9-2018

Crystal Misorietation and Defect Generation during Contact between Two Aluminum Substrates

Milad Khajehvand
Santa Clara University, mkhajehvand@scu.edu

Recommended Citation
Khajehvand, Milad, "Crystal Misorietation and Defect Generation during Contact between Two Aluminum Substrates" (2018). *Mechanical Engineering Master's Theses*. 24.
https://scholarcommons.scu.edu/mech_mstr/24

SANTA CLARA UNIVERSITY

DEPARTMENT OF MECHANICAL ENGINEERING

Date: September 14th, 2018

I HEREBY RECOMMEND THAT THE THESIS PREPARED UNDER MY SUPERVISION BY

MILAD KHAJEHVAND

ENTITLED

CRYSTAL MISORIENTATION AND DEFECT GENERATION DURING CONTACT BETWEEN TWO ALUMINUM SUBSTRATES

BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE

OF

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

09/10/18

Thesis Adviser: Dr. Panthea Sepehrband

Date

Robert A. Marks

9/11/2018

Thesis Reader: Dr. Robert Marks

Date

Sept. /10/2018

Chairman of Department: Dr. Drazen Fabris

Date

CRYSTAL MISORIENTATION AND DEFECT GENERATION DURING CONTACT BETWEEN TWO ALUMINUM SUBSTRATES

By

MILAD KHAJEHVAND

MASTER THESIS

Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science
in Mechanical Engineering
in the School of Engineering at
Santa Clara University, September 2018

Santa Clara, California

Abstract

With the use of molecular dynamics simulations, the nanoscale contact between two aluminum substrates at room temperature is studied. The so-called jump-to-contact (JC) phenomenon between (111) surfaces is observed to occur for interfacial distances below 5.8 ± 0.05 Å. This critical distance is approximately equal to 2.5 times the interplanar spacing of (111) planes in a perfect aluminum crystal at 300 K. The critical distance for JC is shown to be higher at higher contact temperatures because of the higher amplitude of atomic vibrations. The effect of employed interatomic potential on the critical distance is also discussed.

The critical distance for JC is shown to be independent of crystallographic misorientation between substrates. However, the final distribution of crystallographic defects at the bonded interface is controlled by the misorientation angle. Three ranges of misorientation angles of (I) $0° < \theta < 21.79°$, (II) $21.79° < \theta < 38.21°$, and (III) $38.21° < \theta < 60°$ are found to be important with regard to the distribution of crystallographic defects. The initial gap size between substrates is shown to affect the density of crystallographic defects in ranges (I) and (III) of misorientation, where dislocations and point defects exist at the interface. On the contrary, the final density of defects in range (II) is independent of initial separation between surfaces since the whole interface is a typical planar defect.

To my Mom and Dad,

for their endless support.

Acknowledgements

I would like to express my sincere thanks to my adviser, Dr. Panthea Sepehrband, for her continuous encouragement, guidance, and support throughout the course of my master’s degree. I would also like to thank Dr. Robert Marks, for being the second reader of this thesis.

This work was supported by the National Science Foundation (CMMI-1728652), and the School of Engineering at Santa Clara University.

Table of Contents

List of Tables

List of Figures

Nomenclature

γ	Grain boundary energy per unit area
ϵ	Depth of potential well in Lennard-Jones 12-6 potential
θ	Misorientation angle
ρ	Probability density
ϕ_{LJ}	Lennard-Jones 12-6 potential
ϕ_{ij}	Two-body term in potential energy formula
a	Lattice constant
a_i	Acceleration of atom i
a_n	Acceleration of atom at the beginning of timestep n
D	Deviation parameter
D_{ave}	Average of deviation parameter in the last 2 ps of simulations
d	The distance at which the Lennard-Jones 12-6 potential is zero, and Interplanar spacing of (111) planes in Al at 300 K
E	Total Energy
F_i	Forces applied on atom i
f	A function which represents an approximation of the electron density
G_i	Energy to embed an atom i in a uniform electron gas of density $\sum_{j\neq i} f_{ij}(r_{ij})$
$\langle H \rangle$	Average of quantity H
h	Interfacial distance
K	Kinetic energy
k_B	Boltzmann constant

L	Length of simulation cell
$\mathcal{L}$	A constant which is equal to $3N+1$
m_i	Mass of atom i
N	Number of atoms, and
	Number of neighbors for an atom
n	Periodicity
P_s	Momentum associated with variable s in Nose-Hoover thermostat
p	Momenta of an atom
Q	Degree of coupling to the heat bath in Nose-Hoover thermostat
r	Position of an atom
s	A variable that couples the system to a heat bath in Nose-Hoover thermostat
T_s	Temperature of heat bath in Nose-Hoover thermostat
t	Time
Δt	Timestep
U	Potential energy
v_1	One-body term in potential energy formula
v_3	Three-body term in potential energy formula
v_n	Velocity of atom at the beginning of timestep n
x_n	Position of atom at the beginning of timestep n

1 Introduction

Understanding contact behavior between metallic surfaces is essential to solve many problems present in technological applications that involve phenomena such as adhesion, friction, wear and indentation. Studying contact behavior at the nanoscale is particularly important since the first contact occurs at this length scale. Nanoscale contact science can also shed light on some fundamental issues at larger length scales, for instance, issues related to microelectromechanical systems (i.e., MEMS) since they are only a few micrometers in size. Molecular dynamics (MD) simulation is a powerful tool for studying the aforementioned phenomena at the nanoscale [1]. In the MD method, Newton's equations of motion are solved to screen the motion of atoms. The MD method enables one to model the dynamic properties of materials [2].

It has been shown that when two metallic surfaces are brought toward each other, at some critical distance, atoms on the surfaces jump together [3]. The occurrence of jump-to-contact (JC) has been observed in several computational studies [1, 5, 9–12] as well as experimental works using scanning tunneling microscopy [3,4,9], mechanically controllable break junctions [4,10,11], and atomic force microscopy [12]. JC phenomena have been extensively studied before, however, the effect of misorientation between surfaces on the critical distance below which JC occurs has not been investigated. Another important point which should be taken into consideration is that the size of the initial gap between surfaces may affect the defect generation following the JC. The effect of initial gap on the final distribution and density of crystallographic defects at the interface is important since defects affect the mechanical and electrical properties of materials. In addition, defects control diffusional mechanism and phase transformation [13–15] that may occur at a later time.

In the present study, utilizing MD simulations, the effect of misorientation between two aluminum (111) surfaces on the critical distance for JC is investigated. A brief discussion of the effect of contact temperature and interatomic potential on this critical distance is also presented. Moreover, the arrangement of atoms at the interface is analyzed to determine the effect of misorientation and initial interfacial distance on the distribution and density of crystallographic defects.

2 Background and Objectives

2.1 Jump-to-contact phenomenon

Originally, it was believed that metallic surfaces can be brought together in a continuous fashion until contact occurs between them. The reason for this common belief was based on the fact that atoms are more firmly attached to their neighbor atoms than attracted by the atoms on the opposite surface [16]. But, it was shown later that atoms on the opposing surfaces can jump together when the interfacial distance falls below a critical value. This critical value is determined by a competition between decrease in energy due to contact between free surfaces across the interface, and increase in energy due to free surfaces being elastically pulled away from their corresponding substrates [6]. The so-called jump-to-contact (JC) phenomenon, also known as avalanche-in-adhesion [6], was first suggested by Gimzewski and Moller [3]. JC is an atomic-scale mechanical instability which occurs as a result of short-range attractive forces between atoms on the surface layers of the two substrates. The majority of significant developments pertaining to this area of study occurred between 1987 and 1993. This is probably due to the invention of scanning tunneling microscope in 1981 by Binnig and Rohrer, for which they received the Nobel Prize in Physics in 1986. Sections 2.1.1 and 2.1.2 discuss the experimental and computational studies of the JC phenomenon, respectively.

2.1.1 Experimental studies

JC has been observed experimentally by scanning tunneling microscopy (STM) [3,4,9], mechanically controllable break junctions (MCBJ) [4,10,11], and atomic force microscopy (AFM) [12]. In most of these studies, the evidence for JC occurrence is witnessing a jump in the conductance-displacement or force-displacement curve while the two electrodes or substrates are brought closer to each other. The rest of this section includes discussion of some of these studies.

Let's consider a situation where two metallic surfaces are being brought toward each other. At sufficiently large distances, one observes a tunneling current between them which decreases exponentially with the distance. Yet, Gimzewski and Moller [3] showed that when the two surfaces are brought closer together, at some point, a JC occurs and after formation of contact, a staircase-like conductance curve is observed due to the atomic structure. The transition from the tunneling regime (smooth exponential distance dependent) to the contact regime (staircase-like) on the conductance-distance curve is where JC occurs [16]. Gimzewski and Moller [3] were the first researchers that used STM (mounted in an ultrahigh vacuum chamber) to study the point contact of a clean Ir tip with a polycrystalline Ag substrate. They showed that at gap spacings less than 3 Å, a significant decrease in the apparent tunnel barrier height is observed, just before the touching [3]. Later, Agrait et al [9] also observed JC occurrence employing STM at low temperatures. They even went further and proved that by pressing the tip into the sample after JC occurrence, a series of jumps in the conductance at integer multiples of $2e^2/h$ can be observed, where e is the charge on an electron, and h is the Planck's constant. They related these jumps to atomic rearrangements [9].

JC has also been observed in low temperature MCBJ experiments [6, 7]. A schematic of the MCBJ technique is shown in figure 2.1. This technique is performed in vacuum, where an elastic substrate is mounted in a three-point bending configuration between two fixed counter supports and a stacked piezo-element. When the substrate is bent, by moving the piezo-element forward using a mechanical gear arrangement, the wire (i.e., metal of interest) breaks at the notch, which was made in advance. Then, the fractured surfaces are brought toward each other by relaxing the force on the substrate, where the piezo-element is used for fine control, and conductance (or resistance) is recorded as a function of voltage of piezo-element during this time. The MCBJ technique has two main advantages compared to STM. Firstly, the surfaces remain clean for the contact since they were recently fractured and were kept in vacuum. The second advantage is the stability of the two electrodes relative to each other and the fact that the strain is concentrated near the contact area (the unglued section shown by u in figure 2.1) [16]. Krans et al [10] used this technique to study the transition between tunneling and contact regimes. As shown in figure 2.2, they observed occurrence of JC when two MCBJ samples are brought together, for both Pt and Al samples. They also observed a faster-than-exponential behavior in the tunneling regime prior to JC. Furthermore, they showed that the conductance value for a one-atom contact is very close to $2e^2/h$. This conductance value of $2e^2/h$ corresponds to a resistance of $12.9\ \text{k}\Omega$ [10]. Voets et al [11] also observed similar results utilizing the MCBJ technique. They reported that JC occurrence is inevitable, even in very rigid (with high stability) MCBJ configurations. Additionally, they reported that the critical separation for JC depends on the wire material and is estimated to vary from 1.5 Å for Pt to 1.9 Å for Al and Au [11].

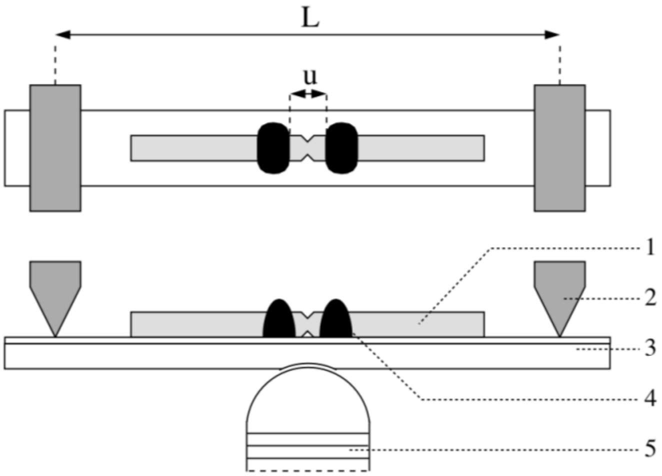


Figure 2.1: Schematic top and side view of the mounting of a MCBJ, with the notched wire (1), two fixed counter supports (2), bending beam (3), drops of epoxy adhesive (4) and the stacked piezo element (5) (Reproduced from [16], with the permission of Elsevier: https://doi.org/10.1016/S0370-1573(02)00633-6).

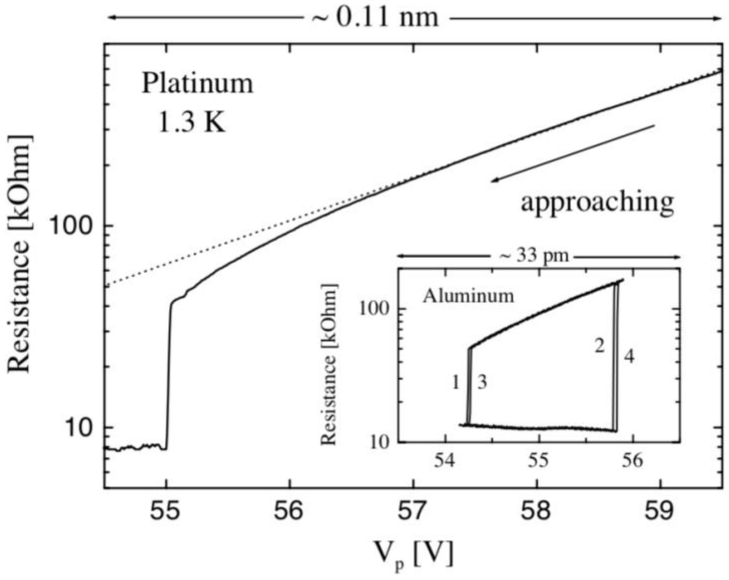


Figure 2.2: Semi-logarithmic plot of an ac-resistance measurement of a Pt MCBJ sample at 1.3 k, as a function of piezo voltage, Vp. Between ~500 kΩ and ~40 kΩ, a downward deviation from exponential behavior is observed, followed by a jump to a stable value. The inset shows two successive cycles of an Al MCBJ sample at 4.2 K, illustrating the hysteresis of the jumps. The numbers indicate the sequential order of the jumps, first up (1), then down (2) and up and down again (3 and 4) (Reproduced from [16], with the permission of Elsevier: https://doi.org/10.1016/S0370-1573(02)00633-6; Permission is also taken from the American Physical Society: https://doi.org/10.1103/PhysRevB.48.14721 which is the original source [10] of the material).

Landman et al [12], used AFM and MD to investigate the mechanisms of adhesion, nanoindentation, separation, and fracture that take place when a Ni tip interacts with an Au substrate. They showed that at small distances between the Ni tip and the Au substrate, JC occurs. They observed that this atomic-scale instability occurs through a quick process where Au atoms in the region of the surface under the Ni tip move approximately 2 Å toward the tip in a short time span of ~ 1 ps [12].

It is also worth mentioning that JC is shown not to occur for some metals. In case JC does not occur, the conductance-displacement curve follows a smooth transition from tunneling regime to the contact regime. Untiedt et al [4], showed that JC does not occur for W, while it always occurs for Au, Pt and Ag. They also reported that Ir and Ni present both behaviors. Their STM, MCBJ, and MD results were all in agreement. This means that JC not only depends on the nature of the bonding in material, but also depends on the elastic properties of material. In other words, softer metals such as Ag and Au always present a JC but this is not the case for more brittle ones like W [4].

