Commun. Comput. Phys.
doi: 10.4208/cicp.020912.150713a

Vol. **15**, No. 2, pp. 470-486
February 2014

A Constrained Cauchy-Born Elasticity Accelerated Multigrid Method for Nanoindentation

Jingrun Chen[1,†], Pingbing Ming[2] and Jerry Zhijian Yang[3,*]

[1] *Institute of Computational Mathematics and Scientific/Engineering Computing, AMSS, Chinese Academy of Sciences, Beijing, 100190, P.R. China.*
[2] *LSEC, Institute of Computational Mathematics and Scientific/Engineering Computing, AMSS, Chinese Academy of Sciences, No. 55 East Road Zhong-Guan-Cun, Beijing, 100190, P.R. China.*
[3] *School of Mathematics and Statistics, Wuhan University, Wuhan, 430072, P.R. China.*

Received 2 September 2012; Accepted (in revised version) 15 July 2013

Communicated by Pingwen Zhang

Available online 10 September 2013

Abstract. We introduce a new multigrid method to study the lattice statics model arising from nanoindentation. A constrained Cauchy-Born elasticity model is used as the coarse-grid operator. This method accelerates the relaxation process and considerably reduces the computational cost. In particular, it saves quite a bit when dislocations nucleate and move, as demonstrated by the simulation results.

AMS subject classifications: 65B99, 65N30, 65Z05, 74G15, 74G65, 74S05

Key words: Multigrid method, constrained Cauchy-Born elasticity, nanoindentation, Cauchy-Born rule.

1 Introduction

As a way of probing mechanical properties of materials in small volumes, nanoindentation has attracted great attention in last few decades [11, 21]. It is a flexible characterization technique by varying indentor geometry and indentation direction in experiments. Compared to the uniaxial tension, the strain field under an indentor is more complicated

†*Current address*: South Hall 6705, Mathematics Department, University of California, Santa Barbara, CA93106, USA; *Email address*: cjr@math.ucsb.edu.
*Corresponding author. *Email addresses:* chenjr@lsec.cc.ac.cn (J. Chen), mpb@lsec.cc.ac.cn (P. B. Ming), zjyang.math@whu.edu.cn (J. Z. Yang)

and highly heterogeneous even for a specimen with isotropic materials. Although the timescale issue is crucial in many indentation tests [13], (quasi-)static properties, such as hardness and modulus, are also of great interest for different materials. During nanoindentation, different length scales coexist in the problem due to indenter size, sample size, dislocation nucleation and dislocation propagation. The atomistic-level simulation tools are necessary for nanoindentation. However, the computationally intensive nature of atomistic simulations of these phenomena restricts the simulation cell to a size which is many orders of magnitude smaller than the typical size of the solid in an experiment. Thus, many multiscale approaches have been proposed to exploit the nanoindentation problem. On the other hand, its multiscale nature also supplies a benchmark problem for multiscale methods [4, 5, 12, 17, 23].

From a numerical point of view, multiscale methods can be divided into two categories. One is based on the domain decomposition method [24], which combines lattice statics and continuum model in a concurrent manner [17]. The other is based on the multigrid method [3, 25], where models are combined in a sequential manner [5, 12].

In this paper, we will focus on the latter. Brandt [4] highlighted many possible applications of the multigrid method, including molecular statics. Goedecker *e*t al. [12] employed the linear elasticity as the coarse-grid operator in the two-grid method. The efficiency of the method is verified by the silicon crystals with some point defects. A more general approach was proposed by Chen and Ming [5]. They used the one-way multigrid method [7], which can automatically bypass many unphysical local minimizers. The coarse-grid operator was constructed by the Cauchy-Born (CB) rule [2]. The consistency between the CB elasticity model and lattice statics has been proven under some stability conditions [8]. Efficiency of the method is demonstrated by FCC Aluminium crystals under different homogeneous deformations. Moreover, this method gives qualitatively reasonable results for nanoindentation with quite a bit cost.

To overcome this issue, we introduce a constrained CB elasticity model as the coarse-grid operator to accelerate the convergence of the multigrid method. In nanoindentation, collective motion of atoms in the strain field formally can be separated into two parts: one is induced by loading applied on boundaries and the other is by dislocation structures after nucleation. It has been shown in [5] that the former can be effectively captured by CB elasticity model on coarse grids. It will be shown in the present work that the constrained CB elasticity model can capture the latter in an effective manner. This will be done by updating the local strain field around dislocations at the atomistic scale and then transmitting updated information globally on grids.

In the following, we first briefly review the one-way multigrid method for lattice statics, and then introduce a constrained CB elasticity model, based on which we propose an accelerated multigrid method. We show its efficiency by studying the lattice statics model in nanoindentation. A heuristic explanation of the constrained model will be given in the appendix.

2 Methodologies

2.1 Lattice statics and elasticity model

In lattice statics, the equilibrium configuration $\{y_i\}_{i=1}^N$ is determined by

$$\{y_1,\cdots,y_N\} = \operatorname{argmin} E^{\text{tot}}(y_1,\cdots,y_N), \tag{2.1}$$

where

$$E^{\text{tot}}(y_1,\cdots,y_N) = V(y_1,\cdots,y_N) - \sum_{i=1}^{N} f_i \cdot y_i$$

with V the potential function, and f_i the external force on the i-th atom. The displacement of the i-th atom is then defined as

$$u_i = y_i - x_i,$$

where x_i is the position of the i-th atom at the undeformed configuration.

