

An application of multigrid methods for a discrete elastic model for epitaxial systems

R.E. Caflisch^{a,*}, Y.-J. Lee^a, S. Shu^b, Y.-X. Xiao^c, J. Xu^d

^a Department of Mathematics, UCLA, 520 Portola Plaza, Los Angeles, CA 90095, USA

^b Institute for Computational and Applied Mathematics, Xiangtan University, Xiangtan 411105, PR China

^c Institute of Fundamental Mechanic and Material Engineering, Xiangtan University, Xiangtan 411105, PR China

^d Department of Mathematics and Center for Computational Mathematics and Application of Pennsylvania State University, PA, USA

Received 4 August 2005; received in revised form 21 February 2006; accepted 10 April 2006

Available online 27 June 2006

Abstract

We apply an efficient and fast algorithm to simulate the atomistic strain model for epitaxial systems, recently introduced by Schindler et al. [Phys. Rev. B 67, 075316 (2003)]. The discrete effects in this lattice statics model are crucial for proper simulation of the influence of strain for thin film epitaxial growth, but the size of the atomistic systems of interest is in general quite large and hence the solution of the discrete elastic equations is a considerable numerical challenge. In this paper, we construct an algebraic multigrid method suitable for efficient solution of the large scale discrete strain model. Using this method, simulations are performed for several representative physical problems, including an infinite periodic step train, a layered nanocrystal, and a system of quantum dots. The results demonstrate the effectiveness and robustness of the method and show that the method attains optimal convergence properties, regardless of the problem size, the geometry and the physical parameters. The effects of substrate depth and of invariance due to traction-free boundary conditions are assessed. For a system of quantum dots, the simulated strain energy density supports the observations that trench formation near the dots provides strain relief.

© 2006 Elsevier Inc. All rights reserved.

Keywords: Algebraic multigrid method; Discrete strain model; Epitaxial growth; Quantum dots; Nanocrystals

1. Introduction

Epitaxial thin film growth is the deposition of a material onto a substrate, in which the crystallographic order of the film is determined by that of the substrate as a result of atomistic matching between the two materials along the interface. Heteroepitaxy, i.e., epitaxial growth of a film on a substrate of a different material, is used

* Corresponding author. Tel.: +1 310 206 0200; fax: +1 310 206 2679.

E-mail addresses: caflisch@math.ucla.edu (R.E. Caflisch), yjlee@math.ucla.edu (Y.-J. Lee), shushi@xtu.edu.cn (S. Shu), xyx610xyx@yahoo.com.cn (Y.-X. Xiao), xu@math.psu.edu (J. Xu).

URLs: <http://www.math.ucla.edu/~caflisch> (R.E. Caflisch), <http://www.math.ucla.edu/~yjlee> (Y.-J. Lee), <http://www.math.psu.edu/xu> (J. Xu).

to grow layered thin films with a wide range of material properties. Strain, induced by lattice mismatch between substrate and film, plays an important role in the electronic, optical and magnetic properties of materials and can also serve as the driving force for a self-organization process. Therefore, an understanding of strain effects is crucial for the engineering of new material systems. Recently, a discrete atomistic strain model has been developed in [30]. In particular, the discrete elastic model is shown to be consistent with the predictions of continuum elastic theory, when appropriate, and to capture many effects that are essentially atomistic in nature.

An example of strain effects in a nanoscale heteroepitaxial system is the self-assembled growth of quantum dots through the Stranski–Krastanov growth mode, [12,15,35]. For the development of a wetting layer and the subsequent growth of islands in Stranski–Krastanov growth, atomistic strain effects are likely to play a crucial role. Simulation of this problem presents a considerable challenge since the relevant domain size contains a very large number of atoms. Furthermore, in order to analyze dynamical aspects of strain effects, the discrete strain model should be solved frequently as the material configuration changes. To meet this requirement for high throughput, a fast solution technique for an atomistic strain system is required.

In this paper, we propose an efficient algorithm that is suitable for solution of a large scale discrete strain model and report extensive numerical results. Our methodology is based on the idea of multiscale or multigrid methods. There exist a vast amount of works concerning multigrid methods; references include monographs [4,14] and survey papers [3,36,38,39]. The multigrid algorithm is known to be the most efficient iterative method for solving large scale algebraic systems. The classical multigrid algorithm based on a geometric hierarchy can be an effective solver for a well-structured problem formulated on a domain where such a hierarchy is immediately available.

The multigrid methodology has been extended to more general systems through the algebraic multigrid (AMG) method, which was first developed in the 1980s, [5,6,25,26,34] for the solution of linear systems coming from the discretization of scalar elliptic PDEs. The idea of AMG is to mimic geometric multigrid methods, namely, their functionality and convergence behavior, in applications where a hierarchy of nested meshes and interlevel transfer operators can not (or only with huge effort) be provided. It is worth remarking that for scalar PDEs, namely in case that there is only one unknown on each grid, there are well-known robust and efficient constructions of coarse grid and prolongation operator that can be used for fast solvers. However, the algebraic multigrid method for a system of PDEs, such as the equations of linear elasticity, is not well developed, and naive use of scalar AMG does not lead to a robust and efficient solver; rather it deteriorates and oftentimes its convergence breaks down, [36]. We refer readers to [7,13,33] for recent efforts to apply algebraic multigrid methods to systems of PDEs. We also remark that the main contribution in our work is not in developing novel algorithms, but rather in appropriately adapting and applying the existing AMG methodology to a physically important application and in demonstrating its effectiveness and wide range of utility. The AMG methods presented in our work can be regarded as an extension of the multigrid technique to a new class of equations, the discrete strain models, for which the detailed description can be found in Section 2.

Oftentimes, the discrete strain model is posed on a semi-infinite lattice geometry. There have been several techniques to reduce the size of the problems by introducing certain artificial boundary conditions. For example, Lam, Lee and Sander [16] introduced a numerical Green's function and solved the problem reformulated as an integral equation. So far, their method has been applied only for problems in two spatial dimensions with uniform elastic parameters throughout the material system. In [19,28], numerical acceleration was provided by implementing an artificial boundary condition (ABC), and the resulting system of equations was solved by the Conjugate Gradient method. Recently, Smereka and Russo developed a multigrid method based on constructing a geometric hierarchy of grids to solve the resulting system that is formulated with ABCs, [27]. To the best of our knowledge, the present work is the first use of an algebraic multigrid method to solve the discrete strain model, which is of apparent advantages compared to the geometric multigrid method when considering the dynamic change of the lattice geometries during growth simulations. Although the results reported in this paper are restricted to two-dimensional systems, extension to three dimensions is straightforward and already carried out, along with use of an ABC. Those will be the subject of future publications.

The rest of the paper is organized as follows. In Section 2, we briefly review and discuss the structure and properties of the discrete strain models. The algorithm to solve the proposed system is then introduced in Section 3. A number of numerical experiments are reported and summarized in Section 4. In Section 5, several physical problems are provided to demonstrate how our method can be effectively used.

2. Strain model of epitaxial films

In this section, we briefly review the discrete elastic strain models that are introduced in [30] and discuss their characteristics. To discuss the similarities and differences between classical elasticity and the discrete strain model, we begin with reviewing classical elasticity. Let $\mathbf{u} = (u_1, u_2)$ denote the Cartesian components of the displacement vector. For linear elasticity in an isotropic material, the components of the strain tensor D and stress tensor S are given in terms of the displacement vector as follows:

$$D_{k\ell} = \frac{1}{2} \left(\frac{\partial u_\ell}{\partial x_k} + \frac{\partial u_k}{\partial x_\ell} \right), \tag{2.1}$$

$$S_{k\ell} = \lambda \delta_{k\ell} \text{tr}(D) + 2\mu D_{k\ell}, \tag{2.2}$$

where λ, μ are the Lamé constants and $\text{tr}(D)$ is the trace of tensor D .

