

# A novel multiscale algorithm for concurrent coupling of atomistic and continuum scales with applications to tribological problems

Pandurangan, Venkataraman

2011

Pandurangan, V. (2011). A novel multiscale algorithm for concurrent coupling of atomistic and continuum scales with applications to tribological problems. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/47639>

<https://doi.org/10.32657/10356/47639>

# A Novel Multiscale Algorithm for Concurrent Coupling of Atomistic and Continuum Scales with Applications to Tribological Problems



Submitted by

**VENKATARAMAN PANDURANGAN**

A dissertation submitted to Nanyang Technological University  
in partial fulfillment of the requirement for the degree of Doctor of Philosophy

**School of Mechanical & Aerospace Engineering  
Nanyang Technological University, Singapore**

2011

# Acknowledgments

My foremost thanks to my supervisors, Dr. Ng Teng Yong and Dr. Li Hua, for their valuable guidance, periodic reviews and constructive comments that made this PhD work productive. I am grateful to my friends in the School of Mechanical and Aerospace Engineering (MAE), with whom I had many fruitful discussions, and whose company made the university life memorable. I would also like to acknowledge MAE for the financial support rendered during these four years. Lastly, I would like to thank my parents and my sister for their prayers, support and encouragement, in all my endeavors. The loving and faithful support of my wife, R. Gomathi during the final years of this PhD is greatly appreciated.

# Contents

<b>Acknowledgments</b>	<b>i</b>
<b>List of Figures</b>	<b>v</b>
<b>List of Tables</b>	<b>viii</b>
<b>List of Symbols</b>	<b>ix</b>
<b>Abstract</b>	<b>xi</b>
<b>1 Introduction</b>	<b>1</b>
1.1 General Classification of Multiscale Methods . . . . .	5
1.2 An Overview of Different Multiscale Models . . . . .	8
1.2.1 The quasicontinuum method . . . . .	8
1.2.2 Cluster-energy quasicontinuum method CQC(m)-E . . . . .	11
1.2.3 Ghost force corrected quasicontinuum method . . . . .	12
1.2.4 Bridging scale method . . . . .	13
1.2.5 Finite element atomistic (FEAt) method . . . . .	16
1.2.6 Coupling of length scales (CLS) method . . . . .	18
1.2.7 Coupled atomistic and discrete dislocation (CAAD) method .	20
1.2.8 Bridging domain method . . . . .	23
1.2.9 AtC method . . . . .	25
1.2.10 A concurrent multiscale method based on the meshfree method and molecular dynamic analysis . . . . .	27
1.2.11 Multiscale simulation based on MLPG method . . . . .	29
1.2.12 Multiscale simulation based on the temperature-related Cauchy-Born rule . . . . .	32
1.3 Research Objectives . . . . .	34
1.4 Report Outline . . . . .	36
<b>2 Mathematical Modeling of Atomistic and Continuum Domains</b>	<b>38</b>
2.1 Continuum Region – Meshless Hermite-cloud Method . . . . .	39

2.1.1	Meshless method : overview and motivation . . . . .	39
2.1.2	Strong-form and weak form methods . . . . .	42
2.1.3	Smooth particle hydrodynamics and reproducing kernel particle methods . . . . .	45
2.1.4	Hermite-cloud method . . . . .	46
2.2	Molecular Dynamics . . . . .	60
2.2.1	Modeling the system - inter-atomic potentials . . . . .	61
2.2.2	Initial and boundary conditions . . . . .	65
2.2.3	Non-dimensional units . . . . .	66
2.2.4	Velocity rescaling . . . . .	67
2.2.5	Numerical integration . . . . .	68
2.2.6	Measuring physical properties . . . . .	69
2.2.7	Limitations of MD simulation . . . . .	71
2.3	Chapter Summary . . . . .	72
<b>3</b>	<b>Atomic-Continuum Coupling</b>	<b>73</b>
3.1	General Framework and Classification of Multiscale Problems . . . . .	73
3.2	Domain Decomposition Methods . . . . .	76
3.3	Atomic-continuum Coupling Algorithm Based on the Schwarz Alternating Method . . . . .	79
3.4	Chapter Summary . . . . .	84
<b>4</b>	<b>Results and Discussions</b>	<b>87</b>
4.1	Static Problems with Coupled Length Scales . . . . .	88
4.1.1	1-D Problems . . . . .	88
4.1.2	2-D Problems . . . . .	97
4.2	Transient Problems with Coupled Length and Time Scales . . . . .	104
4.2.1	Wave propagation problems . . . . .	104
4.2.2	Vibration of a stretched elastic string . . . . .	120
4.2.3	Vibration of stretched elastic membrane . . . . .	123
4.3	Chapter Summary . . . . .	127
<b>5</b>	<b>Multiscale Simulation of Nanoindentation and Nanoscratching</b>	<b>128</b>
5.1	Multiscale Simulation of Nanoindentation . . . . .	129
5.1.1	Review of instrumented indentation . . . . .	129
5.1.2	Multiscale model . . . . .	136
5.1.3	Numerical results . . . . .	140
5.2	Multiscale Simulation of Nanoscratching . . . . .	148
5.2.1	Review of nanoscratch testing . . . . .	148
5.2.2	Multiscale model . . . . .	150

5.2.3	Numerical results . . . . .	151
5.3	Chapter Summary . . . . .	157
<b>6</b>	<b>Conclusions and Recommendations</b>	<b>158</b>
6.1	Major Theoretical Achievements . . . . .	158
6.2	Tribology Studies via the Currently Developed Multiscale Method . .	160
6.3	Final Remarks . . . . .	160
6.4	Recommendations . . . . .	161
	<b>Publications Arising from this Thesis</b>	<b>162</b>
	<b>References</b>	<b>163</b>

# List of Figures

1.1	Schematic of multiscale modeling approach. . . . .	2
2.1	Plane stress problem under pure bending. . . . .	56
2.2	Plane stress problem: comparison of analytical and numerical solution (a) $u$ displacement (b) $v$ displacement. . . . .	58
2.3	Plane stress problem: comparison of analytical and numerical solu- tion: displacement derivatives (a) $u_x$ (b) $v_x$ (c) $u_y$ (d) $v_y$ . . . . .	59
2.4	12 – 6 Lennard Jones potential. . . . .	64
2.5	Morse potential. . . . .	64
3.1	General framework of multiscale problem. . . . .	74
3.2	An example of overlapping subdomains. . . . .	78
4.1	Computational model for static 1-D problems. . . . .	90
4.2	Comparison of analytical and multiscale simulation results for the 1-D Poisson equation. . . . .	90
4.3	Comparison of analytical and multiscale simulation results for the 1-D heat conduction equation with a heat source. . . . .	92
4.4	Comparison of analytical and multiscale simulation results for the Poisson equation with a local high gradient. . . . .	93
4.5	Variation of global error with the number of iterations. . . . .	95
4.6	Effect of the transition region width on the global error. . . . .	95
4.7	Effect of the transition region width on the solution convergence. . . .	96
4.8	Effect of density ratio on the global error. . . . .	96
4.9	Computational model of the first type for 2-D problems. . . . .	98
4.10	Comparison of analytical and multiscale simulation results for the 2-D Poisson equation. . . . .	99
4.11	Comparison of analytical and atomistic simulation results for the 2-D Poisson equation. . . . .	99
4.12	Computational model of the second type for 2-D problems. . . . .	100

4.13	Comparison of analytical and multiscale simulation results for the 2-D Laplace equation. . . . .	101
4.14	Comparison of analytical and atomistic simulation results for the 2-D Laplace equation. . . . .	101
4.15	Comparison of analytical and multiscale simulation results for the 2-D Poisson equation with local high gradient. . . . .	103
4.16	Comparison of analytical and atomistic simulation results for the 2-D Poisson equation with local high gradient. . . . .	103
4.17	Computational model for 1-D wave propagation problems. . . . .	107
4.18	Snapshots of the analytical and multiscale simulation results at different time instances for 1-D wave propagation problem with a Gaussian input wave (case-1). . . . .	109
4.19	Snapshots of the analytical and multiscale simulation results at different time instances for 1-D wave propagation problem with a ‘Hat function’ (case-1). . . . .	110
4.20	Snapshots of pure MD simulation results for 1-D wave propagation with a Gaussian input wave (case-1). . . . .	111
4.21	Snapshots of analytical and multiscale simulation results at different time instances for 2-D wave propagation problem with a Gaussian input wave (case-1). . . . .	112
4.22	Snapshots of the analytical and multiscale simulation results at different time instances for 1-D wave propagation problem (case-2). . . . .	116
4.23	Snapshots of the analytical and multiscale simulation results at different time instances for 1-D wave propagation problem (case-3). . . . .	118
4.24	Global error variation as a function time for wave propagation problems. . . . .	119
4.25	Energy of the atomistic region as a function of time for wave propagation problems. . . . .	120
4.26	Computational model for transverse vibration of a stretched string . . . . .	122
4.27	Comparison of analytical and multiscale simulation results at different time instances for the transverse vibration of a stretched string. . . . .	122
4.28	Snapshots of analytical and multiscale simulation results at different time instances for 2-D wave propagation problem. . . . .	126
5.1	Schematic of nanoindentation test setup (VanLandingham, 2003). . . . .	131
5.2	Schematic of load-displacement curve from an indentation experiment (Oliver and Pharr, 2004). . . . .	133
5.3	Schematic of deformation pattern observed during an indentation test (Oliver and Pharr, 2004). . . . .	133



---

5.4	Multiscale model of nanoindentation showing atomic (AR), continuum (CR) and handshaking (HR) regions. . . . .	137
5.5	Snapshots of the simulation at different stages of indentation. . . . .	142
5.6	Load-displacement curve from multiscale simulation. . . . .	143
5.7	Comparison of the load-displacement curves. . . . .	144
5.8	Displacement profile at the end of simulation. . . . .	146
5.9	Comparison of load-displacement curves from MD and multiscale simulation. . . . .	146
5.10	Comparison of load-displacement curves for different indenter angles. . . . .	147
5.11	Plot of maximum indentation force at different indentation speeds. . . . .	148
5.12	Schematic of nanoscratching test setup. . . . .	149
5.13	Snapshots of the nanoscratch simulation during the indentation and scratch phases. . . . .	154
5.14	Variation of normal force during the indentation phase. . . . .	155
5.15	Variation of tangential force as a function of scratch length. . . . .	155
5.16	Coefficient of friction as a function of scratch length. . . . .	156

# List of Tables

3.1	Multiscale algorithm for atomic-continuum coupling. . . . .	85
4.1	Comparison of global and relative errors for static 1-D problems. . . .	94
4.2	Comparison of global and relative errors for static 2-D problems. . . .	104
4.3	Comparison of global and relative errors for transient 1-D problems. .	124
5.1	Morse potential parameters. . . . .	139
5.2	Force/unit length values from MD and multiscale simulations. . . . .	145

# List of Symbols

S.No	Symbol	Description
1.	$A$	Projected contact area ( $\text{mm}^2$ )
2.	$A_{ij}(x_k, y_k)$	Symmetric moment matrix
3.	$a$	Acceleration ( $\text{m/s}^2$ )
4.	$B(s)$	Basis function vector
5.	$C^*(x, y)$	Column coefficient vector
6.	$C(x, \xi)$	Correction function
7.	$c$	Wave speed ( $\text{m/s}$ )
8.	$E$	Young's modulus ( $\text{N/mm}^2$ )
9.	$F$	Deformation gradient
10.	$f^h(x, y)$	Approximate solution of a two dimensional function $f(x, y)$
11.	$f_i$	Force on atom $i$ (N)
12.	$f_{ij}$	Interactive force between atoms $i$ and $j$
13.	$H$	Hamiltonian or the total energy of the system (J)
14.	$h = \Delta t$	Time step for numerical integration
15.	$K(x - \xi, h)$	Kernel function or smoothing function
16.	$L$	Differential operator
17.	$M_m(x, y)$	Shape function of the 1 <sup>st</sup> order derivative at the $m^{\text{th}}$ node
18.	$m_i$	Mass of the $i^{\text{th}}$ atom (kg)
19.	$N_n(x, y)$	Shape function at the $n^{\text{th}}$ node
20.	$r_{ij}$	distance between the atoms $i$ and $j$

21.	$r_0$	Equilibrium bond distance
22.	$T$	Temperature (k)
23.	$u(x)$	Displacement variable (mm)
24.	$v$	Velocity (m/s)
25.	$W^*(z)$	Cubic spline window function
26.	$X_i$	Atomic position of $i^{th}$ atom
27.	$z$	Cloud size along the $x$ and $y$ directions
28.	$\alpha, \beta$	Linear scaling parameters
29.	$\Gamma_D, \Gamma_N$	Dirichlet and Neumann boundaries
30.	$\pi$	Total potential energy of the system (J)
31.	$\sigma$	Normal stress (N/mm <sup>2</sup> )
32.	$\tau$	Shear stress (N/mm <sup>2</sup> )
33.	$\varphi(r_{ij})$	Pair-wise interaction potential
34.	$\psi, \xi$	Global and relative error measures
35.	$\Omega$	Computational domain

---

# Abstract

Multiscale modeling approaches have attracted a lot of attention in the past decade due to the computationally efficient solutions offered by multiscale models for problems characterized by multiple length/time scales. Multiscale methods take advantage of the localized nature of physical problems and use more than one computational model for an accurate description of a system across different length/time scales. Typical examples would be nanoindentation and nanoscratching problems, where the region near the point of indentation or scratching is subject to large deformation gradients and therefore require Angstrom scale descriptions, while the region away from these points will experience significantly smaller strains and can be satisfactorily described using some suitable continuum models. The logical approach for solving these problems would be to build a multiscale model that advantageously couples different computational models. In these cases, a multiscale model that uses both the molecular dynamics and the finite element/meshless approach could be used, with the molecular dynamics method providing an accurate solution in the region surrounding the crack tip, and the continuum model providing a reasonably accurate solution in the far-field.

