Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 218 (2006) 836–859
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Domain reduction method for atomistic simulations

Sergey N. Medyanik, Eduard G. Karpov, Wing Kam Liu *

Department of Mechanical Engineering, Northwestern University, 2145 Sheridan Road, Evanston, IL 60208-3111, USA

Received 21 January 2005; received in revised form 14 February 2006; accepted 11 March 2006
Available online 25 April 2006

Abstract

In this paper, a quasi-static formulation of the method of multi-scale boundary conditions (MSBCs) is derived and applied to atomistic simulations of carbon nano-structures, namely single graphene sheets and multi-layered graphite. This domain reduction method allows for the simulation of deformable boundaries in periodic atomic lattice structures, reduces the effective size of the computational domain, and consequently decreases the cost of computations. The size of the reduced domain is determined by the value of the domain reduction parameter. This parameter is related to the distance between the boundary of the reduced domain, where MSBCs are applied, and the boundary of the full domain, where the standard displacement boundary conditions are prescribed. Two types of multi-scale boundary conditions are derived: one for simulating in-layer multi-scale boundaries in a single graphene sheet and the other for simulating inter-layer multi-scale boundaries in multi-layered graphite. The method is tested on benchmark nano-indentation problems and the results are consistent with the full domain solutions.

Keywords: Multi-scale modeling; Atomistic simulation; Domain reduction; Deformable boundary; Nano-indentation; Graphite; Carbon nano-structures

1. Introduction

Computational methods for atomistic simulations, such as molecular mechanics and molecular dynamics (MD), find application in many areas of research by providing information about processes happening at the micro- and nano-scales. However, applicability and effectiveness of these methods often depend on the ability to fulfill large scale computations; and despite the availability of high-performance computers, these methods are still restricted to solving systems that are too small for even nano-scale problems. Solvable systems are several orders of magnitude smaller than the scales of the real-life nano-mechanical experiments. This situation is resolved by using the so-called multi-scale, or coupled atomistic–continuum, methods. In this case, atomistic simulations are used only for a reduced domain (fine scale region) where resolution down to the atomic scale is necessary.

* Corresponding author. Tel.: +1 847 491 7094; fax: +1 847 491 3915.
E-mail addresses: medyanik@northwestern.edu (S.N. Medyanik), edkarpov@gmail.com (E.G. Karpov), w-liu@northwestern.edu (W.K. Liu).

0021-9991/$ - see front matter
doi:10.1016/j.jcp.2006.03.008

Depending on the technique used to model the effect of the exterior coarse scale region surrounding the fine scale atomistic domain, there are two major types of coupled atomistic–continuum methods: (1) methods explicitly modeling the exterior region using continuum approach, typically a finite-element method, and (2) methods using continuum or lattice Green's functions to eliminate degrees of freedom associated with the exterior region.

Among the most noticeable work on the methods of the first type is that by Kohlhoff and co-workers [1], who initiated this approach by developing finite-element/atomistic (FEAt) method, quasicontinuum method developed by Tadmor et al. [2], and the work by Shilkrot et al. [3] who introduced a discrete dislocation technique into the continuum along with a mechanism for passing the dislocations through the atomistic–continuum interface (CADD method). Recent advances in this area include the bridging scale method developed by Wagner and Liu [4]. The method is based on the bridging scale decomposition in which finite element approximation co-exists with atomistic description in the fine scale region. Qian et al. [5] used the bridging scale method to develop a virtual atom cluster (VAC) model for quasi-static simulations of carbon nano-tubes. However, despite the overall success of this type of multi-scale methods, there exist a number of complications due to the presence of two different scales and the necessity to model them simultaneously. Thus, the coarse scale region has to be explicitly modeled in the simulations and a special treatment is needed to ensure a smooth coupling at the interface between the continuum and atomistic domains.

A Green's function approach makes it possible to eliminate completely degrees of freedom associated with the coarse scale region and to model by direct simulations only the fine scale domain. Methods of this type are often called the flexible (or deformable) boundary methods. A number of such methods using both continuum and lattice Green's functions have been developed. Thus, Sinclair et al. [6] introduced a flexible boundary scheme for two-dimensional domains and applied it to atomistic simulations of straight dislocations in 2D. Rao et al. [7] have extended this technique to three dimensions and used it to study dislocations in 3D. The main purpose of this method is to relieve the constraints imposed at the rigid boundaries of the finite atomistic domains appearing in the simulations of defects, such as crack tips and dislocations. In these studies, lattice Green's functions were used for atoms that were close to the source of the force, while continuum Green's functions were used for atoms far from the force source. One of the disadvantages of this type of method is that the nature of the defect has to be known beforehand in order to determine the appropriate continuum Green's functions. In addition, the solution procedure used in these studies in order to bring the system to a state close to equilibrium required several repeated steps involving the system relaxation followed by correction of atomic positions based on the Green's functions.

A lattice Green's function approach for lattice statics calculations has been known for a long time (see, e.g. [8]). The method was revisited and further developed by Thomson et al. [9] and then applied to studies of dislocation nucleation and crack stability by Zhou et al. [10]. In general, these methods allowed for a significant domain reduction. However, they were typically designed for infinite lattice structures or structures with periodic boundary conditions. Furthermore, these methods were often specifically tailored for the treatment of lattices with particular point defects. A general multi-scale theory using lattice Green's function approach for finite domains is still in quest. A comprehensive review of the multi-scale methods can be found in recent papers by Curtin and Miller [11] and Liu et al. [12]. A review of the multi-scale methods with a special emphasis on the simulations of defects and dislocations was given by Moriarty et al. [13].

In this work, a multi-scale method for static analyses of finite periodic lattice structures is presented. The method uses a lattice Green's function approach to model the response of the exterior coarse scale region by imposing the so-called multi-scale boundary conditions (MSBCs) at the boundaries of the reduced atomistic domain (fine scale region). The main distinctive feature of the current method, as compared to other methods utilizing the lattice Green's function concept, is that it has an intrinsic multi-scale character. It is designed for modeling large, though *finite*, domains, where two different length scales are involved. The method accounts for both local and peripheral effects. For instance, displacements at the outer boundary of the coarse scale domain can be prescribed. This was not the case in the earlier studies, where the Green's functions were usually found for infinite structures or those with periodic boundary conditions. The current method is concerned with the accurate modeling of finite domains of a given size that can be efficiently divided into coarse and fine scale regions. The size of the coarse scale region is characterized by the special *domain reduction parameter*, which is a key feature of the current method.

The present method was inspired by the MD boundary condition approach by Wagner et al. [14] and Karpov et al. [15], who applied it to molecular dynamics simulations of model problems using simplified interatomic forces. However, there are substantial differences in the mathematical formulation as well as physical interpretation of this method for dynamic and static cases. For instance, the dynamic formulation implies that the full domain is infinite, which results in a dissipation (one-way transfer) of energy through the multi-scale boundary of the reduced domain. The multi-scale boundary conditions in the dynamic formulation are called impedance or non-reflecting boundary conditions. When the energy input into the fine scale region is needed, coupling with continuum simulations of the surrounding coarse scale region is necessary. As a result, the dynamic formulation of the method was predominantly used as an integral part of a more general bridging scale atomistic–continuum coupled method [16–18], where the non-reflecting boundary conditions are used to ensure a smooth transition between the two scales taking care of the wave reflections at the boundary between the atomistic and continuum domains.

The method's formulation is based on Fourier analysis of finite periodic lattice structures. In order to apply the discrete Fourier transform, the whole domain is discretized into identical unit cells that consist of one or several atoms. The choice of a representative unit cell depends on the translational symmetry of the periodic lattice as well as the type of interatomic forces. The static formulation (as opposed to dynamic) requires an introduction of a new parameter (we call it a domain reduction parameter), which is equal to the number of unit cells between the boundary of the fine scale region and the boundary of full domain. The domain reduction parameter determines the size of the cut-off coarse scale region, and therefore defines the size of the full domain. The MSBCs are derived in terms of displacements for the layer of unit cells at the interface between the fine and coarse scale regions. MSBCs applied at the boundary of the reduced domain lead to an actual solution of the problem for the given finite full domain with no need for direct simulations of the coarse scale part. It is only required that known displacement boundary conditions are prescribed at the outer boundary of the full domain and that there are no external forces applied inside the coarse scale region.

In this paper, we test and validate the method applying it to three-dimensional atomistic simulations of a realistic material, namely graphite, that has a complex periodic structure and interatomic forces. A three-body Tersoff–Brenner interatomic potential [19,20] was used for modeling strong covalent bonds between the atoms in the same graphene layer, while weak van der Waals forces between the layers were modeled with a pair-wise Lennard–Jones potential. These two types of interatomic forces led to two different types of the MSBCs: one for simulating the in-layer multi-scale boundaries in a single graphene sheet and the other for simulating the inter-layer multi-scale boundaries in multi-layered graphite. The issue of intersection of the multi-scale boundaries is addressed in this paper on a systematic basis. The accuracy of the method depends on the size of the cut-off region and reduces with increased domain reduction. Error due to the method is estimated for all the employed benchmark problem tests.

The paper is organized as follows. Section 2 contains a description of the mathematical theory behind the method of multi-scale boundary conditions. In Section 3, we apply the method to a single graphene sheet; and we test it on a model indentation problem in Section 4. In Section 5, the method is applied to a multi-layered graphite structure; and in Section 6, the results of simulations for the indentation of multi-layered graphite are presented. Comments on the computational savings due to the method are given in Section 7. Finally, discussion of the results and conclusions are provided in Section 8.

2. Method of multi-scale boundary conditions

In this section we describe the mathematical foundation of the method of multi-scale boundary conditions for the quasi-static case. The method is based on Fourier analysis of regular atomic lattice structures. Previously, a discrete Fourier transform was used by Karpov et al. [21] for the static analysis of repetitive engineering structures, for instance, beam-like and plate-like trusses. Recently, the dynamic formulation of this method was presented and applied to molecular dynamics simulations of infinite repetitive crystal structures [14,15]. In the current work, we present the method for the quasi-static case and describe it below in application to a simple two-dimensional lattice deforming in a three-dimensional space. The method can be easily extended to a more complicated case of a three-dimensional lattice, and this will be shown in Sections 5 and 6.

2.1. Problem statement and a representative unit cell approach

We consider a two-dimensional periodic atomic structure. A simple example of such structure is given in Fig. 1. Here, the whole domain can be reproduced by a translation of a representative unit cell consisting of one atom along the two orthogonal directions n and m, and the cells can be labeled as (n,m) along these directions. In general, the *representative unit cell* may consist of more than one atom, and the choice of the unit cell depends on the translational symmetry of the crystal lattice and the type of interatomic forces. A more detailed discussion on the choice of the unit cell is given in Section 3, where the method is applied to a single graphene layer.