Since lattice statics itself is a discrete model, we need to introduce coarse-grid operators in the context of the multigrid method. The natural choice is the continuum model, where the displacement field u is determined by solving the following minimization problem:

$$u = \operatorname{argmin}_{v\in X} \int_\Omega \big(W(\nabla v(x)) - f(x)\cdot v(x)\big)\,\mathrm{d}x. \tag{2.2}$$

Here Ω is the domain occupied by the material at the undeformed state, X is a suitable function space, and W is the stored energy density depending on the displacement gradient. For example, linear elasticity model [12] can be obtained from (2.2) if W is a quadratic. A more general proposal is to construct W from the atomistic model via the CB rule [2].

For simple lattice, the CB rule works as follows. Let $F = I + Dv$ be the deformation gradient tensor, and $E_0(F)$ be the energy of the unit cell in the deformed lattice whose lattice vectors $\{a_i\}_{i=1}^3$ are given by

$$a_i = FA_i,$$

where $\{A_i\}_{i=1}^3$ are lattice vectors of the undeformed lattice. The stored energy density is given by

$$W_{\text{CB}}(F) = \frac{E_0(F)}{v_0},$$

where v_0 is the volume of the unit cell in the equilibrium state. Considering a two-body potential function V_2, we write W_{CB} as

$$W_{\text{CB}}(F) = \frac{1}{2v_0}\sum_s V_2(Fs),$$

where s runs over the range of the potential function.

As an illustrative example, we consider one-dimensional chain interacting through Lennard-Jones potential [15] with the nearest neighbor interaction

$$V(r)=4\left(\left(\frac{\sigma}{r}\right)^{12}-\left(\frac{\sigma}{r}\right)^{6}\right), \tag{2.3}$$

where σ is a parameter representing the atomic length scale. The explicit expression of W_{CB} as

$$W_{\mathrm{CB}}(F)=\frac{1}{r^*}\left(|F|^{-12}-2|F|^{-6}\right) \tag{2.4}$$

with equilibrium bond length $r^*=\sqrt[6]{2}\sigma$.

The CB elasticity then reads

$$u=\mathrm{argmin}_{v\in X}\int_\Omega \left(W_{\mathrm{CB}}\left(\nabla v(x)\right)-f(x)\cdot v(x)\right)\mathrm{d}x, \tag{2.5}$$

where W_{CB} is the stored energy density obtained from the CB rule. We refer to [22] for the expression of W_{CB} when the potential function V is the EAM potential [10].

2.2 The one-way multigrid method

Consider a nested sequence of triangulations $\mathcal{T}_0\subset\mathcal{T}_1\subset\cdots\subset\mathcal{T}_l$ of Ω. The main steps of the one-way multigrid method [5] are as follows.

Step 1 Take $u_0=0$ as the initial guess and minimize the CB elasticity problem (2.5) discretized over $\mathcal{T}_0$ to obtain $\tilde{u}_0$.

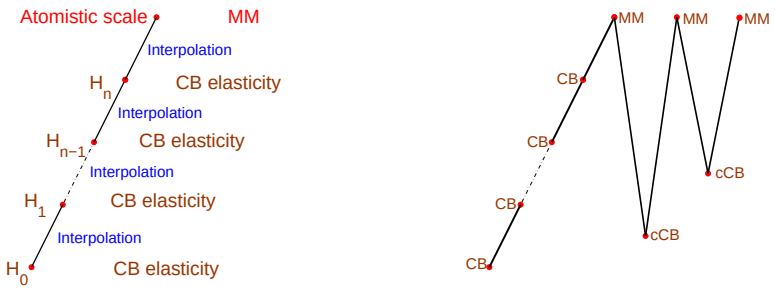
Step 2 For $i=1,\cdots,l$,

1. Interpolate
$$u_i=I_{i-1}^i\tilde{u}_{i-1},$$
where I_{i-1}^i is the standard finite element interpolation operator.
2. Take u_i as the initial guess and minimize the CB elasticity problem (2.5) discretized on $\mathcal{T}_i$ to obtain $\tilde{u}_i$.

Step 3 Solve the atomistic problem (2.1) with the CB state $x+\tilde{u}_l(x)$ as the initial guess.

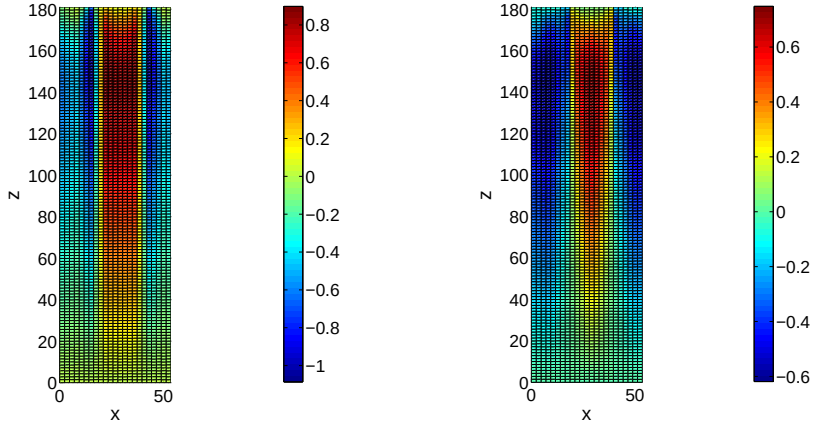
Fig. 1(a) is the schematic picture of the one-way multigrid method. As shown in [5], it is effective to capture collective motion of atoms due to boundary loading and is insensitive to the initial guesses and the parameters of the relaxation process. As the coarse-grid operator, the CB elasticity model can effectively capture the atomistic information at coarse levels for homogeneous deformations.