An atomistic or continuum model cannot in itself be used for these types of problems as it might be computationally prohibitive to simulate the entire problem using an atomistic model, whereas a continuum model may not be able to describe the entire problem accurately. Building a multiscale model thus ensures accurate results by using the most appropriate model to describe the physics at respective scales,

and also substantially reduces the computational expense by restricting the method requiring a higher computational overhead to a small region of the problem domain, only where it is essential, thereby making it feasible to study problems over larger length/time scales.

The major challenge involved in developing a multiscale model is to ensure a seamless interface between the constituent length/time scales. To address this issue, a novel concurrent multiscale numerical method is proposed in this work to provide a seamless coupling or handshaking between the atomistic and continuum scales. The novelty in the proposed multiscale model is that it uses a strong-form meshless Hermite-cloud method, which approximates both the field variable and corresponding first-order derivative simultaneously, for continuum domain discretization. Therefore, the coupling between the atomistic and continuum scales is achieved by ensuring the compatibility of both the field variable and the first-order derivative, and also ensuring force equilibrium across the overlapping transition region. The use of a strong-form method further eliminates the need for any mesh generation.

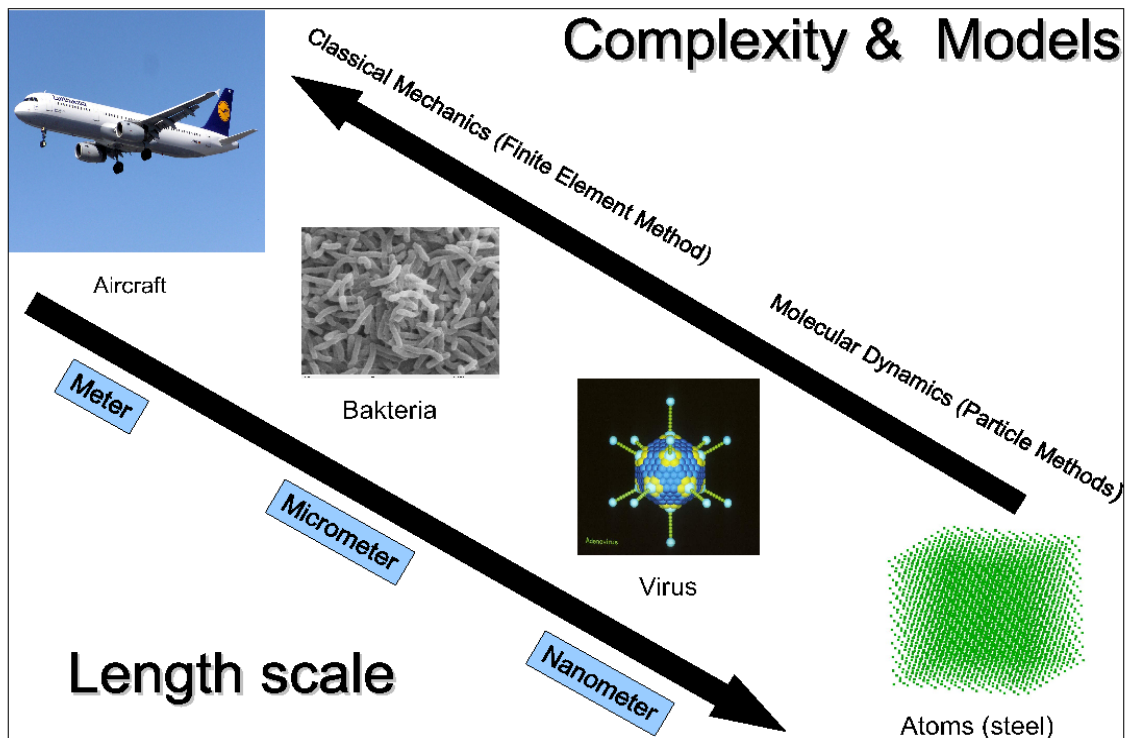
The proposed multiscale model is validated numerically by solving static and transient benchmark problems in one and two-dimensional domains, and the results are presented. In addition, nanoindentation and nanoscratching experiments on copper thin films are simulated using the developed multiscale model and compared with corresponding full atomistic simulations. The material properties obtained from the nanoindentation simulation include the load-displacement graph and the force/unit length values, obtained by dividing the maximum load on the indenter by its contact perimeter. The nanoscratching problem is solved using an adaptive node distribution scheme to maintain the size of the atomistic region constant. The normal and tangential forces, and the coefficient of friction obtained from the simulation are analyzed and compared with the values reported in literature.

# Chapter 1

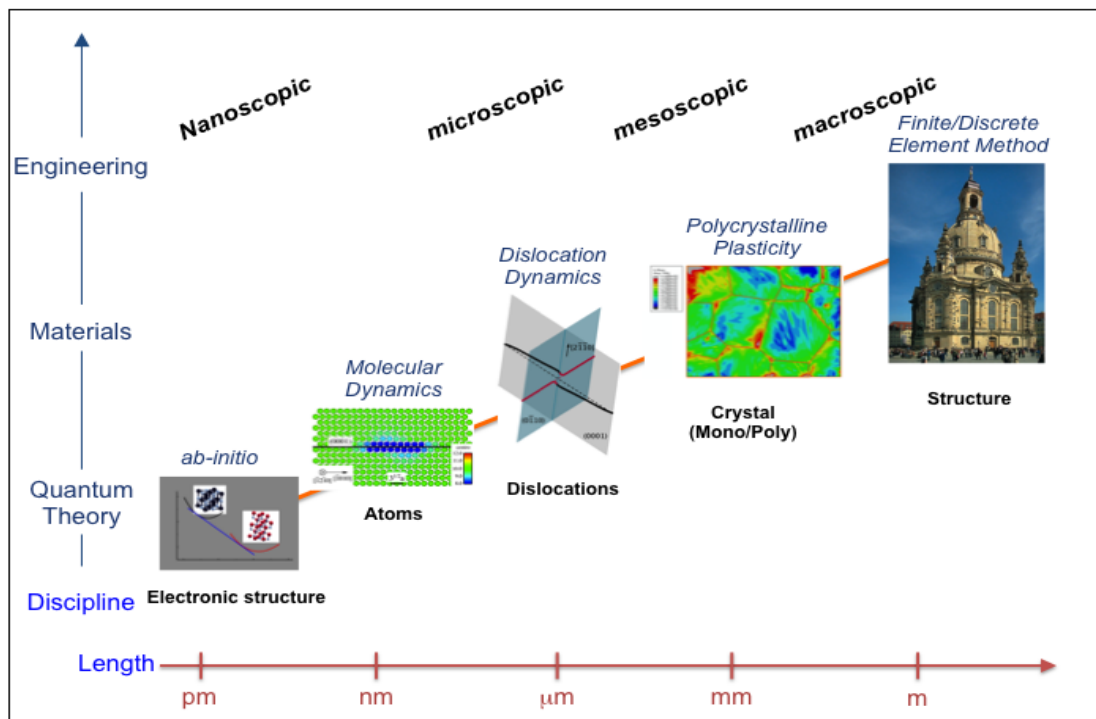
## Introduction

Modeling and simulation has become an indispensable research tool, and this is due to the rapid advancements that are taking place in the field of engineering and science, together with a phenomenal increase in the computational resources. Recently, there is a growing interest in the field of nanotechnology due to the availability of a large number of experimental and modeling tools to characterize materials at nanoscale. Nanoscale materials in particular, have attracted a lot of research interest due to their inherent advantages like smaller size, lower structural weight, better strength and improved electrical and mechanical properties. These materials have a wide application potential in areas such as nano-electronics, drug-delivery, sensors, medical diagnostic systems, material reinforcement etc. (Liu et al., 2004).

One of the major challenges involved in simulating nanoscale systems is that they are characterized by phenomena occurring across multiple length and/or time scales. Figure 1.1 shows a multiscale framework, where the length and time scales typically ranges from a few Angstroms and pico-seconds at the atomistic level to microns and milli-seconds at the continuum level. Multiscale modeling is a newly developing field that promises accurate and computationally efficient solutions to such problems,



Source: <http://www.zib.de/fackeldey/multiscale.html>



Source : <http://tu-freiberg.de/fakult4/imfd/cms/Multiscale/multiscale.html>

**Figure 1.1:** Schematic of multiscale modeling approach.



characterized by multiple length and/or time scales.

Multiscale methods take advantage of the fact that in many of the engineering problems such as crack propagation or nanoindentation the physical phenomena is localized. If we consider nanoindentation problem as an example, we find that the region beneath the indenter, where the deformation gradients are very high and the lattice structure is severely deformed, requires an Angstrom scale description, while the region away from the indenter that experiences smaller strains, can be satisfactorily described using a continuum model. Hence, we find that the problem requires more than one computational model for an accurate description at different scales. In this case, the molecular dynamics (MD) method would be able to provide solution at the atomic scale, where the atoms interact through empirical inter-atomic potentials, while the continuum region can be modeled using the principles of elasticity and can be discretized using the finite element method (FEM) or meshless approaches. In problems such as the above, a monoscale approach such as the MD or FEM cannot in itself be used over the entire problem domain for two reasons; first, a coarse scale description such as FEM may not be accurate and valid everywhere in the problem domain; and second, it might be computationally prohibitive to simulate the entire problem using a fine scale approach such as the MD method.

An effective solution is therefore to build a model that advantageously couples different computational models to create a multiscale model. For example, atomistic and continuum methods could be used in combination to model the nanoindentation problem discussed above. In this case, a multiscale model that uses both MD and FEM approaches may be able to capture the deformation behavior of the system, with the atomistic model providing accurate solution around the indentation site and the continuum model providing a reasonably accurate solution in the far-field.

Building a multiscale model has two major advantages; first, using the most appropriate model to capture the essential physics at respective scales leads to better

and accurate results; second, the computational expense is greatly reduced, as the method requiring a higher computational overhead is limited to a small region of the problem domain, restricted only to those areas where it is essential. This makes it computationally feasible to study problems over larger length scales with reduced computational cost and increased efficiency, without any significant loss in the solution accuracy. Multiscale modeling is not only restricted to modeling nanoscale phenomena; rather the approach can be used for modeling any complex/hybrid system characterized by multiple length/time scales.

Multiscale modeling also reduces the quantum of information that needs to be analyzed. If the system is modeled entirely using the lowest time/length scale, additional degrees of freedom would be involved and the simulation results would contain a lot of unnecessary information from which meaningful results have to be extracted. By reducing the number of degrees of freedom in the system, multiscale models make analysis of results easier. The results obtained from a multiscale model and the ideal one constructed entirely using the lowest length/time scale would almost be the same, making the multiscale model cheaper in terms of the computational cost.

The most important aspect in multiscale modeling is the information exchange between different scales. In order to do that, it is essential to have a proper understanding of the ways in which the different scales involved in the problem are linked together. If we consider the nanoindentation problem, the deformation behavior around the indenter, modeled with atomistic precision, might depend on the deformation behavior of the surrounding region, modeled using a continuum approach and vice-versa. To model this problem, we need to understand how the atomistic and continuum scales are linked together, the degree to which they are linked, the type of coupling that needs to be established between them and the information that needs to be exchanged. The type of coupling and the information to be exchanged may differ from problem to problem, and the multiscale model has

to use the most appropriate coupling approach to obtain accurate and meaningful results. The general classes of multiscale methods are discussed in the next section.

## 1.1 General Classification of Multiscale Methods

The smallest length scales generally encountered in solid state systems is the subatomic scale comprising of the nuclei and the electrons orbiting around them. Tight-binding empirical models based on quantum mechanics principles are used at these scales. At the next higher level is the atomic scale where atoms are treated as a whole without any regard to the subatomic particles, and their motions are governed by inter-atomic potentials. Molecular dynamics method is generally used for modeling systems at these scales. At the highest level is the continuum scale where matter is regarded as continuous and infinitely divisible, without regard to its atomic structure. Constitutive relations and physical laws are used for solving continuum problems.

As mentioned in the previous section, coupling of length scales plays a major role in multiscale modeling. The method used for coupling depends on the degree to which different scales in the problem are linked to each other. In systems where the scales are weakly linked, coupling may be sequential. In such cases, calculations are performed at individual scales separately and then information is passed on to the next higher level. In systems where there is a stronger linking of the scales, a concurrent approach is more appropriate, wherein a coupling scheme ensures seamless exchange of information across the transition region. In certain cases the scales may be so strongly coupled that the use of multiscale model may fail to provide a solution.

The serial or hierarchical coupling is the simplest type of coupling that has been in use for a long time and is very appropriate for modeling systems in which the scales

are weakly coupled. In this type of coupling the problem domain is broken down into different scales and information is passed on from the smallest scale to the next higher level. The use of information from one scale as the input to the next scale provides the necessary coupling. The simplest example of a serial coupling is the use of constitutive rules that are obtained by carrying out atomistic scale simulations on defect free crystals. Another much cited example is the work done by Clementi and Reddaway (1988), where they used quantum mechanical, MD and fluid dynamic simulations to compute the tidal circulation at Buzzard's bay. The hierarchical model is valid only when the scales are well separated and the largest length scale in the problem can be considered homogeneous and quasi-static, meaning that the variations at the highest level do not affect the smaller scales (Rudd and Broughton, 2000). Though simple to implement, the serial approach is limited because the fine scale effects that are parametrized by atomistic simulation performed on simple defect free crystals may not be able to accurately describe complex problems, which might otherwise be possible from an ideal full atomistic simulation.

Concurrent multiscale models offer better solutions to problems where the scales are strongly coupled. In a concurrent multiscale approach, the simulations at the individual scales are carried out at the same time and there is a continuous exchange of information between the scales during the simulation. Atomistic-continuum coupled concurrent multiscale models, that is the focus of this work, have the atomistic scale centered over the domain of interest, with the continuum scale surrounding it. The transition region between the two scales facilitates smooth exchange of information by enforcing appropriate boundary conditions at the interface to ensure consistency.