For the heteroepitaxial growth of a film with lattice constant a_f on a substrate with a lattice constant a_s , the normalized lattice mismatch is defined through

$$\epsilon = \frac{a_f - a_s}{a_s}. \tag{2.3}$$

To indicate that we are considering the strain tensor on the film part, we shall use D^f , which is given by

$$D_{k\ell}^f = \frac{1}{2} \left(\frac{\partial u_\ell^f}{\partial x_k} + \frac{\partial u_k^f}{\partial x_\ell} \right), \tag{2.4}$$

where the superscript f denotes quantities associated with the film. The key modification for the tensor is that using the Eq. (2.3), D^f can be expressed in terms of displacements with respect to the substrate lattice positions as follows:

$$D_{k\ell}^f = D_{k\ell} + \delta_{k\ell} \epsilon, \tag{2.5}$$

$$S_{k\ell}^f = \lambda^f \delta_{k\ell} \text{tr}(D) + 2\mu^f D_{k\ell} + (2\lambda^f + \mu^f) \delta_{k\ell} \epsilon, \tag{2.6}$$

where λ^f, μ^f are the Lamé constants.

In mechanical equilibrium, the forces inside any volume Ω vanish,

$$\text{div} S = 0, \quad \text{div} S^f = 0. \tag{2.7}$$

If there is no external traction, the boundary condition is

$$\mathbf{n} \cdot S = 0, \tag{2.8}$$

where $\mathbf{n} = (n_1, n_2)$ is the vector normal to the surface.

Additionally, at the interface between the film and the substrate, the normal component of the stress is continuous, i.e.,

$$\mathbf{n} \cdot S = \mathbf{n} \cdot S^f. \tag{2.9}$$

The elastic energy density \mathcal{E} is then given by the tensor contraction as follows:

$$\mathcal{E} = \frac{1}{2} D : S = \mu D : D + \frac{1}{2} \lambda (\text{div } \mathbf{u})^2, \tag{2.10}$$

so that the total energy is

$$\mathcal{E}^{\text{total}} = \int_{\Omega} \mathcal{E} \, dx.$$

The force balance equations and boundary conditions for each point then follow from the variational principle of the elastic energy with each of the displacements being equal to zero, i.e.,

$$\frac{\delta \mathcal{E}^{\text{total}}}{\delta u_k} = 0, \quad \text{for } k = 1, 2, 3$$

which is equivalent to the classical linear elasticity equation:

$$\mu \operatorname{div} \mathbf{D} + \frac{1}{2} \lambda \operatorname{grad} \operatorname{div} \mathbf{u} = 0. \tag{2.11}$$

There is no force in the bulk of the substrate or film, but there are forces on the interface between the substrate and film due to the lattice mismatch between substrate and film. There may also be forces at the top of the film (i.e., the interface between film and vacuum), due to the use of a nonequilibrium reference state in the film.

We shall now introduce the discrete atomistic model for the strain in two spatial dimensions; the extension to three dimensions is straightforward. For epitaxial systems, the model is formulated based on lattice statics. The main idea is not to discretize the Eq. (2.11), but to construct a discrete version of energy, (2.10). To ease our presentation, we let $f : \mathbb{R}^2 \mapsto \mathbb{R}$ be any function and \mathbf{e}_k ($k = 1, 2$) be the canonical basis for the Cartesian coordinate system. We then define translation and difference operators as follows:

$$\begin{aligned} T_k^\pm f(i) &:= f(i \pm \mathbf{e}_k), \\ D_k^+ f(i) &:= \frac{(T_k^+ - 1)f(i)}{h}, \\ D_k^- f(i) &:= \frac{(1 - T_k^-)f(i)}{h}, \\ D_k^0 f(i) &:= \frac{(T_k^+ - T_k^-)f(i)}{2h}. \end{aligned}$$

Note that D_k^+ and D_k^- denote forward and backward difference operators and D_k^0 denotes the centered difference operator. The discrete strain component at a point \mathbf{i} is defined as

$$S_{k\ell}^{pq} = \frac{D_k^p u_\ell + D_\ell^q u_k - \epsilon \delta_{k\ell} \chi}{2},$$

where $k, \ell = 1$ or 2 , $p, q = \pm$ and χ is one only if \mathbf{i} is a film point and zero otherwise. In the substrate, $D_k^p u_\ell$ is the displacement relative to the equilibrium lattice, while in the film this is the displacement relative to a reference (non-equilibrium) lattice, which is the reason for the presence of ϵ in the definition of $S_{k\ell}^{pq}$, see Fig. 1 below.

At interior points, i.e., away from any interfaces, the energy density $E(\mathbf{i})$ is defined through:

$$E(i) = \alpha \sum_{p=\pm, k=1,2} (S_{kk}^p)^2 + \sum_{p=\pm, q=\pm} (2\beta(S_{12}^{pq})^2 + \gamma S_{11}^p S_{22}^q), \tag{2.12}$$

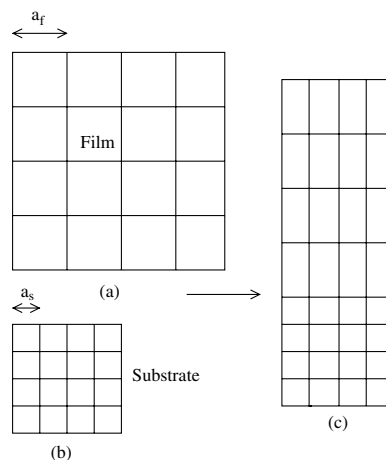


Fig. 1. (a) Equilibrium lattice for film with lattice size a_f . (b) Equilibrium lattice for substrate with lattice size a_s . (c) The reference (non-equilibrium) lattice in which strain is introduced due to lattice mismatch.

where the elastic constant α , β and γ are chosen so that the model is consistent with continuum elasticity, namely,

$$\alpha = C_{11}/4, \beta = C_{44}/4, \gamma = C_{12}/4 \tag{2.13}$$

in which C_{ij} are the Voigt constants. Details of the energy density (2.12), as described below, are based on atomic interactions, [30].

For isotropic elasticity, the coefficients are

$$\alpha = (\lambda + 2\mu)/4, \beta = \mu/4, \gamma = \lambda/4. \tag{2.14}$$

For problems in which the underlying lattice has cubic symmetry but the material geometry includes interfaces, we generalize the energy in (2.12) as follows:

$$E(i) = \sum_{p=\pm, k=1,2} \alpha_k^p (S_{kk}^p)^2 + \sum_{p=\pm, q=\pm} \left(2\beta^{pq} (S_{12}^{pq})^2 + \gamma^{pq} S_{11}^p S_{22}^q \right), \tag{2.15}$$

where each coefficient α_k^p , as well as the lattice mismatch parameter ϵ , corresponds to a bond between two atoms; each of the coefficients β^{pq} and γ^{pq} corresponds to the interaction of two bonds in orthogonal directions from a central point, which defines a square cell.

Consider a system consisting of two materials with elastic parameters $\alpha^m, \beta^m, \gamma^m, \epsilon^m$ for $m = 1, 2$. Denote a cell or bond to be pure if all of its vertices are of a single material type and mixed otherwise. The following is assumed for simplicity:

- (A) For pure bonds (pure cells) in material m , $\alpha_k^p = \alpha^m, \beta^{pq} = \beta^m, \gamma^{pq} = \gamma^m$ and $\epsilon = \epsilon^m$.
- (B) For mixed bonds (mixed cells) in a two-material system, $\alpha_k^p = \frac{1}{2}(\alpha^1 + \alpha^2), \beta^{pq} = \frac{1}{2}(\beta^1 + \beta^2), \gamma^{pq} = \frac{1}{2}(\gamma^1 + \gamma^2)$ and $\epsilon = \frac{1}{2}(\epsilon^1 + \epsilon^2)$.
- (C) For a bond (cell) in which one of the vertices is in the vacuum, $\alpha_k^p = \beta^{pq} = \gamma^{pq} = 0$.

