# MACROSCOPIC EFFECTS OF ATOMIC SCALE DEFECTS IN CRYSTALS: GRAIN BOUNDARY FRACTURE AND BRITTLE-DUCTILE TRANSITIONS

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Valerie R. Coffman

January 2007

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

© 2007 Valerie R. Coffman

ALL RIGHTS RESERVED

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

## BIOGRAPHICAL SKETCH

Valerie Coffman was born in 1978 in Pittsburgh, PA to Charles V. Coffman and Carole L. Grover. The signs that she would one day become a physicist appeared early on in her education. When her third grade teacher asked the class to draw a picture of a scientist, most of the students drew an Einstein-like character with disheveled white hair, a lab coat, and a beaker with fog spilling out of it. Valerie, on the other hand, turned in a picture of a woman looking remarkably like an older version of herself, with long straight hair, wearing a skirt, and peering through a microscope. When her parents read a paragraph she had written in the fifth grade that described how momentum and kinetic energy are transferred from marble to marble when they collide, her mother suspected that Valerie was indeed "going to be a physicist."

At Pittsburgh's Taylor Allderdice High School, she finally took her first physics course. She immediately felt an affinity for the subject, as no other class made sense to her quite like physics did. Now knowing what a "physicist" actually was, Valerie was soon on her way to fulfilling earlier occupational predictions. Graduating in 1996, she then began her undergraduate education at Johns Hopkins University. During summer breaks she worked at Williams College researching quantum computation, in the Johns Hopkins Sounding Rocket Program calibrating rocket borne astronomical observation equipment, and at the Space Telescope Science Institute calibrating data from the Hubble Telescope. During the academic year she tutored students in first year physics with the drop-in tutoring program at Johns Hopkins. In May 2000, she obtained a Bachelor of Science degree in Physics with a concentration in Computer Science, and continued on with graduate work in Physics at Cornell University. After earning her Ph.D. in the fall of



Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

2006, Valerie will begin a research position at the National Institute of Standards and Technology in Gaithersburg, Maryland. At NIST, she will join the Object Oriented Finite Elements (OOF) group, working on software for simulating real material microstructures. She has learned that here will no microscopes or beakers involved in her research, that dress codes will be casual, and that lab coats and disheveled white hair are optional.



Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

To my Parents, and to Max and Mimi.



Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

## ACKNOWLEDGEMENTS

First, I owe many thanks to my advisor, Jim Sethna, for all his guidance, support, enthusiasm, patience, and time. I would also like to thank Nicholas Bailey for helping me get started with *DigitalMaterial*; Anthony Ingraffea for advocating the use of atomistics in fracture mechanics and for suggesting the cube in cube problem; Gerd Heber for maintaining databases, teaching me about SQL and XML, and for much help with visualizing results; Drew Dolgert for help with getting things done in Windows; Paul Wawrzynek for the use of FEMLib; Paul Stodghill for help with getting *OFEMD* to talk to Gerd's database; Erin Iesulauro for her efforts on the cube in cube project; Andy Liu for generating the mesh for the cube in cube model and for help with running cohesive zone models; Surachute Limkumnerd for creating the OpenDX programs used to visualize the atomistic results; Connie Chang for her support and for providing LaTeXcode; Josh Waterfall for trading LaTeXtips and tricks; Seamus Davis and Tomas Arias for serving on my committee; James Slezak for sharing information about Seamus's whereabouts; Mitch Collinsworth, Ralph 'Barry' Robinson, and David Botsch for maintaining and answering many questions about the CCMR computing facilities; Douglas Milton and Deb Hatfield for keeping the LASSP and the physics department running; and finally my parents, brothers, and many friends for their support.

This work has been supported by NSF Grants No. ITR/ASP ACI0085969 and No. DMR-0218475. The computational work involved in this thesis made use of the research computing facility of the Cornell Center for Materials Research (CCMR) with support from the National Science Foundation Materials Research Science and Engineering Centers (MRSEC) program (DMR 0520404) and the resources of the Cornell Theory Center, which receives funding from Cornell University, New



Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

York State, federal agencies, foundations, and corporate partners.



Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

# TABLE OF CONTENTS





Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

## LIST OF TABLES





Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

## LIST OF FIGURES





Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHAPTER 1

# GRAIN BOUNDARY ENERGIES AND COHESIVE LAWS AS A FUNCTION OF GEOMETRY

## 1.1 Introduction

In materials such as silicon and various aluminum alloys, cracks initiate and propagate along the interfaces between polycrystals known as grain boundaries. When the cracks initiate at the site of these microscopic defects, the macroscopic fracture strength of the material is dependent on the microscopic structure of the grain boundaries. The debonding of an interface such as a grain boundary is described by a cohesive law, such as the one shown in figure 1.1.

Cohesive laws are used by finite element cohesive zone models, which simulate fracture initiation at interfaces [34]. It has been shown that the shape and scale of the cohesive law has a large effect on the outcome of the finite element simulation [34, 18]. However, the CZM studies of grain boundary fracture have used cohesive laws that are guessed, chosen for numerical convergence, and do not take into consideration the effect of varying grain boundary geometries within the material – the same cohesive law is often used throughout the material despite the fact that in a real material, grain boundaries of varying geometries must occur [29, 30, 45].

It would be useful to find a formula for the cohesive laws of the grain boundaries of a given material as a function of geometry, for input into finite element simulations. The geometry of a 3D grain boundary depends on 5 parameters that describe the orientations of the two grains. In addition, there are three different modes of fracture (normal to the crack plane, shear in a direction parallel to the



Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

crack line, or shear in a direction perpendicular to the crack line) to explore [1] as well as dependencies on temperature, impurities at the interface, and emission of dislocations to consider.

Thus far, no systematic study of grain boundary cohesive laws as a function of geometry has been done with molecular dynamics or experiment. There is difficulty in measuring cohesive laws experimentally because it is difficult to isolate and measure the displacements on either side of the grain boundary.

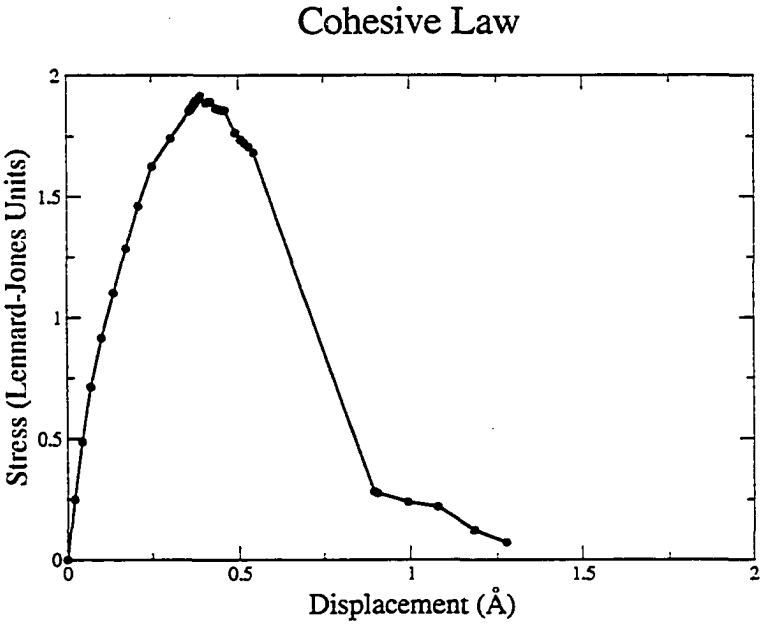



Figure 1.1: **An Example of a Cohesive Law.** The stress vs. strain curve that describes the debonding of a 2D grain boundary with tilt angles 33.418° and 26.58°. Our measurements include the elastic response of the perfect crystal on either side of the grain boundary. For input into finite element simulations, which need the cohesive law of only the interface, we must subtract off the elastic response of the bulk as shown here. This is described in detail in Chapter 2.

Previous atomistic studies of the mechanical response of grain boundaries have

[1]For cracks propagating through a grain boundary, the direction of the crack front in relation to the structure of the grain boundary may affect the fracture toughness, but this dependence is not included in cohesive laws.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

concentrated on a small number of symmetric grain boundaries in 3D [15, 39, 40, 41, 42, 45]. Exploring the complete picture of 3D grain boundary cohesive laws involves exploring a 5-dimensional space with three modes of fracture. Because of this difficulty, we have taken a step back. We seek to systematically explore the cohesive laws for mode I fracture for all possible grain boundary geometries in 2D that can be simulated in periodic boundary conditions for a particular size and strain. We will initially be focusing on symmetric grain boundaries and then expanding the picture to look at asymmetric 2D grain boundaries. The main use of such data would be for finite element simulations of polycrystals, as described above. For this purpose, it would be nice to find a functional form which describes the fracture strength of the grain boundary as a function of geometry.

We find, however, that the fracture strength as a function of tilt angles is discontinuous everywhere, with particularly large jumps at special, high symmetry grain boundaries composed of a simple arrangement of structural units with a low repeat distance. These special boundaries are also associated with cusps in the grain boundary energy. We will describe dependence of the energy and the fracture strength near these special boundaries by treating the them as perfect crystals with added dislocations.

Most studies use the coincidence site model [12] to construct and classify special grain boundaries [15, 39, 40, 41, 45] while a few mention the significance of structural units [11, 15, 31, 40, 48]. The only 2D study we are aware of, the "bubble-raft" model, observes the structural units of several of the special, high angle grain boundaries for the triangular lattice and how patterns of structural units combine to create vicinal geometries [31]. Other 3D studies also discuss how grain boundaries near special grain boundaries can be decomposed into the

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

structural unit of the special grain boundary with added flaws [11, 48]. We develop a systematic way of finding high symmetry geometries and show that the combinations of patterns of structural units at vicinal grain boundaries are key to understanding the dependence of energy and fracture strength on geometry.

Sansoz and Molinari find the grain boundary energy by allowing the grains to relax together from an initial separation of a few angstroms [40] while others perform a conjugant gradients search [15]. We use a systematic method for explicitly imposing a relative shift between the grain and using atomistic relaxation for finding the global energy minimum similar to that used in [39, 48]. It is well established that there exist cusps in the grain boundary energy energy for special high angle grain boundaries [15, 23, 36, 40, 41, 46, 48, 47]. Recent studies of grain boundary constitutive properties focus on the response to shear [40, 45] or compare shear and tension [39, 42]. Warner et. al. claim that the tensile response does not depend on the geometry of the grain boundary [45]. Others have seen jumps in the tensile fracture strength at high angle grain boundaries geometries [15, 41].

We will explain the cusps in energy and jumps in fracture strength by drawing an analogy between perfect crystals and high symmetry, high angle grain boundaries. The dislocation model of low angle grain boundaries gives a $\theta \log \theta$ form for the grain boundary energy. Because vicinal grain boundaries can be thought of as high symmetry grain boundaries with added flaws, the energy cusps at special grain boundaries have the same $\theta \log \theta$ form. Just as adding dislocations to the perfect crystal adds a nucleation point for fracture, and therefore a discontinuity (jump down) in fracture strength, adding a flaw to a high symmetry, high angle grain boundary abruptly changes the local fracture strength by adding a potential nucleation site for fracture.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

## 1.2 Grain Boundary Geometries

In 2D, grain boundary geometries are described by two tilt angles as shown in figure 1.2, but there are several other choices to make in setting up the geometry of a grain boundary simulation. We must decide which boundary conditions, in both directions, will best simulate the bulk. We must consider the effects of the length and width of the simulation. Since we will use periodic boundary conditions in the $y$-direction, we must find which geometries are possible to simulate in a periodic box, i.e. which pairs of grain orientations have finite repeat distances that are rational with respect to one another. We will find that as a consequence of using grain boundaries that have rational repeat distances, we must also find which relative shift in the $y$-direction has the lowest energy, giving the most natural grain boundary configuration.

### 1.2.1 Boundary Conditions and Length

Using free boundary conditions in the direction parallel to the grain results in edge effects, potentially causing cracks to nucleate at the edges. One alternative to free boundary conditions is to use a layer of constrained atoms. Typically, in a finite element simulation of a subsection of a material, the nodes on the surfaces of the model are constrained to not move in a direction perpendicular to the surface. This is known as "rollered" boundary conditions. We can imitate these boundary conditions in an atomistic simulation by constraining a layer of atoms at the surface or edge of the simulation to not move perpendicular to that surface or edge. This will effectively suppress the Poisson effect and has the advantage of allowing us to simulate grain boundaries of any geometry. The other choice of

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

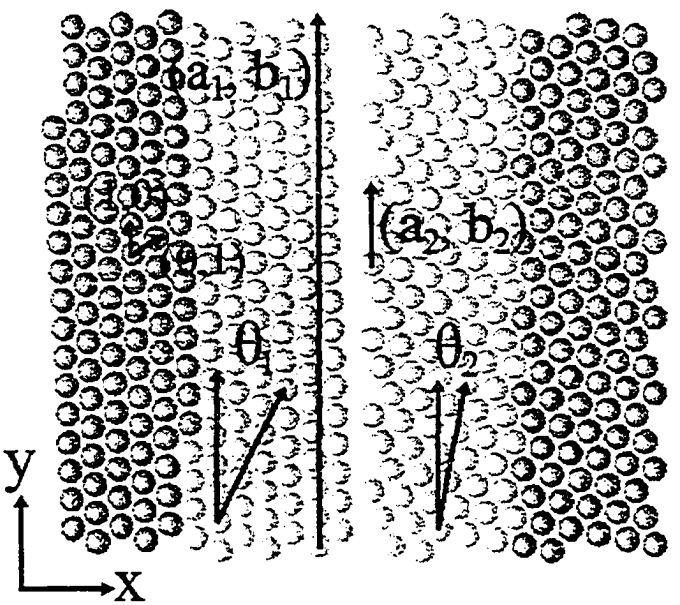



Figure 1.2: **An Example of a Grain Boundary.** The tilt angles are given by $\theta_1$ and $\theta_2$ and the repeat distances are $D_1 = |(a_1, b_1)|$ and $D_2 = |(a_2, b_2)|$. We have assumed the convention where 0° indicates an orientation with miller indices $(2, 1)$.

boundary conditions in the $y$-direction is periodic boundary conditions, which have the disadvantage of only allowing geometries that have finite repeat distances. We have compared simulations of different sizes with rollered boundary conditions in the $y$-direction to a simulation of the same geometry, but with periodic boundary conditions in the $y$-direction. The results of this comparison are shown in figure 1.3. We see that if we use rollered boundary conditions, we need a simulation that is 48 times longer in the $y$-direction in order to converge to within 2% of the results found by a simulation with periodic boundary conditions. For computational efficiency, we choose to use periodic boundary conditions in the $y$-direction. The consequences of this will be discussed in section 1.2.4. If we used periodic boundary conditions in the $x$-direction, this would create a second grain boundary. For this reason,

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

and since we need to pull the grain boundary apart in the $x$-direction, we use a constrained layer of atoms to impose fixed boundary conditions in the $x$-direction. These are represented by the darker atoms in figure 1.2. The constrained layer of atoms has a width equal to twice the cutoff distance from the potential to eliminate surface effects for the free atoms.

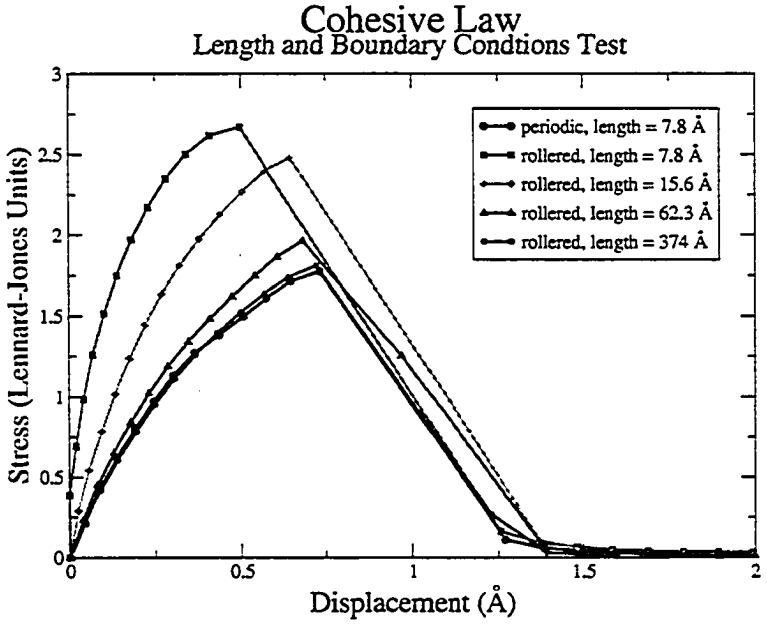



Figure 1.3: **Lengths and Boundary Conditions.** For convergence to within 2% with rollered boundary conditions, we need a system size that is 48 times the necessary system size with periodic boundary conditions.

### 1.2.2 Width and Mechanism of Fracture

The width of the simulation has consequences for the accurate measurement of the grain boundary energy as well as for the mechanism of failure. The strain field extending from the grain boundary will have a width roughly equal to the repeat distance of the boundary according to St. Venant's principle. In order to accurately measure the grain boundary energy, we will need each grain to be

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

roughly as wide as the simulation is high. Tests measuring the grain boundary energies for simulations of different widths found that the energy converged to within 1% when each grain had a width equal to the height of the grain boundary.

For simulations of a few angstroms wide, the crack opens up along the grain boundary with a slow unzipping mechanism, starting at particular flaws. For wider simulations, the crack snaps open, soon after a peak stress is reached. This is shown in figures 1.4 and 1.5. The slow unzipping of the narrow simulations is characteristic of fixed displacement boundary conditions - since the displacements of a region close to the interface is fixed, the opening of the crack is controlled, allowing us to see the details of how the crack opens. The snapping open, seen with a wider simulation, is characteristic of fixed force boundary conditions. In the wider simulations, the material on either side of the grain boundary acts as a spring, which effectively softens the boundary conditions, approximating fixed-force boundaries for long lengths. When the constrained region is far away, the interface snaps open once it reaches the maximum stress that it can sustain.

### 1.2.3 Lattice Symmetries and Tilt Angles

For the triangular lattice, it is clear that we only need to explore tilt angles between 0° and 60°, but we can further reduce the space of grain boundary geometries to consider. Figure 1.6 represents the space of tilt angle pairs. Reflecting a point in this space through the $\theta_2 = \theta_1$ line corresponds to swapping the two grain orientations which, as shown in figure 1.6, is equivalent to flipping the grain boundary in both the vertical and horizontal directions, resulting in the same grain boundary. Reflecting through the $(30, 30)$ point takes $(\theta_1, \theta_2)$ to $(60-\theta_1, 60-\theta_2)$, reversing the sense of rotation of each grain. As shown in figure 1.6, this is equivalent to flipping

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

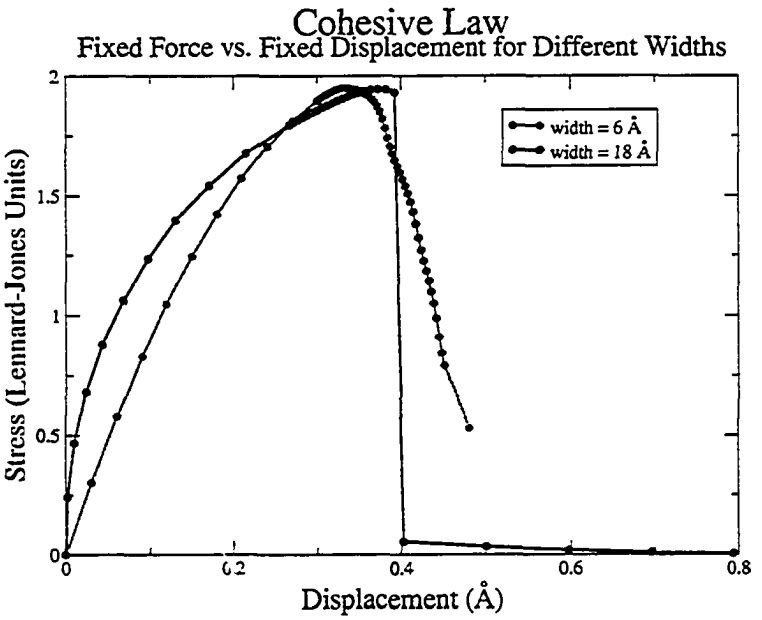



Figure 1.4: **The Effect of Width on the Cohesive Law.** For wider simulations, the grain boundary snaps open, consistent with fixed force boundary conditions. For narrower simulations, the grain boundary slowly unzips, characteristic of fixed displacement boundary conditions.

the grain boundary in the vertical direction. Reflecting through the $\theta_2 = 60 - \theta_1$ line takes $(\theta_1, \theta_2)$ to $(60 - \theta_2, 60 - \theta_1)$, both switching the grains and reversing the senses of rotation. This is equivalent to flipping the grain boundary in the horizontal direction, also shown in figure 1.6. Thus, we only need to consider the pairs of tilt angles in the triangle enclosed by the $\theta_2 = 0$ line, the $\theta_2 = \theta_1$ line, and the $\theta_2 = 60 - \theta_1$ line - the shaded region in figure 1.6.

### 1.2.4 Finding All Possible Geometries for Periodic Boundary Conditions

In order to better simulate the grain boundary in the bulk as described in section 1.2.1, we would like to use periodic boundary conditions along the direction

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

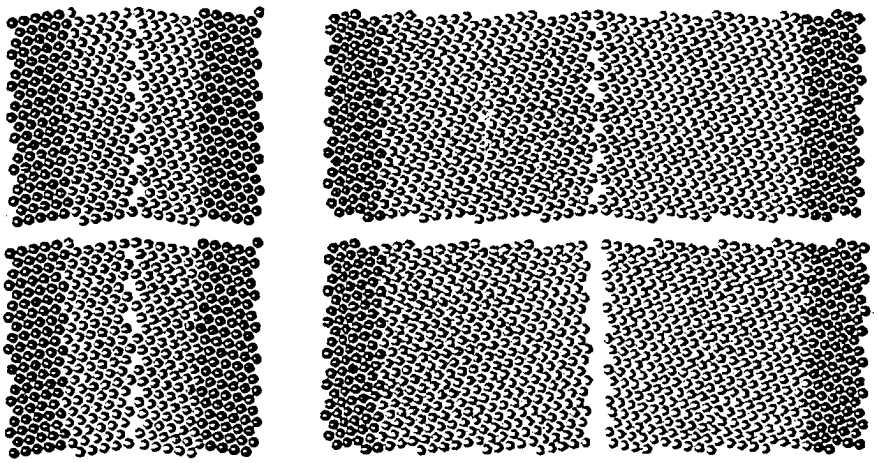


Figure 1.5: **The Effect of Width on Mechanism of Failure.** The top left figure shows the 6 Å wide simulation at its peak stress, occurring at a strain of 2.775%. The top right figure shows the 18 Å wide simulation at its peak stress, occurring at a strain of 2.925%. The bottom two figures show both simulations at a strain of 3%. For the wider simulation, the grains have snapped apart just a few steps after the peak stress, while for the narrower simulation, the grains are still slowly unzipping, several steps after the peak.

of the grain boundary ($y$-direction). In the $x$-direction, a boundary layer of fixed atoms is used to impose the boundary conditions. In order to simulate 2D grain boundaries in periodic boundary conditions, we need not only for each grain orientation to have a finite repeat distance, but also for the repeat distances to be commensurate with one another. Let $(a, b)$ be the lattice vector that is parallel to the edge of a 2D triangular lattice. Since the basis vectors are at a 60° angle to one another, the repeat distance of a particular orientation of a 2D triangular lattice is given by

$$D = \sqrt{(a + \frac{1}{2}b)^2 + (\frac{\sqrt{3}}{2}b)^2} = \sqrt{a^2 + b^2 + ab}. \tag{1.1}$$

The tilt angle is then given by

$$\theta = \sin^{-1}\left(\frac{a + b/2}{\sqrt{a^2 + b^2 + ab}}\right). \tag{1.2}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

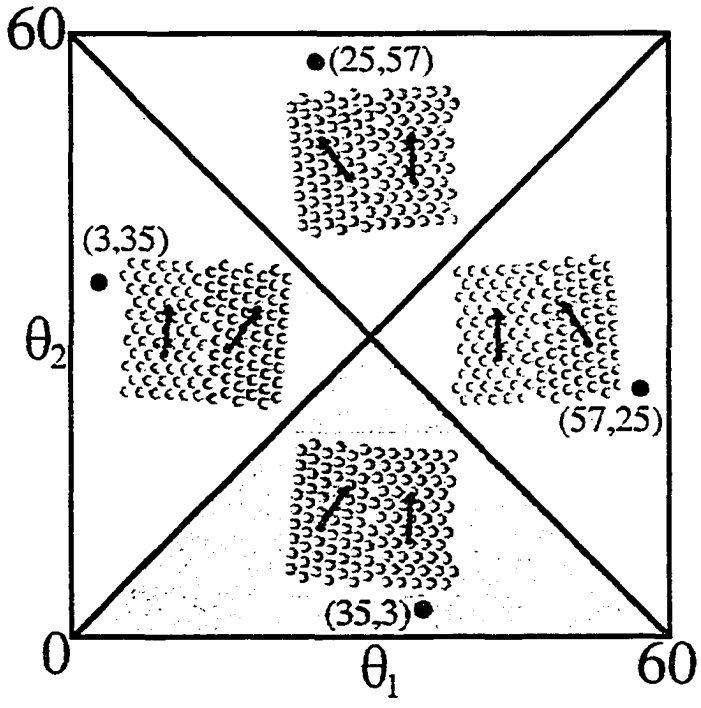



Figure 1.6: **Finding the Minimum Necessary Range of Angles.** For each point in the shaded region, there is an equivalent grain boundary in each of the other regions.

In order to expand the number of possible geometries, we have also considered geometries for which applying a small strain to each grain allows us to fit both grains inside the same periodic box. For any pair of orientations that have repeat distances that are not commensurate, we can find a continued fraction approximation to the ratios of their lengths, $p/q \approx D_1/D_2$. We can strain each grain equally into a box of size $L = (1/2)(pD_2 + qD_1)$, where the strain required is $|L - pD_2|/L$. For a strain of 0.001%, the grain boundary would have to be 50,000 Å long before the strain would alter the structure of the grain boundary. We find all pairs of tilt angles that correspond to commensurate or near commensurate grain boundaries by looping over pairs of surface vectors and comparing the repeat distances.