2.1.2 Computational studies

In 1988, Pethica and Sutton [5], using continuum and atomistic calculations, showed that at sufficiently small separations $\sim 1 - 2$ Å, surfaces jump together, irrespective of the rigidity with which they are mounted. In their atomistic calculations, they used Lennard-Jones potential which has its own limitations when it comes to studying phenomena in metals. However, their model (see figure 2.3) had two features which are still being used in most of the computational studies in this field. Firstly, they used periodic boundary conditions along the two directions that are parallel to the interface (i.e., x and y directions in figure 2.3); and secondly, they kept the outer shoulders rigid [5].

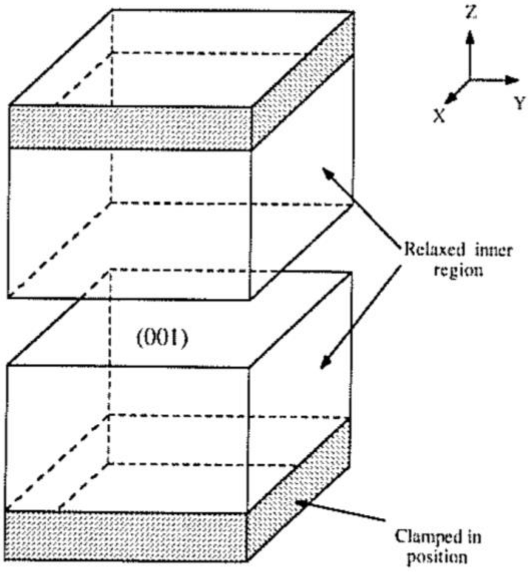


Figure 2.3: Atomistic simulation model for studying JC. Shaded areas are the non-relaxed (fixed) atomic layers (Reproduced from [5], with the permission of the American Vacuum Society: https://doi.org/10.1116/1.575577).

Later, Smith et al [6] performed a many-atom investigation of the avalanche effect between two flat Ni (100) surfaces. They showed that the time that it takes for the avalanche to take place is equal to the time it takes for a sound wave to travel an interplanar spacing, i.e., ~100 fs. Besides, they proved that by increasing the number of relaxed (i.e., Newtonian) layers, the critical distance for avalanche increases. They also showed that the lack of registry between surfaces and high stiffness of surfaces will inhibit the avalanche effect [6].

One can find many computational studies in the literature which investigate the contact between metallic surfaces with nanoscale asperities. For instance, in studies [1, 11, 12], a flat substrate is brought toward another substrate which has an asperity on top of it, contact is formed, compression is made, and then the flat substrate is displaced in the opposite direction until the two substrates are completely detached from one another. The main purpose of these studies was investigating the material transfer and evolution of crystallographic defects during the plastic deformation that occurs as a result of loading and unloading. Nonetheless, they all observed the occurrence of JC as a jump on the force-displacement curve just prior to the contact between an asperity and a flat surface.

In 2010, Lu et al [17] demonstrated experimentally that it is possible to form a contact between two ultrathin single-crystalline gold nanowires at low temperature under ultrahigh-vacuum. They showed that this contact can occur easily and promptly in presence of matching crystallographic orientations [17]. Therefore, the crystallographic misorientation is expected to affect the JC behavior, and when JC occurs and substrates with different crystallographic misorientations reach together, something like a grain boundary is formed. The next section discusses the grain boundaries.

2.2 Grain boundaries

A polycrystalline material is comprised of many single crystals (i.e., grains) which are bonded together at grain boundaries. In other words, a grain boundary is the interface between two grains with different crystallographic orientations. Grain boundaries are 2D crystallographic defects, and their structure and energy can significantly affect the bulk properties of the polycrystalline materials [18,19].

2.2.1 Tilt and twist boundaries

A shared lattice vector exists between the two crystal lattices that meet at the grain boundary plane. In tilt boundaries, this shared lattice vector is parallel to the grain boundary plane, whereas in twist boundaries, the shared lattice vector is perpendicular to the grain boundary plane. Tilt boundaries can be divided into symmetric and asymmetric, where in symmetric tilt boundaries, the crystallographic directions of the two crystal lattices are mirrored with respect to the grain boundary plane. Grain boundaries, in general, exhibit both tilt and twist characters [19].

2.2.2 Coincidence site lattice model

The coincidence site lattice (CSL) model is a geometrical concept which proposes that at particular misorientation angles between the two crystal lattices meeting at the grain boundary plane, some lattice sites coincide with each other. The grain boundaries associated with those particular misorientation angles are called CSL boundaries. In this context, a Σ value is assigned to each of the CSL boundaries, which is equal to the inverse density of coincident sites for that misorientation angle. For instance, a $\Sigma 3$ CSL boundary has $1/3$ of the lattice sites in coincident sites. CSL framework is very important and useful mainly because of the following reasons [19–22]:

- Many so-called "special" (i.e., low energy) grain boundaries are the low-Σ CSL boundaries, as will be further discussed in the next section
- The CSL framework simplifies the use of periodic boundary conditions in computer simulations

2.2.3 Structure and energy of grain boundaries

In general, the grain boundary energy increases as the misorientation between the two crystal lattices increases, until it becomes approximately constant, at a threshold value. Using this value, grain boundaries are classified into low-angle and high-angle grain boundaries. The grain boundary energy (per unit area) of low-angle tilt boundaries (i.e., composed of an array of parallel edge dislocations) can be described by the well-known Read-Shockley model as [23]:

$$\gamma = E_0 \theta (A - \log \theta) \ , \tag{2.1}$$

where θ is the misorientation angle and A and E_0 are two constants. The grain boundary energy (per unit area) of low-angle twist boundaries (i.e., composed of a hexagonal network of perfect dislocations) in an FCC crystal can be derived as [24]:

$$\gamma = E_0 \theta (A + B\theta - \log \theta) \ , \tag{2.2}$$

where A and E_0 are the same constants as in equation 2.1 and B is another constant. The basic assumption in deriving equations 2.1 and 2.2 is that the dislocations are widely separated. Therefore, they fail to predict the energy of high-angle grain boundaries since the distance between dislocations decreases as misorientation increases and their linear elastic fields start to interact with each other [20].

The CSL framework and its associated Σ notation can be used as an alternative approach for studying high-angle grain boundaries. In this context, low-Σ boundaries (i.e., high density of coincident sites) are important because some of them may be "special" grain boundaries [20]. Special grain boundaries have properties which are very different from general high-angle grain boundaries (e.g., uniform structure, low energy, etc.). Figure 2.4 shows the plot of energy of <100> and <110> symmetric tilt boundaries in Al, as a function of misorientation angle, where some low-Σ boundaries are indicated on the plot [19]. As it is shown in the figure, only those low-Σ boundaries that present a cusp on the energy curve are considered "special" boundaries.

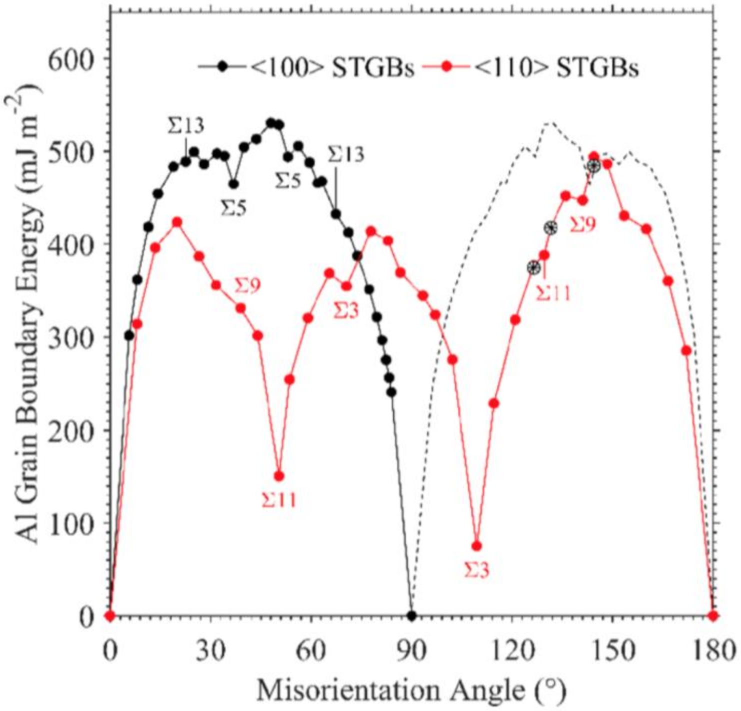


Figure 2.4: Calculated grain boundary energies as a function of misorientation angle for <100> and <110> symmetric grain boundaries in Al. The reference plane for misorientation angles is a {100} plane for both tilt systems (Reproduced from [19]: https://doi.org/10.1186/s40192-015-0040-1, by M. Tschopp et al, licensed under https://creativecommons.org/licenses/by/4.0/).

2.2.4 Computational studies of (111) twist boundaries in FCC crystal

It has been shown experimentally that the most commonly observed grain boundary plane orientation in commercially pure Al is (111), due to its low surface energy and large interplanar spacing [25]. Therefore, studying (111) grain boundaries is important. Due to the three-fold rotation symmetry of FCC crystal about the [111] direction, and the two-fold rotation symmetry about the [110] direction, only boundaries with misorientation angles in the range of 0° and 60°

are distinct. When the misorientation angle is 0°, no grain boundary exists and when the misorientation angle is 60°, the grain boundary is the so-called coherent twin, i.e., $\Sigma 3\ 60° (111)$. The structure of low-angle (near 0°) and near-twin (near 60°) (111) twist boundaries is composed of a triangular network of three crossing sets of Shockley partial dislocations with Burgers vectors of $\frac{a}{6}[\bar{1}2\bar{1}]$, $\frac{a}{6}[\bar{2}11]$, and $\frac{a}{6}[11\bar{2}]$, where a is the lattice parameter. As misorientation angle increases from 0° to 12° or decreases from 60° to 48°, the distance between the partial dislocations decreases. The intermediate misorientation angles in the range of 12° and 48° correspond to the high-angle twist boundaries in which the network of partial dislocations is not visible [20].

Dai et al [20] calculated the energy of Al (111) twist boundaries as shown in figure 2.5. The energy of the boundary with 0° twist angle is found to be zero and the energy of the boundary with 60° twist angle (i.e., coherent twin) is non-zero but very small [20]. Although in many studies including [20] and [22] no energy cusps (except for the coherent twin) is mentioned for (111) twist boundaries, Wolf in [26] mentions that small cusps can be observed for $\Sigma 7\ 38.21° (111)$ and $\Sigma 21\ 21.79° (111)$ twist boundaries. Wolf [26] also explains that the degree of smoothness of energy curves depend on the interatomic potentials and the fact that these two cusps were observed only for some interatomic potentials. However, for sure, these energy cusps for (111) twist boundaries are not as significant as the energy cusps for tilt boundaries (as shown in figure 2.4).

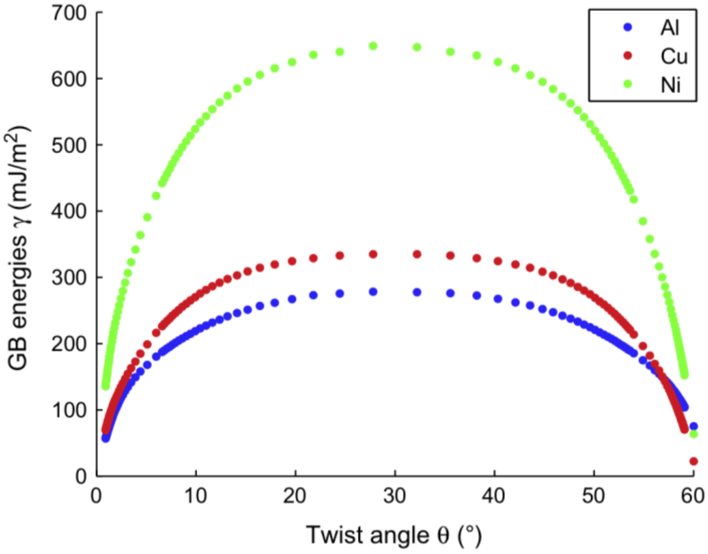


Figure 2.5: Calculated energy of (111) twist boundaries as a function of twist angle for Al, Cu, and Ni (Reproduced from [20], with the permission of Elsevier: https://doi.org/10.1016/j.actamat.2014.01.022).

2.3 Principles of Molecular Dynamics simulations

Molecular Dynamics (MD) is a simulation method for studying dynamic evolution of systems containing particles. In such systems, the particles are allowed to interact with each other during a period of time. Throughout the simulation, the new positions and velocities of the particles are determined by integrating Newton's equations of motion and using the positions and velocities of the particles in the previous step(s).

In order to perform an MD simulation, it is necessary to calculate the force that is being applied on every atom at each timestep. The calculation of forces is very time-consuming since all pairs or triples of atoms have to be considered. This computational expense can be significantly decreased by considering only short-range forces between particles within a cutoff radius [27]. The standard steps in an MD simulation, regardless of the system under study, are [2]:

(a) Initializing the positions $\{r\}$ and momenta $\{p\}$.

(b) Calculating the initial kinetic energy K, potential energy U, total energy: $E = K + U$, and other quantities of interest as well as the forces on each atom F_i.

(c) For $n_{Equilibration}$ timesteps:

  (1) Solving the equations of motion for $\{r_i(t + \Delta\mathrm{t})\}$ and momenta $\{p_i(t + \Delta\mathrm{t})\}$ from values and forces at time t, where $\Delta\mathrm{t}$ is the timestep.

  (2) Calculating the kinetic energy K, potential energy U, $E = K + U$, and other quantities of interest as well as forces F_i.

  (3) Checking for drift of values that indicate that the system is not equilibrated.

  (4) Restarting, when equilibrated.

(d) For $n_{Dynamic\ Run}$ timesteps:

  (1) Solving the equations of motion for $\{r_i(t + \Delta\mathrm{t})\}$ and momenta $\{p_i(t + \Delta\mathrm{t})\}$ from values and forces at time t.

  (2) Calculating the kinetic energy K, potential energy U, $E = K + U$, and other quantities of interest as well as forces F_i.

(3) Accumulating values of K, U, etc. for averaging.

(e) Analyzing data: averages, correlations, etc.

The LAMMPS (Large-scale Atomic-Molecular Massively Parallel Simulator) [28] package is one of the most popular classical MD simulator. LAMMPS is an open-source code which is distributed by Sandia National Laboratories. The result of LAMMPS simulation is the trajectories of atoms throughout the simulation which then can be visualized and post-processed using other software packages such as OVITO (the Open Visualization Tool) [29].

2.3.1 Interatomic potential

Potential energy, which is the sum of the energetic interactions between the atoms, is the most fundamental quantity that governs the thermodynamics of materials. Cohesive energy (i.e., potential energy at 0 K) of a system of N atoms is equal to:

$$U = E(all\ atoms) - \sum_{i=1}^{N} E_i \ , \tag{2.3}$$

where $E(all\ atoms)$ is the total energy of the system, and E_i is the energy of isolated atom i. In a more general form, the potential energy U of system can be written as:

$$U = \sum_{i=1}^{N} v_1(r_i) + \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}{}' \phi_{ij}(r_i, r_j) + \frac{1}{6}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N}{}' v_3(r_i, r_j, r_k) \ , \tag{2.4}$$

where the $'$ sign indicates that the $i = j$ terms and the $i = j,\ j = k,\ k = i,\ i = j = k$ terms are not included in the second and third sets of sums, respectively. In equation 2.4, the second and third terms correspond to two-body and three-body interactions between atoms, respectively [2].