What makes the discrete elastic strain model different from continuum elasticity is the presence of corners (i.e., steps) in the discrete geometry. The total energy can be obtained by summing up all the energy density

$$\mathcal{E}^{\text{total}} = \sum_i E(i) \tag{2.16}$$

and the problem is to find the displacement field that optimizes the total energy, i.e.,

$$\mathcal{U} = \arg \min_{\mathcal{U} \in \ell^2} \mathcal{E}^{\text{total}}. \tag{2.17}$$

The force balance equation can be obtained from the variational principle, i.e.,

$$\frac{\partial \mathcal{E}^{\text{total}}}{\partial u_k} = 0, \quad \text{for } k = 1, 2. \tag{2.18}$$

In particular, at an interior point, Eq. (2.18) can be written as follows:

$$C_{11} D_k^+ D_k^- u_k + C_{44} \sum_{l \neq k, l=1,2} D_l^+ D_l^- u_k + (C_{44} + C_{12}) \sum_{l \neq k, l=1,2} D_k^0 D_l^0 u_l = 0, \tag{2.19}$$

where $k = 1, 2$. More conveniently, one may view the Eq. (2.19) as a linear system that is obtained by taking the finite difference of the following continuous linear elasticity operator:

$$\mathcal{L} = \begin{pmatrix} C_{11} \frac{\partial}{\partial x x} + C_{44} \frac{\partial}{\partial y y} & (C_{44} + C_{12}) \frac{\partial}{\partial x y} \\ (C_{44} + C_{12}) \frac{\partial}{\partial x y} & C_{44} \frac{\partial}{\partial x x} + C_{11} \frac{\partial}{\partial y y} \end{pmatrix} \tag{2.20}$$

with the stencil given as follows:

$$L^{1,1} = \begin{pmatrix} 0 & -C_{44} & 0 \\ -C_{11} & 2(C_{11} + C_{44}) & -C_{11} \\ 0 & -C_{44} & 0 \end{pmatrix},$$

$$L^{1,2} = (C_{44} + C_{12}) \begin{pmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{pmatrix},$$

$$L^{2,1} = (L^{1,2})^T \quad \text{and} \quad L^{2,2} = (L^{1,1})^T.$$

However, at corners and at material interfaces, the corresponding equations do not admit a continuum interpretation. We then conclude that the aforementioned strain model combines atomistic and continuum approaches, so that it is applicable to atomistic geometries that the continuum elasticity is insufficient to describe. This is the main difference between classical linear elasticity and the discrete strain model.

A restriction on the elastic parameters is that, for physical admissibility, the energy (density) should be positive as discussed, for example, in [17]. A sufficient condition for the positivity of the energy density can be given as follows:

$$\min_{k,p} \alpha_k^p \geq \max_{pq} \gamma^{pq} + c \quad (2.21)$$

for some positive constant $c > 0$. In fact, the condition (2.21) is a key ingredient for the well-posedness of the lattice model, [19].

The computational results reported in Sections 4 and 5 (unless otherwise stated) use material parameters that are chosen to mimic those of two atomic species, namely silicon (Si) and germanium (Ge). For those materials, the lattice misfit is 4% and the elastic coefficients (at the temperature $T = 400$ °K) are, [20], in the units of 10^{11} dyn/cm²

$$\text{Si} : \begin{cases} \alpha^1 = 3.4305, \\ \beta^1 = 1.22325, \\ \gamma^1 = 0.984, \end{cases} \quad \text{Ge} : \begin{cases} \alpha^2 = 4.38375, \\ \beta^2 = 1.51225, \\ \gamma^2 = 1.35925. \end{cases} \quad (2.22)$$

The lattice misfit on the material/vacuum interface is 3.69%. We observe that the parameters given in (2.22) are typical elastic coefficients. In particular, in most materials, α is given in a way that in general, it dominates other parameters, [20]. The incompressible limit $\alpha > \gamma \gg \beta$ could present difficulties [31]. Since this limit does not occur in the applications to epitaxial systems, it will not be addressed in this work.

3. Algebraic multigrid methods

In this section, we present an algebraic multigrid method that is suited to solve the aforementioned system of equations. A multigrid method for a given algebraic system, has two primary components, the coarse grid spaces and the smoothing operator (or local relaxation) for each level. In particular, the coarse grid spaces are used to construct a prolongation (i.e., interpolation) operator P , which maps corrections from the coarse grid, ℓ , to the fine grid, $\ell + 1$, and a restriction operator R that maps residuals from the fine grid to the coarse grid. For all methods in our work, $P = R^T$ is used. For the local relaxation or smoothing step, one can use a few (two to three) iterations of stationary iterative processes, such as the Gauss–Seidel or Jacobi methods. For the present work, we have used the point Gauss–Seidel method.

Algebraic multigrid (AMG) generally refers to multigrid methods in which coarse grid spaces and operators are constructed with little or no additional input such as the geometry of the problem or even the continuous problem itself. The classical AMG first introduced in 1980s, [6,25,34] was aimed at solving the linear system $Au = f$ arising from the discretization of scalar elliptic PDEs. Since then, there has been tremendous development of AMG methods that are designed using more sophisticated coarse grids and interpolation, (see e.g., [13] and references cited therein). It is well-known, however, that convergence of a scalar AMG method may break down when it is applied to systems. Most recently Griebel et al. [13], generalized classical AMG to systems of PDEs using the “point-block approach”, first proposed in [5,26].

Our approach was developed using the so-called “unknown approach” that was proposed (but not extensively tested) in [26]. Our method uses a scalar AMG method, the so-called agglomeration multigrid developed by Chan et al. [8], together with the further use of the energy minimizing technique for the prolongation operator due to Mandel et al. [22], that is different from the one employed in [26]. Our method works well for the strain equations with a free boundary, which was found to be a difficulty for the method of [13,26] when applied to the linear elasticity equation. Moreover, our method works for problems in which all of the boundaries are free boundaries, so that the underlying system has a null space (i.e., the set of rigid body motions), as demonstrated in Section 4. The success of our algorithm is probably due to the fact that our prolongation operator preserves the set of rigid body motions away from the boundary, because of a special coarsening procedure that results in nearly structured coarse grids for well-structured lattice domains, see Algorithms 3.1 and 3.2 below. Our method could probably be further enhanced for more unstructured problems, by constructing the prolongation operator in a way that preserves the null space of the system matrix in each coarse space using a recent technique, [37]. On the other hand, since our method performs well on the problems of interest, this is left for a possible future work.

For ease of presentation, we write the force balance equations (2.18) as the following algebraic system of equations:

$$\tilde{A}_h \tilde{U}_h = \tilde{F}_h, \tag{3.1}$$

where

$$\tilde{U}_h = \begin{pmatrix} \tilde{U}_1 \\ \tilde{U}_2 \\ \vdots \\ \tilde{U}_{n_h} \end{pmatrix} \quad \text{and} \quad \tilde{F}_h = \begin{pmatrix} \tilde{b}_1 \\ \tilde{b}_2 \\ \vdots \\ \tilde{b}_{n_h} \end{pmatrix}.$$

Here, n_h is the total number of nodes, \tilde{U}_i represents the displacement field at the node with index i . Note that in two space dimension, the strain system has two degrees of freedom in each node. Therefore, both \tilde{U}_i and \tilde{b}_i are 2×1 block vectors for $i = 1, 2, \dots, n_h$. The system of equation given in (3.1) is well suited for the AMG based on the point-block approach, e.g., [13].

The first step of the construction of our method is to rewrite the system of equations (3.1) as follows:

$$A_h U_h = F_h, \tag{3.2}$$

where

$$A_h = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \quad U_h = \begin{pmatrix} U_1 \\ U_2 \end{pmatrix} \quad \text{and} \quad F_h = \begin{pmatrix} F_1 \\ F_2 \end{pmatrix}.$$

Here U_1 and U_2 are the u and v components of the displacement fields. The matrix A_h can be simply understood as the finite difference of the operator (2.20). From the choice of elastic parameters given in (2.22), it is clear that the operators A_{11} and A_{22} behave like the finite difference of the Laplace operator.

3.1. Coarsening and prolongation

Our coarsening process and the construction of prolongation operator for the system matrix A_h given in (3.2) are mostly based on the application of the scalar AMG for each block matrix A_{11} and A_{22} . More precisely, the main idea is first to construct prolongations P_{11} and P_{22} by applying the scalar AMG for each block A_{11} and A_{22} , and then form the global prolongation operator P for A_h simply as follows:

$$P = \begin{pmatrix} P_{11} & 0 \\ 0^T & P_{22} \end{pmatrix}, \tag{3.3}$$

where 0 is a matrix consisting of zero entries. The coarse grid matrix is constructed by the usual Galerkin approach $A_H = P^T A_h P$. Therefore, the construction of the prolongation operator P is heavily dependent on the scalar AMG. In this section, we briefly describe the scalar AMG to construct $\hat{P} = P_{11}$ for the matrix $\hat{A} = A_{11}$.

