

1-1-2009

# Nanoindentation simulations for large atomic systems : an analysis of molecular dynamics and bridged finite element - molecular dynamics methodologies

Jonathan Vandersluis  
*Ryerson University*

Follow this and additional works at: <http://digitalcommons.ryerson.ca/dissertations>



Part of the [Aerospace Engineering Commons](#)

---

## Recommended Citation

Vandersluis, Jonathan, "Nanoindentation simulations for large atomic systems : an analysis of molecular dynamics and bridged finite element - molecular dynamics methodologies" (2009). *Theses and dissertations*. Paper 635.

This Thesis is brought to you for free and open access by Digital Commons @ Ryerson. It has been accepted for inclusion in Theses and dissertations by an authorized administrator of Digital Commons @ Ryerson. For more information, please contact [bcameron@ryerson.ca](mailto:bcameron@ryerson.ca).

88  
173-39  
V26  
2009

**NANOINDENTATION SIMULATIONS FOR LARGE ATOMIC SYSTEMS:  
AN ANALYSIS OF MOLECULAR DYNAMICS AND BRIDGED  
FINITE ELEMENT – MOLECULAR DYNAMICS METHODOLOGIES**

By

Jonathan Vandersluis, B. Eng  
Aerospace Engineering  
Ryerson University, 2009

A thesis presented to Ryerson University

In partial fulfillment of the requirements for the degree of  
Master's of Applied Science  
in the Program of  
Aerospace Engineering

Toronto, Ontario, Canada, 2009

© Jonathan Vandersluis



# Authors Declaration

---

I hereby declare I am the sole author of this thesis.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

---

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

---

# Abstract

---

## **Nanoindentation Simulations for Large Atomic Systems: An Analysis of Molecular Dynamics and Bridged Finite Element – Molecular Dynamics Methodologies.**

Jonathan Vandersluis, Master of Applied Science, Aerospace Engineering  
Ryerson University, Toronto, 2009

This thesis develops a molecular dynamics (MD) custom made computational tool to perform nanoindentation simulations on copper nanomaterials, a Face Centred Cubic (FCC) metal. The Embedding Atom Method (EAM) is used to model the interatomic forces with the substrate. Further, a bridged finite element – molecular dynamics (FE-MD) simulation tool is also adapted to perform nanoindentation experimentation. Using this bridged FE-MD simulation tool, nanoindentations are performed much more effectively than the MD simulations while saving substantial computational simulation time. While the MD simulation experienced difficulties capturing the behaviour of the system during indentation especially at faster indentation speeds, the bridged FE-MD method is capable of reaching a state of equilibrium within a single step for each indentation depth interval analyzed throughout the nanoindentation. Although the hardness values for these simulations cannot be obtained without larger scale simulations using more powerful computational resources, the simulations provide insight into the behaviour of the copper nanomaterial during nanoindentation. As a result, it is clear that the bridged FE-MD nanoindentation tool is much more effective for executing nanoindentation simulations than the traditional MD methodologies.

# Acknowledgements

---

I would like to take this opportunity to first thank my supervisor, Dr. Kamran Behdinin, and my co-supervisor, Dr. Zouheir Fawaz, who have both been very supportive and provided much guidance throughout the past two years. I would also like to thank Dr. Kai-Uwe Bletzinger and Dr. Roman Lackner, who gave me the opportunity to study with them in Munich, Germany at the Technical University of Munich for four months during the summer of 2008. I am very grateful for the privilege to be able to work on this research at this esteemed university and am thankful that I was welcomed there with open arms. I would also like to thank all my friends and colleagues, for their guidance and helping me stay motivated with their words of encouragement. Lastly, I would like to thank my family, most notably my parents, for all their love and support throughout my life. They have always pushed me to be the best that I could be and I am sure that I would not have been able to accomplish half as much without them. I would like to dedicate this thesis to my Zaidy, who has recently passed away from pancreatic cancer. Despite being diagnosed with a terminal form of cancer, his continued work ethic and positive attitude was truly remarkable. He has always been of great inspiration to me and will be sadly missed.

# Table of Contents

---

<b>1. Introduction.....</b>	<b>1</b>
<b>2. Molecular Dynamics Nanoindentation .....</b>	<b>4</b>
2.1. <i>Molecular Dynamics Introduction.....</i>	4
2.2. <i>Embedding Atom Method for Interatomic Potential.....</i>	5
2.3. <i>Custom Molecular Dynamics Code NI_MD_V_7_00.....</i>	7
2.3.1. MATLAB <sup>TM</sup> Function NI_MD_V_7_00.....	7
2.3.2. MATLAB <sup>TM</sup> Crystal Structure Functions.....	8
2.3.3. MATLAB <sup>TM</sup> Function Indenter_Design_V_7_00.....	13
2.3.4. MATLAB <sup>TM</sup> Function Indenter_Move_V_7_00.....	18
2.3.5. MATLAB <sup>TM</sup> Function Contact_V_7_00.....	18
2.3.6. MATLAB <sup>TM</sup> Function Contact_Area_V_7_00.....	21
2.3.7. MATLAB <sup>TM</sup> Function Force_Summation_V_7_00.....	23
2.4. <i>Molecular Dynamics Periodic Boundary Conditions.....</i>	23
2.5. <i>Additional Nanoindentation Theory .....</i>	25
<b>3. Molecular Dynamics Nanoindentation Results.....</b>	<b>27</b>
3.1. <i>Molecular Dynamics Nanoindentation Introduction.....</i>	27
3.2. <i>Molecular Dynamics Sensitivity Analysis.....</i>	28
3.2.1. Temperature Sensitivity Analysis.....	29
3.2.2. Force Sensitivity Analysis .....	31
3.2.3. Molecular Dynamics Displacement Sensitivity Analysis.....	33
3.2.4. Clamping Sensitivity Analysis.....	38
3.3. <i>Molecular Dynamics Nanoindentation Simulation Conditions .....</i>	39
3.4. <i>Molecular Dynamics Spherical Indenter Results .....</i>	41
3.5. <i>Molecular Dynamics Berkovich Indenter Results.....</i>	43
3.6. <i>Molecular Dynamics Square Pyramidal Indenter Results.....</i>	45
3.7. <i>Discussion.....</i>	47

3.8.	<i>Chapter Summary</i> .....	50
<b>4.</b>	<b>Bridged Finite Element – Molecular Dynamics Nanoindentation</b> .....	<b>51</b>
4.1.	<i>Bridged Finite Element – Molecular Dynamics Introduction</i> .....	51
4.2.	<i>Traditional Finite Elements Method Theory</i> .....	52
4.2.1.	General Form of Finite Element Method Simulations.....	52
4.2.2.	General Form of a FEM Stiffness Equation .....	53
4.2.3.	Traditional FEM Discretization Scheme .....	55
4.3.	<i>Bridged Finite Element – Molecular Dynamics Theory</i> .....	57
4.3.1.	Bridged Finite Element – Molecular Dynamics Discretization Scheme .....	57
4.3.2.	Embedding Atom Method Potentials in Bridged Finite Elements – Molecular Dynamics .....	61
4.4.	<i>Custom Finite Element – Molecular Dynamics Code FE_MD_V_7_00</i> .....	62
4.5.	<i>Custom Finite Element – Molecular Dynamics Nanoindentation Code NI_FE_MD_V_7_00</i> .....	63
4.6.	<i>Chapter Summary</i> .....	65
<b>5.</b>	<b>Bridged Finite Element – Molecular Dynamics Nanoindentation</b> .....	<b>66</b>
5.1.	<i>Bridged Finite Element – Molecular Dynamics Nanoindentation Introduction</i> .....	66
5.2.	<i>Bridged Finite Element –Molecular Dynamics Sensitivity Analysis</i> .....	67
5.2.1.	Indenter Increment Size Sensitivity Analysis .....	67
5.2.2.	Indentation Depth Interval Size Sensitivity Analysis .....	69
5.2.3.	Bridged Finite Element –Molecular Dynamics Displacement Analysis .....	70
5.3.	<i>Bridged Finite Element –Molecular Dynamics Simulation Conditions</i> .....	75
5.4.	<i>Bridged Finite Element –Molecular Dynamics Spherical Indenter Results</i> .....	77
5.5.	<i>Bridged Finite Element –Molecular Dynamics Berkovich Indenter Results</i> .....	78
5.6.	<i>Bridged Finite Element –Molecular Dynamics Square Pyramidal Indenter Results</i> ...	79
5.7.	<i>Discussion and Comparison between Bridged Finite Element –Molecular Dynamics Simulations and Molecular Dynamics Simulations</i> .....	80
5.8.	<i>Chapter Summary</i> .....	82
<b>6.</b>	<b>Concluding Remarks</b> .....	<b>83</b>

<b>References .....</b>	<b>85</b>
<b>Appendix A – Flowcharts for Molecular Dynamics Code .....</b>	<b>88</b>
<b>Appendix B – Flowcharts for Bridged FE-MD Code .....</b>	<b>91</b>

# Table of Figures

Figure 1 – Traditional and Partial Unit Cells for HCP metals.....	12
Figure 2 – Three Dimensional and Overhead Views of a Berkovich Indenter.....	15
Figure 3 – Three Dimensional View of Pyramidal Indenter.....	17
Figure 4 – Geometry of Spherical Indenter .....	21
Figure 5 – Two-dimensional representation of Periodic Boundary Conditions .....	24
Figure 6 – Saraev and Miller’s Hardness Graph for Copper Substrates with Nickel Coating .....	28
Figure 7 – Force vs. Indentation Depth Graph for 0 K to 600 K.....	30
Figure 8 – Force vs Indentation Depth Graph for 600 K to 1300 K.....	31
Figure 9 – Force Summation Method Comparison.....	32
Figure 10 – Comparison of Clamping Techniques using MD Simulations.....	39
Figure 11 – MD Results using Spherical Indenter with Diameter of Four Unit Cells.....	41
Figure 12 – MD Results using Spherical Indenter with Diameter of Five Unit Cells.....	41
Figure 13 – MD Results using Spherical Indenter with Diameter of Six Unit Cells.....	42
Figure 14 – MD Results using Spherical Indenter with Diameter of Seven Unit Cells .....	42
Figure 15 – MD Results using Berkovich Indenter with Base of Length Four Unit Cells.....	43
Figure 16 – MD Results using Berkovich Indenter with Base of Length Five Unit Cells .....	43
Figure 17 – MD Results using Berkovich Indenter with Base of Length Six Unit Cells.....	44
Figure 18 – MD Results using Berkovich Indenter with Base of Length Seven Unit Cells.....	44
Figure 19 – MD Results using Pyramidal Indenter with Base Four by Four Unit Cells.....	45
Figure 20 – MD Results using Pyramidal Indenter with Base Five by Five Unit Cells.....	45
Figure 21 – MD Results using Pyramidal Indenter with Base Six by Six Unit Cells.....	46
Figure 22 – MD Results using Pyramidal Indenter with Base Seven by Seven Unit Cells.....	46
Figure 23 – One dimensional two-node spring element .....	53
Figure 24 – Assembly of two-body potential linear spring elements .....	55
Figure 25 – Bridged FE-MD discretization scheme for element centred about Node 3.....	58
Figure 26 –Increment Size Sensitivity Test Applied Force Results .....	68
Figure 27 –Indentation depth Interval Size Sensitivity Analysis Applied Force Results.....	69
Figure 28 – FE-MD Results using Spherical Indenter with Diameter of Four Unit Cells.....	77

Figure 29 – FE-MD Results using Spherical Indenter with Diameter of Five Unit Cells .....	77
Figure 30 – FE-MD Results using Berkovich Indenter with Base of Length Four Unit Cells .....	78
Figure 31 – FE-MD Results using Berkovich Indenter with Base of Length Five Unit Cells .....	78
Figure 32 – FE-MD Results using Pyramidal Indenter with Base Four by Four Unit Cells .....	79
Figure 33 – FE-MD Results using Pyramidal Indenter with Base Five by Five Unit Cells .....	79
Figure 34 – Program Execution Flowchart for MATLAB™ function NI_MD_V_7_00 .....	88
Figure 35 – Program Execution Flowchart for MATLAB™ function Indenter_Design_V_7_00 .....	89
Figure 36 – Program Execution Flowchart for MATLAB™ function Contact_V_7_00 .....	90
Figure 37 – Program Execution Flowchart for MATLAB™ function NI_FE_MD_V_7_00 .....	91



# Table of Tables

---

Table 1 – Lattice Constants for Copper for Temperatures Ranging from 0K to 1300K .....	29
Table 2 – Atomic Positions at Equilibrium Step #1 from MD Simulations .....	34
Table 3 – Atomic Positions after Indentation Step from MD Simulations.....	35
Table 4 – Atomic Positions at Equilibrium Step #2 from MD Simulations .....	36
Table 5 – Atomic Positions at Equilibrium Step #3 from MD Simulations .....	37
Table 6 – Atomic Positions at Equilibrium Step #1 from Bridged FE-MD Simulations .....	71
Table 7 – Atomic Positions after Indentation Step from Bridged FE-MD Simulations.....	72
Table 8 – Atomic Positions at Equilibrium Step #2 from Bridged FE-MD Simulations .....	73
Table 9 – Atomic Positions at Equilibrium Step #3 from Bridged FE-MD Simulations .....	74

# Nomenclature

---

## English Symbols

$a$	:	Lattice constant of crystal
$a_{berk}$	:	Width of Berkovich indenter base
$a_i$	:	Indenter base, including the atomic radius
$a_{pyr}$	:	Width of square pyramidal indenter base
$a_x(t)$	:	Acceleration of atom in x-direction at time $t$
$A_c$	:	Contact area of indenter
$b$	:	Square Pyramid indenter edge length
$c$	:	Height of HCP unit cell
$C$	:	Compliance of a material
$d, \partial$	:	Differential Operators
$\hat{d}_{1x}$	:	Displacement of atom 1 in x-direction
$E_i$	:	Indenter elastic modulus
$E_r$	:	Reduced elastic modulus
$E_s$	:	Substrate elastic modulus
$\hat{f}_{1x}$	:	Internal force on atom 1 in the x-direction
$F_{1x}$	:	External force on atom 1 in the x-direction
$F_{pi}$	:	Force on atom $i$ in p-direction
$h$	:	Indenter height
$h_c$	:	Contact depth of indentation
$H$	:	Material hardness
$k$	:	Spring stiffness coefficient
$k_b$	:	Boltzmann's constant
$K$	:	Stiffness matrix
$KE_i$	:	Kinetic energy on atom $i$
$l$	:	Length of indenter base
$m$	:	Atomic mass
$N_{tot}$	:	Total number of atoms

$p_1$	:	Vertex #1 for planar indenter design
$r_{ij}$	:	Distance between atoms $i$ and $j$
$\vec{r}_1$	:	Cartesian vector on crystal lattice position #1
$r_{sphere}$	:	Radius of spherical indenter
$r_{atm}$	:	Atomic radius
$S$	:	Stiffness of a material
$t$	:	Time
$T$	:	Temperature
$U_1$	:	Potential energy of element 1
$U_i$	:	Potential energy of node $i$
$U_{ij}$	:	Potential energy of element between atoms $i$ and $j$
$U_{total}$	:	Total potential energy of an atomic system
$v_x(t)$	:	Velocity of atom in x-direction at time $t$
$x(t)$	:	X-coordinate at time $t$
$x_{cen}$	:	Centre of indenter in x-direction
$x_i$	:	Atomic position of atom $i$ in x-direction
$z_{berk}$	:	Distance in the y-direction from the base to tip for Berkovich indenter

### ***Greek Symbols***

$\phi_i$	:	Total potential energy of an atom $i$
$\phi_{ij}$	:	General pair potential between atoms $i$ and $j$
$\nu_i$	:	Poisson's ratio of indenter
$\nu_s$	:	Poisson's ratio of substrate
$\Delta$	:	Increment operator
$\Delta_1$	:	Change in atomic position of node 1
$\rho(r_{ij})$	:	Energy density on atom $i$ resulting from atom $j$
$\rho_{total,i}$	:	Total energy density of atom $i$
$\phi_{ij}$	:	EAM pair potential between atoms $i$ and $j$
$\Psi$	:	EAM embedding function
$\theta$	:	Indenter Angle

## List of Acronyms

---

AFM	:	Atomic Force Microscopy
BCC	:	Body Centred Cubic
DFT	:	Density Functional Theory
EAM	:	Embedding Atom Method
FE	:	Finite Elements
FE-MD	:	Finite Element – Molecular Dynamics
FEM	:	Finite Element Method
FCC	:	Face Centred Cubic
HCP	:	Hexagonal Close Packed
MD	:	Molecular Dynamics
PBC	:	Periodic Boundary Conditions

## List of Units

---

Å	:	Angstroms ( $10^{-10}$ m)
GPa	:	Giga Pascals ( $10^9$ Pa)
K	:	Degrees Kelvin
kg	:	kilograms
nN	:	Nano-newtons ( $10^{-9}$ N)
nm	:	Nanometres ( $10^{-9}$ m)
m	:	Metres
s	:	Seconds
°	:	Degrees

# 1. Introduction

---

Over the past two decades, nanomaterials have revolutionized the field of materials. Nanomaterials are defined by a grain size of less than 100 nm in at least one coordinate. Due to their miniscule grain size, nanomaterials have been found to exhibit superior properties to traditional materials. These superior properties include but are not limited to; improved hardness, higher tensile strength, longer fatigue life, and better ductility. Given these properties, nanomaterials are becoming increasingly popular for use in commercial products. Research studies focusing on nanomaterials have been successful in many areas including biomedicine,<sup>[1-2]</sup> gas detection,<sup>[3]</sup> and water disinfection.<sup>[4]</sup>

Due to the size of nanomaterials, it can often be challenging to analyze nanomaterials because at the nano-level, traditional experimentation often cannot be used. At the macro-level, tensile tests are mostly used in order to quantify the properties of a material. At the nano-level, however, it is very expensive and difficult to perform these tests. As a result, nanoindentation has become one of the most popular methods used to determine nanomaterial properties such as elastic modulus, hardness, and stiffness. A nanoindentation experiment is performed by using an indenter to push down on a material at a constant speed. Based on the response of the material to this indentation, one can establish the properties of the material. Most commonly, in nanoindentation experiments atomic force microscopy (AFM) is used to capture the response of the nanomaterial. These very high-resolution scanning probe microscopes provide excellent resolution even at a fraction of a nanometre and thus, can be used to accurately determine nanomaterial properties.<sup>[5]</sup>

Despite the success of nanoindentation experiments, the process is still tremendously expensive to perform. Therefore, many studies have been performed through the use of computational nanoindentation simulations. Using these computer simulations, users are able to drastically reduce costs and more efficiently investigate a wide array of materials at various conditions. The most common method used to perform computational atomistic simulations is using molecular dynamics (MD) simulations. As a result, MD nanoindentation simulations have become quite popular as a more cost effective replacement to nanoindentation experimentation. MD

nanoindentation simulations have been successful in studies dealing with a wide array of materials such as single walled carbon nanotubes,<sup>[6]</sup> poss materials,<sup>[7]</sup> silicon carbide,<sup>[8-9]</sup> and copper thin films.<sup>[10]</sup>

Prior to the commencement of this research, an Embedding Atom Method (EAM) numerical fitting procedure for pure face-centred cubic (FCC) and body centred cubic (BCC) metals was developed by Narayan, Behdinin and Fawaz. This research employs the EAM fitting procedure using a custom-made MD software to successfully predict the thermal and elastic properties of pure copper.<sup>[11]</sup> Chapter 2 discusses the theory of the custom-made MD software developed by Narayan, Behdinin and Fawaz and its application for this research to execute nanoindentation simulations. In Chapter 3, this code is tested and large scale nanoindentations are performed using three different types of indenters.

In recent years, multi-scale simulations have been implemented to improve the effectiveness of MD computer simulations. These multi-scale methods combine both nano-level and macro-level computational tools to more efficiently model the macro properties of large atomic systems. For example, a computational atomistic tool such as MD is employed to model the behaviour of atoms or molecules at the nano level, and that information is then used to determine the behaviour of the system at the micro and/or macro levels (often using the finite element method). These simulations have enabled users to greatly reduce computational time and resources while still obtaining accurate results. Recent research using multi-scale methods have explored such areas as material science,<sup>[12]</sup> robotics,<sup>[13]</sup> biomedicine,<sup>[14-16]</sup> fractional kinetics,<sup>[17]</sup> turbulence modeling,<sup>[18]</sup> and fluid dynamics.<sup>[19-20]</sup> Multi-scale simulations have also been applied to nanoindentation simulations as shown in research studying thin films,<sup>[21]</sup> cyclic indentation,<sup>[22]</sup> plasticity,<sup>[23]</sup> and nanocavities.<sup>[24]</sup>

A novel bridged finite element – molecular dynamics (FE-MD) method was previously developed by Narayan in his thesis entitled, “A numerical fitting procedure for the Embedding Atom Method Interatomic Potential and a Bridged Finite Element – Molecular Dynamics for Large Atomic Systems.” In this research, a unique discretization scheme was developed which enabled the atomic systems to be modeled more effectively. This bridged FE-MD method was

found to produce almost identical results to traditional MD simulations while requiring significantly less simulation time.<sup>[25]</sup> In this research, the bridged FE-MD code is adapted to perform nanoindentation simulations. Chapter 4 discusses the theory of the bridged FE-MD software, and the development of the FE-MD nanoindentation functions. In Chapter 5, the bridged FE-MD nanoindentation method is tested, full-scale simulations are performed, and the two simulation methods are then compared. Finally, Chapter 6 concludes the thesis as the results are reviewed and future applications are discussed.



## 2. Molecular Dynamics Nanoindentation

---

### 2.1. Molecular Dynamics Introduction

Even with today's technology, it is still not possible to directly observe the behaviour of individual atoms within a large system. Therefore, computer programs have been developed to simulate atomic behaviour at the molecular level. The most common method used for atomistic simulations is molecular dynamics (MD). Molecular dynamics is a time-dependent method that integrates equations of motion to simulate the behaviour of individual atoms for a variety of conditions.

The basic algorithm of all molecular dynamics simulations is as follows:

- 1) Define the initial parameters of the simulation.
- 2) Initialize the system (i.e. the initial positions and velocities of the atoms are defined).
- 3) Calculate the forces on all atoms.
- 4) Integrate the equations of motion to determine new positions and velocities for the time step.
- 5) Record the relevant information.
- 6) Repeat steps 3, 4 and 5 until all time steps are completed.

The force calculation is the most labour-extensive portion of most MD simulations. To calculate the force on a particular atom, one must consider the force contributions of all its neighbouring atoms. Therefore, one must first calculate the distance between all atoms. In most MD simulations, a cut-off distance is used. If distances between atoms are found to exceed the cut-off distance, the forces are assumed to be negligible. A general form of the force between two atoms in the x-direction is shown in Equations 1–2, where  $u$  is the potential energy between two atoms, and  $r$  is the distance between those two atoms.

$$f_x(r) = -\frac{\partial u(r)}{\partial x} \quad [1]$$

$$f_x(r) = -\left(\frac{x}{r}\right) \frac{\partial u(r)}{\partial r} \quad [2]$$

To more effectively model interatomic forces, several advanced methods have been developed. Today, most commonly used interatomic potential functions include the Lennard-Jones pair-potential, the Tersoff potential, and the Embedding Atom Method (EAM) potential functions. EAM potentials are the most effective method with which multi-body metallic systems are modeled.<sup>[26]</sup> In this research, only copper substrates were investigated. Since copper is a face centred cubic (FCC) metal, EAM potentials were used to model the interatomic interactions in this research. The theory of EAM potentials is discussed in greater depth in Section 2.2 below.

In order to integrate the equations of motion, a Taylor series expansion is implemented. The x-component of a particle's position in space at time  $t$  is referred to as  $x(t)$ , while its velocity and acceleration in the x-direction are referred to as  $v_x(t)$  and  $a_x(t)$ . Using the Verlet algorithm, one of the effective algorithms, the following second order Taylor expansions are defined in their simplified form.<sup>[27]</sup> Similarly, the equations can be applied in the same manner to the y and z-directions. The theory used in Equations 3 and 4 is the basis of the behaviour of the atoms throughout the code.

$$x(t + \Delta t) \approx x(t) + \Delta t \cdot v_x(t) + \frac{\Delta t^2}{2} a_x(t) \quad [3]$$

$$v_x(t + \Delta t) \approx v_x(t) + \frac{\Delta t}{2} [a_x(t) + a_x(t + \Delta t)] \quad [4]$$

## 2.2. Embedding Atom Method for Interatomic Potential

The Embedding Atom Method (EAM) is the most effective method for determining interatomic potentials for metallic systems. EAM potentials are based on the Density Functional Theory (DFT), a quantum mechanics theory derived from the understanding that the energy of a collection of atoms can be expressed by a functional of their density. In most interatomic

potential functions, the theory of pair potentials is used as the individual forces on atoms can be calculated and the total potential is simply the summation of the individual atomic forces. The total potential energy of an atom,  $\varphi_i$ , is a function of the distance between atoms  $i$  and  $j$ :

$$\varphi_i = \sum_{\substack{j=1 \\ j \neq i}}^N \varphi_{ij} \quad [5]$$

For metallic systems, the valence electrons are able to move freely throughout the system in what is often referred to as a ‘sea of electrons’. As a result, rather than calculating the atomic forces as pair potentials, one must also look at metallic systems as multi-body potentials. The EAM potential function, proposed by Daw and Baskes,<sup>[28-29]</sup> and later used by Foiles, Daw and Baskes to determine the functions for specific Face Centred Cubic (FCC) metals,<sup>[30]</sup> combines the pair potential and the multi-body potential to determine the total potential energy of the system. They proposed that the total potential energy of a metallic system is a sum of the embedding energy,  $\sum_i F_i(\rho_{h,i})$ , and half the pair potential energy. The embedding function, as shown in Equations 6 and 7, is a function of the electron density. In Equation 7, the variable  $\Psi$ , known as the embedding function, refers to the required energy to embed a nucleus within the sea of electrons.

$$U_{total} = \sum_i F_i(\rho_{total,i}) + \frac{1}{2} \sum_{\substack{i,j \\ i \neq j}} \phi(r_{ij}) \quad [6]$$

$$U_{total} = \sum_{i=1}^{N_{tot}} \left[ \frac{1}{2} \sum_{\substack{j=1 \\ j \neq i}}^N \phi(r_{ij}) + \Psi \left\{ \sum_{\substack{j=1 \\ j \neq i}}^N \rho(r_{ij}) \right\} \right] \quad [7]$$

## **2.3. Custom Molecular Dynamics Code NI\_MD\_V\_7\_00**

Prior to the commencement of this research, extensive work was performed in the areas of molecular dynamics and bridged finite elements. The research performed in this thesis is largely based the research of Narayan, Behdinan and Fawaz in which EAM potential functions were fitted using a custom-made MD simulation tool.<sup>[11]</sup> To enable nanoindentations to be performed using this research, the aforementioned software was significantly altered. Despite these alterations, the essentials of the MD portion of the code were unchanged. The code was programmed using MATLAB<sup>TM</sup>, a commonly used engineering programming tool due to its ability to easily carry out matrix operations. Sections 2.3.1–2.3.7 will outline significant alterations made to the original molecular dynamics code MD\_V\_7\_00 to enable MD nanoindentation simulations to be performed.