The recent reviews on concurrent multiscale models (Broughton et al., 1999; Vvedensky, 2004; Park and Liu, 2004; Liu et al., 2004; Miller and Tadmor, 2009), highlight some of the major developments that have taken place in this field. Due to their vast potential in modeling systems across multiple length and time scales, nanoscale

systems in particular, concurrent multiscale approaches are increasingly finding applications in the field of science and engineering and have been successfully applied to study crack propagation (Lee et al., 2009; Miller et al., 1998b; Kohlhoff et al., 1991; Park et al., 2005a; Rafii-Tabar et al., 1998), nanoindentation (Chang et al., 2007; Picu, 2000; Shenoy et al., 2000; Shiari et al., 2008; Smith et al., 2000; Wang et al., 2008), nanomaterials (Karakasidis and Charitidis, 2007; Liu et al., 2008), wave propagation (Smirnova et al., 1999), nanometric cutting (Sun et al., 2006), stress analysis (Lidorikis et al., 2001; Liu et al., 2003) and in modeling complex biological systems (Ayton et al., 2007; Demongeot et al., 2003). Multiscale approaches that couple the atomistic and continuum scales are of greater significance and find wide applications in modeling systems at the micro/nano-scale (Curtin and Ronald, 2003).

Different concurrent multiscale approaches basically differ from each other in the method used to model the transition region that facilitates the exchange of information between different scales in the domains. The next section provides a brief review of some of the concurrent multiscale models developed to date. The review highlights the salient features of these multiscale models and also identifies their major features. The review mainly focuses on multiscale models that couple atomistic and continuum domains. As the atomic and continuum descriptions are incompatible, some approximation has to be made and boundary conditions enforced, along the interface region, so that there is “handshaking” or a smooth exchange of information. The way in which this “handshaking” is done differentiates various multiscale models.

## 1.2 An Overview of Different Multiscale Models

### 1.2.1 The quasicontinuum method

The quasicontinuum method (QC) is an energy based multiscale approach that has been successfully used to study a number of engineering problems such as crack propagation/fracture, grain boundary deformation, nanoindentation etc. (Miller et al., 1998a; Shenoy et al., 1998; Tadmor et al., 1999; Knap and Ortiz, 2001; Tadmor et al., 1996; Miller et al., 1998b). The review paper by Miller and Tadmor (2002) highlights the major improvements and development that method has undergone, and also summarizes some of the key results obtained using it. The QC method does not differentiate atomistic and continuum regions in the computational domain. Instead, an atom or node in the domain is either classified as a local representative atom (repatom in short) or a non-local repatom. As the inter-atomic potentials used in the atomistic models have a global/non-local support and the shape functions used in the finite element method have a local support, the non-local and local repatoms can be indirectly identified with the atomistic and continuum domains respectively.

The objective of the QC method is to determine an expression for the total potential energy  $\Pi$  of the system and achieve equilibrium by minimization of this energy. In the QC model the total potential energy of the system, consisting of  $N$  atoms, is expressed as

$$\Pi = E_{QC}(\tilde{u}) = E^A - \sum_{i=1}^N f_i^{ext} \tilde{u}_i \quad (1.1)$$

where  $E^A$  is the interaction energy of the atoms and  $f_i \tilde{u}_i$  is the potential energy due to the external load  $f_i$  on atom  $i$ . The interaction energy,  $E^A$ , is a function of atomic displacements  $\tilde{u}_i = x_i - X_i$ , where  $x_i (i = 1, \dots, N)$  is the atomic position of the atoms in the domain, and  $X_i$  is some known reference configuration. Assuming that the interaction energy can be defined for each atom in the computational domain

using the atomistic model, the interaction energy  $E^A$  can be written as

$$E^A = E(\tilde{u}_1, \dots, \tilde{u}_N) = \sum_{i=1}^N E_i(\tilde{u}_1, \dots, \tilde{u}_N) = \sum_{i=1}^N E_i(\tilde{u}) \quad (1.2)$$

In order to reduce the number of degrees of freedom in the system, a set of atoms are identified as repatoms ( $n_{rep}$  such that  $n_{rep} \ll N$ ) and all the remaining atoms in the system are constrained to follow the displacements of these repatoms. The repatom position can be chosen based on any set criteria, one such example being the deformation gradient. If we consider a crack propagation problem as an example, a majority of the atoms in vicinity of the crack tip, where the deformation gradients are large, are chosen as repatoms, while only a few are chosen in the regions away from the crack tip, where the behavior is elastic and the gradients are smaller. A 2D/3D finite element mesh is then constructed from these repatom positions. The displacement of all other atoms in the system is then determined from the nodal repatom displacements  $U$ , using interpolation functions, as done in the finite element method. The modified expression for the interaction energy is then obtained as

$$E(U) = \sum_{i=1}^N E_i(u(\tilde{U})) \quad (1.3)$$

The error introduced due to this approximation is controlled by choosing every atom as repatom in the critical regions.

To determine the total energy of the system, the problem domain is divided into atomistic and continuum sub-domains and the interaction energy, which is the sum of atomistic and continuum energies, is written as

$$E_{QC} = E(U) = \sum_{i \in A} E_i(\tilde{u}(U)) + \sum_{i \in C} E_i(\tilde{u}(U)) \quad (1.4)$$

with subscripts  $A$  and  $C$  denoting the atomistic and continuum domains respectively. In simpler terms, the region with higher repatom density is taken as the atomistic

region and the region with a lower repatom density is considered as the continuum region. The energy in the atomistic region is calculated in the same manner as in any full atomistic model, with an additional condition that the padding atoms in the interface region are constrained to follow the displacements of the continuum region. As the deformation gradient is uniform in the continuum region, the energy of the continuum region, subject to a uniform deformation gradient  $F$ , is computed by summing the energies of the atoms over the elements using the Cauchy-Born rule, resulting in a modified expression for the total energy as

$$E_{QC} = \sum_{i \in A} E_i(\tilde{u}(U)) + \sum_{e \in C} n^e \Omega_0 W(F^e) \quad (1.5)$$

where  $\Omega_0$  is the Wigner-Seitz volume of a single atom in the reference configuration and  $n^e$  is the number of atoms in an element  $e$ . An appropriate weighting function is used for elements that are in the transition region between the atomistic and continuum domains, to avoid the double counting of energy. The weighting function takes the value of unity for elements that are not in the interface region.

The novel approach in QC method is the use of Cauchy-Born rule for computing the energy in the continuum region using an atomistic approach. Also a single expression for calculating the energies, based on the atomistic position alone, permits adaptive re-meshing. However, in the QC method the forces on the nodes and atoms are obtained as the derivatives of the energy functional. This leads to spurious forces in the transition region, also known as ghost forces, and mainly arise due to the mismatch between the non-local and local nature of atoms and nodes in the transition region.



### 1.2.2 Cluster-energy quasicontinuum method CQC(m)-E

In the QC method a set of representative atoms or repatoms are chosen to act as nodes and a finite element mesh is constructed between them. The displacement of the all other atoms in the domain are then determined from the repatoms displacements through interpolation functions. In the cluster-energy quasicontinuum (CQC(m)-E) method (Eidel and Stukowski, 2009), the energy of each atom in the domain is computed from the energy of a cluster of atoms surrounding each node, with  $m$  denoting the number of atoms in the cluster. The energy of every repatom  $I$  is then computed from the average energy of a set of  $C^I$  atoms in the cluster as

$$\bar{E}^I(U) = \frac{1}{m^I} \sum_{i \in C^I} E_i(\tilde{u}(U)) \quad (1.6)$$

where  $m^I$  is the number of atoms in the cluster  $I$ . Obviously, the accuracy is improved by increasing the number of atoms in the cluster, however with an increased computational expense. The clusters are of same size, except in the refined regions where the clusters are closer to each other and overlap, and also in the atomistic region where  $m = 1$ . In regions where the clusters overlap each other, the atoms are assigned to the cluster of the nearest repatom, so that the same atom does not belong to more than one cluster. The energy of an atom in the cluster is dictated by the atomic positions of other atoms in the cluster, the displacements of which are controlled by the nodal/repatom displacements. The total potential energy  $\Pi$  of the system is then computed as the weighted sum of the repatom energies as

$$\Pi = E_{tot}(U) = \sum_{I \in R} n^I \bar{E}^I(U) = \sum_{I \in R} \frac{n^I}{m^I} \sum_{i \in C^I} E_i(\tilde{u}(U)) \quad (1.7)$$

where  $R$  is the set of atoms that are chosen as repatoms, and  $n^I$  is an appropriate weighting function assigned to each repatoms in the domain that satisfies the

condition  $\sum_{I \in R} n^I = N$ , where  $N$  is the total number of atoms in the system. In the fully refined atomistic region, where the cluster size is one, the weighting function also tends to one. The number of atoms in the cluster and the weighting function distinguish the atomistic and continuum regions. A value of  $m = 1$  and  $n = 1$  corresponds to the atomistic region, while for the continuum region both values are greater than one. In summary, the energy of the repatoms in the continuum region is obtained as the average value of the atomistic energy of the cluster, with the atoms undergoing constrained deformation. The accuracy of the method largely depends on an appropriate choice of cluster size.

### 1.2.3 Ghost force corrected quasicontinuum method

As mentioned in Section 1.2.1, the local-non-local mismatch between the continuum and atomistic descriptions, gives rise to spurious forces in the transition region, also known as ghost forces. The ghost force corrected quasicontinuum method (Shenoy et al., 1999), aims to eliminate these ghost forces through a ghost force correction technique. In this method, the ghost forces are determined for a given reference configuration as ‘dead loads’ and the negative of this quantity is then applied at the affected atomic/nodal locations, and the potential energy expression defined in Eq. (1.5) is modified as

$$E_{QC} = \sum_{i \in A} E_i(\tilde{u}(U)) + \sum_{e \in C} n^e \Omega_0 W(F^e) - \sum_g g^I u^I \quad (1.8)$$

where  $g$  is the ghost force computed at the atomic/nodal location  $I$ . Though the above equation corrects the ghost force errors at the interface, it assumes a constant value of the ghost force which is only valid for an undeformed configuration. However, the ghost forces may not remain constant with a changing displacement field and for a non-uniform deformations at the interface, the dead load ghost force correction technique may lead to additional spurious forces that are not intended.

To deal with this problem, a new definition was proposed for ghost force correction that eliminates these forces in both the deformed as well as the undeformed configurations. In this new technique, the ghost force on an atom/node is defined as the force that the atom/node would not feel if the entire domain is discretized using a fully atomistic or continuum approach respectively. The ghost force correction is then incorporated using this revised definition and the problem of constant ghost forces is dealt with by updating the ghost forces continuously. However this leads to a different expressions for the energy functional at every time instance, and no one expression for the potential energy functional for minimization.

The ghost force correction can also be applied to the cluster-energy quasicontinuum method and it is observed that with ghost force correction applied, the method produces better and more accurate results even with smaller clusters. For a given repatom force  $f^{CQC(m)-E}$ , computed at every repatom location using a cluster of radius  $r$  and  $m$  atoms, the unknown correct forces  $f^*$  is defined as the force computed with an infinite cluster radius or in other words with all atoms included in cluster. An approximate correct force  $\tilde{f}^*$ , is then defined as the force on the repatom that is computed with a slightly larger cluster radius ( $r_{new} > r$ ). The difference between the two forces is then taken as the ghost force

$$g = f^{CQC-E} - \tilde{f}^* \quad (1.9)$$

The ghost force correction is then carried out in a similar fashion as done in the case of the QC method using Eq. (1.8)

### 1.2.4 Bridging scale method

The bridging scale method (BSM) (Liu et al., 2006b; Park et al., 2005b; Qian et al., 2004; Tang et al., 2006a; Wagner and Liu, 2003) is also an energy based formulation

that couples atomic and continuum scales. In the BSM, the total displacement  $u$  of a body is divided into fine and coarse scale displacements and written as

$$u = u' + u'' \quad (1.10)$$

where  $u'$  and  $u''$  correspond to the coarse and fine scale displacements respectively.

Consider a system consisting of  $n_a$  atoms with the atomic displacement  $u_i$ , defined at the  $i^{th}$  atom with initial position  $X_i$ . The coarse scale component  $u'(X_i)$ , is a continuous displacement field that can be computed everywhere in the problem domain using finite element or meshfree shape functions defined over a set of nodal locations  $n_c$ . The coarse scale displacement  $u'(X_i)$ , at an atomic position  $X_i$  can therefore be interpolated as

$$u'(X_i) = Nd \quad (1.11)$$

where  $d$  is the nodal displacement vector and  $N$  is the shape function matrix.

The fine scale solution is the part of the total displacement that cannot be represented by the coarse scale. The fine scale displacement  $q_i$  computed using the molecular dynamics approach also includes a part of the solution that can be computed using the coarse scale approach. Therefore, the fine scale solution is obtained by subtracting from the fine scale displacement,  $q_i$ , the projection of the fine scale displacement on the shape function  $N_I$ , using a projection operator that minimizes the mass weighted square of the difference between the coarse and total scales at an atom, and is given as

$$\sum_i m_i \left( q_i - \sum_I N_I^i w_I \right)^2 \quad (1.12)$$

where  $w_I$  is a set of nodal degrees of freedom that minimizes the above equation

and  $m_i$  is the atomic mass. The fine scale displacement is finally obtained as

$$u'' = q - Nw \quad \text{or} \quad u'' = q - Pq \quad (1.13)$$

where  $P$  is the projection operator. The total displacement can be finally obtained as a sum of coarse and fine scale displacements as

$$u = Nd + q - Pq \quad (1.14)$$

The terms  $Pq$  in the above equation is called the bridging scale that must be subtracted from the total displacement to make the coarse and fine scales independent of each other.

