change in length (displacement) to the original length. Therefore, ϵ_y is given by the ratio of displacement along the y direction in the current load step to the original length along the y direction. A continuous rise in the potential energy is observed in Fig. 18, until the strain reaches a value around 0.08 in 25 load steps. At this loading stage, a small drop in the potential energy is noticed, which corresponds to the breakage of the first Silicon bonds leading to crack growth. Further drop in the potential energy indicates the subsequent instances of crack growth. However, a sudden drop in the potential energy is noticed when the strain reaches a value around 0.12 after 40 load steps. This corresponds to brittle fracture, where the crack opened instantly, see Fig. 17(f).

7. Conclusions

In this work, a continuum-based PNM is combined with a VAC based continuum formulations coupled with a molecular statics model to generate a multiscale framework for the simulation of crack growth in three dimensions. The use of the PNM over the standard XFEM formulation has been motivated due to its simpler numerical implementation.

Atomistic simulations in the fine scale region are carried out by triggering the LAMMPS software through the `system` command in an in-house MATLAB code. Cracks have been incorporated into the fine scale model by deleting atomic bonds. The PNM has been used to simulate crack propagation in the coarse scale.

Coupling of the continuum and atomistic models has been realized through the use of a bridging scale method. The original BSM has been additionally enhanced with respect to its original formulation, so that arbitrary cracks are admissible at the coarse scale using the PNM. Therefore, the proposed numerical strategy can be categorized as a novel three-dimensional adaptive multiscale method (TAMM) for thin structures.

To demonstrate the modeling capacity of the current TAMM, it has been used to simulate crack propagation in three scenarios with regard to Silicon specimens with a cubic diamond lattice structure containing propagating cracks.

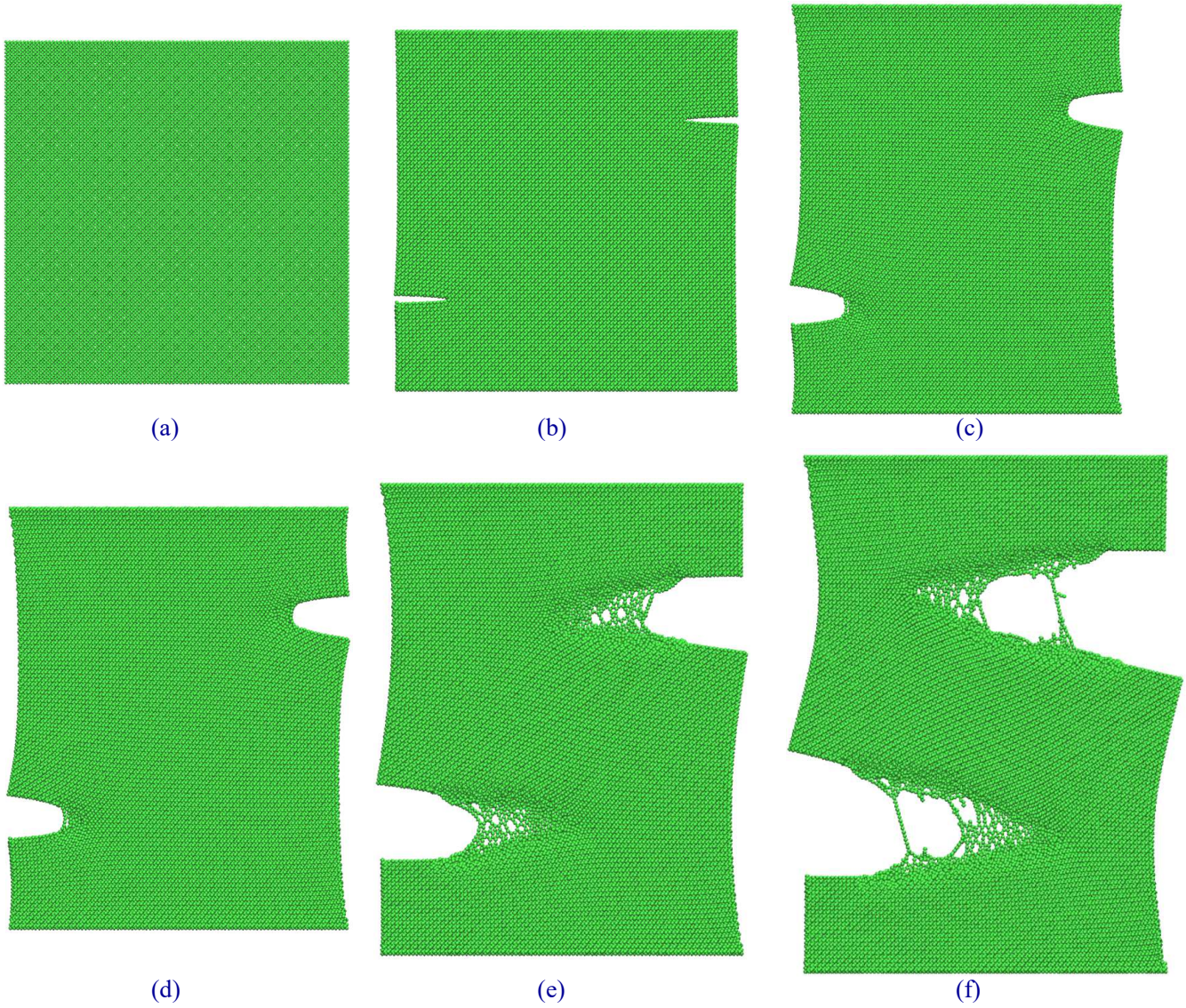

Figure 14: Crack growth simulations of a double edge notched specimen with roller constraints, case 2 as shown Fig. 11(b). (a) Initial and deformed configurations after (b) 200 steps, (c) 800 steps, (d) 834 steps, (e) 1000 steps and (f) 1400 steps. Cracks starts to open until 800 steps, where the first bonds are observed to be broken after 834 steps. Continuous loading leads to further in opposite directions. Lack of roller constraints on the left and right edges leads slow growth cracks, compared to the case with roller constraints shown in Fig. 14.

References

[1] P Egan, R Sinko, PR LeDuc, and S. Keten. The role of mechanics in biological and bio-inspired systems. *Nature communications*, 6:7418, 2005.

[2] Z Tang, NA Kotov, S Magonov, and B. Ozturk. Concurrent multiresolution finite element: formulation and algorithmic aspects. *Nanostructured artificial nacre*, 2:U413–U418, 2003.