### **2.3.1. MATLAB<sup>TM</sup> Function NI\_MD\_V\_7\_00**

In order to perform the nanoindentation process, a number of functions were developed. NI\_MD\_V\_7\_00 is the principal function used to perform the nanoindentation procedure. As previously mentioned, this function is an adaptation of the previously designed function MD\_V\_7\_00. While NI\_MD\_V\_7\_00 is unable to perform the entire nanoindentation process itself, it is the function that ties all other functions together. The major tasks that are performed in this function are as follows:

- 1) Define the initial conditions for the simulation.
- 2) Read the positions and velocities of the substrate as defined in the function FCC\_Gen\_V\_7\_00, BCC\_Gen\_V\_7\_00, HCP\_Gen\_V\_7\_00 or Dia\_Gen\_V\_7\_00.
- 3) Calculate the distances between each atom and its neighbouring atoms and store the distances that are less than the defined EAM potential cut off distance. Use the EAM potential function, EAM\_potential\_V\_7\_00, to determine the interatomic forces.
- 4) Solve the forces on each of the atoms using the predetermined distances between each atom and its neighbouring atoms.

- 5) Calculate the current positions and velocities of each of the atoms using Taylor series expansions.
- 6) Apply the periodic boundary conditions to the system.
- 7) Write relevant information, such as positions, velocities and forces.
- 8) Repeat steps 2 through 7 for the required steps of the indentation procedure.

In Step 5, calculating the change in positions, velocity and acceleration for the next time step is discussed. During the indentation, for some atoms directly contacting the indenter, this step will not be included. Rather, for these atoms, their orientations will be dictated by contact analysis MATLAB<sup>TM</sup> function `Contact_V_7_00`. This algorithm will be discussed in detail in section 2.3.5.

Additionally, MATLAB<sup>TM</sup> function `NI_MD_V_7_00` simulates the multiple stages of a nanoindentation procedure. Prior to nanoindentation, the function executes time steps until the forces acting on the system are relatively unchanged and it is assumed that the system has reached equilibrium. At this point, the nanoindentation process is initiated. Once the indentation is complete, one can also simulate a relaxation period in which an equilibrium step is first implemented while the indenter position is unchanged, followed by a final equilibrium step following the removal of the indenter. This process will be implemented in section 3.2.3 when the author investigates the effects of nanoindentation on individual atomic positions. For a flowchart outlining the complete process that MATLAB<sup>TM</sup> function `NI_MD_V_7_00` executes, the reader can consult Appendix A.

### **2.3.2. MATLAB<sup>TM</sup> Crystal Structure Functions**

In the research conducted by Narayan, Behdinan and Fawaz, the geometries of Face Centred Cubic (FCC) and Body Centred Cubic (BCC) lattice crystals were defined.<sup>[11]</sup> FCC and BCC crystal structures are the crystal structures most commonly found in nature and their inclusion in the research is expected. However, typically, most nanoindentation experiments use diamond indenters because of their superior hardness and resistance to scratching. Furthermore, to make

the code more all encompassing, the geometry of Hexagonal Close Packed (HCP) metals was investigated. Therefore, two additional functions were created to define the crystal structures for diamond and HCP metals. The functions defining each of the crystal structures dealt with in this research will be defined in detail in future sections.

## **MATLAB<sup>TM</sup> Functions FCC\_Gen\_V\_7\_00, BCC\_Gen\_V\_7\_00**

Since the two functions defining the geometries of FCC and BCC metals were established in the work of Narayan, Behdinin and Fawaz, only a brief summary of the methodology used in constructing these functions will be discussed. The methodology for all crystal structure functions are as follows:

- 1) Define the initial positions of a partial cell (in the case of FCC and BCC metals, one-eighth cell was used).
- 2) Duplicate the partial cell in the x, y and z-directions for the defined number of cells.
- 3) Define the velocity of each of the atoms using a Gaussian probability density function.

For the initial velocities of the atoms that are defined in these functions, Equations 8–10 were used. In these equations,  $m$  is the mass of each atom,  $k_B$  is Boltzmann's constant, ( $1.3806503 \times 10^{-23} \text{ m}^2\text{kg/s}^2\text{K}$ ),  $T$  is the temperature of the system in degrees Kelvin, and  $N_{tot}$  is the total number of atoms in the system. Using these equations, the velocity distribution of the entire system is established. As one would expect, the kinetic energy of the system is largely dependent on the initial temperature of the system, as shown in Equations 11–13.

$$f(v_x) = \sqrt{\frac{m}{2\pi k_B T}} \exp\left(-\frac{mv_x^2}{2k_B T}\right) \quad [8]$$

$$f(v_y) = \sqrt{\frac{m}{2\pi k_B T}} \exp\left(-\frac{mv_y^2}{2k_B T}\right) \quad [9]$$

$$f(v_z) = \sqrt{\frac{m}{2\pi k_B T}} \exp\left(-\frac{mv_z^2}{2k_B T}\right) \quad [10]$$

$$\frac{1}{2} m_i (v_x^2)_{avg} = \frac{1}{2} m_i (v_y^2)_{avg} = \frac{1}{2} m_i (v_z^2)_{avg} = \frac{1}{2} k_B T \quad [11]$$

$$KE_{avg,i} = \frac{1}{2} m_i \left( (v_{x,i}^2)_{avg} + (v_{y,i}^2)_{avg} + (v_{z,i}^2)_{avg} \right) = \frac{3}{2} k_B T \quad [12]$$

$$KE_{total,i} = \sum_{i=1}^{N_{tot}} \frac{1}{2} m_i \left( (v_{x,i}^2)_{avg} + (v_{y,i}^2)_{avg} + (v_{z,i}^2)_{avg} \right) \triangleq \frac{3}{2} N_{tot} k_B T \quad [13]$$

### **MATLAB™ Function Dia\_Gen\_V\_7\_00**

For the indentations that were performed throughout this work, a diamond indenter was used to indent a copper substrate. The lattice structure for diamond is diamond cubic crystal. This crystal structure, while first discovered in diamond, has also been found to be present in several elements in group IV of the periodic table, such as tin, silicon, and germanium. To model the lattice structure of diamond cubic crystal materials, the function Dia\_Gen\_V\_7\_00 was developed. Diamond cubic crystal lattice structure, while similar to that of FCC metals, is slightly more complex. For FCC and BCC metals, the lattice parameters were defined by first investigating a one-eighth cell. However, in a diamond cubic crystal structure, since the pattern that occurs in a one-eighth cell is not repeated throughout the lattice structure, a partial unit cell was determined to be a more effective option. The atoms that characterize this partial cell are defined in Cartesian coordinates as shown in Equation 14.

$$\begin{aligned}
\vec{r}_1 &= [0 \quad 0 \quad 0] \\
\vec{r}_2 &= [0 \quad \frac{a}{2} \quad \frac{a}{2}] \\
\vec{r}_3 &= [\frac{a}{2} \quad 0 \quad \frac{a}{2}] \\
\vec{r}_4 &= [\frac{a}{2} \quad \frac{a}{2} \quad 0] \\
\vec{r}_5 &= [\frac{a}{4} \quad \frac{a}{4} \quad \frac{a}{4}] \\
\vec{r}_6 &= [\frac{3a}{4} \quad \frac{3a}{4} \quad \frac{a}{4}] \\
\vec{r}_7 &= [\frac{a}{4} \quad \frac{3a}{4} \quad \frac{3a}{4}] \\
\vec{r}_8 &= [\frac{3a}{4} \quad \frac{a}{4} \quad \frac{3a}{4}]
\end{aligned} \tag{14}$$

In the Equation 14, ‘ $a$ ’ is the defined lattice size of the diamond unit cell. Using this partial cell, the diamond cubic lattice structure for the entire material can then be defined by replicating the eight atoms defined above and translating them as necessary. These atomic positions are then stored in a text file to be accessed in future functions.

To determine the velocities of atoms using the diamond cubic crystal structure, the process is identical to the one described above in section 2.3.2.1. For the nanoindentations performed in this thesis, the indenter was assumed to be rigid. As a result, the atoms within the diamond indenter were assumed to have no velocity or kinetic energy. A detailed explanation of this decision is discussed in Section 3.1.

## **MATLAB<sup>TM</sup> Function HCP\_Gen\_V\_7\_00**

The last crystal structure dealt with in this research is Hexagonal Close Packed (HCP). The HCP lattice structure is observed in metals such as zinc, titanium, beryllium, and magnesium. As in the function Dia\_Gen\_V\_7\_00, a one-eighth unit cell could not be used to capture the nature of



the lattice structure for HCP metals. Since FCC and BCC lattice structures are cubic, the atoms can be modelled with respect to the x, y and z axes. However, the HCP unit cell is hexagonal in shape, hence the atoms are not easily defined by their relationships to the x, y and z axes. To overcome this problem, the unit cell for the HCP crystal structure was defined slightly differently. In HCP metals, the crystal structure is similar to that of FCC metals, but there is a shift in the positions of the atoms in each layer. Consequently, to orient the unit cell along the x, y and z axes, rather than looking at the entire unit cell, a smaller unit cell was constructed by creating a rectangular prism out of five atoms in the bottom row of the unit cell and the original height. A traditional HCP unit cell and the modified HCP unit cell are displayed in Figure 1.

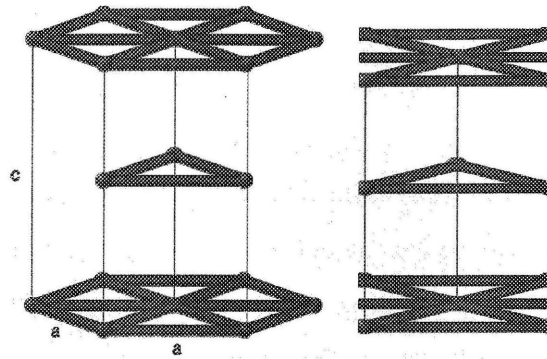


Figure 1 – Traditional and Partial Unit Cells for HCP metals

The length of the modified unit cell is solved using the Pythagorean Theorem as shown in Equation 15. The distance between atoms in the y-direction in successive layers of atoms is calculated in Equations 16.

$$l = 2 \sqrt{a^2 - \left(\frac{a}{2}\right)^2} = \sqrt{3}a \quad [15]$$

$$\Delta y = \sqrt{a^2 - \left(\frac{c}{2}\right)^2} \quad [16]$$

Using the modified unit cell, a design process using a modified one-eighth cell was implemented, the positions of which are shown in Equation 17.

$$\begin{aligned}
\vec{r}_1 &= [0 \quad 0 \quad 0] \\
\vec{r}_2 &= \left[ \frac{a}{2} \quad \frac{l}{2} \quad 0 \right] \\
\vec{r}_3 &= \left[ 0 \quad -\Delta y \quad \frac{c}{2} \right] \\
\vec{r}_4 &= \left[ \frac{a}{2} \quad \frac{l}{2} - \Delta y \quad \frac{c}{2} \right]
\end{aligned} \tag{17}$$

While creating a function to characterize the crystal structure of HCP metals will allow the software to become more all-encompassing, in this research, simulations will not be performed using this crystal structure. For HCP metals, the potential functions are much more complex and therefore, significant work must be done to enable their implementation in future research.

### 2.3.3. MATLAB<sup>TM</sup> Function Indenter\_Design\_V\_7\_00

To perform nanoindentation simulations successfully, the construction of indenter is paramount. MATLAB<sup>TM</sup> function Indenter\_Design\_V\_7\_00 deals with all matters concerning the construction and design of the indenter. The first concern with respect to the indenter design is the material chosen for the indenter. Based on the nature of the function, indenters may be constructed using any of the four crystal structures defined above in Section 2.3.2.

Once the lattice structure for the indenter is established using one of the crystal structure definitions discussed in Section 2.3.2, the indenter shape could then be characterized. While there are several different types of nanoindenters commonly used today, three of the more popular nanoindenters used today are the spherical, Berkovich, and square pyramid indenters. In this research, these three indenters are investigated. The geometry of the three indenters is defined by the following three functions; Spher\_Geo\_V\_7\_00, Berk\_Geo\_V\_7\_00, and Pyr\_Geo\_V\_7\_00. These functions sort through each atom individually to determine whether it fits within the geometry of that specific indenter. In the research of Narayan, Behdinin and Fawaz, MATLAB<sup>TM</sup> function CRACK\_VOID\_V\_7\_00 was used to create a residual function using symbolic MATLAB<sup>TM</sup> to remove atoms present in a specific geometry from the system. In

this function, the same theory is used to remove those atoms that are outside the specified geometry from the system. The functions designed for each of the indenter types are discussed future sections of this chapter. A flowchart illustrating the indenter design process can be found Appendix A.

### **MATLAB™ Function Spher\_Geo\_V\_7\_00**

The first indenter geometry defined was the spherical indenter using MATLAB™ function Spher\_Geo\_V\_7\_00. The geometry of the spherical indenter is defined in Equation 18. The subscript ‘*cen*’ refers to the position of the centre of the sphere in the specified direction, while the variable *i* refers to the atom being investigated.

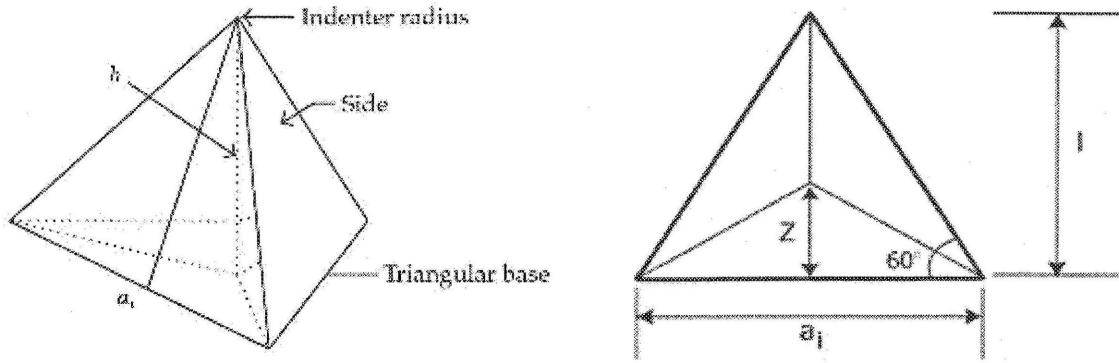
$$r_{sphere} = \sqrt{(x_{cen} - x_i)^2 + (y_{cen} - y_i)^2 + (z_{cen} - z_i)^2} \quad [18]$$

Using the geometry of the sphere, the function evaluates the position of each individual atom to determine whether it lies within the indenter. For large indenters, the contact analysis during nanoindentation is very time consuming process. As a result, to speed up the simulation, both the top half of the sphere and atoms in the centre of the sphere are also removed from the indenter. These atoms cannot contact the substrate and therefore, their removal does not negatively impact the simulation results.

### **MATLAB™ Function Berk\_Geo\_V\_7\_00**

Berkovich indenters are the most common indenters used in modern nanoindentation experiments and simulations. A Berkovich indenter is a triangular pyramid in which the angle between the edge and the normal is 65.3 degrees. In Figure 2, diagrams of a Berkovich indenter in three-dimensional and overhead views are displayed. In contrast to the process used to construct the spherical indenter, the approach for the Berkovich indenter was altered slightly due to the complexity of its geometry. In this case, it is not possible to define one equation that will

characterize the entire geometry of the indenter as was done for the spherical indenter. Instead, MATLAB<sup>TM</sup> function Berk\_Geo\_V\_7\_00 characterizes the geometry of Berkovich indenters by defining three planar equations.



To determine the planar equations for each of the three faces of the indenter, the positions of each of the vertices were first identified. Using the overhead view of the Berkovich indenter as shown above in Figure 2<sup>1</sup>, the geometry of a Berkovich indenter was established. In Figure 2,  $a_i$  is the width of the base,  $l$  is the length of the base,  $z$  is the distance in the y-direction from the base to the indenter tip, and  $h$  is the height of the indenter. These variable definitions are shown in Equations 19–21, below. Next, the four vertices for the Berkovich indenter were determined and are shown in Equation 22 below. Finally, the equations of a plane were developed using Equations 23–27 as illustrated below.

$$l = \frac{\sqrt{3}}{2a_i} \quad [19]$$

$$z_{berk} = \frac{a_i}{2} \tan 30^\circ \quad [20]$$

$$h = \frac{a_i \cos(65.3^\circ)}{2 \sin(65.3^\circ)} = \frac{a_i}{2 \tan(65.3^\circ)} \quad [21]$$

<sup>1</sup> <http://www.hindawi.com/floats/742569/figures/742569.fig1.xht>

$$\begin{aligned}
p_1 &= [0 \quad 0 \quad 0] \\
p_2 &= [a \quad 0 \quad 0] \\
p_3 &= \left[ \frac{a_i}{2} \quad l \quad 0 \right] \\
p_4 &= \left[ \frac{a_i}{2} \quad z_{berk} \quad h \right]
\end{aligned} \tag{22}$$

$$Ax + By + Cz + D = 0 \tag{23}$$

$$A = y_1(z_2 - z_3) + y_2(z_3 - z_1) + y_3(z_1 - z_2) \tag{24}$$

$$B = z_1(x_2 - x_3) + z_2(x_3 - x_1) + z_3(x_1 - x_2) \tag{25}$$

$$C = x_1(y_2 - y_3) + x_2(y_3 - y_1) + x_3(y_1 - y_2) \tag{26}$$

$$D = -[x_1(y_2z_3 - y_3z_2) + x_2(y_3z_1 - y_1z_3) + x_3(y_1z_2 - y_2z_1)] \tag{27}$$

Using the three plane equations described above, the atoms are separated based on their locations relative to the boundaries of the three planes. If an atom was found to be outside the boundaries of the three planes it was removed from the system. As shown the above equations, the indenter tip was designed to be facing in the positive z-direction. In order to perform the indentation, the indenter is reflected so that the tip is facing towards the substrate. The manipulation of the indenter is discussed in more detail in Section 2.3.4 below.

### **MATLAB<sup>TM</sup> Function Pyr\_Geo\_V\_7\_00**

The third type of indenter used in this research is a square pyramidal indenter. A diagram of a square pyramidal indenter is shown in Figure 3. The approach used by MATLAB<sup>TM</sup> function Pyr\_Geo\_V\_7\_00 was quite similar to that of MATLAB<sup>TM</sup> function Berk\_Geo\_V\_7\_00. Once again, a planar approach was used to define the indenter faces and any atoms outside the planar boundaries were determined not to be part of the indenter.

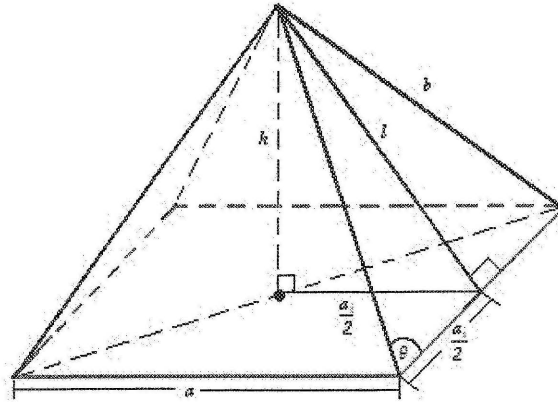


Figure 3 – Three Dimensional View of Pyramidal Indenter

In Equations 28–30, the geometry of the square pyramid indenter is studied. As shown in Figure 3<sup>2</sup>,  $a_i$  refers to the width of the indenter base,  $b$  refers to the length of the edge of the pyramid,  $l$  references the height along the face of the pyramid, and  $h$  refers to pyramid height. It is important to note that ' $\theta$ ', the angle of the pyramid face, must be larger than  $45^\circ$ . As shown in Equation 30, if an angle of less than  $45^\circ$  is used, the height of the pyramid will be an imaginary number. In this research, the value of  $\theta$  used for the square pyramid design was  $54.74^\circ$ . As can be calculated using Equation 30, this angle produces an indenter with height equal to the width of its base. Using the geometry of the square pyramid, the five vertices present on the indenter were defined as shown in Equation 31. The planar equations, as shown above in Equations 23–27, were then used in order to characterize the planes needed to construct the square pyramid.

$$b = \frac{a_i}{2 \cos \theta} \quad [28]$$

$$l = \sqrt{b^2 - \left(\frac{a_i}{2}\right)^2} = \frac{a_i}{2} \sqrt{\frac{1}{\cos^2 \theta} - 1} \quad [29]$$

$$h = \sqrt{l^2 - \left(\frac{a_i}{2}\right)^2} = \frac{a_i}{2} \sqrt{\frac{1}{\cos^2 \theta} - 2} \quad [30]$$

$$p_1 = [0 \quad 0 \quad 0] \quad [31]$$

<sup>2</sup> [http://www.analyzemath.com/Geometry\\_calculators/pyramid\\_1.gif](http://www.analyzemath.com/Geometry_calculators/pyramid_1.gif)

$$p_2 = [a_i \quad 0 \quad 0]$$

$$p_3 = [a_i \quad a_i \quad 0]$$

$$p_4 = [0 \quad a_i \quad 0]$$

$$p_5 = \left[ \frac{a_i}{2} \quad \frac{a_i}{2} \quad h \right]$$

Again, as was done in MATLAB<sup>TM</sup> function Berk\_Geo\_V\_7\_00, the indenter was designed with its tip in the positive z-direction. As a result, this indenter must be reflected so that its tip will be facing toward the substrate.

#### 2.3.4. MATLAB<sup>TM</sup> Function Indenter\_Move\_V\_7\_00

In order to orient the indenter as required, MATLAB<sup>TM</sup> Function Indenter\_Move\_V\_7\_00 is used. This function is very flexible as it allows the user to both move and reflect the indenter to any position or orientation in space. When Berkovich or square pyramidal indenters are used, the indenter must first be reflected. To do so, the centre of the indenter is first determined and for each atom, a new z-position is calculated that is of equal distance from the centre of the indenter but on the opposite side of the centre. As well, this function is used to orient the indenter so that the centre of the indenter is directly above the centre of the substrate and that the nanoindentation will be performed more effectively. Furthermore, when the indentation is taking place, it is used to lower the indenter into the substrate at the beginning of each time step.

#### 2.3.5. MATLAB<sup>TM</sup> Function Contact\_V\_7\_00

The MATLAB<sup>TM</sup> function Contact\_V\_7\_00 is the most important function for the indentation procedure. This function determines if contact occurs between the indenter and the substrate and any substrate atoms being contacted are displaced accordingly. To determine whether contact is occurring, MATLAB<sup>TM</sup> function Contact\_V\_7\_00 looks at each indenter atom and calculates the distance between it and each individual substrate atom. If the calculated distance is less than the

radii of the indenter and substrate atoms, the two atoms are determined to be in contact with one another.

In order to deal with the contact, MATLAB<sup>TM</sup> function `Contact_V_7_00` separates all possible contact into three different scenarios. Each of these scenarios is dealt with differently and will be discussed in detail. The three cases in which contact may occur are as follows:

- 1) Contact between one substrate atom and one indenter atom.
- 2) Contact between one substrate atom and two indenter atoms.
- 3) Contact between one substrate atom and more than two indenter atoms.

When dealing with contact, an iterative procedure was implemented using the bisection method. The bisection method is a technique that iterates by cutting the bracketing interval in half. For every iteration, the midpoint is tested and if the required value is between the midpoint and the lower bound, the midpoint becomes the new upper bound. Conversely, if the required value is between the midpoint and the upper bound, the midpoint becomes the new lower bound. This process is repeated until the value is within the required range for the iteration. This method is often criticized because numerous iterations are required for convergence. In this study, the bisection method is an excellent choice due to its simplicity and tremendous effectiveness.

When only a substrate atom is being contacted by only one indenter atom, dealing with the contact is quite simple. Here, the function first determines the angle of contact and compares it to a pre-specified angle. If the angle of contact is less than this specified angle, the substrate atom will be displaced only in the z-direction. Otherwise, the substrate atom is displaced radially as would be expected for any inelastic contact. For this radial displacement, an iterative procedure is implemented in which a displacement factor is varied until the distance between the atoms is within the required limits. The decision to only displace contacted atoms radially for atoms contacted at an angle was made to ensure that the simulation would be more realistic. If all contact resulted in radial displacement, the bonds between the substrate atoms that are directly contacting the indenter would eventually break as the atoms would be pushed apart as the



indenter is lowered. While it may be expected that this would occur during high speed indentations, it is not expected for the indentation speeds analyzed in this research.

For the second case, the algorithm is substantially more complex. This time, an iterative procedure is used to ensure that both indenter atoms are equidistant from the substrate atom as contact is occurring. Throughout the iterative process, the displacement of the substrate atom is calculated and stored in a temporary variable. If the distance between the substrate atom and an indenter atom is less the contact distance, the substrate atom will be moved further from the indenter atom being examined. Similarly, if the distance between the substrate atom and an indenter atom exceeds the contact distance, the bounds of the bisection method will be altered to ensure that the substrate atom will be moved closer to the indenter atom in question. Using the new position of the substrate of the atom, the process is repeated until the distances between the substrate atom and each of the two indenter atoms are within the specified range. Using this technique, atoms may be moved significantly to ensure that the substrate atom would be equidistant from the two indenter atoms. Using this method, dislocations were found to ensue as atoms were displaced much faster than the indentation speed. Consequently, the displacement of the substrate atom was limited so that it cannot exceed the indentation speed.

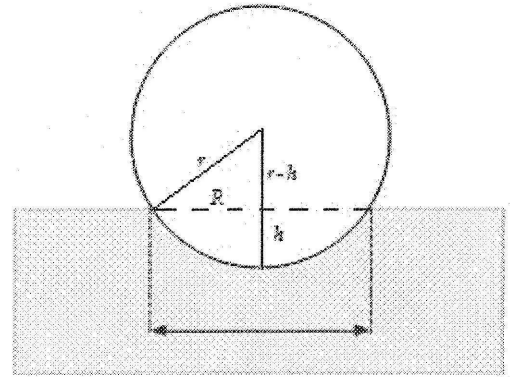
When an atom is contacting more than two indenter atoms at once, the algorithm used to deal with contact is quite straight-forward. It is assumed that when contacting three or more atoms, the substrate atom will be locked in place between the indenter atoms. Therefore, the only displacement that will occur while the indenter is moving in the z-direction will be in the same direction. Thus, MATLAB<sup>TM</sup> function `Contact_V_7_00` merely displaces a substrate atom by the indentation speed in the direction of contact if it is in direct contact with three or more indenter atoms. A flowchart describing the entire procedure undergone by MATLAB<sup>TM</sup> function `Contact_V_7_00` can be found in Appendix A.

### 2.3.6. MATLAB™ Function Contact\_Area\_V\_7\_00

The process of determining the hardness of a material through a nanoindentation simulation requires the identification of both the indenter's contact area and the force applied by the indenter on the substrate. MATLAB™ function Contact\_Area\_V\_7\_00 was created to determine the contact area at any point during the indentation procedure. Contact area is defined as a two-dimensional projection of the area of the substrate that is being contacted by the indenter. The contact function, as described above, takes into account the radii of the diamond and copper atoms. When calculating the contact area, the radii of the diamond atoms are accounted for as well. Since the contact area is dependent on the geometry of the indenter, the calculations for the three types of indenters will be discussed independently in the sections below.