Interatomic potentials are the potentials between atoms that reflect the bonding between them. The interaction energy between atoms i and j, located at r_i and r_j positions is equal to:

$$\phi_{ij}(r_i, r_j) = E(i + j) - E(i) - E(j) \ , \tag{2.5}$$

where $E(i + j)$ is the energy of the pair of atoms, and $E(i)$ and $E(j)$ are the energies of individual atoms with infinite separation. If the distance between atoms i and j is very small, then a repulsive force should exist between them. For greater separations, however, there must be an attractive force between them [2]. An example of pair potentials is the Lennard-Jones 12-6 potential:

$$\phi_{LJ}(r) = \epsilon\left[\left(\frac{r_0}{r}\right)^{12} - 2\left(\frac{r_0}{r}\right)^{6}\right] = 4\epsilon\left[\left(\frac{d}{r}\right)^{12} - \left(\frac{d}{r}\right)^{6}\right], \tag{2.6}$$

where $\epsilon = \phi_{LJ}\left(r = r_0 = 2^{\frac{1}{6}}\, d\right)$ is the depth of potential well and d is the distance at which the potential is equal to zero, i.e., $\phi_{LJ}(d) = 0$. A graph of the Lennard-Jones 12-6 potential is shown in figure 2.6. The dependence r^{-6} in equation 2.6 is obtained theoretically, and the dependence r^{-12} is chosen mainly because of the convenience in numerical calculation. The Lennard-Jones 12-6 potential is effectively a short-range potential, since $\phi_{LJ}(r)$ is practically equal to zero for $r > 2.5\, d$ [27].

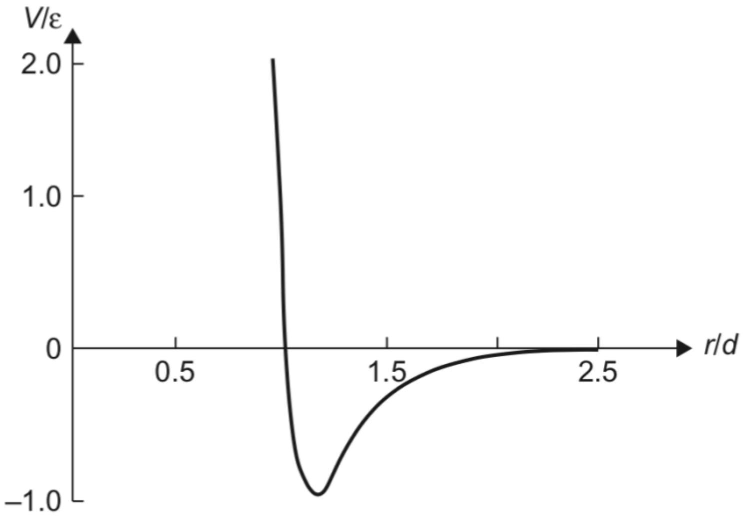


Figure 2.6: The plot of the Lennard-Jones 12-6 potential (Reproduced from [27], with the permission of Elsevier: https://doi.org/10.1016/B978-0-12-420143-9.00001-6).

In metals, the interaction between the electrons must be considered in the description of bonding. As a result, simple pair potentials such as the Lennard-Jones potential cannot adequately describe the binding and other properties of metals [2]. Nowadays, the embedded-atom method (EAM) potentials that were initially proposed by Daw and Baskes [30] are the most common interatomic potentials for metals. EAM potentials are motivated by density functional theory and have the form of:

$$U_{EAM} = \sum_{i} G_i \left[\sum_{j \neq i} f_{ij}(r_{ij}) \right] + \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N}{}' \phi_{ij}(r_{ij}) \ , \tag{2.7}$$

where ϕ is a pair potential, and f is a function of the interatomic distance which represents an approximation of the electron density. In equation 2.7, G_i is the energy needed to embed atom i in a uniform electron gas of density $\sum_{j \neq i} f_{ij}(r_{ij})$ (i.e., considering the effect of neighboring atoms). G is a non-linear function which cannot be written as a sum of pair potentials, thus represents a true many-body interaction. Different types of EAM potentials have different choices for the functions G, f and ϕ. These functions depend on parameters that can be determined by fitting the predictions to the available and reliable experimental or analytical material data, such as cohesive energy, vacancy formation energy, etc. [2,27].

2.3.2 Time integration of Newton's equations of motion

Newton's second law states that the force applied on a particle is equal to its mass times its acceleration:

$$F_i = m_i a_i = m_i \frac{d^2 r_i}{dt^2} \ , \tag{2.8}$$

where, m_i and a_i are the mass and acceleration of particle i, respectively. Also, classical mechanics explains that the force acting on a particle is equal to the negative of the gradient of the potential energy with respect to the particle's position:

$$F_i = -\nabla_i U(r^N) = -\left(\frac{\partial U(r^N)}{\partial x_i} \hat{x} + \frac{\partial U(r^N)}{\partial y_i} \hat{y} + \frac{\partial U(r^N)}{\partial z_i} \hat{z} \right) , \tag{2.9}$$

where $U(r^N) = U(r_1, r_2, \ldots, r_N)$ is the interatomic potential which is a function of position of atoms, i.e., $r_i = (x_i, y_i, z_i)$ [2]. If initial velocity and particle positions are set, then by integration of Newton's equations of motion, their values can be determined at other points in time [27].

The Verlet algorithm [31] is one of the well-known methods for numerically solving the Newton's equations of motion. This algorithm starts with considering the Taylor expansions of x_{t+1} and x_{t-1}:

$$x_{t+1} = x_t + v_t \Delta\mathrm{t} + \frac{1}{2} a_t (\Delta t)^2 + O((\Delta t)^3)\,, \tag{2.10}$$

$$x_{t-1} = x_t - v_t \Delta\mathrm{t} + \frac{1}{2} a_t (\Delta t)^2 + O((\Delta t)^3)\,, \tag{2.11}$$

where $\Delta\mathrm{t}$ is the timestep and x_t, v_t, and a_t are the position, velocity, and acceleration of a particle at the beginning of the time interval, respectively. Combining equations 2.10 and 2.11 results in:

$$x_{t+1} = 2x_t - x_{t-1} + a_t (\Delta t)^2 + O((\Delta t)^3)\,. \tag{2.12}$$

Also, velocity is determined as follows [27]:

$$v_t = \frac{x_{t+1} - x_{t-1}}{2\Delta t}\,. \tag{2.13}$$

The error in the Verlet algorithm is of the third order of Δt in position and of the second order of Δt in velocity. However, the velocity is not used in integration of equations of motion. The Verlet algorithm is simple and accurate enough. Yet, it has one disadvantage: it is not self-starting. In other words, it is necessary to use another algorithm to determine the first couple of points. The velocity form of the Verlet algorithm (the following scheme), however, does not have that limitation [27]:

$$x_{t+1} = x_t + v_t \Delta\mathrm{t} + \frac{1}{2} a_t (\Delta t)^2\,, \tag{2.14}$$

$$v_{t+1} = v_t + \frac{1}{2} (a_{t+1} + a_t) \Delta t\,. \tag{2.15}$$

In any MD simulation, the optimal value for timestep, i.e., Δt, is always a balance between accuracy and computation time. Large timesteps result in larger errors in the calculation since the assumption that the acceleration is constant during a time interval becomes less reasonable. On the other hand, using a smaller timestep lead to higher computational expense since each timestep requires a force calculation [2].

For an MD simulation which is under adiabatic and constant volume constraints, the total energy, i.e., $E = K + U$, should not change with time. As a result, the important criterion for the choice of timestep is to check how well the simulation result follows this conservation law. It is true that since the equations of motion are not integrated exactly, E will never be constant, but for the right choice of timestep, E should fluctuate around an average value and there should not be any significant change in the average value over time. The choice of the timestep depends on the shortest vibration period in the system. In solids, the shortest vibration period is of the order of a picosecond (i.e., 10^{-12} seconds) or less. The number of timesteps needed to accurately integrate the equation of motion over a vibrational period depends on the method that is being used for integration. For the Verlet algorithm, of the order of 50 timesteps per vibrational period are usually needed to result in enough accuracy for energy conservation. In other words, the timestep should typically be of the order of $10^{-15} - 10^{-14}$ seconds [2]. It is also important to note that drastic changes in E do not necessarily imply that the employed timestep was not correct, but it can also be an indication of the poor choice of algorithm for integration of the Newton's equations of motion [2].

2.3.3 Statistical thermodynamics and ensemble averages

Statistical thermodynamics, also referred to as statistical mechanics, is a field of study that explains how the macroscopic properties of a system are related to its microscopic variables. The macrostate of a system refers to the bulk behavior of system that can be described by thermodynamics whereas the microstate of a system is the instantaneous value of its internal variables such as atomic coordinates and momenta. The microstates changes over time as the system evolves. Statistical thermodynamics explains how to relate averages over the microstates to the thermodynamics of the macrostates. There are two ways to connect macrostates to microstates, by time averages or by ensemble averages [2].

The instantaneous microstate of a system that contains N atoms at any specific time t can be described in a $6N$ dimensional space, i.e., phase space. These $6N$ quantities are the positions ($3N$) and velocities ($3N$) of atoms. Therefore, the average of a quantity H would be a time average of that quantity over some long time period t_0:

$$\langle H \rangle = \frac{1}{t_0}\int_0^{t_0} H(t)dt \ , \tag{2.16}$$

where H is a variable of time, and $H(t)$ is the value of H determined at the phase space, at time t. But there is also another way to calculate average quantities, i.e., ensemble averages. Instead of observing the system over a long period of time, it is possible to create a very large number of identical systems and allow them to evolve independently, i.e., an ensemble of systems. Although these systems are identical, they are in different states. The probability density of occupying a specific state can be calculated by dividing the number of times that the given state is observed in the ensemble of systems by the total number of systems. Thus, the average of quantity H is equal to:

$$\langle H \rangle = \sum_{\alpha=1}^{N_{states}} \rho_\alpha H_\alpha \, , \tag{2.17}$$

where H_α is the value of quantity H at each distinct state α, and ρ_α is the probability density of state α, where $\sum_{\alpha=1}^{N_{states}} \rho_\alpha = 1$. The probability density for an ensemble of systems depends on the constraints that are put upon the system. These constraints are normally the thermodynamic quantities that are fixed [2]. Some of the well-known ensembles are as follows:

- Microcanonical ensemble (NVE): constant number of atoms, volume and total energy,
- Canonical ensemble (NVT): constant number of atoms, volume and temperature,
- Isobaric-isothermal ensemble (NPT): constant number of atoms, pressure and temperature.

2.3.4 NPT ensemble

Most experimental systems in the laboratory are under constant temperature and pressure. Neither the microcanonical (NVE) nor the canonical (NVT) ensemble can perfectly describe such a system. That system can be described by the isobaric-isothermal (NPT) ensemble, in which the total energy and the volume are not fixed, and at equilibrium, they both fluctuate around an average value. The NPT ensemble is particularly useful when studying phase transformations during which the volume changes at a constant pressure [2].

Nose-Hoover thermostat [32,33] is an algorithm for fixing the temperature in MD simulations. The main idea is to introduce a new variable s, which couples the system to a heat bath. In this method, the velocity of atoms is related to the time derivative of position, i.e., $v = s\frac{dr}{dt}$. Therefore, changing s will result in a change in kinetic energy and temperature. Also, a fictitious potential energy U_s and a kinetic energy k_s associated with the bath are introduced in this method:

$$U_s = \mathcal{L}\, k_B\, T_s \ln(s)\,, \tag{2.18}$$

$$K_s = \frac{P_s^2}{2Q}\,, \tag{2.19}$$

where $\mathcal{L} = 3N + 1$, k_B is the Boltzmann constant, T_s is the temperature of the heat bath, $P_s = Q\, ds/dt$ is the momentum associated with the variable s, and Q is a coupling parameter. Parameter Q is an input into the simulation that defines the degree of coupling to the heat bath. Too small values for Q result in poor equilibration, while too large values for Q lead to slow energy flow [2]. Similarly, the main idea in Nose-Hoover barostat [33,34] is to introduce an additional variable, which in this case is the volume of simulation cell.

2.3.5 Periodic boundary conditions

The purpose of most material models and simulations is to describe the characteristics of a macroscopic system which is infinitely large compared to the actual model size that is being simulated with the use of computer. Thus, a way for mimicking the characteristics of the real system is needed when simulating a system of manageable size. The most common approach is to use periodic boundary conditions, in which a macroscopic system is described as an infinite array of equivalent finite systems. The system of interest is placed in a finite-sized volume, i.e., simulation cell, which is then replicated throughout the space to approximate the effect of the rest of the material [2].

Figure 2.7(A) shows the concept of periodic boundary conditions where atom A_0 interacts not only with atom A which is located within the simulation cell but also with its image, i.e., atom A'. In this figure, atom A' is shifted relative to A by a distance that is equal to the length of the simulation cell. Figure 2.7(B) shows the periodic boundary conditions which are applied in two directions [27].

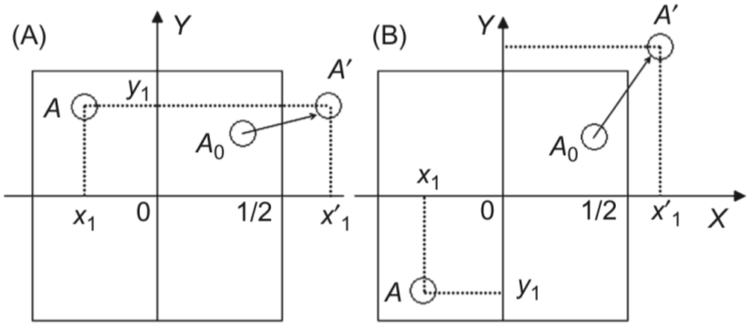


Figure 2.7: Interaction between atoms in presence of periodic boundary conditions (Reproduced from [27], with the permission of Elsevier: https://doi.org/10.1016/B978-0-12-420143-9.00001-6).

Since atoms can move during MD simulation, using periodic boundary conditions also affects the way that the new position of atoms should be calculated. For instance, if atom A exits the simulation cell from one side, then its image, i.e., atom A', will enter the simulation cell from the opposite side. In other words, the position of atom A' is calculated using the equation below:

$$x_{A'} = \begin{cases} x_A - L, & x_A > L \\ x_A + L, & x_A < 0 \end{cases}, \tag{2.20}$$

where L is the length of simulation cell [27]. It is worth mentioning that it is not necessary to use periodic boundary conditions in all directions of the simulation cell. Using non-periodic boundaries can be useful in case of simulating a system which contains free surfaces or fixed surfaces along a specific direction.

2.4 Analysis of Molecular Dynamics simulation results

Crystallographic defects play an important role in materials behavior. Yet, classical MD simulation models do not keep track of the crystallographic defects explicitly. Therefore, in order to identify crystallographic defects, separate algorithms are needed to analyze the positions of atoms predicted by MD simulations. Most of these algorithms attempt to match the local atomic structure to a perfect one (such as FCC or BCC), and measure how closely they fit [35]. Two of the most commonly used structure characterization algorithms for MD simulations are common neighbor analysis (CNA) [36] and dislocation extraction algorithm (DXA) [37]. Both of these algorithms are implemented in the OVITO (the Open Visualization Tool) visualization package [29].

2.4.1 Common neighbor analysis

Common neighbor analysis (CNA) [36] can be used to identify the local crystal structure of an atom. Normally, two atoms are assumed to be neighbors if the distance between them is less than a specified cutoff distance, i.e., r_{cut}. For FCC structure this distance is set to be halfway between the first and second neighbor shells. In other words:

$$r_{cut}^{FCC} = \frac{1}{2}\left(\sqrt{\frac{1}{2}} + 1\right) a_{FCC}\,, \tag{2.21}$$

where a_{FCC} is the lattice constant of the crystal structure [35].