[3] FF Abraham, R Walkup, H Gao, M Duchaineau, TD DeLaRubia, and M. Seager. Simulating materials failure by using up to one billion atoms and the world's fastest computer: Work-hardening. *Proceedings of the National Academy of Sciences*, 99(9):5777–5782, 2002.

[4] MJ Buehler, A Hartmaier, H Gao, M Duchaineau, and FF. Abraham. Atomic plasticity: description and analysis of a one-billion atom simulation of ductile materials failure. *Computer Methods in Applied Mechanics and Engineering*, 193(48-51):5257–5282, 2004.

[5] EB Tadmor, M Ortiz, and R. Phillips. Quasicontinuum analysis of defects in solids. *Philosophical Magazine A*, 73(6):1529–1563, 1996.

[6] WK Liu, H Su, T Belytschko, S Li, and CT. Chang. Multi-scale methods. *International Journal for Numerical Methods in Engineering*, 47:1343–1361, 2000.

[7] RE Miller and EB. Tadmor. The quasicontinuum method: Overview, applications and current directions. *Journal of Computer-Aided Materials Design*, 9(3):203–239, 2002.

[8] T Belytschko and SP. Xiao. Coupling methods for continuum model with molecular model. *International Journal for Multiscale Computational Engineering*, 1(1):115–126, 2003.

[9] SP Xiao and T. Belytschko. A bridging domain method for coupling continua with molecular dynamics. *Computer Methods in Applied Mechanics and Engineering*, 193(17-20):1645–1669, 2004.

[10] WK Liu, HS Park, D Qian, EG Karpov, H Kadowaki, and GJ. Wagner. Bridging scale methods for nanomechanics and materials. *Computer Methods in Applied Mechanics and Engineering*, 195(13-16):1407–1421, 2006.

[11] T Belytschko, S Loehnert, and JH. Song. Multiscale aggregating discontinuities: A method for circumventing loss of material stability. *International Journal for Numerical Methods in Engineering*, 73(6):869–894, 2008.

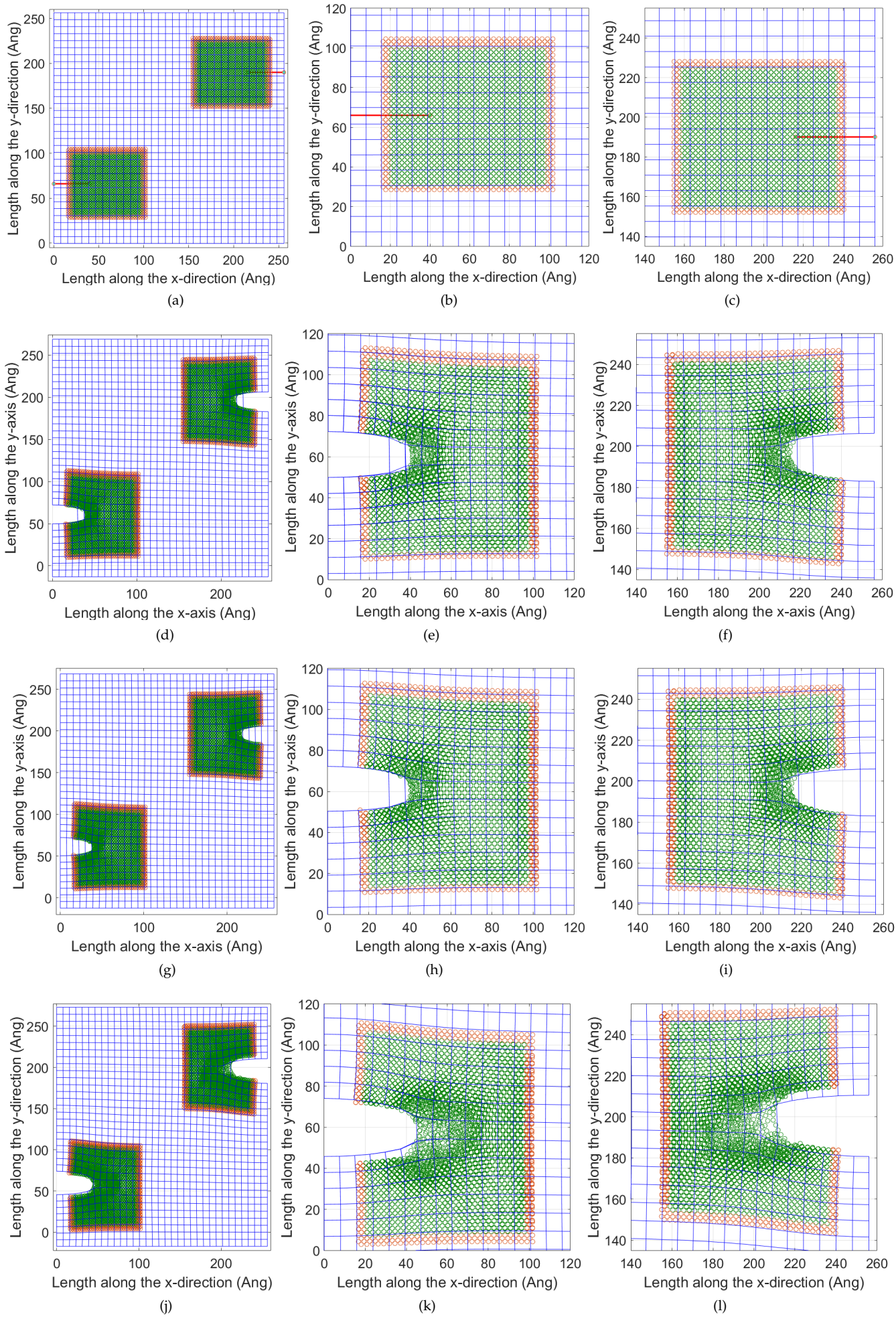

Figure 15: Crack growth simulations based on the multiscale model. The initial configuration is shown in the first row. Deformed configurations after 45, 47 and 52 load steps are shown in second, third and fourth rows, respectively. The first column pictures shows the deformed configurations of the multiscale model and the second and third columns pictures represent a closeup around the lower and upper atomistic regions.