Figure 1.7 shows all possible geometries that can be simulated with a periodic

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

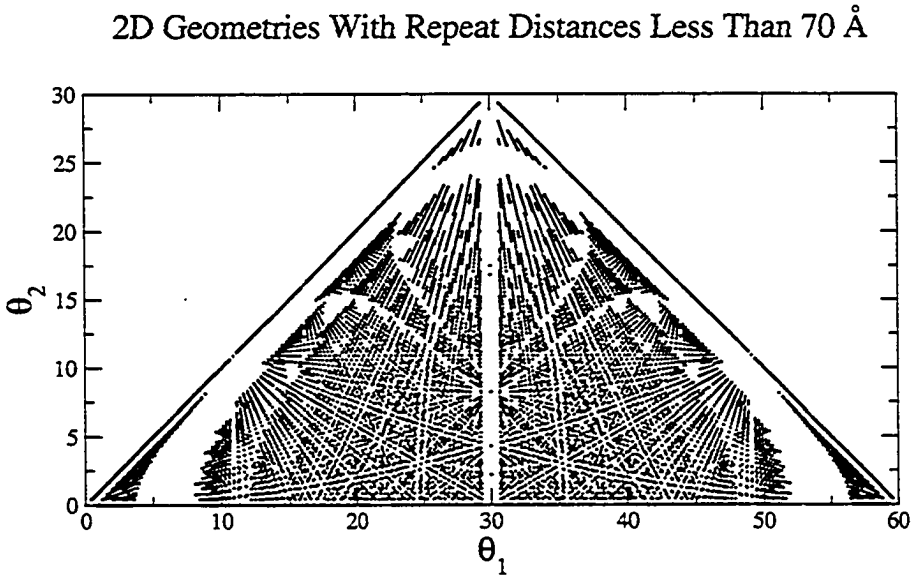



Figure 1.7: **All Possible Geometries.** The set of points above represent all 2D grain boundary geometries that can be simulated in a periodic box of 70 Å or less, with a strain of 0.001 or less. There are gaps near perfect crystals, symmetric grain boundaries, and high symmetry grain boundaries (discussed in section 1.2.6) because creating a new, nearby geometry requires adding flaws at large distances. Lines radiating from high symmetry geometries represent flaws added at closer and closer distances as you move away from the point representing the original geometry.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

length of 70 Å or less, and a strain of 0.001% or less. The $\theta_1 = \theta_2$ line corresponds to perfect crystals, while the $\theta_1 = 60 - \theta_2$ line corresponds to symmetric grain boundaries. There is a gap near each of these lines because creating very small angle grain boundaries, or a geometry very close to symmetric grain boundaries, requires adding flaws that are separated by large distances. We see gaps near other high symmetry grain boundaries for the same reason. High symmetry grain boundaries are discussed in section 1.2.6. The lines radiating out from high symmetry grain boundaries represent adding single flaws to those high symmetry grain boundaries at larger and larger distances, as you approach the high symmetry boundary. Each line represents a different type of flaw. The symmetry about the $\theta_1 = 30$ line is due to the fact that grains with tilt angles $\theta$ and $60 - \theta$ have the same repeat distance. Therefore, the grain boundaries given by the tilt angles $(\theta_1, \theta_2)$ and $(60 - \theta_1, \theta_2)$ have the same overall repeat distance, though these are different grain boundaries.

### 1.2.5 Finding the Most Natural Grain Boundary Configuration

Besides the tilt angles, there are other factors to consider in constructing the grain boundary geometries. For commensurate grain boundaries, there will be an ideal relative displacement in the direction parallel to the grain boundary (the $y$-direction in figure 1.2). This ideal displacement will correspond to the lowest energy and thus is the most natural configuration for a given pair of tilt angles. [2]

[2] The Frenkel-Kontorova model [19] describes a one dimensional chain of atoms, connected by springs, subject to an external, sinusoidal potential. If the relaxed length of the spring and the width of the potential wells are commensurate a minimum energy state exists. For the incommensurate case, there will be a pinned

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

In setting up a grain boundary geometry for measurement of the cohesive law, we must find the displacement along the boundary which gives the lowest energy. We do this by initializing the two grains with a small displacement in the $x$-direction and varying displacements in the $y$-direction, relaxing the atoms, and measuring the grain boundary energy per length. The range of displacements in the $y$-direction that we must search over is given by

$$\Delta = \begin{cases} D_1 & D_2 \% D_1 = 0 \\ \min\left(|D_2 - \lfloor D_2/D_1 \rfloor D_1|, |D_2 - \lceil D_2/D_1 \rceil D_1|\right) & D_2 \% D_1 \neq 0 \end{cases} \tag{1.3}$$

where we assume $D_1 < D_2$. This is illustrated in figure 1.8. . The grain boundary energy per length is defined as

$$E_{GB} = \frac{E_{total} - N_{atoms} * E_{bulk}}{L} \tag{1.4}$$

where $E_{total}$ is the total potential energy for the configuration of atoms (excluding the constrained atoms), $N_{atoms}$ is the number of unconstrained atoms, $E_{bulk}$ is the energy of a single atom in the bulk, and $L$ is the length of a the grain boundary. An example of the results of such a search is shown in figure 1.9. The regions with the same final displacement and energy correspond to basins of attraction around the finite number of final configurations of atoms.

We have also tried thermal annealing and have found that for certain geometries, the grain boundary migrates to form a jagged interface with segments of different grain boundaries that collectively have a lower total energy than the geometry given by the original set of tilt angles. The tendency of certain grain boundaries to corrugate is also discussed by Ishida and Pumphrey [31, 36]. As a corrugated grain boundary is torn apart to measure the cohesive law, the corners

phase with many local minima and an unpinned phase, depending on the depth of the wells in relation to the stiffness of the springs.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

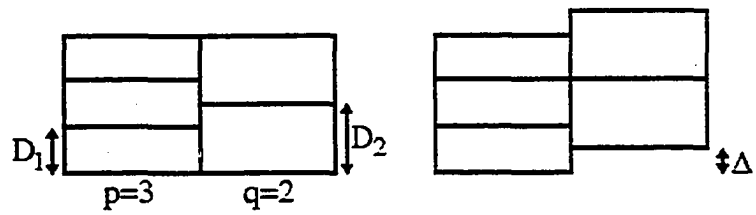



Figure 1.8: **The Necessary Search Range for $y$-displacements.** The minimum range we must search over to find the most natural configuration is equal to the minimum shift that gives an equivalent configuration of atoms along the surface. Assuming that the edges of the repeat cells line up at the bottom, the minimum shift to produce an equivalent configuration is done by lining up the top edge of the first repeat cell in the grain with a larger repeat (the grain on the right in the diagram) with the nearest edge of a repeat cell in the grain with a smaller repeat distance (the grain on the left). If $D_2$ is equal to an integer number of repeats of $D_1$, the minimum search range is equal to $D_1$. $\lfloor D_2/D_1 \rfloor$ gives the number of repeats of the left grain that fit within a single repeat of the right grain and $\lceil D_2/D_1 \rceil$ gives the number of repeats of the grain on the left that contain one repeat of the grain on the right. If $D_2/D_1$ is not an integer, the minimum search range is then $\min\left(|D_2 - \lfloor D_2/D_1 \rfloor D_1|, |D_2 - \lceil D_2/D_1 \rceil D_1|\right)$

form stress concentrations which weaken the grain boundary. Despite the fact that this configuration may be more natural, it is not what we intend to measure. We wish to measure the fracture toughness of all possible pairs of tilt angles, even if those pairs of tilt angles happen to be unstable. In other cases, the minimization procedure described above produced a curved grain boundary. In these cases, we've constrained the displacements in the $y$-direction such that a consistent pattern of flaws along a straight grain boundary is achieved.

### 1.2.6 High Symmetry Grain Boundaries

Certain grain boundary geometries have particularly low repeat distances. These grain boundaries have special properties. They mark the center of cusps in the

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

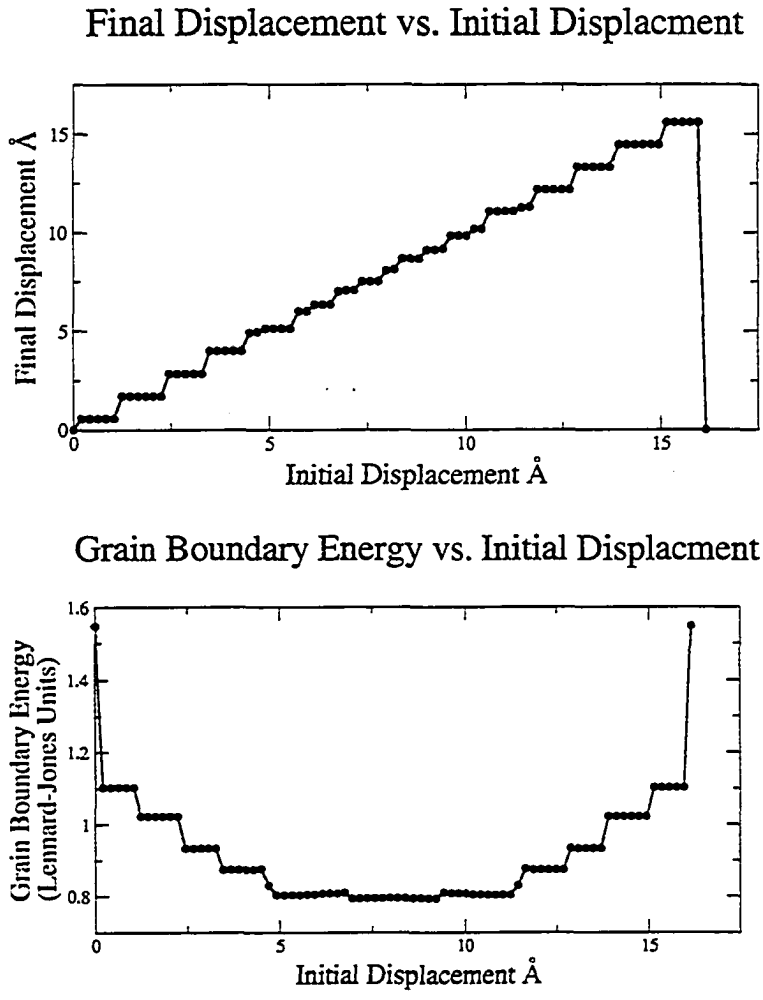
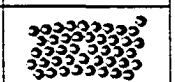
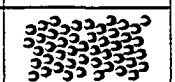
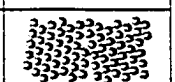
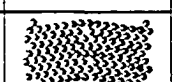



Figure 1.9: **Finding the Energy Minimizing Grain Boundary Configuration.** Each plateau in the top plot and each flat region in the bottom plot correspond to the basin of attraction of a local minimum. The most natural grain boundary configuration corresponds to the global energy minimum.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

grain boundary energy and discontinuous increases in the fracture strength as a functions of tilt angle. Table 1.1 show examples of high symmetry grain boundaries and figure 1.10 shows examples of geometries in between, which constitute adding a single flaw to the high symmetry grain boundary. Note how the added flaws constitute a compromise between the two high symmetry geometries. Furthermore, a less high symmetry grain boundary can be repeated, and have a single flaw added. Because of this the grain boundary energy and fracture strength as a function of tilt angle will have a self similar nature.

Table 1.1: **High Symmetry Grain Boundary Geometries**

| $\theta$ | Miller Indices | Repeat Distance (lattice constants) | $\Sigma$ | Structure |
|---|---|---|---|---|
| 49.10 | (1,4) | 2.64 | 7 |  |
| 43.89 | (2,5) | 3.61 | 13 |  |
| 53.41 | (1,7) | 4.35 | 19 |  |
| 40.89 | (1,2) | 4.58 | 21 |  |
| 38.94 | (4,7) | 5.56 | 31 |  |
| 51.78 | (2,11) | 7.0 | 49 |  |

The coincidence site lattice model (CSL) describes grain boundaries in terms of $\Sigma$, the inverse density of lattice sites that are shared by the two grain orientations when rotated about a common lattice point [12]. In 2D, CSL grain boundaries are necessarily commensurate and vice versa. For the triangular lattice, each common lattice point for a given pair of grain orientations has a repeat cell that is an

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

equilateral triangle with one edge defined by the surface vector with length $D$. $\Sigma$ is equal to the number of lattice points inside this cell and is given by the area of the cell divided by the area of one lattice triangle, $\Sigma = D^2$. While the CSL formulation gives a simple method for finding commensurate grain boundaries, it is misleading to suggest that the actual coincidence of sites plays a physical role. In fact, shifting the grains in the $y$-direction so as to minimize the grain boundary energy (described in section 1.2.5) generally causes the atoms to no longer coincide. Without this shift, the two free surfaces will have atomic planes that meet at the same point. The elastic energy is lowered by staggering the dislocations [27]. Wolf has pointed out that for 3D, the CSL formulation of grain boundaries involves a redundant number of parameters [46].

Bishop and Chalmers introduced the concept of "structural units" - polygonal structures of atoms along high angle grain boundaries [11]. We find that the patterns of "structural units" are more relevant to the physical properties of the grain boundaries. When relaxed, the structural units of grain boundaries in 2D, triangular lattices are 5 atoms forming a regular pentagon or a pentagon that is slightly stretched. Each high symmetry grain boundary is comprised of a simple pattern of pentagonal structural units (figure 1.1). When a high symmetry grain boundary is perturbed, an element of the pattern of structural units from the neighboring high symmetry grain boundary is introduced (figure 1.10). As the tilt angles move closer to the neighboring high symmetry grain boundary, the flaws become closer together until they outnumber the original structure. The roles of flaw and original structure are then reversed. In this manner, combinations of patterns of structural units can be used to build up any commensurate, high angle grain boundary.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

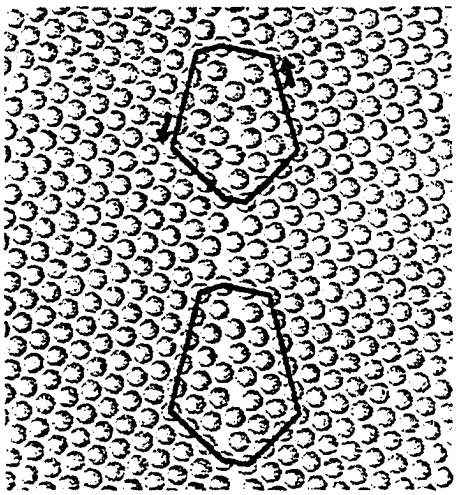


Figure 1.10: **Finding the Burger's Vector for Flaws Along Vicinal Grain Boundaries.** Grain boundaries with tilt angles in the vicinity of high symmetry grain boundaries are comprised of the pattern of structural units of the high symmetry grain boundary with an element from the neighboring high symmetry geometry. When defining the Burger's vector of a flaw added to a high symmetry grain boundary we are no longer comparing the flaw to the perfect lattice but to the repeating pattern of structural units. Therefore, we must cross the grain boundary at equivalent portions of the structural unit in our reference loop and in our loop around the flaw. The loop on the bottom is the reference loop around the region of the grain boundary without the added flaw. The loop on the top surrounds the added flaw, which we model as a grain boundary dislocation. The arrows indicate the segments needed to make the top loop match the bottom loop; for example, the arrow on the upper right indicates that the path around the defect is one atomic length shorter on that leg than the corresponding leg of the reference loop. The arrows sum to the Burger's vector of the added grain boundary dislocation. This grain boundary has a tilt angle of 44.3° and a Burger's vector with a length of 0.495 lattice constants, and a direction along the negative $x$-axis.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

We find the structural units a useful way of conceptualizing the different grain boundary geometries. We have not found the coincidence site lattice model useful in our investigations. In systematically constructing grain boundaries with small repeat distances (section 1.2.4 and appendix A), we have found the surface lattice vectors of the two sides of the grain boundary to be the most useful description. The structure of high angle grain boundaries can also be described in terms of a dislocation model. The "extra flaws" added to create vicinal grain boundaries can be described as partial dislocations. Figure 1.10 shows how to find the Burger's vectors for the added flaws by examining the flaw in the pattern of structural units. We will show in section 1.5.2 that this model gives excellent agreement in the stress field due to the added flaws, outside the background of the original flaw structure. We will show in sections 1.4 and 1.5 that the dislocation model provides the most powerful framework for understanding the geometry dependence of the properties of grain boundaries.

## 1.3 Measuring the Cohesive Law

The atomistic simulations are done using the *DigitalMaterial* package [7]. The potential we are using for 2D simulations is the Lennard Jones potential with a smooth, fourth order cutoff between 2.41 and 2.7 Å. The ground state is a triangular lattice with a lattice constant of 1.11 Å.

We set up the grain boundary by initializing two rectangular grains with the given rotations that define the grain boundary geometry we wish to measure. At either end in the $x$-direction, there is a layer of atoms with a thickness equal to two potential cut-off distances. These atoms are represented by the darker atoms in figure 1.2 and are constrained to not move. If necessary, both grains are strained

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

by equal and opposite amounts in order to have both grains fit into a periodic box.

The most natural grain boundary configuration is found using the method described in section 1.2.5. The energy found by this method is what we record as the grain boundary energy. After finding this ideal $y$-displacement, we increment the strain by displacing the constrained layer of atoms. We relax the atoms and measure the force in the $x$-direction per unit length on the constrained layer of atoms. If the measurement of the stress drops abruptly during one strain step, the simulation restores the positions from a previous step, reduces the size of the strain increment, and proceeds. The result of such a simulation is shown in figure 1.1. The maximum stress in the stress strain curve is what we define as the peak stress.

At high angle grain boundaries, the fracture is very brittle in nature. With a wide enough simulation, as discussed in section 1.2.2, the grain boundary snaps open shortly after reaching the peak stress. For lower angle grain boundaries, the failure is not always brittle. Symmetric, low angle grain boundaries with tilt angles near 0° undergo intragranular fracture, with the crack nucleating at a single dislocation and growing at an angle 30° from the grain boundary direction. Symmetric grain boundaries with tilt angles near 30° fail with alternate dislocations gliding in opposite directions unless the glide is prevented by the boundaries of the narrow simulation. In this case, the grain boundaries undergo brittle fracture with a predictable peak stress. This is discussed further in section 1.5.1

Sections 1.2.3 and 1.2.4 explain the minimum range of pairs of tilt angles necessary to explore all geometries and how to find the geometries that can be simulated in periodic boundary conditions. We have measured the grain boundary energy and peak stress for all symmetric grain boundaries with repeat distances under 20 Å. In order to explore the regions close to high symmetry grain boundaries,

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

we have added a few geometries with longer repeat distances, close to the high symmetry grain boundaries.

## 1.4 Grain Boundary Energy

The energy associated with the series of flaws that make up the grain boundary is defined in equation 1.4. As stated earlier, and found by several earlier studies [40, 15, 23, 36, 46, 35, 47], cusps appear at high symmetry boundaries. For small angle grain boundaries, the grain boundary energy has the form

$$E_{GB} = \frac{\mu b}{4\pi(1-\nu)}|\theta| \log\left(\frac{e\alpha}{2\pi|\theta|}\right) \propto |\theta| \log|\theta|. \tag{1.5}$$

where $\mu$ is the shear modulus, $b$ is the Burger's vector, $\nu$ is the Poisson ratio, and $\alpha$ is a factor that includes the core energy [27]. One can now imagine the same scenario applied to high symmetry boundaries. We can take the a high symmetry grain boundary and add or subtract flaws a distance $d$ apart as shown in figure 1.11. By the same reasoning as used for the low angle grain boundaries, the energy near the high symmetry grain boundary will have the form

$$E_{GB} = E_0 + \frac{\mu b}{4\pi(1-\nu)}|\theta - \theta_0| \log\left(\frac{e\alpha'}{2\pi|\theta - \theta_0|}\right) \tag{1.6}$$

where $E_0$ is the energy of the high symmetry grain boundary which occurs at the angle $\theta_0$ and $\alpha'$ describes the core energy of the flaw within the pattern of flaws.

The grain boundary energies for all of the symmetric grain boundaries that we have measured are shown in figure 1.12. Note that cusps occur at the angles listed in table 1.1. We are able to fit the data for symmetric grain boundaries to a function of the form

$$E_{GB}(\theta) = a_0|\sin 3\theta| \log\frac{b_0}{|\sin 3\theta|} + a_{30}|\sin 3(\theta - 30)| \log\frac{b_{30}}{|\sin 3(\theta - 30)|}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

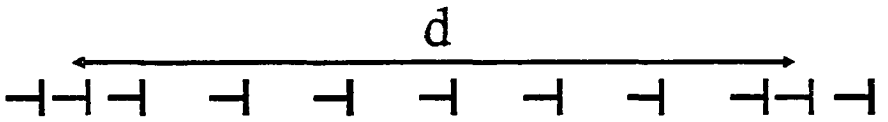



Figure 1.11: **Adding a Flaw to a High Symmetry Grain Boundary.** The lighter dislocations represent flaws added a distance $d$ apart, to an existing pattern of dislocations, shown in black, with a short repeat distance. The added flaws can also move, screen, or cancel the flaws that make up the high symmetry boundary.

$$+ \sum_{i=0}^{n} \left( a_i^{(s)} |\cos 6\theta - \cos 6\theta_i| \log \frac{b_i^{(s)}}{|\cos 6\theta - \cos 6\theta_i|} + a_i^{(a)} (\cos 6\theta - \cos 6\theta_i) \log \frac{b_i^{(a)}}{|\cos 6\theta - \cos 6\theta_i|} \right) + \sum_{j=0}^{m} c_j \cos(6j\theta) + d \tag{1.7}$$

The first two terms fit the cusps at 0° and 30°, where the cusps are symmetrical about their respective center points. The next set of terms in the sum fit the cusps at high symmetry tilt angles. The function $|\cos 6\theta - \cos 6\theta_i| \log \frac{1}{|\cos 6\theta - \cos 6\theta_i|}$ was chosen as a fitting function because it approximately gives a $\theta \log(1/\theta)$ shaped cusp in the near vicinity of $\theta_i$ and because it has the correct symmetry: even mirror symmetry at 0° and 30° and an overall period of 60°. We use one term that is antisymmetric about $\theta_i$ and one term that is symmetric about $\theta_i$ so that we can fit the shape on either side of the cusp independently. We do not expect the slope of the curve on either side of the cusp to be the same since the Burger's vectors of the additional flaws for the geometries on either side of the high symmetry geometry may differ (figure 1.10).

The $\theta_i$ can be any angles that have the shortest repeat distances, such as those given in table 1.1. The curve in figure 1.12 is the result of fitting equation 1.7 to the data shown in the same figure. We have used $\theta_i = (49.10°, 43.89°, 40.89°, 38.94°)$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

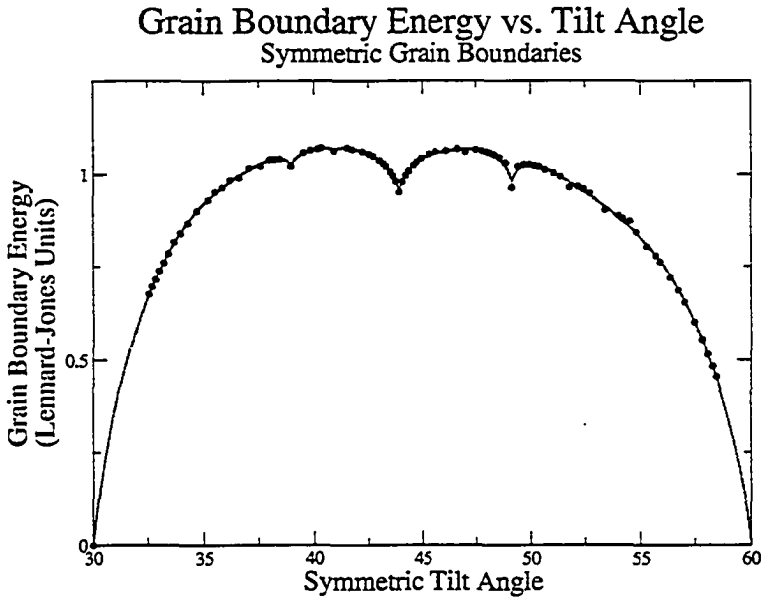



Figure 1.12: **Grain Boundary Energies for Symmetric Geometries.** Cusps appear at high symmetry grain boundaries (listed in table 1.1) and have the same $\theta \log \theta$ shape as the energy of low angle grain boundaries. The line is the fit given by equation 1.7

since these angles have particularly prominent cusps and short miller indices (given in table 1.1). Only three terms were used in the final sum over $\cos 6j\theta$.

The result is analogous to a Devil's Staircase, with a cusp singularity at each angle that corresponds to a special rational number.

## 1.5 Fracture Strength

### 1.5.1 Low Angle Grain Boundaries

The Frank conditions state that the total Burger's vector for the dislocations making up a low angle grain boundary is equal to the difference of the surface vectors that define the orientation of each grain. In order to guarantee that the grain boundary will have only one dislocation per repeat distance after it is relaxed, we must choose surface vectors that have a difference equal to a basis vector and

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table 1.2: **Coefficients for Fitting Grain Boundary Energy to Eq. 1.7.**

| $i$ | $a_i^{(s)}$ | $a_i^{(a)}$ | $b_i^{(s)}$ | $b_i^{(a)}$ |
|---|---|---|---|---|
| 1 | 0.76 | $7.4 \times 10^{-4}$ | 0.196 | 13.6 |
| 2 | 0.68 | -0.023 | 0.292 | $9.19 \times 10^{-7}$ |
| 3 | 0.22 | $-8.9 \times 10^{-3}$ | 0.104 | 4.76 |
| 4 | 0.44 | 0.039 | 0.154 | $3.93 \times 10^{-3}$ |
| | $a_0$ | $a_{30}$ | $b_0$ | $b_{30}$ |
| | 1.83 | 1.65 | 1.72 | 4.14 |
| | $c_1$ | $c_2$ | $c_3$ | $d$ |
| | -0.14 | 1.71 | -0.070 | 0.094 |

have the same repeat distance. One pair of surface vectors is $(2n+1, -n)$ and $(2n+1, -n-1)$. This gives a symmetric grain boundary with a single dislocation with tilt angles close to 0°, a Burger's vector equal to $(0, 1)$ and a repeat distance of $\sqrt{3n^2+3n+1}$. Our simulations show that such grain boundaries fail via intragranular fracture rather than intergranular fracture as shown in figure 1.13.