To assign a local crystal structure to a specific atom, a triplet of indices $(n_{cn}\ n_b\ n_{lcb})$ is calculated for each of the N neighbor bonds of the given atom, where n_{cn} is the number of common neighbors of the central atom and its neighbor, n_b is the number of bonds between these common neighbors, and n_{lcb} is the number of bonds in the longest continuous chain of bonds between the common neighbors. This yields N triplets which then can be compared with table 2.1 to identify the local crystal structure of the given atom [35].

Table 2.1: CNA signatures (triplets) of common crystal structures (Reproduced from [35], with the permission of IOP Publishing: https://doi.org/10.1088/0965-0393/20/4/045021)

fcc ($N = 12$)	hcp ($N = 12$)	bcc ($N = 14$)	cubic diamond ($N = 16$)
$12 \times (421)$	$6 \times (421)$ $6 \times (422)$	$8 \times (666)$ $6 \times (444)$	$12 \times (543)$ $4 \times (663)$

2.4.2 Dislocation extraction algorithm

Dislocation extraction algorithm (DXA) [37] can be used to identify different types of dislocations and their Burgers vector. In this algorithm, first, the strain-free atomic arrangement is determined locally using the CNA algorithm. Then, a two-dimensional manifold which is called interface mesh is constructed. The interface mesh separates the strain-free region from the distorted region. Next, two Burgers circuits are constructed and displaced in the opposite directions on the interface mesh sweeping along the dislocation line (see figure 2.8). During the sweeping phase, the shape of the dislocation line is recorded by computing the position of the center of mass for the two Burgers circuits [37].

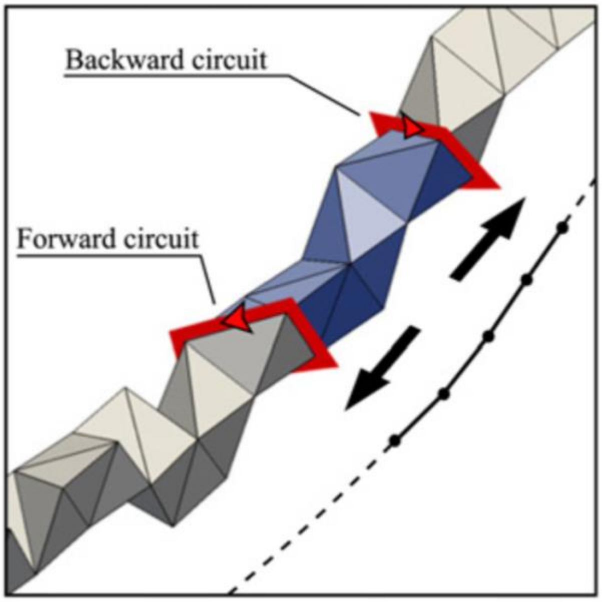


Figure 2.8: Generating the shape of the dislocation line during the sweeping process in DXA (Reproduced from [37], with the permission of IOP Publishing: https://doi.org/10.1088/0965-0393/20/8/085007).

2.5 Objective

The objective of this computational research work is to study the contact between two Al flat substrates at the nanoscale with the use of MD and investigate:

- the JC behavior between (111)-oriented surfaces,
- the effect of interatomic potential, temperature, and misorientation angle between substrates on the critical distance for JC, and
- the effect of misorientation angle and initial separation between substrates on the distribution and density of crystallographic defects after contact, at equilibrium.

3 Simulation process

MD simulations are performed using the LAMMPS [28] simulator and embedded-atom method (EAM) potential for aluminum [38]. The cutoff distance and predicted melting temperature for this interatomic potential are 6.5 Å and 880 K, respectively [38]. Time integration is done using the velocity form of the Verlet algorithm [31]. Temperature and pressure (only in x and y directions) are kept constant (i.e., NPT ensemble) using the Nose-Hoover thermostat and barostat [32–34]. The simulation timestep is chosen to be 0.002 ps. Periodic boundary conditions are used along the x and y directions of simulation cell.

The whole simulation process is comprised of the following four steps:

Step 1- *<u>Creating a simulation cell containing two aluminum substrates:</u>*

Initially, a simulation cell is created which its x, y, and z directions are along the $[1\bar{1}0]$, $[11\bar{2}]$, and $[111]$ crystallographic directions of a perfect FCC crystal, respectively. Two in-registry aluminum substrates are created and placed away from each other. Each substrate contains $12, (111)$ atomic layers along the z direction of simulation cell. Figure 3.1 shows the simulation cell at the end of this step.

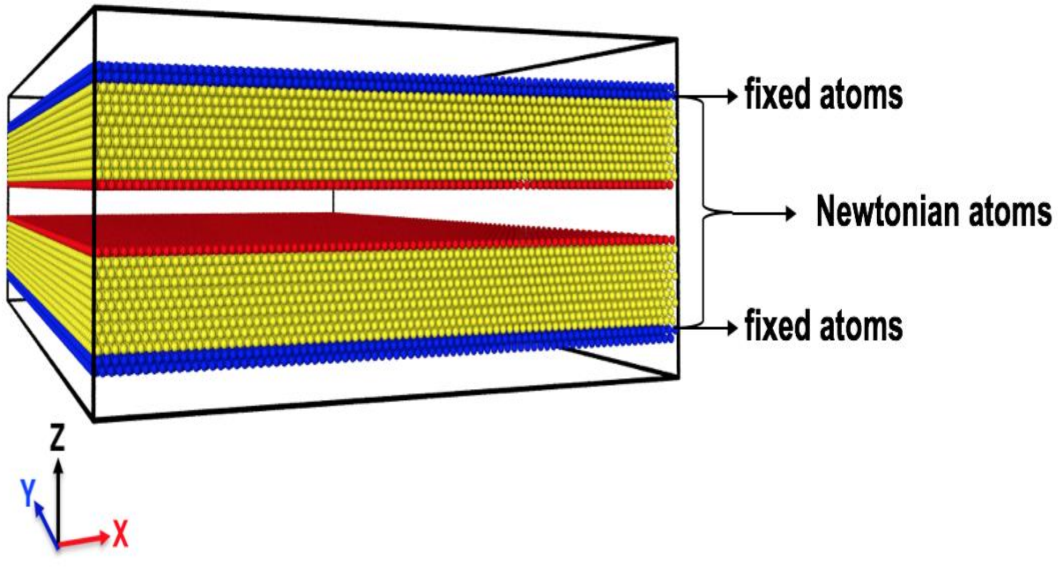


Figure 3.1: Atomic configuration in the simulation cell, at the end of step 1. Red atoms are surface atoms. Blue atoms will be kept fixed during simulation while the yellow and red atoms will be allowed to move (i.e., Newtonian atoms). All atoms are aluminum atoms. Image is visualized using OVITO [29].

Step 2- *<u>Preparing atomic configurations with varied misorientation angles:</u>*

In order to produce atomic configurations with different misorientation angles, the two substrates are rotated by $\theta/2$ and $-\theta/2$ about the z direction of simulation cell. The rotation is performed based on the approach described in [20]. Because of the three-fold rotational symmetry of FCC crystals about $[111]$ and the two-fold rotational symmetry about $[110]$, only configurations with $0 \leq \theta \leq \pi/3$ are distinct. In order to have periodic boundary conditions in the x and y directions, the requirement below should be fulfilled [20]:

$$\begin{cases} n = \dfrac{\sqrt{3}}{\tan\frac{\theta}{2}} & for \quad 0 < \theta < \dfrac{\pi}{6} \quad or \quad \theta = \dfrac{\pi}{3} \\ n = \dfrac{\sqrt{3}}{\tan\left(\frac{\pi}{6} - \frac{\theta}{2}\right)} & for \quad \dfrac{\pi}{6} < \theta < \dfrac{\pi}{3} \quad or \quad \theta = 0 \end{cases}, \tag{3.1}$$

where n, the periodicity in both x and y directions, is an integer. Thus, the minimum simulation cell lengths in these two directions should be [20]:

$$\begin{cases} L_x = \dfrac{n\ a}{\sqrt{2}\ cos(\frac{\theta}{2})} = \dfrac{L_y}{\sqrt{3}} & for \quad 0 < \theta < \dfrac{\pi}{6} \quad or \quad \theta = \dfrac{\pi}{3} \\ L_y = \dfrac{n\ a}{\sqrt{2}\ cos(\frac{\pi}{6} - \frac{\theta}{2})} = \dfrac{L_x}{\sqrt{3}} & for \quad \dfrac{\pi}{6} < \theta < \dfrac{\pi}{3} \quad or \quad \theta = 0 \end{cases}, \tag{3.2}$$

where a is the lattice constant. At the end of this step, 34 atomic configurations are generated corresponding to 34 misorientation angles in the range of $0 \leq \theta \leq \pi/3$. Steps 1 and 2 are done by running an automated Python script which is included in Appendix A.

Step 3- *<u>Replication, relaxation, and thermalization:</u>*

All atomic configurations corresponding to different misorientation angles are replicated to have about 300000 atoms in the simulation cell. Minimization of energy is performed at 0 K for each atomic configuration using the conjugate gradient method which is implemented in LAMMPS. Energy minimization is done by iteratively adjusting the position of atoms until a pre-defined stopping criterion (i.e., a tolerance value for energy or force) is satisfied. The conjugate gradient algorithm determines the search direction along which the atom coordinates are being changed as well as how the minimization should be restarted once it fails to make further progress. Once the energy minimization is finished, each system is thermalized at 300 K for 20 ps. A sample LAMMPS code corresponding to this step is included in Appendix B.

Step 4- <u>Performing contact simulation for varied interfacial distances:</u>

Finally, for each thermalized system corresponding to different misorientation angles, the two substrates are placed at varied interfacial distances and contact simulation is performed for each distance at 300 K for 20 ps. This step is done for several interfacial distances in the range of 2.5 Å and 6.0 Å because of two reasons: firstly, to determine the critical interfacial distance below which JC occurs, and secondly, to investigate the effect of initial interfacial distance on the distribution and density of crystallographic defects after contact. A sample LAMMPS code corresponding to this step is included in Appendix C.

4 Results and Discussion

Sections 4.1 and 4.2 include the discussion of the JC behavior and analysis of crystallographic defects at the interface at equilibrium, respectively. The results are obtained from MD simulations performed for misorientation angles θ in the range of $0°$ and $60°$ and initial interfacial distances in the range of 2.5 Å and 6.0 Å.

4.1 Jump-to-contact behavior

In order to determine the JC occurrence, similar to the approach used in [39], a deviation parameter is defined as follows:

$$D(t,\theta) = \frac{|h(t,\theta) - d|}{d} \,, \tag{4.1}$$

where $h(t,\theta)$ is the instantaneous distance between contacting surfaces as a function of time and misorientation angle, and d is the interplanar spacing of (111) planes at 300 K, in a perfect FCC crystal (~2.35 Å). When JC occurs, the deviation parameter will have a positive value very close to zero. The value of the deviation parameter is calculated throughout the simulations by outputting the average of z positions of all atoms in the contacting layers. Moreover, the average of $D(t, \theta)$ in the last 2 ps of simulations is named D_{ave}.

4.1.1 Occurrence of jump-to-contact

The plot of the deviation parameter, D, as a function of simulation time is shown in figure 4.1. This figure shows the results of simulations with the same misorientation angle ($\theta = 0°$), but different initial interfacial distances. For initial interfacial distances less than or equal to 5.8 Å, JC occurs since the value of D drops to a positive value close to zero over a relatively short period of time and oscillates around that value until the system is at equilibrium. For the case with initial interfacial distance of 6 Å, however, JC does not occur.

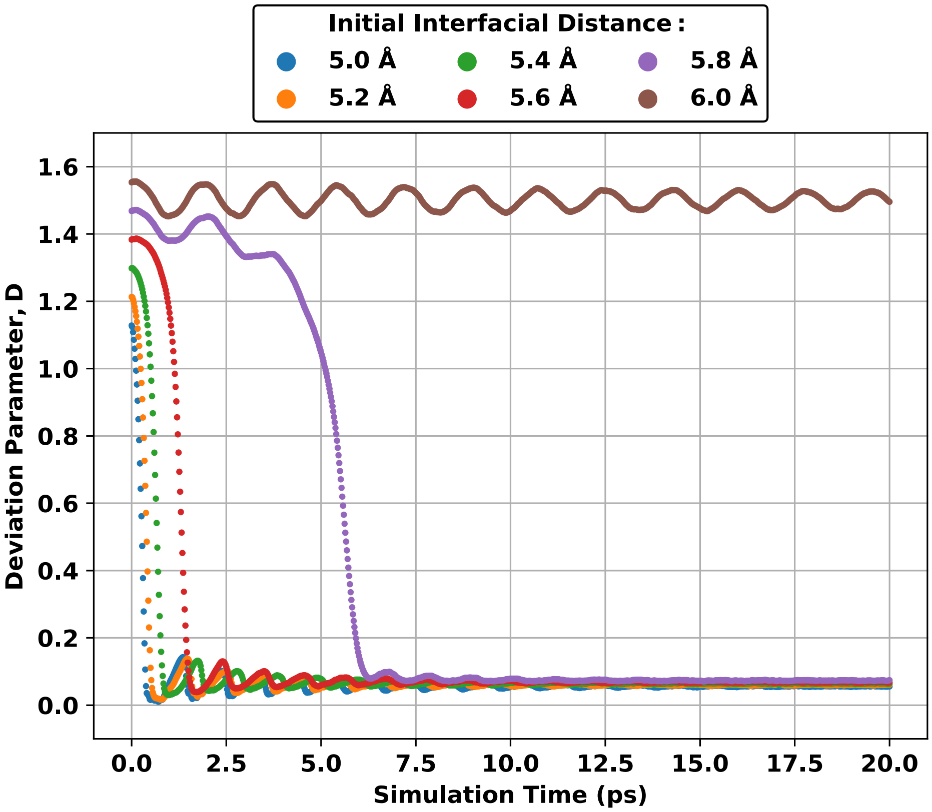


Figure 4.1: Plot of deviation parameter, D, versus simulation time for simulations with θ = 0°, but different initial interfacial distances.

Figure 4.2 shows the 2D maps of the local interfacial distance during the simulation of the case with misorientation angle of $\theta = 0°$ and initial interfacial distance of 5.5 Å. This figure shows that JC occurs locally at random locations (see figure 4.2(b)), and then after a short period of time, contact is formed across the whole interface. Those random locations are the places where the atoms on one surface come into a close vicinity of the atoms on the other contacting surface. It also worth mentioning that since there is no misorientation between surfaces in this case, the final 2D map of the local interfacial distance is very uniform.