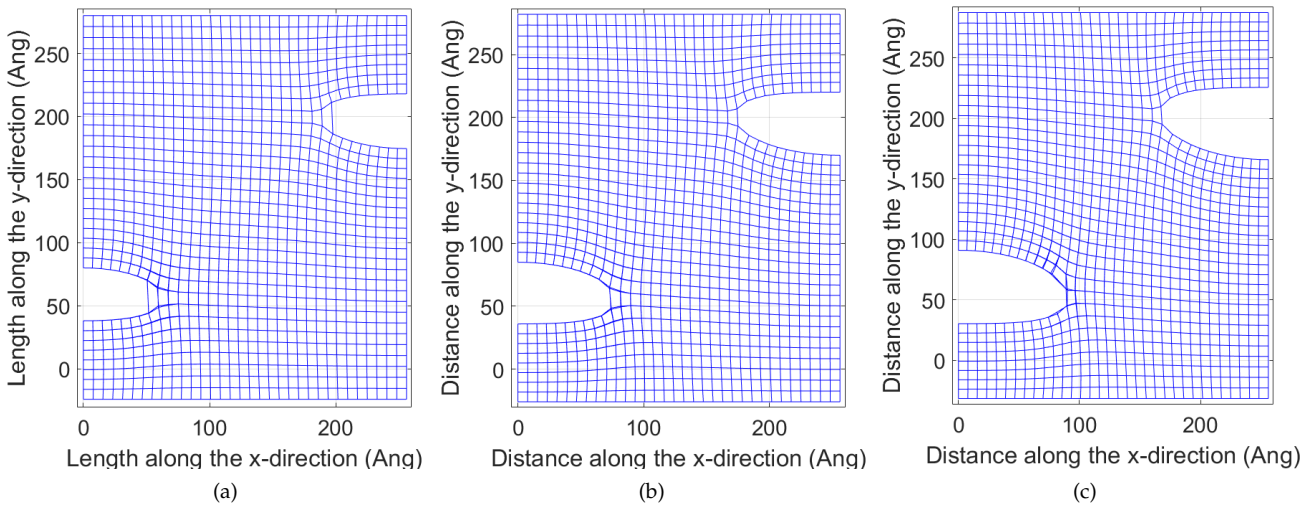

Figure 16: Deformed configurations of the coarse region after 72, 75 and 90 load steps are shown in (a), (b) and (c), respectively.

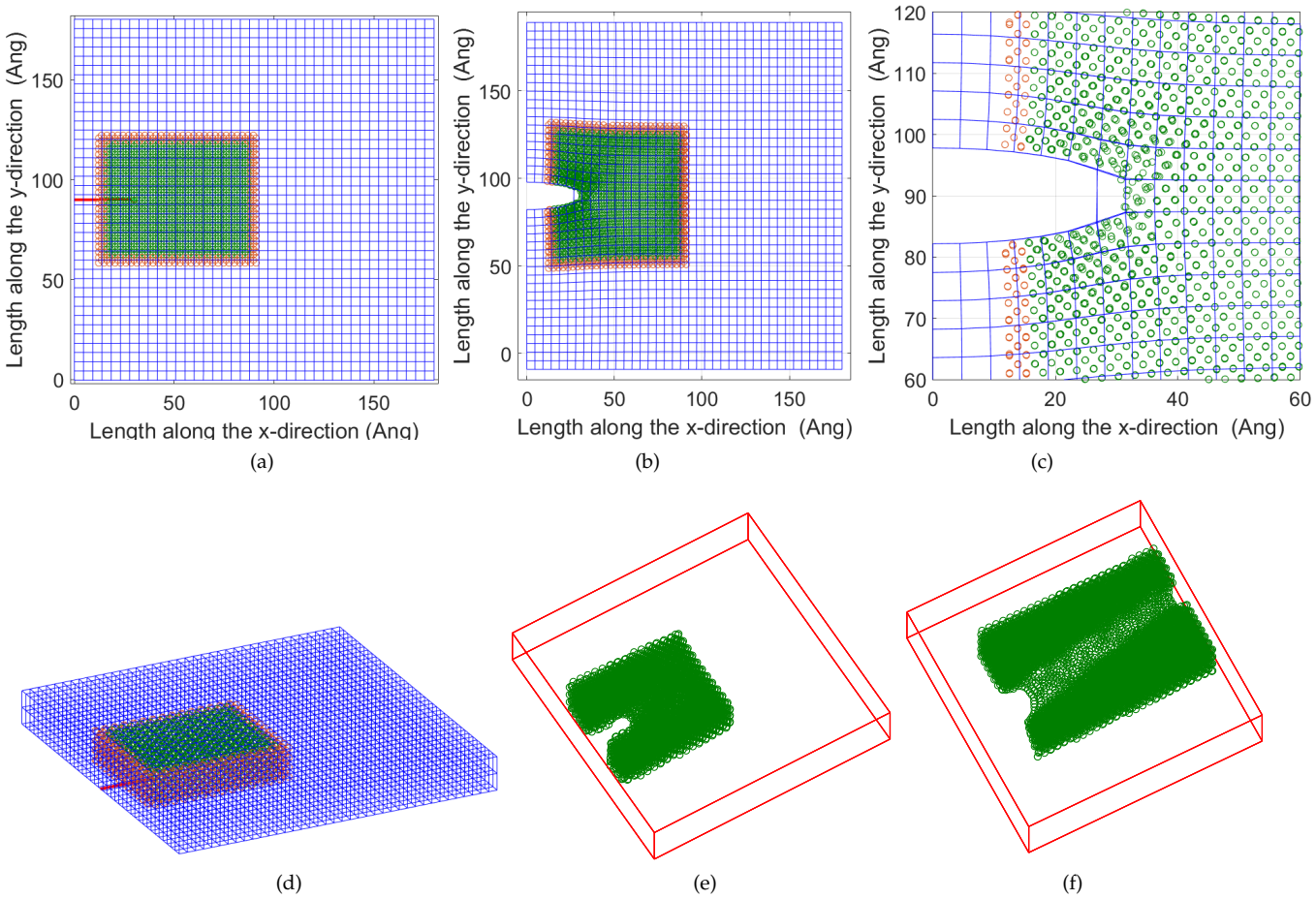

Figure 17: Crack propagation in an initial edge crack specimen. (a) Initial configuration of the multiscale model showing the discretized coarse region, geometry of the crack and the atoms in the fine scale region along with the ghost atoms along the boundaries of the fine scale region. (b) Deformed configuration after 25 load steps, where a close-up around the crack tip is shown in (c). (d) Isometric view of (a). Isometric views of the deformed configurations after 39 and 42 load steps are shown in (e) and (f), respectively.

[12] R Gracie and T. Belytschko. Concurrently coupled atomistic and XFEM models for dislocations and cracks. *International Journal for Numerical Methods in Engineering*, 78(3):354–378, 2008.