However, in nanoindentation, the CB elasticity is less effective since the local inhomogeneous deformation around the indenter is missing in the continuum model. Take state C in Fig. 4 as an example. Fig. 2 includes contour plots of the z component difference



(a) (b)

Figure 1: Schematic pictures of multigrid methods for lattice statics. (a) The one-way multigrid method. (b) The constrained CB (cCB) elasticity accelerated multigrid method.



(a) (b)

Figure 2: Contour plots of the z component difference of the displacement field in an interior slice in the $x-z$ plane. (a) Difference between the equilibrium solution obtained by the one-way multigrid method in Section 2.2 and the solution obtained by only performing **Step 1** and **Step 2** in the one-way multigrid method. (b) Difference between the equilibrium solution and the solution obtained by replacing **Step 3** with five minimization sweeps in the one-way multigrid method.

of the displacement field in an interior slice in the $x-z$ plane. Fig. 2(a) is the difference between the equilibrium solution obtained by the one-way multigrid method here and the solution obtained by only performing **Step 1** and **Step 2** in the one-way multigrid method. Fig. 2(b) is the difference between the equilibrium solution and the solution obtained by replacing **Step 3** with five minimization sweeps in the one-way multigrid method. The x component difference is similar to the z component difference and the y component difference is negligible due to the periodic boundary condition, which are not included here.

The long range displacement due to the boundary loading, can be captured in **Step 1** and **Step 2**. Dislocations, on the other hand, generate a displacement field with $1/r$ decay

where r is the distance to the dislocation core. The second part cannot be captured by Cauchy-Born elasticity, or **Step 1** and **Step 2** in the one-way multigrid method (Fig. 2(a)). Conventional minimization sweeps are efficient to relax atoms locally, but requires a lot of sweeps to generate a global displacement field. In Fig. 2(b), atoms under the top surface are nearly in their equilibrium states while interior atoms are not, which implies why the one-way multigrid method is less efficient in the case.

The good news is that more accurate information around the indenter is obtained with a few conventional minimization sweeps. If this can be properly incorporated at the continuum level, the long range effect of dislocations can be captured in a more effective manner. As we will introduce later, the displacement field around the indenter, obtained from a few minimization sweeps at the atomistic level, will be restricted on meshes at the continuum level, constrained and treated as displacement boundary conditions. The CB elasticity model, together with such boundary conditions, is then called the constrained CB elasticity model. Since the information of dislocations is included, meshes for the constrained CB elasticity model are different from those for the CB elasticity model in general.

2.3 The constrained CB elasticity accelerated multigrid method

We define another sequence of triangulations $\{\mathscr{T}_j\}_{j=1}^m$ and the choice of these triangulations will be discussed in Section 3. As illustrated in Fig. 1(b), the constrained CB elasticity accelerated multigrid method is described as follows.

Step 1 Take $u_0=0$ as the initial guess and minimize the CB elastic problem (2.5) discretized on $\mathcal{T}_0$ to obtain $\tilde{u}_0$.

Step 2 For $i=1,\cdots,l$,

1. Interpolate
$$u_i=I_{i-1}^i\tilde{u}_{i-1},$$
where I_{i-1}^i is the standard finite element interpolation operator.
2. Take u_i as the initial guess and minimize the CB elastic problem (2.5) discretized on $\mathcal{T}_i$ to obtain $\tilde{u}_i$.

Step 3 Relax (2.1) n_r conventional minimization sweeps with the constrained CB state $x+\tilde{u}_l(x)$ as the initial guess and obtain $y^{(0)}$. We do not resolve all details in (2.1) at this step. Choice of n_r will be discussed in Section 3.

Step 4 For $j=1,\cdots,m$,

1. Decompose
$$\hat{u}_j=u_{c,j}+u_{a,j},$$
where $\hat{u}_j=y^{(j-1)}-x$ is the deformation at the atomistic scale, $u_{c,j}$ is the restriction of $\hat{u}_j$ on the triangulation $\mathscr{T}_j$ and $u_{a,j}=\hat{u}_j-u_{c,j}$. The larger components of $u_{c,j}$ are constrained and treated as displacement boundary conditions in the constrained CB elasticity model, which results constrained meshes in Fig. 7 for example.

2. Solve the constrained CB elasticity on $\mathscr{T}_j$ with $u_{c,j}$ as the initial guess and obtain $\tilde{u}_{c,j}$.
3. Relax (2.1) n_r sweeps with the constrained CB state $x+u_{a,j}+\tilde{u}_{c,j}$ as the initial guess and obtain $y^{(j)}$.

Step 5 Solve (2.1) with $y^{(m)}$ as the initial guess. All atomistic details are resolved and the number of sweeps is denoted by n_f.

It is worth mentioning that the first three steps are similar to the one-way multigrid method [5]. In **Step 4**, we also can solve the constrained CB elasticity model in the framework of the one-way multigrid method. Numerical results show that this strategy cannot significantly improve the efficiency of the method.

3 Results

To illustrate how the constrained CB elasticity accelerated multigrid method works for nanoindentation, we start with a small scale problem including 24576 atoms, test all factors that affect the efficiency of the method, and then end up with a large problem with 1572864 atoms.