For simplicity, in this section we assume that each atom of the structure shown in Fig. 1 is directly interacting with its four nearest neighbors only. If the atomic displacements are small enough, we can assume a linear dependence of forces on the displacements. In this case, the internal force acting on the cell (n,m) can be written as

$$\mathbf{f}^{\text{int}}_{n,\text{m}} = \sum_{n'=n-1}^{n+1} \sum_{m'=m-1}^{m+1} \mathbf{K}_{n-n',m-m'} \mathbf{u}_{n',m'}, \tag{1}$$

where components of stiffness matrices $\mathbf{K}_{n,m}$ are either defined explicitly or can be derived from the interatomic potential as

$$\mathbf{K}_{n-n',m-m'} = -\left.\frac{\partial^2 V(\mathbf{u})}{\partial \mathbf{u}_{n,\text{m}} \partial \mathbf{u}_{n',m'}}\right|_{\mathbf{u}=0}. \tag{2}$$

Here it is convenient to introduce a notion of an *associate substructure*. The associate substructure of a given lattice structure includes a given representative unit cell along with all the neighbor cells that have atoms directly interacting with the atoms of the given cell. The form of the associate substructure depends on the geometry of the lattice structure, interatomic forces, and the choice of a representative unit cell. For our example shown in Fig. 1 the associate substructure consists of five single-atom cells. For instance, the associate substructure of cell (n,m) also includes cells $(n,m\pm1)$, $(n\pm1,m)$. The number of cells in the associate substructure defines the number of non-zero matrices $\mathbf{K}$ in Eq. (1).

We divide the whole domain into two major parts. The first one is a domain of interest, where deformation can be so large that the relationship between the atomic forces and displacements is highly non-linear and even the regularity of the lattice structure may break down. We refer to this part of the domain as a *fine scale region* (layers $n \leqslant 0$ in Fig. 1). The second part of the domain consists of a regular atomic structure, and the regularity is preserved throughout the deformation process. Moreover, the deformation in this region is assumed to be small enough so that the dependence between forces and displacements can be approximated by the

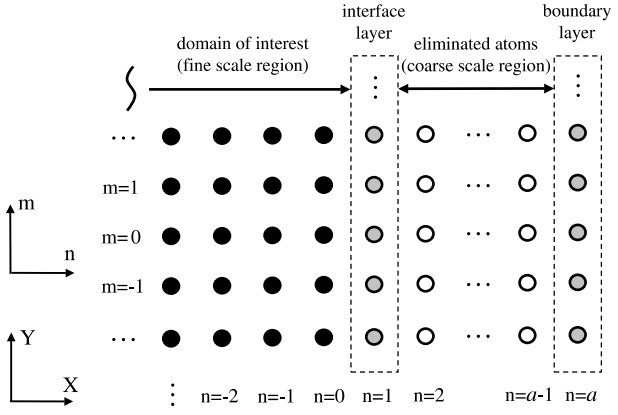

Fig. 1. Periodic two-dimensional atomic lattice structure consisting of a number of infinite atomic layers ($n \leqslant a$). Degrees of freedom associated with the coarse scale part of the domain (layers $1 < n < a$) are eliminated from the simulations. The reduced domain consists of the fine scale region (layers $n \leqslant 0$) and the interface layer $n = 1$.

linear relationship (1). We refer to this part of the domain as a *coarse scale region* (layers $1 < n < a$ in Fig. 1). We also require that the displacements are prescribed at the peripheral boundary of the full domain (layer $n = a$ in Fig. 1), and the external forces can be applied only inside the fine scale region. The number of cell layers between the boundary of the fine scale region and the peripheral boundary of the full domain we call a *domain reduction parameter* and denote as a^{DRP}. Thus, in Fig. 1 the label of the boundary layer is $a = a^{\mathrm{DRP}}$. The value of this parameter defines the size of the coarse scale region, which we want to eliminate from the simulations.

Our final goal is to eliminate all the degrees of freedom associated with the coarse scale part of the full domain while retaining its effect on the fine scale region. This can be achieved by applying special boundary conditions to the layer of atoms $n = 1$, which serves as an interface between the two scales. We called these boundary conditions the multi-scale boundary conditions (MSBCs). As it will be shown in the following sections, such boundary conditions can be derived for the displacements of the interface layer $n = 1$ in terms of the displacements of the layers $n = 0$ (the first layer of the fine scale region next to the interface layer) and $n = a$ (the peripheral boundary of the full domain), where $a = a^{\mathrm{DRP}}$. Thus, due to the applied MSBCs the initially large computational domain will shrink to the size of the reduced domain, consisting of the fine scale region and the interface layer.

2.2. Discrete fourier transform

To achieve the goal stated above, we are going to use the method of discrete Fourier transform (DFT) and apply it to the regular atomic lattice structure, which is discretized using a representative unit cell approach as described above. The DFT of a discrete function f_n is defined as

$$\hat{f}(p) = \mathscr{F}_{n\to p}\{f_n\} = \sum_n f_n \mathrm{e}^{-\mathrm{i}pn}. \tag{3}$$

We are using the traditional hatted notations for the transform functions. Also, in our derivations we use the following important DFT property, which is called the convolution theorem:

$$\mathscr{F}\left\{\sum_{n'} f_{n-n'} g_{n'}\right\} = \hat{f}(p)\hat{g}(p). \tag{4}$$

2.3. Derivation of the method

The equilibrium condition for a current cell (n,m), involving the linearized internal force (1), gives:

$$\sum_{n'=n-1}^{n+1} \sum_{m'=m-1}^{m+1} \mathbf{K}_{n-n',m-m'} \mathbf{u}_{n',m'} + \mathbf{f}^{\mathrm{ext}}_{n,\mathrm{m}} = \mathbf{0}. \tag{5}$$

Application of the two DFTs, $\mathscr{F}_{n\to p}$ and $\mathscr{F}_{m\to q}$, to the above equation and the use of the convolution theorem (4) give:

$$\hat{\mathbf{K}}(p,q)\hat{\mathbf{u}}(p,q) + \hat{\mathbf{f}}^{\mathrm{ext}}(p,q) = \mathbf{0}, \tag{6}$$

from which we can find the transform displacements

$$\hat{\mathbf{u}}(p,q) = -\hat{\mathbf{K}}^{-1}(p,q)\hat{\mathbf{f}}^{\mathrm{ext}}(p,q). \tag{7}$$

Here, it is convenient to introduce a new matrix

$$\hat{\mathbf{G}}(p,q) = -\hat{\mathbf{K}}^{-1}(p,q), \tag{8}$$

which has a meaning of the lattice Green's function in the Fourier domain [21].

Now we consider the part of the domain consisting of rows $n = 0,\ldots,a$ (here $a = a^{\mathrm{DRP}}$) separately and assume that deformation of this subdomain occurs due to some external forces applied at its boundary rows, $n = 0$ and $n = a$. The external force acting on the atomic rows $n = 0,\ldots,a$ will have the form:

$$\mathbf{f}^{\text{ext}}_{n,\text{m}} = \delta_{n,0}\mathbf{f}_{0,\text{m}} + \delta_{n,\text{a}}\mathbf{f}_{a,\text{m}}. \tag{9}$$

Now substituting the above expression for the external force into (7) and applying the inverse ($p \to n$) DFT for both sides, we find for $n = 0, 1, \ldots, a$:

$$\hat{\mathbf{u}}_n(q) = \sum_{n'=0}^{a} \hat{\mathbf{G}}_{n-n'}(q)\hat{\mathbf{f}}^{\text{ext}}_{n'}(q) = \hat{\mathbf{G}}_n(q)\hat{\mathbf{f}}_0(q) + \hat{\mathbf{G}}_{n-a}(q)\hat{\mathbf{f}}_a(q). \tag{10}$$

For $n = 0$, $n = 1$, and $n = a$ we have:

$$\begin{aligned}
\hat{\mathbf{u}}_0(q) &= \hat{\mathbf{G}}_0(q)\hat{\mathbf{f}}_0(q) + \hat{\mathbf{G}}_{-a}(q)\hat{\mathbf{f}}_a(q),\\
\hat{\mathbf{u}}_1(q) &= \hat{\mathbf{G}}_1(q)\hat{\mathbf{f}}_0(q) + \hat{\mathbf{G}}_{1-a}(q)\hat{\mathbf{f}}_a(q),\\
\hat{\mathbf{u}}_a(q) &= \hat{\mathbf{G}}_a(q)\hat{\mathbf{f}}_0(q) + \hat{\mathbf{G}}_0(q)\hat{\mathbf{f}}_a(q).
\end{aligned} \tag{11}$$

Now using the first and the third equations of (11) we write the unknown force vectors $\mathbf{f}_0$ and $\mathbf{f}_a$ in terms of $\mathbf{u}_0$ and $\mathbf{u}_a$:

$$\begin{pmatrix} \hat{\mathbf{f}}_0(q) \\ \hat{\mathbf{f}}_a(q) \end{pmatrix} = \begin{pmatrix} \hat{\mathbf{G}}_0(q) & \hat{\mathbf{G}}_{-a}(q) \\ \hat{\mathbf{G}}_a(q) & \hat{\mathbf{G}}_0(q) \end{pmatrix}^{-1} \begin{pmatrix} \hat{\mathbf{u}}_0(q) \\ \hat{\mathbf{u}}_a(q) \end{pmatrix} \tag{12}$$

and substituting them into the second equation of (11) we get

$$\hat{\mathbf{u}}_1(q) = \begin{pmatrix} \hat{\mathbf{G}}_1(q) & \hat{\mathbf{G}}_{1-a}(q) \end{pmatrix} \begin{pmatrix} \hat{\mathbf{G}}_0(q) & \hat{\mathbf{G}}_{-a}(q) \\ \hat{\mathbf{G}}_a(q) & \hat{\mathbf{G}}_0(q) \end{pmatrix}^{-1} \begin{pmatrix} \hat{\mathbf{u}}_0(q) \\ \hat{\mathbf{u}}_a(q) \end{pmatrix}. \tag{13}$$

Finally, introducing a new pair of matrix functions $\boldsymbol{\Theta}$ and $\boldsymbol{\Xi}$ as

$$\begin{pmatrix} \hat{\boldsymbol{\Theta}}(q) & \hat{\boldsymbol{\Xi}}(q) \end{pmatrix} \equiv \begin{pmatrix} \hat{\mathbf{G}}_1(q) & \hat{\mathbf{G}}_{1-a}(q) \end{pmatrix} \begin{pmatrix} \hat{\mathbf{G}}_0(q) & \hat{\mathbf{G}}_{-a}(q) \\ \hat{\mathbf{G}}_a(q) & \hat{\mathbf{G}}_0(q) \end{pmatrix}^{-1}, \tag{14}$$

we obtain

$$\hat{\mathbf{u}}_1(q) = \hat{\boldsymbol{\Theta}}(q)\hat{\mathbf{u}}_0(q) + \hat{\boldsymbol{\Xi}}(q)\hat{\mathbf{u}}_a(q). \tag{15}$$

Thus, we have expressed the displacements of row $n = 1$ in terms of the displacements of rows $n = 0$ and $n = a$ for the transformed domain. Now, applying the inverse ($q \to m$) DFT to Eq. (15), we find the final form of our multi-scale boundary conditions:

$$\mathbf{u}_{1,\text{m}} = \sum_{m'} (\boldsymbol{\Theta}_{m-m'}\mathbf{u}_{0,\text{m}'} + \boldsymbol{\Xi}_{m-m'}\mathbf{u}_{a,\text{m}'}). \tag{16}$$

Here, the kernel matrices $\boldsymbol{\Theta}$ and $\boldsymbol{\Xi}$ have a 3×3 size (in the case of a 3D deformation) and their values are dimensionless since Eq. (16) relates displacements to displacements. In general, the unit cell may consist of more than one atom. In this case the size of the kernel matrices will be proportional to the number of atoms per cell.