[13] H Talebi, M Silani, SPA Bordas, P Kerfriden, and T. Rabczuk. A computational library for multiscale modelling of material failure. *Computational Mechanics*, 53(5):1047–1071, 2014.

[14] PR Budarapu, R Gracie, SPA Bordas, and T. Rabczuk. An adaptive multiscale method for quasi-static crack growth. *Computational*

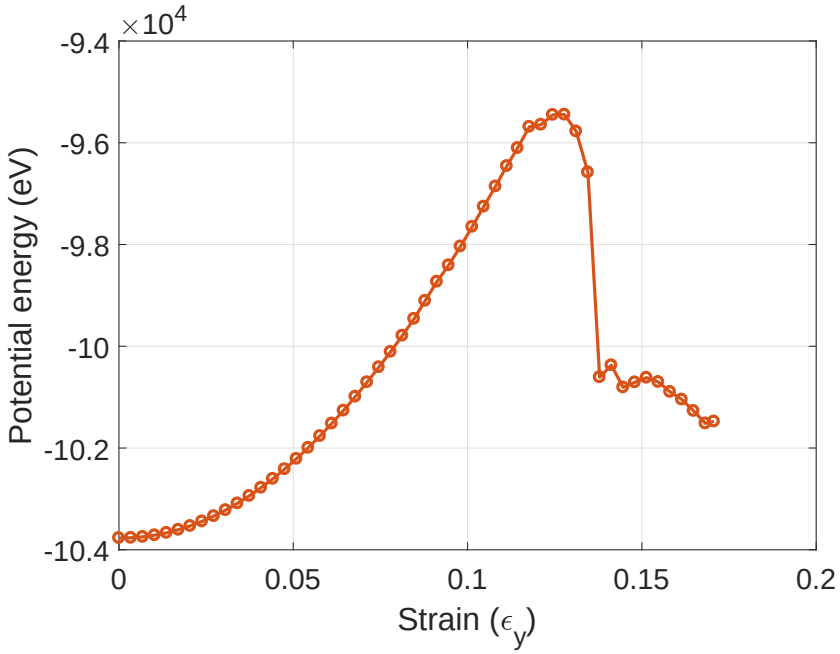

Figure 18: Distribution of the potential energy as function of strain.

Mechanics, 53(6):1129–1148, 2014. doi:10.1007/s00466-013-0952-6.

[15] H Hajibeygi, D Karvounis, and P. Jenny. A hierarchical fracture model for the iterative multiscale finite volume method. *Journal of Computational Physics*, 230:8729–8743, 2011.

[16] M Paggi and P. Wriggers. Stiffness and strength of hierarchical polycrystalline materials with imperfect interfaces. *Journal of the Mechanics and Physics of Solids*, 60:557–572, 2012.

[17] I Wudtke, H Talebi, M Silani, and F. Werner. A hierarchical multi-scale approach to mechanical characterization of heat affected zone in welded connections. *Computational Materials Science*, 96:396–402, 2015.

[18] WB Lawrimore, B Paliwal, MQ Chandler, KL Johnson, and MF. Horstemeyer. Hierarchical multiscale modeling of polyvinyl alcohol/montmorillonite nanocomposites. *Polymer*, 99:386–398, 2016.

[19] D Lyu, H Fan, and S. Li. A hierarchical multiscale cohesive zone model and simulation of dynamic fracture in metals. *Engineering Fracture Mechanics*, 163:327–347, 2016.

[20] F. Feyel. Multiscale FE^2 elastoviscoplastic analysis of composite structures. *Computational Materials Science*, 16(1-4):344–354, 1999.

[21] F Feyel and JL. Chaboche. FE^2 multiscale approach for modeling the elastoviscoplastic behavior of long fiber sic/ti composite materials. *Computer Methods in Applied Mechanics and Engineering*, 183:309–330, 2000.

[22] F Feyel and JL. Chaboche. A multilevel finite element method (FE^2) to describe the response of highly non-linear structures using generalized continua. *Computer Methods in Applied Mechanics and Engineering*, 192:3233–3244, 2003.

[23] H Talebi, G Zi, M Silani, E Samaniego, and T. Rabczuk. A simple circular cell method for multi-level finite element analysis. *Journal of Applied Mathematics*, ID:526846, 2012.

[24] M Silani, S Ziaei-Rad, H Talebi, and T. Rabczuk. A semi-concurrent multiscale approach for modeling damage in nanocomposites. *Theoretical and Applied Fracture Mechanics*, 74:30–38, 2014.

[25] H Zhu, Q Wang, and X. Zhuang. A nonlinear semi-concurrent multiscale method for fractures. *International Journal of Impact Engineering*, 87:65–82, 2016.

[26] GJ Wagner and WK. Liu. Coupling of atomistic and continuum simulations using a bridging scale decomposition. *Journal of Computational Physics*, 190(1):249–279, 2003.

[27] DE Farrell, HS Park, and WK. Liu. Implementation aspects of the bridging scale method and application to intersonic crack propagation. *International Journal for Numerical Methods in Engineering*, 71:583–605, 2007.

[28] PA Guidault and T. Belytschko. Bridging domain methods for coupled atomisticâ€“continuum models with l^2 or h^1 couplings. *International Journal for Numerical Methods in Engineering*, 77(4-5):1566–1592, 2009.

[29] J Broughton, FF Abraham, N Bernstein, and E. Kaxiras. Concurrent coupling of length scales: methodology and application. *Physical Review B*, 60(4):2391–2403, 1999.

[30] H Talebi, M Silani, SPA Bordas, P Kerfriden, and T. Rabczuk. Molecular dynamics/XFEM coupling by a three dimensional extended bridging domain with applications to dynamic brittle fracture. *International Journal for Multiscale Computational Engineering*, 11(6):527–541, 2013.

[31] S-W Yang, PR Budarapu, DR Mahapatra, SPA Bordas, G Zi, and T. Rabczuk. A meshless adaptive multiscale method for fracture. *Computational Materials Science*, 96B:382–395, 2015. doi:10.1016/j.commatsci.2014.08.054.

[32] H Talebi, M Silani, and T. Rabczuk. Concurrent multiscale modeling of three dimensional crack and dislocation propagation. *Advances in Engineering Software*, 80:82–92, 2015.