We consider a FCC Aluminium crystal with EAM potential [6, 10]. The CB elasticity model is approximated by the hexahedron element with standard eight-point Gauss quadrature rule. Newton-Raphson (NR) method with the linesearch [9] is employed to solve the continuum model, and conjugate-gradient (CG) method with Fletcher-Reeves formula and bisection linesearch [19] are used to solve the atomistic model. Resulting linear systems are solved by a parallel sparse direct solver *MUMPS* [1]. We simulate the indentation process along $[\bar{1}10]$ (dislocation) direction with a rectangular indenter (Fig. 3).

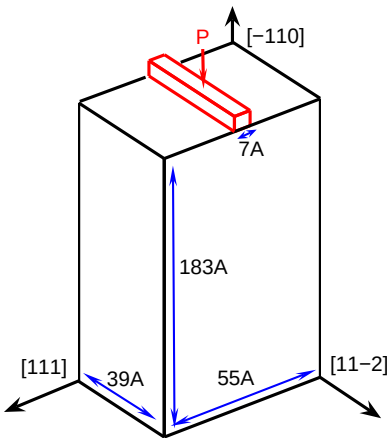
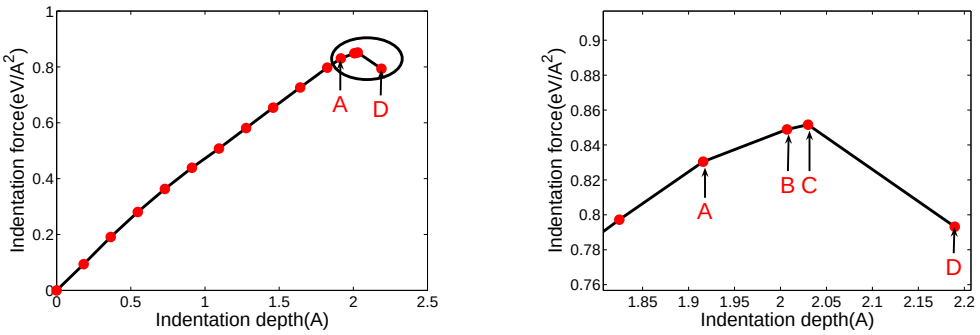


Figure 3: Schematic representation of nanoindentation along dislocation direction. The indenter width is $6.98\mathring{A}$. The occupied domain is $55.9\mathring{A}\times 39.5\mathring{A}\times 182.5\mathring{A}$ in the small problem, and $223.5\mathring{A}\times 158.0.5\mathring{A}\times 729.9.5\mathring{A}$ in the large problem.



(a) (b)

Figure 4: Load-displacement curve in nanoindentation. (a) Full picture. (b) Local picture after zooming in the ellipse.

The $[111],[11\bar{2}]$ and $[\bar{1}10]$ directions are viewed as x,y and z axes, respectively. We impose periodic boundary conditions on both x and y directions, while Dirichlet boundary condition on the top surface under the indenter and the bottom surface in the z direction, and stress free boundary condition on the remaining part of the top surface. The width of the indenter is $6.98\mathring{A}$.

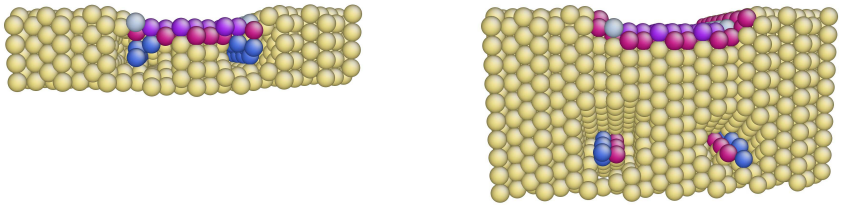
For the small scale problem, the domain Ω is approximately $55.9\mathring{A}\times 39.5\mathring{A}\times 182.5\mathring{A}$. The load-depth curve is plotted in Fig. 4(a). To illustrate the nucleation of dislocations, we choose an ellipse over the region where the slope of the curve is noticeably changed. After zooming in, we select four representative points labeled by A, B, C and D sequentially, among which C is the transition point with the largest strength of loading. We visualize the atomic configuration by Atomeye [16] for states A, B, C and D in Fig. 5 and Fig. 6. The number of atoms touching the given atom is defined as its coordinate number. For an ideal FCC crystal, the ones with coordinate number 12 are called regular atoms, while the others are called irregular atoms. Table 1 shows the one-to-one correspondence between color and coordinate number. In particular, the coordinate number of atoms associated with the dislocation here is 13. There is no dislocation for state A. Dislocations have nucleated from state B. At state C, dislocations grow but still stick to the indenter. Dislocations leave the indenter and propagate along the dislocation direction at state D. Before the load-displacement curve decreases, dislocations indeed occurred (state B). Therefore, one cannot decide the onset of plasticity by this curve. This study reveals the deficiency of the commonly used criterion for dislocation nucleation, and it is conforming with experiment results [18] and numerical results based on cylindrical indenter [14], although the system we test here is small.

Table 1: One-to-one correspondence between color and coordinate number.

Color	Purple	LightSteelBlue	MediumVioletRed	LightGoldenrod	Blue
Coordinate number	7	8	11	12	13

(a) (b)

Figure 5: Atomic configuration of states A and B around the indenter. (a) State A. (b) State B.



(a) (b)

Figure 6: Atomic configuration of state C and D around the indenter. (a) State C. (b) State D.

In what follows, the CPU time is recorded on LSSC-III with one Intel X5550 processor as a measure of the computational cost. Table 2 shows the CPU time of NR method for CB elasticity.

Table 2: CPU time of NR method vs. mesh (**Steps 1 2** in Section 2.2).