The atomic displacements and their time derivatives are used to compute the Lagrangian  $L$  of the system, from which the equations of motion for the fine and coarse scale degrees of freedom derived as

$$M\ddot{d} = -\frac{\partial U}{\partial d} \quad (1.15)$$

$$M_A\ddot{q} = -\frac{\partial U}{\partial q} \quad (1.16)$$

where  $U$  is the potential energy and  $M$ ,  $M_A$  are the mass matrices corresponding to the coarse and fine scales respectively. The force obtained as the negative derivative of the potential energy is a function of both fine and coarse scales. As both  $q$  and  $u$  satisfy the same equations of motion, the coarse scale solution can be obtained from the fine-scale solution by projection.

The objective of the BSM is to restrict the fine scale description to a small region in the problem domain, at the same time retaining the effects of the fine scale in the coarse scale region. The coupling in the BSM is thus achieved in two steps. First, the fine scale degrees of freedom outside the MD region are eliminated and appropriately accounted, so that the coarse scale does not become redundant. The

elimination introduces an additional boundary force in the MD equations of motion in the form of a time history kernel that represents the dissipation of fine scale energy outside the MD region. The time history kernel is solved analytically using Laplace transform taking advantage of the periodic nature of atomic lattices, but can also be solved numerically in complicated cases. Next, the coarse scale internal forces are computed from the fine scale internal force ensuring the exchange of information between MD and continuum regions.

The clear distinction between the coarse and fine scales in the BSM results in a smaller number of integration points and time steps for integrating the equations of motion. Finally, the bridging scale method can also be used to solve dynamic problems at finite temperature.

### **1.2.5 Finite element atomistic (FEAt) method**

The FEAt method (Izumi et al., 2001; Kohlhoff et al., 1991) is one of the earliest multiscale models to be developed. The method is based on a force-based formulation and accounts for the local/non-local mismatch at the atomistic/continuum interface by employing a non-local continuum formulation in the interface region. In the FEAt method, the coupling between the atomistic and continuum domains is effected through a set of interface and pad atoms. The continuum region is graded down at the transition region such that the nodes coincide with both the transition and pad atoms. The method uses the Embedded atom model (EAM) in the atomistic region and hence the padding region thickness must at least be equal to twice the cut-off radius used in the inter-atomic potential, to ensure that the atoms have their complement of neighbors. Coupling between the two domains is achieved by enforcing appropriate boundary conditions at the interface.

The FEAt method in addition, addresses the non-local/local mismatch between the atomistic and continuum domains by using a non-local continuum formulation in

the interface region. The energy of the an element  $i$  in the continuum region is defined in two ways; a local non-linear formulation for elements that are fully in the continuum region and a non-local formulation for the elements in the pad region, which are given as

$$\begin{aligned} E_i^{(non-linear)} &= \int_{\Omega_i} W_{non-linear}(X) dV \\ E_i^{(non-local)} &= \int_{\Omega_i} W_{non-local}(X) dV \end{aligned} \quad (1.17)$$

where  $\Omega_i$  is the volume of element  $i$ , and  $W$ , the strain energy density at a point  $X$ , is defined in the two regions as

$$\begin{aligned} W_{non-linear}(X) &= \frac{1}{2}[C.\epsilon(X)] : \epsilon(X) + \frac{1}{6}[\epsilon(X)^T.C'.\epsilon(X)] : \epsilon(X) \\ W_{non-local}(X) &= \frac{1}{2} \int_{\Omega_i} [C^*(X - X').\epsilon(X')] : \epsilon(X) dV' \end{aligned} \quad (1.18)$$

where  $\epsilon(X)$  is the strain tensor evaluated at the point  $X$ ,  $C^*$  is a material-dependent non-local kernel, and  $C$ ,  $C'$  are elastic constants that are chosen such that the mismatch between the material models used in the atomistic and continuum domains is minimized.

The use of a non-local continuum formulation minimizes transition effects when moving from a non-local atomistic region to a local continuum region. The total energy of the elements in the continuum region is then obtained by summing the local or non-linear and the non-local energies. The energy of the atoms in the atomistic region is found from the atomic positions that includes both the interface and pad atoms and is given as.

$$E^A = \sum_{i \in (A, I, P)} E_i(r_A, r_I, r_P) \quad (1.19)$$

where subscripts  $A, I, P$  correspond to atomistic, interface and padding regions respectively. The model is then iteratively solved, wherein the interface atoms provide the boundary conditions for the FE solution and the interface and pad

atom positions provide the boundary conditions for the atomistic simulation. The problem of ghost forces are automatically taken care in the FEAt method by the use of a force based formulation, and the use of a non-local FE formulation in the transition region results in an improved solution.

### 1.2.6 Coupling of length scales (CLS) method

The CLS method (Rudd and Broughton, 2000; Broughton et al., 1999) is a hybrid multiscale method that couples quantum to continuum scales. The method was initially used to solve the problem of fracture in silicon, wherein the dynamics of crack propagation is progressively modeled using the tight-binding (TB) approximation, MD and FE approaches. The TB model is used in the region closest to the crack tip. The CLS method thus aims to provide a seamless coupling between FE/MD/TB models.

Although the CLS method involves both TB/MD as well as FE/MD coupling, only the FE/MD coupling approach, that is relevant to the present work, is discussed here. As with the FEAt method, the FE/MD coupling in the CLS method involves grading down the finite element mesh in the interface region to coincide with both the pad and the interface atoms. The thickness of the pad atoms is at least equal to twice the cut-off radius used in the inter-atomic potential. The CLS method uses a linear elastic formulation in the continuum region, with the potential energy in the continuum region defined as the sum over  $i$  elements as

$$E^c = \sum_i E_i \tag{1.20}$$

where the energy of an element  $i$ , of volume  $\Omega_i$ , is computed from the strain energy



density  $W(X)$ , defined at a point  $X$ , as

$$E_i = \int_{\Omega_i} W_{linear}(X) dV \quad (1.21)$$

The CLS method uses the Stillinger Weber potential in the atomistic region that consists of both two and three body interactions. The total energy in the atomistic region is then calculated as the sum of the energies of both two and three body interactions. In calculating the energy of the atomistic domain, a bond contributes its full energy if the constituent atoms forming the bond are either in the atomistic or interface region, and is scaled by a factor of half if any one or more of the atoms forming the bond belongs to the padding region.

Finally, the total energy of the system is obtained as the sum of the atomistic and continuum energies, and also the weighted energies in the interface region as

$$E = \sum_{i \in (A,I)} E_i^{(2)}(r_A, r_I, r_P) + \sum_{i \in (A_1)} E_i^{(3)}(r_A, r_I) + \sum_{i \in (A_2, I)} \hat{E}_i^{(3)}(r_A, r_I, r_P) + \sum_i w_i E_i \quad (1.22)$$

where the subscripts  $A, I, P$  correspond to the atomistic, interface and padding regions respectively,  $E^{(2)}$  and  $E^{(3)}$  correspond to the energies due to two and three-body interactions, and  $w_i$  is a weighting factor for the continuum region that is assigned a value one for elements that are entirely in the continuum region and half for those at the interface. The energy term due to the three-body interactions has two components,  $E^{(3)}$  and  $\hat{E}^3$ , in the regions  $A_1$  and  $A_2$ , which are a subset of the atomistic region such that  $A_1 \cup A_2 = A$ . The first term accounts for three-body interactions that are entirely in the atomistic region, while the second term accounts for three-body interactions that cross the interface, and is computed by scaling the three-body energy using a weighting factor  $w_{ij}$ , that depends on the number of

atoms from the padding region that were involved in the bond formation, as

$$\hat{E}_i^{(3)} = \frac{1}{6} \sum_{j \neq i} \sum_{k \neq (i,j)} w_{ij} V_{ijk}^{(3)}(r_{ij}, r_{ik}) \quad (1.23)$$

The total energy functional in the CLS method is similar to that obtained for the QC method except for the weighting factor introduced in the interface region for computing the total energy. The method therefore experiences ghost forces similar to that of the QC method.

The unique feature of the CLS method is that in addition to atomistic/continuum coupling, the method also couples quantum and atomistic scales. The method used for MD/TB coupling is in principle similar to the one used for MD/FE coupling, the difference being that both quantum and atomistic models are non-local. Also, the energy of the quantum region is calculated for all the atoms together and not on a per-atom basis as done in the MD model.

### 1.2.7 Coupled atomistic and discrete dislocation (CAAD) method

The CADD method (Shilkrot et al., 2002a;b; 2004) is similar to the other multiscale approaches such as the Finite element atomistic (FEAt) method and coupling of length scales (CLS) method discussed above, wherein the problem domain is spatially divided into atomistic and continuum domains that are individually modeled using the MD and continuum approaches. The CADD method in addition allows for the presence of discrete dislocations (DD) in the continuum region. These dislocations are permitted to move and interact with one another and also with the atoms in the atomistic region. In CAAD method, the continuum deformation field is thus described by the movement and interactions of the continuum dislocations. The CADD method is developed in two-dimensional setting and allows dislocations

presence only in a direction perpendicular to the  $x - y$  plane, but the dislocations can be of any type. The 2D setting is achieved by employing periodic boundary conditions along the  $z$  direction in the atomistic region, and permitting three degrees of freedom at every node in the continuum region.

The CADD method uses the embedded atom method (EAM) in the atomistic region and the DD method (Van Der Giessen and Needleman, 1995) to handle the continuum region with discrete dislocations. The objective is to solve the boundary value problem, subject to some traction force  $T_0$  and displacements  $u_0$ , to obtain the position of the atoms, nodes and discrete dislocations. To do this, the problem domain  $\Omega$  is first divided into atomistic and continuum regions  $\Omega_A$  and  $\Omega_C$  that are connected through an interface region  $\partial\Omega_I$ , with the subscripts  $A, C$  and  $I$ , referring to atomistic, continuum and interface regions respectively. The dislocations in the continuum domain is defined in terms of the Burgers vector  $b_i$  and dislocation position  $d_i$ , corresponding to the  $i^{th}$  dislocation. The degrees of freedom in the atomistic region are the atomistic positions  $r_A$  and  $r_I$ , while those in the continuum region are the nodal displacements  $U_C$ ,  $U_I$ , and the discrete dislocation positions  $d_i$ .

Coupling between the atomistic and continuum domains, through the interface region, is achieved by introducing a set of pad atoms that ensure that the atoms near the interface have a complement of neighbors. The thickness of the padding region must at least be greater than the cut-off length used in the inter-atomic potential. The position of the pad atoms  $r_P$ , is dictated by the continuum displacement fields. The continuum region is graded down to the atomic scale at the interface region to coincide with the interface atoms and the displacement of these interface nodes is determined from the atomic positions. The total potential energy for the atomistic domain that also includes the pad atoms  $P$  is given as

$$E^A = \sum_{i \in (A, I, P)} E_i(r_A, r_I, r_P) - f^A u_A \quad (1.24)$$

where  $u_A$  is the atomic displacement and  $f^A$  is the external load. The atomistic forces are obtained as the partial derivatives of the atomistic energy  $E^A$ , with respect to the atomic positions  $r_A$  and  $r_I$ . Similarly, the potential energy of the continuum domain is calculated from the sum of the energies of the finite elements, and the external work due to applied initial traction  $T_o$  as

$$E^C = \sum_{i \in (I,C)} E_i(U_I, U_C, d^i) - \int_{\partial\Omega} T_o u dA \quad (1.25)$$

The coupling approach in CADD method also involves the detection and passing of dislocation information between the two domains which can be explained as follows. Dislocations that originate in the atomistic region are first detected and passed on to the continuum regions as discrete dislocations. Dislocations are detected through a detection band, which is a thin strip of elements positioned in the atomistic region at a small distance from the atomistic/continuum interface. Dislocations are detected by defining a plastic strip strain tensor for these elements as

$$E^P = \frac{(b \otimes m)_{sym}}{d} + \frac{(m \otimes b)(b \otimes m)}{2d^2} \quad (1.26)$$

where  $m, b$  and  $d$  correspond to slip plane normal, Burgers vector and inter-planar spacing respectively, and *sym* indicates symmetry. With a prior knowledge of the characteristic of dislocations expected in a crystal, the plastic strain matrix  $E_i^P$  is calculated for all the possible dislocations, and an additional dislocation is added for the no slip condition, for which the Burger vector  $b = 0$ . With an update in the atomistic positions, the actual strains,  $E$  are computed for the elements in the detection band. Whenever the  $E$  value approaches any of the computed  $E_i$  values, a dislocation core with a known Burgers vector is detected in that element, which is then passed on to the continuum domain as discrete dislocations.

To pass the dislocations, the displacements of the atoms and nodes are modified by adding the continuum elastic displacements of a dislocation dipole to the

atomistic/nodal displacements, thereby eliminating the atomistic core and adding a new dislocation in the continuum region. The reverse process of detecting and passing a dislocation in from the continuum to atomistic region is done easily. In addition to dislocations, the atomistic region can also contain defects such as grain boundaries, vacancies, voids, or amorphous regions. However, only the dislocation information is allowed to cross the atomistic/continuum interface.

### 1.2.8 Bridging domain method

The bridging domain method (BDM) (Xiao and Belytschko, 2004) to couple atomistic and continuum domains is an energy based method, that minimizes ghost force effects by making a gradual transition from the non-local atomistic region to the local continuum region. The BDM divides the problem domain into atomistic and continuum regions  $\Omega^a$  and  $\Omega^c$  that overlap each other in the interface region  $\Omega^{int}$ , also known as the bridging domain. In the BDM, we begin with by defining the total energy or Hamiltonian  $H$ , of the system which is the sum of potential and kinetic energies. The Hamiltonian for the atomistic and continuum regions are given as

$$H^a = W^a(x_i) + \sum_i \frac{1}{2m_i} P_i^a \cdot P_i^a \quad (1.27)$$

$$H^c = W^c(x_I) + \int_{\Omega^c} \frac{1}{2} \rho v^T v d\Omega^c - \sum_I f_I^{ext} u_I \quad (1.28)$$

where  $W^a(x_i)$  is the total potential energy of the atomistic region computed as the sum of the energies of the pair-wise interactions, three body potentials and external forces if any and  $m_i$ ,  $P_i$  correspond to the mass and momentum of the  $i^{th}$  atom respectively. In the second equation,  $W^c(x_I)$  is the potential energy of the continuum region that is computed from the strain energy density,  $f^{ext}$  is the external force,  $\rho$  is the density, and  $u_I$ ,  $v(x, t)$  are the displacement and velocity of the  $I^{th}$  node. The method adopts a Lagrangian approach and assumes that the

deformations in the continuum region are small and therefore the region is defect free.