#### Spherical Indenter Contact Area

To calculate the contact area for the spherical indenter, the geometry of the sphere must first be understood. As can be shown in Figure 4,<sup>3</sup> the radius of the sphere and the radius of the cross-sectional area form a right angle. If the indentation depth is known, the Pythagorean Theorem can be used to determine the cross-sectional radius which can then be used to solve the area of the cross-section. In the calculations, the radius of the diamond atoms is also taken into account. The process of calculating the contact area is shown below in Equations 32–34.



of Spherical Indenter

$$r = r_{\text{sphere}} + r_{\text{atm}} \quad [32]$$

$$R = \sqrt{r^2 - (r - h)^2} = \sqrt{h^2 - 2rh} \quad [33]$$

$$A_c = \pi R^2 = \pi(h^2 - 2rh) \quad [34]$$

<sup>3</sup> <http://www.csm-instruments.com/fr/system/files/images/ab-25-photo7.jpg>

## Berkovich Indenter Contact Area

For a Berkovich indenter, by using Equations 19–21 as defined above in Section 2.3.3.2, the two-dimensional projection of the contact surface can be solved as well. Since Berkovich indenters are triangular in nature, by using the equation for the area of a triangle as shown in Equation 37, an expression for the contact area of the indenter was developed.

$$a_i = a_{berk} + 2r_{atm} \quad [35]$$

$$l = l_{berk} + 2r_{atm} \quad [36]$$

$$A_c = \frac{a_i l}{2} = \frac{\sqrt{3}a_i^2}{4} \quad [37]$$

$$h = \frac{a_i}{2 \tan(65.3^\circ)} \quad [38]$$

$$A_c = 3\sqrt{3}h^2 \tan^2(65.3^\circ) = 24.56197h^2 \quad [39]$$

Equation 39 is commonly found in literature with respect to Berkovich indenters. It should be noted that this equation is only valid before the indenter is completely submerged within the substrate. If the indenter is lowered further into the substrate, the indentation depth used in the calculation shown in Equation 39 will be that of the indenter height, rather than the actual indentation depth. In all simulations used in this research, indentations will be performed ensuring that the indenter is never completely submerged within the substrate.

## Pyramid Indenter Contact Area

The contact area calculations for the square-pyramidal indenter are similar to that of the Berkovich indenter. The methodology used to calculate the contact area for a square pyramid indenter can be shown in Equations 40–44.

$$a_i = a_{pyr} + 2r_{atm} \quad [40]$$

$$A_c = a_i^2 \quad [41]$$

$$h = \frac{a_i}{2} \sqrt{\frac{1}{\cos^2 \theta} - 2} \quad [42]$$

$$a_i = \frac{2h}{\sqrt{\sec^2 \theta - 2}} \quad [43]$$

$$A_c = \frac{4h^2}{\sec^2 \theta - 2} \quad [44]$$

As can be shown in Equation 44, the contact area for a square pyramidal indenter is only dependent on the angle and height of the pyramid. However, as is the case with the other two indenters, this equation is only valid so long as the entire tip of the indenter is not submerged within the substrate.

### 2.3.7. MATLAB™ Function Force\_Summation\_V\_7\_00

To determine the forces acting on the system during nanoindentation, MATLAB™ function Force\_Summation\_V\_7\_00 was created. Since the above mentioned MATLAB™ function Contact\_V\_7\_00 is displacement dependent, the force acting on the entire substrate is always preserved. As a result, alternative techniques to determine the applied force on the system during nanoindentation were developed. In Section 3.2.2 of this thesis, a thorough comparison of the differing force techniques is performed and the optimal force summation technique for the purposes of this research is determined.

## 2.4. Molecular Dynamics Periodic Boundary Conditions

Despite the high-performance computing that is available today, it remains a challenge to simulate the large atomic systems that would be required to obtain the properties of a nanomaterial as would be usable for commercial manufacturing, as billions or even trillions of atoms must be simulated over long time periods. To enable such large simulations to be performed, a commonly used technique in MD simulations uses Periodic Boundary Conditions (PBC). Periodic Boundary Conditions enable the user to generate a few unit cells, and replicate

them as needed to simulate the remaining atoms in the system. The behaviour of the original unit cells (usually known as the primary cells) is analyzed and the exact positions and orientations are reproduced for all the replica cells (usually known as image cells). A two-dimensional representation of the use of PBC cells is shown in Figure 5<sup>4</sup> below.

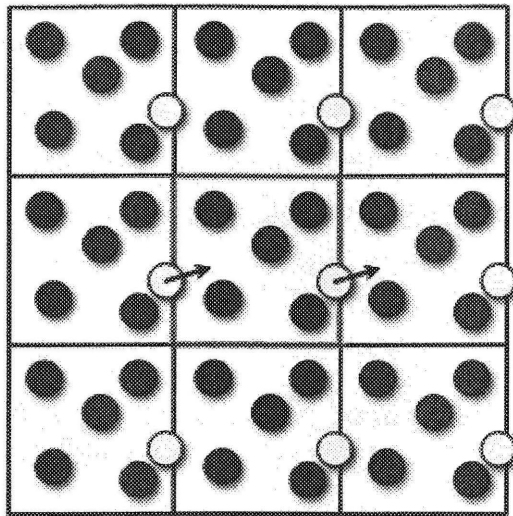


Figure 5 – Two-dimensional representation of Periodic Boundary Conditions

In this research, PBC atoms are treated slightly differently than described above. Rather than duplicating the positions and velocities of each of the atoms of the primary cell for the image cells, all atoms in the image cells are treated as fixed atoms. As such, throughout the simulation, their positions remain fixed. For nanoindentation, this decision is greatly beneficial as one may use PBC without having to replicate the influence of the indenter for each of the image cells. Additionally, for those atoms that are further from the centre of the substrate, the applied forces as a result of the indentation are quickly diminished. Therefore, modeling PBC atoms as rigid still ensures that the results obtained from the nanoindentation will be accurate.

<sup>4</sup> [http://matdl.org/repository/eserv/matdl:857/web\\_wiki2fez2465.jpg](http://matdl.org/repository/eserv/matdl:857/web_wiki2fez2465.jpg)

## 2.5. Additional Nanoindentation Theory

One of the most common purposes of performing a nanoindentation test is to determine certain properties of the material. For example, the equations commonly used to determine properties such as hardness, reduced elastic modulus, stiffness, compliance, projected area and contact depth are calculated using Equations 45–50. The hardness of a material is a measure of the pressure it is able to withstand prior to deformation. The reduced elastic modulus is a measure of a materials tendency to be deformed elastically. The stiffness of a material is the resistance of a material to elastic deformation. The difference between the reduced elastic modulus and stiffness of a material is that the former is an intrinsic property of a material while the latter is dependent on the shape and boundary conditions of that material. Compliance is simply the inverse of stiffness and is helpful to know when performing instrument calibration at the beginning of a nanoindentation experiment. Finally the projected area and contact depth are both properties of the experiment and therefore, will often be defined by the user.

$$\text{Hardness} \quad H = \frac{F}{A_{hc}} \quad [45]$$

$$\text{Reduced Elastic Modulus} \quad E_r = \frac{\sqrt{\pi}}{2} \frac{S}{\sqrt{A_{hc}}} \quad [46]$$

$$\text{Stiffness} \quad S = \left( \frac{dF}{dh} \right)_{max} = \frac{2}{\sqrt{\pi}} E_r \sqrt{A_c} \quad [47]$$

$$\text{Compliance} \quad C = \frac{1}{S} \quad [48]$$

$$\text{Contact Area} \quad A_c = \frac{\pi}{4E_r^2 C^2} \quad [49]$$

$$\text{Contact Depth} \quad h_c = h - 0.75 \frac{F_{max}}{S} \quad [50]$$

The above six equations are the basis for all nanoindentation equations.<sup>[31]</sup> Using Hertzian theory, the reduced elastic modulus,  $E_r$ , can also be studied due to its relationship with the elastic modulus,  $E$ , and Poisson's ratio,  $\nu$ , of the substrate and indenter. In cases where the indenter is

assumed to have infinite stiffness, the Young's Modulus,  $E_i$ , will equal infinity and therefore the second part of the equation will be cancelled out.<sup>[32]</sup>

$$E_r = \left( \frac{1 - \nu_s^2}{E_s} - \frac{1 - \nu_i^2}{E_i} \right)^{-1} \quad [51]$$

In this study, the major focus will be to determine the hardness of the substrates investigated. Therefore, Equation 45 will be used almost exclusively. However, in further research, Equations 46–51 will be very valuable as they may be used to investigate further nanomaterial properties.

### 3. Molecular Dynamics Nanoindentation Results

---

#### 3.1. Molecular Dynamics Nanoindentation Introduction

For each the simulations that are analyzed in this chapter, a copper substrate was indented using a diamond tip indenter. The copper substrate has been modelled using the EAM interatomic potential as discussed in Chapter 2. Traditionally, diamond indenters have been used in nanoindentation, due to their hardness and resistance to scratching. Using computer simulations, one can theoretically define an indenter with infinite stiffness. It has been found that modelling an indenter using infinite stiffness is in fact more accurate than a diamond indenter. It should be noted that using rigid indenters, rather than modelling the interatomic interactions, produces results that differ slightly from experimental results.<sup>[33]</sup> This difference originates from the deformation of the diamond indenter, while miniscule, during experimental nanoindentations. For theoretical nanoindentations using indenters with infinite stiffness, such a deformation does not occur and therefore the results are more accurate. Due to the improved accuracy of the hardness values when the indenter is modeled as rigid, the interatomic interactions between the diamond atoms as well as between the copper and diamond atoms are not modeled in this research. However, the lattice structure of the diamond atoms is still utilized in the simulations.

Prior to performing nanoindentations, the research of Saraev and Miller was reviewed.<sup>[34]</sup> In their work, the impact of nickel coating on the hardness of copper substrates is investigated. Saraev and Miller employ the parallel molecular dynamics program PARADYN to perform their nanoindentation simulations. In their simulations, a 40 by 40 by 30 unit cell (approximately 14 by 14 by 11 nm) copper substrates with differing thicknesses of nickel coating are indented using a diamond spherical indenter with diameter 17 unit cells (approximately 6 nm) at a speed of 5 m/s and at a temperature of 293 K. EAM potential functions are used to obtain the interatomic potentials as are used in this author's research. The hardness vs. indentation depth graph for copper and nickel-coated copper substrates is displayed in Figure 6. As shown in Figure 6, the baseline simulation in their research is a pure copper substrate as is studied in this thesis. In that case, the maximum recorded hardness value during nanoindentation is approximately 16 GPa. As



a result, in this research study, the results obtained can be compared with Saraev and Miller's results to ensure that the results are accurate.

Due to the time constraints and computational limitations, simulations using similar size substrates as used by Saraev and Miller were not possible in this research. As a result, Saraev and Miller's results will be used as a guide this research to ensure that the results obtained by the bridged FE-MD MATLAB<sup>TM</sup> simulations are within reason.

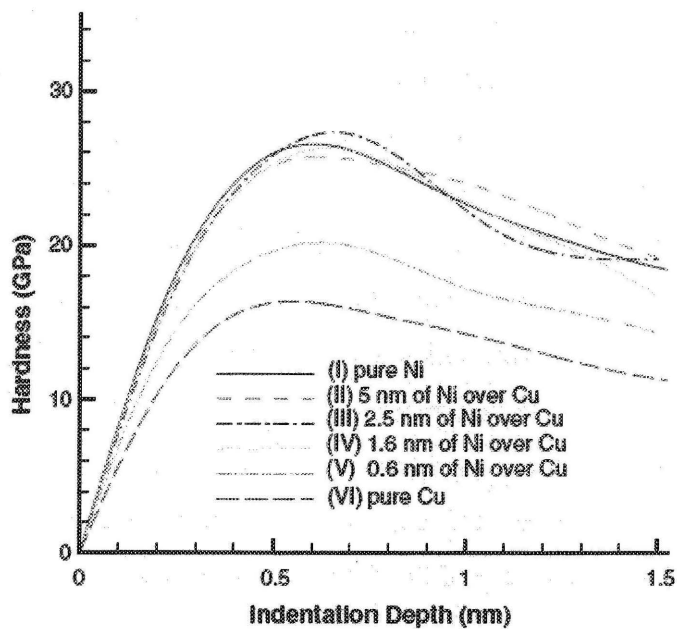


Figure 6 – Saraev and Miller's Hardness Graph for Copper Substrates with Nickel Coating

### 3.2. Molecular Dynamics Sensitivity Analysis

Prior to running full-scale simulations, several sensitivity analyses were performed to provide insight into the responsiveness of the nanoindentation software. Additionally, through these tests, one will be able to better understand the effect of certain changes to the initial conditions of the simulation on the hardness of a material. Four sensitivity analyses were performed prior to running full simulations. These tests will be discussed in Sections 3.2.1–3.2.4 below.

### 3.2.1. Temperature Sensitivity Analysis

The first sensitivity analysis performed investigated the effect of temperature changes on nanoindentation results. By varying temperature while maintaining the initial conditions of the simulation, the effects of temperature on nanohardness could be ascertained. As discussed by Narayan, Behdinan and Fawaz, lattice constants are greatly dependent on temperature. In this research, a set of lattice constants were solved using the custom MATLAB<sup>TM</sup> MD software and were compared to experimental results. The lattice constants, when verified via this MD software, were found to be almost identical to those of the experimental results.<sup>[11]</sup> In this work, the lattice constants of the experimental analysis were used for the temperature analysis and can be found in Table 1.

Temperature (K)	Lattice Constants (Å) <sup>[35]</sup>
0 K	3.603287
25 K	3.603287
50 K	3.603504
100 K	3.604806
200 K	3.609650
293 K	3.615000
400 K	3.621579
500 K	3.628086
600 K	3.634846
700 K	3.641787
800 K	3.648945
900 K	3.656464
1000 K	3.664381
1200 K	3.681444
1300 K	3.690734

Table 1 – Lattice Constants for Copper for Temperatures Ranging from 0K to 1300K

Using the above lattice constants, the effect of temperature on the hardness of copper was analyzed. For the simulations, the following initial conditions were used:

Indenter Type	Spherical
Substrate Size	4 by 4 by 4 unit cells (14.46 Å), PBCs in x, y- directions
Indenter Diameter	2 unit cells (7.134 Å)
Indent Speed	25 m/s
Maximum Indentation Depth	5Å
Clamping	Absolute clamping on all faces except the in the positive z-direction

The simulations were performed using given temperatures and corresponding lattice constants shown in Table 1. A graph that compares the applied force to the indentation depth is shown in Figures 7 and 8.

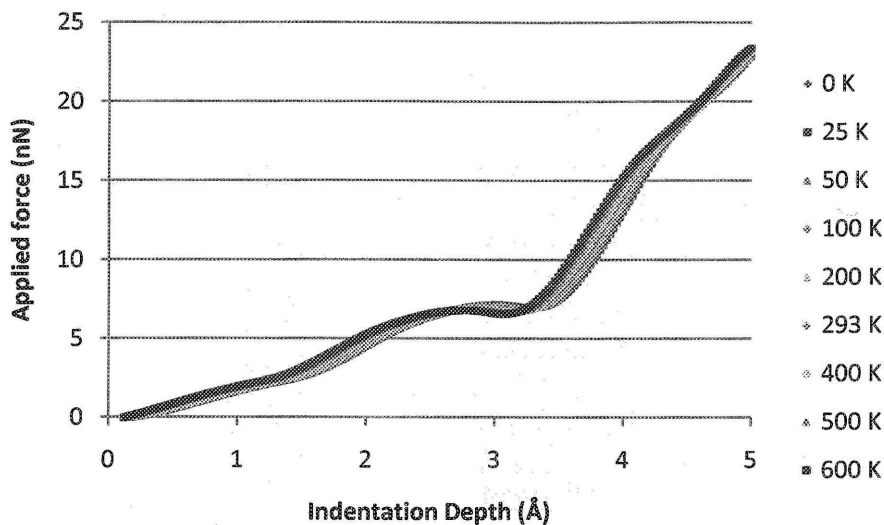


Figure 7 – Force vs. Indentation Depth Graph for 0 K to 600 K

As shown in Figure 7, for temperatures ranging from 0 K to 600 K, changes in temperature and lattice size do not significantly affect the forces acting on the substrate and subsequently do not significantly affect the hardness of the material. It is noticed, however, that as the temperature increases, the forces are slightly increased during indentation. While these differences are quite marginal for a small scale simulation such as was performed in this temperature analysis, it would be expected that the differences would be slightly larger for simulations using large atomic systems.

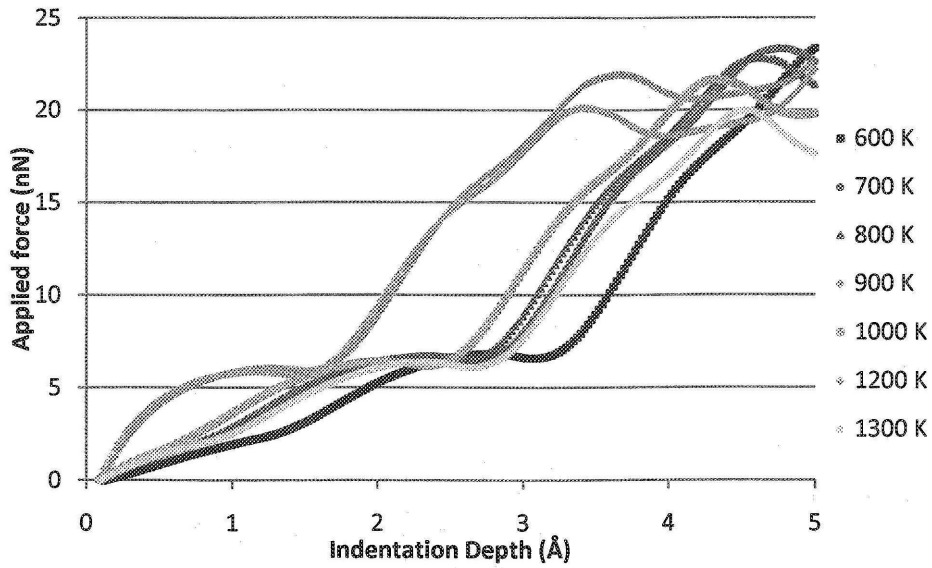


Figure 8 – Force vs Indentation Depth Graph for 600 K to 1300 K

For temperatures above 600 K, as shown in Figure 8, the same trends are not exhibited. Since the temperatures are so high, the kinetic energy within the system, and the corresponding atomic forces are quite large. As a result, the atoms are much more active and the atomic positions when contact occurs can be erratic. Therefore, it is expected that the nanohardness values obtained using the MD software becomes greatly less effective for temperatures in excess of 600 K. Thus, for large-scale simulations using temperatures higher than 600 K, further testing should be performed to ensure the results are accurate.

### 3.2.2. Force Sensitivity Analysis

The second sensitivity analysis performed investigated differing force summation techniques. As previously mentioned, during nanoindentation the total force on the copper substrate is constant since a displacement dependent nanoindentation process is implemented. Therefore, using MATLAB<sup>TM</sup> function Force\_Summation\_V\_7\_00, the following three force summation methods were analyzed:

- 1) Summation of all atoms except those that are clamped.
- 2) Summation of all atoms except those that are clamped or adjacent to clamped atoms.
- 3) Summation of only those atoms that are contacting the indenter.

In order to test the three methods of force summation, a simulation was performed using the following initial conditions:

Indenter Type	Spherical
Temperature	293 K
Substrate Size	6 by 6 by 6 unit cells (21.69 Å), PBCs in x, y directions and negative z-direction.
Indenter Diameter	3 unit cells (10.845 Å)
Indent Speed	20 m/s
Maximum Indentation Depth	5 Å
Clamping	Absolute clamping on all faces except the in the positive z-direction

The three force summation methods produce drastically differing force vs. indentation depth plots as shown in Figure 9. For comparison, Saraev and Miller's results will be used to ensure that the force summation technique chosen produces results that are within reason.

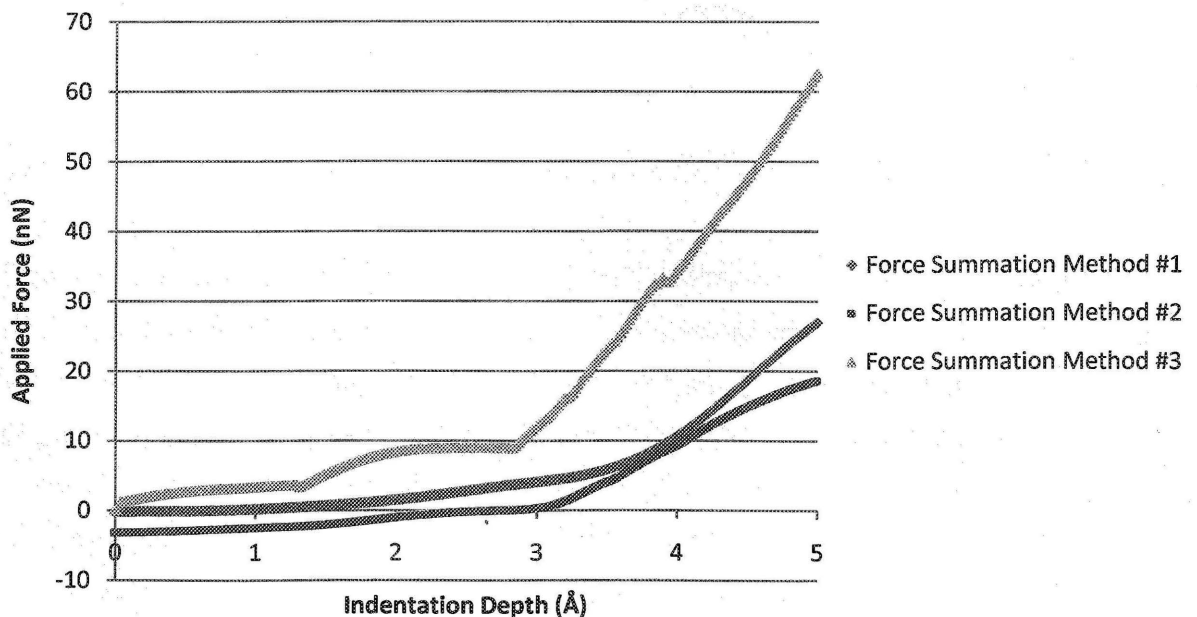


Figure 9 – Force Summation Method Comparison

For the case when only the contact forces are used, the force plot is erratic and the forces acting on the copper substrate are much higher than expected. As well, for the force summation method in which the forces for both the clamped atom and the atoms adjacent to them are discounted, the forces were found to be too low as the indenter must be indent almost  $3\text{\AA}$  into the substrate before the forces become positive values. Therefore, the most effective technique was determined to be the summation of all unclamped forces.

It should be noted, that for larger systems the difference between the first two force summation methods will be quite small as the forces adjacent to the clamped edges are expected to be miniscule for large atomic systems. For the nanoindentation simulations executed in this research, since the substrates used are not very large the forces near the edges are still rather significant. Regardless, it was felt that the force summation method chosen would still be the most effective one as it is the most accurate for the substrate sizes analyzed in this study.

### 3.2.3. Molecular Dynamics Displacement Sensitivity Analysis

The third sensitivity analysis performed studied the effect of nanoindentation on the atomic positions throughout the nanoindentation procedure. In this simulation, in addition to the equilibrium step prior to the nanoindentation and the indentation step itself, two more equilibrium steps have been included. Once the indentation is completed, a second equilibrium step is executed, followed by the removal of the indenter and a third equilibrium step. In this simulation, the deformation of the copper substrate should be clearly illustrated. The following initial conditions were used in the simulation:

Indenter Type	Spherical
Temperature	293 K
Substrate Size	4 by 4 by 4 unit cells ( $14.46\text{ \AA}$ ), PBCs in x, y- directions
Indenter Diameter	2 unit cells ( $7.134\text{ \AA}$ )
Indent Speed	10 m/s
Maximum Indentation Depth	$5\text{\AA}$
Clamping	Absolute clamping on all faces except the in the positive z-direction

The copper substrate in this simulation consists of 365 atoms. For the purposes of this test, only specific relevant atoms are highlighted. For clarity, the tables display the atomic positions ranking them by the value of their z-positions from largest to smallest. The percentage differences used in the tables refer only to the difference between the atomic positions in the z-direction from their starting positions to the current step. All tables display atomic positions in Angstroms (Å). In Tables 2–5, significant atomic displacements from each of the steps throughout the nanoindentation process are presented.

Atom #	Starting Positions (Å)			After Equilibrium #1 (Å)			% Difference
	X-Position	Y-Position	Z-Position	X-Position	Y-Position	Z-Position	
40	3.6150	7.2300	14.4600	3.5437	7.2300	14.3023	-1.0904
60	7.2300	3.6150	14.4600	7.2300	3.5437	14.3023	-1.0908
65	7.2300	7.2300	14.4600	7.2300	7.2301	14.2414	-1.5120
70	7.2300	10.8450	14.4600	7.2300	10.9164	14.3024	-1.0902
90	10.8450	7.2300	14.4600	10.9163	7.2301	14.3023	-1.0906
315	5.4225	5.4225	14.4600	5.3846	5.3846	14.2681	-1.3273
320	5.4225	9.0375	14.4600	5.3845	9.0755	14.2681	-1.3270
335	9.0375	5.4225	14.4600	9.0754	5.3846	14.2681	-1.3274
340	9.0375	9.0375	14.4600	9.0755	9.0755	14.2681	-1.3271
165	7.2300	5.4225	12.6525	7.2300	5.3719	12.5781	-0.5877
169	7.2300	9.0375	12.6525	7.2300	9.0883	12.5782	-0.5875
237	5.4225	7.2300	12.6525	5.3718	7.2301	12.5782	-0.5875
257	9.0375	7.2300	12.6525	9.0882	7.2301	12.5781	-0.5877
64	7.2300	7.2300	10.8450	7.2300	7.2301	10.8316	-0.1233
63	7.2300	7.2300	7.2300	7.2300	7.2300	7.2240	-0.0825
62	7.2300	7.2300	3.6150	7.2300	7.2300	3.5836	-0.8686

Table 2 – Atomic Positions at Equilibrium Step #1 from MD Simulations

As shown in Table 2, the atoms are not greatly displaced to reach equilibrium. In fact, only atoms whose z-positions are at a maximum move in excess of one percent. For larger systems, the displacements are expected to be larger to reach equilibrium, as atoms in the centre of the substrate will not be as greatly affected by the clamping along the walls. In Table 3, below, the atomic positions at the conclusion of the indentation are shown.

Atom #	Starting Positions (Å)			After Indentation (Å)			% Difference
	X-Position	Y-Position	Z-Position	X-Position	Y-Position	Z-Position	
40	3.6150	7.2300	14.4600	3.4757	7.2300	12.8143	-11.3810
60	7.2300	3.6150	14.4600	7.2300	3.4758	12.8143	-11.3810
65	7.2300	7.2300	14.4600	7.2300	7.2301	9.2387	-36.1083
70	7.2300	10.8450	14.4600	7.2300	10.9843	12.8143	-11.3810
90	10.8450	7.2300	14.4600	10.9843	7.2300	12.8143	-11.3810
315	5.4225	5.4225	14.4600	5.6488	5.6488	13.2466	-8.3915
320	5.4225	9.0375	14.4600	5.6488	8.8113	13.2467	-8.3904
335	9.0375	5.4225	14.4600	8.8112	5.6488	13.2466	-8.3915
340	9.0375	9.0375	14.4600	8.8113	8.8113	13.2467	-8.3904
165	7.2300	5.4225	12.6525	7.2300	5.2199	11.4435	-9.5551
169	7.2300	9.0375	12.6525	7.2300	9.2401	11.4437	-9.5542
237	5.4225	7.2300	12.6525	5.2199	7.2302	11.4436	-9.5546
257	9.0375	7.2300	12.6525	9.2401	7.2302	11.4436	-9.5547
64	7.2300	7.2300	10.8450	7.2300	7.2299	6.9723	-35.7097
63	7.2300	7.2300	7.2300	7.2298	7.2304	4.6090	-36.2512
62	7.2300	7.2300	3.6150	7.2301	7.2298	2.4543	-32.1071

**Table 3 – Atomic Positions after Indentation Step from MD Simulations**

The atomic positions at the conclusion of the indentation are illustrated in Table 3. At this point, the greatest displacement is found in atom 65 as it is the first atom contacted and is indented the full 5Å. Other atoms that are contacted directly by the indenter include atoms 40, 60, 70, 90, 315, 320, 335 and 340. Interestingly, as a result of the indentation, all the atoms directly below the first atom indented, atom 65, are displaced significantly. As shown in Table 3, atoms 62, 63, and 64 are each drastically displaced from their original z-positions. These displacements result from the compression of the substrate as atom 65 is pushed in the negative z-direction. This compression of the substrate is also clearly seen, but to a lesser extent, in atoms 165, 169, 237, and 257.