We have also explored low angle grain boundaries centered around the 30° lattice orientation. At exactly 30°, there is an abrupt jump up in fracture strength since the perfect crystal has no nucleation site for fracture. Our simulations find a peak stress of 4.31 (Lennard-Jones Units) for the perfect crystal.

The low angle grain boundaries near the 30° lattice orientation have surface vectors $(1, n)$ and $(n, 1)$, repeat distances $\sqrt{1+n+n^2}$, and total Burger's vector $(-1, 1)$, which splits into two flaws with Burger's vectors $(0, 1)$ and $(-1, 0)$ shown in figure 1.14. For wide enough simulations, these dislocations glide in opposite

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

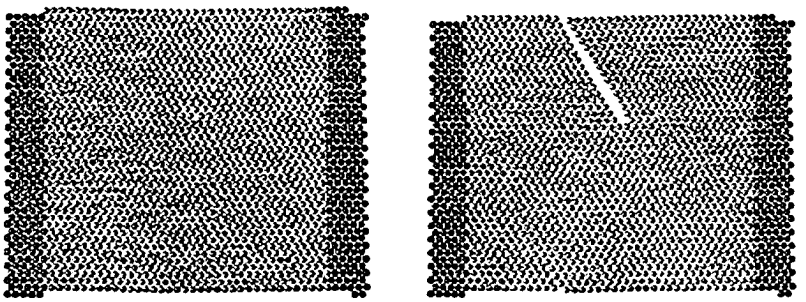


Figure 1.13: **Intragranular Fracture for Low Angle Grain Boundaries.** The figure on the left shows a grain boundary with symmetric tilt angle 0.81° at 0 strain. The same grain boundary is shown on the right with a strain of 3.125%. Symmetric low angle grain boundaries centered around the 0° orientation (miller indices (0, 1)) fail via intragranular fracture rather than intergranular fracture.

directions until they are restricted by the constrained zones on either side. For narrower simulations, the dislocations do not glide but form nucleation points for grain boundary fracture causing an abrupt jump down in the peak stress compared to the peak stress of the perfect crystal.

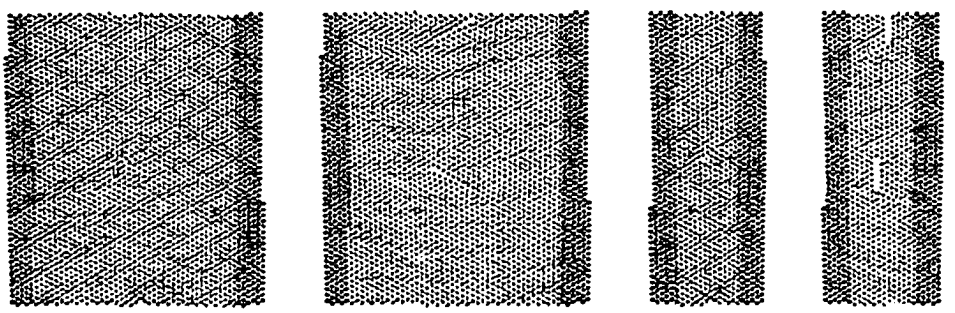


Figure 1.14: **Intergranular Fracture for Low Angle Grain Boundaries.** The first figure shows a grain boundary with symmetric tilt angle of 30.96 at 0 strain. The second figure shows the same grain boundary at 3.625% strain. In this wider simulation, the dislocations glide apart (diagonal white stripes). The third and fourth figures shows a narrower simulation of the same tilt angle where the intergranular fracture nucleates at each dislocation.

For narrow simulations of low angle grain boundaries in this region, we find that the peak stress has a parabolic dependence on angle, shown in figure 1.15.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

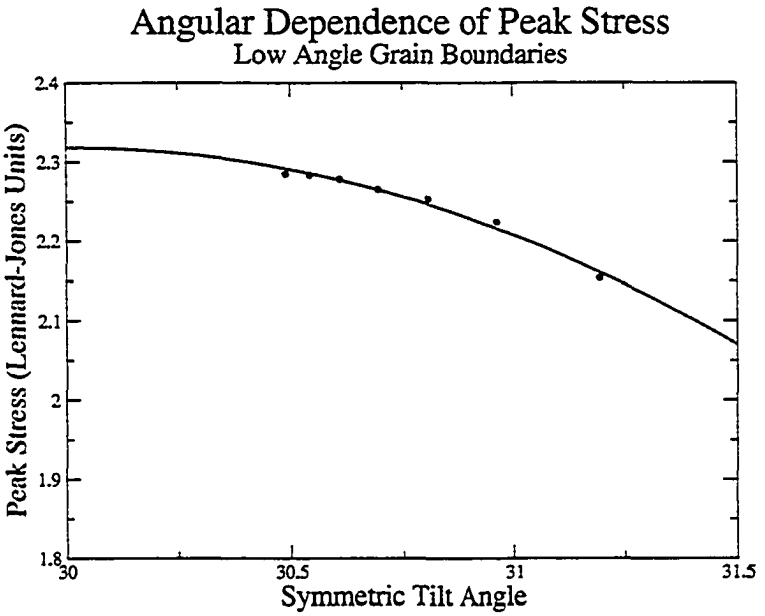



Figure 1.15: **Peak Stress vs. Tilt Angle For Low Angle Grain Boundaries.** The peak stress has a large jump downward as soon as a grain boundary deviates from $\theta = 30$ (perfect crystal). Here the peak stress for the perfect crystal, at the 30° orientation is 4.31 (Lennard-Jones units), which would require a vertical scale 11 times as big. The dislocations forming the boundary act as nucleation sites for fracture no matter how far apart they are. After this jump the peak stress has a parabolic dependence on angle for low angle grain boundaries that are constrained by width to fracture in qualitatively similar ways. Fracture nucleates exactly at the dislocation and the first non-vanishing term in the stress at this point due to neighboring dislocations goes as $1/d^2$, where d is the distance between dislocations.

We can explain this parabolic dependence by assuming that the dislocation has a critical stress for nucleating fracture equal to $\sigma_c$. The dislocation feels a stress due to its neighboring dislocations, each a distance $d$ apart, in addition to the external, applied stress. The total stress felt by each dislocation can be written

$$\sigma = \sigma_{ext} + \sum_{neigh.\ disloc.} \sum_{n=0}^{\infty} \frac{a_n}{d^n} \tag{1.8}$$

where $\sigma_{ext}$ is the external stress. The $n = 1$ term is the Volterra solution given by

$$\sigma_{xx}(x,y) = -\frac{\mu b}{2\pi(1-\nu)} \frac{y(3x^2+y^2)}{(x^2+y^2)^2} \tag{1.9}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

$$\sigma_{yy}(x,y) = \frac{\mu b}{2\pi(1-\nu)}\frac{y(x^2-y^2)}{(x^2+y^2)^2} \tag{1.10}$$

$$\sigma_{xy}(x,y) = \frac{\mu b}{2\pi(1-\nu)}\frac{x(x^2-y^2)}{(x^2+y^2)^2} \tag{1.11}$$

where the $x$-direction is the direction of the Burger's vector. Since, the Volterra solution is odd, the stress at each dislocation vanishes as we sum over the neighboring dislocations on either side. The first nonvanishing term in equation 1.8 is the $n = 2$ term which has three contributions. The $n > 1$ terms are the multiple expansions of the stress field [5] as well as nonlinear terms. The nonlinear term in strain field has the form $du/dx * du/dx$, giving a power law of $1/r^2$, which contributes to the $n = 2$ term. Geometrical restrictions cause some grain boundaries to have flaws unequally spaced in the $y$-direction, though for the results given in figure 1.15 we have only explored geometries with equally spaced flaws. The grain boundaries geometries used in figure 1.15 do have flaws that are not aligned perfectly in the $x$-direction. In each of these cases, shifting the dislocation constitutes adding a dislocation dipole (adding one positive and one negative, canceling a dislocation and adding a new one), and therefore is another contribution to the $1/r^2$ term. The external stress needed to produce a stress equal to $\sigma_c$ at each flaw is then

$$\sigma_{peak} = \sigma_c - \frac{a_2}{d^2} = \sigma_c - A(\theta - \theta_0)^2 \tag{1.12}$$

where $a_2$ combines the three contributions described above.

### 1.5.2 High Angle Grain Boundaries

Figure 1.16 shows the results of the peak stress measurements for high angle symmetric grain boundaries. At the same points for which we had cusps in energy, we have discontinuous increases in fracture strength. By drawing the same analogy between adding dislocations to perfect crystals and adding flaws to high symmetry

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

boundaries as described in figure 1.11 we can understand the discontinuities in the fracture strength at high symmetry grain boundaries and the angular dependence of fracture strength near the high symmetry geometries.

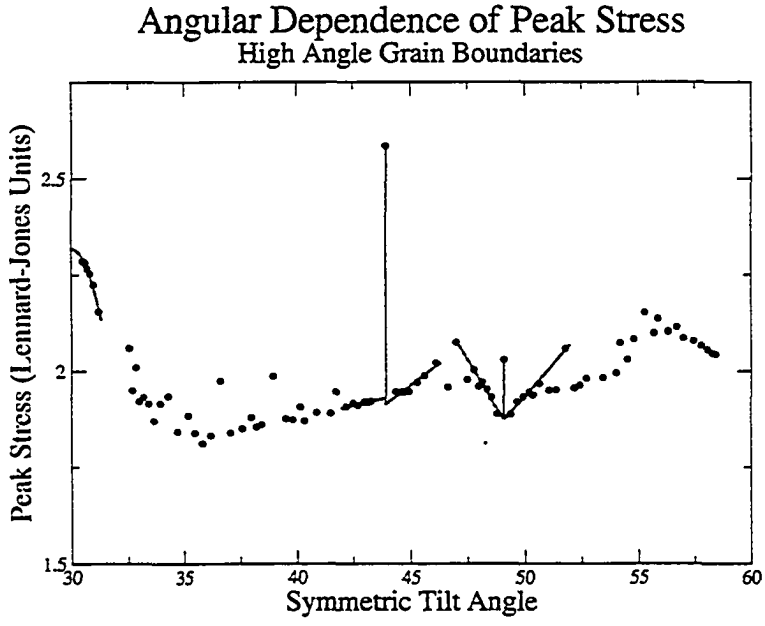



Figure 1.16: **Peak Stress vs. Tilt Angle For High Angle Grain Boundaries.** The peak stress as a function of tilt angle is discontinuous everywhere, with higher values at special tilt angles representing high symmetry grain boundary geometries. The dependence of peak stress on angle near the high symmetry grain boundaries depends on the structure of the additional flaws that make up the nearby geometries.

For high angle grain boundaries, the added flaw is no longer the sole nucleation site for fracture and fracture does not necessarily nucleate in the core of the added flaw. The added dislocation creates a stress field given roughly by the Volterra solution (equation 1.11) with a positive stress on one side, negative stress on the other, and a singularity at the center shown in figure 1.17. The stress field differs slightly from the Volterra solution because the elastic constants of the material at the grain boundary vary from those of the perfect crystal. The fracture nucleates along the boundary in the region where the stress due to the added flaw is

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

positive. Because fracture does not nucleate at the center of the added flaw, the Volterra solution as summed over the neighboring, added flaws does not cancel at the nucleation site. This leads to a linear law for fracture strength as a function of tilt angle, for grain boundaries near high symmetry geometries.

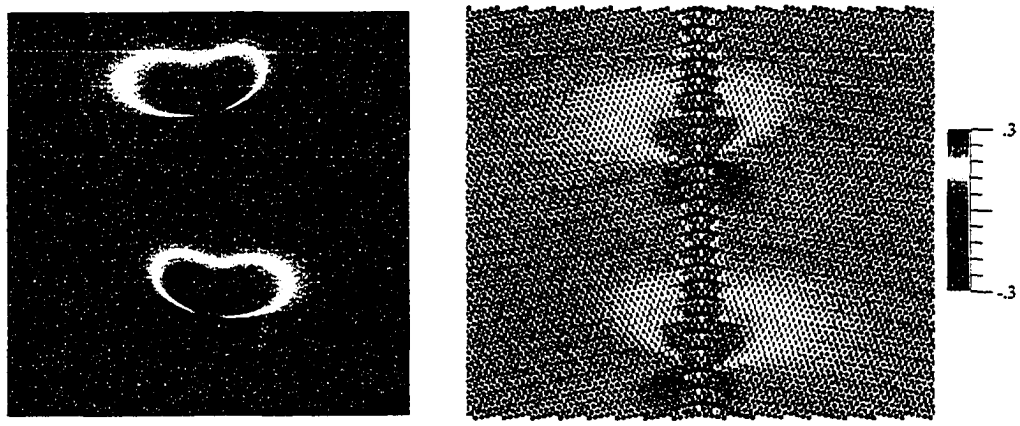



Figure 1.17: **Stress Fields Due to Dislocations.** The figure on the left shows the stress fields surrounding two dislocations according to formula 1.11. The figure on the right shows the stress fields surrounding the added dislocations as calculated according to the virial definition of atomic stress [2].

Consider the grain boundaries with tilt angles ranging from 49.39° to 53.41°, which are close to the high symmetry grain boundary at 49.11°. [show example - describe in caption how to find Burger's vector - refer to plot with peak stress results]. The additional flaws that characterize these grain boundaries have Burger's vectors equal to $(-\sqrt{3}\sin\theta, -\sin\theta)$ and $(-\sqrt{3}\sin\theta, \sin\theta)$, where $\theta$ is the symmetric tilt angle of the lattice. The norms are $2|\sin\theta|$ and the angles are 210 and 150°. For these geometries, the $xx$ component of the stress field (due to two dislocations a distance $D \approx b/2(\theta_0 - \theta)$) along the $y$-axis is

$$\sigma_{xx}(y) = \frac{\mu(2\sqrt{3}y(\theta_0-\theta) - 3\sin(\theta_0-\theta))\sin(\theta_0-\theta)}{2\pi(1-\nu)(y(\theta_0-\theta) - \sqrt{3}\sin(\theta_0-\theta))} \quad (1.13)$$

$$\approx \frac{\mu(3-2\sqrt{3}y)(\theta_0-\theta)}{2\pi y(1-\nu)(y-\sqrt{3})} + O((\theta_0-\theta))^3 \quad (1.14)$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

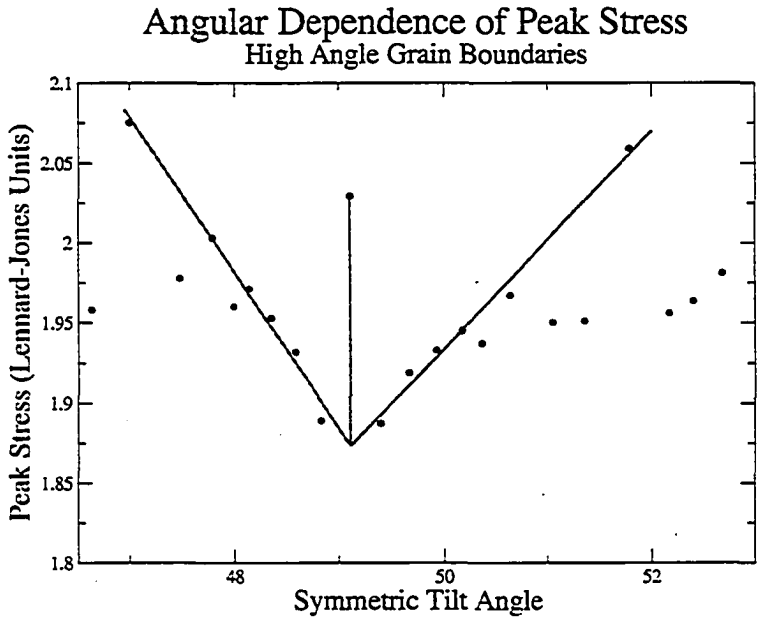



Figure 1.18: **Peak Stress vs. Tilt Angle Near High Symmetry Grain Boundaries.** There is a discontinuity in peak stress at tilt angles close to high symmetry grain boundaries. The plot above shows the peak stress for the grain boundary with tilt angle 49.1 (described in table 1.1) and the nearby geometries.

where $\theta_0$ is the tilt angle of the high symmetry grain boundary. We need to look at simulations in the fixed displacement (narrow width) regime in order to observe where fracture nucleates. We find that for geometries with this pattern of flaws, fracture nucleates at the same distance above the added flaw. The external stress needed to nucleate fracture at a distance y from the added flaw, along the grain boundary is then approximately

$$\sigma_{peak} = \sigma_c - \frac{\mu(3 - 2\sqrt{3}y)(\theta_0 - \theta)}{2\pi y(1-\nu)(y - \sqrt{3})} \tag{1.15}$$

which explains the linear dependence on angle shown in figure 1.18.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHAPTER 2

# MULTISCALE MODELING OF FRACTURE

## 2.1 Introduction

As stated in chapter 1, one of the motivations for finding the cohesive laws for grain boundaries is for input into a finite element, polycrystal model. In this chapter, we will compare a finite element, cohesive zone model (CZM) with atomistically generated cohesive laws to a fully atomistic simulation of the same geometry. We will compare the stress fields of each model and where fracture initiates. The model we will investigate is that of a cube embedded in a boundary that bisects a larger cube (Figure 2.1). The motivation for this model is that of a particle embedded in a grain boundary which is the site of fracture initiation in certain aluminum alloys.

### 2.1.1 The Cohesive Zone Model (CZM)

The cohesive zone model consists of a finite element model with zero volume interface elements placed in between the regular finite elements at interfaces. An example of an interface element is shown in figure 2.2a. These interface elements simulate fracture by debonding according to a cohesive law, the relation between the traction and displacement across the interface. The form of cohesive law used here is the piecewise linear form developed by Tvergaard and Hutchinson [44] also described by Gullerud et. al. [21]. An example is shown in figure 2.2b. The piecewise linear form of the cohesive law is determined by the initial stiffness $k_0$, the peak traction $\tau_p$, and the critical displacement, $\delta_c$ at which the surface is considered fully debonded and traction free.



Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

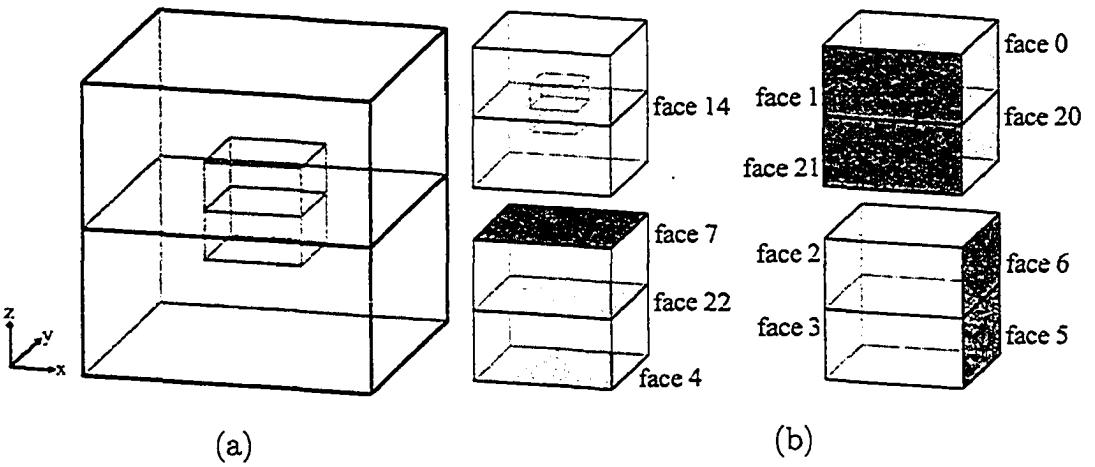



Figure 2.1: **Schematic Diagram of the Cube In Cube Model.** Figure (a) shows a schematic diagram of the cube in cube model. The inner cube is centered within the outer cube and has a length equal to 1/3 that of the outer cube. Figure (b) shows the numbering of the internal faces of the model. The upper-left figure shows the entire cube in cube model, while the rest show only the inner cube. In our simulation, we load the upper face of the model in the $z$-direction. Under such loading, faces 4, 7, 14, and 22 are subject to pure normal traction. Faces 0, 1, 2, 3, 5, 6, 20, and 21, are subject to pure shear traction. (The numbering of the internal faces is not contiguous because the FEM simulation also numbers the ten external faces.) The inner cube is a single crystal, but in order to allow for intragranular fracture through this crystal, we add an internal face through the center. The constitutive relation for this interface is that of a perfect crystal. Notice that pairs 0&1, 2&6, 3&5, and 20&21 are boundaries that macroscopically have identical cohesive laws since they are related by an inversion, i.e. they constitute symmetric pairs of interfaces for which the grains have been swapped.

Camacho and Ortiz [13] describe mixed loading by assigning different weights to the tangential and normal components of displacement, described by a factor $\beta$. We also assume that the resistance relative to tangential displacements is considered to be independent of direction. This leads to an effective, displacement of

$$\delta = \sqrt{\delta_n^2 + \beta\delta_t^2} \tag{2.1}$$

where $\delta_n$ is the normal displacement and $\delta_t$ is the tangential displacement. The

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

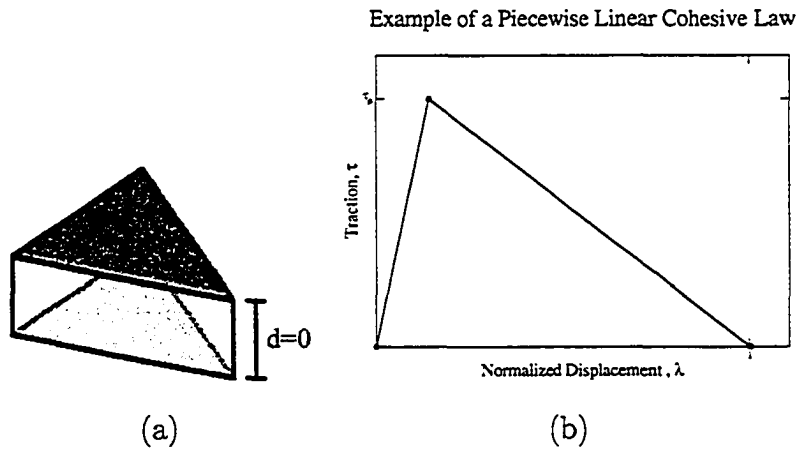



Figure 2.2: **Interface Elements and the Piecewise Linear Cohesive Law.** Figure a shows a schematic diagram of an interface element. The displacement across the interface $\delta$, is initially zero. Each triangle is part of a tetrahedral element in the material on either side of the interface. Figure b shows the form of the constitutive relation for the interface elements. The slope of the first linear segment is the initial stiffness, $k_0$. When the traction across the interface reaches the peak traction, $\tau_p$, the interface element begins to soften. When the normalized displacement, defined by $\lambda = \delta/\delta_c$ reaches a value of 1, the interface has fully debonded.

effective traction is

$$\tau = \sqrt{\tau_n^2 + \beta^{-2}\tau_t^2} \tag{2.2}$$

where $\tau_n$ is the normal component of traction and $\tau_t$ is the tangential component of traction.

## 2.2 Atomistically Determined Material Properties Used by CZM

The parameters needed by the CZM simulation that are determined by atomistics are the elastic constants associated with the atomic potential, the orientation of the lattice in each grain, and the cohesive law of each interface. We are modeling

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

silicon using the Stilling-Weber potential [43], which includes a parameter that tunes the brittleness of the potential. A value of 1.0 is standard, a value of 2.0 makes the material more brittle. We calculate separate material properties (elastic constants and cohesive laws) for each version of Stillinger-Weber.

### 2.2.1 Determining the Elastic Constants

In order to make a direct comparison between the atomistic simulation and FEM, we must determine the elastic constants of each version of Stillinger-Weber for input into the FEM simulation. The elastic constants are measured by initializing a cube of atoms in a diamond lattice, incrementing a strain in one direction and measuring the stress tensor at each increment.

For an interatomic potential with 3-body terms of the form

$$E_i = \sum_{j<k} f(\vec{r_{ij}}, \vec{r_{ik}}), \tag{2.3}$$

we can find the $\alpha\beta$ component of stress at atom $i$ by utilizing the relation [27]

$$(\sigma_i)_{\alpha,\beta} = \frac{1}{V}\frac{\partial E_i}{\partial \epsilon_{\alpha,\beta}} \tag{2.4}$$

where $V$ is the volume per atom. This leads to [6]

$$(\sigma_i)_{\alpha,\beta} = \frac{1}{V}\sum_{j<k}\frac{\partial E_i}{\partial \vec{r_{ij}}}\cdot\frac{\partial \vec{r_{ij}}}{\partial \epsilon_{\alpha,\beta}} + \frac{\partial E_i}{\partial \vec{r_{ik}}}\cdot\frac{\partial \vec{r_{ik}}}{\partial \epsilon_{\alpha,\beta}}. \tag{2.5}$$

Because

$$\partial(r_{ij})_\gamma/\partial\epsilon_{\alpha,\beta} = (r_{ij})_\beta \delta_{\alpha,\gamma}, \tag{2.6}$$

the atomic stress is

$$= \frac{1}{V}\sum_{j<k}\frac{\partial E_i}{\partial (r_{ij})_\alpha}(r_{ij})_\beta + \frac{\partial E_i}{\partial (r_{ik})_\alpha}(r_{ik})_\beta. \tag{2.7}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

We use a value of $V$ equal to the volume per atom in the ground state (perfect lattice). This has the shortcoming that for atoms near dislocations or grain boundaries, the volume per atom will be quite different.