In the interface region or bridging domain, where the atomistic and continuum regions overlap, the total energy is computed as the linear combination of the atomistic and continuum energies, by introducing scaling parameter  $\alpha$ , defined as

$$\alpha = \begin{cases} 1 & \Omega^c - \Omega^{int} \\ [0, 1] & \Omega^{int} \\ 0 & \Omega^a - \Omega^{int} \end{cases} \quad (1.29)$$

The Hamiltonian for the problem, which is the sum of atomistic and continuum Hamiltonians is obtained as

$$H = (1 - \alpha)H^a + \alpha H^c = \sum_i (1 - \alpha) \frac{P_i^a \cdot P_i^a}{2m_i} + (1 - \alpha)W^a + \sum_I \alpha \frac{P_I^c \cdot P_I^c}{2m_I} + \alpha W^c \quad (1.30)$$

Displacement compatibility in the interface region is ensured by constraining the atomistic displacements to follow the interpolated continuum displacement field at every reference atomic position as

$$h_i = u(X_i) - \tilde{u}_i = \sum_I N(X_i)U_I - \tilde{u}_i \quad (1.31)$$

where  $\tilde{u}_i$  is the displacement of an atom in the interface region and  $u(X_i)$  is the interpolated displacement field at an atomic position  $X_i$ . Compatibility is ensured by forcing the difference  $h_i$  to zero, which is achieved using the Lagrange multiplier or the augmented Lagrange method. If the augmented Lagrangian approach is used, the Hamiltonian is modified as

$$H^{modified} = H + \sum_i \lambda_i^T h_i + \frac{1}{2} \sum_i \beta h_i^T h_i \quad (1.32)$$

where  $\lambda$  is a vector of Lagrange multipliers for every degree of freedom of an atom, and  $\beta$  is a penalty term. The forces on the nodes and atoms are obtained by differentiating this energy functional that includes the effect of the displacement constraint. The penalty term  $\beta$  introduced in the BSM does not have any physical significance and is just introduced for mathematical convenience.

The BSM method does not impose strong compatibility in the interface region and therefore the finite element mesh need not be scaled down to atomistic dimensions for enforcing displacement constraints. This is one of the major advantages of this method that permits more freedom in mesh generation. However, this reduces the accuracy of the solution and also makes it less amenable for adaptive simulations, where the atomistic and continuum regions change with time.

### 1.2.9 AtC method

The AtC method (Badia et al., 2007; 2008; Fish et al., 2007; Parks et al., 2008), is similar in principle to the BDM discussed in the previous section, the difference being that it uses a force formulation instead of an energy based approach of the BDM. As done in the BDM, the problem domain is divided into atomistic region  $\Omega^A$  and continuum region  $\Omega^C$  that overlap each other in the interface region  $\Omega^{int}$ , and coupling is achieved by blending of the forces in the interface region.

In the AtC method, we begin with by defining the forces in the atomistic and continuum regions. Neglecting the external forces, the forces in the atomistic and continuum regions can be expressed as

$$\begin{aligned} F^I &= - \sum_{e=1}^{n_{elem}} \int_{\Omega_c} P(F) \frac{\partial N^I}{\partial X} dV \\ f^\alpha &= \sum_{\beta \neq \alpha} f^{\alpha\beta} \end{aligned} \tag{1.33}$$

where  $F^I$  is the force on the node  $I$ ,  $P$  is the first Piola Kirchhoff stress tensor,

$F$  is the deformation gradient,  $N^I$  is the shape function of the  $I^{th}$  node,  $f^\alpha$  is the force on atom  $\alpha$ , and  $f^{\alpha,\beta}$  is the force exerted by the atom  $\beta$  on atom  $\alpha$ . The forces in the interface region are scaled using linear scaling parameters  $\eta$  and  $\chi$ , and the atomistic and continuum forces in the interface region are obtained as

$$\begin{aligned} f^\alpha &= \sum_{\beta \neq \alpha} \eta^{\alpha,\beta} f^{\alpha\beta} \\ F^I &= - \sum_{e=1}^{n_{elem}} \int_{\Omega_c} \chi P(F) \frac{\partial N^I}{\partial X} dV \end{aligned} \quad (1.34)$$

The correction factor  $\eta^{\alpha,\beta} = (\eta^\alpha + \eta^\beta)/2$ , introduced for scaling the atomic forces in the atomistic and interface region, ensures that the Newton's third law is satisfied, meaning that the scaled force on an atom  $\alpha$  due to the atom  $\beta$  is the same as the scaled force exerted on atom  $\beta$  due to atom  $\alpha$ .

Coupling between the atomistic and continuum regions is achieved by first ensuring strong compatibility in the handshaking region by constraining the displacements of the atoms to follow the displacement of the continuum nodes. Next, atomic forces in the interface region are transmitted to the nodes as

$$F^I = \sum_{e=1}^{n_{elem}} \int_{\Omega_c} \chi \left[ -P(F) \frac{\partial N^I}{\partial X} \right] dV + \sum_{\alpha=S^I} \sum_{\beta \neq \alpha} \eta^{\alpha,\beta} f^{\alpha\beta} N^I(X_\alpha) \quad (1.35)$$

Only the atoms  $\alpha$  within the element  $S$ , containing the node, are considered for transferring the forces. The AtC method also suffers from spurious ghost forces that affects its accuracy and can be corrected using the dead load correction technique used to correct the ghost forces in the QC method.



### 1.2.10 A concurrent multiscale method based on the mesh-free method and molecular dynamic analysis

One of the common features of the multiscale methods discussed so far is that they use the finite element approach for discretizing the continuum region. Multiscale models that use a meshless formulation in the continuum region are relatively new, and are gaining more attention due to the advantages offered by meshless methods over the conventional finite element approaches. The following sections provide a brief overview of some of the multiscale models that use a meshless method for continuum domain discretization.

The concurrent multiscale method to couple atomic and continuum domains developed by Gu and Zhang (2006), uses the weak-form radial point interpolation method (RPIM), in the continuum region and couples it with the MD method used in the atomistic region. The RPIM method uses the radial basis function (RBF) for constructing the shape function in which the field variable such as the displacement  $u$ , is interpolated as

$$u(x) = \sum_{i=1}^n R_i(r) a_i + \sum_{j=1}^l p_j(x) b_j = R^T a + B^T b \quad (1.36)$$

where  $R_i(r)$  is the multiquadratics RBF with  $r$  defined as the distance between the interpolation point and the field node,  $p_j(x)$  are monomial functions,  $l$  is the number of polynomial basis functions,  $n$  is the number of nodes included in the interpolation domain of a point, and  $a$  and  $b$  are interpolation constants that are determined by constraining Eq. (1.36) to be satisfied at all  $n$  field nodes included in the interpolation domain of a point. Solving for the constants, the nodal displacement vector is finally obtained using the RPIM shape function  $\varphi(x)$  as

$$u(x) = \sum_{i \in n} \varphi_i(x) u_i \quad (1.37)$$

An important property of the RBF shape functions is that they satisfy the kronecker delta property. The shape function can also be used to approximate the velocity vector in a similar function.

The constitutive equations in the continuum domain are obtained using the Cauchy-Born rule. The atoms in the deformed and undeformed configuration are related through a deformation gradient  $F$ . The Piola-Kirchhoff stress tensor is then constructed from the deformation gradient  $F$ , and the strain energy density  $w_c$  as  $P = \partial w_c(F)/\partial F$ . The strain energy density is computed by summing the atomistic potential,  $w^a$ , within a single element, as done in the case of the quasicontinuum method. As the RPIM is a meshfree technique, a virtual representative cell is constructed around every point of interest and the deformation gradient at that point is applied over the whole cell. The strain energy density of the continuum point is then found by summing the energies of the atoms within the cell as

$$w^c = \frac{1}{V^C} \left[ \sum_j w_{jv}^a(r) - \sum_j w_{jv}^a(r^0) \right] \quad (1.38)$$

where  $V^C$  is the volume of the representative cell and  $w_{jv}^a(r)$ ,  $w_{jv}^a(r^0)$  are the potential energy of the atoms in the representative cell in the deformed and undeformed configurations respectively. The Hamiltonian Canonical equations of motion for the continuum domain are then obtained as

$$\dot{P}_I^c = -\frac{\partial H^C}{\partial X_I^c} = \int_{\Omega_0^c} \frac{\partial w_c}{\partial u_I^c} d\Omega \quad (1.39)$$

$$\dot{X}_I^C = \dot{u}_I^c = \frac{\partial H^C}{\partial P_I^C} = \frac{p_I^c}{m_I^c} \quad (1.40)$$

where  $H^C$  is the Hamiltonian in the continuum region, and  $m_I^c$ ,  $P_I^C$  and  $X_I^C$  are the mass, momentum and position of node  $I$  respectively. A lumped mass matrix is used while solving the equations. The equations of motion for the atomistic region are obtained using the MD model.

The atomic domain  $\Omega^a$  and continuum domain  $\Omega^c$  are coupled through a transition region  $\Omega^t$ , in which additional transition particles are introduced, to ensure displacement compatibility and force equilibrium. For a transition particle  $k$  with displacement  $u_k$ , and subject to a force  $f_k$ , the compatibility and equilibrium conditions can be written as

$$u_k^c = u_k^a \quad (1.41)$$

$$f_k^c + f_k^a = 0 \quad (1.42)$$

To ensure conservation of mass and energy, the total energy in the transition domain is obtained as a linear combination of atomistic and continuum energies, as done in the case of the Bridging Domain method (see Section 1.2.8). The Hamiltonian of the problem domain is then obtained as the linear combination of atomistic, continuum and transition particles using Eq. (1.32). The introduction of transition particles permit independent distributions of the atoms and nodes and also help in the transfer of the energy between the two domains. The number of transition particles can be varied based on the compatibility requirements.

### 1.2.11 Multiscale simulation based on MLPG method

The multiscale method based on the Meshless Local Petrov-Galerkin (MLPG) method (Shen and Atluri, 2004), to couple atomic and continuum domains, uses the weak-form MLPG method for discretizing the continuum region. The MLPG method involves a meshless interpolation for the trial function and integration of the weak-form by an appropriate choice of a test function. In the MLPG method, the trial and test functions can be chosen from different function spaces with different support domains sizes, making the method very flexible. In the MLPG based multiscale method, radial basis function (RBF) is chosen to interpolate the trial function due to its Kronecker delta property. The field variable  $u(x)$  in the domain  $\Omega$

is interpolated using the radial basis function in a same manner as done in Eq. (1.36). The method uses a multiquadratics RBF function of the form  $R_i(x) = (r_i^2 + c^2)^\beta$ , where  $c$  and  $\beta$  are the shape parameters that are assigned the value 1 and 1.03 respectively.

The computational domain is divided into an equivalent continuum model (ECM) region, where the deformation is assumed to be homogeneous and defined in terms of the deformation gradient  $F$ , and a pure MD region, where the deformation is inhomogeneous. The ECM and MD regions are then coupled through the MLPG method. The ECM region contains both atoms and meshless nodes. The distribution of the nodes in the ECM region can be random and need not be coincident with the atoms.

The constitutive relation in the ECM region is obtained from atomistic calculations. The strain energy density,  $W$ , in the ECM region is obtained as  $W = \Pi_s/\Omega_s$ , where  $\Pi_s$  is the potential energy of the atoms in the representative cell and  $\Omega_s$  is the volume of the local domain. The Piola-Kirchhoff stress tensor  $S$  and the tangent stiffness material modulus,  $D$ , are then derived as from the strain energy density as

$$S = \frac{\partial W}{\partial E} = \frac{2}{\Omega_s} \frac{\partial \Pi_s}{\partial C} \quad (1.43)$$

$$D = 4 \frac{\partial^2 W}{\partial C \partial C} = \frac{4}{\Omega_s} \frac{\partial^2 \Pi_s}{\partial C \partial C} \quad (1.44)$$

where  $C$  is the Cauchy-Green stress tensor that is given as  $F^T F$ . Using the law of conservation of linear momentum, the following equation is obtained

$$\nabla(SF^T) + f = \rho w \quad (1.45)$$

where  $f$  is the body force, and  $\rho$  and  $w$  correspond to the density and acceleration respectively. The weak-form of the above equation, using the test function,  $V$ , in

the local domain  $\Omega_s$ , is obtained as

$$\int_{\Omega_s} [\nabla(SF^T) + f - \rho w] V \partial\Omega = 0 \quad (1.46)$$

The non-linear local symmetric weak-form of the above equation is solved using the total Lagrangean method. Finally, the position and displacement of an atom in the ECM region is obtained by interpolation using the RBF shape function. The displacement of an atom  $u_i$  in the ECM region is interpolated from the nodal displacements  $u_\alpha$ , using the RBF shape function  $\phi^\alpha$  as

$$u_i = \sum_{\alpha=1}^n \phi^\alpha(X_i) u_\alpha \quad (1.47)$$

As the number of nodes in the ECM region is less than the number of atoms, the atomic displacements computed using the above equation is only an average value and is therefore unable to capture the thermal fluctuations.