Atom #	Starting Positions (Å)			After Equilibrium #2 (Å)			% Difference
	X-Position	Y-Position	Z-Position	X-Position	Y-Position	Z-Position	
40	3.6150	7.2300	14.4600	3.4757	7.2300	12.8143	-11.3810
60	7.2300	3.6150	14.4600	7.2300	3.4758	12.8143	-11.3810
65	7.2300	7.2300	14.4600	7.2300	7.2301	9.2387	-36.1083
70	7.2300	10.8450	14.4600	7.2300	10.9843	12.8143	-11.3810
90	10.8450	7.2300	14.4600	10.9843	7.2300	12.8143	-11.3810
315	5.4225	5.4225	14.4600	5.7346	5.7347	13.1599	-8.9911
320	5.4225	9.0375	14.4600	5.7346	8.7256	13.1602	-8.9889
335	9.0375	5.4225	14.4600	8.7255	5.7347	13.1599	-8.9910
340	9.0375	9.0375	14.4600	8.7254	8.7256	13.1602	-8.9888
165	7.2300	5.4225	12.6525	7.2300	5.2108	11.3357	-10.4073
169	7.2300	9.0375	12.6525	7.2300	9.2492	11.3359	-10.4059
237	5.4225	7.2300	12.6525	5.2108	7.2304	11.3358	-10.4066
257	9.0375	7.2300	12.6525	9.2492	7.2304	11.3358	-10.4066
64	7.2300	7.2300	10.8450	7.2300	7.2300	6.9499	-35.9159
63	7.2300	7.2300	7.2300	7.2300	7.2300	4.5482	-37.0926
62	7.2300	7.2300	3.6150	7.2300	7.2300	2.3876	-33.9516

**Table 4 – Atomic Positions at Equilibrium Step #2 from MD Simulations**

The atomic positions after the second equilibrium has been reached are illustrated in Table 4. During this step, the equilibrium is established as the indenter is held in the position it had reached at the conclusion of the indentation step. Seeing as the atoms directly contacting the indenter are unable to move, there is no change in z-positions for any of the atoms that are directly contacting the indenter. However, for the atoms that become compressed due to the indented atoms above them, there is still displacement in the negative z-direction. For each of the aforementioned atoms, the displacement in the negative z-direction continues as the system responds to the compression from the indented atoms above. Due to the speed of the nanoindentation, however, the system has been able to respond fairly well to the nanoindentation as the indenter is lowered into the substrate. As a result, to reach the second equilibrium step range the atomic displacements are quite small.

Atom #	Starting Positions (Å)			After Equilibrium #3 (Å)			% Difference
	X-Position	Y-Position	Z-Position	X-Position	Y-Position	Z-Position	
40	3.6150	7.2300	14.4600	3.5153	7.2300	14.2965	-1.1309
60	7.2300	3.6150	14.4600	7.2300	3.5153	14.2965	-1.1309
65	7.2300	7.2300	14.4600	7.2300	7.2300	10.8244	-25.1426
70	7.2300	10.8450	14.4600	7.2300	10.9447	14.2965	-1.1309
90	10.8450	7.2300	14.4600	10.9447	7.2300	14.2965	-1.1309
315	5.4225	5.4225	14.4600	5.3063	5.3063	14.2512	-1.4438
320	5.4225	9.0375	14.4600	5.3063	9.1537	14.2512	-1.4438
335	9.0375	5.4225	14.4600	9.1537	5.3063	14.2512	-1.4438
340	9.0375	9.0375	14.4600	9.1537	9.1537	14.2512	-1.4438
165	7.2300	5.4225	12.6525	7.2300	5.4061	12.6583	0.0460
169	7.2300	9.0375	12.6525	7.2300	9.0539	12.6583	0.0460
237	5.4225	7.2300	12.6525	5.4061	7.2300	12.6583	0.0460
257	9.0375	7.2300	12.6525	9.0539	7.2300	12.6583	0.0460
64	7.2300	7.2300	10.8450	7.2300	7.2300	7.9159	-27.0084
63	7.2300	7.2300	7.2300	7.2300	7.2299	5.7012	-21.1454
62	7.2300	7.2300	3.6150	7.2300	7.2300	3.3763	-6.6033

**Table 5 – Atomic Positions at Equilibrium Step #3 from MD Simulations**

Table 5 demonstrates the behaviour of the atoms after the indenter is removed and a third equilibrium step is performed. In this table, it is evident that after letting the system relax, the deformation still remains. It is noted however, that the system does partially expand once the indenter is removed. Interestingly, after the third equilibrium step is completed, significant displacements only remain in atoms 62–65. For the remaining atoms, however, their positions are almost identical to their original equilibrium positions. One significant change in the atomic positions is observed in atoms 165, 169, 237, and 257. After the third equilibrium step is completed, these atoms actually exhibit displacements in the positive z-direction. While this may seem puzzling, these atoms are being pushed in the positive z-direction due to their proximity to atom 65. Through this analysis, it is evident that MD MATLAB™ function NI\_MD\_V\_7\_00 models the behaviour of a substrate undergoing nanoindentation effectively.

### 3.2.4. Clamping Sensitivity Analysis

The final sensitivity analysis performed studied the effect of clamping on the nanohardness. When clamping a material, the software enables two different types of clamping to be performed; absolute clamping or relative clamping. Absolute clamping restricts motions of clamped atoms in all directions, while relative clamping restricts motion only in the direction that is being clamped. In this sensitivity analysis the following four methods of clamping were investigated:

- 1) Relative clamping only on bottom face
- 2) Absolute clamping only on bottom face
- 3) Relative clamping on all faces except top face
- 4) Absolute clamping on all faces except top face

Using the following initial conditions, simulations using each of the clamping techniques were performed:

Indenter Type	Spherical
Temperature	100 K
Substrate Size	6 by 6 by 4 unit cells, PBCs in x, y-directions
Indenter Diameter	2 unit cells (7.134 Å)
Indent Speed	20 m/s
Maximum Indentation Depth	5Å

Using these simulation conditions, nanoindentation were performed for each of the clamping techniques. The simulation results are shown in Figure 10.

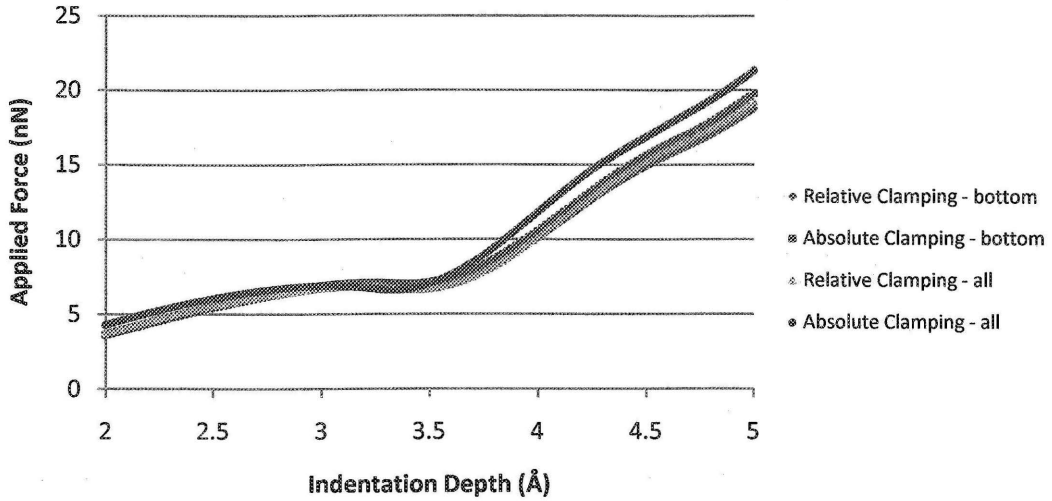


Figure 10 – Comparison of Clamping Techniques using MD Simulations

From the graph, it is apparent that the difference between the clamping techniques is not drastic. As a result, the clamping technique implemented is not expected to have a great impact on the hardness of the material. However, the graph indicates that absolute clamping is more effective than relative clamping. Additionally, absolute clamping on all faces except the face that directly contacts the indenter requires significantly less time steps to reach equilibrium than absolute clamping on the bottom face only.

### 3.3. Molecular Dynamics Nanoindentation Simulation Conditions

In Sections 3.4–3.6, the simulations results are shown. Each simulation was performed using a 10 by 10 by 10 unit cell (36.15 by 36.15 by 36.15 Å) copper substrate with PBCs in the x and y-directions as well as the negative z-direction. Simulations were performed using speeds ranging from 10 m/s and 30 m/s, while the indenter sizes used (before the indenter is trimmed) ranged from 4 by 4 by 4 unit cells (14.46 by 14.46 by 14.46 Å) to 7 by 7 by 7 (25.3 by 25.3 by 25.3 Å) unit cells. The simulation temperature is maintained at 293 K, and absolute clamping is used on all faces except the face coinciding with the maximum z-positions.

The indenters used for the simulations are quite unique and are each expected to produce distinct results. The spherical indenter's height is equal to its radius. The Berkovich indenter produces a height approximately equal to one-third of the length of the base. For the square pyramidal, the angle used was  $54.74^\circ$ . As a result, the height of the pyramid will be equal to the width of the base or twice the height of the spherical indenter. As a result, the Berkovich indenter is expected to produce the smallest hardness values followed by the spherical and square pyramidal indenters respectively. In Figures 11–22, below, the simulation results are graphed.

### 3.4. Molecular Dynamics Spherical Indenter Results

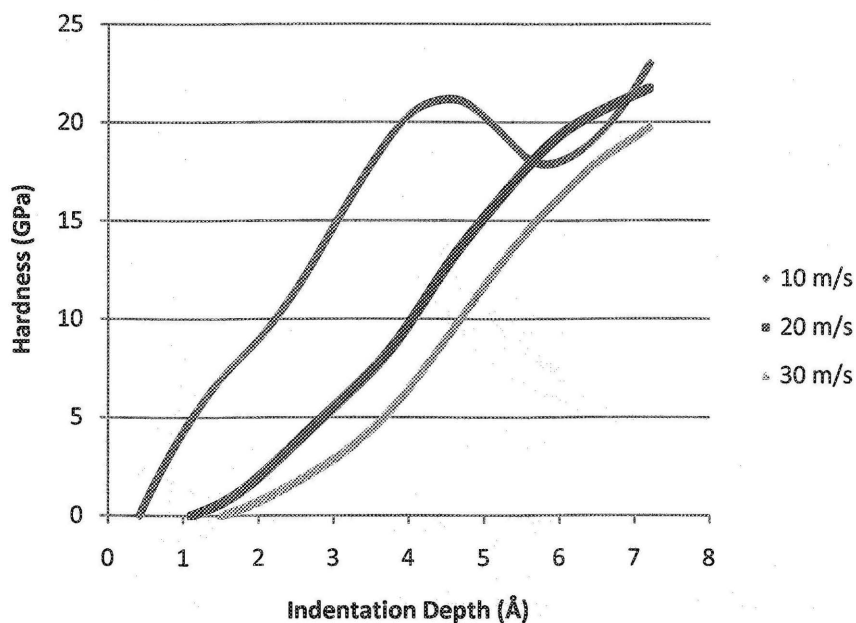


Figure 11 – MD Results using Spherical Indenter with Diameter of Four Unit Cells

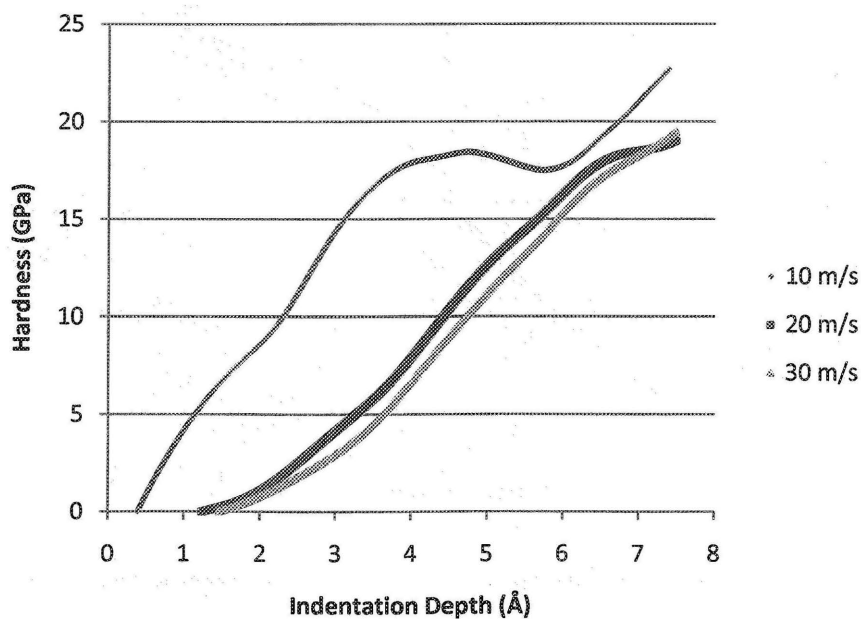


Figure 12 – MD Results using Spherical Indenter with Diameter of Five Unit Cells

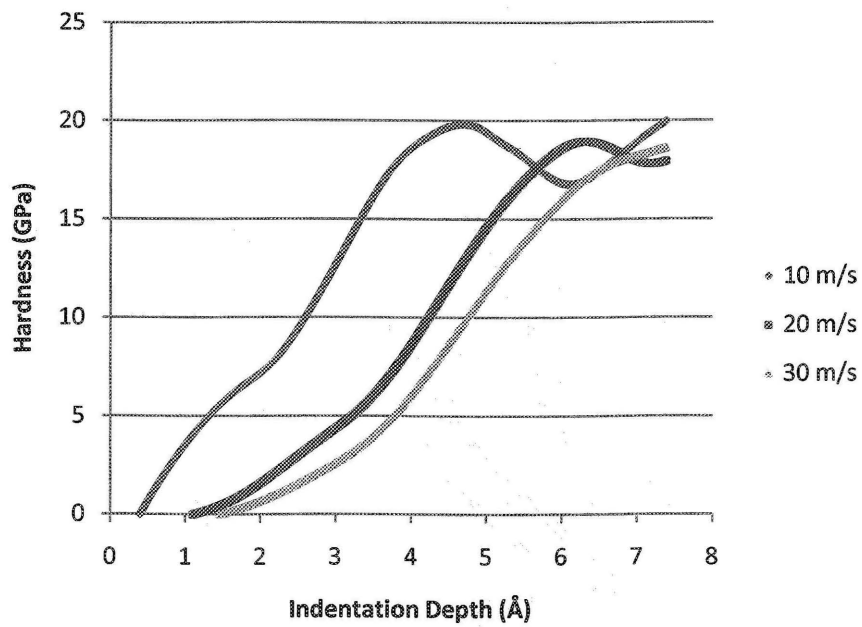


Figure 13 – MD Results using Spherical Indenter with Diameter of Six Unit Cells

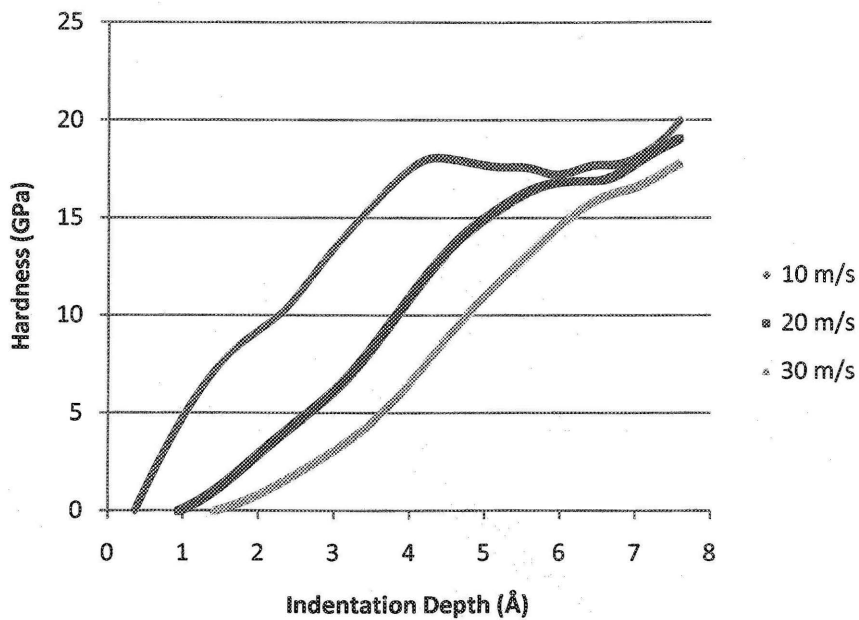


Figure 14 – MD Results using Spherical Indenter with Diameter of Seven Unit Cells

3.5. Molecular Dynamics Berkovich Indenter Results

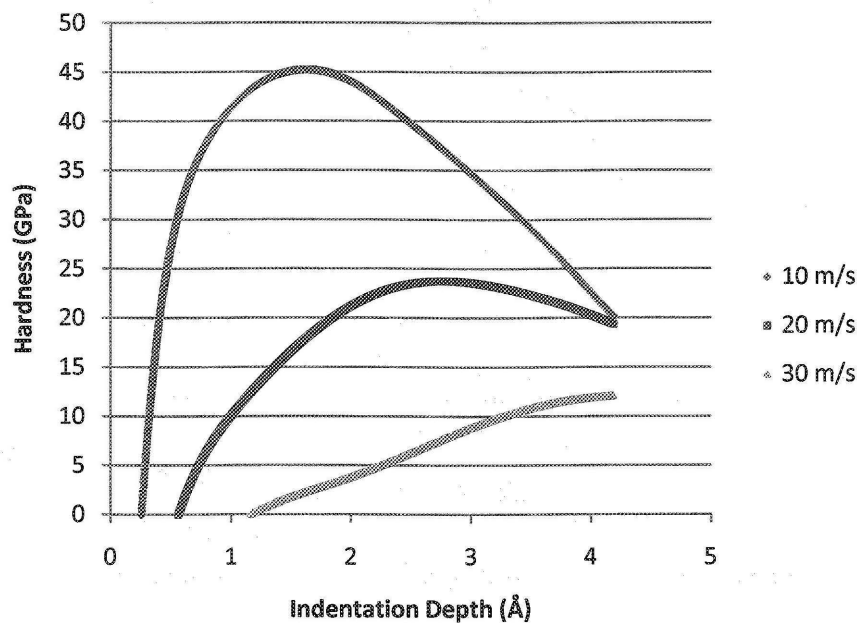


Figure 15 – MD Results using Berkovich Indenter with Base of Length Four Unit Cells

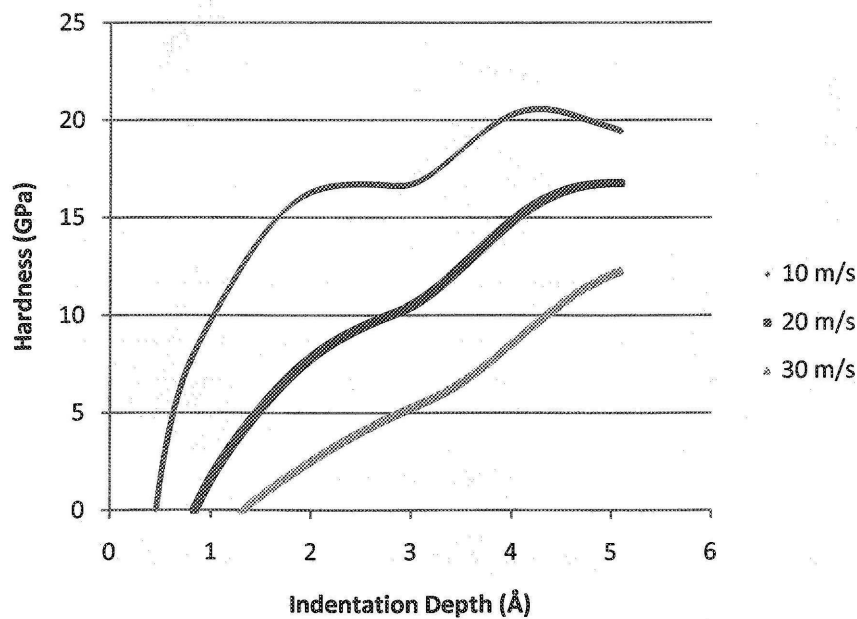


Figure 16 – MD Results using Berkovich Indenter with Base of Length Five Unit Cells



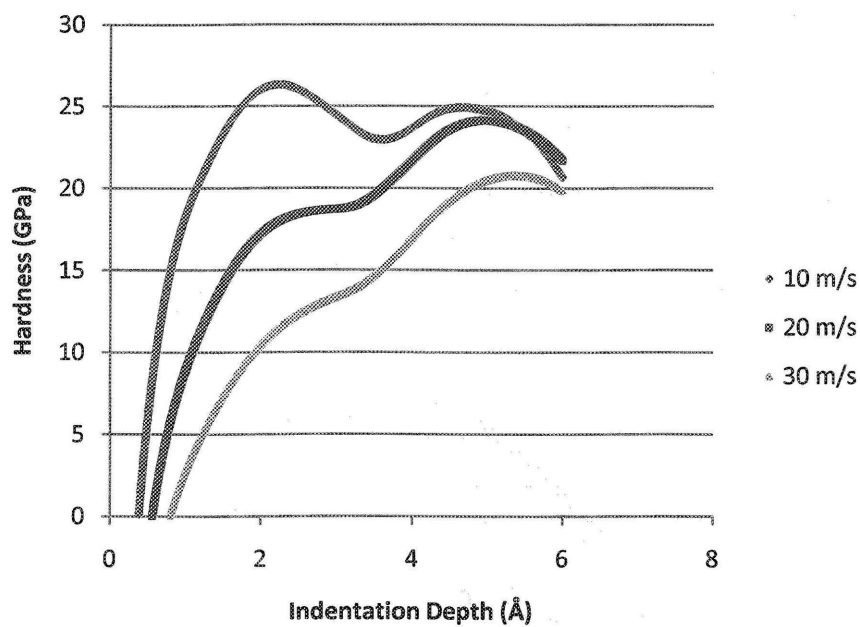


Figure 17 – MD Results using Berkovich Indenter with Base of Length Six Unit Cells

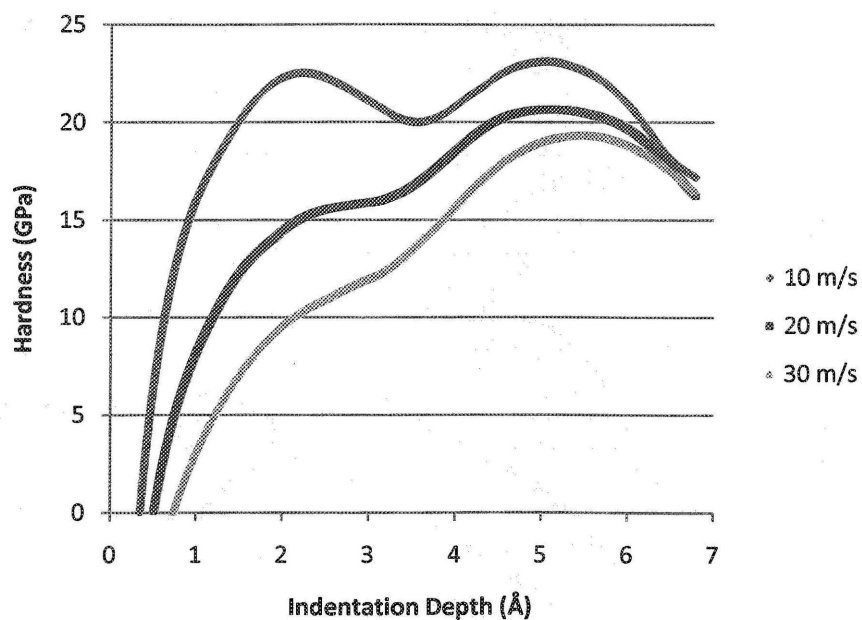


Figure 18 – MD Results using Berkovich Indenter with Base of Length Seven Unit Cells

### 3.6. Molecular Dynamics Square Pyramidal Indenter Results

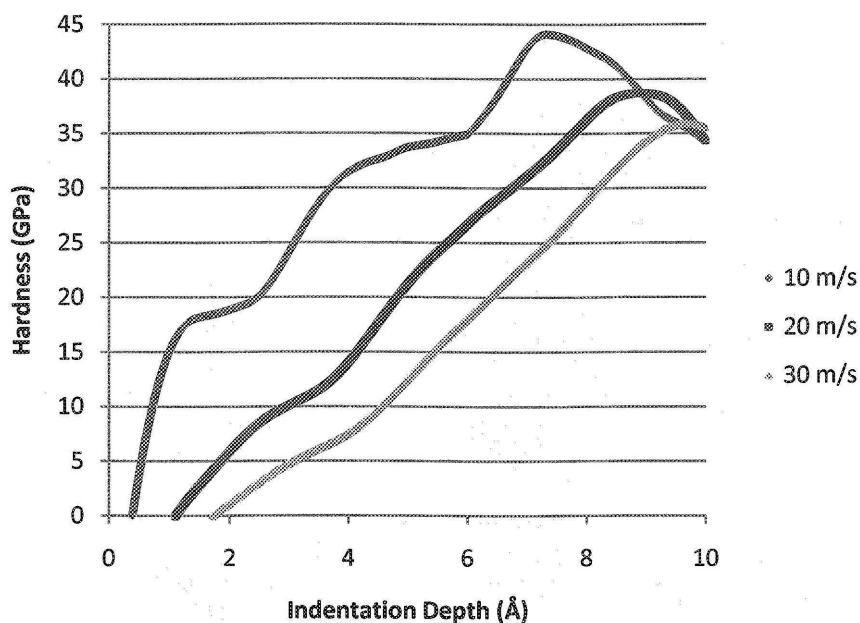


Figure 19 – MD Results using Pyramidal Indenter with Base Four by Four Unit Cells

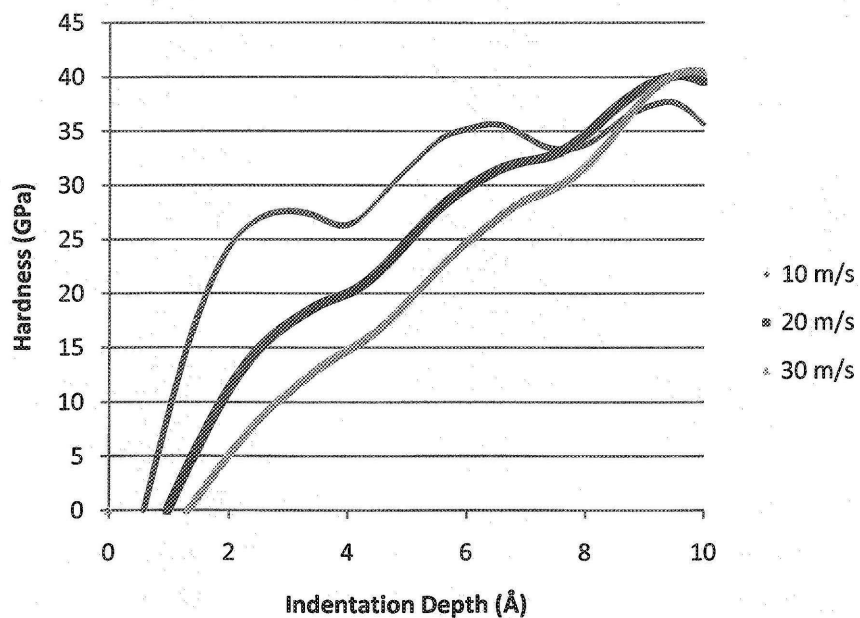


Figure 20 – MD Results using Pyramidal Indenter with Base Five by Five Unit Cells