The prolongation P_{22} is obtained in the same way from the matrix A_{22} and the next subsequent level prolongations can be constructed in the recursive fashion.

The scalar AMG that will be used in our algorithm is the one originally described in the paper, [8], for which the coarse grid selection relies on a combinatorial approach based on graph theory. A full presentation of the method requires a large amount of introduction and notation, which is not relevant to the purpose of this paper and much of the algorithms described in this section can be found in the papers, [1,8]. Therefore, we present only the basic descriptions without details.

To discuss and review the algorithm, it is convenient to introduce some basic definitions and notation related to graph theoretic tools, see e.g., [24]. Given a stiffness matrix $\hat{A} = (a_{ij})$, we consider the graph corresponding to it, namely $G(\hat{A}) = (V, E)$, where V is the set of vertices and E is the set of edges (ordered pairs of vertices). This graph represents the nonzero structure of the matrix in the following way: We say that the pair $(i, j) \in E \iff a_{ij} \neq 0$. Associated with the graph $G(\hat{A})$, an *independent set* (IS, for short) is a set of vertices having the following property: any two vertices in this set are independent in the sense that they are not connected by an edge. An independent set is called a *maximal independent set* (MIS for short), if addition of any vertex to the set makes it not an IS.

The algorithm described here was originally used to construct the prolongation for a system matrix obtained from a finite element discretization using piecewise linear elements. Here it is simply used like a black-box to construct the prolongation for \hat{A} . To understand how it works, it would be useful to think of our matrix \hat{A} as the stiffness matrix obtained from a finite element triangulation using piecewise linear elements. Under this interpretation, the concepts of vertex and edge agree with the underlying triangulation and the graph corresponding to the stiffness matrix \hat{A} .

In the following, we describe the algorithm for a two level implementation, since a multilevel implementation can be done in a recursive fashion.

Algorithm 3.1. (Coarse grid selection). Given a graph $G_h = (V_h, E_h)$ corresponding to the matrix \hat{A} ,

- Identify coarse grid nodes by finding an MIS, using e.g., a greedy algorithm, [21].
- For any coarse node $k \in \text{MIS}$, remove all dual edges, [8], surrounding it and form them as agglomerated elements.

The result of **Algorithm 3.1** is a coarse grid graph corresponding to the triangulation obtained by the agglomerated elements. We then construct the initial (tentative) prolongation \hat{P}_0 by the following:

Algorithm 3.2. (Initial prolongation). Given two nested graphs, $G_h(V_h, E_h)$ and $G_H(V_H, E_H)$, define the initial (tentative) prolongation \hat{P}_0 by the following:

- For a fine grid node that is also a coarse grid node, define the prolongation to be the identity.
- For a fine grid node that is interior to a macroelement, use the arithmetic average of the values at the coarse grid nodes defining the macroelement.
- For a fine grid node that lies on a single macroelement edge, define its prolongation to be the average of the two coarse grid nodes defining the macroelement edge. For a fine grid node that lies on more than one macroelement edge, define its prolongation to be the arithmetic average of all the coarse grid nodes that define the macroelement edges.

Construction of the prolongation matrix $\hat{P} = \{p_{ij}\}$ from the initial prolongation matrix \hat{P}_0 is made by solving the following optimization problem:

$$\hat{P} = \arg \min_q J(q) = \frac{1}{2} \sum_j q_{*j}^T \hat{A} q_{*j}, \quad (3.4)$$

where the q_{*j} is the j th column of the matrix q . The constraints on the matrix q are (1) the sparsity pattern of q is the same as that of \hat{P}_0 and (2) $\sum_j q_{ij} = 1$ for all nodes i , for which i and its neighbors do not satisfy a Dirichlet boundary condition. The second constraint is the requirement that the coarse space basis reproduces a constant function. Note that the problem (3.4) is in general solved using some iterative method with the initial approximation of \hat{P}_0 . This is a typical construction of a coarse grid basis function by minimization of its energy, [22].

Note that for the discrete elastic strain model, the sparsity structures for both A_{11} and A_{22} are the same, the resulting MIS is in general expected to be the same for both displacement field u and v , and it corresponds to nearly structured coarse grids in general, i.e., the point-symmetric grid at the fine grid points, [13]. In this case, rigid body motions can be preserved by the prolongation operator, at least locally in the interior. Note that the prolongation described in the Algorithm 3.2 is a linear interpolation.

Let P be the prolongation matrix constructed from the combination of each prolongation P_{11} and P_{22} for unknowns U_1 and U_2 as in (3.3) and L and G the splitting matrices of A_h , chosen for smoothing errors on fine grids. For example, for the Gauss–Seidel method, $L = D_h - L_h$, where D_h is the diagonal of A_h and $-L_h$ is the strictly lower triangular part of A_h .

Algorithm 3.3. (Basic two-level cycle). Let m_1, m_2 be the number of pre- and post-smoothings. Let U_h^ℓ be an approximation to the solution of $A_h U_h = F_h$. Then the next approximate solution $U_h^{\ell+1}$ is determined by

- Pre-smoothing

$$\bar{U}_h^\ell = U_h^\ell + L^{m_1} (F_h - A_h U_h^\ell).$$

- Coarse grid correction

- Compute the residual

$$\bar{r}_h^\ell = F_h - A_h \bar{U}_h^\ell.$$

- Restrict the residual on coarse grids

$$\bar{r}_H^\ell = P^T \bar{r}_h^\ell.$$

- Compute an accurate approximation \hat{v}_H^ℓ of the residual equation

$$A_H \hat{v}_H^\ell = \bar{r}_H^\ell, \tag{3.5}$$

- Prolongation of the correction

$$\hat{v}_h^\ell = P \hat{v}_H^\ell.$$

- Correction

$$\tilde{U}_h^\ell = \bar{U}_h^\ell + \hat{v}_h^\ell.$$

- Post-smoothing

$$U_h^{\ell+1} = \tilde{U}_h^\ell + G^{m_2} (F_h - A_h \tilde{U}_h^\ell).$$

Note that the vectors with subscript H belongs to the coarse space and the vectors with subscript h belongs to the fine grid. In the actual computation, the coarse grid problem (3.5) is solved iteratively by one recursive application of Algorithm 3.3, starting from the initial approximation for \hat{v}_H^ℓ being zero, resulting in a so-called V-cycle, which shall be called AMG-V cycle for convenience. We shall call AMG-CG, the conjugate gradient method with one iteration of AMG-V cycle as the preconditioner. The latter is a typical acceleration technique and in general, shows much better performance.

4. Numerical experiments

The purpose of this section is to demonstrate the effectiveness and robustness of the multigrid algorithm when applied to strain computations for three geometries: an infinite step train, a system of quantum dots and a layered nanocrystal. Although the computations reported here are mainly performed with the parameter set as given in (2.22), physically relevant variations of parameters in the equations do not cause any deterioration for the performance of the algorithm. Throughout this section, NC and NR denote the number of atoms in x - and y -directions, respectively.

Several remarks on the computational results are in order:

- (1) The numerical experiments are performed on one CPU node with MIPS R12000 IP35 400 MHz of SGI Origin 3200 and the operating system is Linux Red Hat 7.0.
- (2) The stopping criteria for AMG-V cycle is $\|r^\ell\|/\|r^0\| \leq 10^{-6}$, where r^ℓ is the residual vector at ℓ th iteration, and for AMG-CG, $\|Br^\ell\|/\|Br^0\| \leq 10^{-6}$, where B is the action of the preconditioner.
- (3) The relaxation method for the V cycle is chosen to be the point Gauss–Seidel method with three sweeps of pre- and post-smoothing.

In most of the experiments described below, three sweeps of (pre- and post-) smoothing in each levels are used. The performance of the algorithms, as measured by the CPU time and the number of iterations taken to achieve the prescribed tolerance, is not much different if two sweeps of (pre- and post-) smoothing are used. This is demonstrated in the numerical experiments summarized in Table 6.