Strictly, the convolution summation in (16) should be done over a number of subscripts m': $m - M_\text{c} \leqslant m' \leqslant m + M_\text{c}$, where $M_\text{c} \to \infty$. Meanwhile, in practical applications we have to use a finite, preferably small, cut-off parameter $m_\text{c} = M_\text{c}$. Thus, effectiveness of the method of multi-scale boundary conditions hinges upon the fast decay of matrices $\boldsymbol{\Theta}_m$ and $\boldsymbol{\Xi}_m$ with the growth of the absolute value of index m.

In many applications the boundary of the coarse scale domain ($n = a$) is fixed, ($u_a = 0$). In this case the second term in Eq. (16) vanishes; and, using the summation truncation discussed above, we can rewrite (16) as:

$$\mathbf{u}_{1,\text{m}} = \sum_{m'=m-m_\text{c}}^{m+m_\text{c}} \boldsymbol{\Theta}_{m-m'}\mathbf{u}_{0,\text{m}'}. \tag{17}$$

Fig. 2 illustrates the case when the peripheral boundary is fixed, and the cut-off parameter $m_\text{c} = 1$ is used as an example. In this case, by Eq. (17) the displacements of a single atom cell $(1, 0)$ of row $n = 1$ can be written as

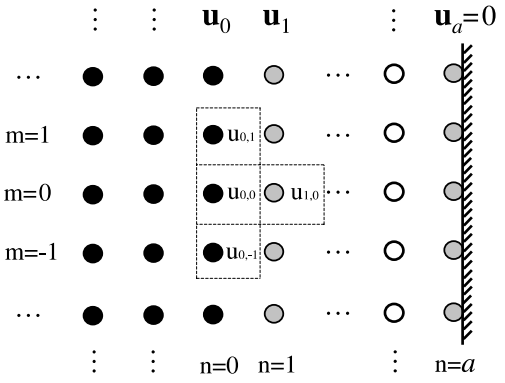

Fig. 2. Degrees of freedom associated with the coarse scale region (atomic layers $1 < n < a$, $a = a^{\mathrm{DRP}}$) are eliminated from the simulations. Reduced domain consists of the fine scale region (layers $n \leqslant 0$) and the interface layer $n = 1$. Displacements of the interface layer u_1 are determined only by the displacements u_0 of the layer $n = 0$, if the peripheral boundary $n = a$ is fixed.

$$\mathbf{u}_{1,0} = \mathbf{\Theta}_1 \mathbf{u}_{0,-1} + \mathbf{\Theta}_0 \mathbf{u}_{0,0} + \mathbf{\Theta}_{-1} \mathbf{u}_{0,1}. \tag{18}$$

In the following section the method is applied to the atomic lattice of a single graphene sheet, and then it is used for molecular mechanics simulations of a nano-indentation problem.

3. Application to a single graphene sheet

3.1. A representative unit cell approach for a single graphene sheet

In graphite, carbon atoms form strong covalent bonds with three neighboring atoms in the same plane, thus forming a honeycomb structure as shown in Fig. 3. There are two major shapes of edges in a graphene layer – zigzag and armchair. For instance, the two vertical edges of the rectangular atomic structure explicitly shown in Fig. 3 (identified as layers $n = -1$ and $n = a$) are zigzag, while the two horizontal edges are armchair. The MSBCs for a graphene layer can be applied along the edges of both types.

To employ the method for a specific lattice structure, first we have to divide the whole domain into unit cells so that every atom belongs to one and only one cell and then find a way of labeling those cells throughout the domain. The choice of a representative unit cell and the labeling nomenclature depends on the geometry of the

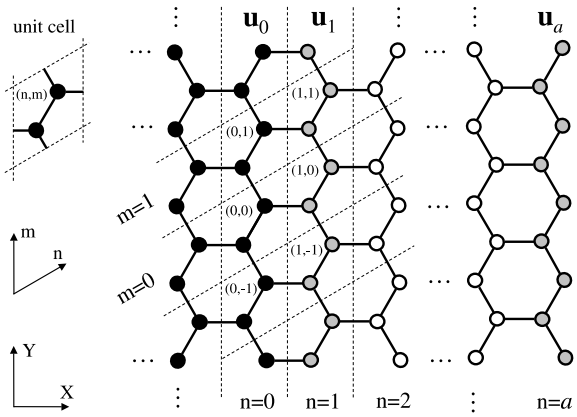

Fig. 3. Method applied to a single graphene layer. Degrees of freedom associated with the coarse scale region (atomic layers $1 < n < a$, $a = a^{\mathrm{DRP}}$) are eliminated from the simulations. Reduced domain consists of the fine scale region (layers $n \leqslant 0$) and the interface layer $n = 1$. Displacements of the interface layer u_1 are determined by the displacements u_0 and u_a of the layer $n = 0$ and the peripheral boundary $n = a$.

repetitive lattice structure and is not unique for a given lattice, but there are reasons that elucidate the best choice.

First of all, using a translational symmetry of the lattice structure, we should be able to reproduce the whole domain by translation of one unit cell in two different (not necessarily orthogonal) directions for a two-dimensional lattice and in three directions for a three-dimensional lattice. Second, the directions of translation and, thus, labeling the cells should be aligned with the boundary of the domain where we want to apply our multi-scale boundary conditions. Third, atoms in each cell should be directly interacting through interatomic forces only with the atoms of the current cell and the immediately neighboring cells. And, finally, for elegance and convenience of implementation we want the representative unit cell to be as small as possible.

Taking into account the above recommendations, for a single graphene sheet with the MSBCs applied along a zigzag face, we chose the unit cell consisting of two atoms and the numbering along directions n and m as shown in Fig. 3. Note that there are no principal differences in applying the method to the armchair boundary, but the unit cell would then have to be extended to four atoms and translation would be done along two orthogonal directions. In the current work we are using the method with a zigzag type of boundary only. As it will be seen from the nano-indentation example in the following section, applying the MSBCs to six zigzag boundaries of a hexagonal shaped domain allows for the whole area of localized deformation to be conveniently surrounded by a multi-scale boundary.

In the molecular mechanics simulations, the interatomic interaction forces are derived from a classical potential. For the interactions between carbon atoms in the same layer we use a Tersoff–Brenner potential [19,20] with a set of parameters denoted as Potential II in [20]. Due to a three-body component of this potential, every atom interacts not only with its three nearest, but also with six secondary neighbors. Based on this, we can now construct an associate substructure. As it was defined in Section 2.1, the associate substructure consists of a given unit cell and all the neighbor cells that have atoms directly interacting with at least one atom of the given unit cell. Thus, the associate substructure for a single graphene layer shown in Fig. 3 consists of a total of seven unit cells. For instance, the associate substructure of cell (n,m) also includes cells $(n,m\pm1)$, $(n\pm1,m)$, $(n+1,m-1)$ and $(n-1,m+1)$. The form of the associate substructure defines the number of stiffness matrices $\mathbf{K}$ that relate the linearized forces to displacements by Eq. (1). The $\mathbf{K}$ matrices are then used to compute the kernel matrices $\mathbf{\Theta}$ that are the key element of the present method.

3.2. $\mathbf{\Theta}$ matrices

Following the procedure described in Section 2, matrices $\mathbf{\Theta}_m$ were computed. The left plot in Fig. 4 shows a typical dependence of the matrices on parameter m, demonstrating a fast decay of the matrix components with the growth of the absolute value of m. The largest component of the kernel matrices is $\Theta_0(6,3)$. The reason for this becomes obvious upon inspection of the lattice structure of the graphene layer (Fig. 3). Indeed, according to Eq. (17) this component relates the vertical (normal to the sheet's plane) displacements of the two closest

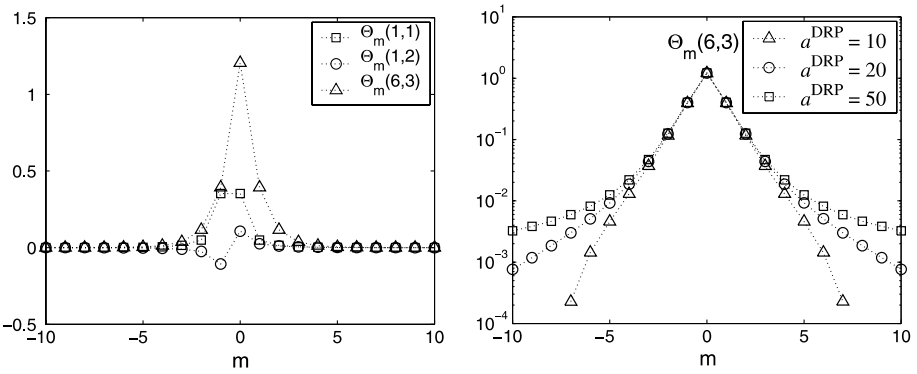

Fig. 4. Values of components of kernel matrices $\mathbf{\Theta}_m$ vs. parameter m. Left: typical shapes of several components, domain reduction parameter $a^{\mathrm{DRP}}=10$. Right: logarithmic plot of one component vs. parameter m for $a^{\mathrm{DRP}}=10$, 20, 50.

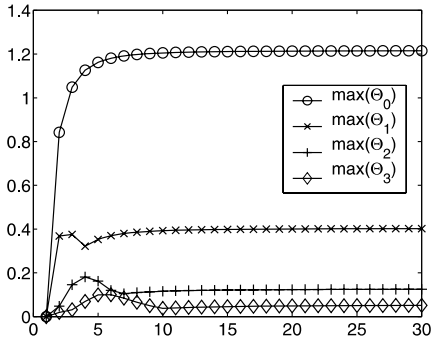

Fig. 5. Maximum values of matrices $\mathbf{\Theta}_m$ vs. domain reduction parameter a^{DRP}.

atoms of rows $n = 1$ and $n = 0$. For instance, if we use the unit cell labeling as shown in Fig. 3 and calculate displacements of a current cell $(1,0)$ of the row $n = 1$ from the displacements of row $n = 0$, then the component $\Theta_0(6,3)$ is relating the vertical displacement of the lower atom of cell $(1,0)$ to the vertical displacement of the upper atom of cell $(0,0)$.

Fast decay of the matrix components with the growth of the absolute value of m allows using a small value of the truncation parameter m_c in Eq. (17), which considerably reduces the computational cost of the method. A dependence of this decay on the domain reduction parameter can be seen from a logarithmic plot of $\Theta_0(6,3)$ for the three values $a^{\mathrm{DRP}} = 10$, 20, 50, which is shown in the right plot of Fig. 4.

Fig. 5 shows the values of the maximum components of matrices $\mathbf{\Theta}_m$ ($m = 0$, 1, 2, 3) vs. domain reduction parameter a^{DRP}. It can be seen that with the growth of a^{DRP} the maximum components for each of the four matrices asymptotically approach some constant value. On the other hand, as can be seen from the logarithmic plot (Fig. 4), the decay of the matrices $\mathbf{\Theta}_m$ with the growth of the absolute value of m becomes less steep with increased a^{DRP}. This seems quite natural, since with the growth of the domain reduction parameter the fixed boundary of the full domain (row $n = a$, $a = a^{\mathrm{DRP}}$) is moving further from the multi-scale boundary of the reduced domain (row $n = 1$), and its effect becomes more non-local. Thus, using the multi-scale boundary conditions with larger values of the domain reduction parameter requires using a larger value of the cut-off parameter m_c in (17). However, due to the fast decay of the kernel matrices for graphite, we found that using a value of $m_c = 4$ is sufficient, and we used this value in our simulations.