[33] PR Budarapu, J Reinoso, and M. Paggi. Concurrently coupled solid shell-based adaptive multiscale method for fracture. *Computer Methods in Applied Mechanics and Engineering*, 319:338–365, 2017. DOI: 10.1016/j.cma.2017.02.023.

[34] F Greco, L Leonetti, R Luciano, and PN. Blasi. An adaptive multiscale strategy for the damage analysis of masonry modeled as a composite material. *Composite Structures*, 153:972–988, 2016.

[35] D Qian, GJ Wagner, and WK. Liu. A multiscale projection method for the analysis of carbon nanotubes. *Computer Methods in Applied Mechanics and Engineering*, 193(17-20):1603–1632, 2003.

[36] S. Plimpton. Fast parallel algorithms for short-range molecular dynamics. *Journal of Computational Physics*, 117:1–19, 1995.

[37] A Hansbo and P. Hansbo. A finite element method for the simulation of strong and weak discontinuities in solid mechanics. *Computer*

Variable	Silicon (Si)	Carbon (C)
$\mathcal{R}$ (Å)	2.7	1.8
$\mathcal{S}$ (Å)	3.0	2.1
$\mathcal{A}$ (eV)	1.8308×10^3	1.3936×10^3
$\mathcal{B}$ (eV)	4.7118×10^2	3.4670×10^2
$\mathcal{D}$ (Å^{-1})	2.4799	3.4879
$\mathcal{E}$ (Å^{-1})	1.7322	2.2119
$\mathcal{P}$	1.1000×10^{-6}	1.5724×10^{-7}
q	7.8734×10^{-1}	7.2751×10^{-1}
c	1.0039×10^5	3.8049×10^4
d	1.6217×10^1	4.3840×10^0
h	-5.9825×10^{-1}	-5.7058×10^{-1}

Table A.1: Parameters for Silicon and Carbon to be used in Eq. (3), adopted from [40]. $\mathcal{R}$ and $\mathcal{S}$ were not systematically optimized.

Methods in Applied Mechanics and Engineering, 193(33-35):3523–3540, 2004.
[38] JH Song, PMA Areias, and T. Belytschko. A method for dynamic crack and shear band propagation with phantom nodes. *International Journal for Numerical Methods in Engineering*, 67(6):868–893, 2006.
[39] T Rabczuk, G Zi, A Gerstenberger, and WA. Wall. A new crack tip element for the phantom node method with arbitrary cohesive cracks. *International Journal for Numerical Methods in Engineering*, 75(5):577–599, 2008.
[40] J. Tersoff. Modeling solid-state chemistry: Interatomic potentials for multicomponent systems. *Physical Review B*, 39(8):5566–5568, 1989.
[41] PR Budarapu, R Gracie, S-W Yang, X Zhuang, and T. Rabczuk. Efficient coarse graining in multiscale modeling of fracture. *Theoretical and Applied Fracture Mechanics*, 69:126–143, 2014. doi:10.1016/j.tafmec.2013.12.004.
[42] R Gracie and T. Belytschko. Adaptive continuum-atomistic simulations of dislocation dynamics. *International Journal for Numerical Methods in Engineering*, 86(4-5):575–597, 2011.
[43] J Mergheim, E Kuhl, and P. Steinmann. A finite element method for the computational modelling of cohesive cracks. *International Journal for Numerical Methods in Engineering*, 63:276–289, 2005.
[44] T Chau-Dinh, G Zi, PS Lee, JH Song, and Rabczuk T. Phantom-node method for shell models with arbitrary cracks. *Computers and Structures*, 92-93:242–256, 2010.

AppendixA. Tersoff potential function

The mathematical expression of the bond energy based on the Tersoff potential is given by Eq. (3). The bond energy in the Tersoff framework is a combination of repulsive (f_R) energy function which is exponentially decaying and attractive (f_A) energy function that exponentially increases; with the increase of distance between the atoms. f_c is a smooth spherical cutoff function around atom α based upon the distance to the first nearest-neighbor shell. The function f_c in Eq. (3) is defined as [40]:

$$f_c(r_{\alpha\beta}) = \begin{cases} 1 & \text{when } r_{\alpha\beta} < \mathcal{R}_{\alpha\beta} \\ \frac{1}{2} + \frac{1}{2}\cos\left(\frac{\pi(r_{\alpha\beta}-\mathcal{R}_{\alpha\beta})}{(\mathcal{S}_{\alpha\beta}-\mathcal{R}_{\alpha\beta})}\right) & \text{when } \mathcal{R}_{\alpha\beta} < r_{\alpha\beta} < \mathcal{S}_{\alpha\beta} \\ 0 & \text{when } r_{\alpha\beta} > \mathcal{S}_{\alpha\beta} \end{cases} \tag{A.1}$$

from Eq. (A.1), f_c returns a value of 1 if $r_{\alpha\beta}$ is less than $\mathcal{R}_{\alpha\beta}$ and 0 when $r_{\alpha\beta}$ greater than $\mathcal{S}_{\alpha\beta}$. The values of the constants $\mathcal{R}_{\alpha\beta} = \sqrt{\mathcal{R}_\alpha \mathcal{R}_\beta}$ and $\mathcal{S}_{\alpha\beta} = \sqrt{\mathcal{S}_\alpha \mathcal{S}_\beta}$, (where α and β can be two different atom types, like Silicon and Carbon) are listed for Silicon and Carbon atoms in table (A.1). The repulsive and attractive potential energies are tuned with the parameters $\mathcal{A}_{\alpha\beta}$ and $\mathcal{B}_{\alpha\beta}$, respectively. The repulsive potential energy is defined as [40]

$$f_R(r_{\alpha\beta}) = \mathcal{A}_{\alpha\beta} e^{-\mathcal{D}_{\alpha\beta} r_{\alpha\beta}} \tag{A.2}$$

and the attractive potential energy is estimated from [40]:

$$f_A(r_{\alpha\beta}) = -\mathcal{B}_{\alpha\beta} e^{-\mathcal{E}_{\alpha\beta} r_{\alpha\beta}} \tag{A.3}$$

where $\mathcal{A} = \sqrt{\mathcal{A}_\alpha \mathcal{A}_\beta}$, $\mathcal{B} = \sqrt{\mathcal{B}_\alpha \mathcal{B}_\beta}$, $\mathcal{D}_{\alpha\beta} = (\mathcal{D}_\alpha + \mathcal{D}_\beta)/2$ and $\mathcal{E}_{\alpha\beta} = (\mathcal{E}_\alpha + \mathcal{E}_\beta)/2$, in equations (A.2) and (A.3), are the constants listed in table (A.1). The variable $b_{\alpha\beta}$ in Eq. (3) is designed to represent the bond strength of the potential, where $b_{\alpha\beta}$ is inversely proportional to the coordination number, given by [40]:

$$b_{\alpha\beta} = \xi_{\alpha\beta}\left(1 + \mathcal{P}_\alpha^{q_\alpha} \zeta_{\alpha\beta}^{q_\alpha}\right)^{-1/2q_\alpha} \tag{A.4}$$

where $\mathcal{P}$ and q are the constants listed in table (A.1). $\zeta_{\alpha\beta}$ provides a weighted measure of the number of other bonds (γ) competing with the bond α-β, which is defined as [40]:

$$\zeta_{\alpha\beta} = \sum_{\gamma\neq\alpha,\beta} f_c\left(r_{\alpha\gamma}\right) g\left(\theta_{\alpha\beta\gamma}\right) \tag{A.5}$$

where $\xi_{\alpha\beta}$ is the strengthening or weakening factor of the hetero-polar bonds and $g\left(\theta_{\alpha\beta\gamma}\right)$ provides a measure of dependence on the bonding angle $\theta_{\alpha\beta\gamma}$, subtended at atom α by atoms β and γ. The variable $g\left(\theta_{\alpha\beta\gamma}\right)$ is included to stabilize the atomic geometry under shear operations and to provide an effective coordination contribution based on the elastic energy of the current configuration, which is defined as :[40]:

$$g\left(\theta_{\alpha\beta\gamma}\right) = 1 + \frac{c_\alpha^2}{d_\alpha^2} - \frac{c_\alpha^2}{d_\alpha^2 + \left(h_\alpha - \cos\left(\theta_{\alpha\beta\gamma}\right)\right)^2} \tag{A.6}$$

where c_α, d_α and h_α are the constants listed in table(A.1). Details of the derivation of the first derivative of the potential function in Eq. (3) is given in appendix. The energy minimization is carried out using the conjugate gradient method and hence a tangent stiffness matrix is not required.

AppendixB. Fine scale model

In molecular statics (MS) the goal is to determine the positions of the atoms for the given boundary conditions, by minimizing the system's potential energy. The total potential energy of the system is expressed as

$$\Pi = W^{\text{int}} - W^{\text{ext}} \tag{B.1}$$

where W^{int} represents the internal energy of the system and W^{ext} is the external work done on the system. Consider the simplest atom-atom interactions in which the potential energy is only a function of the distance between two atoms, the total internal energy of the system is the summation of the bond energies of all the atoms, as given below:

$$W^{\text{int}} = \sum_{\alpha=1}^{n^{\text{A}}} \phi_\alpha = \frac{1}{2}\sum_{\alpha=1}^{n^{\text{A}}}\sum_{\beta\neq\alpha}^{n^{\text{A}}} V(r_{\alpha\beta}) \tag{B.2}$$

where ϕ_α is the potential energy associated with atom α, as defined below

$$\phi_\alpha = \frac{1}{2}\sum_{\beta\neq\alpha}^{n^{\text{A}}} V(r_{\alpha\beta}) \tag{B.3}$$

and $V(r_{\alpha\beta})$ is the bond potential between the atoms α and β, separated by distance $r_{\alpha\beta}$. The system potential energy is minimized, when the first derivative of the potential function with respect to the positions of the atoms is equal to zero. Therefore, the internal and external forces acting on any given atom λ based on the minimization of energy are given by [14]:

$$\mathbf{F}_\lambda^{\text{int}} = \frac{1}{2}\sum_{\alpha=1}^{n^{\text{A}}}\sum_{\beta\neq\lambda}^{n^{\text{A}}} -\frac{\partial V(r_{\alpha\beta})}{\partial r_{\alpha\beta}}\frac{\partial r_{\alpha\beta}}{\partial \mathbf{r}_\lambda} = -\sum_{\beta\neq\alpha}^{n^{\text{A}}} \frac{\partial V(r_{\alpha\beta})}{\partial r_{\alpha\beta}}\left(\frac{\mathbf{r}_\alpha - \mathbf{r}_\beta}{r_{\alpha\beta}}\right) \tag{B.4a}$$

$$\mathbf{F}_\lambda^{\text{ext}} = -\frac{\partial W^{\text{ext}}}{\partial \mathbf{r}_\lambda}. \tag{B.4b}$$

The residual forces on each atom is can be estimated as: $\mathbf{R} = \mathbf{F}^{\text{int}} - \mathbf{F}^{\text{ext}}$. Details of the above derivation and the computer implementation steps are explained in [14]. In the present work, the atom to atom interactions between the atoms in graphene and Silicon are modeled based on the Tersoff potential [40]. The detailed of the internal forces in Eq. (B.4)(a) in the framework of the Tersoff potential are explained in AppendixC.

AppendixC. Derivation of the internal forces: fine scale

The internal forces in Eq. (B.4)(a) are evaluated by considering the first derivative of the bond potential function (V($r_{\alpha\beta}$)) in Eq. (3), with respect to the position of a particular atom λ in a particular direction k, i.e.,

$$\begin{aligned}\frac{\partial V(r_{\alpha\beta})}{\partial r_{\lambda k}} &= \frac{\partial(f_c(r_{\alpha\beta})[f_R(r_{\alpha\beta}) + b_{\alpha\beta}f_A(r_{\alpha\beta})])}{\partial r_{\lambda k}} \\ &= f_c\left(r_{\alpha\beta}\right)\left[\frac{\partial f_R(r_{\alpha\beta})}{\partial r_{\lambda k}} + b_{\alpha\beta}\frac{\partial f_A(r_{\alpha\beta})}{\partial r_{\lambda k}} + \frac{\partial b_{\alpha\beta}}{\partial r_{\lambda k}}f_A(r_{\alpha\beta})\right] + \frac{\partial f_c(r_{\alpha\beta})}{\partial r_{\lambda k}}[f_R(r_{\alpha\beta}) + b_{\alpha\beta}f_A(r_{\alpha\beta})].\end{aligned} \tag{C.1}$$