Mesh	$8\times4\times2$	$8\times4\times8$	$8\times4\times32$
CPU time (s)	2.2	0.5	2.1

Table 3 shows the CPU time of CG method in the one-way multigrid method for lattice statics. This problem has local inhomogeneous deformations around the indenter. The largest CPU time is required at state C which has the most inhomogeneous deformations. Dislocations carry away certain plasticity when they leave the indenter and inhomogeneity around the indenter is weakened. This explains that CPU time at state D is smaller than that at state C.

Table 3: CPU time of CG method vs. depth (**Step 3** in Section 2.2).

Depth($\mathring{A}$)	0.55	1.64	1.91(A)	2.00(B)	2.03(C)	2.55(D)
CPU time (s)	698.5	1382.1	3420.0	3412.7	18235.2	9111.8

In what follows, we discuss some factors that affect the efficiency of the constrained CB elasticity accelerated multigrid method. The first one is the choice of the constrained CB elasticity model, which closely depends on triangulations $\{\mathscr{T}_j\}_{j=1}^m$. For the time being, we choose the constrained points according to the magnitude of local deformations. Since the atomistic system undergoes inhomogeneous deformation around the indenter, we

(a) (b)

Figure 7: Constrained meshes in $x-z$ plane. Red solid squares are constrained and their information will not be updated when solving the CB elasticity, while blue points are updated. (a) $8\times8\times4$. (b) $8\times8\times8$.

select mesh points under the indenter as the constrained points. When solving the CB elasticity model on $\{\mathscr{T}_j\}_{j=1}^m$, we fix the constrained points and only relax the rest ones. Fig. 7 is the $x-z$ section of meshes for the constrained CB elasticity model. Red solid squares are constrained points and their information is not updated when solving the constrained CB continuum model, while blue points are updated. We set meshes for the constrained CB continuum model as in Table 4.

Table 4: Meshes for the constrained CB continuum model.

Triangulation	$\mathscr{T}_1$	$\mathscr{T}_2$	$\mathscr{T}_3$	$\mathscr{T}_4$	$\mathscr{T}_5$
Mesh	$8\times8\times4$	$8\times8\times8$	$8\times8\times16$	$8\times8\times32$	$8\times8\times64$

The second one is the choice of n_r. We solve the constrained CB elasticity on $\mathscr{T}_1$ and $\mathscr{T}_2$. For state C, we choose different n_r and count n_f (**Step 5** in Section 2.3); see Table 5. Results show that the best value for n_r is smaller than five. This observation is similar to pre-smoothing and post-smoothing steps in traditional multigrid methods [25].

Table 5: CPU time (**Step 5** in Section 2.3) vs. n_r.

n_r	1	2	3	5	10
CPU time (s)	2648.0	2679.5	2671.2	2735.8	3008.4

The last one is the choice of m. Still for state C, we fix $n_r=3$ and measure the total CPU time in terms of m. Before that, we need to measure CPU time on different meshes and atomistic level and add them together. CPU time on different meshes is shown in Table 6, and the total CPU time in the constrained CB elasticity accelerated multigrid method is shown in Table 7 when $m=2$. From Table 8, we can see that $m\leq2$ is the best choice.

Compared with the one-way multigrid method, the constrained CB elasticity accelerated multigrid method saves quite a bit. For state C, the ratio of the total CPU time

Table 6: CPU time on different meshes (**Steps 1, 2, 4.1, 4.2** in Section 2.3) when $m=2$.

Mesh	$\mathcal{T}_0$	$\mathcal{T}_1$	$\mathcal{T}_2$	$\mathscr{T}_1$	$\mathscr{T}_2$	Sum
CPU time (s)	2.2	0.5	2.1	19.3	86.5	110.6

Table 7: The total CPU time of the constrained CB elasticity accelerated multigrid method for state C when $m=2$.

Contribution	**Step 1, 2, 4.1, 4.2**	**Step 3, 4.3**	**Step 5**	Sum
CPU time (s)	110.6	89.6	2903.5	3103.7

Table 8: The total CPU time of the constrained CB elasticity accelerated multigrid method vs. m at state C.

m	1	2	3	4
Total CPU time (s)	3029.8	3103.7	3310.2	3552.8

Table 9: Total CPU time (in seconds) of different methods for different states.

State	A	B	C	D
The cCB elasticity accelerated multigrid method	2478.5	3075.2	3103.7	2000.2
The one-way multigrid method	3424.8	3417.5	18240.0	9116.6
Ratio	72%	90%	17%	22%

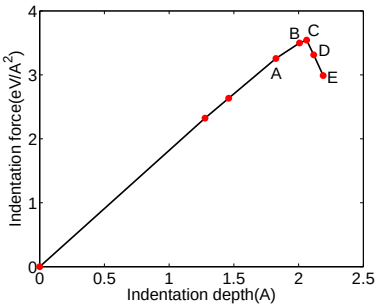
between these two methods is

$$\frac{3103.7}{2.2+0.5+2.1+18235.2}\approx 17\%.$$

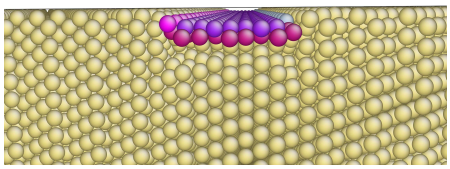
With the above preparations, we use the constrained CB elasticity accelerated multigrid method to simulate the whole nanoindentation process. In particular, we list results of states A, B, C and D in Table 9. We learn from the table that the constrained CB elasticity model can effectively accelerate the convergence of the relaxation process and reduce the computational cost to a reasonable level. When dislocations occur and move, it saves over 80% computational cost.