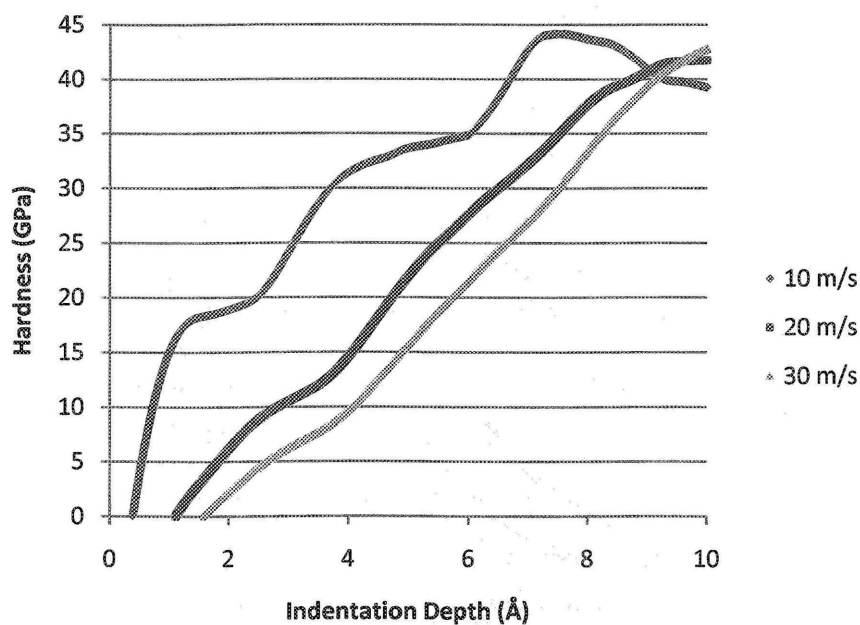


Figure 21 – MD Results using Pyramidal Indenter with Base Six by Six Unit Cells

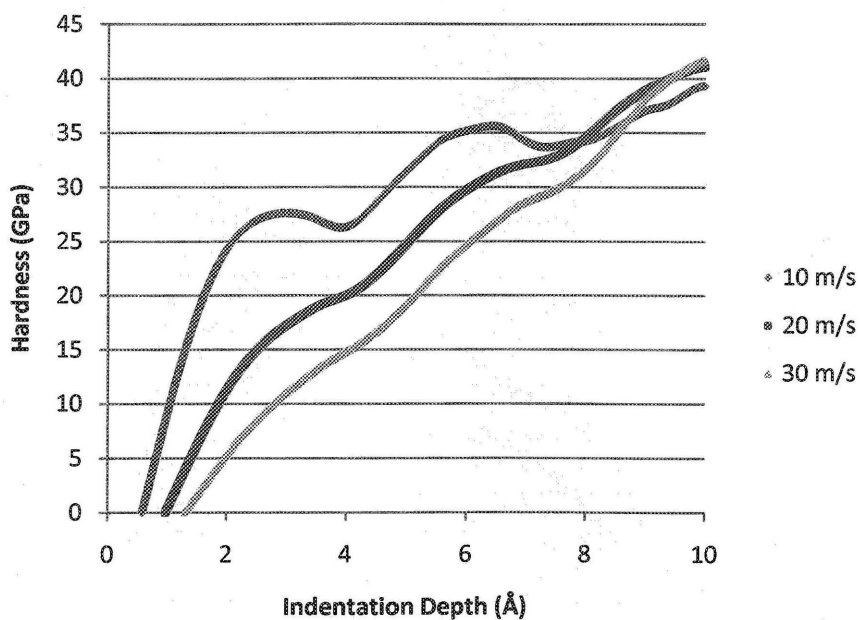


Figure 22 – MD Results using Pyramidal Indenter with Base Seven by Seven Unit Cells

### 3.7. Discussion

In Figures 11–22, it can be observed that the hardness vs. indentation depth plot for the MD simulations never start right at an indentation depth of 0 Å. Due to the algorithm used for the first equilibrium step, the indentation step is commenced once the change in applied force per time step is within the specified limits. As a result, the applied force on the system is still not a positive value. Due to the size of the copper substrate used for the MD simulations, the data shown in Figures 11–22 cannot be used to determine the hardness of copper metal nor can it be used in the design process of products using copper nanomaterials. However, the trends illustrated in the hardness graphs reveal valuable information with respect to nanoindentation.

In Figures 11–14, the results of the spherical indenter nanoindentation simulations are illustrated. In these graphs, significant changes in the hardness vs. indentation depth plot can be observed when the indentation speed is altered. As the indentation speed increases, the hardness values increase quicker over indentation depth. Most notably, for the indentation speed of 10 m/s, the hardness responds very quickly to changes in indentation depth. This is expected as the 10 m/s indentation speed allows significantly more time for the system to deform and for the forces to be transferred to the system.

Interestingly, for the 10 m/s indentation, the spherical indenter plots all level off at indentation depths ranging between 4 Å and 5 Å. For the higher speed nanoindentation, however, this phenomenon does not occur. At these speeds, the plots are only seen to level off when the indentation depth approaches the maximum indentation depth. For slower speeds, the atomic deformations are much smaller and therefore, the other atoms are able to adjust as the indenter is moved down. For the faster speed indentations, the system is unable to adjust to the larger deformation as quickly. Therefore, for higher speed indentations, the hardness plots only reach their maximum as the indentation depth approaches the maximum indentation depth.

An additional important result to note for simulations using spherical indenters is that the hardness decreases as the indenter size decreases. This trend is most clearly illustrated in the simulations using the indentation speeds of 20 m/s and 30 m/s. Commonly referred to as the

“indenter size effect,” the reduction of hardness values for larger indenter sizes is consistent with the findings of many other research studies.<sup>[36]</sup>

In Figures 15–18, it is evident that the simulations using the Berkovich indenter are not as effective as those using spherical indenter. For example, in Figure 15, the 10 m/s indentation reaches its maximum hardness at a indentation depth of approximately 1.5 Å. Since the Berkovich indenter’s height is much smaller than its width, the majority of the indenter contacts the substrate very quickly in the simulation. As a result, prior to the system being able to react to the indentation, the indenter is completely submerged within the substrate. This is especially the case for the two smaller indenters. One explanation for the irregularities that are observed for the smaller indenters lies within the method with which contact area is calculated. Since the contact area is calculated based on the geometry of the indenter rather than being based on the atoms actually used in the simulation, there are large discrepancies for such small indenters. However, for larger indenters, as will be used for nanoindentations using large atomic systems, these discrepancies will be relatively insignificant.

For the two larger indenters, as shown in Figures 17 and 18, the hardness vs. indentation depth plots for the three indentation speeds appear to converge more effectively. The increased convergence is expected as for the larger Berkovich indenters, the indenter can be submerged further within the substrate than for the smaller sized indenters. For the Berkovich indenters, the “indenter size effect” is not observed. It is expected that for simulations using larger substrates, the “indenter size effect” would indeed be observed, however, for such as small indentation depth, the system is unable to respond effectively to the indentation.

For the square pyramidal indenters, shown in Figures 19–22, many of the same trends as shown for spherical and Berkovich indenters are observed. As in the simulations using spherical and Berkovich indenters, the hardness vs. indentation depth plots increase faster for slower indentation speeds. However, for square pyramidal indenters, the three plots crossover each other much before the maximum indentation depth is reached. This trend was not observed for the other two indenter types. As well, the “indenter size effect” is not presented for square pyramidal indenters. If the indenters were to approach the maximum indentation depth, it is expected that

the “indenter size effect” would be illustrated in the results. However, for a substrate size of 10 by 10 by 10 unit cells (36.15 by 36.15 by 36.15 Å) as used, indent the substrate to such an extent is not advisable. In that case, the system would be challenged to respond to such a compression within the system and the results obtained would be inaccurate.

When comparing the three indenter types, the most striking difference is the maximum nanohardness observed in the different types of indenters. For spherical indenters, the maximum hardness is found to be between 18–20 GPa. This is consistent with the results of Saraev and Miller discussed Section 3.1. For the Berkovich indenter, the maximum hardness differs substantially between the different indenter sizes. Thus, it is concluded that for more effective results, testing must be performed using larger substrate and indenter sizes. For the square pyramidal indenter, the maximum hardness observed in the plots is between 35–40 GPa. While it was expected that the maximum hardness using a square pyramidal indenter would be larger than when using a spherical indenter, the difference between the two is much larger than expected.

In all of the MD simulations performed, the difference between the simulations performed using indentation speeds of 10 m/s and the two faster indentation speeds of 20 m/s and 30 m/s is quite substantial. Based on this, it is inferred that in further studies, testing for intermediate speeds should be employed. With the completion of further testing, it would be possible to determine the optimal indentation speed with which to perform nanoindentation experiments that would provide the best results while requiring the shortest simulation time.

### 3.8. Chapter Summary

This chapter describes MD simulations sensitivity tests performed and larger scale nanoindentation simulations. These tests were very effective, as expected trends were illustrated and the maximum hardness results for spherical indenters were found to be extremely similar to the results expected based on the research conducted by Saraev and Miller. It was evident, however, that the nanoindentation simulations in this research are merely the starting point to determine the actual hardness of a copper nanomaterial. For the nanohardness of copper to be determined, one must perform much larger scale simulations as performed in the aforementioned research while using a more appropriately sized indenter. In Chapter 4, a bridged finite element – molecular dynamics (FE-MD) method will be introduced. This bridged FE-MD method will be used to performed nanoindentation experimentation more effectively while significantly reducing simulation time.

## **4. Bridged Finite Element – Molecular Dynamics Nanoindentation**

---

### **4.1. Bridged Finite Element – Molecular Dynamics Introduction**

Despite the success of MD simulations, recent studies have introduced revolutionary methods that enable nanosized materials to be analyzed more efficiently. These methods, commonly referred to as multi-scale methods, couple the macro and nano scale to more effectively analyze the behaviour of large atomic systems at the macro scale. Most multi-scale methods employ the use of both macro (such as FEM) and atomistic (such as MD) simulation tools to perform their experimentation. In this chapter, a unique bridged method is introduced and applied to nanoindentation simulations. This method, developed by Narayan, combines the theories of finite elements method (FEM) and molecular dynamics (MD).<sup>[25]</sup> This tool significantly reduces simulation time required by traditional MD simulations while maintaining precision. Compared to the thousands of time steps that may be required for MD simulations to reach equilibrium, it can be reached in only a few steps using the bridged FE-MD technique.

The strength of this method lies with the novel discretization scheme utilized that allows the molecular dynamics equations of motion to be easily solved using traditional FEM techniques. This bridged method has proven extremely effective in the aforementioned work of Narayan. However, due to the tremendous size of the stiffness matrices produced for large atomic systems, significant computational resources are required to execute these simulations. Therefore, to fully explore the strength of the bridged FE-MD method and use it for meaningful simulations, more powerful computational tools than those available to this author must be employed.

This chapter will examine the bridged FE-MD method in detail and the software created for the original MATLAB<sup>TM</sup> code FE\_MD\_V\_7\_00 as well as its nanoindentation adaptation MATLAB<sup>TM</sup> code NI\_FE\_MD\_V\_7\_00 will be discussed.



## **4.2. Traditional Finite Elements Method Theory**

Finite element method (FEM) is a very common numerical method used to solve many engineering problems. FEM simulations are most commonly used for structural, thermal, electromagnetic, and fluid problems. A general overview of some relevant FEM theories will be discussed in this section; however, for a more detailed overview of basic FEM theory, the reader is urged to consult FEM reference materials.

### **4.2.1. General Form of Finite Element Method Simulations**

While FEM solutions have been used for a wide scope of problems, the general algorithm used by all FEM tools is virtually identical. This general algorithm is defined by the following steps:

- 1) Discretize the system and select element type – the nodes and elements for the system are defined.
- 2) Select a displacement function.
- 3) Define the strain/displacement and stress/strain relationships.
- 4) Derive the element stiffness equations and matrices.
- 5) Assemble the individual elemental stiffness matrices to derive the global stiffness matrix or global equations.
- 6) Introduce boundary conditions – these boundary conditions can be used to remove rows or columns from the global stiffness matrix and speed up the simulation.
- 7) Solve for unknown degrees of freedom (in this case the only unknowns are nodal displacements) and elemental stress/strains.

The elemental stresses and strains as calculated using FEM are not relevant to the research being conducting in this study. Therefore, while the code solves for the elemental stresses and strains, those results will not be discussed in this thesis. In the coming sections, traditional FEM discretization methodology and stiffness matrices derivation will be discussed in more depth.

### 4.2.2. General Form of a FEM Stiffness Equation

To demonstrate the formulation of the stiffness matrices used in FEM in its most general form, the most basic element in finite elements, the spring element, is discussed. A diagram of a one-dimensional spring element is shown in Figure 23. The nodal displacements for the spring element are denoted by  $\hat{d}_{1x}$  and  $\hat{d}_{2x}$ , while  $F_{1x}$  and  $F_{2x}$  represent the external forces on those nodes.

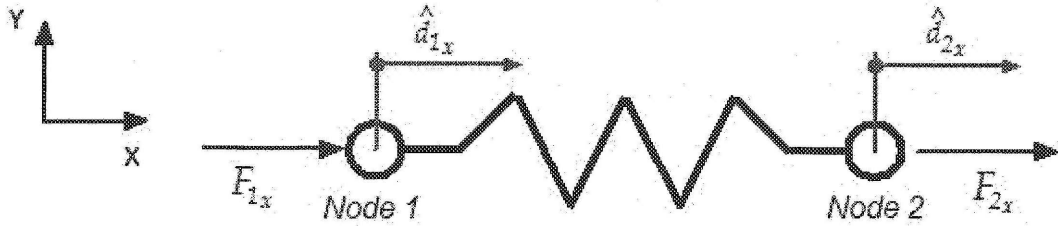


Figure 23 – One dimensional two-node spring element

The internal energy of the element can be expressed with respect to the displacements as,

$$U = U(\hat{d}_{1x}, \hat{d}_{2x}) \quad [52]$$

The internal forces of the element,  $\hat{f}_{1x}$  and  $\hat{f}_{2x}$ , can be expressed in relationship to the internal energy of the element as shown in Equations 53 and 54. Additionally, the total force acting on each node is total zero, shown in Equation 55.

$$\hat{f}_{1x} = \left. \frac{-\partial U}{\partial(\hat{d}_{1x})} \right|_{\substack{\hat{d}_{1x}=\Delta_1 \\ \hat{d}_{2x}=\Delta_2}} \quad [53]$$

$$\hat{f}_{2x} = \left. \frac{-\partial U}{\partial(\hat{d}_{2x})} \right|_{\substack{\hat{d}_{1x}=\Delta_1 \\ \hat{d}_{2x}=\Delta_2}} \quad [54]$$

$$\begin{bmatrix} F_{1x} \\ F_{2x} \end{bmatrix} + \begin{bmatrix} \hat{f}_{1x} \\ \hat{f}_{2x} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad [55]$$

Using a multi-variable Taylor series expansion, the internal nodal force derivatives can be expressed with respect to the displacement in each node.

$$\left. \frac{\partial U}{\partial(\hat{d}_{1x})} \right|_{\substack{\hat{d}_{1x}=\Delta_1 \\ \hat{d}_{2x}=\Delta_2}} \approx \left. \frac{\partial U}{\partial(\hat{d}_{1x})} \right|_{\substack{\hat{d}_{1x}=0 \\ \hat{d}_{2x}=0}} + (\Delta_1) \left. \frac{\partial^2 U}{\partial(\hat{d}_{1x})^2} \right|_{\substack{\hat{d}_{1x}=0 \\ \hat{d}_{2x}=0}} + (\Delta_2) \left. \frac{\partial^2 U}{\partial(\hat{d}_{1x})\partial(\hat{d}_{2x})} \right|_{\substack{\hat{d}_{1x}=0 \\ \hat{d}_{2x}=0}} \quad [56]$$

$$\left. \frac{\partial U}{\partial(\hat{d}_{2x})} \right|_{\substack{\hat{d}_{1x}=\Delta_1 \\ \hat{d}_{2x}=\Delta_2}} \approx \left. \frac{\partial U}{\partial(\hat{d}_{2x})} \right|_{\substack{\hat{d}_{1x}=0 \\ \hat{d}_{2x}=0}} + (\Delta_1) \left. \frac{\partial^2 U}{\partial(\hat{d}_{2x})\partial(\hat{d}_{1x})} \right|_{\substack{\hat{d}_{1x}=0 \\ \hat{d}_{2x}=0}} + (\Delta_2) \left. \frac{\partial^2 U}{\partial(\hat{d}_{2x})^2} \right|_{\substack{\hat{d}_{1x}=0 \\ \hat{d}_{2x}=0}} \quad [57]$$

By combining Equations 56 and 57, a stiffness matrix for the non-linear spring element can be established. Equation 58, is given in standard FEM stiffness matrix  $[K]\{d\} = [F]$  as can be found in any traditional FEM resource.

$$\begin{bmatrix} \left. \frac{\partial^2 U}{\partial(\hat{d}_{1x})^2} \right|_{\substack{\hat{d}_{1x}=0 \\ \hat{d}_{2x}=0}} & \left. \frac{\partial^2 U}{\partial(\hat{d}_{2x})\partial(\hat{d}_{1x})} \right|_{\substack{\hat{d}_{1x}=0 \\ \hat{d}_{2x}=0}} \\ \left. \frac{\partial^2 U}{\partial(\hat{d}_{1x})\partial(\hat{d}_{2x})} \right|_{\substack{\hat{d}_{1x}=0 \\ \hat{d}_{2x}=0}} & \left. \frac{\partial^2 U}{\partial(\hat{d}_{2x})^2} \right|_{\substack{\hat{d}_{1x}=0 \\ \hat{d}_{2x}=0}} \end{bmatrix} \begin{Bmatrix} \Delta_1 \\ \Delta_2 \end{Bmatrix} = \begin{Bmatrix} F_{1x} \\ F_{2x} \end{Bmatrix} - \begin{bmatrix} \left. \frac{\partial U}{\partial(\hat{d}_{1x})} \right|_{\substack{\hat{d}_{1x}=0 \\ \hat{d}_{2x}=0}} \\ \left. \frac{\partial U}{\partial(\hat{d}_{2x})} \right|_{\substack{\hat{d}_{1x}=0 \\ \hat{d}_{2x}=0}} \end{bmatrix} \quad [58]$$

For a system of elements, a global stiffness matrix can be established by the same process. The general form of a global stiffness matrix is illustrated in Equation 59 for a one-dimensional non-linear system. It should be noted that for more complex elements such as those used in the bridged FE-MD software, the stiffness matrices constructed will much more intricate. This issue will be discussed in more detail in Section 4.3 when the bridged FE-MD methodology is discussed. Additionally, as Equation 59 only displays a system of equations used to solve the unknown degrees of freedom in the x-direction, the size of the global stiffness matrix will be enlarged for each additional degree of freedom (additional displacements in y and z-directions, or rotations in all three directions) added to the system.

$$\begin{bmatrix} \frac{\partial^2 U_{tot}}{\partial \hat{d}_{1x}^2} & \dots & \frac{\partial^2 U_{tot}}{\partial \hat{d}_{nx} \partial \hat{d}_{1x}} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 U_{tot}}{\partial \hat{d}_{1x} \partial \hat{d}_{nx}} & \dots & \frac{\partial^2 U_{tot}}{\partial \hat{d}_{nx}^2} \end{bmatrix} \begin{Bmatrix} \hat{d}_{1x} \\ \hat{d}_{2x} \\ \vdots \\ \hat{d}_{nx} \end{Bmatrix} = \begin{Bmatrix} F_{1x} \\ \vdots \\ F_{nx} \end{Bmatrix} - \begin{Bmatrix} \frac{\partial U_{tot}}{\partial \hat{d}_{1x}} \\ \vdots \\ \frac{\partial U_{tot}}{\partial \hat{d}_{nx}} \end{Bmatrix} \quad [59]$$

### 4.2.3. Traditional FEM Discretization Scheme

In traditional FEM discretization schemes, linear spring elements are viewed as two-body potential, as elements are defined between each pair of nodes. In Figure 24, a seven element example using two-body potential has been created using five nodes. While elements 2, 4 and 6 appear to be three-node elements, they are simply elements using two nodes that are not adjacent to one another.

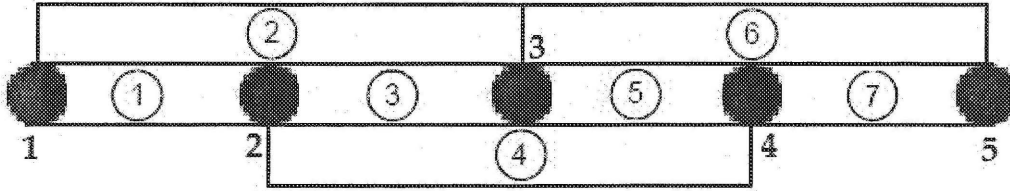


Figure 24 – Assembly of two-body potential linear spring elements

Since all elements in Figure 24 are defined as having only two nodes, the potential energy can be defined by the following linear spring equation which defines that between any two nodes,  $i$  and  $j$ , the potential energy is merely a function of the spring constant,  $k$ , and the distance between the two nodes.

$$U_{ij} = \frac{1}{2} k (\hat{d}_{ix} - \hat{d}_{jx})^2 \quad [60]$$

Using the definition of potential energy as described in Equation 60, the element stiffness matrices for each of the seven elements can be determined as shown in Equations 61–67.

$$[K]_1 = k_1 \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad [61]$$

$$[K]_2 = k_2 \begin{bmatrix} 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad [62]$$

$$[K]_3 = k_3 \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad [63]$$

$$[K]_4 = k_4 \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad [64]$$

$$[K]_5 = k_5 \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad [65]$$

$$[K]_6 = k_6 \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 \end{bmatrix} \quad [66]$$

$$[K]_7 = k_7 \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \quad [67]$$

If it is assumed that the stiffness coefficient,  $k$ , is constant throughout the material as will be done for all simulations performed in this research, the seven elemental stiffness matrices can be summed to create a global stiffness matrix as shown in Equation 68. The complete stiffness system of equations is shown in matrix form in Equation 69.

$$[K]_{Total} = k \begin{bmatrix} 2 & -1 & -1 & 0 & 0 \\ -1 & 3 & -1 & -1 & 0 \\ -1 & -1 & 4 & -1 & -1 \\ 0 & -1 & -1 & 3 & -1 \\ 0 & 0 & -1 & -1 & 2 \end{bmatrix} \quad [68]$$

$$k \begin{bmatrix} 2 & -1 & -1 & 0 & 0 \\ -1 & 3 & -1 & -1 & 0 \\ -1 & -1 & 4 & -1 & -1 \\ 0 & -1 & -1 & 3 & -1 \\ 0 & 0 & -1 & -1 & 2 \end{bmatrix} \begin{Bmatrix} \hat{d}_{1x} \\ \hat{d}_{2x} \\ \hat{d}_{3x} \\ \hat{d}_{4x} \\ \hat{d}_{5x} \end{Bmatrix} = \begin{Bmatrix} F_{1x} \\ F_{2x} \\ F_{3x} \\ F_{4x} \\ F_{5x} \end{Bmatrix} \quad [69]$$

### 4.3. Bridged Finite Element – Molecular Dynamics Theory

In this section, the FEM theory specific to the bridged FE-MD method is discussed. While similar to the traditional FEM theory, a discretization scheme that is unique to this bridged FE-MD method will be introduced. This discretization scheme is designed to be dynamic in nature so that the theory of the Embedding Atom Method can be implemented in the stiffness equations.

#### 4.3.1. Bridged Finite Element – Molecular Dynamics Discretization Scheme

To demonstrate the discretization scheme used in the bridged FE-MD software, the five node example used in the previous section was studied. However, in this case, the definition of an element is slightly altered. For this discretization scheme, rather than defining elements as two-node bodied, elements are defined as being centred at nodes. Figure 25 demonstrates an element established with node three as its centre.

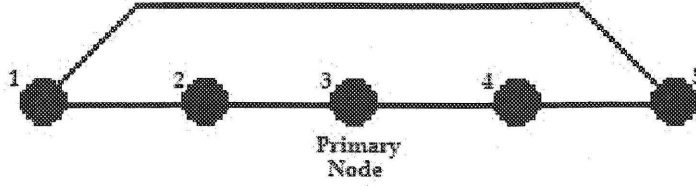


Figure 25 – Bridged FE-MD discretization scheme for element centred about Node 3

As shown for the element in Figure 25, node three is referred to as the primary node, as the elemental energy is merely the energy of only that node. The remaining four nodes are referred to as secondary nodes, as their atomic energies do not factor in the total energy of the element. For this discretization scheme, secondary nodes include all atoms that the forces between the two nodes are significant. This is discussed in section 4.3.2 when the EAM potential functions are included in the bridged FE-MD method. Due to this discretization method, the number of elements used for a system will be identical to the number of nodes or atoms in that system. Therefore, the stiffness matrices that are constructed using this method are more manageable than they would otherwise be.