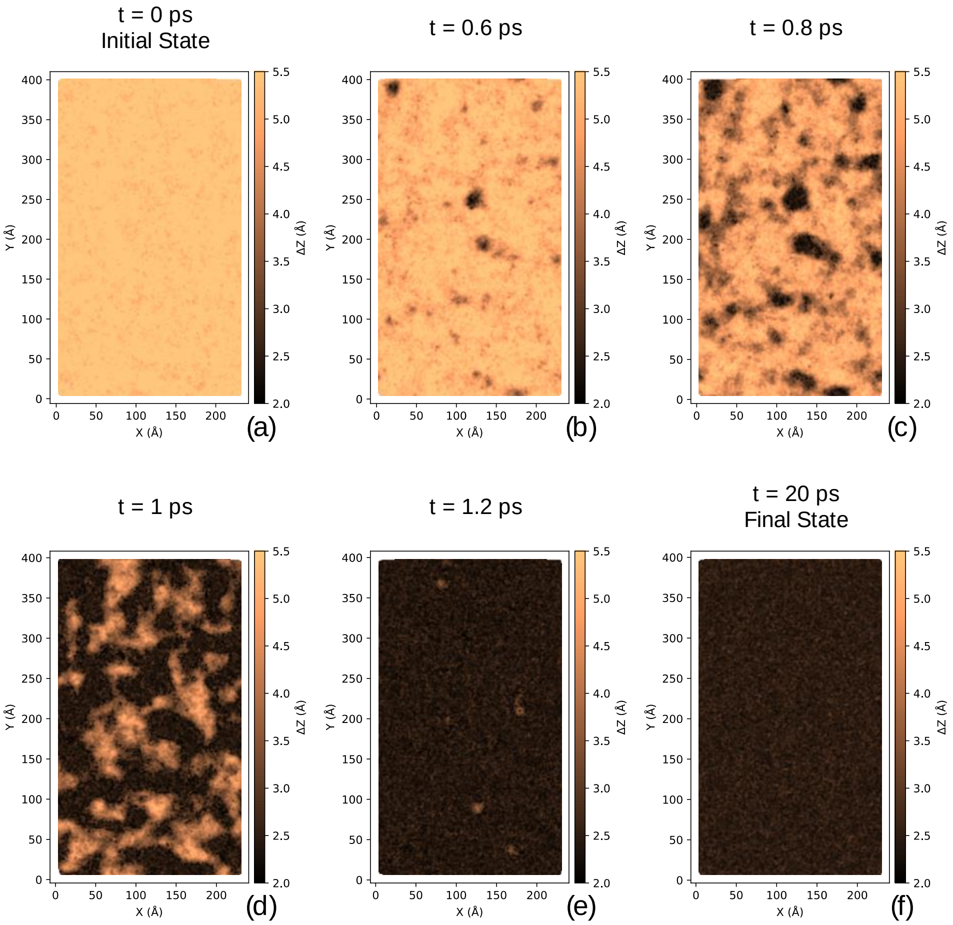


Figure 4.2: 2D maps of the local interfacial distance for the simulation with θ = 0° and initial interfacial distance of 5.5 Å.

4.1.2 Critical distance for jump-to-contact

To determine the critical distance for JC, the plot of D_{ave} (i.e., average of D in the last 2ps) as a function of initial interfacial distance for simulations with misorientation angle of $\theta = 0°$ is presented in figure 4.3. This figure shows that the critical distance below which JC occurs is 5.8 Å, which is approximately $2.5 \times d$, where d is the interplanar spacing of (111) planes in a perfect FCC Al crystal at 300 K. This distance corresponds to approximately 7% elastic strain in the system.

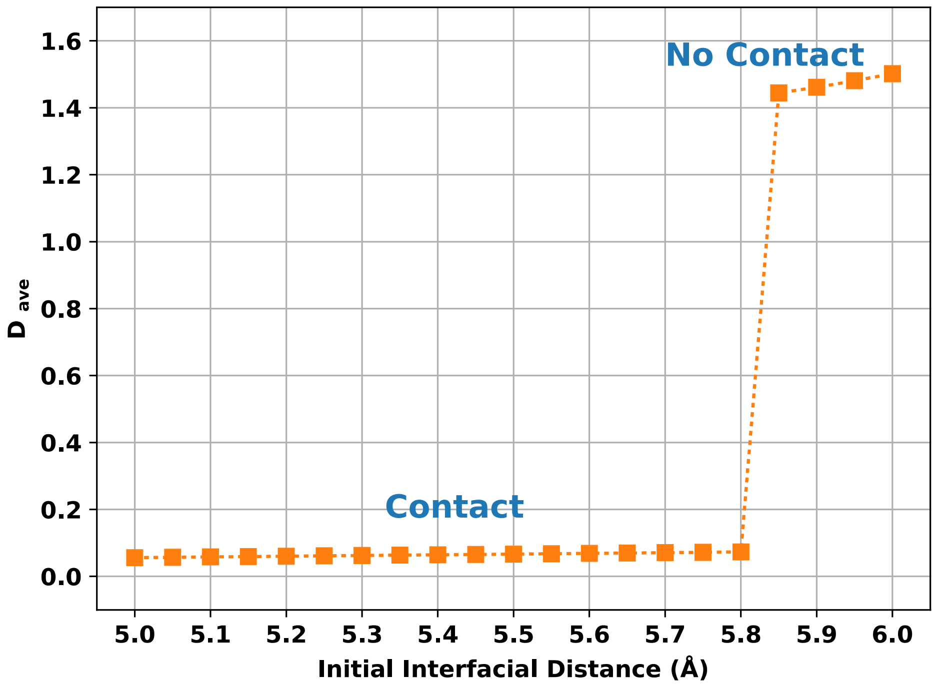


Figure 4.3: Plot of D_{ave} versus initial interfacial distance for simulations with $\theta = 0°$. The critical distance for JC is 5.8 Å.

4.1.2.1 Effect of interatomic potential

In order to find out whether the interatomic potential has an effect on the critical distance for JC or not, three additional sets of simulations are performed employing different interatomic potentials. All simulations are performed for the case with misorientation angle of $\theta = 0°$, but with various initial interfacial distances between 4.5 Å and 6.5 Å. Table 4.1 shows the details of the interatomic potentials that are used, as well as the resulting critical distance for JC computed from simulations using those interatomic potentials. *Al.lammps.eam* [38] is the main interatomic potential used in this thesis. *Al99.eam.alloy* [40] is another well-tested EAM interatomic potential for Al. The values of σ and ε for the two Lennard-Jones (LJ) potentials are based on [41], where σ is the distance at which the potential is zero, and ε is the depth of potential well. The cutoff distance for the two LJ potentials is chosen to be 6.5 Å to enable comparison with the *Al.lammps.eam* potential and determine the effect of type of potential. Also, in both cases, 6.5 Å is very close to the value of 2.5σ that is generally considered as an acceptable cutoff for LJ potentials.

Comparing the two EAM potentials, it can be seen that the potential with a larger cutoff distance results in a greater critical distance for JC. This is due to considering more long-range attractive forces between the atoms located on the two substrates. Also, comparing the *Al.lammps.eam* EAM potential with any of the two LJ potentials (with the same cutoff distance), it is clear that the critical distance for JC is lower when an LJ potential is used. This is due to the fact that LJ potentials are two-body potentials and they neglect the effect of surrounding atoms on the force between two specific atoms. This is the main reason that they are not ideal for studying phenomena in metals. Lastly, comparing the two LJ potentials with each other, it can be seen that the critical distance for JC is smaller when the LJ 12-6 potential is used. This is due to LJ 12-6 potential having a steeper repulsion term compared to the LJ 9-6 potential.

Table 4.1: Details of four employed interatomic potentials as well as the resulted critical distance for JC from simulations of the case with misorientation angle of θ = 0°

Interatomic Potential	**Cutoff (Å)**	**σ (Å)**	**ε (eV)**	**Critical distance for JC (Å)**
Al.lammps.eam	6.5	-	-	5.8
Al99.eam.alloy	6.287	-	-	5.6
Lennard-Jones 9-6	6.5	2.601	0.141	5.5
Lennard-Jones 12-6	6.5	2.607	0.174	5.2

4.1.2.2 Effect of contact temperature

To investigate the effect of contact temperature on the value of critical distance for JC, the contact simulations are performed for two additional temperatures, i.e., $450\,\mathrm{K}$ and $600\,\mathrm{K}$. The misorientation angle is $0°$ in all simulations. Table 4.2 shows that the critical distance for JC is larger at higher temperatures, meaning that JC can occur at greater distances between surfaces. This observation is mainly due to higher amplitude of atomic vibrations at higher temperatures resulting in a higher possibility of surface atoms to come into a close vicinity of the atoms located on the opposing surface.

Table 4.2: The critical distance for JC as a function of contact temperature for the simulations of the cases with misorientation angle of θ = 0°

Temperature (K)	**300**	**450**	**600**
Critical distance for JC (Å)	5.8	6.0	6.4

4.1.2.3 Effect of crystallographic misorientation

Simulations are performed for all 34 misorientation angles to see if misorientation angle has an effect on the critical distance for JC. The effect of misorientation angle is found to be insignificant since the critical distance was 5.8 ± 0.05 Å for all angles studied. The plot of D_{ave} versus initial interfacial distance is shown in figure 4.4, only for five misorientation angles. The results of simulations for some other misorientation angles between 0° and 30° are included in Appendix D. Although the critical distance is found to be independent of misorientation angle, the value of D_{ave} is found to be a function of it, for the cases that JC occurs. This suggests that misorientation controls the distribution of resultant crystallographic defects at the interface since the value of D (and so D_{ave}) represents how close the interfacial distance is to the interplanar spacing of (111) planes in a perfect crystal (see equation 4.1).

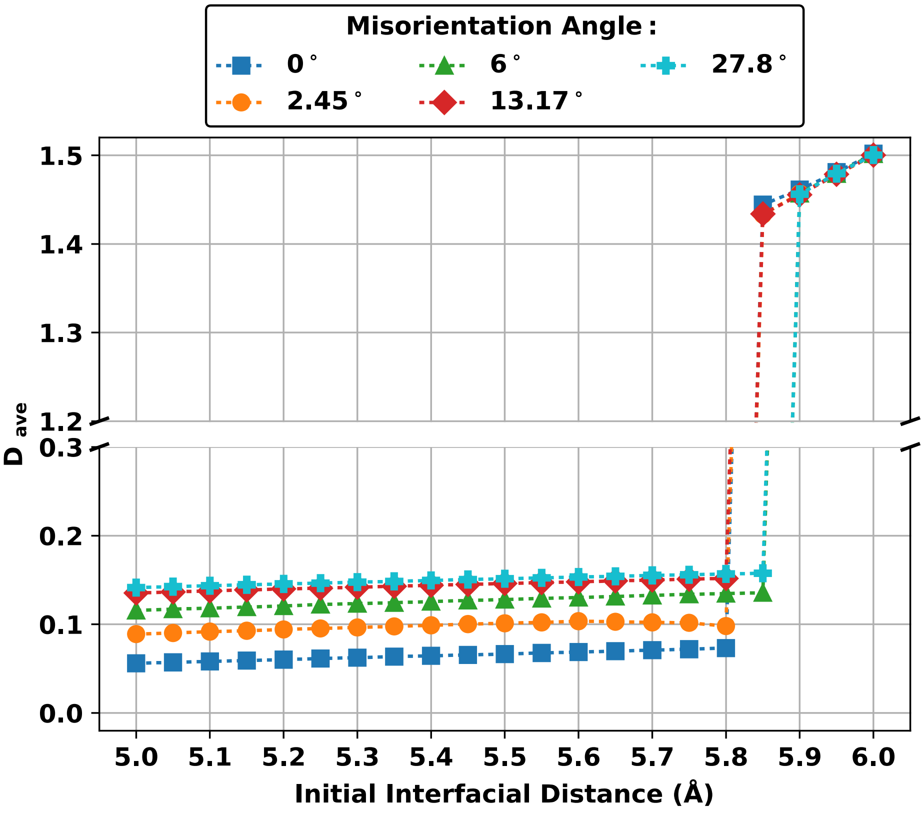


Figure 4.4: Plot of D_{ave} versus initial interfacial distance for five simulations with different misorientation angles, θ.

To further investigate on this matter, the values of D_{ave} for all 34 misorientation angles are shown in figure 4.5, for three cases with different initial interfacial distances. The curves in figure 4.5 follow a similar trend to the trend of the energy of Al (111) twist boundaries shown in figure 2.5. This is not surprising since the effect of misorientation in our system has been expected to be similar to the twist boundaries. In other words, existence of interfacial gap does not disrupt the general effect of misorientation. Additionally, the two special CSL angles of 21.79° (Σ21) and 38.21° (Σ7), reported by Wolf in [26], are noticeable on the curves. Using these two angles, the range of misorientation angle is divided into three ranges. It is seen that the curvature of the curves in ranges (I) and (III) is a function of the initial interfacial distance but is independent of it in range (II). This suggests that the existence of interfacial gap has to be compensated by defect formation, and different types of crystallographic defects should be expected in the middle range compared to the other two ranges.

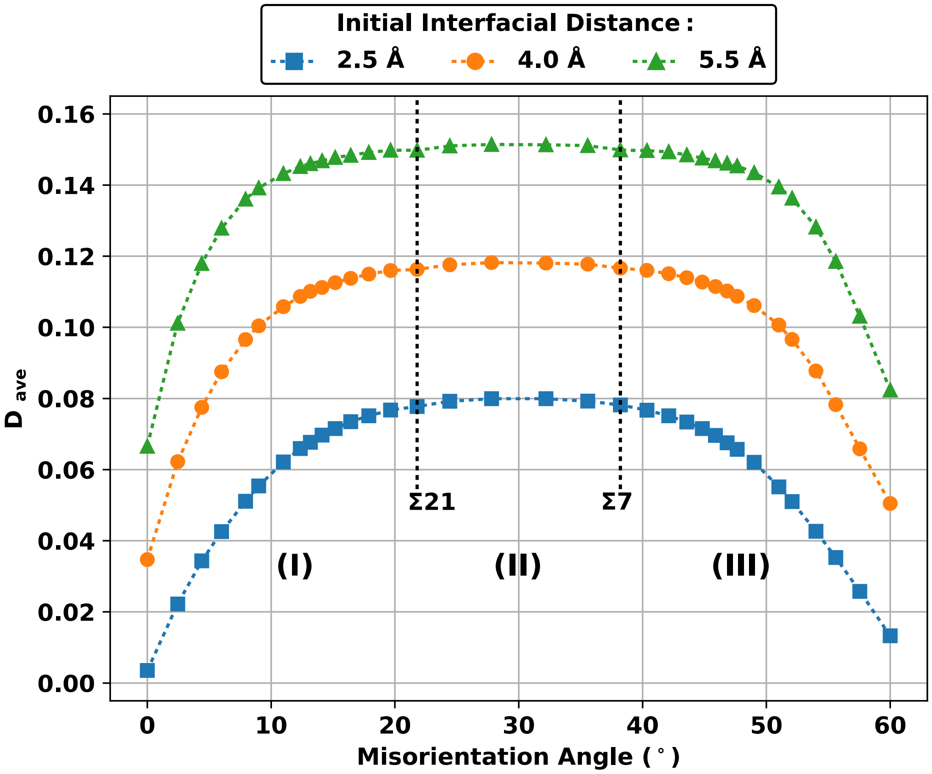


Figure 4.5: The plot of D_{ave} versus misorientation angle for three different initial interfacial distances. These initial interfacial distances are chosen because 2.5 Å is very close to the interplanar spacing of (111) planes in a perfect Al FCC crystal at 300 K and 5.5 Å is very close to the critical distance for JC.

4.2 Analysis of crystallographic defects at the interface

In the second part of this research work, the effect of crystallographic misorientation and initial interfacial distance on the final distribution and density of crystallographic defects at the interface after contact is investigated. Section 4.2.1 discusses the effect of the two parameters on the distribution of crystallographic defects, while section 4.2.2 presents a quantitative analysis of defects density.