In order to capture the thermal fluctuations and also enable the elastic waves generated in the atomistic region to pass into the continuum region, thereby ensure seamless coupling, appropriate inter-facial conditions are employed. The real displacement,  $q_i$ , of an atom  $i$  in the ECM region, taking into account the thermal fluctuations is written as  $q_i = u_i + \delta u_i$ , where  $\delta u_i$  is the additional component that takes into account the thermal fluctuations. The total potential energy of the system can then be written as

$$\Pi(q_1, \dots, q_{N_{MD}+N_{ECM}}) = \Pi_0(u_{ECM}; q_{MD}) + \frac{\partial \Pi_0}{\partial u_{ECM}} \delta u_{ECM} \quad (1.48)$$

where  $\Pi_0$  is the zeroth-order approximation of the potential energy and  $\delta u_{ECM}$  is the thermal fluctuation vector that accounts for the missing atomistic degrees of freedom. Replacing  $\Pi_0$  with the first-order approximation of the potential energy,

the Newton's equation of motion can finally be obtained in a matrix form as

$$M_{MD}\ddot{q}_{MD} = f_{MD}^0 - K\delta u_{ECM} \quad (1.49)$$

where  $M_{MD}$  is the diagonal mass matrix,  $f_{MD}^0$  is the force vector and  $K = \partial^2 \Pi_0 / \partial u_{ECM} \partial q_{MD}$  is the tangent stiffness matrix. The MLPG multiscale method thus ensures seamless coupling by taking into account the thermal fluctuation of atoms in the ECM region that permits the successful transfer of atomistic energy across the atomic/continuum interface.

### **1.2.12 Multiscale simulation based on the temperature-related Cauchy-Born rule**

Multiscale methods that use the Cauchy-Born rule in the continuum domain to derive the constitutive relations from the atomistic potentials, assume that the deformations in the region are homogeneous and takes place at zero temperature, and therefore cannot be used to study the temperature effects on materials at nanoscale. Temperature is found to have an impact on the material properties at nanoscale (Xiao et al., 2006), which necessitates the need for a temperature-related homogenization technique. The multiscale simulation based on the temperature-related Cauchy-Born (TCB) rule (Xiao and Yang, 2007) addresses this issue by developing a constitutive relations that takes into account the thermal energy of the atoms in addition to their potential energy.

As per the TCB rule, the atoms are subject to local harmonic vibrational modes in addition to the homogeneous deformation at a given temperature. These vibrational modes are mostly uncoupled. The total free energy,  $W_H$ , for a crystalline solid

consisting of  $N$  atoms at a temperature  $T(X)$ , is obtained as

$$W_H(F, T) = \sum_i^{N_q} w_c(F_i^q) A_i + nk_B \sum_i^{N_q} n_i^q T_i^q \ln \left[ \frac{\hbar (\bar{D}(F_i^q)^{1/2n})}{k_B T} \right] \quad (1.50)$$

where  $w_c$  is the strain energy per unit volume,  $F$  is the deformation gradient,  $n$  is the number of degrees of freedom per atom,  $\hbar$  is the Plank's constant,  $k_B$  is the Boltzmann constant,  $\rho_n$  is the number of atoms per unit volume,  $N_q$  is the number of quadrature points in the domain,  $A_i$  is the volume associated with one quadrature point representing  $n_i^q$  atoms, and  $D$  is the dynamic matrix that is calculated from the vibrational co-ordinates and mass of the atoms. The second term in the above equation accounts for the temperature effects. The modified expression for the Piola-Kirchhoff stress tensor is obtained using the the free energy density,  $w_H$ , that is a function of the deformation gradient as well as the temperature as

$$P(F, T) = \frac{\partial w_H(F, T)}{\partial F} \quad (1.51)$$

Equation 1.51 is used as the constitutive relation in multiscale modeling of materials at finite temperature.

The meshfree particle method is used in the continuum region, wherein the displacements  $u$  are approximated as

$$u^h(X, t) = \sum_I w_I(X) u_I(t) \quad (1.52)$$

where  $X$  is the material co-ordinate in the reference configuration and  $w_I$  are Lagrangian kernels that are computed using a quadratic spline weighting functions. Substituting the above equation in the Galerkin weak-form of the momentum conservation equation leads to the equation of motion as

$$m_I \ddot{u}_{iI} = \rho_0 V_I^0 \ddot{u}_{iI} = f_{iI}^{ext} - f_{iI}^{int} \quad (1.53)$$

where  $V_I^0$  is the volume associated with particle  $I$ , and  $f_{iI}^{ext}$ ,  $f_{iI}^{int}$  are the external and internal nodal forces respectively.

Thus, the multiscale method based on the meshfree particle method uses the TCB rule to obtain the constitutive relations from atomistic calculations. This is done as follows. First, a voronoi cell is constructed with a meshfree particle,  $P_I$ , at the center and subjecting the atoms in the cell to the same deformation gradient as that experienced by the particle, and the strain energy density of the cell is computed at zero temperature. The dynamic mass matrix is then computed by assuming that the atoms within the cell undergo identical harmonic vibration. The atoms in the cell are then assumed to be at the same constant temperature as that of particle  $P_I$  and the free energy is computed using Eq. (1.50). The stresses on the particle is then computed using Eq. (1.51). Finally, numerical integration is carried out to compute the internal and external nodal forces and the equation of motion defined in Eq. (1.53) is solved.

### 1.3 Research Objectives

In general, most of the multiscale models discussed above, with just a few exceptions, use the finite element method for the discretization of the continuum domain. Though the finite element method is a popular choice for continuum simulations, it suffers from certain drawbacks such as mesh generation and re-meshing issues when solving nonlinear and moving boundary problems. Meshless methods (Belytschko et al., 1996), which are more recent in origin, have been successful in handling some these issues linked to conventional finite element approaches. Multiscale methods that use a meshless technique instead, for continuum discretization, are relatively new and have been successful in solving engineering problems. Most of the multiscale methods that use a meshless formulation in the continuum region (Wang et al., 2009; Shen and Atluri, 2004; Gu and Zhang, 2006), use a weak-form approach such as the



Element-Free Galerkin (EFG) method or the Radial Basis function (RBF) approach for discretizing the governing partial differential equation, and therefore still require some form of background mesh for numerical integration. To address the above issues, we propose in this project, a novel multiscale algorithm and its applications with the following objectives:

- *Develop a multiscale algorithm based on a strong-form meshless technique, namely the Hermite-Cloud method to couple atomistic and continuum scales*  
As mentioned above, most multiscale models use the finite element method to discretize the continuum region, and even those using a meshless framework, use the weak-form formulation for the discretization of the governing partial differential equation, and still require a background mesh for integration. In the present work, we proposed a multiscale model that uses the strong-form meshless Hermite-Cloud method, which totally avoids mesh generation in the continuum domain, making the computer implementation of the multiscale algorithm much simpler.
- *Ensure higher order compatibility in the transition region.* Handshaking models generally ensure field variable or displacement compatibility in the transition region, meaning that the continuity of the field variable is ensured between the two scales. The Hermite-cloud method, used in this work for continuum domain discretization, approximates both the field variable and the first-order derivative. Hence, in this work we propose an improved coupling/handshaking algorithm that ensures compatibility of both the field variable and its first-order derivative and also ensures the equilibrium of forces, across the overlapping transition region.
- *Numerically validate the developed multiscale model via comparison with benchmark problems.* The developed multiscale model is validated by solving several benchmark static and transient problems that involve both length and time

scale coupling. The simulation results are compared with both the analytical solution as well as a full atomistic simulation. The Laplace equation, Poisson equation with different forcing functions and local high gradients, and the classical wave propagation problem in one and two space dimensions are used to demonstrate the capability of the handshaking algorithm in coupling both length and time scales.

- *Simulate nanoindentation and nanoscratch numerical experiments using the developed multiscale model.* Nanoindentation and nanoscratch techniques have emerged as successful tools to evaluate material properties at nanoscale. Nanoindentation involves the continuous monitoring of the load and displacement data during the testing process and analyzing the resulting load-displacement graph to evaluate the mechanical properties, without the need for imaging the resulting indent. Nanoscratching involves scratching the sample surface with an indenter at a constant speed and at a defined normal force for a defined distance to determine tribological properties such as surface friction, abrasive wear etc. Though the MD method has been successfully used to simulate indentation and scratch experiments, the simulation sizes that are possible with this approach are greatly limited by the computational power. This limitation can be overcome with the use of multiscale models. Therefore, the proposed multiscale model is finally used to simulate indentation and scratching experiments on a copper thin film.

## 1.4 Report Outline

The need for a multiscale model, the different multiscale models developed to couple the atomistic and continuum length scales, and finally the objective and scope of the present work were presented in this chapter. The remaining chapters are organized as follows. Following this introduction, Chapter 2 provides an overview

of the individual computational models that make up the multiscale framework, namely the MD method used in the atomistic region and the meshless Hermite-cloud method used for modeling the continuum region. The algorithmic details of the newly developed handshaking/coupling algorithm, proposed in this work, to couple atomistic and continuum domains is discussed in Chapter 3. The numerical results of the static and transient test cases, carried out in both one and two space dimensions to validate the multiscale model, are presented in Chapter 4, while Chapter 5 presents the simulation results of the indentation and scratch experiments performed on a copper specimen using the developed multiscale model. Finally, Chapter 6 presents the conclusion and makes recommendations for further work in this direction.

## Chapter 2

# Mathematical Modeling of Atomistic and Continuum Domains

The main objective of this work is to develop a multiscale model that couples atomistic and continuum domains. The mathematical models used in the atomistic and continuum domain namely, the Molecular dynamics (MD) method and the meshless Hermite-cloud method are discussed in this chapter. MD method has been traditionally used for simulating atomic scale behavior and is used to simulate systems with a large number of atoms, which is otherwise too large from a quantum mechanics perspective. Meshless methods are relatively new when compared to the finite element approaches and are increasingly being used to solve continuum problems. Meshless methods use a set of nodes scattered in the domain for solving the differential equation, eliminating the need for a predefined mesh.

This chapter is organized in three sections. The first section provides an overview of meshless methods and highlights their salient features in comparison to the finite element method. The meshless Hermite-cloud method, used for continuum discretization in this work is also discussed. The second section provides an introduction to the molecular dynamics method and the basic steps involved in

it, and the last section provides the summary.

## 2.1 Continuum Region – Meshless Hermite-cloud Method

### 2.1.1 Meshless method : overview and motivation

Building a mathematical model is the first step in describing any physical problem in engineering or science. Most mathematical models of such problems are in the form of partial differential equations (PDE), with appropriate initial and boundary conditions prescribed. Closed-form analytical solutions to such equations rarely exist, and therefore numerical approaches have emerged as a popular choice to solve PDE's. Three of the most popular approaches used in the numerical solutions of PDE's include, the finite element method (FEM), the finite difference method (FDM) and the finite volume method (FVM).

The finite element method involves discretizing the domain into elements and approximating the field variable in each element using polynomial functions. The finite element method thus converts the PDE into a system of algebraic equations, which are then readily solved. The method, primarily developed to solve elasticity and structural problems, is increasingly finding applications in design and product development activities in the field of aerospace and automobile engineering, wherein it has significantly reduced lead times. The capability of the method to model complicated geometries together with a strong mathematical base has led to the development of a number of commercial softwares such as ABAQUS, ANSYS, etc., that are based on the finite element formulation. These softwares allow easy visualization of the field variable distribution in the domain and also provide a number of options for fine tuning the model, depending on the accuracy level

required. In the finite difference method, the differential equations is replaced with finite difference equations that approximate the derivatives at a set of grid points in the domain. These algebraic finite difference approximations are then solved to get the solution at these grid points. In the finite volume method, the domain is divided into a number of control volumes, instead of elements, with the field variable located at the centroid of the control volume. The governing equations are then integrated to obtain the solution. As the solution provided by the FVM satisfies the conservation of mass, momentum and energy, the method is widely used in the field of fluid mechanics.

The FEM and FDM, commonly used in the field of solid mechanics, have a few issues when solving PDE's

- The FEM, popularly used for the analysis of solid and structures, has problems mainly related to mesh generation. The main disadvantage with the FEM is that it requires a predefined mesh for solving the problem. Mesh generation is a very time consuming activity that requires a lot of human intervention.
- Generally in FEM, the primary or the field variables such as displacement or temperature is well approximated. However, secondary variables like stresses obtained from the method are generally less accurate and also discontinuous at the element interfaces.
- Element distortions is a serious concern in handling problems with large deformations when using the FEM. It is also difficult to simulate complex problems such as crack propagation, where the direction of crack growth might be arbitrary and might not be in line with the nodal locations, or problems that involve phase transition or material breakage.
- Adaptive meshing that involves re-meshing the domain at regular intervals is very challenging and computationally expensive when using the FEM. Also, the solution accuracy is greatly affected due to numerical errors introduced by

the mapping of field variables between successive stages of mesh refinement.

- The FDM, though well suited for modeling problems in the domain of fluid mechanics, suffers from the fact that it requires a regular uniform mesh and its application is limited to problems in rectangular domain or simpler geometries

Meshless methods (Nguyen et al., 2008), were mainly motivated from these issues related to other numerical approaches. The basic idea in meshless methods is to eliminate, either partially or fully, the need for a mesh and thereby overcome the issues associated with it. In a meshless approach, the problem is solved by scattering a set of discrete nodes in the problem domain including the boundaries. There is no mesh and therefore no prior information is required about the relationship between nodes. The field variable is approximated using shape function generated in real time and the problem is solved by discretizing the governing equation. Due to their attractive features, meshless methods are increasing finding applications in the area of applied mechanics (Li and Liu, 2002). Some of the salient features of these methods are

- The shape functions for the scattered nodes are generated in real time and are not predefined. The time consuming activity of creating a mesh with predefined shape functions is thus eliminated. There are also no connectivity issues as in the case of FEM.
- As no prior information is required to describe the relationship between nodes, the process of node generation can be easily automated. Also, the geometry can be more accurately represented using nodes than while using elements.
- As meshless methods do not require a predefined mesh, nodes can be easily added or subtracted in the problem domain. Adaptive meshing for problems such as crack-propagation or those involving large deformations, is therefore handled more easily. The accuracy of the method can be easily controlled as nodes can be easily added or subtracted at the required locations

- Secondary variables such as stress and strains can be approximated with higher accuracy than in FEM by the use of higher-order basis functions.