For this discretization scheme, the potential energy of an element is looked at differently. Here, the traditional equation for potential energy of an element can be altered to calculate the potential energies at each node of the element. If one assumes that the total potential energy of the element can be attributed equally to each node, Equation 60 can be reorganized, as shown in Equation 70, to calculate the nodal potential energies.

$$U_{ij} = \frac{1}{2} k (\hat{d}_{ix} - \hat{d}_{jx})^2 \quad [60]$$

$$U_i = U_j = \frac{1}{4} k (\hat{d}_{ix} - \hat{d}_{jx})^2 \quad [70]$$

For each of the five elements in the example given, assuming that the sphere of influence for an element is the distance between three adjacent nodes, the elemental energies for each of the five elements in this system can be determined as shown in Equations 71–75.

$$U_1 = \frac{1}{4}k \left[ (\hat{d}_{2x} - \hat{d}_{1x})^2 + (\hat{d}_{3x} - \hat{d}_{1x})^2 \right] \quad [71]$$

$$U_2 = \frac{1}{4}k \left[ (\hat{d}_{1x} - \hat{d}_{2x})^2 + (\hat{d}_{3x} - \hat{d}_{2x})^2 + (\hat{d}_{4x} - \hat{d}_{2x})^2 \right] \quad [72]$$

$$U_3 = \frac{1}{4}k \left[ (\hat{d}_{1x} - \hat{d}_{3x})^2 + (\hat{d}_{2x} - \hat{d}_{3x})^2 + (\hat{d}_{4x} - \hat{d}_{3x})^2 + (\hat{d}_{5x} - \hat{d}_{3x})^2 \right] \quad [73]$$

$$U_4 = \frac{1}{4}k \left[ (\hat{d}_{2x} - \hat{d}_{4x})^2 + (\hat{d}_{3x} - \hat{d}_{4x})^2 + (\hat{d}_{5x} - \hat{d}_{4x})^2 \right] \quad [74]$$

$$U_5 = \frac{1}{4}k \left[ (\hat{d}_{3x} - \hat{d}_{5x})^2 + (\hat{d}_{4x} - \hat{d}_{5x})^2 \right] \quad [75]$$

Using the stiffness matrix shown in Equation 59, the stiffness matrices for each of the elements can be evaluated. For element 1, the stiffness matrix is formulated in Equation 76 and then evaluated for the five element example in Equation 77.

$$[K]_i = \begin{bmatrix} \frac{\partial^2 U_i}{\partial \hat{d}_{1x}^2} & \dots & \frac{\partial^2 U_i}{\partial \hat{d}_{nx} \partial \hat{d}_{1x}} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 U_i}{\partial \hat{d}_{1x} \partial \hat{d}_{nx}} & \dots & \frac{\partial^2 U_i}{\partial \hat{d}_{nx}^2} \end{bmatrix} \quad [59]$$

$$[K]_1 = \begin{bmatrix} \frac{\partial^2 U_1}{\partial \hat{d}_{1x}^2} & \frac{\partial^2 U_1}{\partial \hat{d}_{2x} \partial \hat{d}_{1x}} & \frac{\partial^2 U_1}{\partial \hat{d}_{3x} \partial \hat{d}_{1x}} & \frac{\partial^2 U_1}{\partial \hat{d}_{4x} \partial \hat{d}_{1x}} & \frac{\partial^2 U_1}{\partial \hat{d}_{5x} \partial \hat{d}_{1x}} \\ \frac{\partial^2 U_1}{\partial \hat{d}_{1x} \partial \hat{d}_{2x}} & \frac{\partial^2 U_1}{\partial \hat{d}_{2x}^2} & \frac{\partial^2 U_1}{\partial \hat{d}_{3x} \partial \hat{d}_{2x}} & \frac{\partial^2 U_1}{\partial \hat{d}_{4x} \partial \hat{d}_{2x}} & \frac{\partial^2 U_1}{\partial \hat{d}_{5x} \partial \hat{d}_{2x}} \\ \frac{\partial^2 U_1}{\partial \hat{d}_{1x} \partial \hat{d}_{3x}} & \frac{\partial^2 U_1}{\partial \hat{d}_{2x} \partial \hat{d}_{3x}} & \frac{\partial^2 U_1}{\partial \hat{d}_{3x}^2} & \frac{\partial^2 U_1}{\partial \hat{d}_{4x} \partial \hat{d}_{3x}} & \frac{\partial^2 U_1}{\partial \hat{d}_{5x} \partial \hat{d}_{3x}} \\ \frac{\partial^2 U_1}{\partial \hat{d}_{1x} \partial \hat{d}_{4x}} & \frac{\partial^2 U_1}{\partial \hat{d}_{2x} \partial \hat{d}_{4x}} & \frac{\partial^2 U_1}{\partial \hat{d}_{3x} \partial \hat{d}_{4x}} & \frac{\partial^2 U_1}{\partial \hat{d}_{4x}^2} & \frac{\partial^2 U_1}{\partial \hat{d}_{5x} \partial \hat{d}_{4x}} \\ \frac{\partial^2 U_{tot}}{\partial \hat{d}_{1x} \partial \hat{d}_{5x}} & \frac{\partial^2 U_1}{\partial \hat{d}_{2x} \partial \hat{d}_{5x}} & \frac{\partial^2 U_1}{\partial \hat{d}_{3x} \partial \hat{d}_{5x}} & \frac{\partial^2 U_1}{\partial \hat{d}_{4x} \partial \hat{d}_{5x}} & \frac{\partial^2 U_1}{\partial \hat{d}_{5x}^2} \end{bmatrix} \quad [76]$$



$$[K]_1 = \frac{k}{2} \begin{bmatrix} 2 & -1 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad [77]$$

Similarly, for the remaining four elements, the elemental stiffness matrices can be easily evaluated as shown in Equations 78–81.

$$[K]_2 = \frac{k}{2} \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 3 & -1 & -1 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad [78]$$

$$[K]_3 = \frac{k}{2} \begin{bmatrix} 1 & 0 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ -1 & -1 & 4 & -1 & -1 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 0 & 1 \end{bmatrix} \quad [79]$$

$$[K]_4 = \frac{k}{2} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & -1 & -1 & 3 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \quad [80]$$

$$[K]_5 = \frac{k}{2} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & -1 & -1 & 2 \end{bmatrix} \quad [81]$$

For the system, one can then determine the global stiffness matrix by summing the individual elemental matrices as shown in Equation 82.

$$[K]_{Total} = k \begin{bmatrix} 2 & -1 & -1 & 0 & 0 \\ -1 & 3 & -1 & -1 & 0 \\ -1 & -1 & 4 & -1 & -1 \\ 0 & -1 & -1 & 3 & -1 \\ 0 & 0 & -1 & -1 & 2 \end{bmatrix} \quad [82]$$

As hypothesized, the global stiffness matrices for each of the two discretization techniques are identical. Therefore, it has been confirmed that the bridged FE-MD discretization scheme indeed models the energy of the system effectively. This finding indicates that there is great benefit to the proposed discretization model. Due to the complexity of the EAM multi-body potentials, using this method enables the total energy acting on each node to be solved individually. In the coming section, the application of these EAM potentials to the stiffness equations will be discussed.

#### 4.3.2. Embedding Atom Method Potentials in Bridged Finite Elements – Molecular Dynamics

For the bridged FE-MD method, to calculate the forces on the atoms, the theory of molecular dynamics is still implemented. As done for the molecular dynamics simulations, EAM potentials are used to model the interatomic forces. To model these forces as elemental forces, Equation 7 determines the total force acting on each atom resulting from all the surrounding atoms within specified the distance limits.

$$U_{i,total} = \frac{1}{2} \sum_{\substack{j=1 \\ j \neq i}}^N \phi(r_{ij}) + \Psi \left\{ \sum_{\substack{j=1 \\ j \neq i}}^N \rho(r_{ij}) \right\} \quad [7]$$

For the total atomic force on a given atom, or node for the bridged FE-MD simulation, the derivative of the EAM potential energy equation is derived with respect to a given Cartesian coordinate  $p$ , as shown in Equation 83.

$$F_{p_i} = -\frac{\partial U_{i,EAM}}{\partial p_i} = -\sum_{\substack{j=1 \\ j \neq i}}^N \left[ \frac{d\phi(r_{ij})}{dr_{ij}} + \left( \frac{d\Psi}{d\rho_{tot,i}} + \frac{d\Psi}{d\rho_{tot,j}} \right) \frac{d\rho(r_{ij})}{dr_{ij}} \right] \frac{\partial r_{ij}}{\partial p_i} \quad [83]$$

The variable  $r_{ij}$  references the distance between two atoms,  $i$  and  $j$ , as calculated in Equation 84. The first derivative of Equation 84 with respect to each Cartesian coordinate is shown in Equations 85–87.

$$r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \quad [84]$$

$$\frac{\partial r_{ij}}{\partial x_i} = -\frac{\partial r_{ij}}{\partial x_j} = \frac{x_i - x_j}{r_{ij}} \quad [85]$$

$$\frac{\partial r_{ij}}{\partial y_i} = -\frac{\partial r_{ij}}{\partial y_j} = \frac{y_i - y_j}{r_{ij}} \quad [86]$$

$$\frac{\partial r_{ij}}{\partial z_i} = -\frac{\partial r_{ij}}{\partial z_j} = \frac{z_i - z_j}{r_{ij}} \quad [87]$$

Using the above equations, the EAM potential forces on each atom and in each Cartesian coordinate can be calculated and stored in the global force matrix. These calculations are completely independent of any boundary condition on the system. Therefore, the only unknown variables in the global stiffness are the nodal displacements. In the Sections 4.4 and 4.5, this theory is applied to MATLAB<sup>TM</sup> function FE\_MD\_V\_7\_00 and its nanoindentation adaptation MATLAB<sup>TM</sup> function NI\_FE\_MD\_V\_7\_00.

#### 4.4. Custom Finite Element – Molecular Dynamics Code FE\_MD\_V\_7\_00

In Narayan's research, the bridged FE-MD code FE\_MD\_V\_7\_00 was developed and three applications were analyzed using the software. These analyses were found to be extremely effective, as the results were very similar to that of the molecular dynamics simulation in a fraction of the simulation time.

#### 4.5. Custom Finite Element – Molecular Dynamics Nanoindentation Code NI\_FE\_MD\_V\_7\_00

For the purposes of performing a nanoindentation using the bridged FE-MD software, the code was altered. The major differences between MATLAB<sup>TM</sup> function NI\_FE\_MD\_V\_7\_00 and MATLAB<sup>TM</sup> function FE\_MD\_V\_7\_00 lie in the inclusion of the indenter design and the contact analysis functions. Despite these additions, the theory of the bridged FE-MD software as designed for MATLAB<sup>TM</sup> function FE\_MD\_V\_7\_00 is not adapted. A flowchart outlining the process undergone by MATLAB<sup>TM</sup> function NI\_FE\_MD\_V\_7\_00 can be found in Appendix B. However, the major tasks that are performed in MATLAB<sup>TM</sup> function NI\_FE\_MD\_V\_7\_00 are as follows:

- 1) Define the initial conditions for the simulation.
- 2) Read the positions and velocities of the substrate as defined in the function FCC\_Gen\_V\_7\_00, BCC\_Gen\_V\_7\_00, HCP\_Gen\_V\_7\_00 or Dia\_Gen\_V\_7\_00.
- 3) Calculate the distances between each atom and its neighbouring atoms and use that information to discretize the system.
- 4) Define the element stiffness matrices.
- 5) Assemble the global stiffness matrix by summing the individual element stiffness matrices.
- 6) Define the boundary conditions and apply them to the system.
- 7) If the indentation step has commenced, add the nodal displacements obtained from MATLAB<sup>TM</sup> function Contact\_V\_7\_00 to the current positions of the contacted atoms and apply boundary conditions on those nodes to prohibit them from being displaced.
- 8) Solve for all unknown nodal displacements and all corresponding nodal forces.
- 9) Write relevant information, such as positions, forces.
- 10) Repeat steps 3 through 9 for the required steps of the nanoindentation procedure.

Although the substrate is analyzed using a bridged FE-MD simulation, the indenter is not designed to be included in the FEM displacement calculations. Rather, MATLAB<sup>TM</sup> function Indenter\_Design\_V\_7\_00 creates an indenter in identical fashion as described in Chapter 2.

Since the indenter is designed as a rigid atomic system, there is no need to apply the FE-MD code to the indenter design. Therefore, as with the MD simulations, the only influence of the indenter on the substrate is modeled using MATLAB<sup>TM</sup> function `Contact_V_7_00`. However, for the bridged FE-MD simulations, the contact between the indenter and the substrate is dealt with in a slightly different manner than the method employed in Chapter 2.

While MD simulations are time-dependent, the bridged FE-MD code is completely time-independent. As a result, the contact analysis methodology was varied for MATLAB<sup>TM</sup> function `NI_FE_MD_V_7_00`. In molecular dynamics simulations, for each time step, the indenter is lowered by a predetermined indentation speed and the molecular dynamics simulation is performed to determine the positions and velocities for each of the atoms in the system. In this case, the indenter is still lowered in small increments, but the bridged FE-MD simulation is only performed at specified indentation depth intervals.

For each incremental change in indenter position, the corresponding displacements in the substrate are recorded and stored. At the conclusion of the indentation depth interval, the total nodal displacements resulting from the indentation are added to the current atomic positions of the substrate. Next, for each atom that is displaced by the indenter, nodal boundary conditions are applied to prevent them from being displaced from their new positions. At this time, bridged FE-MD simulation is performed and the displacements on the remaining atoms are calculated.

In this fashion, MATLAB<sup>TM</sup> function `NI_FE_MD_V_7_00` is able to simulate the effect of the indenter being lowered incrementally without requiring the bridged FE-MD simulation to be executed for each incremental displacement of the indenter. Consequently, the simulation can be executed in a reasonable time and is in fact still executed faster than in the MD simulations. For nanoindentations using this method, it is expected that there would be marginal differences in the results for the indenter increments used. However, the indentation depth interval size used is presumed to have much more bearing on the simulation results. These hypotheses are examined in detail in Section 5.2.

## 4.6. Chapter Summary

This chapter discussed a novel bridged FE-MD simulation tool for large atomic systems as presented in the research of Narayan. The theory surrounding the bridged FE-MD method was summarized and the bridged FE-MD code was adapted to create MATLAB<sup>TM</sup> function `NI_FE_MD_V_7_00`, which is specifically tailored towards performing nanoindentation simulations. In the next chapter, the aforementioned function will be used to perform nanoindentations and its effectiveness will be analyzed.

## 5. Bridged Finite Element – Molecular Dynamics Nanoindentation

---

### 5.1. Bridged Finite Element – Molecular Dynamics Nanoindentation Introduction

In the previous chapter, the theory characterizing the bridged FE-MD MATLAB™ software was described. In this chapter, the software is tested and large scale nanoindentation simulations are executed. As performed using molecular dynamics MATLAB™ code NI\_MD\_V\_7\_00, nanoindentations use rigid diamond indenters to indent copper substrates. Using each of the three designed indenter types; spherical, Berkovich and square pyramidal indenters, the sensitivity of the code is tested and larger scale simulations are performed. The bridged FE-MD simulation results are then analyzed and compared to MD results.

For each of the tests conducted using the bridged FE-MD code, prior to the commencement of the indentation procedure, an equilibrium step is first completed. While this was done for the MD simulations, using the bridged FE-MD method for this process is much more effective. For the bridged FE-MD method, an iterative method is executed until the maximum atomic displacement throughout the system is within required limits. For all the simulations in this research, for the system to be considered in a state of equilibrium, the maximum atomic displacement within the system must be less than 0.1 Å. As compared to the MD simulations when thousands of iterations are required for equilibrium, the bridged FE-MD simulations discussed in Sections 5.4–5.6 require only six iterations to reach equilibrium. Furthermore, for the bridged FE-MD simulations, the applied force prior to indentation is measured to be zero rather than the negative value that was calculated at the conclusion of the MD equilibrium step. Therefore, the bridged FE-MD method projects to be much more efficient than traditional MD simulations.

## 5.2. Bridged Finite Element –Molecular Dynamics Sensitivity Analysis

Prior to nanoindentation simulations being executed, the sensitivity of MATLAB<sup>TM</sup> function NI\_FE\_MD\_V\_7\_00 is analyzed. In Sections 5.2.1–5.2.3, the sensitivity of the code is tested by analyzing the material response to indenter increment size changes and indentation depth size interval changes. Next, the atomic displacements of the system during nanoindentation are analyzed as previously tested for the MD analysis and the two results are compared.

### 5.2.1. Indenter Increment Size Sensitivity Analysis

As described in Section 4.5, since the bridged FE-MD method is time independent, to simulate the effect of lowering the indenter into the substrate, the indenter is moved incrementally into the substrate and the contact analysis is performed for each increment. It is expected that the increment size will not have a great effect on the results obtained from the simulation. In order to test this hypothesis, the indentation depth interval size is maintained constant at 0.1 Å while the number of increments per indentation depth interval is varied from ten increments to one hundred increments. For these simulations, the following initial conditions are used:

Indenter Type	Spherical
Temperature	293 K
Substrate Size	4 by 4 by 4 unit cells (14.46 Å)
Indenter Diameter	2 unit cells (7.134 Å)
Maximum Indentation Depth	5Å
Clamping	Absolute clamping on all faces except the in the positive z-direction

Simulations using the above conditions are performed for five different increment sizes. The recorded forces are graphed over indentation depth and are shown in Figure 26. In the legend used in Figure 26, the increment size is described based on the number of increments that is used for the contact analysis per FEM interval.



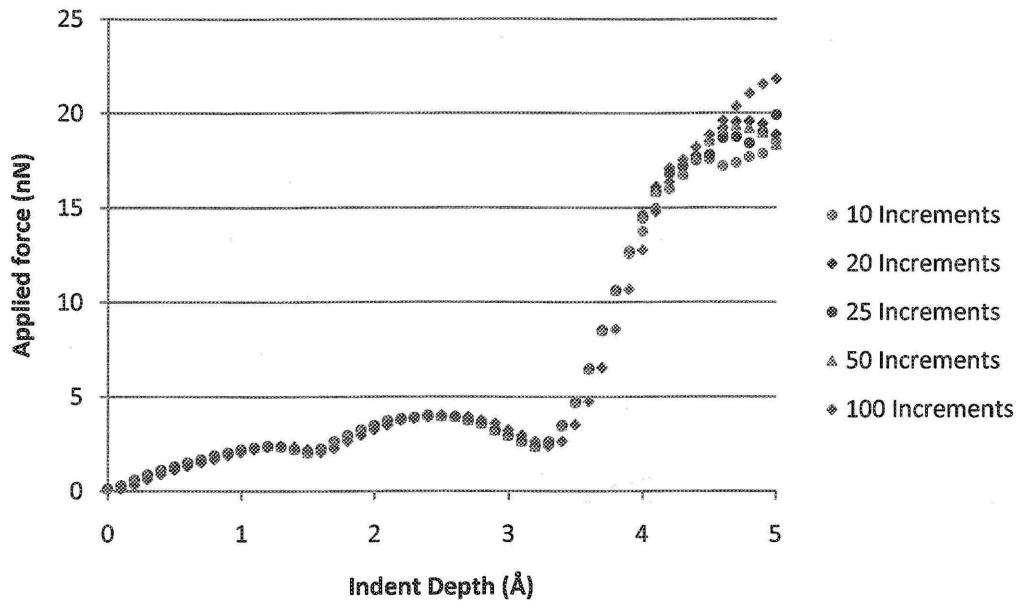


Figure 26 –Increment Size Sensitivity Test Applied Force Results

As shown in Figure 26, the applied force results are not greatly affected by the number of increments used per interval size. However, for the case when only ten increments are used per FEM interval, there are slight changes to the force results that indicate that the simulation is unable to effectively capture the true nature of the system. Additionally, when 100 increments are used per FEM interval, the applied force values differ slightly from the remaining increment size tests. In this case, the forces are slightly greater at the conclusion of the nanoindentation than the remaining increment sizes. However, it is found that due to the algorithm used for MATLAB<sup>TM</sup> function `Contact_V_7_00`, the atoms are being pushed too quickly away from the indenter and the applied forces are again not true to the behaviour of the material. As a result, it is determined that each of the other three increment sizes would be adequate to perform the simulations effectively. For large scale simulations, having fewer increments per indentation depth interval could pose problems when many atoms are being contacted by the indenter simultaneously. Therefore, in this research, the optimal increment size was chosen to be fifty increments per FEM interval (each increment lowers the indenter by 0.002 Å), as this ensures that the behaviour of the material is captured effectively even for large scale simulations.

### 5.2.2. Indentation Depth Interval Size Sensitivity Analysis

Once the increment size sensitivity analysis is completed, it is necessary to investigate the indentation depth interval size to ensure that the interval size used will effectively capture the material response to the nanoindentation. Using the same initial conditions as used in the increment size sensitivity analysis, interval sizes ranging from 0.05 Å to 0.2 Å are studied. For each of these simulations, the increment size is maintained at 0.002 Å, as determined in the previous section. The results from the interval size tests are graphed in Figure 27.

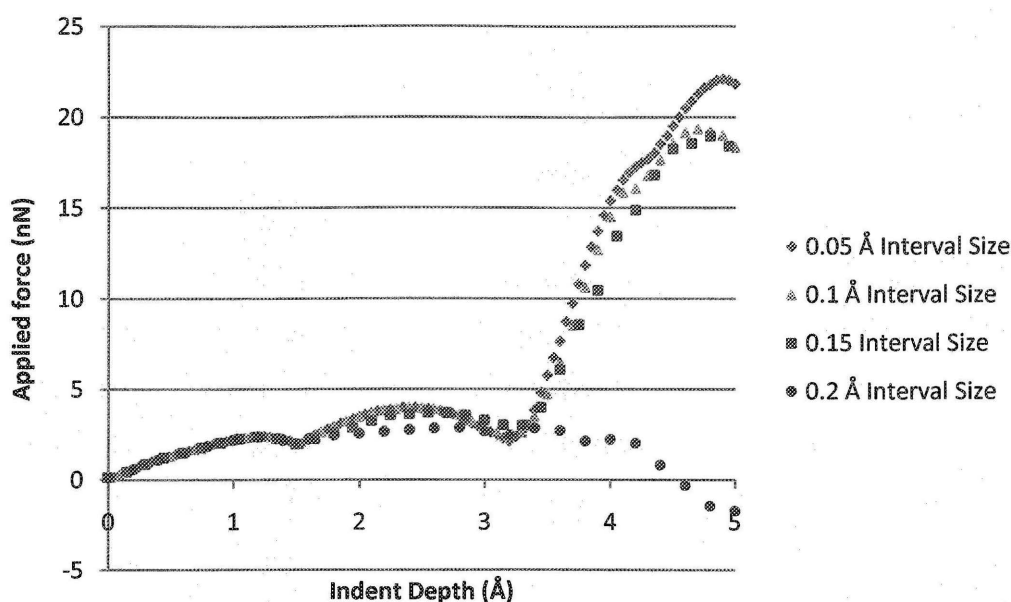


Figure 27 –Indentation depth Interval Size Sensitivity Analysis Applied Force Results

Figure 27 illustrates that the interval size used in the analysis drastically affects the bridged FE-MD simulation results. For interval sizes of 0.05 Å and 0.1 Å, there is little change in applied force between the two simulations. For the simulations using interval sizes of 0.15 Å and 0.2 Å, however, there are significant changes from the smaller interval sizes. This is particularly true for the 0.2 Å interval size as the system is unable to respond to the large atomic displacements that result from the contact with the indenter. In fact, it is shown in Figure 27 that the applied forces are calculated to be negative values when the indentation depth reaches approximately 4.5 Å. This occurs because atoms are pushed together so quickly that the distance between them is less

than the cut-off distance as specified by the EAM potential function. As a result, the code is unable to calculate the forces between the atoms and the total applied force on the system is skewed. Therefore, interval sizes of 0.05 Å and 0.1 Å could both be used to provide accurate results in nanoindentation simulations. However, the optimal size for the purposes of this research is an interval size of 0.1 Å, as the minuscule added accuracy of the 0.05 Å interval size does not warrant doubling the number of nanoindentation steps, and therefore, double the time required for the simulations to be executed.

### 5.2.3. Bridged Finite Element –Molecular Dynamics Displacement Analysis

The final sensitivity analysis of the bridged FE-MD method investigates the displacement of atomic positions throughout the nanoindentation procedure. This sensitivity test compares the differences between the displacements recorded in the MD and bridged FE-MD simulation techniques. For the simulation, just as was performed for the MD simulation, two equilibrium steps are investigated in addition to the first equilibrium step and the nanoindentation. A second equilibrium step is executed once the indenter reaches its maximum indentation depth, followed by a third equilibrium step that is executed after the indenter is removed. The following initial conditions are used in the simulation as was used in Section 3.2.3:

Indenter Type	Spherical
Temperature	293 K
Substrate Size	4 by 4 by 4 unit cells (14.46 Å)
Indenter Diameter	2 unit cells (7.134 Å)
Maximum Indentation depth	5Å
Clamping	Absolute clamping on all faces except the in the positive z-direction

As in the MD displacement analysis, for the bridged FE-MD simulation, only specific atoms are highlighted. While a more detailed displacement analysis could be performed using more of the substrate atoms, the goals of this analysis are to compare the two simulation methods and to ensure that the substrate does not merely return to its original form once the indenter is removed from the system. In Tables 6-9, the results from the displacement sensitivity analysis is shown. In addition to merely displaying those atoms that present large displacements, some atoms that were

discussed in Section 3.2.3 will also be included. This inclusion will allow the two methods to be better compared.

Atom #	Starting Positions (Å)			After Equilibrium #1 (Å)			% Difference
	X-Position	Y-Position	Z-Position	X-Position	Y-Position	Z-Position	
40	3.6150	7.2300	14.4600	3.5407	7.2300	14.3007	-1.1136
60	7.2300	3.6150	14.4600	7.2300	3.5407	14.3007	-1.1136
65	7.2300	7.2300	14.4600	7.2300	7.2300	14.2371	-1.5653
70	7.2300	10.8450	14.4600	7.2300	10.9193	14.3007	-1.1136
90	10.8450	7.2300	14.4600	10.9193	7.2300	14.3007	-1.1136
315	5.4225	5.4225	14.4600	5.3841	5.3841	14.2651	-1.3661
320	5.4225	9.0375	14.4600	5.3841	9.0759	14.2651	-1.3661
335	9.0375	5.4225	14.4600	9.0759	5.3841	14.2651	-1.3661
340	9.0375	9.0375	14.4600	9.0759	9.0759	14.2651	-1.3661
153	3.6150	9.0375	12.6525	3.5073	9.0948	12.6424	-0.0796
165	7.2300	5.4225	12.6525	7.2300	5.3699	12.5769	-0.6015
169	7.2300	9.0375	12.6525	7.2300	9.0901	12.5769	-0.6015
181	10.8450	5.4225	12.6525	10.9527	5.3652	12.6424	-0.0796
237	5.4225	7.2300	12.6525	5.3699	7.2300	12.5769	-0.6015
241	5.4225	10.8450	12.6525	5.3652	10.9527	12.6424	-0.0796
253	9.0375	3.6150	12.6525	9.0948	3.5073	12.6424	-0.0796
257	9.0375	7.2300	12.6525	9.0901	7.2300	12.5769	-0.6015
64	7.2300	7.2300	10.8450	7.2300	7.2300	10.8317	-0.1232
314	5.4225	5.4225	10.8450	5.3648	5.3648	10.8535	0.0783
339	9.0375	9.0375	10.8450	9.0952	9.0952	10.8535	0.0783
63	7.2300	7.2300	7.2300	7.2300	7.2300	7.2230	-0.0965
62	7.2300	7.2300	3.6150	7.2300	7.2300	3.5827	-0.9003

Table 6 – Atomic Positions at Equilibrium Step #1 from Bridged FE-MD Simulations

Table 6 lists starting atomic positions and corresponding equilibrium atomic positions prior to the start of the indentation. In this table, only the percentage difference of the displacement in the z-direction is calculated. As seen in Table 6, there are no significant atomic displacements for the system to reach a state of equilibrium. The most significant displacements are located on the face that coincides with the maximum z-position values where the displacements are slightly larger than one percent. These findings are consistent with the results of the MD displacement sensitivity analysis.