4.1. Infinite step trains

We first demonstrate the robustness of our scheme on an infinite step train; i.e., an infinite periodic sequence of steps on a substrate/film system, as illustrated in Fig. 2, which shows a single period of a step train. This system includes the following four material types: vacuum, substrate, film and the interface between the vacuum and the film. Periodicity of the step train is formulated as shifted periodic boundary conditions relating the left and right sides; i.e., $u(x_\ell, y) = u(x_r, y + nh)$ for a system with n steps going up to the right, in which x_ℓ and x_r are the positions of the left and right boundaries. To be consistent with the shifted-periodic boundary conditions, n steps per period are also prescribed along the lower, artificial boundary. The atoms immediately on the lower boundary are held at zero displacement in all of the subsequent relaxation calculations, to approximate bulk behavior. This system has been thoroughly investigated in [30] to validate the strain relaxation model. Tables 1–4 list various computational results showing that the convergence property of our multigrid algorithm is not dependent on the size, shape of geometry and also the number of steps. Note that three consecutive numbers connected by ‘–’ in the right top row, four columns in each Tables 1–4 are the total number of steps in each boundaries, Boundary 1, Boundary 2 and Boundary 3 respectively. Note that period-

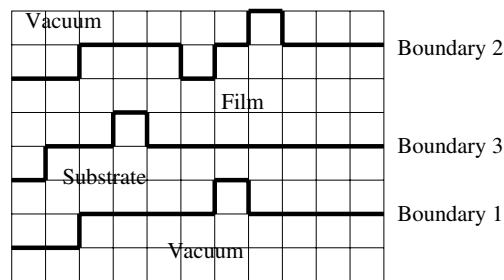


Fig. 2. Typical atomistic configuration of the infinite step train with three boundaries, Boundary 1, Boundary 2 and Boundary 3.

Table 1
AMG-V cycle and AMG-CG iteration counts in strain computations for a periodic step train in case $NC = NR$

Methods	NC × NR	1-3-1	5-7-5	11-17-11	23-35-23
AMG-V cycle	150 × 150	13	18	21	20
	200 × 200	19	17	17	28
	250 × 250	16	18	20	26
	300 × 300	15	21	20	24
AMG-CG	150 × 150	9	10	11	11
	200 × 200	9	10	10	12
	250 × 250	9	11	10	12
	300 × 300	9	11	11	12

Table 2
AMG-V cycle and AMG-CG iteration counts in strain computations for a periodic step train when $NC > NR$

Methods	$NC \times NR$	1-3-1	5-7-5	11-17-11	23-35-23
AMG-V cycle	250×100	13	15	15	17
	300×100	11	15	15	17
	350×100	12	15	17	17
	400×100	12	13	14	16
AMG-CG	250×100	9	9	9	10
	300×100	8	9	9	10
	350×100	8	9	10	10
	400×100	8	9	9	10

Table 3
AMG-V cycle and AMG-CG iteration counts in strain computations for a periodic step train when $NC < NR$

Methods	$NC \times NR$	1-3-1	5-7-5	11-17-11	23-35-23
AMG-V cycle	100×250	16	23	20	20
	100×300	13	16	22	21
	100×350	14	22	22	21
	100×400	14	19	17	20
AMG-CG	100×250	9	11	11	12
	100×300	9	10	11	12
	100×350	10	11	12	12
	100×400	9	11	12	12

Table 4
Iteration number comparison of AMG-CG and ILU(0)-CG in strain computations for a periodic step train

Methods	$NC \times NR$	1-3-1	5-7-5	11-17-11	23-35-23
AMG-CG	250×100	8	9	9	9
	300×100	8	9	9	9
	350×100	9	8	9	9
	400×100	9	9	9	9
ILU(0)-CG	250×100	201	204	202	201
	300×100	219	217	207	212
	350×100	231	232	227	225
	400×100	238	227	226	231

icity requires that the difference between the number of up steps and down steps should be the same in each interface. For example the geometry in Fig. 2 would be described as 3-5-3, since there are two up and one down steps in Boundaries 1 and 3 and three up and two down steps in Boundary 2.

In particular, our algorithm is much more efficient than the preconditioned Conjugate Gradient method with a popular ILU(0) preconditioner [29], as is shown clearly in Table 4.

4.2. Quantum dots

For a film of one material growing on top of a substrate of a different material (i.e., heteroepitaxial growth), lattice mismatch between the substrate and film leads to strain in the film. Quantum dots are nanoscale structures on the surface of the film that serve to reduce this strain [32].

A simplified and idealized quantum dot system, consisting of a substrate/film system with quantum dots on top of the film, is depicted in Fig. 3. This section provides details on the performance of the proposed multigrid methods for a quantum dot system. Further computational results of physics interest are discussed in Section 5.

As a Dirichlet boundary condition on the artificial boundary at the bottom of the substrate, the displacement is set to zero, as in the previous infinite step train case, and a periodic boundary condition is prescribed in the x -direction. The lattice sizes that are used in the computations are $NC \times NR = 100 \times 155$, $NC \times NR = 150 \times$

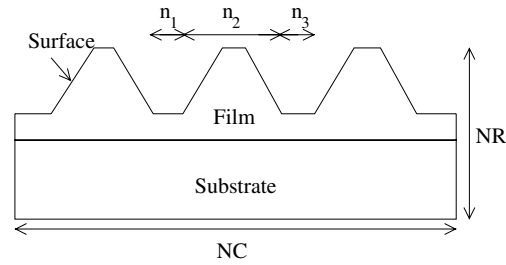


Fig. 3. Schematic drawing of typical atomistic configuration of quantum dots.

205 and $NC \times NR = 200 \times 255$. Note that as the size of the domain (i.e., NC and NR) varies, the interspacing distances between quantum dots represented by n_1 and n_3 (see Fig. 3) are also adjusted correspondingly. Table 5 shows how much memory and CPU times are necessary for SuperLU to solve the proposed problems. Our construction of the new AMG algorithms was strongly motivated by the limited applicability of SuperLU to these atomistic strain computations.

In Table 5, M1, M2 and M3 signify some particular ordering used for the SuperLU solver. More precisely, M1 indicates that the natural ordering is used, M2 is for the minimum degree ordering of the structure $A_h^T A_h$ and M3 is for the minimum degree ordering of the structure $A_h^T + A_h$. Interested readers can find further details in [10]. The symbol * denotes that SuperLU fails to solve the system of equations due to insufficient machine memory. Table 6 shows that the performance of our AMG solver is superior to that of ILU(0)-CG, an ILU preconditioned conjugate gradient method, [29], on these applications. See also Fig. 4 for a plot of the CPU time as a function of problem size for various solvers.

4.3. Layered nanocrystal

In this section, we consider a layered nanocrystal structure as illustrated in Fig. 5. Simulations of the atomistic energy density in a layered nanocrystal has significance for the structural stability of a nanocrystal, as studied for example in [2].

Table 5

The memory requirement and the CPU time for the SuperLU solver in strain computations for a quantum dot system

$NC \times NR$	100×155	150×205	200×255
Memory (MB)			
M1	155.67	455.87	*
M2	83.12	189.24	341.00
M3	61.09	152.85	268.14
CPU time (s)			
M1	28.34	111.93	*
M2	8.96	30.87	71.15
M3	6.49	26.33	56.14

Table 6

Comparison of the performance of AMG-V, AMG-CG and ILU(0)-CG in strain computations for a quantum dot system

$NC \times NR$	100×155			150×205			200×255		
	Iter	CPU	Mem	Iter	CPU	Mem	Iter	CPU	Mem
AMG-V (2)	10	2.47	12.71	11	6.02	25.44	11	11.20	42.33
AMG-CG (2)	8	2.22	13.18	7	4.71	26.38	8	9.25	43.88
AMG-V (3)	8	2.64	12.71	9	6.59	25.44	9	12.22	42.33
AMG-CG (3)	7	2.47	13.18	7	5.71	26.38	7	10.53	43.88
ILU(0)-CG	127	8.71	8.85	171	24.93	17.63	217	53.13	29.30

The number of iterations (Iter), the CPU time in seconds (CPU) and the memory in mega bytes (Mem). The number (2) and (3) in the leftmost column indicate the number of smoothings.