We finish this section with a larger problem with 1572864 atoms. The occupied domain is approximately $223.5\mathring{A}\times 158.0.5\mathring{A}\times 729.9.5\mathring{A}$. The load-depth curve is reported in Fig. 8(a) with five representative points labeled by A, B, C, D and E sequentially, and atomic configurations in Figs. 8(b), 9(a), 9(b), 10(a), and 10(b), respectively.

Similar to the previous example, dislocations have nucleated from state B ($2.01\mathring{A}$) while the maximum load is achieved at state C ($2.06\mathring{A}$). An interim state D ($2.12\mathring{A}$) is observed where only dislocations in the left leave the indenter. All observations here, consistent with results of the small problem, are conforming with experiment results [18] and numerical results [14]. Results of state A, B, C, D and E in Table 10 show that the constrained CB elasticity model can effectively accelerate the convergence of the relaxation process and save over 70% CPU time when dislocations nucleate and move.

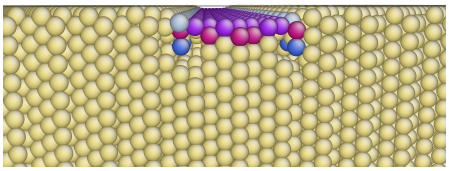


(a)

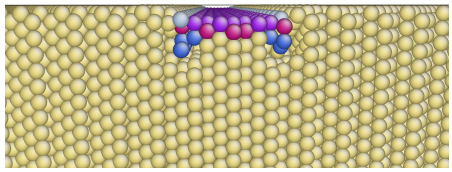


(b)

Figure 8: (a) Load-displacement curve in the larger problem. (b) Atomic configuration of state A around the indenter in the larger problem.

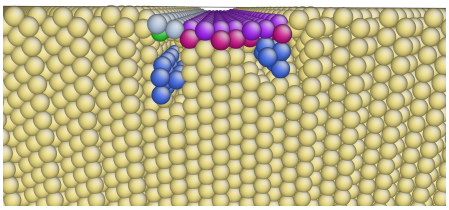


(a)

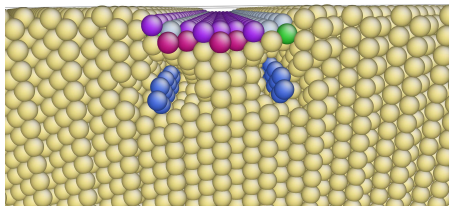


(b)

Figure 9: Atomic configuration of states B and C around the indenter in the larger problem. (a) State B. (b) State C.



(a)



(b)

Figure 10: Atomic configuration of states D and E around the indenter in the larger problem. (a) State D. (b) State E.

Table 10: Ratio of CPU time between the constrained CB accelerated multigrid method and the one-way multigrid method for different states in the large scale problem.

State	A	B	C	D	E
Ratio	90%	41%	30%	30%	25%

4 Discussions

The selection of constrained points in the constrained CB elasticity model in Section 3 is based on local deformations. This can be done in a more rigorous way by a posteriori error estimate [20], which has shown its robustness and efficiency in various situations.

To understand why the constrained CB elasticity can accelerate the relaxation process in nanoindentation, we look at a one-dimensional chain with $N+1$ atoms and Lennard-Jones potential up to the nearest neighbor interaction and give an explanation in the appendix.

Multigrid method has been proven to efficiently capture the collective motion of atoms, while domain decomposition method can separate different length scales into parts and then treat them in a divide-and-conquer manner [17]. Further work will involve with the combination of these two methods which can solve problems more efficiently in general situations.

Acknowledgments

The work of Chen is supported by National Science Foundation grant DMS-1217315. The work of Ming was supported by National Natural Science Foundation of China under the grant 10932011, 91230203 and by the funds for creative research group of China (Grant No. 11021101), and by the support of CAS National Center for Mathematics and Interdisciplinary Sciences. The work of Yang was supported by National Natural Science Foundation of China under the grants 11001210 and 11171305 and 91230203.

Appendix: Illustration of the efficiency of the constrained CB elasticity

We keep the left-most atom fixed and apply the displacement δ for the right-most atom, i.e.,

$$u_1=0, \quad u_{N+1}=\delta.$$

We choose positions of atoms in the undeformed state as the horizontal coordinates and their displacements as the vertical coordinates. For simplicity, only nearest neighbor interaction will be considered. By the symmetry argument, the force of a atom is 0 if its distances to two neighbors are equal. Thus in a minimization step, only those atoms whose distances to their two neighbors are not equal will be updated.

Let us firstly study conventional lattice statics. In Fig. 11, red circles denote positions of atoms before one minimization step, whose displacements $u_i=0$, $i=2,\cdots,N$. Dotted line in blue is the exact displacement field (elastic state), whose deformation $u_i=(i-1)\delta/N$, $i=2,\cdots,N$. Circles in black are atoms whose positions are updated during one minimization step. We describe the five sub-figures in Fig. 11 as follows:

1. Initial guess.
$$u_i^{(1)}=0, \quad i=2,\cdots,N.$$
2. Update configuration according to one minimization step. Only position of the first atom (next to the right-most atom) is changed since other atoms are all in the