4.2.1 Distribution of crystallographic defects

The resultant crystallographic defects at the interface (i.e., the two surface layers) after contact are detected using the dislocation analysis modifier [37] of OVITO [29]. Figure 4.6 illustrates the structure of interface for three different simulations, all within range (I) of misorientation angle.

The images in the first column show the top-view snapshots of the interface, where the dislocations and atoms are colored based on their type (not Burgers vector) and local crystal structure, respectively. Shockley partials and perfect dislocations are shown in yellow and blue colors, respectively. Also, atoms with FCC, HCP, and unidentified local crystal structure are colored as green, red, and white, respectively. As it can be seen, the disordered atoms (i.e., atoms with unidentified local crystal structure) are located near the Shockley partial dislocations with Burgers vectors of $1/6 < 112 >$. The structure of interface is similar to the structure of Al (111) twist boundaries, as discussed in section 2.2.4, where three crossing sets of Shockley partial dislocations separate the regions of atoms with perfect FCC local crystal structure from the stacking fault regions, i.e., regions of atoms with HCP local crystal structure. The structure of interface in range (III) of misorientation angle is very similar to range (I). The only difference is that the partial dislocations separate two structurally different stacking fault regions from each other instead of a perfect FCC region from a stacking fault region. More information on the difference between the structure of interface in ranges (I) and (III) is provided in Appendix E. In range (II), however, the whole interface is composed of disordered atoms and no dislocations exist.

The images in the second column of figure 4.6 show the disordered atoms only. Comparing figures 4.6(a) and (b) with figures 4.6(g) and (h), it can be understood that for a constant initial interfacial distance, as misorientation angle increases, the distance between Shockley partial dislocations decreases, hence, the number of disordered atoms increases. Moreover, comparison of figures 4.6(a) and (b) with figures 4.6(d) and (e) shows that for a constant misorientation angle, the number of disordered atoms at the interface is increased at greater initial interfacial distances. Therefore, it may be concluded that although, the general pattern of atomic arrangements and the type of crystallographic defects is mainly a function of misorientation angle, the initial interfacial distance affects the density of disordered atoms.

Additionally, 2D maps of the local interfacial distance are shown in the third column of figure 4.6. These figures show that the interfacial distance is greater in the areas of disordered atoms, and, this distance is increased by increase in initial interfacial distance.

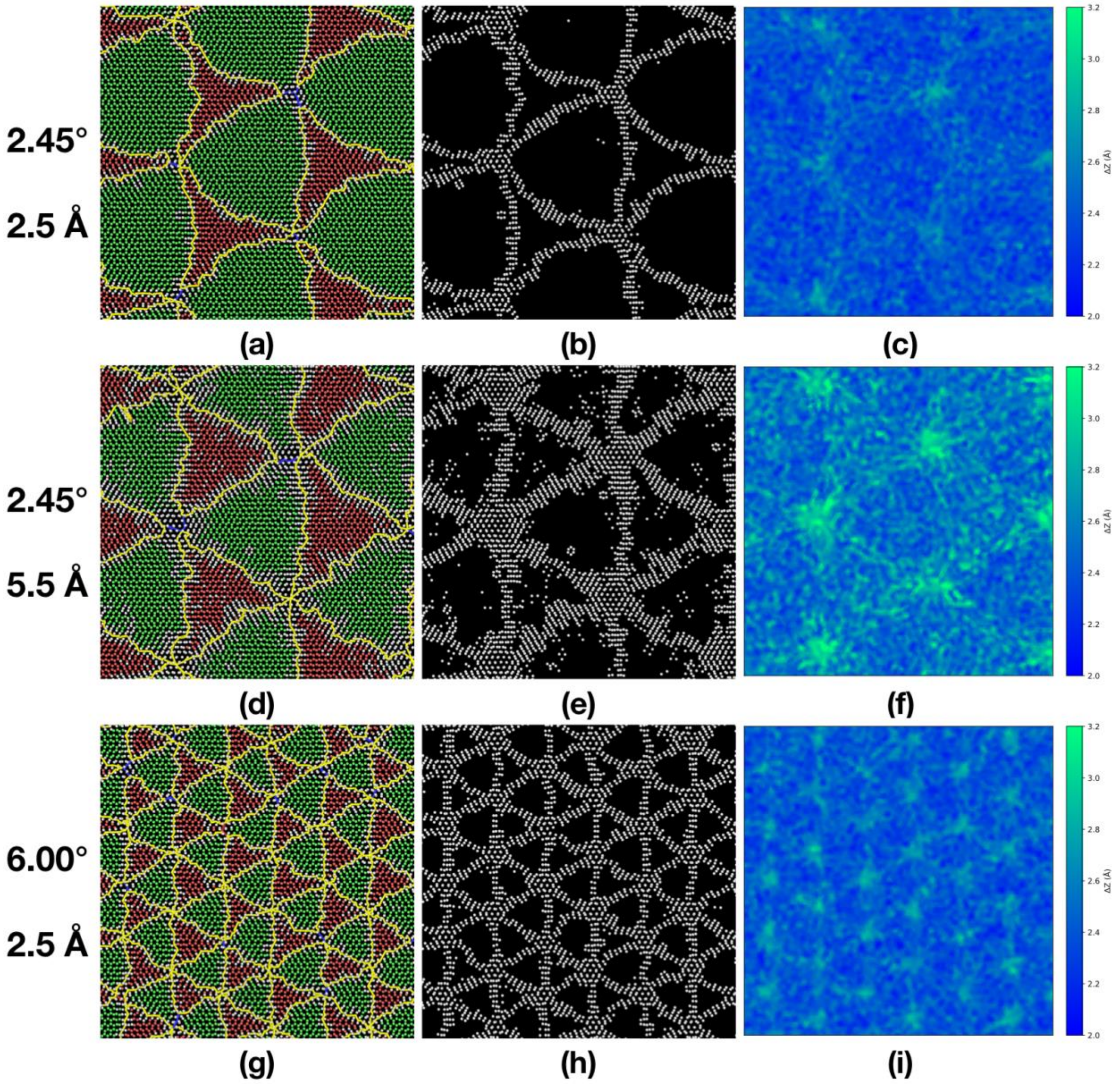
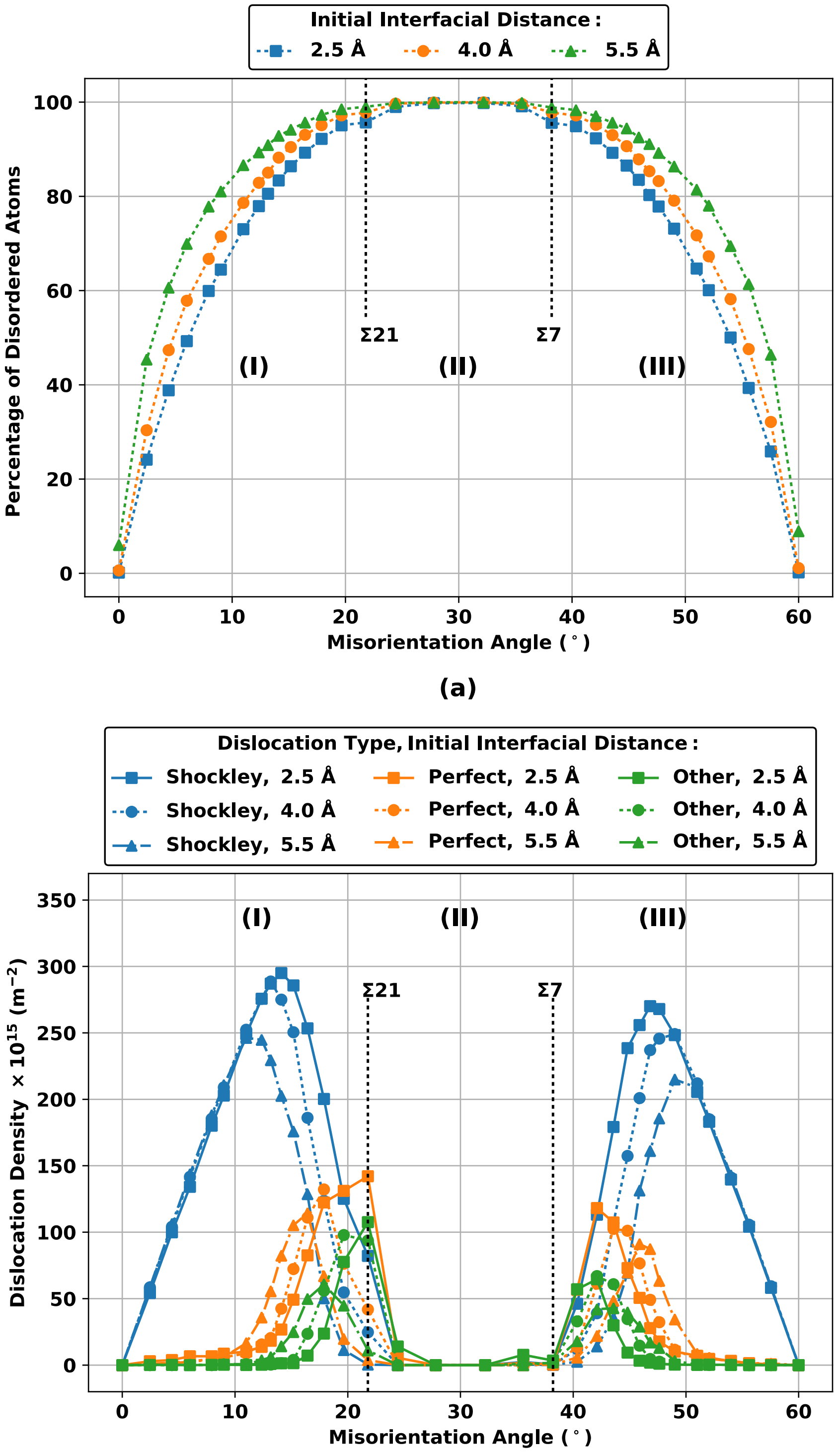


Figure 4.6: Top-view snapshots of equal size (150 Å × 150 Å), showing the final state of crystallographic defects at the interface (the first two columns) and local interfacial distance (the third column) in simulations with misorientation angles and initial interfacial distances of (a-c) 2.45°, 2.5 Å, (d-f) 2.45°, 5.5 Å, and (g-i) 6.00°, 2.5 Å. Atoms with FCC (green), HCP (red), unidentified (white) local crystal structure are indicated. Shockley partial (yellow) and perfect (blue) dislocations are specified. Detection of crystallographic defects is done with the use of dislocation analysis modifier [37] of OVITO [29].

4.2.2 Density of crystallographic defects

The density of point defects (i.e., disordered atoms) and various types of dislocations as a function of misorientation angle are calculated and shown in figure 4.7, for three different initial interfacial distances. As figure 4.7(a) shows, the percentage of disordered atoms increases as the misorientation angle deviates from $0°$ or $60°$, and it becomes approximately constant in range (II), where all atoms at the interface are disordered atoms. Figure 4.7(b) shows that as the misorientation angle deviates from $0°$ or $60°$, the density of Shockley partial dislocations with Burgers vectors of $1/6 < 112 >$ increases until they start to interact with each other and that is the point where perfect dislocations with Burgers vectors of $1/2 < 110 >$ start to appear. The interaction between Shockley partial dislocations is due to the decrease in the distance between them as misorientation angle increases, as discussed in the previous section. Dislocations that are named “other” in figure 4.7(b) are the dislocations formed as a result of interaction between Shockley partials and perfect dislocation. Figure 4.7 also shows that no dislocations exist in range (II) of misorientation angle. In other words, the mismatch between the two crystals cannot be accommodated by formation of dislocations. Thus, the interface is a typical planar defect in range (II) of misorientation angle, since all atoms are disordered atoms and no dislocations exist. Most importantly, the effect of initial interfacial distance on the density of defects can also be understood from figure 4.7. In ranges (I) and (III) of misorientation angle, the density of point defects and dislocations is a function of initial interfacial distance but independent of it in range (II).



(a)

(b)

Figure 4.7: Density of crystallographic defects as a function of misorientation angle. (a) and (b) show the density of point defects and various types of dislocations, respectively. Detection of crystallographic defects is done with the use of dislocation analysis modifier [37] of OVITO [29].

5 Conclusions

In this thesis, the contact between two Al substrates at the nanoscale is simulated using classical MD to analyze the JC behavior and generation of crystallographic defects at the interface as a result of contact. The critical distance for JC between (111) surfaces at room temperature is shown to be equal to 5.8 ± 0.05 Å, independent of misorientation angle between surfaces. It is shown that the critical distance for JC increases with temperature. Moreover, the effect of employed interatomic potential on the critical distance is discussed.

Although misorientation is shown to not have an effect on the critical distance for JC, it is found to affect the generation of crystallographic defects at the interface once the contact is formed and equilibrium is reached. The distribution of crystallographic defects is found to be distinctive in misorientation ranges of (I) $0° < \theta < 21.79°$, (II) $21.79° < \theta < 38.21°$, and (III) $38.21° < \theta < 60°$. In ranges (I) and (III), dislocations and point defects exist at the interface, while in the middle range, the interface is a 2D defect. The density of crystallographic defects is shown to be a function of initial interfacial distance between substrates in the middle ranges, but independent of it in range (II).

6 Future work

The JC phenomenon is a competition between short-range attractive forces between surfaces and the stiffness of substrates. Therefore, an interesting topic for future research would be to investigate how surface energy and elastic properties of material would affect the critical distance for JC. This can be achieved by comparing a few metals with each other.

It would also be interesting to see under what conditions, the interfacial dislocations can cross-slip away from the interface and enter the substrates. Parameters such as contact temperature, stacking fault energy of the material and the value of strain (due to existence of initial gap in the system) may be effective in that regard.

References

[1] Fortini, A., Mendelev, M. I., Buldyrev, S., and Srolovitz, D., 2008, "Asperity Contacts at the Nanoscale: Comparison of Ru and Au," J. Appl. Phys., **104**(7), p. 074320.

[2] Lesar, R., 2013, *Introduction To Computational Materials Science: Fundamentals to Applications*, Cambridge University Press.

[3] Gimzewski, J. K., and Möller, R., 1987, "Transition from the Tunneling Regime to Point Contact Studied Using Scanning Tunneling Microscopy," Phys. Rev. B, **36**(2), pp. 1284–1287.

[4] Untiedt, C., Caturla, M. J., Calvo, M. R., Palacios, J. J., Segers, R. C., and Van Ruitenbeek, J. M., 2007, "Formation of a Metallic Contact: Jump to Contact Revisited," Phys. Rev. Lett., **98**(20), pp. 1–4.

[5] Pethica, J. B., and Sutton, A. P., 1988, "On the Stability of a Tip and Flat at Very Small Separations," J. Vac. Sci. Technol. A Vacuum, Surfaces, Film., **6**(4), pp. 2490–2494.

[6] Smith, J., Bozzolo, G., Banerjea, A., and Ferrante, J., 1989, "Avalanche in Adhesion," Phys. Rev. Lett., **63**(12), pp. 1269–1272.

[7] Cha, P.-R., Srolovitz, D. J., and Kyle Vanderlick, T., 2004, "Molecular Dynamics Simulation of Single Asperity Contact," Acta Mater., **52**(13), pp. 3983–3996.

[8] Song, J., and Srolovitz, D. J., 2007, "Atomistic Simulation of Multicycle Asperity Contact," Acta Mater., **55**(14), pp. 4759–4768.

[9] Agraït, N., Rodrigo, J. G., and Vieira, S., 1993, "Conductance Steps and Quantization in Atomic-Size Contacts," Phys. Rev. B, **47**(18), pp. 12345–12348.