$C_{11}$, $C_{12}$, and $C_{44}$ are determined by $\sigma_{xx}/\epsilon_{xx}$, $\sigma_{xx}/\epsilon_{yy}$, and $\sigma_{xy}/\epsilon_{xy}$ respectively. The results are given in table 2.1

Table 2.1: **Elastic Constants of Stillinger-Weber Silicon**

| | brittle factor = 1.0 | brittle factor = 2.0 | experiment [16] |
|---|---|---|---|
| $C_{11}$ | 69.74 GPa | 92.78 GPa | 166 GPa |
| $C_{12}$ | 35.20 GPa | 23.69 GPa | 64 GPa |
| $C_{44}$ | 52.00 GPa | 83.37 GPa | 80 GPa |

### 2.2.2 Measuring the Cohesive Laws

The method for measuring the cohesive law of a grain boundary with an atomistic simulation is described in section 1.3. Here we are measuring fully 3D grain boundary geometries. In order to simulate grain boundaries of any geometry (not geometries restricted by commensurability), we use rollered boundary conditions all around instead of periodic boundary conditions. Each grain is 30 Å wide and pulled apart with a strain increment of 0.5%. An example of the stress-strain curve that results from such a simulation is given in Figure 2.3.

The CZM uses a traction displacement law that describes the debonding at the interface in question [49, 13, 44] as discussed in section 2.1.1. The piecewise linear form is determined by the initial stiffness $k_0$, the peak traction $\tau_p$, and the final displacement $\delta_c$. We will need to extract these parameters from the output of our

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

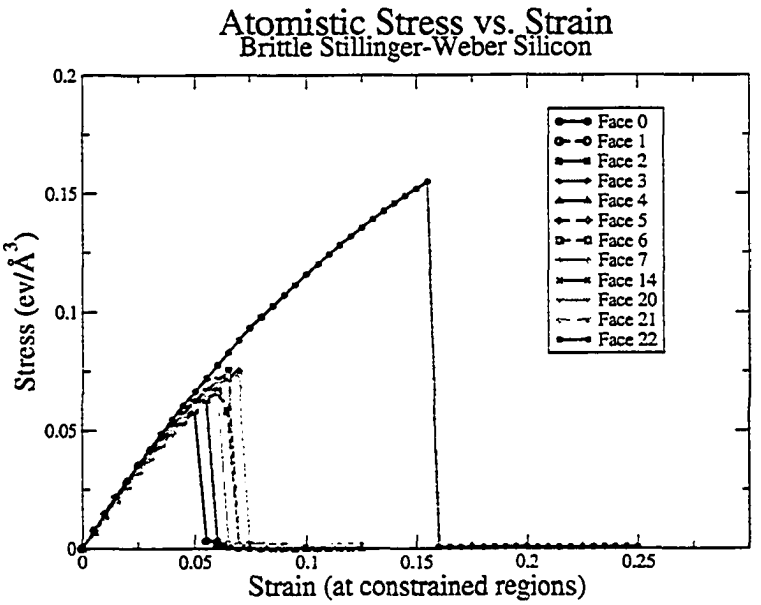



Figure 2.3: **Strain versus Stress: Brittle Stillinger–Weber Silicon.** Strain versus stress for the twelve different interfaces needed for the continuum FEM cube-in-cube simulation. Each grain is 30Å on a side. Notice that the vertical interface cohesive law is invariant under inversion, so some pairs of faces would have identical cohesive laws if measured in an infinite-sized system (or one with periodic boundary conditions, with micro-parameters completely optimized). Thus the differences between faces 0&1, 2&6, 3&5, and 20&21, both here and in Figures 2.4,2.5, and 2.6 reflect the discreteness effects of the choice of lattice origin and positions of the edges of the simulation.

grain boundary simulation (figure 2.3).

Because we are measuring the displacements 30 Å from the actual boundary, we need to subtract off the elastic response of the grain. The elastic constant describing strain normal to the grain boundary is found by rotating the elastic constants found in section 2.2.1 by the same rotation matrix that describes the rotation of the lattice vectors in each grain

$$C'_{1111} = R_{1i}R_{1j}R_{1k}R_{1l}C_{ijkl}. \tag{2.8}$$

We must then combine $C'_{1111}$ from each grain such that the stress in each grain is

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

equal (analogous to springs in series)

$$\sigma = C^{(1)}_{1111}\frac{d_1}{W} = C^{(2)}_{1111}\frac{d_2}{W} = C^{eff}_{1111}\frac{d_1+d_2}{2W} \tag{2.9}$$

$$C^{eff}_{1111} = 2/(\frac{1}{C^{(1)}_{1111}} + \frac{1}{C^{(2)}_{1111}}) \tag{2.10}$$

where $d_1$ and $d_2$ refer to the displacement in each grain and $W$ is the width of each grain. The displacement near the grain boundary is then given by

$$d_{gb} = 2W\epsilon - \frac{\sigma}{C^{eff}_{1111}}2(W - W_{gb}) \tag{2.11}$$

where $W_{gb}$ represents a finite width associated with the interface. Since the grain boundary is more stiff than the perfect crystal for the silicon geometries we have studied, this finite width is necessary so that equation 2.11 does not give a negative value. The initial stiffness is then given by the peak stress divided by the displacement at peak stress. The final displacement is set such that the Griffiths criterion is met i.e. such that the area under the curve is equal to the difference between the final surface energies of the broken grain and the initial energy of the grain boundary interface, $\lambda_c = 2(\gamma - \gamma_{gb})/\sigma_c$.

For the perfect crystal, we can simply scale the cohesive law to a width equal to the finite width used to process the grain boundary cohesive laws since we do not need to separate the behavior of the bulk from the behavior of an interface. This has the effect of preserving the non-linear elastic response. The non-linear elastic response of the bulk is not separated from the response of the interface for the case of grain boundaries, since the elastic response of the bulk that we subtract off is assumed to be linear.

In principle, two boundaries for which the grains have been swapped (such as faces 0&1, 2&6, 3&5, 20&21 as shown in figure 2.1b) should have the same overall structure and therefore have the same cohesive law. In practice, when simulating

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

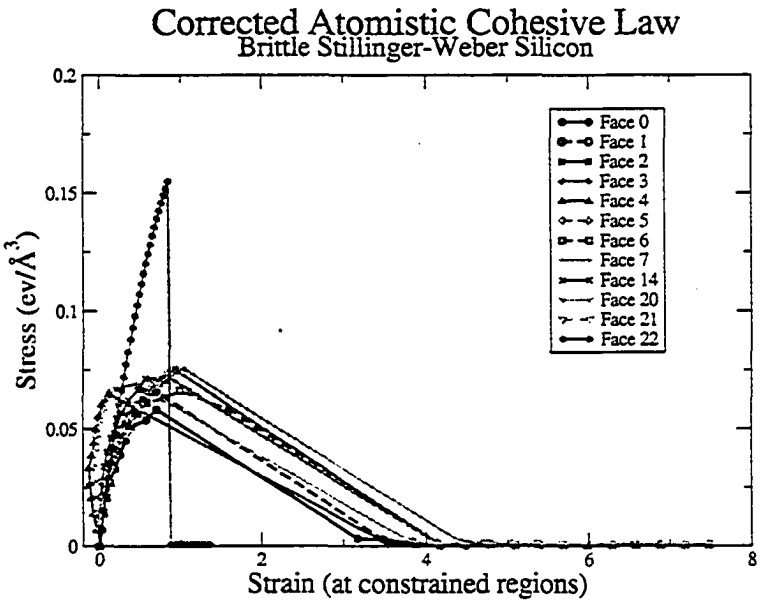



Figure 2.4: **Cohesive Law: Brittle Stillinger–Weber Silicon.** Cohesive law, displacement versus stress, for the brittle potential and twelve interfaces of figure 2.3. The transformation from strain to effective displacement at the interface is as described in section 2.2.2. The effective thickness of the interface is 9Å on each side. (Note that this is comparable to the entire size of the smaller MD cube-in-cube simulations.)

a finite region of a grain boundary, microparameters (the choice of section of the interface, the translations of the grains relative to one another, and the cutting plane of each lattice orientation) alter the grain boundaries which would otherwise be the same by symmetry. The differences between the cohesive laws for the pairs 0&1, 2&6, 3&5, 20&21 in figures 2.5 and 2.6 indicate the scope of this effect.

## 2.3 Fully Atomistic Model

The fully atomistic model is run with a software package called Overlapping Finite Elements and Molecular Dynamics (OFEMD) which is described in detail in chapter 3). OFEMD uses the *DigitalMaterial* [7] library and FemLib to run atomistic simulations of any geometry within a finite element mesh. The geometry and mesh

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

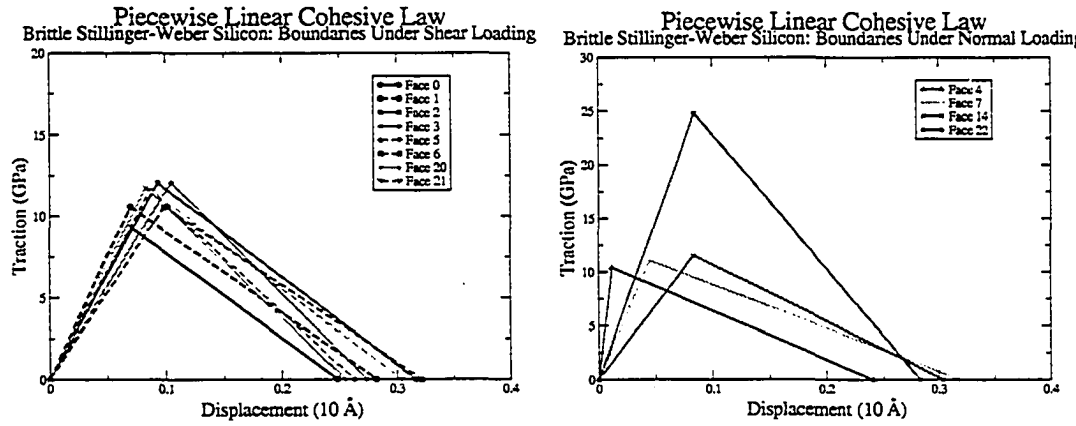



Figure 2.5: **Piecewise Linear Cohesive laws: Brittle Stillinger–Weber Silicon.** Simplified piece-wise linear cohesive law used in the FEM simulations. The peak stress and its corresponding displacement were taken from 2.4, and the critical displacement where the force vanishes is chosen to make the area under the curve equal the Griffiths energy.

information of the model is stored in a relational database [24, 25, 26]. OFEMD retrieves the geometry and mesh information by sending XQueries to a webservice that returns the information in xml format. Using FemLib, OFEMD parses this xml and stores the and geometry information in python objects.

OFEMD then loops over each material region in the mesh, finds a bounding box for each, initializes a rectangular cluster of atoms with the given orientation to fill this bounding box, removes the atoms that are not within the material region, and then sets up constrained regions of atoms that are used to enforce the rollered boundary conditions. The constrained zones are found by detecting which atoms are within two cutoff distances of any outer face of the model. Because Stilling-Weber contains three body terms, two cutoff distances are needed to ensure that the free atoms are not subject to surface effects. There is a constrained zone for each face (atoms that are within a constrained zone width of only one face), edge (atoms that are within a constrained zone width of exactly two faces), and

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

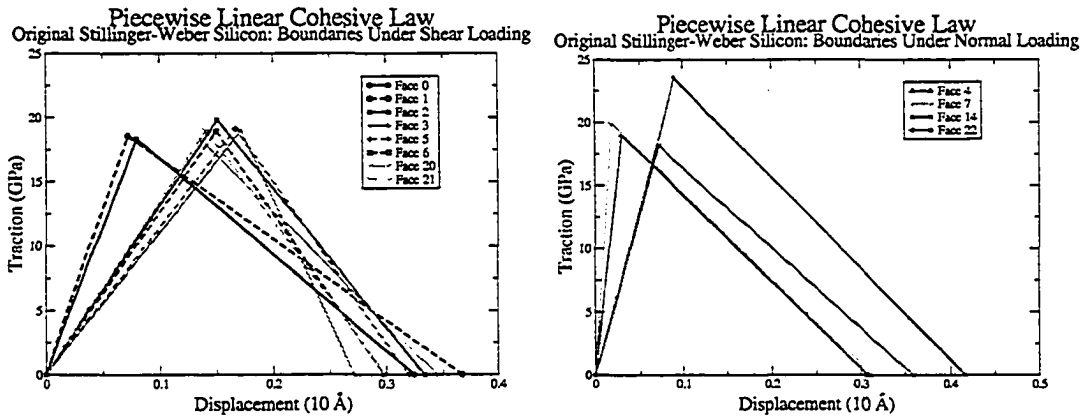



Figure 2.6: **Piecewise Linear Cohesive laws: Ordinary Stillinger–Weber Silicon.** The same as figure 2.5 but for the original, ductile Stillinger-Weber potential for silicon.

corner (atoms that are within a constrained zone width of more than two faces). The atoms on the faces are constrained to not move perpendicular to the face, the atoms on the edge are constrained to move only parallel to the edge, and the atoms in the corners are totally fixed in position to simulate rollered boundary conditions.

OFEMD retrieves the kinematic boundary conditions that are specified in the database. In this model, a normal loading is imposed on the upper face. We manually update the positions of the atoms in the constrained zones that are adjacent to the upper face to impose this boundary condition, incrementing up to 15% strain in 0.5% strain increments, relaxing the atoms at each step.

## 2.4 Cohesive Zone Model Comparison

In this section, we compare the fracture behavior of atomistic and continuum FEM simulations. We use both the standard Stillinger–Weber potential (which is ductile for intragranular fracture) and the modified, brittle Stillinger–Weber potential. We

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

explore simulations of two sizes (inner cube sizes of 10 and 20Å). The interfacial cohesive laws in each case were measured as in Figures 2.5 and 2.6, from MD simulations with 30Å grains. (The snapshots of CZM frames were saved at strain levels .1% greater than those for the atomistic simulations.) The color scales denote $\sigma_{zz}$, the vertical component of stress. The stress for the MD simulations was calculated using in equation 2.7.

Figures 2.7 through 2.10 shows the results of both the atomistic and continuum simulations for brittle Stillinger-Weber silicon, with a length scale (defined by the dimension of the inner cube) of 10 Å. The first row of figures shows the $xy$ center plane of the atomic simulation (roughly the plane of fracture). The second row shows the same plane of fracture for the CZM simulation. The third row shows the $xz$ center plane of the atomistic simulation, illustrating the stresses around the fracture zone and the crack opening. The fourth row shows the $xz$ center plane of the CZM simulation. The stress free state, indicated by the color blue, is an indication that decohesion has occurred across the interface within that region.

We shall see that the MD simulations and the FEM simulations differ in several important respects. First, the FEM simulations fracture overall at a higher stress level. This might be a nucleation effect; the irregular atomic structures at the external faces and internal edges and corners could be acting as nucleation points for fracture in ways that are not reflected in the continuum simulation. Second, the pattern of fracture—which interfaces break in which order—is in some cases different for the two simulations. Some of these differences are accidental; the system has inversion symmetries across the $xz$ and $yz$ planes that is broken only by the microparameter choices in the grain-boundary cohesive law MD simulations and the fully atomistic cube-in-cube simulations. Hence an MD simulation that

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

breaks first along the 'front' edge is equivalent to a FEM simulation breaking along the 'back'. Indeed, were we to use fully converged, infinite-system cohesive laws such as the periodic boundary conditions used in chapter 1, an ideal FEM simulation would break symmetrically. It is also a possibility that asymmetries in the mesh and rounding errors in the finite element simulation contribute to the breaking of symmetry.

### 2.4.1 Brittle Stillinger-Weber with a Length Scale of 10 Å

Figure 2.7 shows the comparison between the smaller simulations of the brittle potential (an inner cube length of 10Å, with the brittle modification of the Stillinger-Weber potential). The atomistic simulation appears to begin fracture at 11% strain in the upper right corner in figure 2.7a with the fracture spreading across the right side and finally across the center plane, excluding the inner cube at 15% strain. At this small scale, the inner cube is amorphized during the first relaxation step.

The finite element simulation begins fracture on the right side as well between 11.1% strain and 12.1% strain, approximately where the MD simulation fractures. The only feature which breaks the 90 degree rotation symmetry for the finite element simulations are the differences in cohesive laws. The finite element simulation fractures slightly more rapidly, also ending by breaking through the inner cube but at 14.1% strain rather than 15%.

### 2.4.2 Brittle Stillinger-Weber with a Length Scale of 20 Å

For the 20 Å length scale atomistic simulations, fracture also begins at the upper right corner in figure 2.8a, however fracture begins noticeably earlier at 8% strain and propagates through the center plane more rapidly. At 9% strain, the atomistic

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

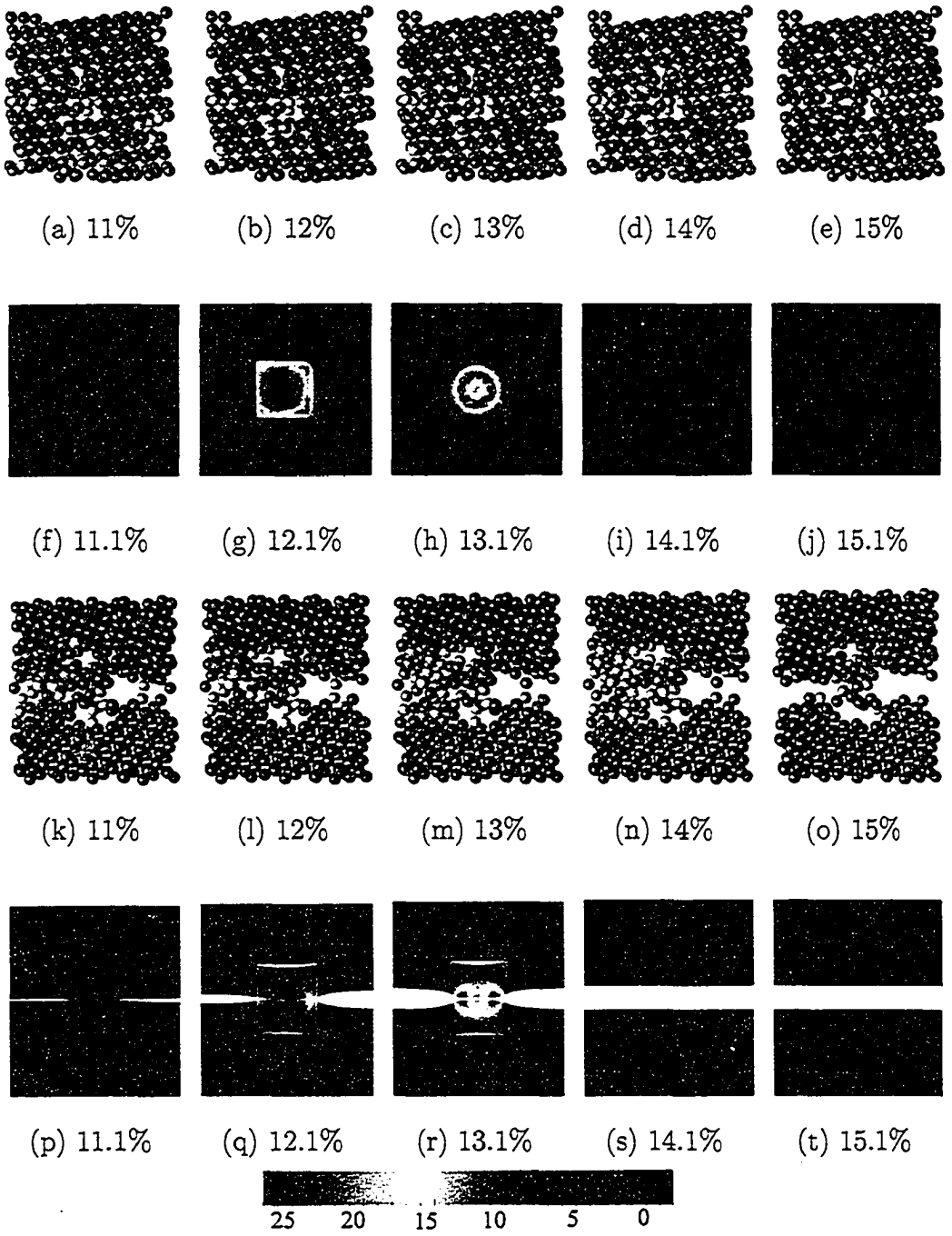



Figure 2.7: **Comparison of the Atomistic and CZM Simulations of the Cube-In-Cube with a Length Scale of 10 Å, using Brittle Stillinger-Weber Silicon.** As described in the text.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

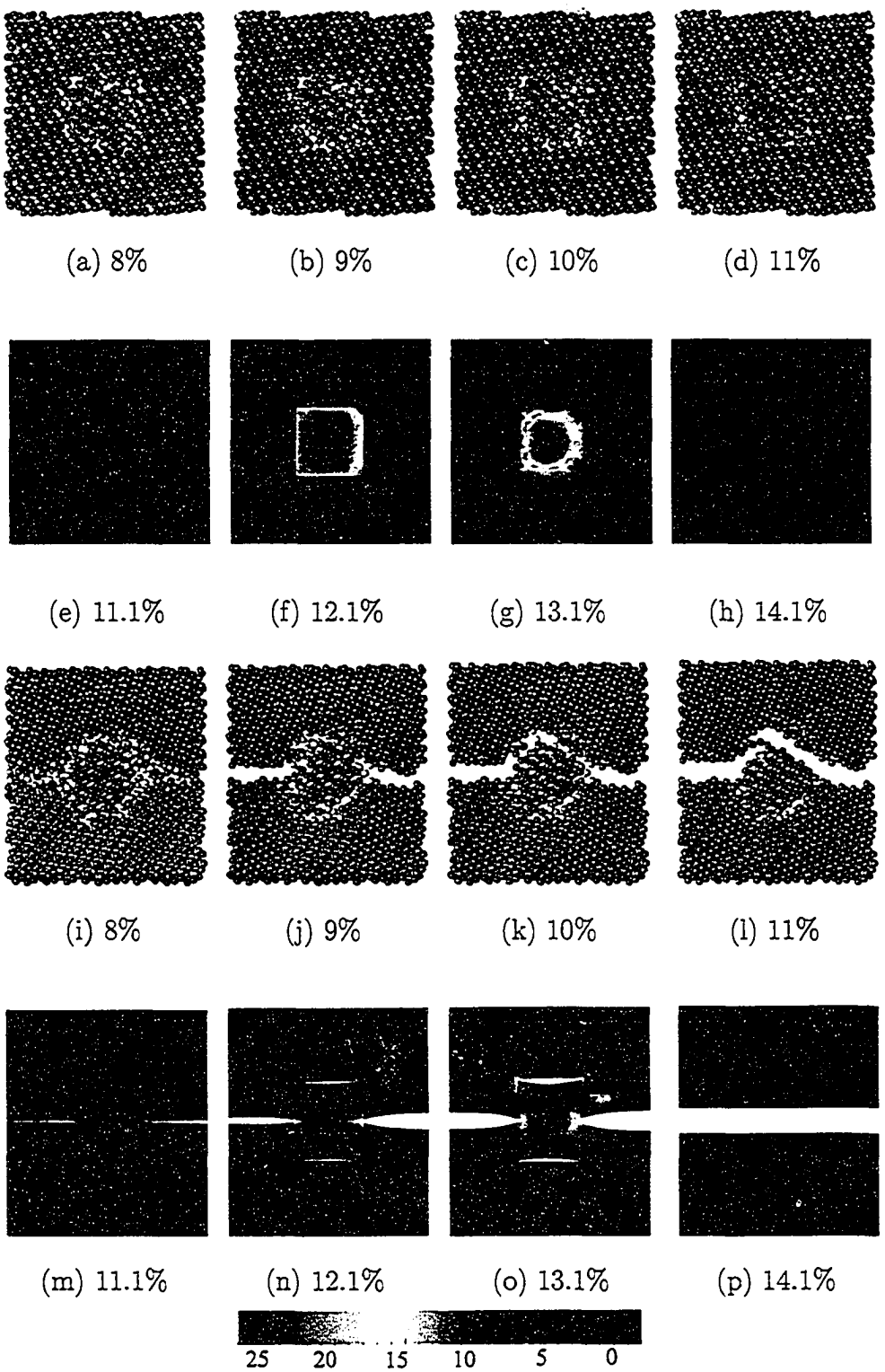



Figure 2.8: **Comparison of the Atomistic and CZM Simulations of the Cube-In-Cube with a Length Scale of 20 Å, using Brittle Stillinger-Weber Silicon.** As described in text.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

simulation is comparable to the continuum simulation at 13% strain, with the center plane, excluding the inner cube, cracked through. This more rapid fracture of the atomistic simulation could be due to microstructure differences, but could also be due to the larger system size. A larger system height means there is more energy stored in elastic strain per unit area of interface. Once a given region reaches the maximum stress that it can sustain, it snaps open. With a smaller system size, the opening of the interface is controlled since the constrained zones are closer. This is related to the effect described in section 1.2.2 where larger systems effectively approach fixed force boundary conditions (figures 1.4 and 1.5).

Both the MD simulation and the finite element simulation begin to decohere at the upper face of the inner cube (compare figure 2.8k with the slight blue decohered region above the inner cube in figure 2.8o). However, the FEM simulation ultimately decoheres at the center plane instead. In the atomistic simulation, we also see a competition between cracking at the top of the inner cube and cracking through the center plane. Ultimately, the crack propagates partially through the inner cube at an angle, reaching the top of the inner cube. This effect cannot be replicated in the finite element simulation because it did not have a interface elements in position to crack at this angle.