Atom #	Starting Positions (Å)			After Indentation (Å)			% Difference
	X-Position	Y-Position	Z-Position	X-Position	Y-Position	Z-Position	
40	3.6150	7.2300	14.4600	3.3289	7.3886	12.6988	-11.2016
60	7.2300	3.6150	14.4600	7.3886	3.3289	12.6988	-11.2016
65	7.2300	7.2300	14.4600	7.2300	7.2300	9.2371	-36.1194
70	7.2300	10.8450	14.4600	7.0714	11.1311	12.6988	-11.2016
90	10.8450	7.2300	14.4600	11.1311	7.0714	12.6988	-11.2016
315	5.4225	5.4225	14.4600	5.3592	5.3592	10.6672	-26.2297
320	5.4225	9.0375	14.4600	5.0340	9.4260	12.8889	-9.6476
335	9.0375	5.4225	14.4600	9.4260	5.0340	12.8889	-9.6476
340	9.0375	9.0375	14.4600	9.1008	9.1008	10.6672	-26.2297
153	3.6150	9.0375	12.6525	2.9066	9.5392	12.3948	-2.0370
165	7.2300	5.4225	12.6525	7.6156	5.1833	11.0667	-12.5332
169	7.2300	9.0375	12.6525	6.8444	9.2767	11.0667	-12.5332
181	10.8450	5.4225	12.6525	11.5534	4.9208	12.3948	-2.0370
237	5.4225	7.2300	12.6525	5.1833	7.6156	11.0667	-12.5332
241	5.4225	10.8450	12.6525	4.9208	11.5534	12.3948	-2.0370
253	9.0375	3.6150	12.6525	9.5392	2.9066	12.3948	-2.0370
257	9.0375	7.2300	12.6525	9.2767	6.8444	11.0667	-12.5332
64	7.2300	7.2300	10.8450	7.2300	7.2300	7.0480	-35.0114
314	5.4225	5.4225	10.8450	5.5774	5.5774	8.3577	-22.9352
339	9.0375	9.0375	10.8450	8.8826	8.8826	8.3577	-22.9352
63	7.2300	7.2300	7.2300	7.2300	7.2300	4.8108	-33.4612
62	7.2300	7.2300	3.6150	7.2300	7.2300	2.6531	-25.9488

**Table 7 – Atomic Positions after Indentation Step from Bridged FE-MD Simulations**

In Table 7, the atomic positions once the indentation has been completed are presented. At the conclusion of the indentation step, the atomic displacements in the z-direction for many of the atoms are quite substantial. As found in MD displacement sensitivity test, there are also significant displacements calculated in atoms that are compressed due to the indentation. Interestingly, atoms 315 and 340 experience displacements of 26.2%, while those atoms are displaced with atoms 320 and 335 in the MD simulation. In the bridged FE-MD simulations, atoms 320 and 335 are also displaced in the x and y-directions. It is found that these atoms are contacted by the indenter at more of an angle than was the case for atoms 315 and 340. Due to the equilibrium steps performed for each interval during the FE-MD simulation, the system responded to the indentation faster and therefore, atoms 315 and 340 are marginally displaced prior to being contacted by indenter atoms.



Additionally, the displacements in atoms 314, 339, and 62 differ significantly from the MD simulation. While in the MD simulations, atoms 314 and 339 displaced only 2.2% in the negative z-direction at the conclusion of the nanoindentation, displacements using the bridged FE-MD method are calculated as 22.9% in the negative z-direction. Atom 62, in contrast, is not displaced for the bridged FE-MD simulations as significantly as observed for the MD simulations. These large differences can once again be attributed to the equilibrium steps implemented by the bridged FE-MD method.

Atom #	Starting Positions (Å)			After Equilibrium #2 (Å)			% Difference
	X-Position	Y-Position	Z-Position	X-Position	Y-Position	Z-Position	
40	3.6150	7.2300	14.4600	3.3289	7.3886	12.6988	-11.2016
60	7.2300	3.6150	14.4600	7.3886	3.3289	12.6988	-11.2016
65	7.2300	7.2300	14.4600	7.2300	7.2300	9.2371	-36.1194
70	7.2300	10.8450	14.4600	7.0714	11.1311	12.6988	-11.2016
90	10.8450	7.2300	14.4600	11.1311	7.0714	12.6988	-11.2016
315	5.4225	5.4225	14.4600	5.3592	5.3592	10.6672	-26.2297
320	5.4225	9.0375	14.4600	5.0340	9.4260	12.8889	-9.6476
335	9.0375	5.4225	14.4600	9.4260	5.0340	12.8889	-9.6476
340	9.0375	9.0375	14.4600	9.1008	9.1008	10.6672	-26.2297
153	3.6150	9.0375	12.6525	2.9046	9.5373	12.3955	-2.0314
165	7.2300	5.4225	12.6525	7.6156	5.1833	11.0667	-12.5332
169	7.2300	9.0375	12.6525	6.8444	9.2767	11.0667	-12.5332
181	10.8450	5.4225	12.6525	11.5554	4.9227	12.3955	-2.0314
237	5.4225	7.2300	12.6525	5.1833	7.6156	11.0667	-12.5332
241	5.4225	10.8450	12.6525	4.9227	11.5554	12.3955	-2.0314
253	9.0375	3.6150	12.6525	9.5373	2.9046	12.3955	-2.0314
257	9.0375	7.2300	12.6525	9.2767	6.8444	11.0667	-12.5332
64	7.2300	7.2300	10.8450	7.2300	7.2300	6.9977	-35.4753
314	5.4225	5.4225	10.8450	5.5775	5.5775	8.3315	-23.1763
339	9.0375	9.0375	10.8450	8.8825	8.8825	8.3315	-23.1763
63	7.2300	7.2300	7.2300	7.2300	7.2300	4.7426	-34.4045
62	7.2300	7.2300	3.6150	7.2300	7.2300	2.5836	-27.8870

Table 8 – Atomic Positions at Equilibrium Step #2 from Bridged FE-MD Simulations

Table 8 shows the atomic positions at the conclusion of the second equilibrium step. As in the MD simulations, the indenter is held in place during this step and subsequently, the atoms directly contacting the indenter are unable to move. For the remaining atoms, there is almost no

displacement within the system. Due to the methodology of the bridged FE-MD code, the system is already in a state of quasi-equilibrium at the conclusion of the indentation. Therefore, for the equilibrium step to be completed, only one additional simulation step is required. As expected, at the conclusion of this additional step there is no significant change in the atomic positions throughout the system.

Atom #	Starting Positions (Å)			After Equilibrium #3 (Å)			% Difference
	X-Position	Y-Position	Z-Position	X-Position	Y-Position	Z-Position	
40	3.6150	7.2300	14.4600	3.6782	7.5839	13.8459	-3.1808
60	7.2300	3.6150	14.4600	7.5839	3.6782	13.8459	-3.1808
65	7.2300	7.2300	14.4600	7.2300	7.2300	10.7020	-25.9886
70	7.2300	10.8450	14.4600	6.8761	10.7818	13.8459	-3.1808
90	10.8450	7.2300	14.4600	10.7818	6.8761	13.8459	-3.1808
315	5.4225	5.4225	14.4600	5.0184	5.0184	12.1081	-16.2649
320	5.4225	9.0375	14.4600	4.4173	10.0427	12.9867	-8.9618
335	9.0375	5.4225	14.4600	10.0427	4.4173	12.9867	-8.9618
340	9.0375	9.0375	14.4600	9.4416	9.4416	12.1081	-16.2649
153	3.6150	9.0375	12.6525	2.0498	9.2095	14.0699	11.2029
165	7.2300	5.4225	12.6525	7.3158	5.4944	12.4538	-1.5701
169	7.2300	9.0375	12.6525	7.1442	8.9656	12.4538	-1.5701
181	10.8450	5.4225	12.6525	12.4102	5.2505	14.0699	11.2029
237	5.4225	7.2300	12.6525	5.4944	7.3158	12.4538	-1.5701
241	5.4225	10.8450	12.6525	5.2505	12.4102	14.0699	11.2029
253	9.0375	3.6150	12.6525	9.2095	2.0498	14.0699	11.2029
257	9.0375	7.2300	12.6525	8.9656	7.1442	12.4538	-1.5701
64	7.2300	7.2300	10.8450	7.2300	7.2300	8.3538	-22.9708
314	5.4225	5.4225	10.8450	5.6022	5.6022	10.0283	-7.5303
339	9.0375	9.0375	10.8450	8.8578	8.8578	10.0283	-7.5303
63	7.2300	7.2300	7.2300	7.2300	7.2300	6.1495	-14.9453
62	7.2300	7.2300	3.6150	7.2300	7.2300	3.6088	-0.1705

Table 9 – Atomic Positions at Equilibrium Step #3 from Bridged FE-MD Simulations

The atomic positions after the third equilibrium step is completed are shown in Table 9. As was case for the MD simulation, the system again expands once the indenter is removed. For the bridged FE-MD simulation, the removal of the indenter has a greater effect on the system. For atoms 62–65, the movement after the indenter is removed is drastically greater. Atom 62, after being indented by 27.9%, returns to a position only 0.17% less than its starting position.

Similarly, the displacements of atoms 63–65 using the bridged FE-MD technique are greater than was found for the MD simulations. If additional steps were to be executed for the MD simulations, the atoms would eventually exhibit the same displacements; however, for the bridged FE-MD this process takes only a few steps.

Atoms 153, 181, 241 and 253 are subjected to unique reactions due to the removal of the indenter. As shown in Table 9, these atoms are displaced by 11.2% in the positive z-direction. It is also seen that each of the atoms experience significant displacements either in the x or y-direction. It is evident that as the system expands, these four atoms are pushed too close to other atoms. As a result, the atomic forces acting on them are quite large and the atoms are pushed further in the z-direction and above their initial positions.

For the remaining atoms, the displacements observed at the conclusion of the indentation step are reduced as the system expands. While similar to the results of the MD simulations, the atomic displacements at the end of the final equilibrium step are still larger for the bridged FE-MD simulation. Since the MD simulation is not as effective at modeling the system as it undergoes the indentation, the MD simulation never actually reaches a state of equilibrium. Therefore, when the indenter is removed, the atomic positions of all atoms except atoms 62–65 are returned to positions very similar to their initial positions. Using the bridged FE-MD method, as expected, the indentation leaves more of a long lasting effect on the system. As such, it is evident that the bridged FE-MD method is more effective than the MD simulation method for nanoindentation experimentation. In the coming sections, this point will be further illustrated as full-scale nanoindentation simulations are performed and their results are compared to the MD simulations results.

### **5.3. Bridged Finite Element –Molecular Dynamics Simulation Conditions**

For the bridged FE-MD simulations, to intention was to perform simulations using the identical conditions as used for the MD simulations. However, because of computing constraints, modeling 10 by 10 by 10 unit cell sized copper substrates with PBC was not possible. Instead,



smaller copper substrates sized 8 by 8 by 6 unit cells were analyzed without PBC. While performing simulations using different substrate sizes for the two simulation techniques makes comparison more difficult, the purpose of this research is not to determine the exact hardness values for a copper nanomaterial. Rather, it aims to determine the effectiveness of the bridged FE-MD simulation tool in performing nanoindentation experiments. In Figures 28–33, the bridged FE-MD simulations results are graphed. For each of the simulations, in addition to plotting the bridged FE-MD results, the MD results for an indentation speed of 10 m/s are also plotted on the same graph. At that speed, the system is found to be capable of responding to the indentation most effectively. As such, comparison between the trends of each of the simulation methodologies is possible.

For each of these tests, the simulation temperature used was 293 K, and absolute clamping is used on all faces except in the positive z-direction. As in the MD simulations, testing is performed using each of the three designed indenters: spherical, Berkovich, and square pyramidal. For the MD simulations, four different indenter sizes were used. Due to the smaller substrate size used for the bridged FE-MD method, simulations using the two larger indenter sizes are not possible. Using these larger indenter sizes, contact would occur almost adjacent to the clamped edges of the substrate and therefore, the clamping would interfere with the displacements due to contact and the simulation results would be inaccurate.

5.4. Bridged Finite Element –Molecular Dynamics Spherical Indenter Results

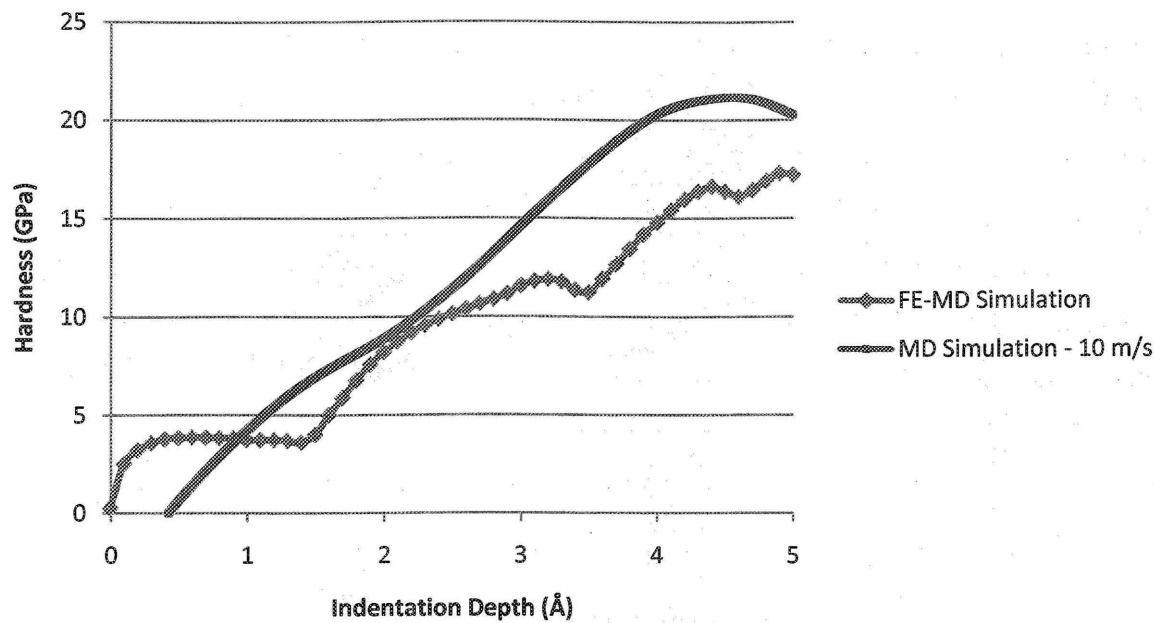


Figure 28 – FE-MD Results using Spherical Indenter with Diameter of Four Unit Cells

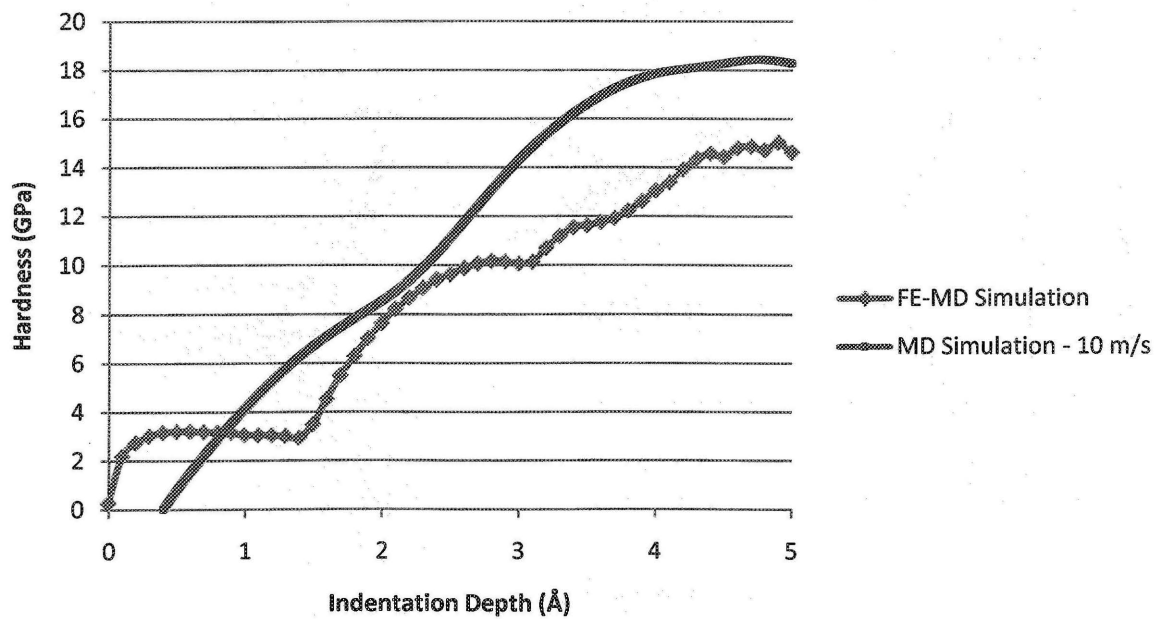


Figure 29 – FE-MD Results using Spherical Indenter with Diameter of Five Unit Cells

5.5. Bridged Finite Element –Molecular Dynamics Berkovich Indenter Results

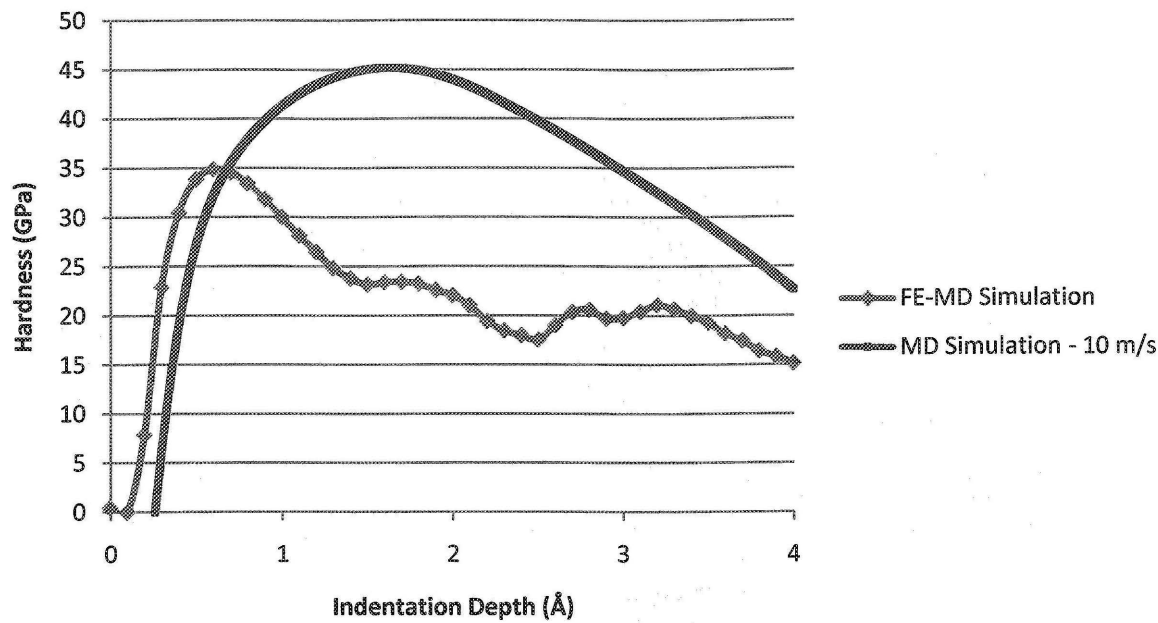


Figure 30 – FE-MD Results using Berkovich Indenter with Base of Length Four Unit Cells

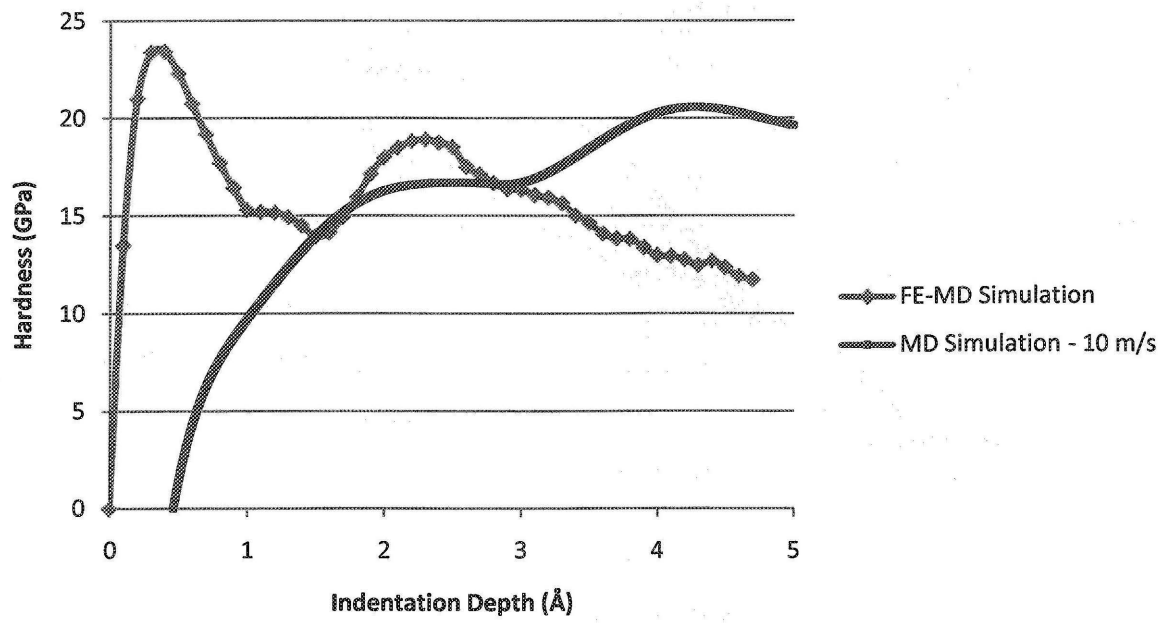


Figure 31 – FE-MD Results using Berkovich Indenter with Base of Length Five Unit Cells

## 5.6. Bridged Finite Element –Molecular Dynamics Square Pyramidal Indenter Results

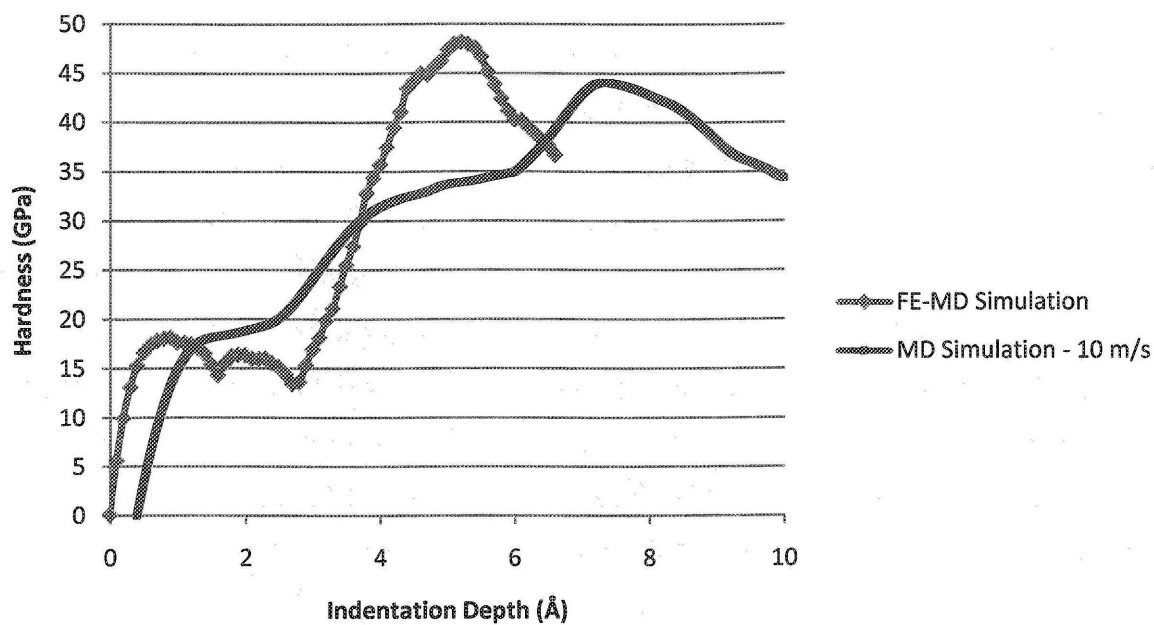


Figure 32 – FE-MD Results using Pyramidal Indenter with Base Four by Four Unit Cells

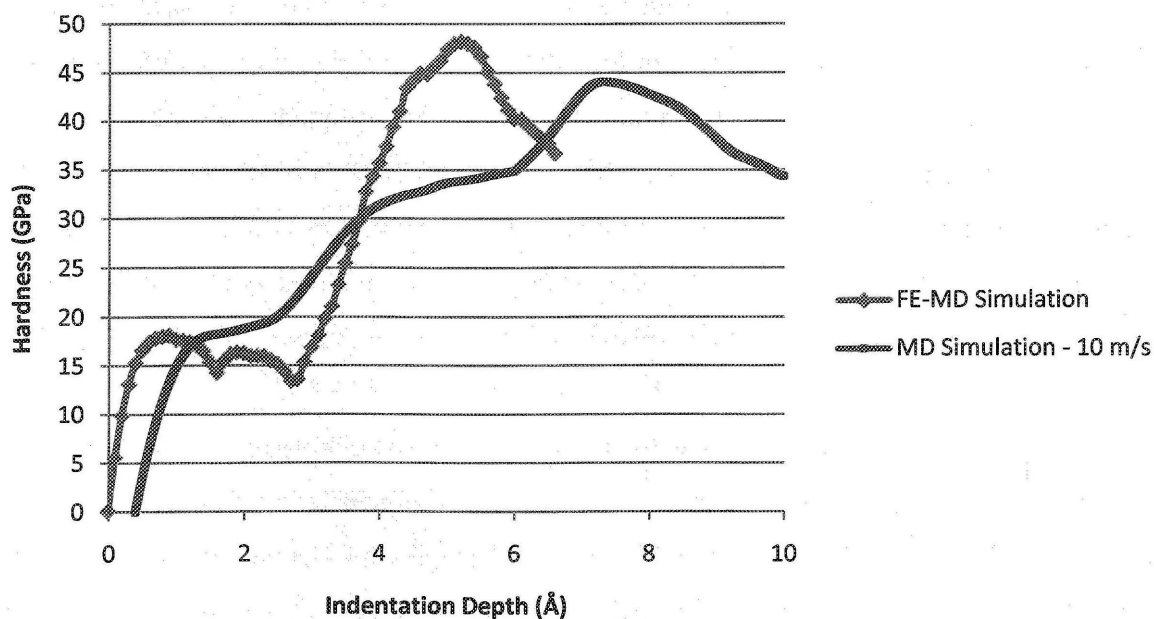


Figure 33 – FE-MD Results using Pyramidal Indenter with Base Five by Five Unit Cells

## **5.7. Discussion and Comparison between Bridged Finite Element – Molecular Dynamics Simulations and Molecular Dynamics Simulations**

For the spherical indenter simulations, as shown in Figures 28–29, the bridged FE-MD results appear to follow the same trends as were found in the MD simulations. Due to the smaller substrate size for the bridged FE-MD simulations, the hardness values for the bridged FE-MD simulations are slightly less than those of the MD simulations. Interestingly, in both cases, the hardness values level off at approximately the same indentation depth. Additionally, the bridged FE-MD simulations exhibit the “indenter size effect,” as was presented for the MD simulations. Furthermore, when comparing the results to those of Saraev and Miller, it is evident that the bridged FE-MD produces hardness values that more closely resemble Saraev and Miller’s results. Based on this information, it is evident that the bridged FE-MD method is extremely successful for simulations using the spherical indenters.

For the simulations using Berkovich indenters, the differences between the bridged FE-MD and MD simulation results are more pronounced. While the MD simulations differ greatly as the indenter size is altered, the bridged FE-MD simulation results follow the same trends despite indenter size changes. As shown in Figures 30–31, in both of the two hardness plots, the maximum hardness value occurs within an indentation depth of 1 Å. This peak in the hardness plot occurs as a large number of atoms contact the substrate very quickly during the indentation step of the simulation. Furthermore, for the bridged FE-MD simulations using Berkovich indenters, again the “indenter size effect” is exhibited in the results. Based on the information shown in Figures 30–31, it appears that the bridged FE-MD method is able to determine the hardness of a material much more effectively than the MD simulations.

The bridged FE-MD technique produces more uniform results than the MD method for simulations using square pyramidal indenters as well. As shown in Figures 32–33, in the MD simulations the hardness plots differ significantly for changes in indenter size, while the bridged FE-MD simulations produce almost identical plots. Due to the geometry of the square pyramidal indenters, the two indenters used in the bridged FE-MD simulations are identical except for the additional atoms included in the base of the larger indenter. Therefore, it is expected that the

results will not differ greatly. Due to the size of the square pyramidal indenter and the substrate, the system is prohibited from responding properly to the nanoindentation. As a result, the hardness results obtained from the square pyramidal indentation in this research are not indicative of the actual hardness values that would be obtained from a simulation using a large atomic system.