The terms, $\frac{\partial f_R(r_{\alpha\beta})}{\partial r_{\lambda k}}$, $\frac{\partial f_A(r_{\alpha\beta})}{\partial r_{\lambda k}}$ and $\frac{\partial f_c(r_{\alpha\beta})}{\partial r_{\lambda k}}$ in Eq. (C.1) are further split, based on the chain rule of the derivatives as shown below:

$$\frac{\partial f(r_{\alpha\beta})}{\partial r_{\lambda k}} = \frac{\partial f(r_{\alpha\beta})}{\partial r_{\alpha\beta}} \frac{\partial r_{\alpha\beta}}{\partial r_{\lambda k}} \tag{C.2}$$

where $r_{\alpha\beta}$ is defined as

$$r_{\alpha\beta} = |\mathbf{r}_\alpha - \mathbf{r}_\beta| = \sqrt{\sum_{j=1}^{3} (r_{\alpha j} - r_{\beta j})^2}. \tag{C.3}$$

Therefore, the derivative $\frac{\partial r_{\alpha\beta}}{\partial r_{\lambda k}}$ in Eq. (C.2) can be evaluated by substituting definition of $r_{\alpha\beta}$ from Eq. (C.3)

$$\frac{\partial r_{\alpha\beta}}{\partial r_{\lambda k}} = \frac{\partial \sqrt{\sum_{j=1}^{3} (r_{\alpha j} - r_{\beta j})^2}}{\partial r_{\lambda k}} = \frac{(r_{\alpha j} - r_{\beta j})}{\sqrt{\sum_{j=1}^{3} (r_{\alpha j} - r_{\beta j})^2}} (\delta_{\alpha\lambda} - \delta_{\beta\lambda}) = \frac{r_{\alpha\beta j}}{r_{\alpha\beta}} (\delta_{\alpha\lambda} - \delta_{\beta\lambda}). \tag{C.4}$$

Lets consider two possible cases for λ. First, when $\lambda = \alpha$ and $\lambda \neq \beta$

$$\frac{\partial r_{\alpha\beta j}}{\partial r_{\alpha k}} = \frac{(r_{\alpha j} - r_{\beta j})}{r_{\alpha\beta}} (1) \tag{C.5}$$

and secondly, when $\lambda = \beta$ and $\lambda \neq \alpha$

$$\frac{\partial r_{\alpha\beta j}}{\partial r_{\beta k}} = \frac{(r_{\alpha j} - r_{\beta j})}{r_{\alpha\beta}} (-1). \tag{C.6}$$

Since α and β are interchangeable, after exchanging β with α and vice-versa, Eq. (C.6) becomes,

$$\frac{\partial r_{\beta\alpha j}}{\partial r_{\alpha k}} = \frac{(r_{\beta j} - r_{\alpha j})}{r_{\beta\alpha}} (-1) = \frac{(r_{\alpha j} - r_{\beta j})}{r_{\alpha\beta}} = \frac{r_{\alpha\beta j}}{r_{\alpha\beta}}. \tag{C.7}$$

Combining both cases in equations (C.5) and (C.7) for any arbitrary atom λ, yields

$$\frac{\partial r_{\alpha\beta}}{\partial \mathbf{r}_\lambda} = 2 \frac{(\mathbf{r}_\alpha - \mathbf{r}_\beta)}{r_{\alpha\beta}}. \tag{C.8}$$

The first derivative of the function f with respect to $r_{\alpha\beta}$ in Eq. (C.2) can now be estimated as

$$\frac{\partial f_R(r_{\alpha\beta})}{\partial r_{\alpha\beta}} = -\mathcal{D}_{\alpha\beta} \mathcal{A}_{\alpha\beta} e^{-\mathcal{D}_{\alpha\beta} r_{\alpha\beta}} \tag{C.9}$$

and

$$\frac{\partial f_A(r_{\alpha\beta})}{\partial r_{\alpha\beta}} = \mathcal{E}_{\alpha\beta} \mathcal{B}_{\alpha\beta} e^{-\mathcal{E}_{\alpha\beta} r_{\alpha\beta}}. \tag{C.10}$$

The cutoff function (f_c) is differentiable only in the range of $\mathcal{R}_{\alpha\beta} < r_{\alpha\beta} < \mathcal{S}_{\alpha\beta}$. Therefore,

$$\frac{\partial f_c(r_{\alpha\beta})}{\partial r_{\alpha\beta}} = -\frac{1}{2} \sin\left(\frac{\pi (r_{\alpha\beta} - \mathcal{R}_{\alpha\beta})}{\mathcal{S}_{\alpha\beta} - \mathcal{R}_{\alpha\beta}} \right) \frac{\pi}{\mathcal{S}_{\alpha\beta} - \mathcal{R}_{\alpha\beta}}. \tag{C.11}$$

Now, the derivative of $b_{\alpha\beta}$ in Eq. (C.1) is given by

$$\begin{aligned} \frac{\partial b_{\alpha\beta}}{\partial r_{\lambda k}} &= -\frac{1}{2q_\alpha} \xi_{\alpha\beta} \left(1 + \mathcal{P}_\alpha^{q_\alpha} \zeta_{\alpha\beta}^{q_\alpha} \right)^{(-1/2q_\alpha - 1)} q_\alpha \mathcal{P}_\alpha^{q_\alpha} \zeta_{\alpha\beta}^{q_\alpha - 1} \frac{\partial \zeta_{\alpha\beta}}{\partial r_{\lambda k}} \\ &= -\frac{1}{2} \frac{b_{\alpha\beta} \mathcal{P}_\alpha^{q_\alpha}}{\left(\zeta_{\alpha\beta}^{(1-q_\alpha)} + \mathcal{P}_\alpha^{q_\alpha} \zeta_{\alpha\beta} \right)} \frac{\partial \zeta_{\alpha\beta}}{\partial r_{\lambda k}} \end{aligned} \tag{C.12}$$

where

$$\frac{\partial \zeta_{\alpha\beta}}{\partial r_{\lambda k}} = \sum_{\gamma \neq \alpha,\beta} \left(\frac{\partial f_c(r_{\alpha\gamma})}{\partial r_{\alpha\gamma}} \frac{\partial r_{\alpha\gamma}}{\partial r_{\lambda k}} g(\theta_{\alpha\beta\gamma}) + f_c(r_{\alpha\gamma}) \frac{\partial g(\theta_{\alpha\beta\gamma})}{\partial \cos(\theta_{\alpha\beta\gamma})} \frac{\partial \cos(\theta_{\alpha\beta\gamma})}{\partial r_{\lambda k}} \right) \tag{C.13}$$

and

$$\frac{\partial g(\theta_{\alpha\beta\gamma})}{\partial \cos(\theta_{\alpha\beta\gamma})} = -\frac{2c_\alpha^2(h_\alpha - \cos(\theta_{\alpha\beta\gamma}))}{\left(d_\alpha^2 + (h_\alpha - \cos^2(\theta_{\alpha\beta\gamma}))\right)^2}. \tag{C.14}$$