3.3. Treatment of the multi-scale boundaries intersections

In practical applications of the method of MSBCs one will have to deal with intersections of the multi-scale boundaries. Two linear multi-scale boundaries can intersect in a two-dimensional lattice structure, creating an angle in the overall boundary of the reduced domain. In the case of a three-dimensional lattice, two or three plane multi-scale boundaries can intersect, thus creating an edge. We discuss the issues concerning the intersections of the multi-scale boundaries considering the angle formed by the two zigzag faces of the graphene layer, see Fig. 6. For convenience, we introduce a local cell numbering as shown in the figure. We will refer to the two sides that form the angle as the right and the bottom edges. The atoms of the cells labeled $(1,m)$, $m \geqslant -1$ belong to the right edge of the reduced domain, and the atoms of the cells $(n,-1)$, $n \leqslant 1$ belong to the bottom edge. The cell $(1,-1)$, corresponding to a point where the two edges meet, belongs to both of them. Covalent bonds between the atoms of the reduced domain are shown in the figure with solid lines.

The fixed boundary of the full domain (its atoms shown in the figure are connected with dotted lines) has an angle formed by the intersection of the two layers, $n = a$ and $m = -a$, which represent, correspondingly, the right and the bottom faces of the full domain. The cell $(a,-a)$ is at the vertex of the angle. Here we assume that the MSBCs applied at the two edges of the reduced domain correspond to the same value of a domain reduction parameter and $a = a^{\mathrm{DRP}}$. In general, this is not necessary, and different values of the domain reduction parameter can be used for each individual part of the intersecting multi-scale boundaries.

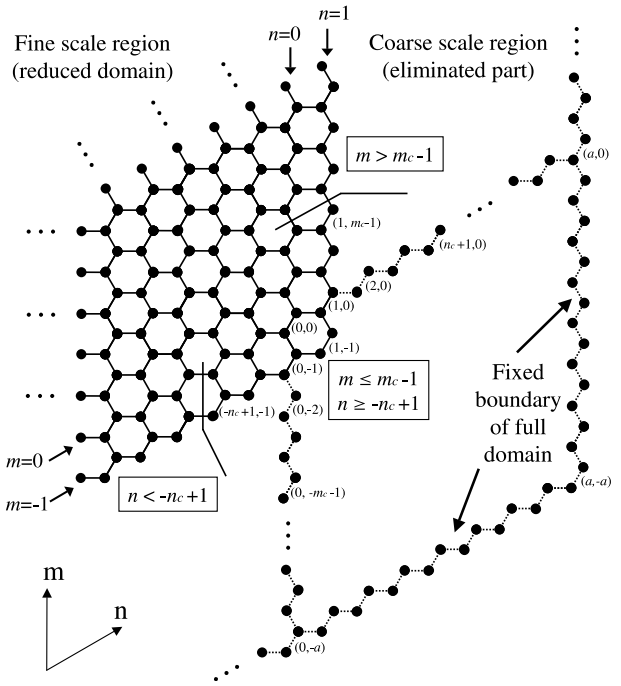

Fig. 6. Intersection of the two multi-scale boundaries in a graphene layer. The fixed boundary of the full domain has an angle formed by the intersection of the two layers of unit cells, $n = a$ and $m = -a$. The multi-scale boundary of the reduced domain has an angle formed by the intersection of layers $n = 1$ and $m = -1$.

There are two major issues regarding the intersection of the multi-scale boundaries. The first one results from the fact that when we treat each of the two edges individually, we have two sets of Eq. (17), one for the right edge of the reduced domain and the other for the bottom edge; and the cells that are close to the intersection should enter both sets of equations.

Similar to (17), the MSBCs can be written for the right edge as

$$\mathbf{u}_{1,\mathrm{m}} = \sum_{m'=m-m_\mathrm{c}}^{m+m_\mathrm{c}} \mathbf{\Theta}^{\mathrm{R}}_{m-m'} \mathbf{u}_{0,\mathrm{m}'}, \tag{19}$$

and for the bottom edge as

$$\mathbf{u}_{n,-1} = \sum_{n'=n-n_\mathrm{c}}^{n+n_\mathrm{c}} \mathbf{\Theta}^{\mathrm{B}}_{n-n'} \mathbf{u}_{n',0}. \tag{20}$$

Here, Eq. (19) looks exactly like (17) as we derived it in Section 2.3, but Eq. (20) is written in a different form. This is because Eq. (17) was derived for a special case when the interface and peripheral boundary layers were labeled $n = 1$ and $n = a$ correspondingly, whereas in general the same can be derived using other layers (both along n and m directions) as the interface and the full domain boundary. Thus, Eq. (20) is used for the MSBCs applied along the interface layer $m = -1$ and relates the displacements of its atoms to those of the layer $m = 0$ when the layer $m = -a$ is a fixed boundary of the full domain. To distinguish the kernel matrices in the above equations, we denoted them as $\mathbf{\Theta}^{\mathrm{R}}$ and $\mathbf{\Theta}^{\mathrm{B}}$. The kernel matrices $\mathbf{\Theta}^{\mathrm{B}}$ can be derived in a similar manner by repeating steps (9)–(16) of Section 2.3.

As can be seen from Fig. 6, in order to compute atomic displacements for the cells $(1, m)$, $-1 \leqslant m \leqslant m_\mathrm{c} - 1$ using Eq. (19), we need to know the displacements for the cell $(0, -1)$, which in its turn lies on the multi-scale boundary and has to be found from Eq. (20). Similarly, the cell $(1, 0)$ has to be determined from (19) in order to be used in (20). Thus, the two sets of Eqs. (19) and (20), namely (19) for $-1 \leqslant m \leqslant m_\mathrm{c} - 1$ and (20) for $-n_\mathrm{c} + 1 \leqslant n \leqslant 1$, are coupled and have to be solved simultaneously. The area affected by the intersection

depends on the values of the cut-off parameters n_c, m_c and includes the cells $(1,m)$, $-1 \leqslant m \leqslant m_c - 1$ at the right edge and cells $(n,-1)$, $-n_c + 1 \leqslant n \leqslant 1$ at the bottom edge of the reduced domain.

Here we note that use of the two different types of kernel matrices is needed only when treating the intersections of the multi-scale boundaries. For the areas of the multi-scale boundaries not affected by the intersection we can use the same set of kernel matrices, say $\mathbf{\Theta}^R$, by rotating the frame of reference and using the local cell labeling consistent with the one which the matrices were derived for.

The second issue regarding the intersection of the multi-scale boundaries is caused by unavailability of a group of atoms in the reduced domain which are required to compute the atomic displacements for the cells at the multi-scale boundary. It turns out that the atoms needed for one multi-scale boundary are cut-off by the other. For instance, to compute displacements for the cell $(1,0)$ at the right edge using the value of truncation parameter $m_c = 3$, we need to know the displacements of the seven cells from $(0,-3)$ to $(0,3)$, where the cells $(0,-3)$ and $(0,-2)$ are not present in our atomistic simulations of the reduced domain but exist in the full domain being modeled. It can be seen that this issue is relevant for the cells up to $(1,m_c - 1)$ on the right edge and down to $(-n_c + 1,-1)$ on the bottom edge of the reduced domain.

Neglecting this issue is equivalent to assuming zero displacements for those "non-existing" atoms and results in a non-physical fixation of the angles (or the edges) of the reduced domain where the multi-scale boundaries intersect. In order to resolve this issue, one needs to extrapolate the displacements field beyond the boundaries of the reduced domain by introducing the "ghost" atoms that belong to the missing cells. These atoms are shown in the figure as a continuation of the reduced domain and are connected by the dashed lines. There are several different ways to define the values for the displacements of these new unit cells. In our calculations we simply use a constant extrapolation, taking the displacements of the "ghost" cells equal to those of the last existing cell of the same row that lies on the multi-scale boundary of the reduced domain. For instance, we assume the displacements of the atoms in the cells from $(2,0)$ to $(n_c + 1,0)$ equal to those of the cell $(1,0)$ and the displacements of the cells $(0,-2)$ to $(0,-m_c - 1)$ equal to those of the cell $(0,-1)$.

4. Example: nano-indentation of a single graphene sheet

4.1. Problem statement

A spherical indenter of radius R is applied in a displacement control manner to the center of a graphene sheet of hexagonal shape in a direction perpendicular to the plane of the sheet (see Fig. 7). On each step, the indenter is lowered by 0.5 Å, and then energy is minimized using the conjugate gradient method.

The indenter is modeled using a repulsive potential. On every atom located in the vicinity of the center of the indenter closer than the indenter radius acts a repulsive force:

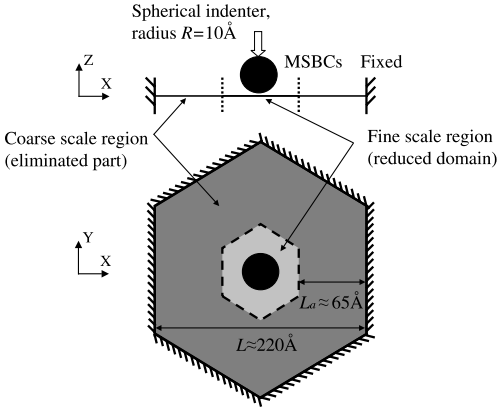

Fig. 7. Schematic representation of the problem statement for nano-indentation of a single graphene sheet. The characteristic dimension of the coarse scale region being eliminated from simulations, L_a, is proportional to the domain reduction parameter. The indenter radius R is 10 Å.

$$F_i = A(R - r_i)^2, \tag{21}$$

where R is the radius of the indenter, A is a parameter governing its stiffness, and r_i is a distance between the centers of the indenter and current atom. In our simulations we used the value of $A = 100$ eV/Å^3, corresponding to a very stiff (nearly rigid) indenter, and the radius of the indenter was $R = 10$ Å.

The outer boundary of the full domain (represented by a solid line in the figure) is fixed. Multi-scale boundary conditions are applied at the boundary of the reduced domain, which is shown with a dashed line in Fig. 7. The size of the reduced domain is determined by the domain reduction parameter, a^{DRP}, which represents the number of unit cell layers between the boundaries of the reduced and full domains. The characteristic dimension of the coarse scale region being eliminated from the simulations, L_a, is equal to the domain reduction parameter times the size of the representative unit cell. The MSBCs are applied to each of the six zigzag sides of the reduced domain. The six angles (intersections of the zigzag sides) are treated using the method explained earlier in Section 3.3.

4.2. Results of simulations

Molecular mechanics simulations were carried out for reduced domains of different sizes, using MSBCs with different values of the domain reduction parameter, and the results were compared to the corresponding full domain solutions. For the multi-scale simulations presented in this section we have used a cut-off parameter value $m_c = 4$. Here, we demonstrate our results on one example when the size of the reduced domain is 2646 atoms, and the value of the domain reduction parameter is $a^{\mathrm{DRP}} = 30$, which corresponds to the full domain size of 15,000 atoms.

Fig. 8 shows the reduced domain along with the fixed boundary of the full domain at four stages of the indentation process starting from the initial configuration, when all the atoms were in the same plane, and then after the indenter penetrates to the depths of 10, 20, and 30 Å. The pictures in Fig. 8 have true aspect ratio, i.e., they use the same scale in all dimensions.