### 2.4.3 Original Stillinger-Weber with a Length Scale of 10 Å

For the original version of Stillinger-Weber silicon (which is more ductile for single-crystal fracture), the MD simulations fracture at around 14-15% strain, similar to the fracture threshold seen for the brittle potential MD simulations at that size. The continuum simulations, however, at a much higher strain, 30% compared to 15%, despite using cohesive-zone models derived from the original potential.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

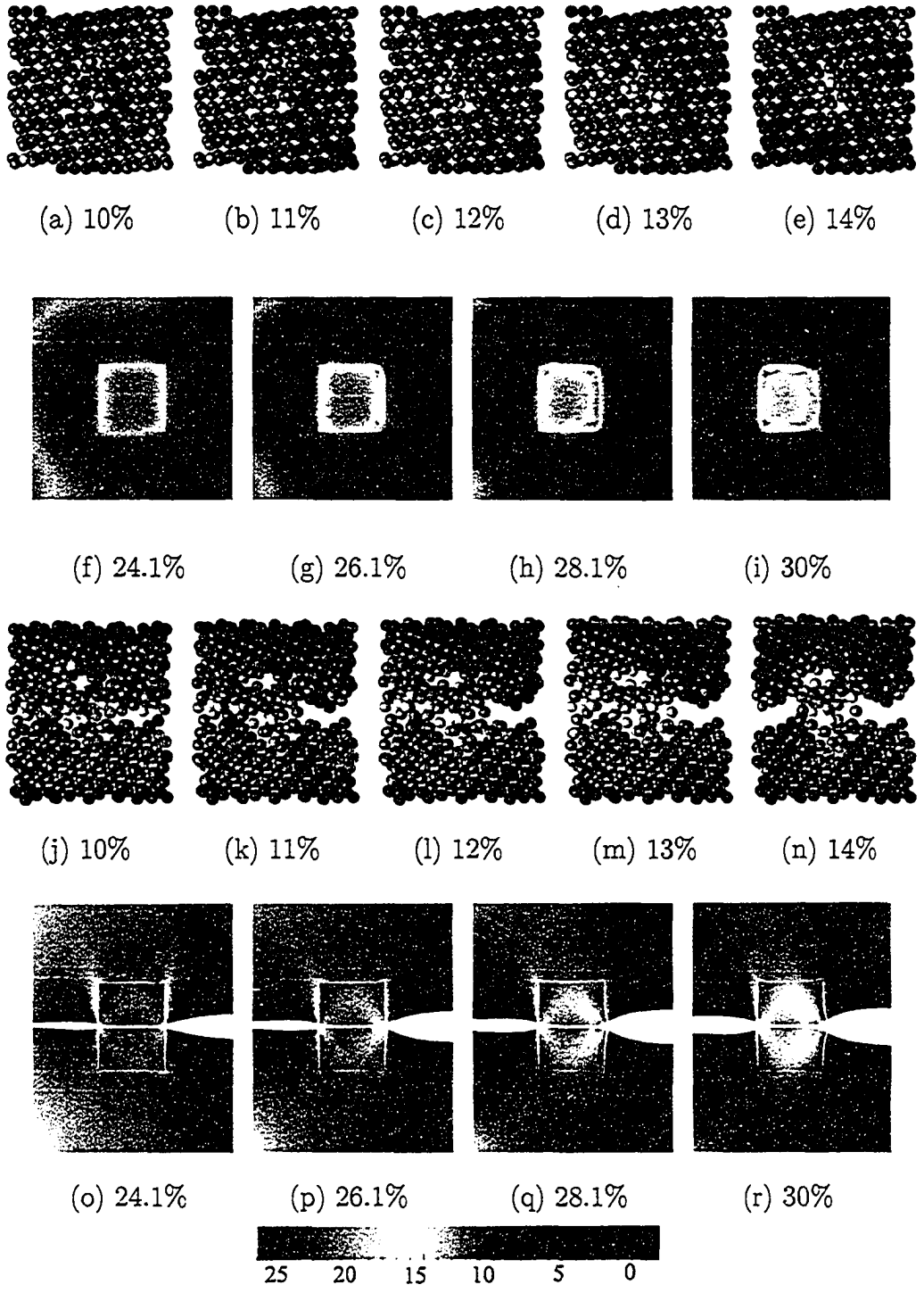



Figure 2.9: **Comparison of the Atomistic and CZM Simulations of the Cube-In-Cube with a Length Scale of 10 Å, using Original Stillinger-Weber Silicon**

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

The atomistic simulation begins fracture at the top (10% strain figure) and spreads along the right side (11, 12% figures). At the end of the MD simulation (14% strain), it has cracked through all but the center cube. The CZM simulation begins fracture along the external edge along the side, and has also not cracked through or around the inner cube at the conclusion of the simulation.

### 2.4.4 Original Stillinger-Weber with a Length Scale of 20 Å

For the 20 Å case, the atomistic simulation again fractures at a much lower stress than does the CZM simulation. The atomistic simulation fractures through all but the center cube very rapidly between 14% and 15% strain, reflecting again the effective soft-spring fixed-stress fracture conditions from the larger system size; the CZM simulation fractures more gradually, showing a sweep from right to left. The behavior of the CZM simulation is similar to that of the 10 Å case with fracture beginning on the right side and slowly propagating through the center plane.

## 2.5 Conclusion

In this chapter, we have described a method for comparing finite element simulations of polycrystal models to fully atomistic simulations of the same geometry. We have based the material properties, including the cohesive laws for grain boundaries, used in the polycrystal simulation on atomistic calculations. We find that in one case, the 10 Å brittle Stillinger-Weber simulation, the agreement between the two simulations in terms of the strain at which the fracture begins, and the pattern of fracture is fair. However it is unclear whether this agreement is accidental.

Many of the differences between the atomistic simulations and the finite element simulations can be attributed to the difference in choice of microparameters defin-

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

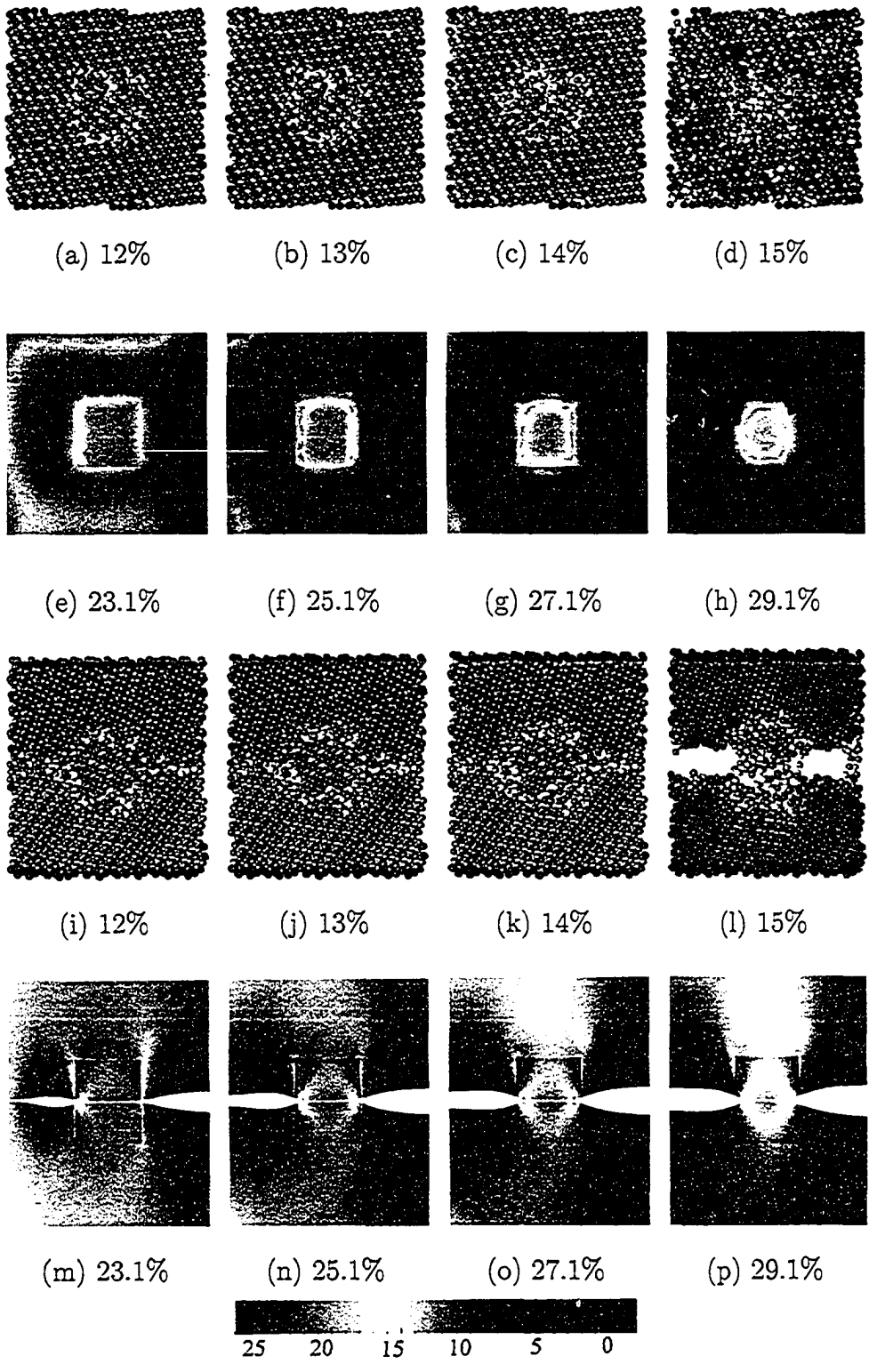



Figure 2.10: **Comparison of the Atomistic and CZM Simulation of the Cube-In-Cube with a Length Scale of 20 Å, using Original Stillinger-Weber Silicon.** As described in text.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

ing the grain boundary geometries (the location of the lattice origin with respect to the interface). Carefully matching these microparameters in future comparisons can only partially correct for these differences – there will always be discreteness effects in atomistic simulations that cannot be replicated in finite element simulations, due to the distortion of atoms at interface corners and junctions of grain boundaries. In Chapter 3 we describe a method for decorating regions of a finite element mesh with atoms to extract atomistic information about the cohesive properties of geometrical features such as interface corners and grain boundary junctions.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHAPTER 3

# OVERLAPPING FINITE ELEMENTS AND MOLECULAR DYNAMICS (OFEMD)

## 3.1 Introduction

OFEMD [1] is a flexible tool for running molecular dynamics simulations within a finite element mesh. OFEMD can perform molecular dynamics simulations with any geometry and a variety of materials and dynamical algorithms using the *DigitalMaterial* package [7]. The finite element model may have regions representing different grains of a single material [2]. OFEMD retrieves geometry information from a remote XML repository by sending an XQuery to a webservice or by executing SQL commands on a remote database [24, 25, 26]. The user has the option to decorate an arbitrary point, mesh vertex, mesh edge, mesh face, or the entire mesh. The geometry options are to create a sphere, cylinder, or cube of atoms with user specified dimensions, or an arbitrary shape defined by the entire finite-element model. The motivations for the possible geometries are scenarios such as decorating the bulk of a nanoscale model (entire mesh), a grain boundary within a model (rectangle decorating a face), a triple junction of grains (cylinder decorating an edge), or a more complicated multiple junction of grains (sphere decorating a vertex). OFEMD then initializes the chosen geometry with atoms. If the chosen geometry parameters span multiple grains within the finite element model,

[1] The name OFEMD and portions of the code are from Nick Bailey.

[2] In principal, the finite element geometry may have multiple materials and an arbitrary overall geometry. If interatomic potentials for multiple materials were to be added to *DigitalMaterial*, OFEMD could simulate bi-material interfaces, junctions of multiple materials, and multiphase nanoscale systems.



Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

OFEMD initializes each grain with a different lattice orientation [3]. OFEMD then performs an atomistic simulation in which the atoms are either deformed according to the mesh displacements solved for by the finite element simulation, or a constrained zone of atoms is used to impose the same kinematic boundary conditions (rollered boundary conditions on all faces with a strain imposed on one face, for example) in the finite element model. OFEMD outputs the atomic positions and stresses (if implemented in the potential) at each time step to a file readable by OpenDX. OFEMD can feed back to larger scale, continuum simulations in three ways: with decision making information such as how a crack propagates at a triple junction of grains, crack initiation, and overall comparison with small scale continuum simulations. OFEMD can also extract the grain boundary geometries for each topological face within the model and print this information for use in a grain boundary fracture simulation using *DigitalMaterial* [7] and Python.

## 3.2 Interface

OFEMD's numerous options can be accessed by passing options in a command line. This simplifies implementing a web service interface as well as use with scheduling software such as OpenPBS [1]. Internally, the user defined parameters are stored in a Python dictionary object. Also, another interface such as a graphical interface or another python script can spawn an OFEMD simulation by passing a Python dictionary with the relevant user parameters to OFEMD.

The parameters are described in table 3.1. The user can get list of parameters by typing at command line:

[3]Ideally, the lattice orientation should be specified in the finite element model. If not, OFEMD randomly generates a lattice orientation.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
> python OFEMD.py --help
usage:
python ScriptName.py [option] [value] ...
options (in no particular order):
['webService', 'timestep', 'height', 'radius', 'strainIncrement',
'nSteps', 'subMesh', 'verbose', 'atomsGeometry',
'findGrainBoundaryGeometries', 'verboseMover', 'databaseUserName',
'cylinderAxis', 'edgeToDecorate', 'databasePassword', 'width',
'rectangleAxes', 'topologyUUID', 'meshUUID', 'meshSource',
'caseID', 'bc', 'brittleFactor', 'lengthScale', 'material',
'QMtolerance', 'checkpointing', 'maxQMIterations',
'deformationMethod', 'displacementScale', 'decoratedFeature',
'parallel', 'randomNumberSeed', 'databaseServer', 'faceToDecorate',
'center', 'database', 'soapAction', 'dispsUUID', 'attributeUUID',
'moverType', 'vertexToDecorate', 'length', 'directory']
```

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table 3.1: **OFEMD Parameters**

| Parameter Name | Type | Default | Description |
|---|---|---|---|
| Atoms Geometry Parameters | | | |
| atomsGeometry | string | "Sphere" | Choices: "Full", "Rectangle", "Cylinder", "Sphere". |
| decoratedFeature | string | "vertex" | Choices: "entire", "point", "edge", "face", vertex. |
| edgeToDecorate | two ints | 0 1 | Two connected mesh vertex ID's specifying an edge in the mesh. |
| faceToDecorate | three ints | 0 1 2 | Three connected mesh vertex ID's specifying a face in the mesh. |
| vertexToDecorate | int | 0 | Any mesh vertex ID. |
| center | three floats | 0.0 0.0 0.0 | Any coordinate within the scaled mesh. This option is only used if decoratedFeature is equal to "point". |
| radius | float | 5.0 | Only used if atomsGeometry is "Sphere" or "Cylinder". |
| cylinderAxis | three floats | 0.0 0.0 1.0 | Only used if atomsGeometry is Cylinder. If decoratedFeature is edge, this option is overridden and the cylinder axis is given by the orientation of the decorated edge. |
| length | float | 10.0 | This option is not used if atomsGeometry is "Sphere" or "Full". |
| width | float | 10.0 | Only used if atomsGeometry is "Rectangle". |
| height | float | 10.0 | Only used if atomsGeometry is "Rectangle". |
| rectangleAxes | nine floats | 1.0 0.0 0.0<br>0.0 1.0 0.0<br>0.0 0.0 1.0 | Only used if atomsGeometry is Rectangle. This option is overridden if decoratedFeature is face or edge. The nine floats must represent the rows of an orthonormal matrix |
| Simulation Type | | | |
| deformationMethod | string | "meshDisp" | Choices: "kinematicBC", "meshDisp". "kinematicBC" can only be used if decoratedFeature is "entire". |

continued on next page

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

continued from previous page

| Parameter Name | Type | Default | Description |
|---|---|---|---|
| findGrainBoundary-Geometries | bool | 0 | If set to 1, OFEMD outputs a file containing the grain boundary geometry for each of the internal faces in the model. |
| Geometry and Mesh Access Parameters | | | |
| meshSource | string | "xml" | Choices: "xml" or "database" |
| soapAction | string | N/A | Valid soapAction URL. |
| webService | string | N/A | Valid XDocExchange web service URL. |
| meshUUID | string | N/A | Valid mesh uuid for XMLRepository webservice. |
| dispsUUID | string | N/A | Valid displacements uuid for XMLRepository webservice. |
| attributeUUID | string | N/A | Valid attributes uuid for XMLRepository webservice. |
| caseID | int | N/A | Valid case ID number for attributes file. |
| topologyUUID | string | N/A | Valid topology uuid for XMLRepository webservice. |
| databaseServer | string | N/A | The domain name of the server hosting the database that contains the mesh |
| database | string | N/A | The name of the database that contains the mesh that the user wishes to access. |
| databaseUserName | string | N/A | A username that has access to the database. |
| databasePassword | string | N/A | The password associated with the username given by databaseUser-Name |
| Scaling Parameters | | | |
| lengthScale | float | 1000000.0 | OFEMD scales the mesh by this factor. This is helpful if the mesh is defined with different length units than those used by the inter-atomic potential. Must make sense with chosen geometry parame-ters, i.e. the scale of the decorated region should be comparable to a mesh element size. |

continued on next page

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

continued from previous page

| Parameter Name | Type | Default | Description |
|---|---|---|---|
| displacementScale | float | 1.0 | OFEMD scales the mesh displacements by this factor. Only used if deformationMethod is "meshDisp". |
| subMesh | string | connectivity | Choices: "cube", "connectivity", only used if atomsGeometry is not "Full" |
| Scaling Parameters | | | |
| nSteps | int | 2 | The number of steps taken in a "kinematicBC" simulation |
| QMtolerance | float | 0.01 | Any float between 0 and 1. This is the tolerance used by the quickMin atoms mover. |
| maxQMIterations | integer | 100 | The maximum number of iterations to be used by the quickMin atoms mover. |
| timestep | float | 0.001 | Any reasonable timestep. |
| MD Parameters | | | |
| material | string | "SW_evA" | Determines the interatomic potential used by *DigitalMaterial* [7]. Choices: "EDIP" (Silicon [10, 33]), "SW_nat" (Stillinger-Weber silicon with natural units [43]), "SW_evA" (Stillinger-Weber silicon with eV and Åas units [43]), "CLJ" (Lennard-Jones with a cutoff described in Chapter 1), "HolianCLJ" (Holian cut Lennard-Jones [28], "MEAM_Al", "MEAM_Si","MEAM_W", "MEAM_Fe" (Modified Embedded Atom Method for aluminum, silicon, tungsten, and iron [9, 8]). |

continued on next page

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

| *continued from previous page* | | | |
|---|---|---|---|
| Parameter Name | Type | Default | Description |
| brittleFactor | float | 2.0 | Used only by the Stillinger-Weber potentials, brittleFactor is a parameter that tweaks the "brittleness" of the material. |
| bc | string | "constrained" | Choices: "rollered", "constrained". "rollered" is only appropriate if atomsGeometry is "Rectangle" or "Full". |
| randomNumberSeed | int | 5 | Any positive integer, this changes the randomly generated lattice orientations, allowing for repeatable results. |
| moverType | string | "qm" | Choices: "qm" (QuickMin), "verlet", "cg" (Conjugant Gradients), "langevin", "combo" (a combination of Conjugant Gradients and Quick Min). |
| System Parameters | | | |
| checkpointing | bool | 0 | Whether to periodically save the state. |
| verbose | int | 3 | Any integer between 0 and 5. Defines the level of debugging output. |
| verboseMover | bool | 0 | Whether to output debugging info from the atoms mover. |
| directory | string | ./ | The directory in which to output data from the OFEMD simulation. |
| parallel | bool | 0 | OFEMD needs to be explicitly told whether it is running in parallel or not. |

## 3.3 Example Problems

The following example problems were run using a finite element geometry of a cube consisting of 5 grains generated with a Voronoi tessellation. All atomistic simulations were done with a MEAM [9, 8] potential for Si with a diamond lattice as the ground state. Figure 3.1 is a sphere of atoms decorating a vertex in the mesh where 4 grains meet. An example of a command to run this simulation (excluding the mesh access parameters) is:

```
python OFEMD.py atomsGeometry Sphere decoratedFeature vertex
vertexToDecorate 10 radius 20.0 lengthScale 200000
displacementScale 50 QMtolerance 0.0001 maxQMIterations 10000
```

Figure 3.2 is a cylinder of atoms decorating an edge which is a triple junction. Example command line options are:

```
python OFEMD.py atomsGeometry Cylinder decoratedFeature edge
edgeToDecorate 9 5 radius 20.0 length 20 lengthScale 200000
displacementScale 50 QMtolerance 0.0001 maxQMIterations 10000
```

Figure 3.3 shows a rectangle of atoms decorating a face of the mesh that coincides with a grain boundary. Example command line options are:

```
python OFEMD.py atomsGeometry Rectangle decoratedFeature face
faceToDecorate 16 28 98 length 30 width 30 height 30
displacementScale 100 QMtolerance 0.0001 maxQMIterations 10000
```

Figure 3.4 shows the entire finite element mesh decorated with atoms. An example of a command line that would produce such a simulation is:

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

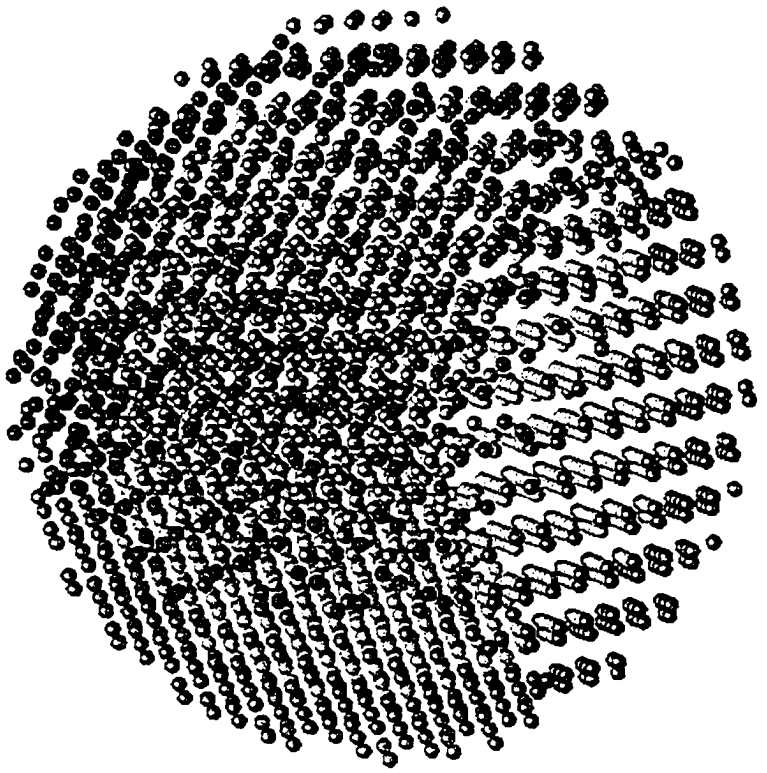


Figure 3.1: **Sphere Of Atoms Decorating a Vertex.** In this example, the vertex constitutes a meeting point of four grains. Each hue indicates a different grain and darker atoms indicate the constrained layer of atoms.