[10] Krans, J. M., Muller, C. J., Yanson, I. K., Govaert, T. C. M., Hesper, R., and Van Ruitenbeek, J. M., 1993, "One-Atom Point Contacts," Phys. Rev. B, **48**(19), pp. 14721–14724.

[11] Voets, J., Keijsers, R. J. P., Shklyarevskii, O. I., and van Kempen, H., 1996, "Effects of Electrode Interactions Observed in a Mechanically Controllable Break Junction," Phys. Rev. B, **53**(3), pp. 1072–1075.

[12] Landman, U., Luedtke, W. D., Burnham, N. A., and Colton, R. J., 1990, "Atomistic Mechanisms and Dynamics of Adhesion, Nanoindentation, and Fracture," Science (80-.)., **248**(4954), pp. 454–461.

[13] Miyata, K., Kushida, T., Omura, T., and Komizo, Y., 2003, "Coarsening Kinetics of Multicomponent MC-Type Carbides in High-Strength Low-Alloy Steels," Metall. Mater. Trans. A, **34**(August), pp. 1565–1573.

[14] Gholamirad, M., Soltani, S., and Sepehrband, P., 2018, "Dislocation Assisted Diffusion: A Mechanism for Growth of Intermetallic Compounds in Copper Ball Bonds," Microelectron. Reliab., **81**(December 2017), pp. 210–217.

[15] Huang, J., Meyer, M., and Pontikis, V., 1989, "Is Pipe Diffusion in Metals Vacancy Controlled? A Molecular Dynamics Study of an Edge Dislocation in Copper," Phys. Rev. Lett., **63**(6), pp. 628–631.

[16] Agraït, N., Yeyati, A. L., and van Ruitenbeek, J. M., 2003, "Quantum Properties of Atomic-Sized Conductors," Phys. Rep., **377**(2–3), pp. 81–279.

[17] Lu, Y., Huang, J. Y., Wang, C., Sun, S., and Lou, J., 2010, "Cold Welding of Ultrathin Gold Nanowires," Nat. Nanotechnol., **5**(3), pp. 218–224.

[18] Mishin, Y., Asta, M., and Li, J., 2010, "Atomistic Modeling of Interfaces and Their Impact on Microstructure and Properties," Acta Mater., **58**(148), pp. 1117–1151.

[19] Tschopp, M. A., Coleman, S. P., and McDowell, D. L., 2015, "Symmetric and Asymmetric Tilt Grain Boundary Structure and Energy in Cu and Al (and Transferability to Other Fcc Metals)," Integr. Mater. Manuf. Innov., **4**(1), p. 11.

[20] Dai, S., Xiang, Y., and Srolovitz, D., 2014, "Atomistic, Generalized Peierls–Nabarro and Analytical Models for (1 1 1) Twist Boundaries in Al, Cu and Ni for All Twist Angles," Acta Mater., **69**, pp. 162–174.

[21] Olmsted, D. L., 2009, "A New Class of Metrics for the Macroscopic Crystallographic Space of Grain Boundaries," Acta Mater., **57**(9), pp. 2793–2799.

[22] Olmsted, D. L., Foiles, S. M., and Holm, E. A., 2009, "Survey of Computed Grain Boundary Properties in Face-Centered Cubic Metals: I. Grain Boundary Energy," Acta Mater., **57**(13),

pp. 3694–3703.

[23] Read, W. T., and Shockley, W., 1950, "Dislocation Models of Crystal Grain Boundaries," Phys. Rev., **78**(3), pp. 275–289.

[24] Vitek, V., 1987, "On the Difference between the Misorientation Dependences of the Energies of Tilt and Twist Boundaries," Scr. Metall., **21**(5), pp. 711–714.

[25] Saylor, D. M., El Dasher, B. S., Rollett, A. D., and Rohrer, G. S., 2004, "Distribution of Grain Boundaries in Aluminum as a Function of Five Macroscopic Parameters," Acta Mater., **52**(12), pp. 3649–3655.

[26] Wolf, D., 1985, "On the Relationship Between Symmetrical Tilt, Twist, 'Special', and 'Favored' Grain Boundaries," J. Phys. Colloq., **46**, pp. C4-197-C4-211.

[27] Ovrutsky, A. M., Prokhoda, A. S., and Rasshchupkyna, M. S., 2014, *Computational Materials Science,* Elsevier.

[28] Plimpton, S., 1995, "Fast Parallel Algorithms for Short – Range Molecular Dynamics," J. Comput. Phys., **117**(June 1994), pp. 1–19.

[29] Stukowski, A., 2010, "Visualization and Analysis of Atomistic Simulation Data with OVITO–the Open Visualization Tool," Model. Simul. Mater. Sci. Eng., **18**(1), p. 015012.

[30] Daw, M. S., and Baskes, M. I., 1984, "Embedded-Atom Method: Derivation and Application to Impurities, Surfaces, and Other Defects in Metals Murray," Phys. Rev. B, **29**(12).

[31] Verlet, L., 1967, "Computer 'Experiments' on Classical Fluids. I. Thermodynamical Properties of Lennard-Jones Molecules," Phys. Rev., **159**(1).

[32] Nose, S., 1984, "A Molecular Dynamics Method for Simulations in the Canonical Ensemble.Pdf," Mol. Phys., **52**(2).

[33] Hoover, W. G., 1985, "Canonical Dynamics: Equilibrium Phase-Space Distributions," Phys. Rev. A, **31**(3), pp. 1695–1697.

[34] Hoover, W. G., 1986, "Constant-Pressure Equations of Motion," Phys. Rev. A, **34**(3), pp. 2499–2500.

[35] Stukowski, A., 2012, “Structure Identification Methods for Atomistic Simulations of Crystalline Materials,” Model. Simul. Mater. Sci. Eng., **20**(2012), p. 045021.

[36] Honeycutt, J. D., and Andersen, H. C., 1987, “Molecular Dynamics Study of Melting and Freezing of Small Lennard-Jones Clusters,” J. Phys. Chem., **91**(19), pp. 4950–4963.

[37] Stukowski, A., Bulatov, V. V, and Arsenlis, A., 2012, “Automated Identification and Indexing of Dislocations in Crystal Interfaces,” Model. Simul. Mater. Sci. Eng., **20**, p. 085007.

[38] Sheng, H. W., Kramer, M. J., Cadien, A., Fujita, T., and Chen, M. W., 2011, “Highly Optimized Embedded-Atom-Method Potentials for Fourteen FCC Metals,” Phys. Rev. B - Condens. Matter Mater. Phys., **83**(13), pp. 1–20.

[39] Miesbauer, O., Götzinger, M., and Peukert, W., 2003, “Molecular Dynamics Simulations of the Contact between Two NaCl Nano-Crystals: Adhesion, Jump to Contact and Indentation,” Nanotechnology, **14**, pp. 371–376.

[40] Mishin, Y., Farkas, D., Mehl, M. J., and Papaconstantopoulos, D. A., 1999, “Interatomic Potentials for Monoatomic Metals from Experimental Data and *Ab Initio* Calculations,” Phys. Rev. B, **59**(5), pp. 3393–3407.

[41] Heinz, H., Vaia, R. A., Farmer, B. L., and Naik, R. R., 2008, “Accurate Simulation of Surfaces and Interfaces of Face-Centered Cubic Metals Using 12−6 and 9−6 Lennard-Jones Potentials,” J. Phys. Chem. C, **112**(44), pp. 17281–17290.

Appendix A: Python script for steps 1 and 2 of simulation process

The following Python script corresponds to the steps 1 and 2 of simulation process section. Further information is provided within the script.

```python
# importing numpy library
import numpy as np

# function for creating an FCC substrate containing (111) planes along z direction
# this function is taken from:
# https://icme.hpc.msstate.edu/mediawiki/index.php/FCC_(111)_Surface_Generation_Script
def generate_substrate(a,nx,ny,nz,t,zi):
    # atom coordinates and types
    x=[]; y=[]; z=[]; ty=[]
    ax = a/np.sqrt(2)
    ay = a/np.sqrt(2)*np.sqrt(3)
    az = a*np.sqrt(3)
    x0 = 0.0
    x2 = np.sqrt(2)/4*a
    y2 = np.sqrt(6)/4*a
    y3 = np.sqrt(6)/6*a
    y4 = np.sqrt(6)*5/12*a
    y5 = np.sqrt(6)*2/6*a
    y6 = np.sqrt(6)/12*a
    xmin,xmax,ymin,ymax,zmin,zmax=-ax*nx,ax*nx,-ay*ny,ay*ny,zi,zi+az*nz
    for i in range(-nx,nx):
        for j in range(-ny,ny):
            layer = 0
            for k in range(0,nz):
                x.append(x0+i*ax); y.append(x0+j*ay); z.append(zi+layer/3.0*az)
                ty.append(t)
                x.append(x2+i*ax); y.append(y2+j*ay); z.append(zi+layer/3.0*az)
                ty.append(t); layer += 1
                x.append(x0+i*ax); y.append(y3+j*ay); z.append(zi+layer/3.0*az)
                ty.append(t)
                x.append(x2+i*ax); y.append(y4+j*ay); z.append(zi+layer/3.0*az)
                ty.append(t); layer += 1
                x.append(x0+i*ax); y.append(y5+j*ay); z.append(zi+layer/3.0*az)
                ty.append(t)
                x.append(x2+i*ax); y.append(y6+j*ay); z.append(zi+layer/3.0*az)
                ty.append(t); layer += 1
    return ty,x,y,z,xmin,xmax,ymin,ymax,zmin,zmax

# function for generating a LAMMPS datafile
# this function is taken from:
# https://icme.hpc.msstate.edu/mediawiki/index.php/FCC_(111)_Surface_Generation_Script
def datafile(name,ty1,x1,y1,z1,ty2,x2,y2,z2,xmin,xmax,ymin,ymax,zmin,zmax):
    fout = open(name,"w")
    fout.write("Atomic locations\n\n")
    fout.write("%d atoms\n"%(len(x1)+len(x2)))
    fout.write("6 atom types\n")
    fout.write(" %22.16f  %22.16f   xlo xhi\n"%(xmin,xmax))
    fout.write(" %22.16f  %22.16f   ylo yhi\n"%(ymin,ymax))
    fout.write(" %22.16f  %22.16f   zlo zhi\n"%(zmin,zmax))
    fout.write("\nAtoms\n\n")
    for i in range(len(x1)):
        fout.write("%4d %3d %22.16f %22.16f %22.16f\n"%(i+1,ty1[i],x1[i],y1[i],z1[i]))
```

```
    for j in range(len(x2)):
        fout.write("%4d %3d %22.16f %22.16f %22.16f\n"%(i+j+2,ty2[j],x2[j],y2[j],z2[j]))
    fout.close()

# function for rotating a supercell by phi degrees around z axis
def rotate(x,y,z,phi):
    xn = []; yn = []; zn = []
    for i in range(len(x)):
        xn.append(np.cos(phi)*x[i]+np.sin(phi)*y[i])
        yn.append(-np.sin(phi)*x[i]+np.cos(phi)*y[i])
        zn.append(z[i])
    return xn,yn,zn

# main function
def main(theta,n,name):
    a = 4.05
    theta_rad = theta*np.pi/180
    # creating lower and upper supercells
    # nz*3 is the number of atomic layers in each slab
    nx = 5*n; ny = 5*n; nz = 4
    # z1 is the size of gap below lower slab and above upper slab
    ty = 1; z1 = 10
    type_ls,x_ls,y_ls,z_ls,xmin_ls,xmax_ls,ymin_ls,ymax_ls,zmin_ls,zmax_ls = generate_substrate(a,nx,ny,nz,ty,z1)
    ty = 2; z2 = zmax_ls+20
    type_us,x_us,y_us,z_us,xmin_us,xmax_us,ymin_us,ymax_us,zmin_us,zmax_us = generate_substrate(a,nx,ny,nz,ty,z2)
    # rotating supercells by theta/2 degree in opposite directions around x-azis
    xnew_ls,ynew_ls,znew_ls = rotate(x_ls,y_ls,z_ls,theta_rad/2)
    xnew_us,ynew_us,znew_us = rotate(x_us,y_us,z_us,-theta_rad/2)
    # calculating periodic distances and simulation cell lengths
    if theta < 30:
        # Burgurs vector of perfect dislocation
        b = a/np.sqrt(2)
        ax = b
        ay = np.sqrt(3)*b
        lx = n * ax / np.cos(theta_rad/2)
        ly = n * ay / np.cos(theta_rad/2)
    elif theta > 30:
        b = a/np.sqrt(2)
        ax = np.sqrt(3)*b
        ay = b
        lx = n * ax / np.cos(np.pi/6-theta_rad/2)
        ly = n * ay / np.cos(np.pi/6-theta_rad/2)
    # removing atoms that are outside of the simulation cell box
    xmin = 1; xmax = lx+1
    ymin = 1; ymax = ly+1
    zmin = zmin_ls-z1; zmax = zmax_us+z1
    xnewest_ls=[]; ynewest_ls=[]; znewest_ls=[]
    xnewest_us=[]; ynewest_us=[]; znewest_us=[]
    for i in range(len(xnew_ls)):
        if (xmin<xnew_ls[i]<=xmax) and (ymin<ynew_ls[i]<=ymax):
            xnewest_ls.append(xnew_ls[i])
            ynewest_ls.append(ynew_ls[i])
            znewest_ls.append(znew_ls[i])
    for j in range(len(xnew_us)):
        if (xmin<xnew_us[j]<=xmax) and (ymin<ynew_us[j]<=ymax):
            xnewest_us.append(xnew_us[j])
            ynewest_us.append(ynew_us[j])
            znewest_us.append(znew_us[j])
```

```
        datafile(name,type_ls,xnewest_ls,ynewest_ls,znewest_ls,type_us,xnewest_us,ynewest_us,znewest_us,xmin,xmax,ymin,ymax,zmin,zmax)

    # main code
    # main(theta,n), where n is the required periodicity in x and y directions
    # theta = 2*arctan(sqrt(3)/n) for theta<30
    # theta = 2*(30-arctan(sqrt(3)/n)) for theta>30
    main(0.00,3,"0.00")
    main(2.45,81,"2.45")
    main(4.40,45,"4.40")
    main(5.36,37,"5.36")
    main(6.00,33,"6.00")
    main(6.60,30,"6.60")
    main(7.34,27,"7.34")
    main(7.93,25,"7.93")
    main(8.26,24,"8.26")
    main(8.61,23,"8.61")
    main(9.00,22,"9.00")
    main(9.43,21,"9.43")
    main(9.88,20,"9.88")
    main(10.42,19,"10.42")
    main(10.99,18,"10.99")
    main(11.63,17,"11.63")
    main(12.36,16,"12.36")
    main(13.17,15,"13.17")
    main(14.11,14,"14.11")
    main(15.18,13,"15.18")
    main(16.43,12,"16.43")
    main(17.89,11,"17.89")
    main(19.65,10,"19.65")
    main(21.79,9,"21.79")
    main(24.43,8,"24.43")
    main(27.80,7,"27.80")

    main(32.20,7,"32.20")
    main(35.57,8,"35.57")
    main(38.21,9,"38.21")
    main(40.35,10,"40.35")
    main(42.11,11,"42.11")
    main(43.57,12,"43.57")
    main(44.82,13,"44.82")
    main(45.89,14,"45.89")
    main(46.83,15,"46.83")
    main(47.64,16,"47.64")
    main(48.37,17,"48.37")
    main(49.01,18,"49.01")
    main(49.58,19,"49.58")
    main(50.12,20,"50.12")
    main(50.57,21,"50.57")
    main(51.00,22,"51.00")
    main(51.39,23,"51.39")
    main(51.74,24,"51.74")
    main(52.07,25,"52.07")
    main(52.66,27,"52.66")
    main(53.40,30,"53.40")
    main(54.00,33,"54.00")
    main(54.64,37,"54.64")
    main(55.60,45,"55.60")
    main(57.55,81,"57.55")
    main(60.00,3,"60.00")
```