Meshless methods may be classified based on two criteria: first, based on the method used for constructing the shape function or interpolation function, and second, based on the method used to discretize the governing partial differential equation. Based on the method used to construct the interpolation function, they may be classified as finite difference, finite series and finite integral representation methods. Meshless methods that use a finite series or the finite integral representation ensure consistency by an appropriate choice of basis and weighting functions respectively, while methods that use a finite difference representation ensure convergence using Taylor's series.

Depending on the method used to discretize the governing PDE meshless methods may be classified as strong-form or weak-form based approaches. Strong-form based approaches use a collocation technique to discretize the PDE, while weak-form approach are based on Galerkin weak formulations. Different techniques used for constructing shape functions are combined with either a strong or weak-form approach resulting in different meshless schemes. The strong and weak-form methods used to discretize the PDE and the method used for constructing shape function are discussed in the following sections.

### **2.1.2 Strong-form and weak form methods**

Meshless methods based on strong-form or weak-form formulation, approximate the strong or weak-form of the governing partial differential equation respectively. Strong-form methods use a collocation approach to discretize the governing differential equation. Weak-form methods on the other hand, numerically integrate the weak-form of the governing equation using a background mesh, and therefore these methods are not strictly meshless. However, the mesh is required only for numerical integration and can therefore be much simpler than the one used in FEM.



Meshless methods based on weak-form approaches include the diffuse element method (Nayroles et al., 1992), element free galerkin method (Belytschko et al., 1994) and the point interpolation method (Liu and Gu, 2001b), to name a few. Their characteristics such as good stability, accuracy, and the ability to naturally satisfy Neumann boundary condition due to their weak form, have made these a popular choice in the field of solid mechanics. However, as mentioned earlier, weak-form methods are not strictly meshless as they still require a background mesh for numerical integrating the system matrices, which is computationally expensive. To avoid global integration that is computationally demanding, certain weak-form approaches such as the meshless local Petrov-Galerkin method (Atluri and Zhu, 1998), the local point interpolation method (Liu and Gu, 2001a) etc., have been developed. In these methods the local weak-form is integrated in local domains that are in the form of simple regular shapes such as circles, rectangles or triangles. Even with these methods, numerical integration is still an issue when dealing with complex shapes or in the region close to the boundary.

In comparison to weak-form based approaches, meshless methods based on strong-form discretization have an early developmental history, starting from the smooth particle hydrodynamics (SPH) (Lucy, 1977; Gingold and Monaghan, 1977; Monaghan, 1982). In SPH, the governing equation in strong-form is localized by introducing a smoothing function. The method has been popularly used to simulate problems in astrophysics such as the formation of stars and galaxies, supernovas, black holes, stellar collisions and even modeling the universe (Springel, 2010; Berczik and Kolesnik, 1993). The major shortcoming of the SPH method is its inability to accurately enforce boundary conditions, tensile instability zero-energy mode and a lack of consistency in interpolation (Li and Liu, 2002). A number of improvements have been proposed to SPH to address some of these issues (Monaghan, 2000; Randles and Libersky, 1996). The generalized finite difference method (GFDM) (Perrone and Kao, 1975; Jensen, 1972), based on strong-form discretization, was developed

in the seventies and was one of the earliest methods to introduced the concept of an arbitrary mesh. In the GFDM, the partial differential equation is replaced by numerical differential formulae that are based on polynomial interpolation. This method was initially used to solve problems expressed in a local formulation and later on extended to those expressed in a variational form. The GFDM method also suffers from the fact that the discrete equations developed by these are not symmetric, positive definite or well-conditioned etc., and has also been improved upon and extended by many researchers (Gavete et al., 2003).

Some of the other meshless methods developed based on a strong-form approach include the vortex method (Chorin, 1973), meshfree collocation method (Zhang et al., 2001), hp-cloud method (Liszka et al., 1996) and the finite point method (Onate et al., 1996). The major advantages of the strong-form methods are: they are algorithmically simple to implement, computationally efficient and truly meshless in the sense that they do not require a mesh either for interpolating the field variable or for numerical integration (Li and Liu, 2002). Due to these advantages strong-form methods have been used to solve a number problems in the field of fluid mechanics. However, these method suffer from certain drawbacks that include stability and accuracy issues in solving partial differential equations with derivative and Neumann boundary conditions and therefore find little application in solving solid mechanics problems that involves stress boundary conditions (Liu and Gu, 2003). Various approaches have been proposed to overcome some of these issues associated with strong-form methods, and some of major developments in this direction include the Hermite-cloud method (Li et al., 2003), stabilized least-square radial point collocation method (Liu et al., 2006a) and the meshfree weak-strong form methods (Liu and Gu, 2003). The current work uses one such method, namely the Hermite-cloud method that is based on the classical reproducing kernel particle method. The reproducing kernel particle method and the Hermite-cloud method based on it are explored in the following sections.

### 2.1.3 Smooth particle hydrodynamics and reproducing kernel particle methods

Smooth particle hydrodynamics method (Gingold and Monaghan, 1977; Lucy, 1977), is a meshless technique that uses the integral transform technique to develop an approximate solution  $f^h(x)$  of a field variable  $f(x)$  as

$$f^h(x) = \int_{\Omega} K(x - \xi, h) f(\xi) d\xi \quad (2.1)$$

where  $K(x - \xi, h)$  is known as the kernel, weighting or the smoothing function,  $h$  is the smoothing length that determines the size of the smoothing domain  $\Omega$ , also known as the support or influence domain. The field value at a given point is then approximated as the weighted sum of the field variable values,  $f$ , within the support domain  $\Omega$ . The kernel function should be chosen in such a way that it satisfies certain conditions, in order for Eq. (2.1) to be valid. The kernel function should be compact meaning that  $K(x - \xi, h)$  should be equal to zero outside the support domain, it should be orthogonal and its integral over the domain  $\Omega$  should be unity. Some possible choice of window functions include Gaussian functions, wavelets and splines. The SPH method was mainly used to solve problems without boundaries, such as those in the areas of astrophysics.

The major problem associated with the SPH method is its inability to provide accurate results at the boundaries (Monaghan, 1988). The Reproducing kernel methods (RKPM) developed by Liu et al. (1995b) addresses this issue by introducing a correction function in the SPH method, in addition to the kernel function, for construction of the window function. The correction function introduced in RKPM results in an improved and accurate solution at the boundaries, adds stability to the solution and also ensures linear or  $C^1$  consistency near the boundary (Liu et al., 1996a; Liu and Jun, 1998). In the RKPM, The field variable  $f(x)$  is approximated

as

$$f^h(x) = \int_{\Omega} C(x, \xi) K(x - \xi, h) f(\xi) d\xi \quad (2.2)$$

where  $C(x, \xi)$  is the correction function that is developed based on the concept that any function can be represented as a sum of linearly independent functions (Liu et al., 1995a). A typical choice for the correction function is given as

$$C(x, \xi) = C_1(x) + C_2(x)(\xi - x) \quad (2.3)$$

where the correction co-efficients  $C_1$  and  $C_2$  are determined by defining moments over a region  $B(x)$ , where the kernel function is not zero. The correction function takes a values of unity in the interior of the domain and therefore has no effects in the interior. It is mainly introduced to provide accurate solutions at the boundaries, and also adds stability to the kernel (Liu et al., 1995a). RKPM has been widely used in solving problems in many areas of engineering (Liu et al., 1996b). The Hermite-cloud method, used for continuum discretization in this work, is based on the RKPM and is discussed in detail in the next section.

#### 2.1.4 Hermite-cloud method

The Hermite-Cloud method (Li et al., 2003) is a strong-form meshless method that uses the classical reproducing kernel particle method (RKPM), the point collocation technique and the Hermite interpolation theorem to obtain an approximate solution for both the field variable and its first-order derivative. As highlighted in the previous section, the classical RKPM constructs an approximate solution,  $f^h(x)$  or  $f^h(x, y)$  corresponding to unknown functions  $f(x)$  and  $f(x, y)$  respectively, using

the correction and kernel functions as

$$f^h(x, y) = \int_{\Omega} C(x, s) K(x - s) f(s) ds \quad (2.4)$$

$$f^h(x, y) = \int_{\Omega} C(x, y, s, t) K(x - s, y - t) f(s, t) ds dt \quad (2.5)$$

The Hermite-Cloud method uses the fixed RKPM, where the kernel function  $k(x - s, y - t)$  is fixed about the center point  $(x_k, y_k)$ , and the approximate solution given in Eq. (2.5) is modified as

$$f^h(x, y) = \int_{\Omega} C(x, y, s, t) K(x_k - s, y_k - t) f(s, t) ds dt \quad (2.6)$$

The kernel function in Eq. (2.6) is constructed by a suitable choice of a weighted window function. In the Hermite-Cloud method, a cubic spline window function is chosen for constructing the kernel function as

$$K(x_k - s, y_k - t) = [W^*((x_k - s)/\Delta x) W^*((y_k - t)/\Delta y)]/(\Delta x \Delta y) \quad (2.7)$$

where  $W^*(z)$  is the cubic spline window function of the form

$$W^*(z) = \begin{cases} 0 & |z| \geq 2 \\ (2 - |z|)^3/6 & 1 \leq |z| \leq 2 \\ (2/3) - z^2(1 - 0.5|z|) & |z| \leq 1 \end{cases} \quad (2.8)$$

The parameter  $z$  is defined as  $z = (x_k - s)/\Delta x$  or  $z = (y_k - s)/\Delta y$ , where  $\Delta x$  and  $\Delta y$  correspond to the cloud size, with respect to the center point  $(x_k, y_k)$ , along the  $x$  and  $y$  directions respectively. The cloud size may be varied depending on the problem and the accuracy level desired.

Next, the correction function  $C(x, y, s, t)$  in Eq. (2.6) is constructed as a product of

the row basis function vector  $B(s, t)$  and column coefficient vector  $C^*(x, y)$  as

$$C(x, y, s, t) = B(s, t)C^*(x, y) \quad (2.9)$$

The basis function vector is constructed from a set of linearly independent polynomial basis functions as  $B(s, t) = \{b_1(s, t), b_2, \dots, b_m\}$ , where  $m$  is the degree of the polynomial basis function. The basis functions vector is appropriately chosen based on the dimension and order of the problem to be solved. For example, to solve a second order partial differential equation, the basis function vector in one and two space dimensions may be chosen as

$$B(s) = \{1, s, s^2\}; m = 3 \quad (2.10)$$

$$B(s, t) = \{1, s, t, s^2, st, t^2\}; m = 6 \quad (2.11)$$

The column coefficient vector  $C^*(x, y) = \{c_1, c_2, \dots, c_m\}^T$  is a  $m \times 1$  vector of correction function coefficients, which are the unknowns and are determined using the fact that Eq. (2.6) satisfies the consistency condition for any basis function  $b_i(x, y)$  as

$$b_i(x, y) = \int_{\Omega} C(x, y, s, t)K(x_k - s, y_k - t)b_i(s, t)dsdt \quad (i = 1..m) \quad (2.12)$$

As the governing PDE is discretized using a strong-form approach, Eq. (2.6) above can be written in the discrete form as

$$\begin{aligned} b_i(x, y) &= \sum_{n=1}^{NP} C(x, y, s_n, t_n)K(x_k - s_n, y_k - t_n)b_i(s_n, t_n)\Delta V_n \\ &= \sum_{n=1}^{NP} B(s_n, t_n)C^*(x, y)K(x_k - s_n, y_k - t_n)b_i(s_n, t_n)\Delta V_n \quad i = 1, 2, \dots, m \end{aligned} \quad (2.13)$$

where  $NP$  is the total number of scattered nodes in the interior and boundaries of the domain  $\Omega$ , and  $\Delta V_n$  is the cloud area corresponding to the  $n^{th}$  node. In the Eq. (2.13) above, the basis function vector and the kernel function are known. Therefore the equation can be re-written in matrix form with the correction function coefficients as unknowns

$$\begin{aligned} A(x_k, y_k)C^*(x, y) &= B^T(x, y) \quad \text{or} \\ C^*(x, y) &= A^{-1}(x_k, y_k)B^T(x, y) \end{aligned} \quad (2.14)$$

where  $A$  is a symmetric moment matrix of size  $m \times m$  that is independent of  $x$  and  $y$  and is related to the fixed cloud point  $(x_k, y_k)$  as

$$A_{ij}(x_k, y_k) = \sum_{n=1}^{NP} b_i(s_n, t_n)K(x_k - s_n, y_k - t_n)b_j(s_n, t_n)\Delta V_n \quad i, j = 1, 2, \dots, m \quad (2.15)$$