```
python OFEMD.py atomsGeometry Full decoratedFeature entire
deformationMethod kinematicBC lengthScale 50000 nSteps 10
QMtolerance 0.0001 maxQMIterations 10000
```

## 3.4 Infrastructure

### 3.4.1 Retrieving the Mesh

The first step in an OFEMD simulation is to retrieve the mesh information. This is done by the Mesher class which uses FemLib (a library of finite element routines) to retrieve relevant mesh info from either a remote database or an XQuery webservice.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

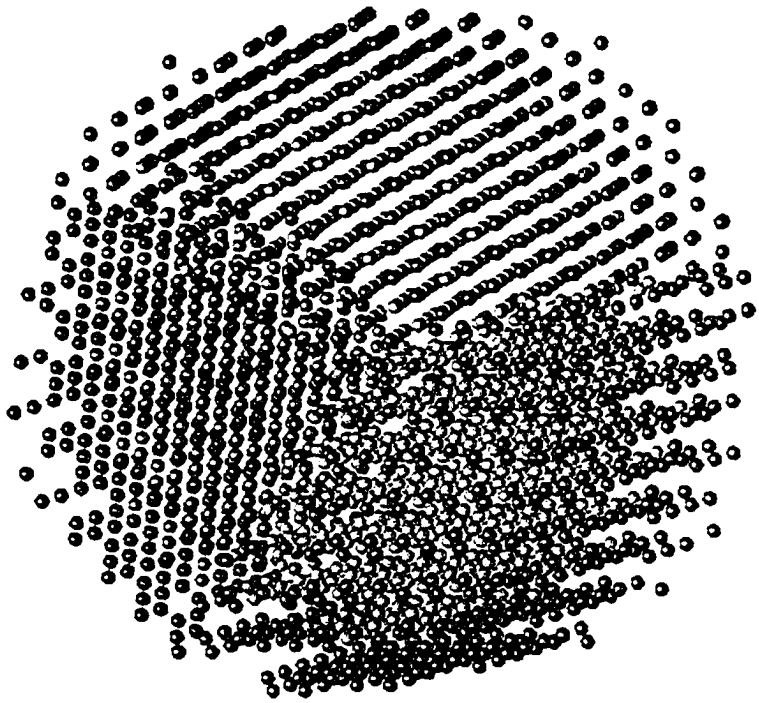


Figure 3.2: **Cylinder of Atoms Decorating and Edge.** The natural shape to use when examining a triple junction of grains is a cylinder. The endcaps are rollered allowing for the possibility of examining how a crack that intersects the faces of the cylinder would propagate upon meeting the triple junction of grains.

If atomsGeometry is set to "Full", Mesher retrieves the full mesh. Otherwise, for efficiency, Mesher retrieves a subset of the mesh with a method that depends on the value of the subMesh parameter. This step presents a challenge in that before we have retrieved the mesh, we are unaware of how the scale of the typical element size in the region we are decorating (often, element sizes vary by orders of magnitude within a mesh) compares to the scale of the decorated region. OFEMD provides two methods for retrieving sub-meshes: "cube", which is more appropriate for cases where the scale of the decorated region is comparable or larger than the size of the elements, and "connectivity", which is more appropriate if the scale of the decorated region is small compared to the size of an element. If subMesh

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

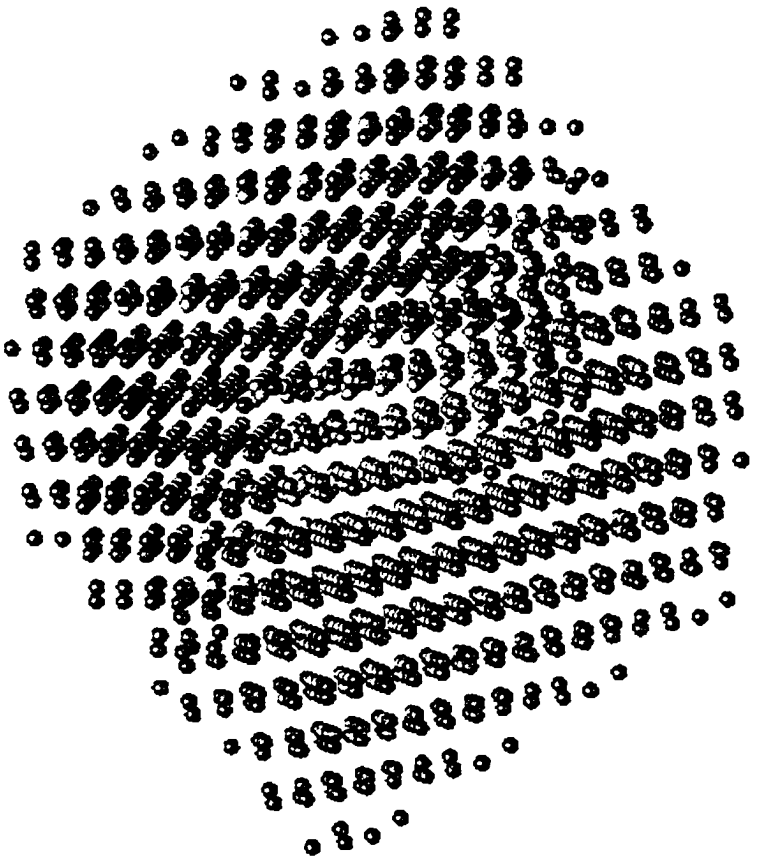


Figure 3.3: **Cube Decorating a Face.** Decorating a grain boundary face with a rectangular cluster of atoms would allow the user to see if and how a crack might initiate at that grain boundary under mixed mode loading.

is set to "cube", the Mesher retrieves a portion of the mesh contained by a cube with length determined by geometry choices made by the user (*radius* for sphere, $\sqrt{radius^2 + \frac{1}{4}length^2}$ for cylinder, $\sqrt{length^2 + width^2 + height^2}$ for rectangle) plus a cutoff. If this cube contains no complete elements, the length is doubled until at least one complete element is found. The advantage of this method is that the mesh retrieved will contain the entire atomistically decorated region if it extends past many elements but has the pitfall that it might not retrieve all of the needed elements that contain the decorated region if the scale of the decorated region is small compared to the scale of an element: it is time consuming to check that each

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

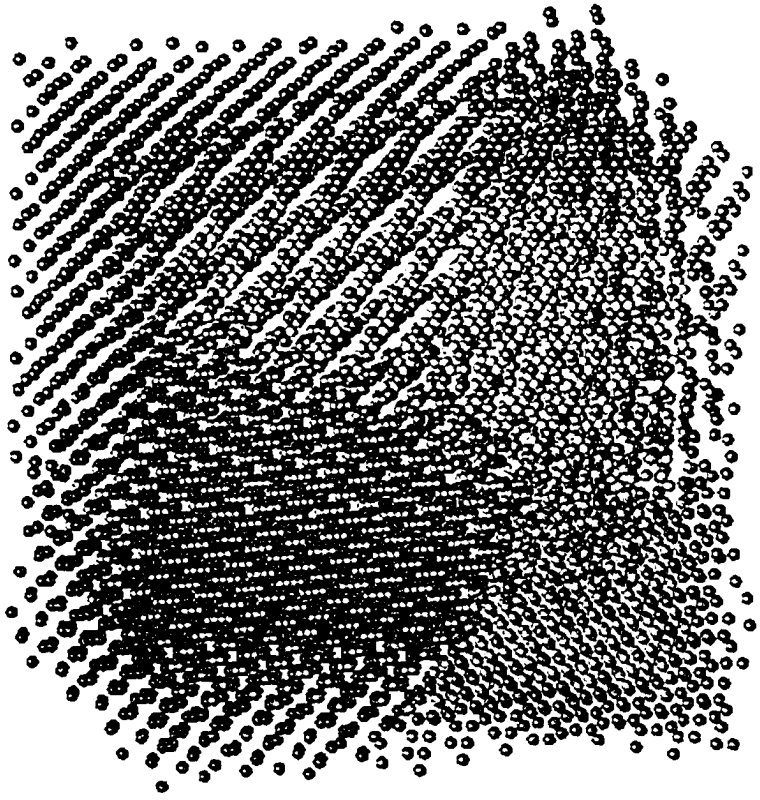


Figure 3.4: **Decorating an Entire Finite Element Mesh.** Decorating the entire finite element mesh is only tractable if the length scale of the mesh (adjustable with the lengthScale parameter) is small. For such simulations we can make a direct comparison to finite elements.

atom is contained in an element. For simulations where the scale of the region decorated with atoms is small compared to the size of an element, and a specific feature of the mesh is being decorated, the "connectivity" option for subMesh is more appropriate since it will retrieve the minimum portion of the mesh necessary and avoids doubling the length of the cube many times. If subMesh is set to "connectivity", the Mesher retrieves all elements that are connected to the feature of the mesh (vertex, edge, or face) that is being decorated.

The mesher is also responsible for overriding the user set geometry parameters based on which feature of the mesh is to be decorated. The center is reset to the location of the vertex to be decorated, the center of the edge to be decorated, or

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

the center of the face to be decorated. If the decoratedFeature is an edge, then the cylinder axis or one of the rectangle axes is set such that it is aligned with the edge. If the decorated feature is a face, then the cylinder axis or one of the rectangle axes is set to be perpendicular to the face.

### 3.4.2 Decorating With Atoms

The AtomsDecorator classes initialize all of the ListOfAtoms objects (the data structure used by *DigitalMaterial* [7]) and the rest of the molecular dynamics objects used by *DigitalMaterial*. There are four classes derived from a base AtomsDecorator class corresponding to the four geometry options: SphereDecorator, CylinderDecorator, RectangleDecorator, and FullDecorator.

The SphereDecorator, CylinderDecorator, or RectangleDecorator loops over each material contained in the sub-mesh found by the mesher, and initializes a sphere, cylinder, or rectangle of atoms with the given dimensions and lattice orientation. It then loops over each of the atoms and removes any atoms that are not contained within the current material. A shell of atoms for the constrained layer is initialized in a similar manner. Note that the final dimensions of a sphere, cylinder, or rectangle will be equal to the user specified dimensions plus a shell of atoms with a thickness given by twice the cutoff distance of the chosen interatomic potential. Two cutoff distances are needed to completely shield the inner atoms from edge effects since some of the potentials have three body terms as discussed in section 2.3. For sphere or cylinder simulations, the only boundaryConditions option is fixed. For the sphere this fixes all of the atoms in the constrained shell in all directions. For the cylinder, the constrained layer in the curved shell is fixed in three dimensions while the endcaps are constrained to not move along the direction

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

of the cylinder's axis (rollered boundary conditions). For a rectangular cluster, the constrained layers can be rollered or fixed in all three directions.

The FullDecorator fills an arbitrary geometry by performing a similar routine to the other AtomsDecorators, except it finds a bounding box for each material region and initializes a rectangular cluster of atoms to fill this box. The atoms outside the current material are then removed from the rectangular cluster, allowing for arbitrarily shaped material regions to be filled. The constrained layer is found by detecting which atoms are within two cut-off distances of the faces of the model. Presently, the FullDecorator assumes rollered boundary conditions. In order to implement rollered boundary conditions, the atoms adjacent to each face are put into a different ListOfAtoms branch. These atoms are then constrained to not move in a direction perpendicular to the face. Separate branches are needed for the atoms that are adjacent to more than one face such that there is a branch for each face, edge, and corner. If an atom is adjacent to two faces, it is constrained to move only in the direction parallel to the edge where the two faces meet. If an atom is in a corner of the model (adjacent to more than two faces) it is fixed.

### 3.4.3 Deforming the Atoms

The third step in an OFEMD simulation is to deform the atoms. If deformationMethod is set to "kinematicBC" (which is only appropriate if whole mesh is decorated), OFEMD uses the kinematic boundary conditions, a deformation on a face or faces specified by the finite element model, to increment the positions of the appropriate constrained zone. The atomic positions are then relaxed or integrated forward in time according to the chosen dynamical algorithm.

The AtomsDeformer class is responsible for deforming the atoms according

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

to the mesh displacements if the deformationMethod is set to "meshDisp". The AtomsDeformer class loops over the atoms and uses FemLib to find the natural coordinates of each atom relative to the element that contains that atom. AtomsDeformer then retrieves the mesh displacements from the database or XML repository. The displacement of each atom is then found by interpolating the displacements of the mesh nodes to find the displacement of each atom using the natural coordinate (relative to the element) found earlier. The positions of all of the atoms (including the constrained layers) are updated according to the mesh displacements. The constrained layers are then fixed (in one or more dimensions) according to the user specified boundary conditions and the atomic positions are relaxed or integrated forward in time according to the chosen dynamical algorithm.

### 3.4.4 Retrieving Grain Boundary Geometries

One of OFEMD's options, findGrainBoundaryGeometries, does not decorate a mesh or perform an atomistic simulation but simply retrieves information from the mesh for use in grain boundary simulations using *DigitalMaterial* [7] as described in Chapter 1. If this option is chosen, the only other options that are relevant are the mesh specification options.

The GrainBoundaryGeometryFinder class is responsible for retrieving grain boundary geometry information from the mesh. The GrainBoundaryGeometryFinder retrieves the basic mesh information using FemLib and then loops over each of the internal topological faces in the model. For each face, it finds the vector normal to the face and a rotation matrix that rotates this vector to the $x$-axis (assuming that the grain boundary python script places the grain boundary on the $yz$-plane). GrainBoundaryGeometryFinder then rotates the lattice orientation

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

vectors for the two materials that meet at that topological face and outputs this information to ascii files labeled by the face ID number.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHAPTER 4

# THE BRITTLE-DUCTILE TRANSITION FOR GRAIN BOUNDARIES

## 4.1 Introduction

A classic model of brittle vs. ductile behavior was proposed by Rice and Thomson [38] in which the determining factor for brittle or ductile behavior at zero temperature is whether or not there is an energy barrier for dislocation nucleation. In cases where there is an energy barrier, a brittle to ductile transition temperature can be found. This model gives good estimates for whether or not a material is brittle at zero temperature but bad results for the transition temperature. We will review this model in detail and adapt it for fracture at grain boundaries. Since this model depends on surface energies and grain boundary energies, we have calculated these energies atomistically for several geometries in Al, Si, Fe, and W. We then use the adapted model and atomistic surface and grain boundary energies to estimate which grain boundaries are brittle or ductile at zero temperature and to find the transition temperature.

An estimation of whether a grain boundary is brittle or ductile at zero temperature is useful in preparing for atomistic simulations of grain boundary fracture. Atomistic simulations of ductile fracture are not computationally feasible because the dislocations spread out rapidly, increasing the size of the simulation needed for accurate results. Even simulations of brittle fracture, done in parallel, can take 12 hours or more to complete. Measurements of surface and grain boundary energies take only minutes, allowing for an estimate of whether or not a grain boundary geometry is brittle to be made very quickly before a much more time consuming



Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

fracture simulation is made.

## 4.2 Rice and Thomson's model

Rice and Thomson describe brittle vs. ductile behavior of cracks as a competition between cleavage of atomic planes and the emission of dislocations at the critical stress intensity for a brittle crack to grow [38]. If a dislocation loop with a Burger's vector having a component normal to the crack line is nucleated and expands, an atomically sharp crack will be blunted. The external stress necessary to cause such a crack to grow is substantially increased. Materials for which this scenario is probable are considered ductile. If a dislocation loop does not nucleate and expand, the crack will continue to grow in a brittle fashion.

The nucleation and growth of a dislocation loop at a crack tip is described by an energy function derived by Rice and Thomson[38]. They assume that the dislocation loop is a half circle in a slip plane which intersects the crack front as shown in figure 4.1. The total energy of this configuration consists of three parts: the self energy of the dislocation loop, the surface energy of the ledge created when the crack is blunted, and the energy of the dislocation loop in the stress field of the crack.

In a dislocation loop, the stress field of each part of the loop exerts a force on the rest of the loop. The self energy of a dislocation loop is the work done against these forces [27]. The interaction energy of two dislocation loops is given by

$$W_{12} = \int_{A_1} dA_{1_\beta} b_{1_\alpha} \sigma_{2_{\alpha\beta}} \tag{4.1}$$

where $A_1$ is the area within loop 1, $b_1$ is the Burger's vector of loop 1, and $\sigma_2$ is the stress field produced by loop 2. If the Peach Koehler formula for the stress of

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

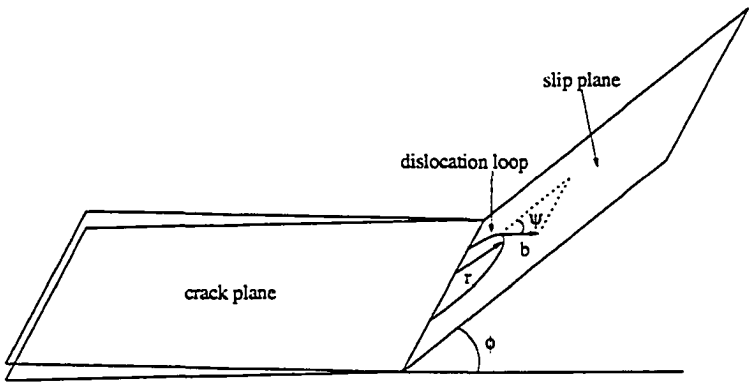



Figure 4.1: **Crack and Slip Plane Geometry.** The geometry of the dislocation loop is defined by the inclination of the slip plane relative to the crack plane $\phi$, the radius of the dislocation loop $r$, the Burger's vector of the dislocation loop $b$ and its inclination within the slip plane, $\psi$.

a dislocation loop is inserted into equation 4.1, the result is

$$W_{12} = -\frac{\mu}{2\pi}\oint_{C_1}\oint_{C_2}\frac{(\mathbf{b}_1\times\mathbf{b}_2)\cdot(d\mathbf{l}_1\times d\mathbf{l}_2)}{R}+\frac{\mu}{4\pi}\oint_{C_1}\oint_{C_2}\frac{(\mathbf{b}_1\cdot d\mathbf{l}_1)(\mathbf{b}_2\cdot d\mathbf{l}_2)}{R} + \frac{\mu}{4\pi(1-\nu)}\oint_{C_1}\oint_{C_2}(\mathbf{b}_1\times d\mathbf{l}_1)\cdot\frac{\partial^2 R}{\partial x_i\partial x_j}\cdot(\mathbf{b}_2\times d\mathbf{l}_2) \quad (4.2)$$

where R is the distance between $d\mathbf{l}_1$ and $d\mathbf{l}_2$, $\mu$ is the shear modulus of the material and $\nu$ is the Poisson ratio. The self energy of a loop can be obtained from this by setting $C_1 = C_2 = C$ and $\mathbf{b}_1 = \mathbf{b}_2 = \mathbf{b}$ and dividing by 2 since equation 4.2 counts the interaction between two given elements twice. This gives

$$W_s = \frac{\mu}{8\pi}\oint_C\oint_C\frac{(\mathbf{b}\cdot d\mathbf{l}_1)(\mathbf{b}\cdot d\mathbf{l}_2)}{R}+\frac{\mu}{8\pi(1-\nu)}\oint_C\oint_C(\mathbf{b}\times d\mathbf{l}_1)\cdot\frac{\partial^2 R}{\partial x_i\partial x_j}\cdot(\mathbf{b}\times d\mathbf{l}_2). \quad (4.3)$$

For a circular dislocation loop (figure 4.2) this is

$$\begin{aligned} W_s &= \frac{\mu b^2}{8\pi}\oint_C \sin\theta_1 dl_1\int_{l1+R_0}^{l1+2\pi R-R_0}\frac{\sin\theta_2}{R'}dl_2 \\ &+ \frac{\mu b^2}{8\pi(1-\nu)}\oint_C\cos\theta_1 dl_1\int_{l1+R_0}^{l1+2\pi R-R_0}\frac{\cos\theta_2}{R'}dl_2 \quad (4.4)\\ R' &= 2R\sin\frac{\theta_2-\theta_1}{2} \qquad dl_1 = Rd\theta_1 \qquad dl_2 = Rd\theta_2, \end{aligned}$$

where $R_0$ is the core cutoff of a dislocation. The integration limits are such that the interactions of elements of the loop that are within a core cutoff distance of

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

each other do not contribute to the integral. Performing the integration gives:

$$U_{self} = \mu b^3 r \frac{2-\nu}{8(1-\nu)} \ln \frac{8r}{e^2 \xi_0}, \tag{4.5}$$

the self energy of a circular dislocation loop with radius $r = R/b$ and core cutoff $\xi_0 = R_0/b$ in units of Burger's vector b. This formula is also appropriate for a dislocation half circle at a surface as it represents the self energy of the half circle and its image.

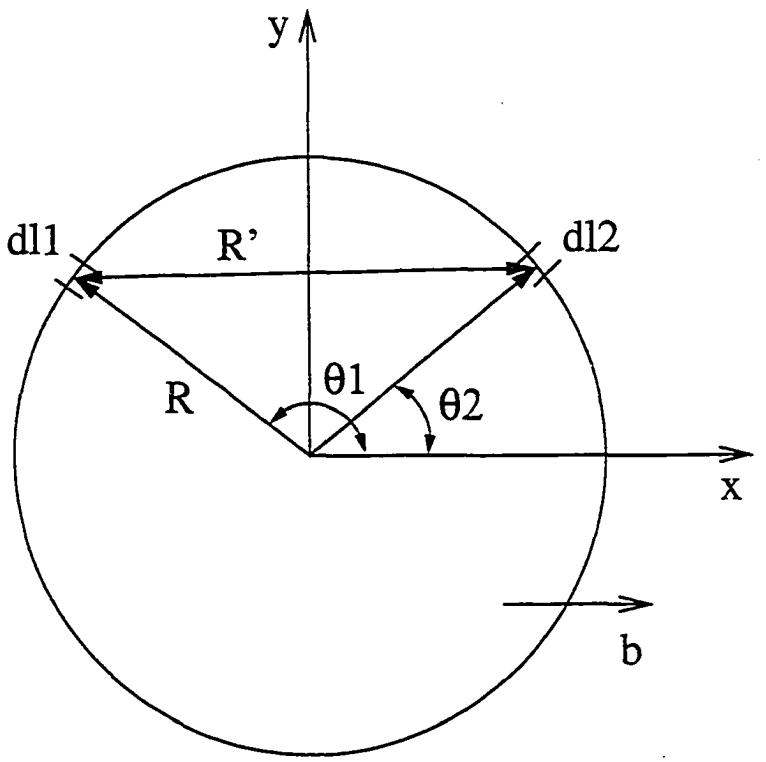



Figure 4.2: **A Circular Dislocation Loop.** To find the self energy of a dislocation loop, we perform a double integral over the same loop. We assume that the Burger's vector is in the direction of the $x$-axis.

The energy of the ledge left behind at the crack front, by the dislocation loop (figure 4.3) is given by

$$U_{ledge} = 2\gamma b^2 \cos\psi \sin\phi (r - \xi_0). \tag{4.6}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

The surface energy per unit area is $\gamma$. The height of the ledge is $b\cos\psi\sin\phi$ since $\psi$ is the angle of the Burgers vector with the crack front [1] and $\phi$ is the inclination of the slip plane. The width of the ledge is $2br$, since $r$ is in units of $b$. $\xi_0$ is subtracted from $r$ because the ledge is only fully formed when $r$ is greater than the cut-off radius. Rice and Thomson [38] explain equation 4.6 in terms of the misfit function for a Peierls model of a dislocation rather than as a surface energy times an area. They take the factor of $\sin\phi$ to "approximately represent the modification of ledge energy from $\gamma\cos\phi$ due to slip plane inclination". It is unclear whether this factor is actually necessary in terms of the area description as the inclination of the single atom wide ledge is undefined. Also note that $\gamma$ is an average surface energy, used here to represent the surface energy of a single row of atoms. As a continuum approximation of a single atomic plane blunting the crack tip, this constitutes a serious limitation of the model. A more accurate model would include atomistic calculations of the energy associated with a row of atoms blunting the crack tip.

The shear stress, given by elastic theory of normal (mode I) loading of a crack tip, at a distance $\rho$ on the slip plane (figure 4.1) is

$$\sigma_{\rho\phi} = K_I(8\pi\rho)^{-1/2}\sin\phi\,\cos\phi/2 \tag{4.7}$$

where $K_I$ is the elastic stress intensity factor [2] and $\rho$ must be small compared to the overall crack length. The stress intensity factor that would cause the crack to grow (if no dislocations were emitted) can be found by solving the Griffith's

[1] Rice and Thomson [38] first treat the dislocation emitted from the crack tip as a line parallel to the crack front. In this case, $\cos\psi$ is the edge component of the dislocation.

[2] The stress around a crack tip is often written in the form $K_{I,II,III}(2\pi r)^{-1/2}f(\theta)$, where $K_I$, $K_{II}$, and $K_{III}$ are the stress intensity factors of the three modes of loading: normal, shear parallel to the crack line, and shear perpendicular to the crack line. Because of the $r^{-1/2}$ singularity in the stress field, the stress at a crack tip is often described in terms of the 'stress intensity factors': $K_I$, $K_{II}$, and $K_{III}$.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

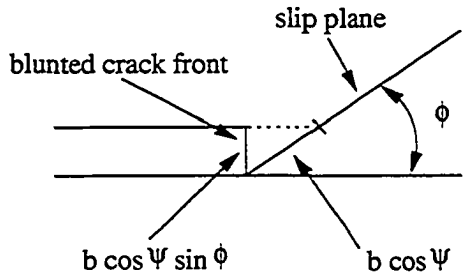



Figure 4.3: **Ledge Left Behind at Crack Front by an Emitted Dislocation Loop.** As a dislocation is emitted from an atomically sharp crack front, the crack front is blunted, creating a ledge with a height roughly equal to $b\cos\psi$. Rice and Thomson [38] include a factor of $\sin\phi$ to represent an adjustment to the ledge energy due to slip plane inclination. It is unclear whether this factor is necessary and it appears that the energy measurement of an atomic scale ledge can best be achieved atomistically.

criterion [3]:

$$G = \frac{1-\nu^2}{E} K_I^2 = 2\gamma \tag{4.8}$$

where E is the Young's modulus of the material. The energy to expand the loop from a radius of $\xi_0$ to $r$ is

$$\begin{aligned} U_\sigma &= \int_{\xi_0}^{r}\int_0^{\pi} \sigma_{\rho\phi} b_\perp r \, dr \, d\theta \\ &= -\left[\frac{E\gamma b}{4\pi(1-\nu^2)}\right]^{1/2} b^2 \cos\psi \sin\phi \cos\phi/2 \int_{\xi_0}^{r} dr \int_0^{\pi} d\theta \frac{r}{\sqrt{r\sin\theta}}, \end{aligned} \tag{4.9}$$

$b_\perp = b\cos\psi$ is the component of the dislocation which is perpendicular to the crack front, $\rho = br\sin\theta$. Integrating equation 4.9 gives

$$U_\sigma = -0.9862\sqrt{\frac{2}{1-\nu}}\mu b^3 \sqrt{\frac{\gamma}{\mu b}} \sin\phi \cos\psi \cos\phi/2 (r^{3/2} - \xi_0^{3/2}). \tag{4.10}$$

The total energy of the dislocation loop is then:

$$U_{act} = \mu b^3 \left[ r U_0 \ln\frac{r}{\xi_0} + U_l(r-\xi_0) - \frac{2}{3} U_s (r^{3/2} - \xi_0^{3/2}) \right],$$

[3]The Griffith's criterion states that the plane strain, $\frac{1-\nu^2}{E}K_I^2$ relieved by fracture equals the energy of the surfaces created.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

$$
\begin{aligned}
U_0 &= \frac{2-\nu}{8(1-\nu)}, \\
U_l &= 2\frac{\gamma}{\mu b}\cos\psi\,\sin\phi, \\
U_s &= \frac{2.092}{\sqrt{1-\nu}}\sqrt{\frac{\gamma}{\mu b}}\sin\phi\,\cos\psi\cos\phi/2
\end{aligned}
\tag{4.11}
$$

$(8/e^2) \approx 1$. For certain combinations of the properties of the material, the geometry of the slip plane, and the surface energies of the crack, the energy will be monotonically decreasing in $r$ as shown in figure 4.4 on the left. This implies that dislocation loops can be spontaneously emitted at zero temperature and the material and crack geometry are considered ductile. For other combinations of the parameters listed above, there will be a stable and unstable equilibrium, shown in figure 4.4 on the right, meaning that in order for a dislocation loop to nucleate and grow, it must overcome an energy barrier, $U_b$. Such materials and crack geometries will be brittle at zero temperature. This change in behavior comes in the form of a saddle node bifurcation of the energy function 4.11.

For materials and geometries that do not spontaneously emit dislocation loops, a transition temperature may be found from the energy barrier. The transition temperature is the temperature at which the average waiting time for a dislocation loop to nucleate is equal to the time a brittle crack takes to propagate through a distance comparable to the size of the dislocation loop [3]. Solving

$$
t = \frac{1}{c}e^{\left(\frac{U_b}{kT}\right)} = \frac{2r_c}{v}, \tag{4.12}
$$

where c is the speed of sound, gives a brittle to ductile transition temperature of

$$
T_{BD} = \frac{U_b}{k}\left(ln\left(\frac{2r_c c}{v}\right)\right)^{-1}. \tag{4.13}
$$

Another formula for the brittle to ductile temperature is given by Argon[4]:

$$
T_{BD} = \left(\frac{k}{U_b}ln(c/v) + \frac{\eta}{T_m}\right)^{-1} \tag{4.14}
$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

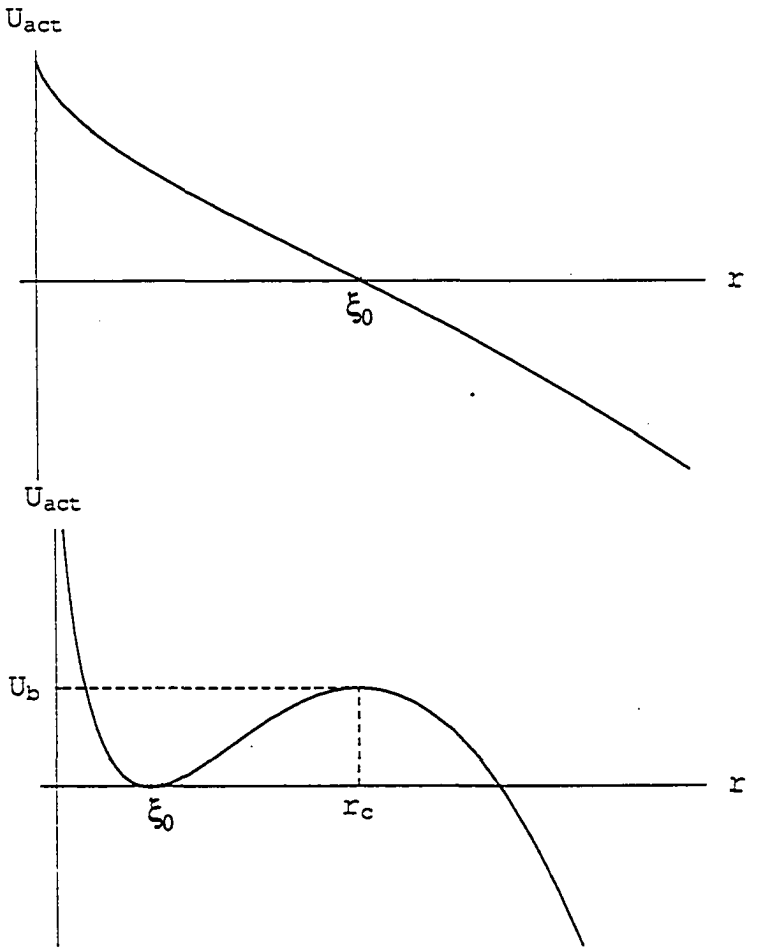



Figure 4.4: **Ductile and Brittle $U_{act}(r)$.** The figure on the top shows the form of $U_{act}(r)$ for inherently ductile materials. The plot on the bottom shows the form of $U_{act}(r)$ materials that are brittle at zero temperature and have a transition to brittle-ductile transition at a temperature given by $U_b$.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

where $T_m$ is the melting temperature and $\eta$ is a parameter close to 0.5 which relates to the temperature dependence of the shear modulus. No explanation is given in [4] for the term involving $T_m$.

We will find in section 4.4 that the Rice and Thomson model gives appropriate qualitative (brittle vs. ductile) behavior for different materials but gives energy barriers for brittle materials leading to transition temperatures that are above the melting temperatures for Si and W. This is due to several limitations of the model. One limitation of this model is the simplified geometry. Argon [4] calculated numerically the energy barrier for three different crack and slip plane geometries: a slip plane which contains the crack line (the one discussed here), a slip plane which intersects the crack plane but does not contain the crack line, and a crack plane with a ledge. He finds that of these three geometries the geometry with a slip plane which contains the crack line gives the highest energy barriers with values of $T_{BD}$, found with equation 4.14, above the melting temperature. The geometry with the crack plane containing a ledge gave the most reasonable energy barriers. Scandian et. al. [17] showed experimentally that $T_{BD}$ was lowered as the density of defects along the crack line was increased. Another feature lacking in the Rice and Thomson model is the mobility of dislocations [4, 22]. If dislocations are emitted but do not relieve the stresses at the crack tip, brittle fracture will still occur [4]. Gumbsch[22] found that at low temperatures, dislocation nucleation governed brittle vs. ductile behavior while at higher temperatures dislocation mobility was the determining factor. Lastly, an accurate energy barrier for dislocation nucleation must be obtained by atomistic methods, rather than continuum methods.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

## 4.3 Fracture at Grain Boundaries

We have extended Rice and Thomson's model to predict whether grain boundaries are brittle or ductile. Since the ductileness or brittleness of a crack depends on the geometry and the fracture energy, and grain boundaries are often more susceptible to fracture than single crystals, it is possible that certain grain boundaries geometries in an inherently ductile material are brittle. We are also interested in cases where a certain orientation of an inherently brittle material may fracture in a ductile manner. Such predictions are useful for atomistic modeling of grain boundary fracture since only brittle fracture is computationally feasible.

In order to model fracture at grain boundaries, equation 4.11 must be modified. The surface energy (or half the Griffith's energy) in the stress field term must be replaced by the fracture energy of a grain boundary, $\gamma_{gb} = \frac{1}{2}(\gamma_1 + \gamma_2 - E_{gb})$ where $\gamma_1$ and $\gamma_2$ are the surface energies of the two grains and $E_{gb}$ is the energy of the grain boundary. The surface energy in the ledge term is considered a constant of the material, $\gamma_l$. Here the surface energies used by Rice [38] are used for $\gamma_l$. The energy barrier for dislocation nucleation at grain boundaries is:

$$\begin{aligned} U_{act} &= \mu b^3 \left[ r U_0 \ln \frac{r}{\xi_0} + U_l(r - \xi_0) - \frac{2}{3} U_s (r^{3/2} - \xi_0^{3/2}) \right], \\ U_0 &= \frac{2-\nu}{8(1-\nu)}; \\ U_l &= 2\frac{\gamma_l}{\mu b} \cos\psi \sin\phi, \\ U_s &= \frac{2.092}{\sqrt{1-\nu}} \sqrt{\frac{\gamma_{gb}}{\mu b}} \sin\phi \cos\psi \cos\phi/2. \end{aligned} \tag{4.15}$$

The bifurcation between spontaneous emission of dislocations and the activated state occurs when both $\frac{dU_{act}}{dr} = 0$ and $\frac{d^2U_{act}}{dr^2} = 0$ for the same $r$. Solving for the

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

grain boundary fracture energy gives

$$\gamma_{gb_c} = \frac{\mu b^3}{68.029\,\xi_0} \csc^2\phi \sec^2\frac{\phi}{2} \sec^2\psi \frac{(2-\nu)^2}{1-\nu} \exp\left(-1 + \frac{16(1-\nu)}{2-\nu}\frac{\gamma_l}{\mu b}\sin\phi\cos\psi\right). \tag{4.16}$$

From equation 4.16, and the material properties given by Rice [38] and shown in table 4.1, we can find the minimum fracture energies (minimizing over $\phi$ and $\psi$) which allow ductile behavior for Si, Al, Fe, and W. These are shown in table 4.1. We have mentioned in section 4.2 how the $\sin\phi$ factor in the ledge energy may not be necessary. Removing that factor, and calculating the critical fracture energy again gives an estimate of the error in the Rice and Thomson model. This value, $\gamma^{(2)}_{gb_c}$ is also shown in table 4.1.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table 4.1: **Material Properties and Minimum Griffith's Energy for Ductile Behavior.**

| Material | $\gamma$ ergs/cm$^2$ | $\mu$ dynes/cm$^2$ $\times 10^{11}$ | $\nu$ | $b$ Å | $\xi_0$ Burger's vectors | $\gamma_{gb_c}$ ergs/cm$^2$ | $\gamma^{(2)}_{gb_c}$ ergs/cm$^2$ |
|---|---|---|---|---|---|---|---|
| Si | 1200 | 6.05 | 0.215 | 3.83 | 0.25 | 4689 | 4801 |
| Al | 840 | 2.51 | 0.347 | 2.85 | 2 | 263 | 279 |
| Fe | 1975 | 6.92 | 0.291 | 2.49 | 2/3 | 1907 | 2020 |
| W | 1700 | 16.0 | 0.278 | 2.74 | 2/3 | 3055 | 3106 |

Initially, the same value of $\gamma_{gb}$ was used for $\gamma_l$. In this case an analytic form for $\gamma_{gb_c}$ could not be found. Even a criterion for whether or not there was a solution for $\gamma_{gb_c}$ could not be put in an analytic form. Solving numerically showed that for Al, Fe, and W, there was a range of $\gamma_{gb}$ that gave ductile behavior. For Si, this form of the dislocation loop energy did not give ductile behavior for any value of $\gamma_{gb_c}$. This is because of the positive contribution of the ledge term which depends on the surface energy. Because using $\gamma_{gb}$ in the ledge term gave this unphysical dependence on the grain boundary energy, a single value of the surface energy was used for all geometries of each material.

## 4.4 Results

In order to calculate energy barriers and transition temperatures using equations 4.15 and 4.13, we need to calculate surface energies and grain boundary energies. This was done using the *DigitalMaterial* [7] package with the MEAM potential[9, 8] for all four elements. Two cubes of size 12 Å were initialized with the appropriate lattice rotations and set at a distance of 0.5 Å apart (no gap was used for the simulations of perfect crystals). At each face, other than the faces making up the grain boundary, a layer of atoms two cut-off distances thick was placed and constrained to not move along the perpendicular direction. In engineering terms this is known as "rollered" boundary conditions and is used to simulate the material in the bulk by suppressing the Poisson effect. The rollered boundary conditions were used instead of periodic boundary conditions so that the simulation was not limited to lattice orientations that fit in periodic boxes. The atoms were then allowed to relax, and the grain boundary energy was found by taking the total potential energy of the atoms (not including those in the rollered boundary

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

layers) minus the energy of a perfect crystal with the same number of atoms. The same procedure was repeated to find the surface energies, only the two grains were translated many cut-off distances apart before they were relaxed and the energy measurement was taken. The grains were initialized with the same gap so that the same surface cut was used for both the grain boundary energies and the surface energies. The orientations were chosen to explore high index and low index surfaces as well as irrational orientations. The entire calculation of both surface energies and grain boundary energies takes only about 10 minutes to perform.

The results for the measurements of surface energies, grain boundary energies, the corresponding energy barriers, and brittle-to-ductile transition temperatures (if any) are given in table 4.2. The energy barriers given here are lower bounds. The value of $\psi$ which minimizes the energy barrier is always 0. This corresponds to a Burger's vector that is perpendicular to the crack line. Thus the lower bound for $U_b$ is the saddle point in $U_{act}(r, \phi, \psi = 0)$. The minimizing $\phi$ is shown in table 4.2 and appears to always be near 69°. Note that there may not actually be a slip plane at this inclination with a Burger's vector perpendicular to the crack line and the true slip plane and Burger's vector inclinations would give a higher energy barrier. $U_b^{(2)}$, $r_c^{(2)}$, $\phi_{min}^{(2)}$, and $T_{BD}^{(2)}$ are the energy barrier, critical radius, minimizing slip plane inclination, and brittle-to-ductile transition temperature given by equation 4.15 if the $\sin\phi$ factor is not included in the ledge term. These values give estimates of the error in the Rice and Thomson model.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

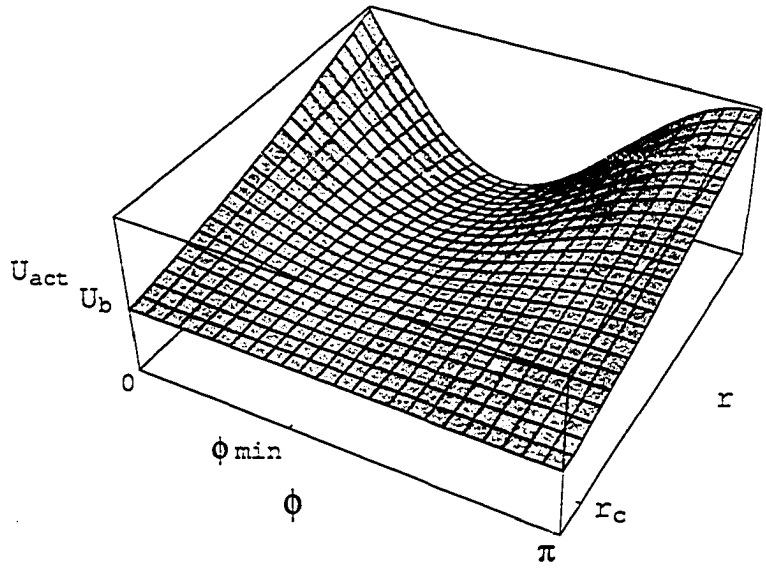



Figure 4.5: $U_{act}(r, \phi)$ **for Grain Boundaries with a Brittle-Ductile Transition** The lower bound of the transition temperature is given by the saddle point in $U_{act}(r, \phi)$ because we are looking for the minimum in $\phi$ and the barrier energy as a function of $r$.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table 4.2: **Surface Energies and Resulting Energy Barriers for Two Slightly Different Models.**

| Material | Orientations | $\gamma_{gb}$ (ergs/cm$^2$) | $U_b$ (eV) | $r_c$ (b) | $\phi_{min}$ | $T_{BD}$ (K) | $U_b^{(2)}$ ($eV$) | $r_c^{(2)}$ (b) | $\phi_{min}^{(2)}$ | $T_{BD}^{(2)}$ (K) |
|---|---|---|---|---|---|---|---|---|---|---|
| Si | $[1,0,0]$ | 2229 | 12.1 | 4.6 | 68.8 | 5798 | 12.7 | 4.7 | 70.5 | 6082 |
| | $[\sqrt{7},1,\frac{\sqrt{5}+1}{2}]$ | 1299 | 52.1 | 11.8 | 69.1 | 24069 | 53.6 | 11.9 | 70.5 | 24777 |
| | $[1,17,83]$ | 1360 | 46.8 | 10.9 | 69.1 | 21679 | 48.2 | 11.1 | 70.5 | 22338 |
| Al | $[1,0,0]$ $[1,1,1]$ | 1059 | Inherently Ductile | | | | Inherently Ductile | | | |
| | $[1,0,0]$ $[\sqrt{7},1,\frac{\sqrt{5}+1}{2}]$ | 689 | | | | | | | | |
| | $[-\sqrt{13},0,1]$ $[\sqrt{7},1,\frac{\sqrt{5}+1}{2}]$ | 778 | | | | | | | | |
| | $[1,19,73]$ $[0,23,107]$ | 926 | | | | | | | | |
| Fe | $[1,0,0]$ $[1,1,1]$ | 1991 | Inherently Ductile | | | | 0.00788 | 1.09 | 70.5 | 4.2 |
| | $[1,0,0]$ $[\sqrt{7},1,\frac{\sqrt{5}+1}{2}]$ | 2131 | | | | | Inherently Ductile | | | |
| | $[-\sqrt{13},0,1]$ $[\sqrt{7},1,\frac{\sqrt{5}+1}{2}]$ | 1923 | | | | | 0.0493 | 1.37 | 70.05 | 26 |
| | $[1,19,73]$ $[0,23,107]$ | 2283 | | | | | Inherently Ductile | | | |
| W | $[1,0,0]$ $[1,1,1]$ | 1864 | 14.5 | 8.2 | 69.2 | 7061 | 15.3 | 8.3 | 70.5 | 7415 |
| | $[1,0,0]$ $[\sqrt{7},1,\frac{\sqrt{5}+1}{2}]$ | 1635 | 24.1 | 10.8 | 69.3 | 11608 | 25.1 | 10.9 | 70.5 | 12074 |
| | $[-\sqrt{13},0,1]$ $[\sqrt{7},1,\frac{\sqrt{5}+1}{2}]$ | 1583 | 27.1 | 11.4 | 69.3 | 12976 | 28.1 | 11.6 | 70.5 | 13472 |
| | $[1,19,73]$ $[0,23,107]$ | 1994 | 10.7 | 7.1 | 69.2 | 5267 | 11.4 | 7.2 | 70.5 | 5534 |

Equation 4.15 gives approximately correct qualitative behavior for the different materials. Silicon and tungsten are very brittle at zero temperature, while aluminum is ductile. We find that iron is the only material for which certain grain boundaries are brittle and certain are ductile, at least using the modified ledge energy. For those grain boundary geometries that are listed as inherently ductile in table 4.2, only a range of $\phi$ gives ductile behavior. For instance, the grain boundary with miller indices $[-\sqrt{13}, 0, 1]$, $[\sqrt{7}, 1, \frac{\sqrt{5}+1}{2}]$ (with the unmodified ledge term) is only ductile if there is a slip plane with an inclination in the range of 60° to 69°. If the true slip plane is outside of this range, the grain boundary will be brittle at zero temperature and have an energy barrier for emission of a dislocation loop.

These results are relevant to future work in atomistic simulation of grain boundary fracture because they suggests that there are no brittle grain boundaries in Al. Therefore atomistic simulations of Al grain boundaries are not computationally feasible. Previous attempts at grain boundary fracture of Al confirm this. The results also suggest that no orientations of Si or W are ductile at zero temperature, meaning that Si and W are good choices of materials for atomistic fracture simulations. The most interesting results are for Fe, the brittleness or ductileness of which may depend on grain boundary geometry. Atomistic simulations of grain boundary fracture may be possible for some geometries and not others. Before performing a lengthy fracture simulation, a quick measure of the surface and grain boundary energies could be done to check whether or not a given grain boundary geometry has a fracture energy near the cutoff. Atomistic simulations of grain boundary fracture in iron could also be used to test the accuracy of the cutoff given by equation 4.15.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

The surface energies found above can be used to find a bound for the grain boundary energies needed to allow for brittle fracture in bimaterial interfaces, specifically Al-Si. Using the lowest surface energies found for Si and Al, and the fracture energy cut off for Al, (since the ductile behavior will occur in the Al half) 263 ergs/cm$^2$, the minimum grain boundary energy which would predict brittle fracture is 1572 ergs/cm$^2$. Since this is in the range of other grain boundaries energies for single element, Al-Si interfaces with such an energy are plausible.

While the qualitative behavior is reasonably predicted by equation 4.15, the energy barriers and corresponding brittle-to-ductile transition temperatures for Si and W are nowhere near experimental values. In fact, they are above the melting temperature for each element. Transition temperatures found with thermal gradient crack arrest experiments for Si, range from 843 K to 1034 K [20]. For W, experiments find a brittle-to-ductile transition temperature ranging from 370 K to 470 K [22]. The crack velocity used to calculate $T_{BD}$ was $1 \times 10^{-15}$ m/s, similar to the experimental values found in Argon's experiments [20]. The transition temperatures for the brittle Fe grain boundaries are actually lower than experimental values. Thermal gradient crack arrest experiments with Fe-3wt.%Si range from 106K to 130K [37]. These experiments found a crack velocity on the order of $1e^{-5}$ m/s. With such a velocity, equation 4.13 actually gives a negative value.

## 4.5 Conclusion

We have found a way of predicting whether a grain boundary or bi-material interface is brittle at zero temperature based on an adaptation of the classic model by Rice and Thomson [38] and a relatively fast atomistic calculation. We have seen that Rice's model has a number of limitations. A better method of calculating the

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

energy barrier of a dislocation loop would involve atomistic measurements rather than continuum approximations. One suggested strategy for calculating the energy barrier is the nudged elastic band method [32]. Two configurations of a cracked cube of atoms, one with a dislocation loop at a radius greater than the estimated critical radius, one without would be initialized. The minimum energy path of configurations between them would be found by the nudged elastic band method. This path would give an atomistic calculation of the energy barrier of a dislocation loop emitted from a crack tip.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

APPENDIX A

# COMMENSURABILITY OF 3D GRAIN BOUNDARIES

## A.1 Introduction

A grain boundary is commensurate if the structure has repeating, long range order, as shown by Wolf [46] this includes not only grain boundaries for which the pair of surface cells constitute a commensurate pair of rectangles, but may include parallelogram surface cells. Note that such a grain boundary cannot be simulated in periodic boundary conditions. A grain boundary can be simulated in periodic boundary conditions if each surface has a unit cell that can be described by two orthogonal surface vectors which have the same distances in both directions. In this appendix, we present an algorithm for finding all possible 3D grain boundaries that can be simulated in periodic boundary conditions, for a simple cubic lattice.

## A.2 Algorithm

The steps needed for systematically finding all commensurate grain boundaries with a surface unit cell with area under a given cutoff, $A_{max}$ are:

1. Find all surfaces with a primitive unit cell with area less than $A_{max}$ with corresponding lattice vectors, $\vec{s_1}$ and $\vec{s_2}$ that define the primitive unit cell.

2. For each surface found in step 1, find all new surface unit cells defined by linear combinations of $\vec{s_1}$ and $\vec{s_2}$.

3. Find all pairs of the surface unit cells found in step 2 that are compatible, allowing for small strains.



Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Given a normal lattice vector $\vec{n}$, we must find the two orthogonal vectors that describe the minimum unit cell of the surface. We find a pair of orthogonal surface vectors by taking the cross product of $\vec{n}$ and a low index lattice vector $\vec{v}$. A pair of orthogonal surface vectors are then given by $\vec{s_1} = \vec{n} \times \vec{v}$ and $\vec{s_2} = \vec{n} \times \vec{s_1}$. The area of the unit cell is then given by $|\vec{s_1}||\vec{s_2}|$. In order to quickly find a pair of orthogonal surface vectors describing a small unit cell area, we try a few different low index lattice vectors for $\vec{v}$, specifically all permutations of $(1,0,0)$ and $(1,1,0)$ that are not equal to $\vec{n}$. In order to guarantee that we have the smallest unit cell, we then loop through all pairs of orthogonal surface vectors for which the largest component of the vector is smaller than the area of the current surface unit cell. If a smaller unit cell is found, the search recurses to find only unit cells that are still smaller. We can extend the whole algorithm for finding commensurate grain boundaries for simple cubic lattices to any lattice with cubic symmetry (FCC, BCC, Diamond) in this step if we search over not only integer vectors, but all lattice vectors for the particular type of lattice we are interested in.

Utilizing a function that returns the orthogonal surface vectors that define the smallest unit cell, we can loop over all normal lattice vectors to find all the surfaces that have a primitive unit cell with an area under a given cutoff. Because of cubic symmetry, we can avoid duplicate surfaces by only looping over miller indices $(i,j,k)$ where $i >= j >= k$ and all $i,j,k$ are positive integers with a greatest common denominator greater than 1.

Once we have a list of all the surfaces that have a unit cell with an area under the given cutoff and the surface vectors that define the primitive unit cell for each surface, we can loop over this list to find all linear combinations of the unit cell vectors that define a new unit cell that is still under our area limit. If the primitve

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

unit cell for a given surface is described by the vectors $\vec{s_1}$ and $\vec{s_2}$), we are looking for $\vec{S_1} = k_1\vec{s_1} + k_2\vec{s_2}$ and $\vec{S_2} = l_1\vec{s_1} + l_2\vec{s_2}$ such that $\vec{S_1} \cdot \vec{S_2} = 0$ and $|\vec{S_1}||\vec{S_2}|$ is less than the given maximum area. Specifically, for each surface, we loop over $k_1$, $k_2$ and $l_2$. In order to avoid finding duplicate linear combinations and to guarantee that $(\vec{S_1}, \vec{S_2}, \vec{n})$ form a set of right handed, orthogonal vectors [show figure of quadrants] we only allow $\vec{S_1}$ to be in the first quadrant or parallel to $\vec{s_1}$ and $\vec{S_2}$ to be in the second quadrant or parallel to $\vec{s_2}$. This translates to loop limits of $(1, \lceil k_{max} \rceil)$ (where $k_{max}$ is $A_{max}/(|\vec{s_1}||\vec{s_2}|)$) for $k_1$ and $l_2$ with $k_2$ having loop limits $(0, \lceil k_{max} \rceil)$. $l_1$ is then set to be

$$l_1 = -\frac{k_2 l_2 |\vec{s_2}|^2}{k_1 |\vec{s_1}|^2}. \tag{A.1}$$

If the resulting area, $|\vec{S_1}||\vec{S_2}|$ is less than $A_{max}$, then the linear combination information is stored in a list.

Once we have a list of all possible surface cells with an area less than $A_{max}$, we can compare each pair to see if they are compatible. First, in order to speed up the search for compatible pairs, we sort the list of surface cells by area. A pair of surface cells will only be compatible if $|A_1 - A_2|/A_avg <= 2 * \epsilon_max + \epsilon_max^2$. This allows us to limit our search more effectively. Then, if both dimensions of the unit cell for both surfaces are within the strain limit, the grain boundary is commensurate and can be simulated in periodic boundary conditions.

Five parameters are necessary to define a 3D grain boundary. One convenient choice of parameters for defining grain boundary geometries are the two normal vectors $\vec{n_1}$ and $\vec{n_2}$ (each constituting two parameters when normalized) and the twist angle, $\phi$. The twist angle is defined to be zero when the grain boundary is a pure tilt grain boundary. These five parameters are enough to specify a grain boundary, however it is also useful to find the tilt axis and tilt angle. The tilt axis

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

is given by [46]

$$\vec{t} = \vec{n_1} \times \vec{n_2} \tag{A.2}$$

and the tilt angle is defined as

$$\psi = \sin^{-1} |\vec{n_1} \times \vec{n_2}|. \tag{A.3}$$

In order to calculate the twist angle we find the rotation matrix that performs only the tilt rotation. Specifically, we find the rotation matrix $R$ that takes the orthogonal basis defined by $\{\vec{t}, -\vec{n_1}, \vec{t}\times\vec{n_1}\}$ to $\{\vec{t}, \vec{n_2}, \vec{t}\times\vec{n_2}\}$. We apply this rotation matrix to $\vec{S_{11}}$ (the first of the pair of surface vectors defining the unit cell for the first surface) to find $\vec{S_{11}}' = R\vec{S_{11}}$. The twist angle is then found by comparing $\vec{S_{11}}'$ to $\vec{S_{21}}$:

$$\phi = \cos^{-1}\left(\frac{\vec{S_{11}}' \cdot \vec{S_{21}}}{|\vec{S_{11}}'||\vec{S_{21}}|}\right) \tag{A.4}$$

## A.3 Commensurate Grain Boundary Results

### A.3.1 Simple Cubic Crystals

## A.4 Code

To illustrate exactly how the algorithm described in section A.2 is implemented, we provide the python code.