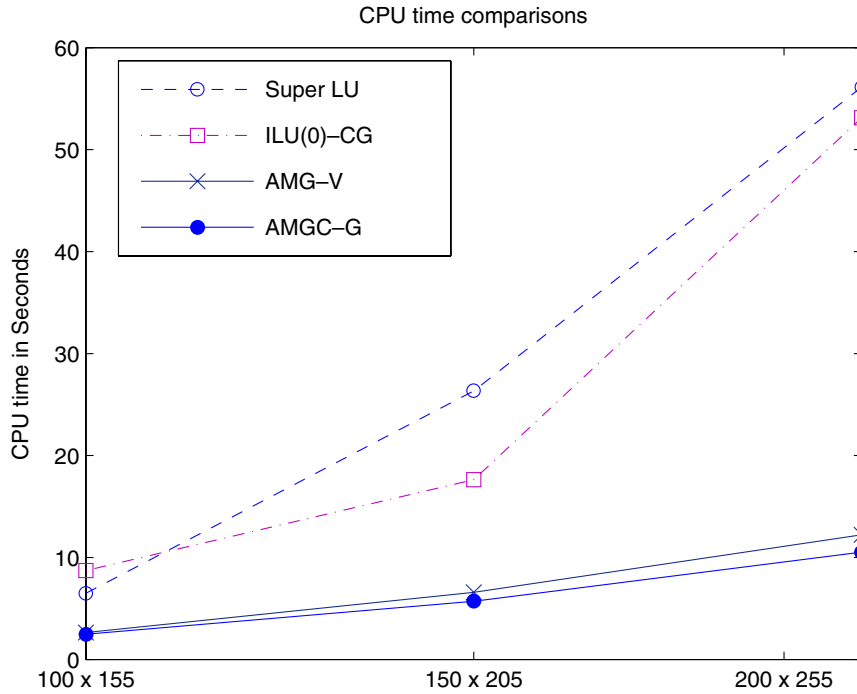


Fig. 4. CPU time in seconds as a function of the lattice size in strain computations for a quantum dot system.

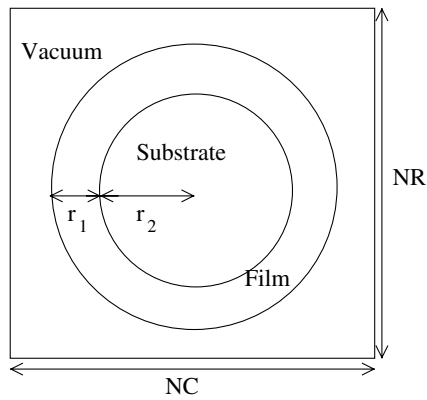


Fig. 5. Schematic drawing of the geometry of a layered nanocrystal.

Table 7
The iteration counts for AMG-V and AMG-CG for strain calculations in a layered nanocrystal configuration

	NC × NR	4:1	2:1	1:1	1:2	1:4
AMG-V cycle	200 × 200	14	15	16	16	17
	300 × 300	17	18	19	21	21
	400 × 400	19	20	21	22	23
AMG-CG	200 × 200	9	9	9	9	9
	300 × 300	11	10	10	10	11
	400 × 400	11	10	10	10	11

Typical boundary conditions for a layered nanocrystal configuration are free boundary conditions, i.e., traction free boundary conditions. It is then easy to show that the total energy is invariant with respect to rigid body motions, [2]. As mentioned before (Section 3), because of the special structure of our problem and the special caution taken in the construction of our prolongation operator, our AMG method does not seem to suffer from the possible problems due to the rigid body motion, and it exhibits optimal convergence properties as shown in the numerical experiments given in Table 7. In Table 7, the performance of AMG-V and AMG-CG for various layered nanocrystal geometries are summarized. Note that the ratio $r_1:r_2$ given in the top row from the third columns represents the ratio between the shell thickness r_1 and the radius of core r_2 , see Fig. 5.

5. Materials physics applications

In this section, we shall address some computational results for on the quantum dot geometry illustrated in Fig. 3.

5.1. Sensitivity to the thickness of the substrate

The first issue to be addressed is that of the boundary condition on the lower boundary. As described in the previous section, finite thickness of the substrate is a numerical approximation since the actual substrate is so thick as to be effectively infinite. We then immediately encounter a problem of choice of boundary condition on the lower boundary of the substrate. In general, the zero boundary condition approximates the bulk behavior well, [30]. There is a competition, however, between numerical accuracy and numerical efficiency; namely, the substrate must be chosen to be thick enough that this numerical boundary condition does not lead to the loss of numerical accuracy, but thin enough that the problem is computationally tractable. In this subsection, we investigate how thick a substrate should be used for reasonable computational results for the quantum dot geometry shown in Fig. 3.

More precisely, we study dependence on substrate thickness NR for a problem with NC = 512 with $n_1 + n_2 + n_3 = 372$. As a reference solution, we first solve the atomistic strain equations and determine both the surface energy $E_{s,\text{ref}}$ (i.e., the total sum of energy densities at points on the surface) and the total energy \mathcal{E}_{ref} for NR = 900. Gradually increasing the thickness of the substrate NR starting from 30, we compute both the surface energy E_s and the total energy \mathcal{E} for each NR and then compute the relative L^2 errors, $\|E_s - E_{s,\text{ref}}\|/\|E_{s,\text{ref}}\|$ and $\|\mathcal{E} - \mathcal{E}_{\text{ref}}\|/\|\mathcal{E}_{\text{ref}}\|$. In Fig. 6 the relative L^2 error norm is plotted as a function of the substrate thickness NR. Note that as the thickness NR increases, the errors decay rapidly, reaching about 6×10^{-6} at NR = 350. This thickness is comparable with the size of the left and right interspacing distances between quantum dots plus the base size of the quantum dot, which we believe to be the required substrate thickness for reliable computational results. It is worth remarking that such an assessment has not been reported in any literatures and also shows that for numerical reliability, one needs to introduce thicker substrates as the width of the lattice gets larger. An alternative approach using an artificial boundary condition is discussed in Section 6.

5.2. Strain energy density for a quantum dot array

Control and regularity of the size and spacing of quantum dots in an array is an essential step in their technological application, and the elastic interaction between the quantum dots has significant influence over these properties. The typical size of a quantum dot and of the interspacing between dots ranges from 20 to 100 nm, and the lattice size for Si is approximately 0.27 nm. Trenches formed at Ge/Si(100) island bases are an effective strain-relief mechanism at high growth temperatures [9,23] and are known to affect the morphological evolution of islands. Fig. 7 from [9] shows the (110) cross section of a dome grown at $T = 600^\circ\text{C}$, showing trench formation on the left base of the dot.

We compute the strain field due to an array of quantum dots, each having size and interspacing of approximately 100 lattice constants. The size of the computational domain is 512×512 lattice constants in 2-D, and the computation includes 3 quantum dots. The size of NR is chosen so that it is comparable with NC as discussed in the previous section. To demonstrate the possible difference that may occur in a random array and a regular array of quantum dots, two types of arrays are tested:

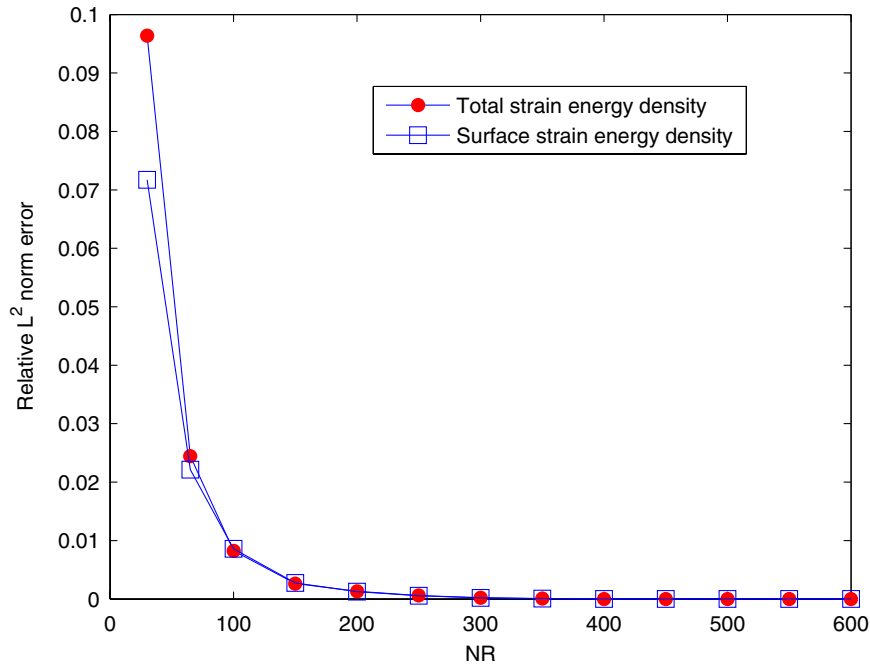


Fig. 6. Plot of the L^2 norm of the relative error as a function of the thickness NR for the surface strain energy density (circles) and the total strain energy density (squares).