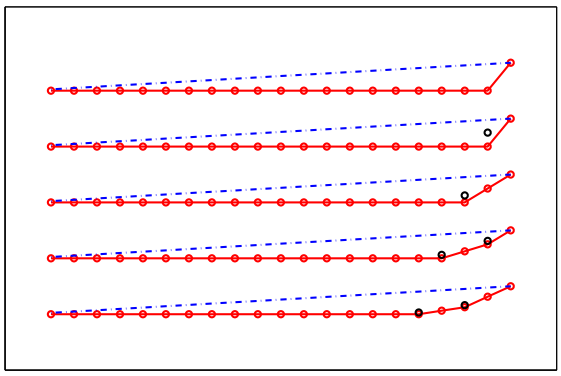


Figure 11: Step by step illustration of lattice statics under tensile deformation. Loci of atoms in the undeformed state and their displacements are chosen as horizontal coordinates and vertical coordinates, respectively. Atoms before one minimization step are denoted by red circles, while dotted lines represent the elastic state (exact displacement). Circles in black are atoms whose positions are updated during one minimization step since only nearest neighbor interaction is considered. Initial configuration and configurations after four minimization steps are illustrated from top to bottom.

equilibrium state.

$$u_i^{(2)}=0,\quad i=2,\cdots,N-1,\quad u_N^{(2)}=\frac{1}{2}\delta.$$

3. Only configuration of the second atom (next to the right-most atom) is updated.

$$u_i^{(3)}=0,\quad i=2,\cdots,N-2,\quad u_{N-1}^{(3)}=\frac{1}{4}\delta,\quad u_N^{(3)}=\frac{1}{2}\delta.$$

4. Configurations of the first and third atoms (next to the right-most atom) are updated.

$$u_i^{(4)}=0,\quad i=2,\cdots,N-3,\quad u_{N-2}^{(4)}=\frac{1}{8}\delta,\quad u_{N-1}^{(4)}=\frac{1}{4}\delta,\quad u_N^{(4)}=\frac{5}{8}\delta.$$

5. Configurations of the second and fourth atoms (next to the right-most atom) are updated.

$$u_i^{(5)}=0,\quad i=2,\cdots,N-4,\quad u_{N-3}^{(5)}=\frac{1}{16}\delta,\quad u_{N-2}^{(5)}=\frac{1}{8}\delta,\quad u_{N-1}^{(5)}=\frac{3}{8}\delta,\quad u_N^{(5)}=\frac{5}{8}\delta.$$

The second atom (next to the left-most atom) will be updated only after N minimization steps. Therefore, at least $\mathcal{O}(N)$ minimization steps are needed in lattice statics to obtain the elastic state. Since the optimal computational cost for one minimization step is $\mathcal{O}(N)$, the total computational cost is $\mathcal{O}(N^2)$.

We now turn to the performance of the constrained CB elasticity accelerated multigrid method (Fig. 1(b)). We use the displacements of green points as the boundary condition in the constrained CB elasticity model. The descriptions for the six sub-figures are as follows:

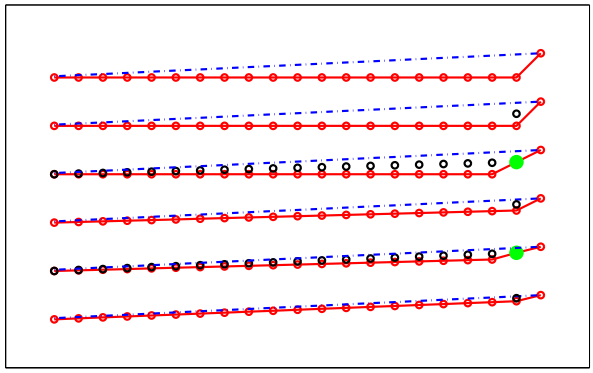


Figure 12: Step by step illustration of the constrained CB elasticity accelerated multigrid method under tensile deformation. Loci of atoms in the undeformed state and their displacements are chosen as horizontal coordinates and vertical coordinates, respectively. Atoms before one minimization step are denoted by red circles, while dotted lines represent the elastic state (exact displacement). Circles in black are atoms whose positions are updated during one minimization step. Green points are boundary points used in the constrained CB elasticity model and their displacements are treated as boundary conditions for the constrained CB elasticity model. Initial configuration and configurations after the five minimization steps are illustrated from top to bottom.

1. Initial guess.
$$u_i^{(1)}=0, \quad i=2,\cdots,N.$$

2. Update configuration according to one minimization step. Only configuration of the first atom (next to the right-most atom) is changed since other atoms are all in the equilibrium state.
$$u_i^{(2)}=0, \quad i=2,\cdots,N-1, \quad u_N^{(2)}=\frac{1}{2}\delta.$$

3. Update configuration according to the constrained CB elasticity. Configurations of all atoms left to the constrained atoms are updated.
$$u_i^{(3)}=\frac{i-1}{2(N-1)}\delta, \quad i=2,\cdots,N.$$

4. Only configuration of the first atom (next to the right-most atom) is updated according to one minimization step.
$$u_i^{(4)}=\frac{i-1}{2(N-1)}\delta, \quad i=2,\cdots,N-1, \quad u_N^{(4)}=\frac{u_{N-1}^{(3)}+u_{N+1}}{2}=\frac{3N-4}{4(N-1)}\delta.$$

5. Update configuration according to constrained CB elasticity.
$$u_i^{(5)}=\frac{i-1}{N-1}\frac{3N-4}{4(N-1)}\delta, \quad i=2,\cdots,N.$$

6. Only configuration of the first atom (next to the right-most atom) is updated according to one minimization step.

$$u_i^{(6)}=\frac{i-1}{N-1}\frac{3N-4}{4(N-1)}\delta,\quad i=2,\cdots,N-1,\quad u_N^{(6)}=\frac{u_{N-1}^{(5)}+u_{N+1}}{2}=\frac{7N^2-18N+12}{8(N-1)^2}\delta.$$