The deformed configuration of the reduced domain at the final stage (after indenting to the depth of 30 Å) can be seen in more detail in Fig. 9. Initially, the graphene sheet was positioned in the plane $Z = 0$. It can be seen that due to the applied MSBCs the whole boundary of the reduced domain has moved from its initial position ($Z = 0$) downwards in the direction of indentation, representing a deformable boundary.

Fig. 10 shows a comparison of the vertical displacements of the reduced domain to the full domain solution for the atoms at the cross-section of the sheet by the plane $Y = 0$. This comparison proves to be very close. Note that Figs. 9 and 10 show the actual scale of deformation, proving the method works well even for quite large deformations. The largest error in displacement is found at the edges of the reduced domain where multi-scale boundary conditions are applied. The maximum error here is estimated at 4.9%. The error reduces

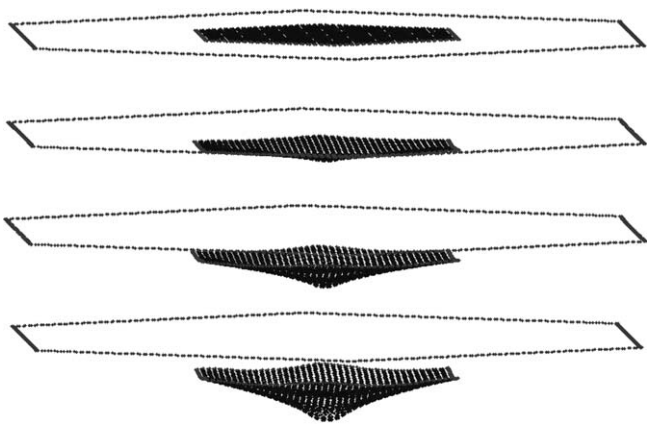

Fig. 8. Multi-scale simulations of the indentation of a graphene layer; reduced domain is shown along with the fixed boundary of the full domain. Four steps: depth of indentation = 0, 10, 20, 30 Å.

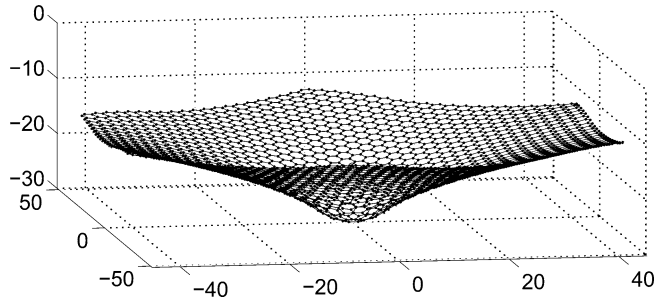

Fig. 9. Deformed configuration of the reduced domain after indenting to the depth of 30 Å. Size of the reduced domain is 2646 atoms, size of the corresponding full domain is 15,000 atoms. Domain reduction parameter $a^{\mathrm{DRP}} = 30$.

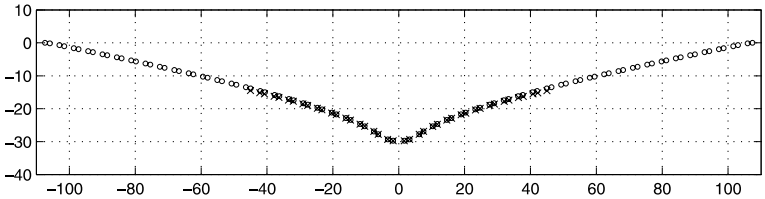

Fig. 10. Vertical displacements of the atoms lying on the middle line cross-section: reduced domain multi-scale simulations (crosses) vs. full domain solution (circles). Domain reduction parameter $a^{\mathrm{DRP}} = 30$.

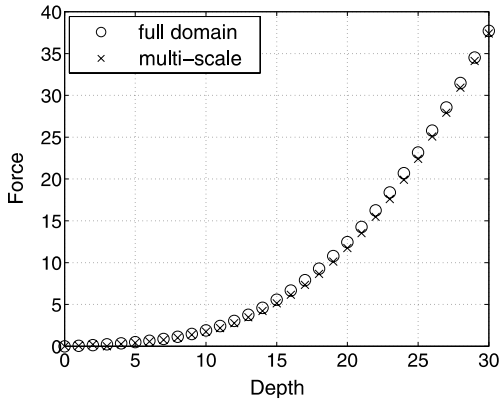

Fig. 11. Force (eV/Å) vs. indentation depth (Å).

steadily towards the center of the domain and is almost zero at the center, right under the indenter. Mean error over the cross-section of the reduced domain shown in Fig. 10 is 1.36%. Fig. 11 shows very good agreement between multi-scale and full domain results for the force vs. depth of indentation curve. The force here is given in units of eV/Å, and 1 eV/Å $\approx$ 1.602 nN. The maximum deviation in force between the full and reduced domain solutions is at the indentation depth of 23 Å and is equal to 0.785 eV/Å, which corresponds to the percentage error of 4.27%.

5. Application to multi-layered graphite

5.1. Multi-layered graphite and inter-layer forces

Multi-layered graphite consists of carbon atom layers (basal planes) shifted with respect to each other and stacked one above another. There are two major forms of multi-layered graphite depending on the stacking

sequence of the single graphene layers: hexagonal and rhombohedral (e.g., see [22]). In hexagonal graphite, which is also known as Bernal graphite, every second layer is identical, i.e., located exactly one above another (ABAB stacking), whereas in rhombohedral graphite only every third layer is identical (ABCABC stacking). Since the latter type of graphite is thermodynamically unstable, the hexagonal form is prevailing under natural conditions. Thus, we chose hexagonal type of graphite for our consideration. Fig. 12 shows the structure of the multi-layered graphite. In every plane, one half of the atoms are located directly above the atoms of the adjacent plane, whereas the other half face the empty centers of the hexagons of the neighboring layers.

For simulation of inter-layer forces we are using Lennard–Jones potential:

$$E = 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right], \tag{22}$$

where r is a distance between two atoms located in different basal planes, and parameters have the following values: $\epsilon = 0.00188$ eV, $\sigma = 3.3264$ Å [23]. With these parameters, the spacing between the layers in equilibrium, L_{AB}, is about 3.348 Å which agrees well with the values observed experimentally as well as with results of ab initio simulations [23,24]. For more information on interatomic potentials used for modeling carbon nano-structures see, e.g., a review by Qian et al. [25].

5.2. Modeling inter-layer multi-scale boundaries

Application of the method of MSBCs to multi-layered graphite involves two types of multi-scale boundaries. For the lateral boundaries of the reduced domain in each basal plane, we can apply MSBCs that were derived from the in-layer interactions based on Brenner potential exactly in the same manner as we did it for a single sheet of graphite in Section 3. Another type of MSBCs should be used for reducing the vertical (in-depth) dimension of the lattice structure. This new type of MSBCs should be applied to the whole bottom basal plane of the reduced domain, and it should be derived based on the inter-layer interactions.

Again, as in the beginning of Section 3, we have to choose a representative unit cell and a way of numbering the cells in different directions. These are dictated by the periodicity and repetitiveness of the crystal structure. We can see that in the vertical direction, the smallest period of the lattice is two single layers of graphite that constitute an AB pair. For the two horizontal in-plane directions, we can use the same numbering as we used for a single graphene layer in Section 3. Thus, we chose a representative unit cell consisting of four atoms, two of them lying in layer A and two in layer B.

Fig. 13 shows six layers of graphite that form three unit cell layers $l = -1, 0, 1$. The four atoms of a representative unit cell are shown as large solid dots. Since we have to use a cut-off for the Lennard–Jones potential, the number of neighbors for every atom (and consequently the size of the associate substructure) will depend on the cut-off radius R_c. In our simulations we used value of $R_c = 6.0$ Å. When using this value for the cut-off, each atom interacts directly only with the atoms of the two immediately adjacent layers (upper and lower). The total number of atoms interacting with at least one of the four atoms of the given unit cell is 116 (all solid dots in Fig. 13), 29 atoms per each of the four layers.

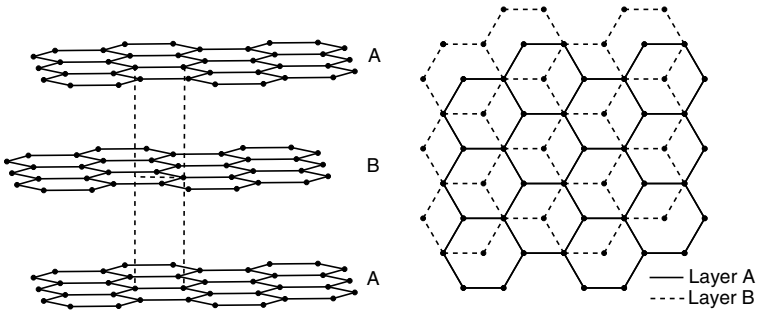

Fig. 12. Atomic structure of hexagonal graphite.

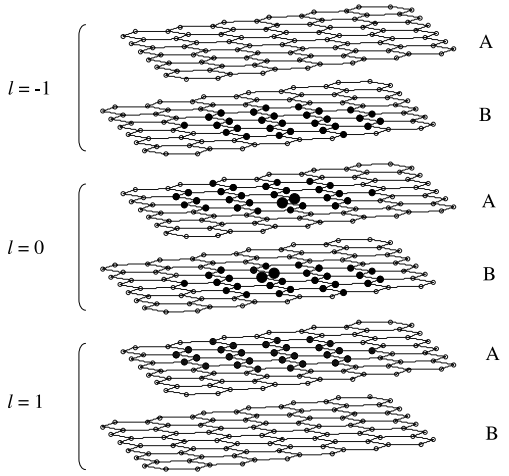

Fig. 13. A representative unit cell and vertical cell labeling for multi-layered graphite. Interlayer distance is $L_{AB} = 3.348$ Å. Unit cell consists of 4 atoms (big solid dots) located in the two adjacent graphene layers. When using cut-off $R_c = 6.0$ Å, atoms of each layer interact only with the atoms of the immediately adjacent layers. The total number of atoms interacting with the atoms of the given unit cell through Lennard–Jones potential is 116 (all solid dots), 29 atoms per layer.

5.3. $\boldsymbol{\Theta}$ matrices for the inter-layer MSBCs

The procedure for deriving kernel matrices $\boldsymbol{\Theta}$ for the inter-layer MSBCs is similar to that used in Section 2 for derivation of the in-layer MSBCs. The only difference is that now the multi-scale boundary is not a line but a plane. In this case two spatial parameters are used with the matrices, and Eq. (17) of Section 2 is transformed into:

$$\mathbf{u}_{n,\mathrm{m},1} = \sum_{n'=n-n_c}^{n+n_c} \sum_{m'=m-m_c}^{m+m_c} \boldsymbol{\Theta}_{n-n',m-m'} \mathbf{u}_{n',m',0}. \tag{23}$$

The size of the matrices $\boldsymbol{\Theta}_{n,m}$ is 12×12 since the unit cell now consists of four atoms. Fig. 14 shows a typical behavior of the matrices $\boldsymbol{\Theta}_{n,m}$. Here, the values of the two largest components of the matrices are given vs. spatial parameter n at a fixed parameter m. Similar behavior of the components is observed vs. parameter m at fixed n. Again, as in the case of in-layer multi-scale boundaries (Section 3), fast decay of the matrix components with the growth of the absolute values of the spatial parameters n and m allows using small values for the truncation parameters n_c and m_c in Eq. (23), thus, resulting in a considerable reduction of the computational cost of the method.