```
from math import *
from exgcd import *
from Numeric import *

def FindMinimumOrthogonalSurfaceUnitCell(mi,maxarea):
    # takes miller index and returns surface vectors that represent
    # minimum unit cell for surface we assume that mi[0] is never 0

    vlist=[(0,1,0),(0,1,1),(0,0,1),(1,0,1)]
```

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

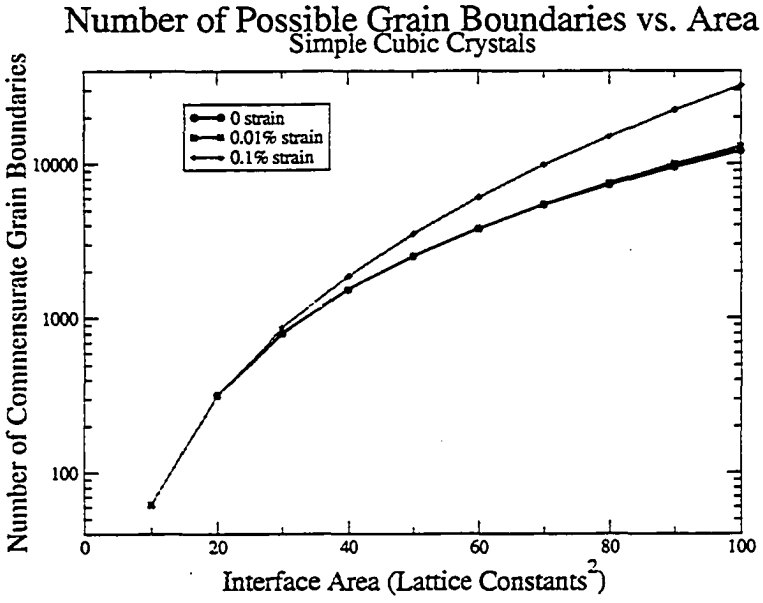



Figure A.1: **Dependence of the Number of Commensurate Grain Boundaries on Interface Area and Allowed Strain** As the allowed strain is increased, the number of commensurate grain boundaries rises more rapidly with maximum interface area.