If N is large enough, $u_N^{(2)}=\delta/2$, $u_N^{(4)}=3\delta/4$ and $u_N^{(6)}=7\delta/8$ asymptotically. A simple induction gives that $u_N^{(2k)}=(2^k-1)\delta/2^k$. We know the exact deformation of the N-th atom $u_N=(N-1)\delta/N$. Therefore, the exact deformation of the N-th atom can be obtained only after $\mathcal{O}(\log_2 N)$ iterations asymptotically, and then the exact deformation of other atoms can be obtained by the constrained CB elasticity model as shown above. Since the optimal computational cost for one minimization step is $\mathcal{O}(N)$, the total computational cost is asymptotically $\mathcal{O}(N\log N)$.

In nanoindentation, the constrained CB elasticity model plays a similar role as that in the one-dimensional example. The constrained CB elasticity model can effectively capture the collective motion of atoms induced by dislocations, and then accelerate the convergence. Therefore, the constrained CB elasticity accelerated multigrid method is more efficient than the conventional lattice statics.

References

[1] Mumps: a parallel sparse direct solver, `http://graal.ens-lyon.fr/MUMPS/`.
[2] M. Born and K. Huang, Dynamical theory of crystal lattices, Oxford University Press, 1954.
[3] A. Brandt, Multi-level adaptive solutions to boundary-value problems, Math. Comp. 31 (1977), 333–390.
[4] A. Brandt, Multigrid methods in lattice field computations, Nuclear Physics B (Proc. Suppl.) 26 (1992), 137–180.
[5] J. Chen and P.B. Ming, An efficient multigrid method for molecular mechanics modeling in atomic solids, Commun. Comput. Phys. 10 (2011), 70–89.
[6] M.S. Daw and M.I. Baskes, Embedded-atom method: Derivation and application to impurities, surfaces, and other defects in metals, Phys. Rev. B 29 (1984), 6443.
[7] P. Deuflhard, P. Leinen and H. Yserentant, Concepts of an adaptive hierarchical finite element code, IMPACT Comput. Sci. Engrg. 1 (1989), 3–35.
[8] W. E and P.B. Ming, Cauchy-Born rule and the stability of the crystalline solids: Static problems, Arch. Rational Mech. Anal. 183 (2007), 241–297.
[9] S.C. Eisenstat and H.F. Walker, Choosing the forcing terms in an inexact Newton method, SIAM J. Sci. Comput. 17 (1996), 16–32.
[10] F. Ercolessi and J.B. Adams, Interatomic potentials from first-principles calculations: The force-matching method, Europhys. Lett. 26 (1994), 583.
[11] A.C. Fischer-Cripps, Nanoindentation, Springer-Verlag, 2004.
[12] S. Goedecker, F. Lançon, and T. Deutsch, Linear scaling relaxation of the atomic positions in nanostructures, Phys. Rev. B 64 (2001), 161102.
[13] A. Gouldstone, N. Chollacoop, M. Dao, J. Li, A.M. Minor, and Y.-L. Shen, Indentation across size scales and disciplines: Recent developments in experimentation and modeling, Acta Mater. 55 (2007), 4015–4039.

[14] J. Knap and M. Ortiz, Effect of indenter-radius size on Au(001) nanoindentation, Phys. Rev. Lett. 90 (2003), 226102.
[15] J.E. Lennard-Jones, On the determination of molecular fields. ii. from the equation of state of a gas, Proc. Roy. Soc. London Ser. A 106 (1924), 463–477.
[16] J. Li, Atomeye: an efficient atomistic configuration viewer, Model. Simul. Mater. Sci. Eng. 11 (2003), 173–177.
[17] R.E. Miller and E.B. Tadmor, A unified framework and performance benchmark of fourteen multiscale atomistic/continuum coupling methods, Model. Simul. Mater. Sci. Eng. 17 (2009), 053001.
[18] A.M. Minor, S.A.S. Asif, Z.W. Shan, E.A. Stach, E. Cyrankowski, T.J. Wyrobek, and O.L. Warren, A new view of the onset of plasticity during the nanoindentation of aluminium, Nature Materials 5 (2006), 697–702.
[19] J. Nocedal and S. Wright, Numerical optimization, second ed., Springer-Verlag, 2006.
[20] J.T. Oden, S. Prudhomme, A. Romkes, and P. Bauman, Multiscale modeling of physical phenomena: Adaptive control of models, SIAM J. Sci. Comput. 28 (2006), 2359–2389.
[21] B. Poon, D. Rittel, and G. Ravichandran, An analysis of nanoindentation in linearly elastic solids, Int. J. Solids Struct. 45 (2008), 6018–6033.
[22] E.B. Tadmor, M. Ortiz, and R. Phillips, Quasicontinuum analysis of defects in solids, Phil. Mag. A 73 (1996), 1529–1563.
[23] Phillips R. Tadmor, E.B. Miller and M. Ortiz, Nanoindentation and incipient plasticity, J. Mater. Res. 14 (1999), 2233–2250.
[24] A. Toselli and O. Widlund, Domain decomposition methods – algorithms and theory, Springer Series in Computational Mathematics, 2004.
[25] U. Trottenberg, C.W. Oosterlee, and A. Schüller, Multigrid, Academic Press Inc., San Diego, CA, 2001, with contributions by A. Brandt and P. Oswald and K. Stüben.