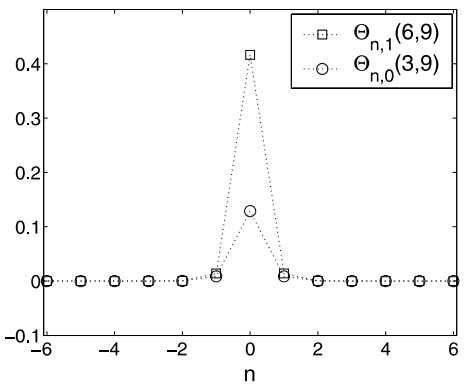

Fig. 14. Components of matrices $\boldsymbol{\Theta}_{n,m}$ vs. parameter n. Domain reduction parameter $a^{DRP} = 3$.

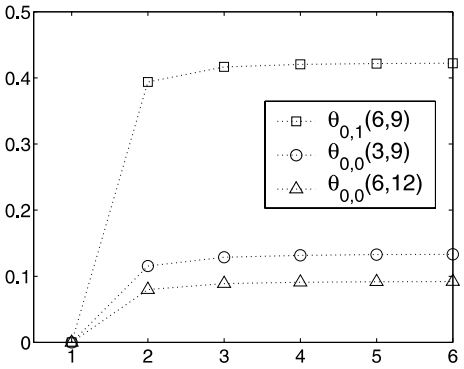

Fig. 15. Maximum components of matrices $\mathbf{\Theta}_{n,m}$ vs. domain reduction parameter.

Fig. 15 shows the values of the three maximum components of matrices $\mathbf{\Theta}_{n,m}$ vs. the domain reduction parameter a^{DRP}. Similar to the case of an in-layer multi-scale boundary, the components of the matrices approach some asymptotic values with the growth of a^{DRP}, and all the comments regarding this property made in Section 3.2 hold here as well.

6. Example: nano-indentation of multi-layered graphite

6.1. Problem statement

The problem statement is similar to that of Section 4 (indentation of a single graphene sheet). The only difference is that now we have L layers stacked in a vertical direction. Schematic representation of the problem statement is given in Fig. 16. A spherical indenter of radius R is applied to the center of the top layer in a step-like manner. On each step, the indenter is lowered by 0.5 Å and then the energy is minimized using the conjugate gradient method. The bottom and lateral boundaries of the full domain are fixed. The boundary of the reduced domain, where multi-scale boundary conditions are applied, is indicated in the figure with a dashed line. Now we have two domain reduction parameters: $a^{\mathrm{DRP}}_{\mathrm{lateral}}$ for the in-layer and $a^{\mathrm{DRP}}_{\mathrm{bottom}}$ for the inter-layer

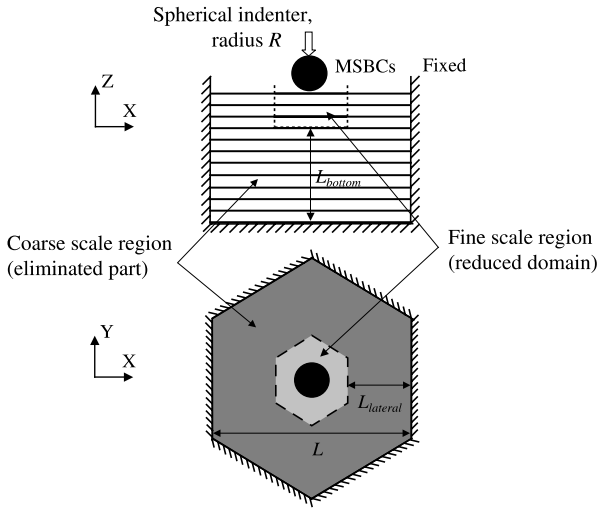

Fig. 16. Schematic representation of the problem statement for nano-indentation of multi-layered graphite. The characteristic dimensions of the coarse scale region being eliminated from simulations in the horizontal and vertical directions, L_{lateral} and L_{bottom}, respectively, are proportional to the domain reduction parameters used correspondingly for the lateral and bottom MSBCs. Values of the indenter radius R used in the simulations are 20 and 40 Å.

multi-scale boundaries. The characteristic dimensions of the coarse scale region eliminated from the simulations in the horizontal and vertical directions, respectively, are L_{lateral} and L_{bottom}. These values are proportional to the domain reduction parameters $a^{\text{DRP}}_{\text{lateral}}$ and $a^{\text{DRP}}_{\text{bottom}}$, respectively, and are equal to the value of the domain reduction parameter times the size of the representative unit cell in that direction. Values of the indenter radius R used in these simulations are 20 and 40 Å.

6.2. Results of simulations

Testing the method performance in a consistent way can be done in three steps, each time comparing the results to the full domain atomistic solutions. In the first example, we apply multi-scale boundary conditions only to the lateral boundaries of every individual layer of the reduced domain in the same fashion as it was done in Section 4 for a single sheet of graphite. The bottom layer in this case is free. In the second example, we use only the inter-layer MSBCs which were derived in the previous section, and we apply them to the bottom of the reduced domain. This allows for the reduction of the computational domain by a given number of layers, depending on the value of the domain reduction parameter $a^{\text{DRP}}_{\text{bottom}}$, whereas each layer of the reduced domain is fully simulated. Finally, in the last examples, both in-layer (lateral) and inter-layer (bottom) MSBCs are employed, reducing the domain in both horizontal and vertical directions. For all the simulations we used the cut-off parameters $m_c = 4$ for the in-layer and $n_c = m_c = 4$ for the inter-layer multi-scale boundaries, except for the results where these values are specified explicitly.

6.2.1. Lateral multi-scale boundary

Simulations of the reduced domain consisting of three graphite layers were performed with in-layer MSBCs applied to the lateral boundaries of each layer using the value of the domain reduction parameter $a^{\text{DRP}}_{\text{lateral}} = 10$. The size of the reduced domain is 7938 atoms, and the size of the full domain that corresponds to $a^{\text{DRP}}_{\text{lateral}} = 10$ is 16,200 atoms. Thus, the effective size of the computational domain is reduced more than two times.

Here we note that for modeling the lateral multi-scale boundaries in multi-layered graphite, we are employing the in-layer MSBCs in the same form as they are presented for a single sheet of graphite in Section 3, thus modeling only in-layer interactions. This methodology is not fully rigorous since it does not account for the inter-layer interactions beyond the lateral boundaries of the reduced domain. The admission of this simplification is based on the assumption that inter-layer forces are crucial only right under the indenter and in the close vicinity around it, and they are less important at a larger distance from the indentation area. This assumption can be justified by the close proximity of the results to the full domain solutions in the current and following sections.

Fig. 17 shows positions of the atoms at the cross-section of the domain by the XZ-plane ($Y = 0$). The cross-section also goes through the center of the spherical indenter. It can be seen that after indenting to the depth of 15 Å, the atomic positions of the reduced domain are very close to those of the full domain even at considerably large displacements. Also, a good agreement of the multi-scale and full domain solutions is observed for the relationship between the force acting on the indenter and the depth of indentation which is shown in Fig. 18. The maximum error in displacement in each layer is at the multi-scale boundaries of the reduced

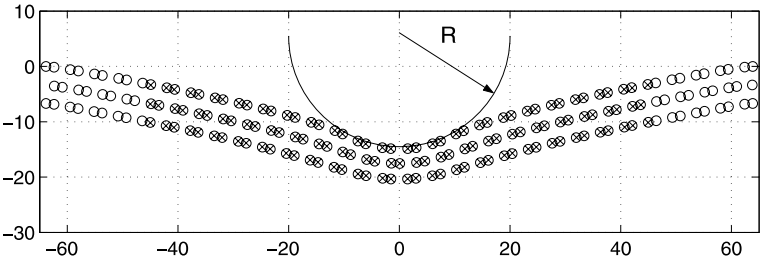

Fig. 17. Middle cross-section of the graphite layers under the indenter: reduced domain simulation (crosses) vs. full domain solution (circles). $R = 20$ Å. Domain reduction parameter $a^{\text{DRP}}_{\text{lateral}} = 10$. Depth of indentation is 15 Å.

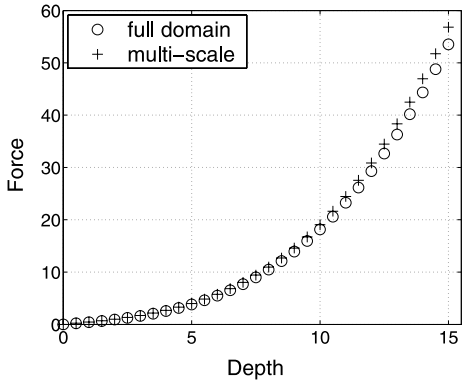

Fig. 18. Force (eV/Å) vs. indentation depth (Å).

domain, similarly to the case of a single graphene layer test. Among all the layers, the maximum error is in the bottom one, and it is estimated at 3%. Mean error over the whole reduced domain cross-section shown in Fig. 17 is around 0.9%. The maximum deviation in force is at the last step, indentation depth of 15 Å, and is equal to 3.29 eV/Å, which corresponds to the percentage error of 6.15%.

6.2.2. Bottom multi-scale boundary

In these simulations we use only the inter-layer MSBCs, derived in Section 5, applying them to the bottom of the reduced domain. Due to such a multi-scale boundary, the number of layers used for actual simulations (present in the reduced domain) is decreased, while each layer of the reduced domain has the same number of atoms as the layers in the original full domain. An insight into implementation of the method in the case of the inter-layer multi-scale boundaries is provided in Fig. 19 which shows the numbered double layers of the unit cells as they were defined in Section 5 and used here for the calculation of the MSBCs.

According to the theoretical derivations of Section 5, displacements of layer $l = 1$ should be calculated from displacements of layer $l = 0$ using Eq. (23) provided that layer $l = a^{\mathrm{DRP}}_{\mathrm{bottom}}$ is fixed. For the current problem, we used only the top single graphene layers of the $l = 1$ and $l = a^{\mathrm{DRP}}_{\mathrm{bottom}}$ double-layers in the actual multi-scale and full domain calculations, respectively, as shown in Fig. 19. This became possible because of a relatively small cut-off radius for the Lennard–Jones potential used for modeling the inter-layer interactions, resulting in the very small (if any) forces between the layers that are not immediate neighbors. The sizes of the reduced and full domains for this and other problems, where the inter-layer MSBCs are used, are given for the domains, that were abridged in such a way.

For this problem we used the value of the domain reduction parameter $a^{\mathrm{DRP}}_{\mathrm{bottom}} = 4$. The size of the reduced domain is 5 layers and total 13,230 atoms, and the size of the full domain corresponding to the given value of the domain reduction parameter is 11 layers and 29,106 atoms, making the effective size of the computational domain reduced more than two times. Fig. 20 shows the force–displacement curve for this problem and reveals that multi-scale and full domain results are close up until the point when breakage occurs. The breakage of the graphite in both simulations occurred at the same step, corresponding to the indentation depth of 11.5 Å and force of about 500 eV/Å.