Appendix B: LAMMPS script for step 3 of simulation process

The following LAMMPS script corresponds to the step 3 of simulation process section. Further information is provided within the script.

```
1.  # defining unit system
2.  units metal
3.  # 3D simulation cell
4.  dimension 3
5.  # periodic boundary conditions along x and y directions, fixed z direction
6.  boundary p p f
7.  # defining particle style
8.  atom_style atomic
9.  # defining parameters needed for building pairwise neighbor lists, 2.0 Å is the extra distance beyond the interatomic potential cutoff
10. neighbor 2.0 bin
11. neigh_modify every 1 delay 5 check yes
12. # reading datafile which contains IDs and coordinates of atoms
13. # each datafile corresponds to one specific misorientation angle
14. # depending on the datafile, lines 17, 19, 20, 25, 26, 31, 32, 36, and 37 should be modified
15. read_data datafile
16. # replicating simulation cell along periodic directions, so approximately 300000 atoms exist in the simulation cell
17. replicate 1 1 1
18. # defining regions and groups of atoms for the fixed and mobile atoms in the lower slab
19. region rLoSlFix block INF INF INF INF 9 13 units box
20. region rLoSlMob block INF INF INF INF 14 36 units box
21. group gLoSlFix region rLoSlFix
22. group gLoSlMob region rLoSlMob
23. group gLoSl union gLoSlMob gLoSlFix
24. # defining regions and groups of atoms for the fixed and mobile atoms in the upper slab
25. region rUpSlFix block INF INF INF INF 73 76 units box
26. region rUpSlMob block INF INF INF INF 49 72 units box
27. group gUpSlFix region rUpSlFix
28. group gUpSlMob region rUpSlMob
29. group gUpSl union gUpSlMob gUpSlFix
30. # defining regions and groups of atoms for the first and last Newtonian atomic layers in the lower slab
31. region rLoSlF block INF INF INF INF 14 15 units box
32. region rLoSlL block INF INF INF INF 35 36 units box
33. group gLoSlF region rLoSlF
34. group gLoSlL region rLoSlL
35. # defining regions and groups of atoms for the first and last Newtonian atomic layers in the upper slab
36. region rUpSlF block INF INF INF INF 71 72 units box
37. region rUpSlL block INF INF INF INF 49 51 units box
38. group gUpSlF region rUpSlF
39. group gUpSlL region rUpSlL
40. # defining groups of atoms for all mobile atoms and all fixed atoms
```

```
group gMob union gLoSlMob gUpSlMob
group gFix union gLoSlFix gUpSlFix
# defining atom types for different layers only for the purpose of coloring
set group gLoSlFix type 1
set group gLoSlMob type 2
set group gLoSlL type 3
set group gUpSlMob type 5
set group gUpSlL type 4
set group gUpSlFix type 6
# defining the style of interaction between atoms
pair_style eam/alloy
# reading the interatomic potential file
pair_coeff * * Al.lammps.eam Al Al Al Al Al Al
# defining computations for temperature and pressure
compute TLF gLoSlFix temp
compute TLM gLoSlMob temp
compute TUF gUpSlFix temp
compute TUM gUpSlMob temp
compute TMob gMob temp
compute PMob all pressure TMob
# defining computations for everage of z positions of atoms
compute AZLSF gLoSlF reduce ave z
compute AZLSL gLoSlL reduce ave z
compute AZUSF gUpSlF reduce ave z
compute AZUSL gUpSlL reduce ave z
# outputting the values of thermodynamic variables and computations each 1000 timesteps
thermo 1000
thermo_style custom step etotal pe ke lx ly lz vol temp c_TLF c_TLM c_TUF c_TUM c_TMob press pxx pyy pzz c_PMob c_PMob[1] c_PMob[2] c_PMob[3] c_PMob[4] c_PMob[5] c_PMob[6] c_AZLSF c_AZLSL c_AZUSF c_AZUSL atoms
# minimzation of energy
min_style cg
minimize 1.0e-4 1.0e-6 1000 10000
# outputting the coordinates, force in z direction, and velocity in z direction each 1000 timesteps
dump d1 all custom 1000 r1.txt id type x y z fz vz
# difining timestep to be 2 femtosecond
timestep 0.002
# restting timestep value to zero
reset_timestep 0
# assigning velocity to atoms based on gaussian distribution
velocity gMob create 300.0 8930793 dist gaussian
# assinging velocity of zero to fixed atoms
velocity gLoSlFix set 0.0 0.0 0.0 units box
velocity gUpSlFix set 0.0 0.0 0.0 units box
# fixing the fixed atoms during the dynamic run
fix f1 gLoSlFix setforce 0.0 0.0 0.0
fix f2 gUpSlFix setforce 0.0 0.0 0.0
# integrating Newton's equations of mothion using the velocity-verlet algorithm in an NPT ensemble
fix f3 gMob npt temp 300.0 300.0 1.0 x 0.0 0.0 1.0 y 0.0 0.0 1.0 couple xy
fix_modify f3 temp TMob
```

```
# initializing the dynamic run for 10000 timesteps
run 10000
# unfixing the pre-defined fixes
unfix f1
unfix f2
unfix f3
# writing the current state of MD simulation in a restart file
# the restart file will be used as an input for the simulations of step 4
write_restart rf1.txt
```

Appendix C: LAMMPS script for step 4 of simulation process

The following LAMMPS script corresponds to the step 4 of simulation process section. Further information is provided within the script.

```
# defining varibales for running simulations
# this set of variables will run the code for initial interfacial distances of 2.5 Å, 4 Å, and 5.5 Å
variable d1 equal 2.50
variable d equal ${d1}
variable interval equal 1.50
variable d2 equal 5.50
# defining a label for the line so we can jump to this line again and run the code from this line
label lab1
# clear all variables in the memory except the equal style variables
clear
# read the restarl file which is the final state of simulation of step 3
read_restart rf1.txt
# defining parameters needed for building pairwise neighbor lists, 2.0 Å is the extra distance beyond the interatomic potential cutoff
neighbor 2.0 bin
neigh_modify every 1 delay 5 check yes
# defining the style of interaction between atoms
pair_style eam/alloy
# reading the interatomic potential file
pair_coeff * * Al.lammps.eam Al Al Al Al Al Al
# making a directory with the name of initial interfacial distance and entering that directory
shell mkdir ${d}
shell cd ${d}
# defining computations for temperature and pressure
compute TLF gLoSlFix temp
compute TLM gLoSlMob temp
compute TUF gUpSlFix temp
compute TUM gUpSlMob temp
compute TMob gMob temp
compute PMob all pressure TMob
# defining computations for everage of z positions of atoms
compute AZLSF gLoSlF reduce ave z
compute AZLSL gLoSlL reduce ave z
compute AZUSF gUpSlF reduce ave z
compute AZUSL gUpSlL reduce ave z
# outputting the values of thermodynamic variables and computations each 10 timesteps
thermo 10
thermo_style custom step etotal pe ke lx ly lz vol temp c_TLF c_TLM c_TUF c_TUM c_TMob press pxx pyy pzz c_PMob c_PMob[1] c_PMob[2] c_PMob[3] c_PMob[4] c_PMob[5] c_PMob[6] c_AZLSF c_AZLSL c_AZUSF c_AZUSL atoms
# outputting the coordinates, force in z direction, and velocity in z direction each 1000 timesteps
```

```
dump d1 all custom 1000 r2.txt id type x y z fz vz
# running the simulation for 0 timestep just for the values of computations get updated
run 0
# displacing the upper slab downwards to the desired distance from the lower slab
variable m equal -(c_AZUSL-c_AZLSL)+${d}
displace_atoms gUpSl move 0.0 0.0 $m units box
# change the name of the file which is being used for outputting the values of thermodynamic variables and computations
log log2.lammps
# difining timestep to be 2 femtosecond
timestep 0.002
# restting timestep value to zero
reset_timestep 0
# fixing the fixed atoms during the dynamic run
fix f1 gLoSlFix setforce 0.0 0.0 0.0
fix f2 gUpSlFix setforce 0.0 0.0 0.0
# integrating Newton's equations of mothion using the velocity-verlet algorithm in an NPT ensemble
fix f3 gMob npt temp 300.0 300.0 1.0 x 0.0 0.0 1.0 y 0.0 0.0 1.0 couple xy
fix_modify f3 temp TMob
# initializing the dynamic run for 10000 timesteps
run 10000
# unfixing the pre-defined fixes
unfix f1
unfix f2
unfix f3
# writing the current state of MD simulation in a restart file just in case it is needed later
write_restart rf2.txt
# going back one directory
shell cd ..
# defining the finishing condition for the simulations which corresponds to the last initial interfacial distance
if "${d}==${d2}" then quit
# assigning the next value in the list to the initial interfacial distance variable
variable d equal ${d}+${interval}
# jump to the line of label lab1 within the same script (filename: v2.lampps)
jump v2.lammps lab1
```

Appendix D: supplementary material for section 4.1.2.3

Figure D.1 is a supplementary material for figure 4.4 which shows that the effect of crystallographic misorientation on the critical distance for JC is insignificant. It also shows that the misorientation has an effect on the value of D_{ave} for the cases that JC occurs. A more in-depth discussion is provided in section 4.1.2.3.

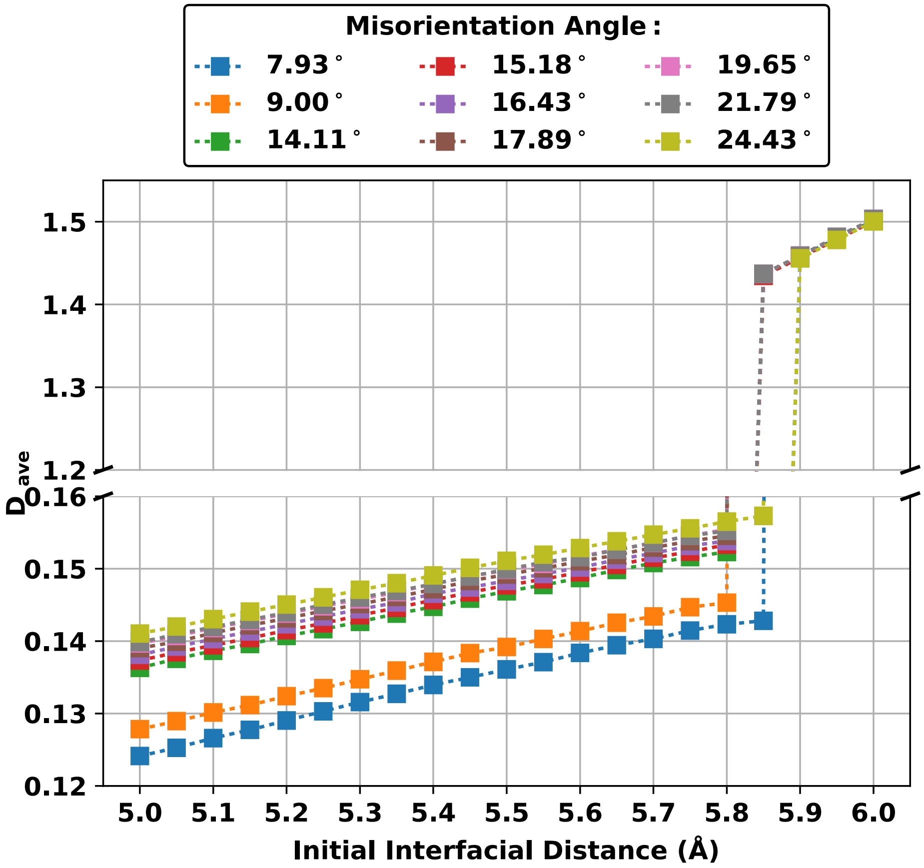
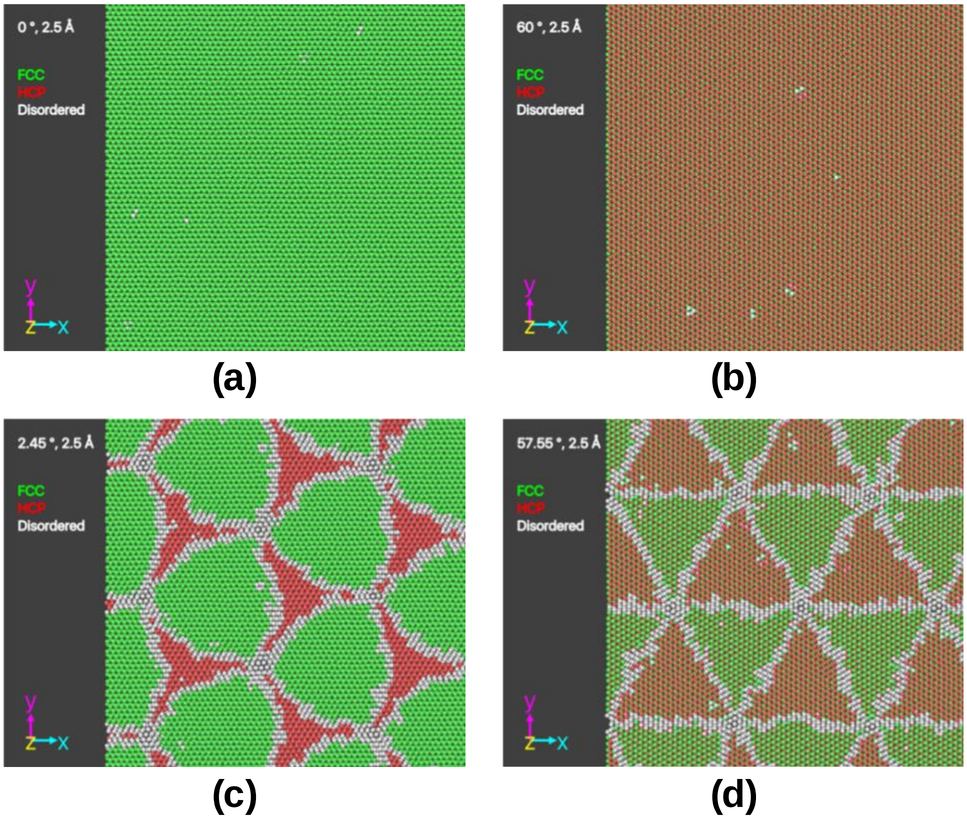


Figure D.1: Plot of D_{ave} versus initial interfacial distance, for some other misorientation angles between 0° and 30°, which are not shown in figure 4.4.

Appendix E: supplementary material for section 4.2.1

Figure E.1 is a supplementary material for section 4.2.1 which shows the difference in the structure of interface for ranges (I) and (III) of misorientation angle.



(a) (b) (c) (d)

Figure E.1: Top-view snapshots of equal size. showing the final state of the structure of interface (the two contacting layers) in simulations with misorientation angles of (a) 0°, (b) 60°, (c) 2.45° and (d) 57.55°. The initial interfacial distance is 2.5 Å for all of them. Atoms with FCC (green), HCP (red), and unidentified (white) local crystal structure are indicated.