```
    minarea=1e10

    for v in vlist:
        s1 = dividebyGCD(cross(mi,v))
        s2 = dividebyGCD(cross(mi,s1))
        area = norm(s1)*norm(s2)
        if area < minarea:
            minarea = area
            S1=s1
            S2=s2

    #return S1,S2
    return FindIfSmallerOrthogonalUnitCellUnderCutoff(
        S1,S2,mi,maxarea)

def FindIfSmallerOrthogonalUnitCellUnderCutoff(s1,s2,mi,maxarea):
    # takes surface vectors and finds if there is a smaller unit
    # cell for surface

    area = norm(s1)*norm(s2)
    minarea = area
```

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

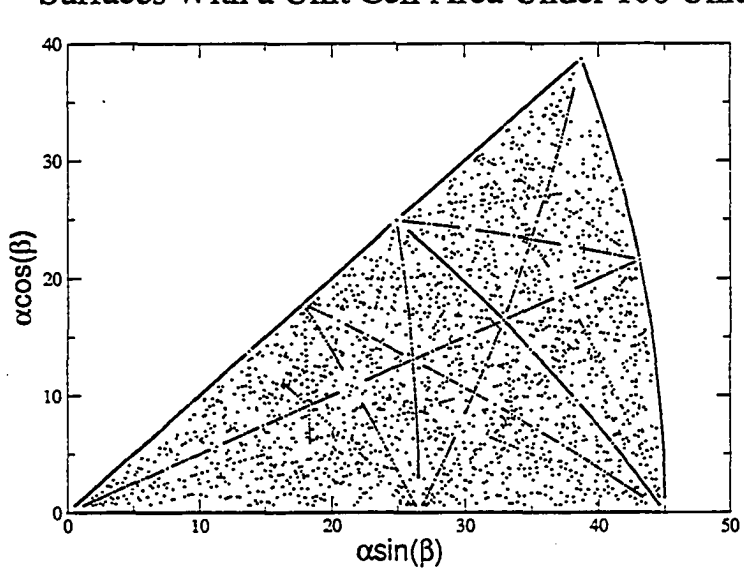



Figure A.2: **Stereographic Projection of Surface Normal Vectors for Which the Surface has a Primitive Unit Cell with Area less than 100 Lattice Constants** The sterographic projection is found using the relation $(i^2 + j^2 + k^2)^{-1/2} * (i, j, k) = (\sin\alpha\cos\beta, \sin\alpha\sin\beta, \cos\alpha)$. In order to plot this particular sterographic projection, we have limited $(i, j, k)$ to $k <= j <= i$.

```
mins1=s1
mins2=s2
limit = min(int(ceil(area)),int(ceil(maxarea)))
for j in xrange(0,limit):
    for k in xrange(0,limit):
        # we assume that mi[0] is never 0
        if (j*mi[1]+k*mi[2]) % mi[0] == 0:
            i = -(j*mi[1]+k*mi[2])/mi[0]
            if not (i==0 and j==0 and k==0):
                S1 = dividebyGCD((i,j,k))
                S2 = dividebyGCD(cross(mi,S1))
                newarea = norm(S1)*norm(S2)
                if newarea - minarea < -.1:
                    if newarea < 0.75*limit:
                        return \
                  FindIfSmallerOrthogonalUnitCellUnderCutoff(\
                            S1,S2,mi,maxarea)
                    else:
                        minarea = newarea
```

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

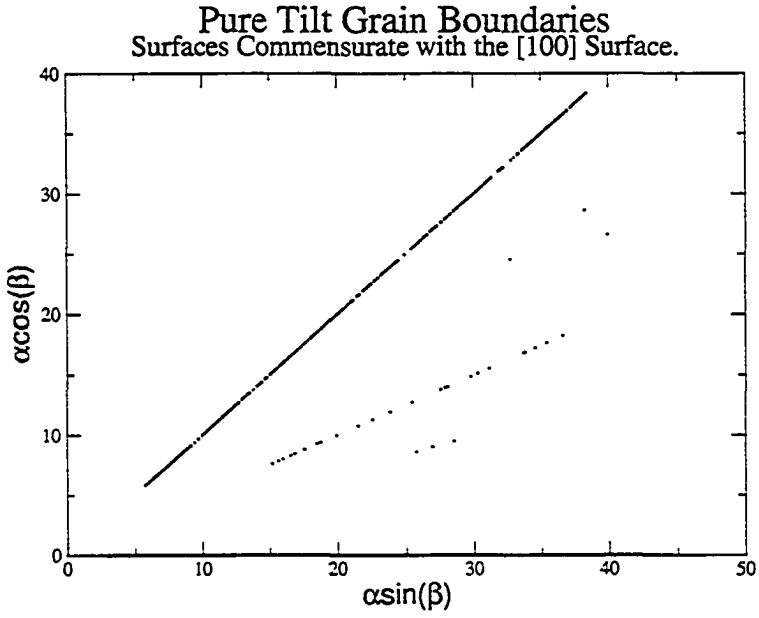



Figure A.3: **Stereographic Projection of Surfaces that are Commensurate with the (100) Surface in a Pure Tilt Grain Boundary.** Since the space of 3D grain boundaries is five dimensional, it is most convenient to examine the results by plotting a 2D cut of the 5D space.

```
                mins1 = S1
                mins2 = S2

    return mins1,mins2

def FindSurfacesOrthoCells(limit):
    geometries = []
    millerIndices = []
    file = open("SurfacesUnder"+repr(limit)+".dat","w")

    # these loop limits create vectors with i >= j >= k and
    # i cannot be 0.
    for i in xrange(1,int(ceil(limit))+1):
        for j in xrange(0,i+1):
            for k in xrange(0,j+1):

                if (i,j,k) != (0,0,0):
                    mi = dividebyGCD((i,j,k))

                    if mi not in millerIndices and norm(mi)<=limit:
                        millerIndices.append(mi)
```

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

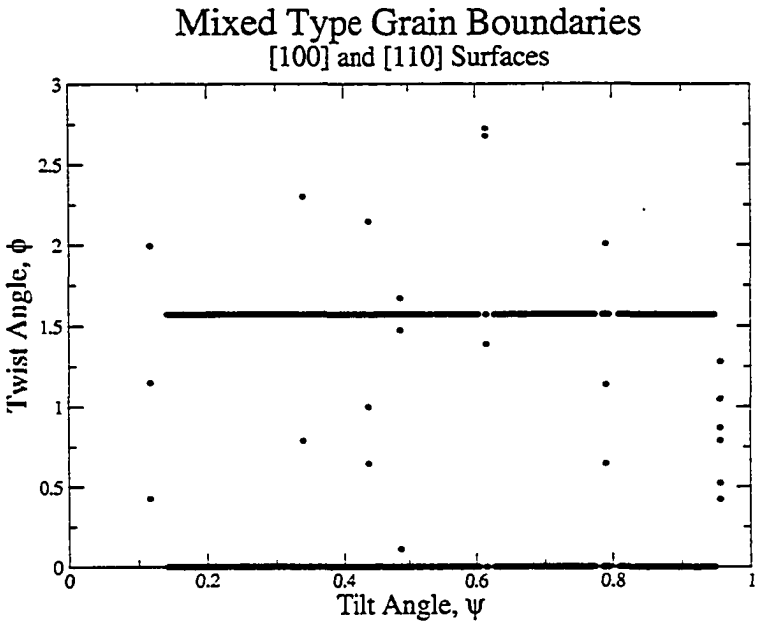



Figure A.4: **Tilt Angles and Twist Angles of All Boundaries between the (100) and (110) surfaces.**

```
                s1,s2 = \
                      FindMinimumOrthogonalSurfaceUnitCell(
                    mi,limit)
                area = norm(s1)*norm(s2)

                if area<=limit:
                    # change to dictionary
                    geometries.append([mi,s1,s2,area])
                    file.write(repr(mi)+" "+repr(s1)+
                               " "+repr(s2)+" "+repr(area)+
                               "\n")

    return geometries

def FindOrthoLinearCombinations(surface,maximumArea):
    # given a set of surface vectors, find all linear combinations
    # of the surface vectors that give a new unit cell with an area
    # under maximumArea

    mi=surface[0]
    a1=array(surface[1])
    a2=array(surface[2])
```

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
    normsqa1 = normsq(a1)
    normsqa2 = normsq(a2)
    unitArea=surface[3]
    limit = float(maximumArea)/unitArea

    linearCombinations = []

    # we only need to search the k1 and k2 positive quadrant for
    # the lc that will constitute A1
    # we include (1,0) in the search, but not (0,0) or (0,1)
    for k1 in xrange(1,int(ceil(limit))+1):
        for k2 in xrange(0,int(ceil(limit))+1):
            # if k1 is positive, l2 can only be positive in order
            # for A2 to be orthogonal to
            # A1 and for the system to be right handed
            for l2 in xrange(1,int(ceil(limit))+1):
                # ensure that k1, k2, l1, and l2 give linear
                # combinations of a1 and a2 that are orthogonal and
                # have integer components
                if (l2*k2*normsqa2) % (k1*normsqa1) == 0:
                    l1 = -l2*(k2*normsqa2)/(k1*normsqa1)
                    A1 = k1*a1+k2*a2
                    A2 = l1*a1+l2*a2

                    crossproduct = cross(A1,A2)
                    area = norm( crossproduct )
                    normA1 = norm(A1)
                    normA2 = norm(A2)

                    if area <= maximumArea:
                        normA1 = norm(A1)
                        normA2 = norm(A2)
                        linearCombinations.append((
                            mi,(k1,k2,l1,l2),list(A1),list(A2),
                            normA1,normA2,area))

    return linearCombinations

def CompareArea(lc1,lc2):

    if lc1[6] > lc2[6]:
        return 1
    elif lc1[6] == lc2[6]:
        return 0
```

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
    else:
        return -1

def FindCommensurateBoundariesOrtho(maximumArea, strainLimit):

    surfaces = FindSurfacesOrthoCells(maximumArea)
    geometries = []
    linearCombinations = []
    macroparams = []
    numLCs = 0

    file = open("BoundariesUnder"+repr(maximumArea)+".dat","w")

    # loop over surfaces and find linear combinations for each
    for i in xrange(len(surfaces)):
        lc = FindOrthoLinearCombinations(surfaces[i],maximumArea)
        linearCombinations.extend(lc)
        numLCs += len(lc)

    linearCombinations.sort(CompareArea)

    for i in xrange(len(linearCombinations)):
        for j in xrange(min(i+1,len(linearCombinations))):

            area1 = linearCombinations[i][6]
            area2 = linearCombinations[j][6]

            if 2*fabs(area2-area1)/(area2+area1) < 3*strainLimit:

                mi1 = linearCombinations[i][0]
                mi2 = linearCombinations[j][0]

                tilt1 = cross(-1*array(mi1),mi2)
                if tilt1 != (0,0,0):
                    tiltAngle = asin( norm(tilt1) / \
                                      (norm(mi1)*norm(mi2)))
                    tiltAxis = dividebyGCD(tilt1)
                else:
                    tiltAngle = 0
                    tiltAxis = (0,0,0)

                A1 = linearCombinations[i][2]
                A2 = linearCombinations[i][3]
                normA1 = linearCombinations[i][4]
```

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
normA2 = linearCombinations[i][5]
B1 = linearCombinations[j][2]
B2 = linearCombinations[j][3]
normB1 = linearCombinations[j][4]
normB2 = linearCombinations[j][5]
area = linearCombinations[j][6]

#if normA1 == normB1 and normA2 == normB2:
if (2*fabs(normA1-normB1)/(normA1+normB1) <=
    strainLimit and
    2*fabs(normA2-normB2)/(normA2+normB2) <=
    strainLimit):
    # twist angle
    twistAngle = CalculateTwistAngle(
        tiltAngle,tiltAxis,mi1,mi2,A1,B1)

    xxstrain = 2*fabs(normA1-normB1)/ \
               (normA1+normB1)
    yystrain = 2*fabs(normA2-normB2)/ \
               (normA2+normB2)

    # check that it's a unique boundary
    if (mi1,mi2,twistAngle) not in macroparams:
        geometries.append([mi1,mi2,(A1,A2),(B1,B2),
                           tiltAxis,tiltAngle,
                           twistAngle,xxstrain,
                           yystrain,area])
        macroparams.append((mi1,mi2,twistAngle))
        file.write(repr([mi1,mi2,tiltAxis,
                         tiltAngle,twistAngle,
                         xxstrain,yystrain,area])+
                   "\n")

#if normA1 == normB2 and normA2 == normB1:
if (2*fabs(normA1-normB2)/(normA1+normB2) <=
    strainLimit and
    2*fabs(normA2-normB1)/(normA2+normB1) <=
    strainLimit):
    # twist angle
    twistAngle = CalculateTwistAngle(
        tiltAngle,tiltAxis,mi1,mi2,A1,B2)

    xxstrain = 2*fabs(normA1-normB2)/ \
               (normA1+normB2)
```

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
                    yystrain = 2*fabs(normA2-normB1)/ \
                               (normA2+normB1)

                    # check that it's a unique boundary
                    if (mi1,mi2,twistAngle) not in macroparams:
                        geometries.append([mi1,mi2,(A1,A2),(B1,B2),
                                           tiltAxis,tiltAngle,
                                           twistAngle,xxstrain,
                                           yystrain,area])
                        macroparams.append((mi1,mi2,twistAngle))
                        file.write(repr([mi1,mi2,tiltAxis,
                                         tiltAngle,twistAngle,
                                         xxstrain,yystrain,area])+
                                   "\n")

            else:
                j=len(linearCombinations)

    return geometries

def CalculateTwistAngle(tiltAngle,tiltAxis,mi1,mi2,surface1,
                        surface2):
    if tiltAngle != 0:
        t = normalize(tiltAxis)
        m1 = normalize(-1*array(mi1))
        s1 = cross(t,m1)
        m2 = normalize(mi2)
        s2 = cross(t,m2)

        #r1 rotates (t,m2,s2) to x,y,z
        r1 = array((t,m2,s2))
        #r2 rotates x,y,z to t,m1,s1
        r2 = transpose(array((t,m1,s2)))
        #r3 rotates t,m2,s2 to t,m1,s1
        r3 = matrixmultiply(r2,r1)

        # rotate first surface vector
        A = array(surface1)
        B = array(surface2)
        Bprime = matrixmultiply(r3,B)
    else:
        A = array(surface1)
        Bprime = array(surface2)
```

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
    cosangle = dot(A,Bprime)/(norm(Bprime)*norm(A))
    if fabs(fabs(cosangle) - 1.0) < 1e-3:
        twistAngle = 0
    else:
        twistAngle = acos(cosangle)

    return twistAngle

#####################################################################

limit = 100
strainLimit = 0.01
FindCommensurateBoundariesOrtho(limit,strainLimit)
```

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

# APPENDIX B

## MDWEBSERVICES CODE: GRAINBREAKER AND OFEMD

The software used in the simulations described in chapters 1, 2, and 3 are included in a package called *MDWebServices* and can be downloaded from `http://www.lassp.cornell.edu/sethna/DM/mdwebservices/`. *MDWebServices* contains two python packages (*GrainBreaker* and *OFEMD*) as well as their shared infrastructure. The name *MDWebServices*, reflects that both packages were originally designed to be accessed via a WebService [14] interface. Both packages can also be accessed by a simple command line interface as described in chapter 3 or driven by separate python scripts that utilize the subroutines. Both packages use *DigitalMaterial* [7] as the core for the molecular dynamics simulations.

*GrainBreaker* measures cohesive laws for either 2D or 3D grain boundaries and was used for the systematic study of 2D grain boundaries in chapter 1 and the 3D grain boundary simulations in chapter 2. The method that *GrainBreaker* implements is described in detail in chapter 1. It can simulate any material for which there is an interatomic potential implemented in *DigitalMaterial*, any grain boundary geometry (pair of lattice rotations), and either fixed, rollered, or periodic boundary conditions (provided the user has provided a set of lattice rotations that constitute a commensurate grain boundary as described in appendix A).

*OFEMD* was used for the fully atomistic simulations of the cube in cube model in chapter 2 and is described in detail in chapter 3. The code that retrieves the mesh information from a relational database or XML repository is implemented as two python extensions to the finite element library (FEMLib) written by Paul Wawrzynek. These two files, XMLFemModel.py and DBFemModel.py are also available from `http://www.lassp.cornell.edu/sethna/DM/mdwebservices/`.



Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

# BIBLIOGRAPHY


[1] Portable batch system. Open Source Software Package. http://www.openpbs.org/.

[2] M. P. Allen and D. J. Tildesley. *Computer Simulation of Liquids.* Oxford Science Publications, 1987.

[3] A. S. Argon. *Acta. metall.*, 35:185, 1987.

[4] A. S. Argon. *Eng. Mat. and Tech.*, 123:1, 2001.

[5] Nicholas Bailey. *Digital Material: Modern Molecular Dynamics Applied to Dislocation Mobility, Notched Fracture Initiation and Adaptive Software.* PhD thesis, Cornell, 2002.

[6] Nicholas Bailey and Thierry Cretegny. Private communication, 2005.

[7] Nicholas Bailey, Thierry Cretegny, James P. Sethna, Valerie R. Coffman, Andrew J. Dolgert, Christopher R. Myers, Jakob Schiotz, and Jens Jorgen Mortensen. Digital material: a flexible atomistic simulation code, 2006.

[8] M. I. Baskes. Modified embedded-atom potentials for cubic materials and impurities. *Phys. Rev. B*, 46:2727, 1992.

[9] M. I. Baskes, J. S. Nelson, and A. F. Wright. Semiempirical modified embedded-atom potentials for silicon and germanium. *Phys. Rev. B*, 40:6085–6099, 1989.

[10] M. Z. Bazant, E. Kaxiras, and J. F. Justo. Environment-dependent interatomic potential for bulk silicon. *Phys. Rev. B*, 56:8542, 1997.

[11] G. H. Bishop and B. Chalmers. A coincidence - ledge - dislocation description of grain boundaries. *Scripta Metallurgica*, 2:133–140, 1968.

[12] W. Bollmann. *Crystal Defects and Crystalline Interfaces.* Springer-Verlag, New York, 1970.

[13] G. T. Camacho and M. Ortiz. Computational modelling of impact damage in brittle materials. *International Journal of Solids and Structures*, 33:2899–2938, 1996.

[14] Ethan Cerami. *Web Services Essentials.* O'Reilly Media, 2002.

[15] S. P. Chen, D. J. Srolovitz, and A. F. Voter. Computer simulation on surfaces and [001] symmetric tilt grain boundaries in ni, al, and $ni_3al$. *Journal of Materials Research*, 4:62, 1989.

[16] J. deLaunay. *Solid State Phys.*, 2:220, 1956.




Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

[17] C. Scandian et. al. *Phys. Stat. Sol. (a)*, 171:67, 1999.

[18] M. Falk, A. Needleman, and J. Rice. A critical evaluation of dynamic fracture simulations using cohesive surfaces, 2001.

[19] Y. I. Frenkel and Kontorova T. *Zhurnal tekhnicheskoi fiziki*, 8:1340, 1938.

[20] B. J. Gally and A. S. Argon. *Philos Mag.*, 81:699, 2001.

[21] Arne Gullerud, Kyle Koppenhoefer, Arun Roy, Sushovan RoyChowdhury, Matt Walters, Barron Bichon, Kristine Cochran, Adam Carlyle, and Robert H. Dodds Jr. Warp3d-release 15.6: 3d dynamic nonlinear fracture analysis of solids using parallel computers and workstations. Technical report, University of Illinois at Urbana-Champaign, 2006.

[22] P. Gumbsch. *J. of Nuc. Mats.*, 323:304, 2003.

[23] R. J. Harrison, G. A. Bruggeman, and G. H. Bishop. *Grain Boundary Structure and Properties*, chapter Computer Simulation Methods applied to Grain Boundaries, pages 45–91. Academic Press Inc., London, 1976.

[24] Gerd Heber and Jim Gray. Supporting finite element analysis with a relational database backend; part i: There is life beyond files. Technical report, Microsoft Research, 2005.

[25] Gerd Heber and Jim Gray. Supporting finite element analysis with a relational database backend part ii: Database design and access. Technical report, Microsoft Research, 2006.

[26] Gerd Heber, Chris Pelkie, Andrew Dolgert, Jim Gray, and David Thompson. Supporting finite element analysis with a relational database backend; part iii: Opendx - where the numbers come alive. Technical report, Microsoft Research, 2005.

[27] John Price Hirth and Jens Lothe. *Theory of Dislocations*. John Wiley & Sons, New York, 1982.

[28] B. L. Holian, A. F. Voter, N. J. Wagner, R. J. Ravelo, S. P. Chen, W. G. Hoover, C. G. Hoover, J. E. Hammerberg, and T. D. Dontje. Effects of pairwise versus many-body forces on high-stress plastic deformation. *Physical Review A*, 43:2655, 1991.

[29] E. Iesulauro, A. R. Ingraffea, S. Arwade, and P. A. Wawrzynek. Simulation of grain boundary decohesion and crack initiation in aluminum microstructure models. In W. G. Reuter and R. S. Piascik, editors, *Fatigue and Fracture Mechanics*, volume 33, West Conshohocken, PA, 2002. American Society for Testing and Materials.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

[30] E. Iesulauro, A. R. Ingraffea, G. Heber, and P. A. Wawrzynek. A multi-scale modeling approach to crack initiation in aluminum polycrystals. In *44th AIAA/ASME/ASCE/AHS Structures, Structural Dynamics, and Materials Conference*, Norfolk, VA, April 2003. AIAA.

[31] Yoichi Ishida. *Grain Boundary Structure and Properties*, chapter The Bubble Raft as a Model for Grain Boundary Structure, pages 93–106. Academic Press Inc., London, 1976.

[32] H. Jonson, G. Mills, and K. W. Jacobsen. *Classical and Quantum Dynamics in Condensed Phase Simulations*, chapter Nudged elastic band method for finding minimum energy paths of transitions, page 385. World Scientific, Singapore, 1998.

[33] J. F. Justo, M. Z. Bazant, E. Kaxiras, V. V. Bulatov, and S. Yip. Interatomic potential for silicon defects and disordered phases. *Phys. Rev. B*, 58:2539, 1998.

[34] A. Needleman. An analysis of tensile decohesion along an interface. *Journal of the Mechanics and Physics of Solids*, 38(3):289–324, 1990.

[35] G. Palumbo and K. T. Aust. *Materials Interfaces: Atomic-level structure and properties*, chapter Special Properties of $\Sigma$ grain boundaries., pages 190–207. Chapman & Hall, 1992.

[36] P. H. Pumphrey. *Grain Boundary Structure and Properties*, chapter Special High Angle Grain Boundaries, pages 139–200. Academic Press Inc., London, 1976.

[37] Y. Qiao and A. S. Argon. *Mech. Mat.*, 35:903, 2003.

[38] J. R. Rice and R. Thomson. *Philos Mag.*, 29:73, 1974.

[39] F. Sansoz and J. F. Molinari. Incidence of atom shuffling on the shear and decohesion behavior of a symmetric tilt grain boundary in copper. *Scripta Materialia*, 50:1283–1288, 2004.

[40] F. Sansoz and J. F. Molinari. Mechanical behavior of $\sigma$ tilt grain boundaries in nanoscale cu and al: A quasicontinuum study. *Acta Materialia*, 53:1931–1944, 2005.

[41] O. A. Shenderova, D. W. Brenner, A. Omeltchenko, X. Su, and L. H. Yang. Atomistic modeling of the fracture of polycrystalline diamond. *Phys. Rev. B*, 61:3877–3888, 2000.

[42] Douglas E. Spearot, Karl I. Jacob, and David L. McDowell. Non-local separation constitutive laws for interfaces and their relation to nanoscale simulations. *Mechanics of Materials*, 36:825–847, 2004.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

[43] F. H. Stillinger and T. A. Weber. Computer simulation of local order in condensed phases of silicon. *Phys. Rev. B*, 31:5262, 1985.

[44] Viggo Tvergaard and John W. Hutchinson. The relation between crack growth resistance and fracture process parameters in elastic-plastic solids. *Journal of the Mechanics and Physics of Solids*, 40(6):1377–1397, 1992.

[45] D. H. Warner, F. Sansoz, and J. F. Molinari. Atomistic based continuum investigation of plastic deformation in nanocrystalline copper. *International Journal of Plasticity*, 22:754–774, 2006.

[46] D. Wolf *aterials Interfaces: Atomic-level structure and properties*, chapter Atomic-level geometry of crystalline interfaces, pages 1–52. Chapman & Hall, 1992.

[47] D. Wolf and J. A. Jaszczak. *Materials Interfaces: Atomic-level structure and properties*, chapter Role of interface dislocations and surface steps in the work of adhesion, pages 662–690. Chapman & Hall, 1992.

[48] D. Wolf and K. L. Merkle. *Materials Interfaces: Atomic-level structure and properties*, chapter Correlation between the structure and energy of grain boundaries in metals, pages 88–150. Chapman & Hall, 1992.

[49] X. P. Xu and A. Needleman. Numerical simulations of fast crack groth in brittle solids. *Journal of the Mechanics and Physics of Solids*, 42(9):1397–1434, 1994.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.