The maximum error in displacement in each of the graphene layers of the reduced domain is in the areas half-way between the center of the domain where the indenter is applied and the domain boundaries where fixed boundary conditions are enforced. The maximum error here ranges from under 3% in the top layer to nearly 10% in the bottom layer. Mean error for the reduced domain cross-section shown in Fig. 19 is estimated at approximately 5%. Relatively high error in this example may be explained by the fact that it is estimated for a configuration at the very last step before fracture. This final stage is associated with a strong non-linear deformation. The maximum deviation in force is found at the indentation depth of 10 Å and is estimated at 15.1 eV/Å, which corresponds to 5.46% error.

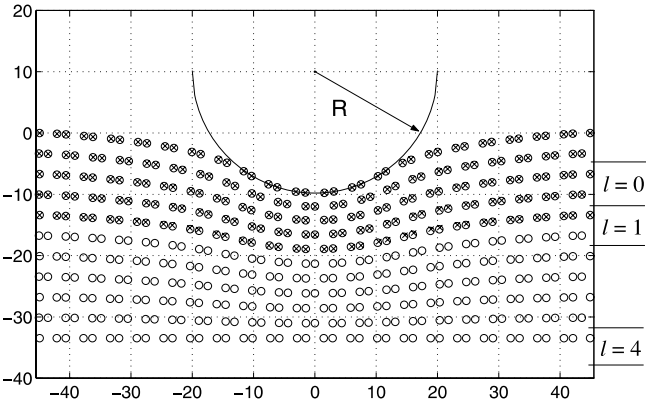

Fig. 19. Middle cross-section of the graphite layers under the indenter: reduced domain simulation (crosses) vs. full domain solution (circles). $R = 20$ Å. Domain reduction parameter $a^{\mathrm{DRP}}_{\mathrm{bottom}} = 4$. Depth of indentation is 10 Å.

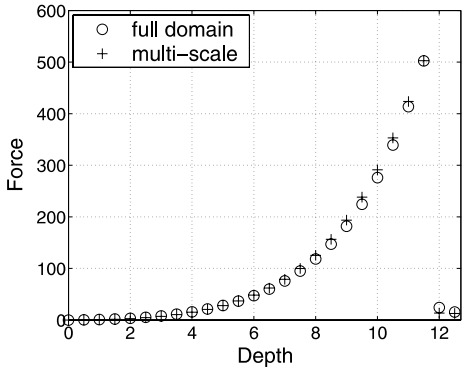

Fig. 20. Force (eV/Å) vs. indentation depth (Å).

6.2.3. Combined bottom and lateral multi-scale boundaries

Finally, we employ both in-layer and inter-layer MSBCs, reducing the domain in all three directions. We used a reduced domain consisting of 7 graphene layers with a total size of 18,522 atoms. Values of the domain reduction parameters were 10 for the in-layer (lateral) and 4 for the inter-layer (bottom) multi-scale boundaries. The corresponding full domain has 13 layers and a total size of 70,200 atoms. The reduction of the effective computational domain due to the applied MSBCs is almost four times. The indenter radius was 40 Å in this simulation.

Fig. 21 shows vertical positions of atoms at the middle cross-section for both reduced and full domains after the indenter penetrated to the depth of 12.5 Å. The results of the multi-scale simulations are in good agreement with the benchmark full domain solution. The maximum error in displacement is found at the edges of the bottom layer of the reduced domain where two multi-scale boundaries (bottom and lateral) of the reduced domain intersect, and the error here is estimated at approximately 12%. However, the MSBCs intersection point represents an extreme case, and the error is considerably lower for the rest of the domain. Thus, the mean error over the reduced domain cross-section shown in Fig. 21 is only about 3%.

Load vs. indentation depth curves are shown in Fig. 22 and reveal a rather close comparison between the multi-scale and full domain solutions. It is particularly remarkable that the breakage of the graphite layers occurred at almost the same depth and load for both types of simulations. The maximum deviation in force is at the last step before the fracture in the reduced domain test (the fracture occurs one step later in the case of full domain calculations). The deviation is approximately 256 eV/Å, which corresponds to the percentage error of 19.2%. The error goes up considerably only at the last several indentation steps. For instance, the error is under 5% until the indentation depth of 10 Å. At the same time, the error in the depth at which

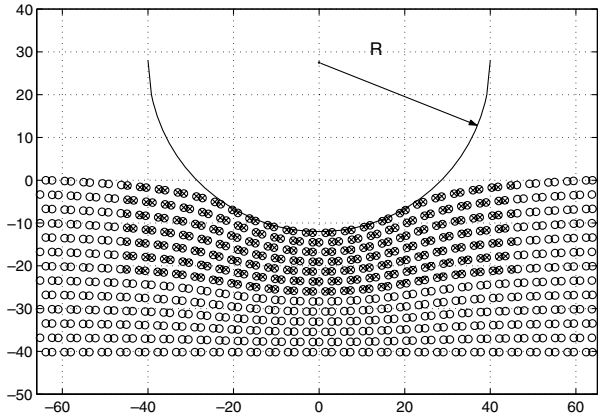

Fig. 21. Middle cross-section of the graphite layers under the indenter: reduced domain simulation (crosses) vs. full domain solution (circles). $R = 40$ Å. Domain reduction parameters: $a^{\mathrm{DRP}}_{\mathrm{lateral}} = 10$, $a^{\mathrm{DRP}}_{\mathrm{bottom}} = 4$. Depth of indentation is 12.5 Å.

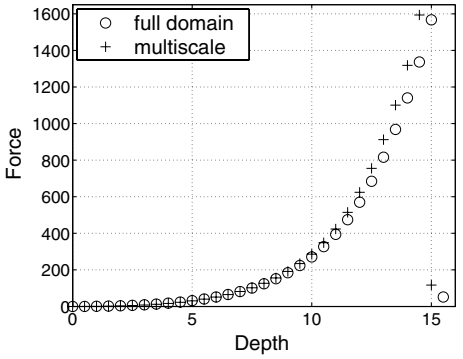

Fig. 22. Force (eV/Å) vs. indentation depth (Å).

the fracture occurs is only one indentation step, or just 3.3%. Again, as in the previous example, the main source of the error lies in considerably high non-linear deformations present at the last stage of the nano-indentation test.

We have also investigated the effect of the multi-scale boundaries when the larger values of the domain reduction parameters are used and show some results of these simulations below. Unfortunately, for these simulations we cannot provide comparisons to the full domain solutions similar to those that were given previously in this section, because the sizes of the corresponding full domains are too large.

For the reduced domain in these simulations we have used the same domain that served as a full domain for the previous example. The domain has 13 layers and a total size of 70,200 atoms. The size of the indenter was the same, $R = 40$ Å. We have used several values of the domain reduction parameters: 6 and 20 for the bottom multi-scale boundaries, 30 and 50 for the lateral. Note, that for the bottom domain reduction parameter, each unit corresponds to two single graphene layers, doubling the actual number of graphene layers virtually added to the reduced domain through the applied MSBCs.

Here we present the results of the two multi-scale simulations, identified as MSBC1 and MSBC2. The domain reduction parameters used for these simulations were $a^{\mathrm{DRP}}_{\mathrm{lateral}} = 30$, $a^{\mathrm{DRP}}_{\mathrm{bottom}} = 6$ for MSBC1 and $a^{\mathrm{DRP}}_{\mathrm{lateral}} = 50$, $a^{\mathrm{DRP}}_{\mathrm{bottom}} = 20$ for MSBC2. The sizes of the corresponding full domains are: 480,378 atoms for MSBC1 and 1,909,746 atoms for MSBC2. Values of all the cut-off parameters used for calculation of the MSBCs (m_{c} for the lateral, and n_{c}, m_{c} for the bottom multi-scale boundaries) were 4 for MSBC1 and 6 for MSBC2. Fig. 23 shows the middle cross-section of the deformed reduced domain for the MSBC2 simulations after indenting to the depth of 30 Å. The initial positions of the top and bottom layers of the reduced domain are shown with the dashed lines in the figure. As we can see, the deformation of the multi-scale (or deformable) boundaries is considerable.

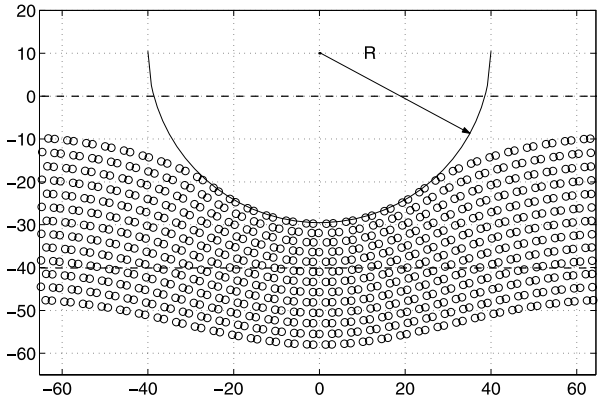

Fig. 23. Middle cross-section of the graphite layers under the indenter for multi-scale simulations example MSBC2. Only reduced atomistic domain is shown. $R = 40$ Å. Domain reduction parameters: $a^{\mathrm{DRP}}_{\mathrm{lateral}} = 50$, $a^{\mathrm{DRP}}_{\mathrm{bottom}} = 20$. Depth of indentation is 30 Å. The dashed lines represent positions of the top and bottom layers of the reduced domain prior to indentation.

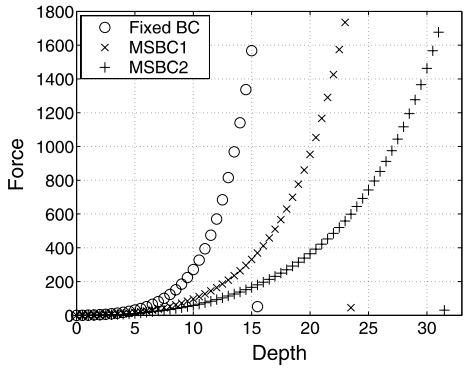

Fig. 24. Force (eV/Å) vs. indentation depth (Å).

The force–depth curves are given in Fig. 24. Here we present results of the two multi-scale simulations, earlier identified as MSBC1 and MSBC2, along with the results when standard fixed boundary conditions were applied to the same domain. Such a comparison helps to see the effect of the applied MSBCs. The behavior of the system with the fixed boundaries is too stiff, which results in a very steep slope of the force–depth curve and the breakage of the graphite layers at a relatively small indentation depth. The applied MSBCs virtually move the fixed boundaries further away from the indentation area, thus reducing the overstiffening of the system due to the fixed boundary conditions. The size of the virtual full domain grows with the growth of the domain reduction parameters; and as a result, the slope of the force–depth curve reduces. The breakage of the graphite layers occurs at approximately the same force for all the three simulations. Here we have to take into account that due to the constant value of the indenter displacement step (0.5 Å) and a very steep growth of the curve corresponding to the fixed boundaries example, the actual value of the force at the break point for this simulation is higher than the one shown at the plot. While the force values at the breakage points are close for all the three simulations, the indentation depths are quite different. Thus, for the MSBC2 example the breakage occurred after indenting to the depth of 31 Å, which is approximately twice that of the simulation when fixed boundary conditions were used.