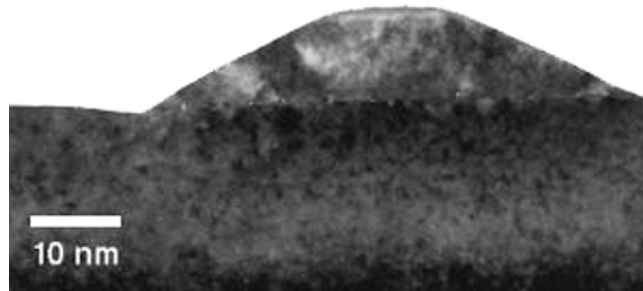


Fig. 7. TEM cross section of a dome grown at 600 C°, [9].

- (1) Regular array: $n_1 = 125$ $n_2 = 72$ $n_3 = 125$.
- (2) Random array: $n_1 = 100$ $n_2 = 72$ $n_3 = 150$.

For both cases, the maximum local energy density occurs at the base edges of the dots, as illustrated in Fig. 8, (a). Much of the substrate and pyramid tops are not visible due to the rapid decay of the energy density away from the interface.

We have also performed computations to demonstrate that trench formation is an effective strain-relief mechanism. Results discussed below are for a 160-atom-wide, 160-atom-tall substrate, a 5-atom-thick wetting layer, and 45° contact angle pyramids. Fig. 8 displays grayscale images of the elastic energy density for each configuration with trenches of various depths. In these images, darker gray corresponds to higher elastic energy. The extra mass from the trenches in geometries (b)–(d) has been attached uniformly to each terrace of the pyramids to increase the volume of the pyramids [11]. Note that as the trench depth increases, the strain energy density reduces. See also Fig. 9, which shows the total energy as a function of trench depth.

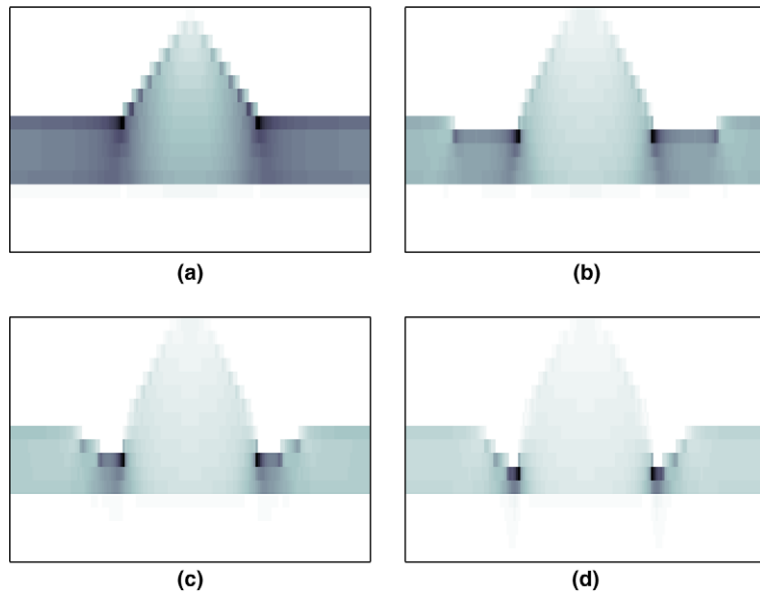


Fig. 8. Elastic energy density for 160 atom wide pyramids. (a)–(d) are for 0-, 1-, 2-, 3-atom-deep trenches, respectively. Darker gray corresponds to higher values of energy density.

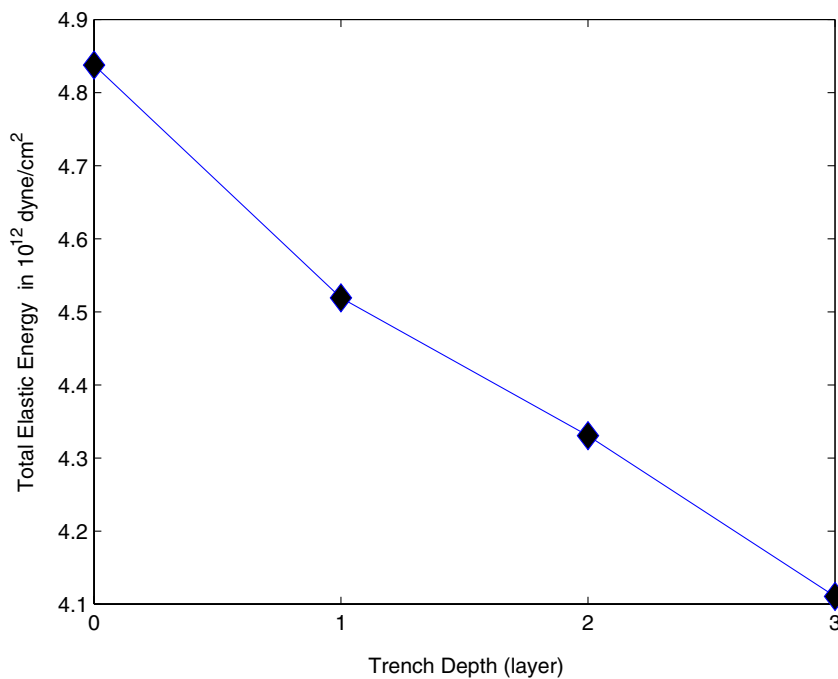


Fig. 9. Elastic energy reduction v.s. trench depth for 160 atom wide pyramids. Total energy plot corresponds to the geometries given in Fig. 8(a)–(d).

6. Conclusion and future works

In this paper, we have constructed fast and robust algebraic multigrid solvers, i.e., AMG-V cycle and AMG-CG, for the atomistic strain problem. The robustness and effectiveness of the proposed algorithms are confirmed by various numerical experiments. We have shown that both AMG-V cycle and AMG-CG

methods converge (essentially) uniformly when applied to strain problems for thin films, independent of the size of problem and the atomistic configurations. This is also true for free boundary problems such as that of a layered nanocrystal.

These results demonstrate that the AMG method is appropriate for large scale computations of strain in epitaxial thin film growth. We are in the process of applying this method to investigate strain effects for heteroepitaxial thin film growth phenomena, including Stranski–Krastanov growth, formation of quantum dots and directed self-assembly.

A Dirichlet boundary condition (i.e., zero strain) on the bottom of the substrate has been shown to be sufficiently accurate for the overall solution behavior if the thickness of substrate is comparable to the horizontal size of lattice in the computational domain. As the width of the lattice is expanded, this requires an increasingly thick substrate to obtain reliable results. In [18,28] an exact ABC at the top of the substrate has been developed to avoid this requirement. In particular it avoids the need for a thick substrate, and it retains the symmetric positive definite character of the original linear system. The price for use of this procedure is that the resultant system becomes dense, so that direct application of the algebraic multigrid is challenging. In preliminary results to be reported later, we have successfully developed a multigrid technique also for this system.

As an example of application of the AMG algorithm on a large scale numerical simulation to address a materials physics phenomenon, we showed that trench formation can provide strain relief in epitaxial growth of quantum dots.