Overall, the bridged FE-MD simulation results are quite favourable. Despite the smaller substrate size used in the bridged FE-MD simulations, the results obtained in the bridged FE-MD simulations are in fact much better than in the MD simulations. While the MD results all require significant indentation to occur before the applied forces are recorded as positive values, the iterative technique employed by the bridged FE-MD MATLAB<sup>TM</sup> function NI\_FE\_MD\_V\_7\_00 ensures that the recorded applied forces on the system are very close to zero prior to the commencement of the nanoindentation step. As a result, positive applied force and hardness results are recorded from the very start of the nanoindentation procedure.

While the MD simulations often were unable to capture the behaviour of the system due to the speed of nanoindentation, this is not an issue with the bridged FE-MD simulations. This is especially seen for the simulations using Berkovich indenters. Due to the smaller height of the Berkovich indenter, there were great difficulties in many of the MD simulations performed in capturing the behaviour of the system as a result of the little time the system has to respond to the indenter. As shown in Figures 28–33, the system does not appear to have any such difficulties when the bridged FE-MD method is implemented.

Besides the increased accuracy of the nanoindentation simulation results using the bridged FE-MD method, significantly less simulation time is required for the simulations to be completed. For example, while more than twenty-four hours is required to execute simulations using MD simulations with an indentation speed of 10 m/s, the bridged FE-MD method simulations were completed in ten hours. While a portion of this difference stems from the smaller substrate size used for the bridged FE-MD simulations, the bridged FE-MD technique still saves several hours of simulation time compared to traditional MD simulation techniques.

## 5.8. Chapter Summary

In this chapter, the nanoindentation bridged FE-MD MATLAB<sup>TM</sup> function NI\_FE\_MD\_V\_7\_00 was first analyzed through a variety of sensitivity tests. These tests were extremely effective as the optimal initial conditions with which to run the simulations were determined and the atomic displacements obtained from a bridged FE-MD nanoindentation simulation were found to be similar to those of the MD simulation. Next, larger scale simulations were performed using spherical, Berkovich, and square pyramidal indenters. These simulations were extremely successful as the results were found to be an improvement upon the MD simulation results.

## 6. Concluding Remarks

---

In this research, computational tools were implemented to study the behaviour of atomic systems undergoing nanoindentations. Using an adaptation of a pre-existing custom-made molecular dynamics code, nanoindentations were performed to analyze the behaviour of copper nanomaterials under a variety of conditions. The interatomic potential forces were modeled using the embedding atom potential function. Copper substrates were analyzed using three different rigid diamond indenter types, each of which was designed in four different sizes, and at three indentation speeds. Overall, these simulations were successful but at the faster indentation speeds, in particular, the system was not always able to respond quickly to the indentations.

To more efficiently model the behaviour of the system during nanoindentations, a bridged FE-MD code adapted for the needs of this research. This bridged FE-MD method, proposed by Narayan, combines the theories of MD and FEM to allow the equations of motions found in MD to be solved using traditional FEM theory. This method introduces a discretization scheme that is uniquely suited to modeling the interatomic forces in large atomic systems and enables the atomic displacements to be solved efficiently.

Using this bridged FE-MD simulation, large scale nanoindentation simulations were performed. These simulations are found to be more effective and less time consuming than the traditional MD simulation technique. MD simulations were found to have great difficulties in reaching a state of equilibrium, resulting in less accuracy within the results. The bridged FE-MD simulation method, in contrast, reaches a state of quasi-equilibrium after each indentation depth interval and therefore, the simulations produce more accurate nanohardness results. While the bridged FE-MD simulations were found to be extremely effective, further testing must be done. Due to the computational constraints, the simulations investigated using the bridged FE-MD method were performed mainly to assess its effectiveness. Therefore, it will be necessary to conduct larger scale simulations so that the results can be used commercially in the design process.



While this research implements the bridged FE-MD method for a very specific purpose, future research possibilities using the bridged FE-MD nanoindentation software are vast. These possible avenues include but are not limited to: studying the effects of nanoindentation on alloys, and imperfect crystal structures; determining the optimal size of coatings on a material to ensure the greatest hardness values; analyzing the hardness of a material undergoing significant stress or experiencing fatigue or failure; determining the effect of nanoindentation experimentation on the fatigue life of a nanomaterial; etc. To ensure that the bridged FE-MD software can be used to explore other materials such as non-metals, powders, and alloys, additional potential functions must first be incorporated within the bridged FE-MD software. As this research group progresses, these adaptations to the bridged FE-MD software will be implemented to provide researchers with a comprehensive tool with which they will be able to significantly impact the field of nanomaterials.

## References

---

- [1] Ratto, F., Matteini, P., Rossi, F., Menabuoni, L., Tiwari, N., Kulkarni, S. K., et al. (2009). Photothermal effects in connective tissues mediated by laser-activated gold nanorods. *Nanomedicine: Nanotechnology Biology and Medicine* (5) , 143-151.
- [2] Inoue, K.-i., Takano, H., Yanagisawa, R., Koike, E., & Shimada, A. (2009). Size effects of latex nanomaterials on lung inflammation in mice. *Toxicology and Applied Pharmacology* (234) , 68-76.
- [3] Kim, J., Yun, J.-H., & Han, C.-S. (2009). Nanomaterial-embedded gas sensor fabrication. *Current Applied Physics* (9) , e38-e41.
- [4] Li, Q., Mahendra, S., Lyon, D. Y., Brunet, L., Liga, M. V., Li, D., et al. (2008). Antimicrobial nanomaterials for water disinfection and microbial control: Potential applications and implications. *Water Research* (42) , 4591-4602.
- [5] Schulte, J. (2005). *Nanotechnology, global strategies, industry trends and applications*. West Sussex, England: Jon Wiley and Sons, Ltd.
- [6] Jeng, Y.-R., Tsai, P.-C., & Fang, T.-H. (2005). Molecular-dynamics studies of bending mechanical properties of empty and C60-filled carbon nanotubes under nanoindentation. *The Journal of Chemical Physics* (122) .
- [7] Zeng, F., & Sun, Y. (2006). Molecular dynamics simulations of nanoindentation of porous materials. *Computational Methods* , 1655–1660.
- [8] Noreyan, A., Amarb, J., & Marinescu, I. (2005). Molecular dynamics simulations of nanoindentation of beta-SiC with diamond indenter. *Materials Science and Engineering B* (117) , 235–240.
- [9] Kalia, R. K., Nakano, A., & Vashishta, P. (2007). A molecular dynamics study of nanoindentation of amorphous silicon carbide. *Journal of Applied Physics* (102) .
- [10] Huo, D., Liang, Y., & Cheng, K. (2006). An investigation of nanoindentation tests on the single crystal copper thin film via an atomic force microscope and molecular dynamics simulation. *Journal of Mechanical Engineering Science* (221) , 259-266.
- [11] Narayan, K., Behdinan, K., & Fawaz, Z. (2007). An engineering-oriented embedded-atom-method potential fitting procedure for pure fcc and bcc metals. *Journal of Materials Processing Technology* (182) , 387-397.
- [12] Andrade, J. E., & Tu, X. (2009). Multiscale framework for behavior prediction in granular media. *Mechanics of Materials* (41) , 652–669.
- [13] Hamdi, M., & Ferreira, A. (2009). Multiscale design and modeling of protein-based nanomechanisms for nanorobotics. *The International Journal of Robotics Research* (28) , 436-449.

- [14] Ayton, G. S., & Voth, G. A. (2009). Systematic multiscale simulation of membrane protein systems. *Current Opinion in Structural Biology* (19) , 138–144.
- [15] Bavafaye-Haghighi, E., Yazdanpanaha, M., Kalaghchi, B., & Soltanian-Zadeh, H. (2009). Multiscale cancer modeling: In the line of fast simulation and chemotherapy. *Mathematical and Computer Modelling* (49) , 1449-1464.
- [16] Martins, M., Ferreira Jr., S., & Vilela, M. (2007). Multiscale models for the growth of avascular tumors. *Physics of Life Reviews* (4) , 128-156.
- [17] Zaslavsky, G., Guzdar, P., Edelman, M., Sitnov, M., & Sharma, A. (2008). Multiscale behavior and fractional kinetics from the data of solar wind–magnetosphere coupling. *Communications in Nonlinear Science and Numerical Simulation* (13) , 314–330.
- [18] Peters, N. (2009). Multiscale combustion and turbulence. *Proceedings of the Combustion Institute* (32) , 1–25.
- [19] John, V., & Roland, M. (2007). Simulations of the turbulent channel flow at  $Re=180$  with projection-based finite element variational multiscale methods. *International Journal for Numerical Methods in Fluids* (55) , 407–429.
- [20] He, X., & Ren, L. (2009). An adaptive multiscale finite element method for unsaturated flow problems in heterogeneous porous media. *Journal of Hydrology* (374) , 56–70.
- [21] Wang, C.-T., Jian, S.-R., Jang, J. S.-C., Lai, Y.-S., & Yang, P.-F. (2008). Multiscale simulation of nanoindentation on Ni (1 0 0) thin film. *Applied Surface Science* (255) , 3240-3250.
- [22] Shiari, B., & Miller, R. E. (2008). Multiscale modeling of ductile crystals at the nanoscale subjected to cyclic indentation. *Acta Materialia* (56) , 2799–2809.
- [23] Jin, J., Shevlin, S., & Guo, Z. (2008). Multiscale simulation of onset plasticity during nanoindentation of Al (001) surface. *Acta Materialia* (56) , 4358–4368.
- [24] Yu, W., & Shen, S. (2009). Multiscale analysis of the effects of nanocavity on nanoindentation. *Computational Materials Science* .
- [25] Narayan, K. (2007). *A numerical fitting procedure for the Embedding Atom Method interatomic potential and a bridged finite element - molecular dynamics method for large atomic systems*. Toronto: Ryerson University.
- [26] Mansoori, G. A. (2005). *Principles of nanotechnology*. World Scientific.
- [27] Frenkel, D., & Smit, B. (1996). *Understanding molecular simulation - from algorithms to applications*. San Diego, CA: Academic Press.
- [28] Daw, M. S., & Baskes, M. (1983). Semiempirical, quantum mechanical calculation of hydrogen embrittlement in metals. *Physical Review Letters* (50) , 1285-1288.

- [29] Daw, M. S., & Baskes, M. (1984). Embedding Atom Method: Derivation and application to impurities, surfaces, and other defects in materials. *Physical Review B* (29) , 6443-6453.
- [30] Foiles, S., Daw, M. S., & Baskes, M. (1986). Embedding Atom Functions for the fcc metals Cu, Ag, Au, Ni, Pd, Pt and their alloys. *Physical Review B* (33) , 7983-7991.
- [31] Kaupp, G. (2006). *Atomic force microscopy, scanning nearfield optical microscopy and nanoscratching*. Springer.
- [32] Van Swygenhoven, H., Hasnaoui, A., & Derlet, P. M. (2004). *Nanoindentation in nanocrystalline metallic layers: A molecular dynamics study on size effects*. Villigen, Switzerland: Paul Scherrer Institut, NUM/ASQ.
- [33] Shi, Y., & Falk, M. L. (2007). Simulations of nanoindentation in a thin amorphous metal film. *Thin Solid Films* (515) , 3179-3182.
- [34] Saraev, D., & Miller, R. E. (2006). Atomic-scale simulations of nanoindentation-induced plasticity in copper crystals with nanometer-sized nickel coatings. *Acta Materialia* (54) , 33-45.
- [35] Touloukian, Y., Kirby, R., Taylor, R., & Desai, P. (1970). *Thermophysical properties of matter Vol. 12*. IFI/Plenum.
- [36] Gerberich, W., Tymyak, N., Grunlan, J., Horstemeyer, M., & Baskes, M. (2002). Interpretations of indentation size effects. *Journal of Applied Mechanics* (69) , 433-442.

# Appendix A – Flowcharts for Molecular Dynamics Code

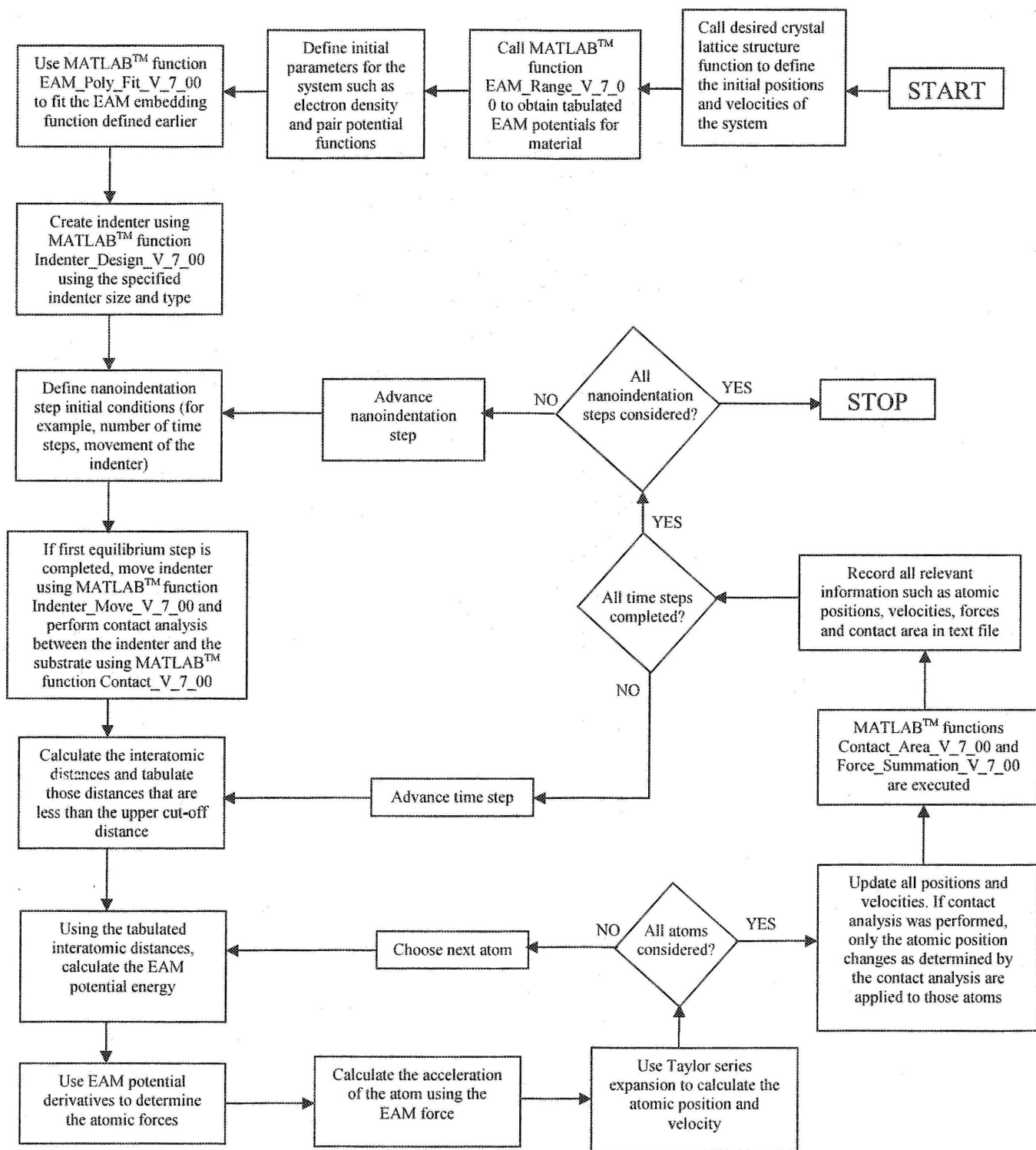


Figure 34 – Program Execution Flowchart for MATLAB™ function NI\_MD\_V\_7\_00

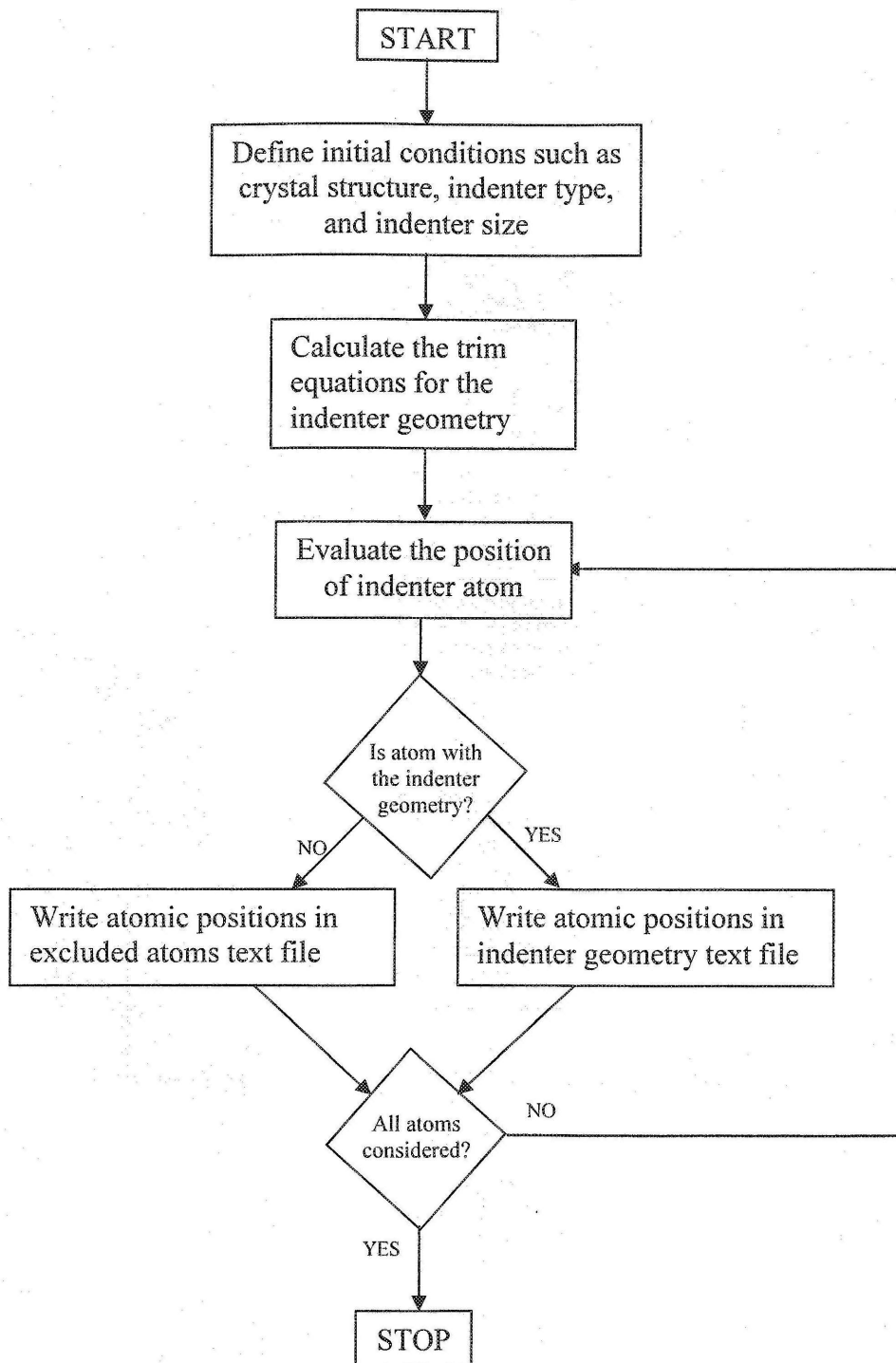


Figure 35 – Program Execution Flowchart for MATLAB™ function `Indenter_Design_V_7_00`

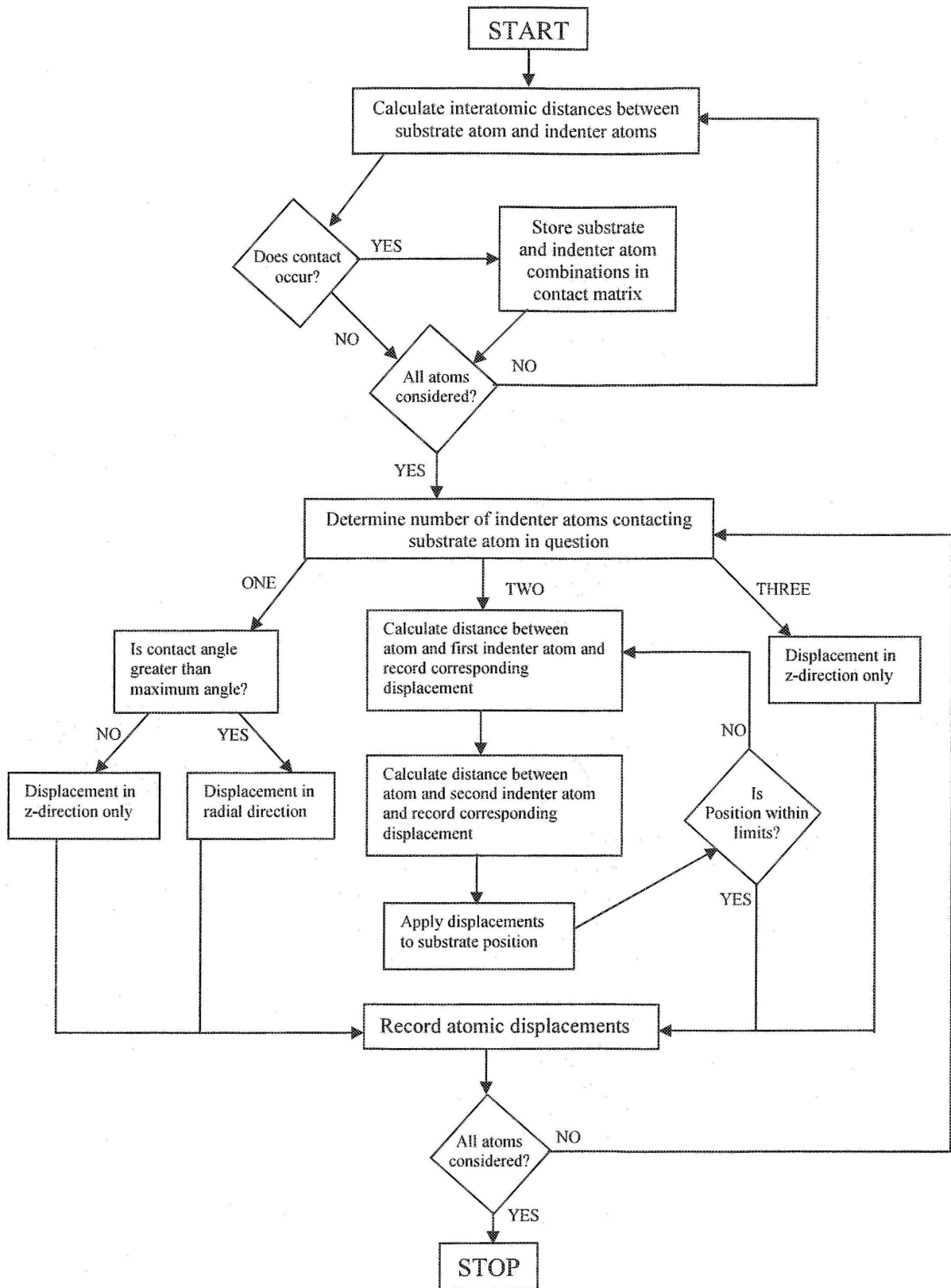


Figure 36 – Program Execution Flowchart for MATLAB™ function Contact\_V\_7\_00

## Appendix B – Flowcharts for Bridged FE-MD Code

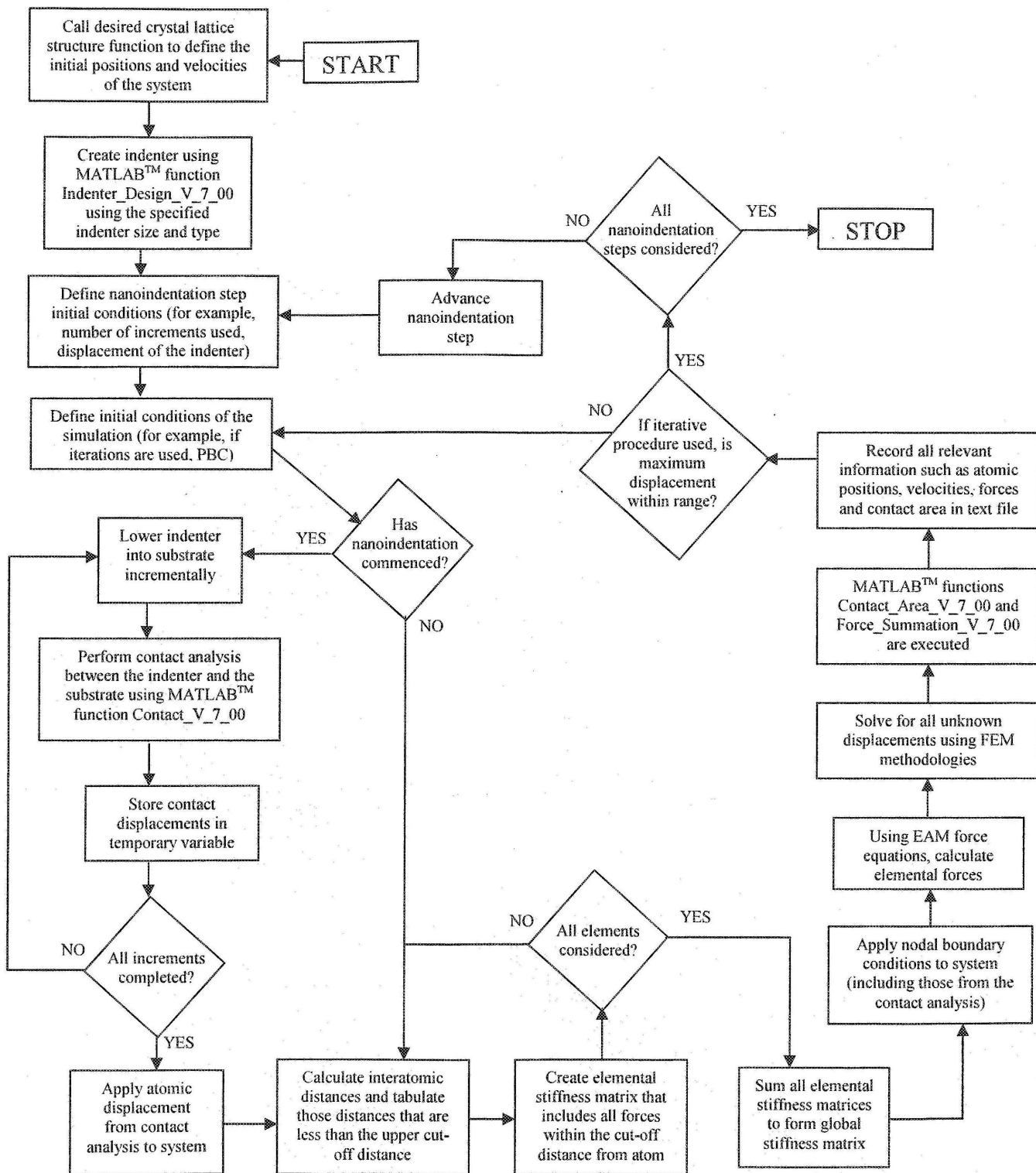


Figure 37 – Program Execution Flowchart for MATLAB™ function NI\_FE\_MD\_V\_7\_00