7. Computational savings due to the method

The most expensive part by far of the current calculations is the evaluation of inter-atomic forces using the three-body Tersoff–Brenner and two-body Lennard–Jones potentials. The number of operations corresponding to this part of calculations for the whole system is roughly proportional to the total number of atoms

(assuming that a finite cut-off for the Lennard–Jones interactions is used), which in its turn is proportional to the *volume* of the computational domain, or area of the computational domain in the case of 2D. For instance, if we assume a cubic domain, the number of atoms can be roughly estimated as N^3, where N is the number of atoms across a single linear dimension.

At the same time, the number of operations required for the evaluation of MSBCs is proportional to the number of atoms lying on the *boundary* of the computational domain, which is on the order of N^2 in the case of 3D and on the order of N in the 2D case. If we consider a nano-indentation problem similar to that shown in Fig. 16, but use a cubic full domain with N atoms across a single linear dimension and assume that the same value of the domain reduction parameter, $a^{\mathrm{DRP}} = a$, is used in all three directions, the number of atoms eliminated from the simulations can be estimated as

$$N^3 - (N-2a)^2(N-a) = 5N^2a - 8Na^2 + 4a^3. \tag{24}$$

The number of atoms lying on the surface of the reduced domain where MSBCs have to be enforced can be estimated as

$$4(N-a)(N-2a) + (N-2a)^2 = 5N^2 - 16Na + 12a^2. \tag{25}$$

As follows from (24) and (25), the use of MSBCs saves $\sim a^3$ computations, while their cost scales only as $\sim a^2$. If we take $a = \alpha N$, where $0 < \alpha < 1/2$, then the number of the eliminated atoms and the number of atoms at the multi-scale boundaries will be correspondingly

$$N^3(5\alpha - 8\alpha^2 + 4\alpha^3) \quad \text{and} \quad N^2(5 - 16\alpha + 12\alpha^2). \tag{26}$$

Thus, if the ratio of a to N is kept constant, the size of the eliminated domain scales as N^3, whereas the size of the boundary of the reduced domain, where MSBCs have to be evaluated, scales as N^2.

The order of the number of operations required for the MSBC calculation for each unit cell at the multi-scale boundary can be estimated based on the formulas (17) and (23) for the 2D and 3D cases correspondingly as

$$(2m_c + 1)n_{\mathrm{dof}}^3 \quad \text{and} \quad (2m_c + 1)(2n_c + 1)n_{\mathrm{dof}}^3. \tag{27}$$

In the above equations, m_c and n_c are the cut-off parameters used in the summations in formulas (17) and (23), and n_{dof} is the number of degrees of freedom per unit cell. According to our estimations, the number of operations per single atom at the multi-scale boundary resulting from (27) is not larger than the number of operations required for calculations of the inter-atomic forces for a single atom.

Based on the above estimations, the total number of operations saved due to the application of the MSBCs is most strongly influenced by the size of the eliminated part of the computational domain. The actual CPU times for the simulations presented in the current work seem to follow this general trend. For instance, the time of the calculations for the example of Section 6.2.1 (lateral multi-scale boundary) using the in-layer MSBCs was approximately half of that for the corresponding full domain atomistic simulations, while the reduction of the computational domain was slightly more than two times.

8. Discussion and conclusions

A quasi-static formulation of the method of multi-scale boundary conditions (MSBCs) has been presented. The method allows simulating deformable boundaries in periodic atomic lattice structures. Due to these deformable boundaries, the simulated reduced domain behaves as if it were a part of a much larger system of a given size (a full domain) with standard displacement boundary conditions applied. The MSBCs reduce the effective size of the computational domain and save the computational cost of the atomistic simulations. The assumption of regularity and periodicity, as well as a linear dependency between forces and displacements, have to hold only in the vicinity of the multi-scale boundary and in the part of the domain (coarse scale region) which is excluded from the computations. At the same time, large deformations, including non-linearity and even lattice defects, may be present inside the simulated reduced domain (fine scale region) provided that they do not have a profound effect on the vicinity where the MSBCs are applied. The accuracy of the method goes down as a function of domain reduction. However, the performed quantitative error analysis

suggests that the method's accuracy is satisfactory up to the final stages of the tests when considerable nonlinear deformations of the multi-scale interface were present. Even in these cases, the method allowed for reasonable predictions, often up until the fracture points.

The method was applied to multi-scale atomistic simulations of nano-indentation of single graphene sheets and multi-layered graphite. However, applications of the method are not limited to these types of carbon nano-structures. The MSBCs derived here, with some modifications, can be applied to simulations of carbon nano-tubes (CNTs). For instance, the in-layer MSBCs similar to those presented in Section 3 for a single graphene layer can be used for multi-scale simulations of single-wall CNTs, and additional inter-layer MSBCs similar to those derived in Section 5 may be applied to multi-wall CNTs.

The method of MSBCs may also be viewed as an advanced type of boundary conditions that can be used for the standard molecular mechanics simulations. Usually, in atomistic simulations two types of boundary conditions are employed: fixed and periodic. For some problems however, such as indentation, neither of these are appropriate. The periodic boundary conditions introduce inadequate periodicity, while fixed boundary conditions lead to an artificial stiffening of the system. One possible important application of the method of MSBCs is the reduction of overstiffening of the atomic system due to fixed boundaries. This issue was addressed in the end of Section 6. Application of the multi-scale boundary conditions helps to artificially move the fixed boundaries further away without the need to simulate a larger domain. This approach can be particularly useful for systems and materials that are able to undergo large elastic deformations (e.g., graphite and CNTs). In this case the effect of the fixed boundaries may extend very far inside the domain. The MSBC approach allows an adequate and straightforward multi-scale modeling of such systems.

Another interesting possibility for the present method consists in the application to crystalline lattice structure, including atomistic simulations of crystalline metallic nano-wires and films; efforts in this direction have already been initiated [26]. Crystal structure applications raise the issue of the passage of lattice dislocations across the multi-scale interface. Such an extension of the present methodology is believed to be straightforward due to the absence of an explicit continuum model over the coarse scale region, and it comprises another exciting challenge for the future.

Acknowledgments

The authors acknowledge the support of the National Science Foundation (NSF), the NSF-IGERT program and the NSF Summer Institute on Nano Mechanics and Materials.

References

[1] S. Kohlhoff, P. Gumbsch, H.F. Fischmeister, Crack propagation in bcc crystals studied with a combined finite-element and atomistic model, Phil. Mag. A 64 (1991) 851–878.
[2] E.B. Tadmor, M. Ortiz, R. Phillips, Quasicontinuum analysis of defects in solids, Phil. Mag. A 73 (1996) 1529.
[3] L.E. Shilkrot, R.E. Miller, W.A. Curtin, Multiscale plasticity modeling: coupled atomistics and discrete dislocation mechanics, J. Mech. Phys. Solids 52 (2004) 755–787.
[4] G.J. Wagner, W.K. Liu, Coupling of atomistic and continuum simulations using a bridging scale decomposition, J. Comput. Phys. 190 (2003) 249–274.
[5] D. Qian, G.J. Wagner, W.K. Liu, A multiscale projection method for the analysis of carbon nanotubes, Comput. Meth. Appl. Mech. Eng. 19 (17–20) (2004) 1603–1632.
[6] J.E. Sinclair, P.C. Gehlen, R.G. Hoagland, J.P. Hirth, Flexible boundary conditions and nonlinear geometric effects in atomic dislocation modeling, J. Appl. Phys. 4 (7) (1978) 3890–3897.
[7] S. Rao, C. Hernandez, J.P. Simmons, T.A. Parthasarathy, C. Woodward, Green's function boundary conditions in two-dimensional and three-dimensional atomistic simulations of dislocations, Phil. Mag. A 77 (1) (1998) 231–256.
[8] V.K. Tewary, Green-function method for lattice statics, Adv. Phys. 2 (6) (1973) 757–810.
[9] R. Thomson, S.J. Zhou, A.E. Carlsson, V.K. Tewary, Lattice imperfection studied by use of lattice Green's functions, Phys. Rev. B 46 (1992) 10613–10622.
[10] S.J. Zhou, A.E. Carlsson, R. Thomson, Dislocation nucleation and crack stability: lattice Green's functions treatment of cracks in a model hexagonal lattice, Phys. Rev. B 47 (1993) 7710–7718.
[11] W.A. Curtin, R.E. Miller, Atomistic/continuum coupling in computational materials science, Model. Simul. Mater. Sci. Eng. 11 (2003) 33–68.
[12] W.K. Liu, E.G. Karpov, S. Zhang, H.S. Park, An introduction to computational nano mechanics and materials, Comput. Meth. Appl. Mech. Eng. 193 (2004) 1529–1578.

[13] J.A. Moriarty, V. Vitek, V.V. Bulatov, S. Yip, Atomistic simulations of dislocations and defects, J. Comp. Aided Mater. Des. 9 (2002) 99–132.
[14] G.J. Wagner, E.G. Karpov, W.K. Liu, Molecular dynamics boundary conditions for regular crystal lattices, Comput. Meth. Appl. Mech. Eng. 19 (17–20) (2004) 1579–1601.
[15] E.G. Karpov, G.J. Wagner, W.K. Liu, A Green's function approach to deriving wave-transmitting boundary conditions in molecular dynamics simulations, Int. J. Numer. Meth. Eng. 6 (9) (2005) 1250–1262.
[16] H.S. Park, E.G. Karpov, W.K. Liu, P.A. Klein, The bridging scale for two-dimensional atomistic/continuum coupling, Phil. Mag. 85 (1) (2005) 79–113.
[17] H.S. Park, E.G. Karpov, P.A. Klein, W.K. Liu, Three-dimensional bridging scale analysis of dynamic fracture, J. Comput. Phys. 207 (2005) 588–609.
[18] H.S. Park, E.G. Karpov, W.K. Liu, Non-reflecting boundary conditions for atomistic, continuum and coupled atomistic/continuum simulations, Int. J. Numer. Meth. Eng. 64 (2005) 237–259.
[19] J. Tersoff, Modeling solid-state chemistry: interatomic potentials for multicomponent systems, Phys. Rev. B 3 (8) (1989) 5566–5568.
[20] D.W. Brenner, Empirical potential for hydrocarbons for use in simulating the chemical vapor-deposition of diamond films, Phys. Rev. B 4 (15) (1990) 9458–9471.
[21] E.G. Karpov, N.G. Stephen, D.L. Dorofeev, On static analysis of finite repetitive structures by discrete Fourier transform, Int. J. Solids Struct. 3 (16) (2002) 4291–4310.
[22] R.W.G. Wyckoff, Crystal Structures, vol. 1, Interscience Publishers, New York, 1963.
[23] Y.C. Wang, K. Scheerschmidt, U. Gosele, Theoretical investigations of bond properties in graphite and graphitic silicon, Phys. Rev. B 61 (19) (2000).
[24] M.C. Schabel, J.L. Martins, Energetics of interplanar binding in graphite, Phys. Rev. B 4 (11) (1992) 7185–7188.
[25] D. Qian, G.J. Wagner, W.K. Liu, M.F. Yu, R.S. Ruoff, Mechanics of carbon nanotubes, Appl. Mech. Rev. 5 (6) (2002) 495–553.
[26] E.G. Karpov, H. Yu, H.S. Park, W.K. Liu, J.Q. Wang, D. Qian, Multiscale Boundary Conditions in Crystalline Solids: Theory and Application to Nanoindentation, International Journal of Solids and Structures, accepted.