Acknowledgements

The authors are thankful to the anonymous referees whose remarks led us to improve the presentations of the results in this paper. Research of R.E. Caflisch and Y.-J. Lee was supported in part by the MARCO Center on Functional Engineered NanoArchitectonics (FENA) and by the NSF through Grant DMS-0402276. S. Shu and Y.-X. Xiao's research was supported in part by the NSAF-10376031 of China, the Basic Research Program of China under the Grant 2005CB321702. J. Xu's research was supported by NSF through DMS-0209497 and NSF DMS-0215392 and the Furong Scholar Program of Hunan Province through Xiangtan University.

References

- [1] M.F. Adams, Parallel multigrid solvers for 3d unstructured finite element problems in large deformation elasticity and plasticity, *Int. J. Numer. Meth. Eng.* 48 (8) (2000) 1241–1262.
- [2] Y. Bae, R.E. Caflisch, Strain in layered nanocrystals, submitted for publication.
- [3] J.H. Bramble, X. Zhang, *The Analysis of Multigrid Methods Handbook of Numerical Analysis*, Vol. VII, North-Holland, Amsterdam, 2000.
- [4] A. Brandt, Multi-level adaptive solutions to boundary-value problems, *Math. Comp.* 31 (138) (1977) 333–390.
- [5] A. Brandt, Algebraic multigrid theory: the symmetric case, *Appl. Math. Comput.* 19 (1–4) (1986) 23–56, Second Copper Mountain conference on multigrid methods (Copper Mountain, Colo., 1985).
- [6] A. Brandt, S. McCormick, J. Ruge, Algebraic multigrid (AMG) for sparse matrix equations, in: *Sparsity and Its Applications* (Loughborough, 1983), Cambridge University Press, Cambridge, 1985, pp. 257–284.
- [7] M. Brezina, R. Falgout, S. MacLachlan, T. Manteuffel, S. McCormick, J. Ruge, Adaptive smoothed aggregation (αSA), *SIAM J. Sci. Comput.* 25 (6) (2004) 1896–1920 (electronic).
- [8] T.F. Chan, J. Xu, L. Zikatanov, An agglomeration multigrid method for unstructured grids, *Contempor. Math.*, AMS 218 (1998) 67–81.
- [9] S.A. Chaparro, Y. Zhang, J. Drucker, Strain relief via trench formation in Ge/Si(100) islands, *Appl. Phys. Lett.* 76 (24) (2000) 3534–3536.
- [10] James W. Demmel, Stanley C. Eisenstat, John R. Gilbert, Xiaoye S. Li, Joseph W.H. Liu, A supernodal approach to sparse partial pivoting, *SIAM J. Matrix Anal. Appl.* 20 (3) (1999) 720–755 (electronic).
- [11] U. Denker, A. Rastelli, M. Stoffel, J. Tersoff, G. Katsaros, G. Costantini, K. Kern, N.Y. Jin-Phillipp, Lateral motion of SiGe islands driven by surface-mediated alloying, *Phys. Rev. Lett.* 94 (2005) 216103.
- [12] F. Liu, J. Tersoff, M.G. Lagally, Self-organization of steps in growth of strained films on vicinal substrates, *Phys. Rev. Lett.* 80 (1998) 1268.
- [13] M. Griebel, D. Oeltz, M.A. Schweitzer, An algebraic multigrid method for linear elasticity, *SIAM J. Sci. Comput.* 25 (2) (2003) 385–407.

- [14] W. Hackbusch, *Multigrid Methods and Applications*, Springer-Verlag, Berlin, 1985.
- [15] K. Jacobi, Review: atomic structure of InAs quantum dots on GaAs, *Prog. Surf. Sci.* 71 (2003) 185–215.
- [16] C.-H. Lam, C.-K. Lee, L.M. Sander, Competing roughening mechanisms in strained heteroepitaxy: a fast kinetic Monte Carlo study, *Phys. Rev. Lett.* 89 (18) (2002) 216102.
- [17] L.D. Landau, E.M. Lifshitz, *Course of theoretical physics. Vol. 7. Pergamon Press, Oxford, third edition, 1986. Theory of elasticity*, Translated from the Russian by J.B. Sykes and W.H. Reid.
- [18] S. Lee, Artificial boundary conditions for linear elasticity and atomistic strain models, PhD thesis, UCLA, Los Angeles, June 2005.
- [19] S. Lee, R.E. Caffisch, Y.-J. Lee, Artificial boundary conditions for continuum and discrete elasticity, *SIAM. J. Appl. Math.*, in press.
- [20] D.R. Lide, *CRC Handbook of Chemistry and Physics*, 83rd ed., CRC Press, 2002.
- [21] Michael Luby, A simple parallel algorithm for the maximal independent set problem, *SIAM J. Comput.* 15 (4) (1986) 1036–1053.
- [22] J. Mandel, M. Brezina, P. Vanek, Energy optimization of algebraic multigrid bases, *Computing* 62 (3) (1999) 205–228.
- [23] Y.-W. Mo, D.E. Savage, B.S. Swartzentruber, M.G. Lagally, Kinetic pathway in Stranski–Krastanov growth of Ge on Si(001), *Phys. Rev. Lett.* 65 (8) (1990) 1020–1026.
- [24] Sergio Pissanetzky, *Sparse Matrix Technology*, Academic Press Inc. [Harcourt Brace Jovanovich Publishers], London, 1984.
- [25] J. Ruge, K. Stüben, Efficient solution of finite difference and finite element equations, in: *Multigrid Methods for Integral and Differential Equations (Bristol, 1983)* Inst. Math. Appl. Conf. Ser. New Ser., vol. 3, Oxford University Press, New York, 1985, pp. 169–212.
- [26] J.W. Ruge, K. Stüben, Algebraic multigrid, in: *Multigrid Methods* Frontiers Appl. Math., vol. 3, SIAM, Philadelphia, PA, 1987, pp. 73–130.
- [27] G. Russo, P. Smereka, A multigrid-fourier method for the computation of elastic fields with application to heteroepitaxy, preprint.
- [28] G. Russo, P. Smereka, Computation of strained epitaxial growth in three dimensions by kinetic Monte Carlo, *J. Comp. Phys.* 214 (4) (2006) 809–828.
- [29] Y. Saad, *Iterative Methods for Sparse Linear Systems*, second ed., Society for Industrial and Applied Mathematics, Philadelphia, PA, 2003.
- [30] A.C. Schindler, M.F. Gyure, G.D. Simms, D.D. Vvedensky, R.E. Caffisch, C. Connell, Erding Luo, Theory of strain relaxation in heteroepitaxial systems, *Phys. Rev. B* 67 (2003) 075316.
- [31] Joachim Schöberl, Multigrid methods for a parameter dependent problem in primal variables, *Numer. Math.* 84 (1) (1999) 97–119.
- [32] V.A. Shchukin, D. Bimberg, Spontaneous ordering of nanostructures on crystal surfaces, *Reviews of Modern Physics* 71 (4) (1999) 1125–1171.
- [33] S. Shu, J. Xu, Y. Xiao, L. Zikatanov, Algebraic multigrid methods for lattice block materials. *Recent Progress in Computational and Applied PDEs*. Edited by T.F. Chan, et.al. Boston/London, 22 (2002) 287–307.
- [34] Klaus Stüben, Algebraic multigrid (AMG): experiences and comparisons, *Appl. Math. Comput.* 13 (3–4) (1983) 419–451.
- [35] Y. Tu, J. Tersoff, Origin of apparent critical thickness for island formation in heteroepitaxy, *Phys. Rev. Lett.* 93 (21) (2004) 216101.
- [36] C.W. Oosterlee, U. Trottenberg, A. Schueller, *Multigrid with Contributions by A. Brandt, P. Oswald and K. Stüben*, Academic Press Inc., San Diego, CA, 2001.
- [37] P.S. Vassilevski, L.T. Zikatanov, Multiple vector preserving interpolation mappings in algebraic multigrid, Lawrence Livermore National Laboratory Technical Report UCRL-JRNL-208036, November 2004.
- [38] J. Xu, Iterative methods by space decomposition and subspace correction, *SIAM Rev.* 34 (1992) 581–613.
- [39] J. Xu, L. Zikatanov, The method of subspace corrections and the method of alternating projections in Hilbert space, *J. Amer. Math. Soc.* 15 (3) (2002) 573–597.