The subtended angle $\cos(\theta_{\alpha\beta\gamma})$ can be evaluated from

$$\cos(\theta_{\alpha\beta\gamma}) = \frac{\mathbf{r}_{\alpha\beta} \cdot \mathbf{r}_{\alpha\gamma}}{r_{\alpha\beta} r_{\alpha\gamma}} = \frac{r_{\alpha\beta i} r_{\alpha\gamma i} + r_{\alpha\beta j} r_{\alpha\gamma j} + r_{\alpha\beta k} r_{\alpha\gamma k}}{r_{\alpha\beta} r_{\alpha\gamma}}. \tag{C.15}$$

Therefore, the derivative of $\cos(\theta_{\alpha\beta\gamma})$ with respect to $r_{\lambda k}$ can be evaluated as

$$\frac{\partial \cos(\theta_{\alpha\beta\gamma})}{\partial r_{\lambda k}} = \frac{r_{\alpha\beta} r_{\alpha\gamma} \frac{\partial(\mathbf{r}_{\alpha\beta} \cdot \mathbf{r}_{\alpha\gamma})}{\partial r_{\lambda k}} - \frac{\partial(r_{\alpha\beta} r_{\alpha\gamma})}{\partial r_{\lambda k}} (\mathbf{r}_{\alpha\beta} \cdot \mathbf{r}_{\alpha\gamma})}{(r_{\alpha\beta} r_{\alpha\gamma})^2} \tag{C.16}$$

where the term $\frac{\mathbf{r}_{\alpha\beta} \cdot \mathbf{r}_{\alpha\gamma}}{\partial r_{\lambda k}}$ in Eq. (C.16) can be calculated as

$$\begin{aligned} \frac{\partial(\mathbf{r}_{\alpha\beta} \cdot \mathbf{r}_{\alpha\gamma})}{\partial r_{\lambda k}} &= \frac{\partial(r_{\alpha\beta i} r_{\alpha\gamma i} + r_{\alpha\beta j} r_{\alpha\gamma j} + r_{\alpha\beta k} r_{\alpha\gamma k})}{\partial r_{\lambda k}} = \frac{\partial(\sum_{l=1}^3 r_{\alpha\beta l} r_{\alpha\gamma l})}{\partial r_{\lambda k}} \\ &= \frac{\partial[\sum_{l=1}^3 (r_{\beta l} - r_{\alpha l})(r_{\gamma l} - r_{\alpha l})]}{\partial r_{\lambda k}} = r_{\alpha\beta l}(\delta_{\alpha\lambda} - \delta_{\gamma\lambda}) + (\delta_{\alpha\lambda} - \delta_{\beta\lambda}) r_{\alpha\gamma l}. \end{aligned} \tag{C.17}$$

The term $\frac{\partial(r_{\alpha\beta} r_{\alpha\gamma})}{\partial r_{\lambda k}}$ in Eq. (C.16) becomes

$$\begin{aligned} \frac{\partial(r_{\alpha\beta} r_{\alpha\gamma})}{\partial r_{\lambda k}} &= \frac{\partial\left(\sqrt{\sum_{l=1}^3 (r_{\alpha l} - r_{\beta l})^2} \sqrt{\sum_{l=1}^3 (r_{\alpha l} - r_{\gamma l})^2}\right)}{\partial r_{\lambda k}} \\ &= r_{\alpha\beta} \frac{r_{\alpha\gamma l}(\delta_{\alpha\lambda} - \delta_{\gamma\lambda})}{r_{\alpha\gamma}} + \frac{r_{\alpha\beta l}(\delta_{\alpha\lambda} - \delta_{\beta\lambda})}{r_{\alpha\beta}} r_{\alpha\gamma} \end{aligned} \tag{C.18}$$

Therefore, substituting equations (C.17) and (C.18) in Eq. (C.16), yields

$$\frac{\partial \cos(\theta_{\alpha\beta\gamma})}{\partial r_{\lambda k}} = \frac{r_{\alpha\beta l}(\delta_{\alpha\lambda} - \delta_{\gamma\lambda}) + (\delta_{\alpha\lambda} - \delta_{\beta\lambda}) r_{\alpha\gamma l}}{r_{\alpha\beta} r_{\alpha\gamma}} - \cos(\theta_{\alpha\beta\gamma}) \left(\frac{r_{\alpha\gamma l}(\delta_{\alpha\lambda} - \delta_{\gamma\lambda})}{r_{\alpha\gamma}^2} + \frac{r_{\alpha\beta l}(\delta_{\alpha\lambda} - \delta_{\beta\lambda})}{r_{\alpha\beta}^2} \right). \tag{C.19}$$

Therefore, considering the cases when (i) $\lambda = \alpha$, $\lambda \neq \beta$ and $\lambda \neq \gamma$

$$\frac{\partial \cos(\theta_{\alpha\beta\gamma})}{\partial r_{\alpha k}} = \frac{r_{\alpha\beta l} + r_{\alpha\gamma l}}{r_{\alpha\beta} r_{\alpha\gamma}} - \cos(\theta_{\alpha\beta\gamma}) \left(\frac{r_{\alpha\gamma l}}{r_{\alpha\gamma}^2} + \frac{r_{\alpha\beta l}}{r_{\alpha\beta}^2} \right), \tag{C.20}$$

(ii) $\lambda = \beta$, $\lambda \neq \alpha$ and $\lambda \neq \gamma$

$$\frac{\partial \cos(\theta_{\alpha\beta\gamma})}{\partial r_{\beta k}} = -\frac{r_{\alpha\gamma l}}{r_{\alpha\beta} r_{\alpha\gamma}} + \cos(\theta_{\alpha\beta\gamma}) \frac{r_{\alpha\beta l}}{r_{\alpha\beta}^2} \tag{C.21}$$

and (iii) $\lambda = \gamma$, $\lambda \neq \alpha$ and $\lambda \neq \beta$

$$\frac{\partial \cos(\theta_{\alpha\beta\gamma})}{\partial r_{\gamma k}} = -\frac{r_{\alpha\beta l}}{r_{\alpha\beta} r_{\alpha\gamma}} + \cos(\theta_{\alpha\beta\gamma}) \frac{r_{\alpha\gamma l}}{r_{\alpha\gamma}^2} \tag{C.22}$$