From Eqs. (2.9) & (2.14), the approximate solution of the unknown function  $f(x, y)$  in Eq. (2.6), can be obtained as

$$\begin{aligned} f^h(x, y) &= \int_{\Omega} B(s, t)C^*(x, y)K(x_k - s, y_k - t)f(s, t)dsdt \\ &= \int_{\Omega} B(s, t)A^{-1}(x_k, y_k)B^T(x, y)K(x_k - s, y_k - t)f(s, t)dsdt \end{aligned} \quad (2.16)$$

which can be written in a discrete form as

$$f^h(x, y) = \sum_{n=1}^{NP} B(s_n, t_n)A^{-1}(x_k, y_k)B^T(x, y)K(x_k - s_n, y_k - t_n)\Delta V_n f_n \quad (2.17)$$

and finally in a concise form as

$$f^h(x, y) = \sum_{n=1}^{NP} N_n(x, y)f_n \quad (2.18)$$

where  $N_n(x, y)$  corresponds to the shape function of the  $n^{th}$  node. The shape

functions are polynomial functions in the  $x$  and  $y$  directions, and their derivatives are obtained by differentiating the basis function vector. The shape functions satisfy the consistency condition for all the independent basis functions. As an example, if the basis function vector is chosen as  $B(s, t) = \{1, x, y\}$ , the consistency condition in Eq. (2.13) can be written as

$$\sum_{n=1}^{NP} N_n(x, y) = 1 \quad (2.19)$$

$$\sum_{n=1}^{NP} N_n(x, y) x_n = x \quad (2.20)$$

$$\sum_{n=1}^{NP} N_n(x, y) y_n = y \quad (2.21)$$

The first-order derivatives of the unknown function  $f(x, y)$  can also be approximated in a similar fashion as

$$f_x^h(x, y) = \sum_{m=1}^{NT} M_m(x, y) f_{xm} \quad (2.22)$$

$$f_y^h(x, y) = \sum_{m=1}^{NT} M_m(x, y) f_{ym} \quad (2.23)$$

where  $NT$  is the total number of scattered nodes in  $\Omega$  with  $NT \leq NP$ , and  $M_m(x, y)$  are the shape function corresponding to the first-order derivatives  $f_x(x, y)$  and  $f_y(x, y)$  of the unknown function  $f(x, y)$  at the  $m^{th}$  nodes, which are constructed in a similar fashion as  $N_n(x, y)$ . Using the Hermite interpolation theorem, an approximate solution of the function  $f(x, y)$  can be finally obtained as

$$\begin{aligned} f^h(x, y) = & \sum_{n=1}^{NP} N_n(x, y) f_n + \sum_{m=1}^{NT} \left( x - \sum_{n=1}^{NP} N_n(x, y) x_n \right) M_m(x, y) f_{xm} \\ & + \sum_{m=1}^{NT} \left( y - \sum_{n=1}^{NP} N_n(x, y) y_n \right) M_m(x, y) f_{ym} \end{aligned} \quad (2.24)$$

In order to evaluate the additional unknowns  $f_x(x, y)$  and  $f_y(x, y)$ , an additional set of auxiliary conditions are introduced. These are obtained by taking the partial



derivatives of the Eq. (2.24) with respect to variables  $x$  and  $y$  as

$$\begin{aligned}
 f_{,x}^h(x, y) = & \sum_{n=1}^{NP} N_{n,x}(x, y) f_n + \\
 & + \sum_{m=1}^{NT} \left( 1 - \sum_{n=1}^{NP} N_{n,x}(x, y) x_n \right) M_m(x, y) f_{xm} - \sum_{m=1}^{NT} \left( \sum_{n=1}^{NP} N_{n,x}(x, y) y_n \right) M_m(x, y) f_{ym} \\
 & + \sum_{m=1}^{NT} \left( x - \sum_{n=1}^{NP} N_n(x, y) x_n \right) M_{m,x}(x, y) f_{xm} + \sum_{m=1}^{NT} \left( y - \sum_{n=1}^{NP} N_n(x, y) y_n \right) M_{m,x}(x, y) f_{ym}
 \end{aligned} \tag{2.25}$$

$$\begin{aligned}
 f_{,y}^h(x, y) = & \sum_{n=1}^{NP} N_{n,y}(x, y) f_n + \\
 & + \sum_{m=1}^{NT} \left( 1 - \sum_{n=1}^{NP} N_{n,y}(x, y) y_n \right) M_m(x, y) f_{ym} - \sum_{m=1}^{NT} \left( \sum_{n=1}^{NP} N_{n,y}(x, y) x_n \right) M_m(x, y) f_{xm} \\
 & + \sum_{m=1}^{NT} \left( x - \sum_{n=1}^{NP} N_n(x, y) x_n \right) M_{m,y}(x, y) f_{xm} + \sum_{m=1}^{NT} \left( y - \sum_{n=1}^{NP} N_n(x, y) y_n \right) M_{m,y}(x, y) f_{ym}
 \end{aligned} \tag{2.26}$$

where  $f_{,x}^h(x, y)$  and  $f_{,y}^h(x, y)$  are the approximate partial derivatives of the approximate solution defined in Eq. (2.24), with subscripts after comma indicating differentiation with respect to variables  $x$  or  $y$  direction. The above set of equation can be simplified using Eqs. (2.19)-(2.23) as

$$\begin{aligned}
 & \sum_{n=1}^{NP} N_{n,x}(x, y) f_n - \sum_{m=1}^{NT} \left( \sum_{n=1}^{NP} N_{n,x}(x, y) x_n \right) M_m(x, y) f_{xm} - \\
 & \sum_{m=1}^{NT} \left( \sum_{n=1}^{NP} N_{n,x}(x, y) y_n \right) M_m(x, y) f_{ym} = 0
 \end{aligned} \tag{2.27}$$

$$\begin{aligned}
 & \sum_{n=1}^{NP} N_{n,y}(x, y) f_n - \sum_{m=1}^{NT} \left( \sum_{n=1}^{NP} N_{n,y}(x, y) y_n \right) M_m(x, y) f_{ym} - \\
 & \sum_{m=1}^{NT} \left( \sum_{n=1}^{NP} N_{n,y}(x, y) x_n \right) M_m(x, y) f_{xm} = 0
 \end{aligned} \tag{2.28}$$

where  $N_{n,x}(x, y)$  and  $N_{n,y}(x, y)$  are the first-order derivatives of the shape function  $N_n(x, y)$  with respect to the  $x$  and  $y$  directions, respectively. The derivatives of the

shape function can be obtained by differentiating the basis function vector. The auxiliary conditions are mainly introduced to solve for the additional unknowns  $f_x(x, y)$  and  $f_y(x, y)$ .

Finally, the approximate solution defined in Eqs. (2.22)-(2.24) along with the auxiliary equations defined in Eqs. (2.27)-(2.28) are solved together to obtain the approximate solutions  $f^h(x, y)$ ,  $f_x^h(x, y)$ , and  $f_y^h(x, y)$  corresponding to the function  $f(x, y)$  and its first-order derivatives  $f_x(x, y)$  and  $f_y(x, y)$ , respectively.

### Problem discretization

In the Hermite-Cloud method, the *PDE* system is discretized using the point collocation method (Russell and Shampine, 1972), wherein the governing equation and the boundary conditions are satisfied at all the collocation points in the domain. Consider a partial differential boundary value problem defined in the domain  $\Omega$  as

$$Lf(x, y) = F(x, y) \quad (2.29)$$

$$f(x, y) = G(x, y) \quad \text{along } \Gamma_D \quad (2.30)$$

$$\partial f(x, y)/\partial n = H(x, y) \quad \text{along } \Gamma_N \quad (2.31)$$

where  $L$  is a differential operator, and  $\Gamma_D$ ,  $\Gamma_N$  are the Dirichlet and Neumann boundaries. Using the point collocation approach, Eqs. (2.29)-(2.31) can be discretized at a scattered point  $(x_i, y_i)$  as

$$Lf^h(x_i, y_i) = F(x_i, y_i) \quad i = 1, \dots, N_\Omega \quad (2.32)$$

$$f^h(x_i, y_i) = G(x_i, y_i) \quad i = 1, \dots, N_D \quad (2.33)$$

$$\partial f^h(x_i, y_i)/\partial n = H(x_i, y_i) \quad i = 1, \dots, N_N \quad (2.34)$$

where  $f^h(x_i, y_i)$  is the approximate solution of the function  $f(x, y)$  and  $N_\Omega$ ,  $N_D$ ,  $N_N$  are the number of scattered points inside the domain and along the Dirichlet and

Neumann boundaries, respectively. The approximate solutions defined in Eqs. (2.22)-(2.24) are substituted into the discretized form of the PDE defined in Eqs. (2.32)-(2.34), and combined with the auxiliary conditions defined in Eqs. (2.27)-(2.28), leading to a set of algebraic equations corresponding to the unknown variables  $f_i$ ,  $f_{xi}$  and  $f_{yi}$ , which can be expressed in a matrix form as

$$[A_{ij}]_{(NP+2NT) \times (NP+2NT)} \{X_i\}_{(NP+2NT) \times 1} = \{B_i\}_{(NP+2NT) \times 1} \quad (2.35)$$

where  $A$  is the coefficient matrix, which can be constructed from the discretized form of PDE defined in Eqs. (2.32)-(2.34) and the auxiliary conditions defined in Eqs. (2.27)-(2.28) as follows. Consider a node  $i$ , in the interior of the domain and not subject to any boundary condition, with the nodal location given as  $(x_i, y_i)$ . The entries in the  $i^{th}$  row and  $j^{th}$  column of matrix  $A$  is then given as

$$\begin{aligned} A_{ij} = & \left[ [LN_j(x_i, y_i)]_{1 \times NP}, \left[ L \left( \left( x_i - \sum_{n=1}^{NP} N_n(x_i, y) x_n \right) M_j(x_i, y_i) \right) \right]_{1 \times NT}, \dots \right. \\ & \left. \dots \left[ L \left( \left( y_i - \sum_{n=1}^{NP} N_n(x_i, y) y_n \right) M_j(x_i, y_i) \right) \right]_{1 \times NT} \right] \end{aligned} \quad (2.36)$$

where  $N_j$  and  $M_j$  are the shape function computed at the  $j^{th}$  node. Similarly, for a boundary node  $b$ , subject to Dirichlet or Neumann boundary condition, the corresponding  $b^{th}$  row entries in the  $A$  matrix are given as

$$\begin{aligned} A_{dj} &= [[N_j(x_i, y_i)]_{1 \times NP}, [0]_{1 \times NT}, [0]_{1 \times NT}] \\ A_{nj} &= [[0]_{1 \times NP}, [M_j(x_i, y_i)]_{1 \times NT}, [M_j(x_i, y_i)]_{1 \times NT}] \end{aligned} \quad (2.37)$$

with row index  $d$  and  $n$  corresponding to Dirichlet and Neumann boundary condition respectively. Finally, the entries corresponding to the auxiliary conditions are added

to the coefficient matrix making it complete.

$$\begin{aligned}
 A_{axj} = & \left[ [N_{j,x}(x_i, y_i)]_{1 \times NP}, \left[ - \left( \sum_{n=1}^{NP} N_{n,x}(x_i, y_i) x_n \right) M_j(x_i, y_i) \right]_{1 \times NT}, \dots \right. \\
 & \left. \dots \left[ - \left( \sum_{n=1}^{NP} N_{n,x}(x_i, y_i) y_n \right) M_j(x_i, y_i) \right]_{1 \times NT} \right] \quad (2.38)
 \end{aligned}$$

$$\begin{aligned}
 A_{ayj} = & \left[ [N_{j,y}(x_i, y_i)]_{1 \times NP}, \left[ - \left( \sum_{n=1}^{NP} N_{n,y}(x_i, y_i) x_n \right) M_j(x_i, y_i) \right]_{1 \times NT}, \dots \right. \\
 & \left. \dots \left[ - \left( \sum_{n=1}^{NP} N_{n,y}(x_i, y_i) y_n \right) M_j(x_i, y_i) \right]_{1 \times NT} \right] \quad (2.39)
 \end{aligned}$$

where row index  $ax$  and  $ay$  corresponding to Eqs. (2.27) and (2.28) respectively.

The unknown column vector  $X$ , in Eq. (2.35) includes the field variable  $f(x, y)$  and its first-order derivatives  $f_x(x, y)$  and  $f_y$ , and is given as

$$\{X_i\}_{(NP+2NT \times 1)} = \{\{f_i\}_{(1 \times NP)}, \{f_{xi}\}_{(1 \times NT)}, \{f_{yi}\}_{(1 \times NT)}\}^T \quad (2.40)$$

Finally, the right hand side vector  $B$  is obtained from governing differential equation, Dirichlet and Neumann boundary conditions (Eqs. (2.32)-(2.34)) and the auxiliary equation (Eqs. (2.27)-(2.28)) as

$$\{B_i\}_{(NP+2NT \times 1)} = \{\{F(x_i, y_i)\}_{(1 \times N_\Omega)}, \{G(x_i, y_i)\}_{(1 \times N_D)}, \{H(x_i, y_i)\}_{(1 \times N_N)}, \{0\}_{(1 \times 2NT)}\}^T \quad (2.41)$$

The set of algebraic equations given in Eq. (2.35) are solved using any of the direct or iterative solvers, and finally the results obtained are substituted back in Eqs. (2.22)-(2.24) to obtain approximate solutions of the field variable  $f(x, y)$  and its first-order derivatives  $f_x(x, y)$  and  $f_y(x, y)$  at all the scattered points.

### Numerical validation

The source code used in this work for continuum simulation using the Hermite-cloud method is validated by solving a benchmark test problem. A rectangular domain of dimensions  $L \times H$ , under plane stress condition and subject to pure bending load  $M$ , is considered as shown in Fig. 2.1. The governing equations for this problem are the plane stress equilibrium equations that are given as (Lam et al., 2006)

$$\frac{E}{1-\mu^2} \left\{ \frac{\partial^2 u}{\partial x^2} + \frac{1+\mu}{2} \frac{\partial^2 v}{\partial x \partial y} + \frac{1-\mu}{2} \frac{\partial^2 u}{\partial y^2} \right\} + X = 0 \quad (2.42)$$

$$\frac{E}{1-\mu^2} \left\{ \frac{\partial^2 v}{\partial y^2} + \frac{1+\mu}{2} \frac{\partial^2 u}{\partial x \partial y} + \frac{1-\mu}{2} \frac{\partial^2 v}{\partial x^2} \right\} + Y = 0 \quad (2.43)$$

where  $u$  and  $v$  are the displacements along the  $x$  and  $y$  directions,  $E$  and  $\mu$  are the Young's modulus and Poisson's ratio of the material respectively, and  $X, Y$  are the body forces that are taken as zero for this problem.

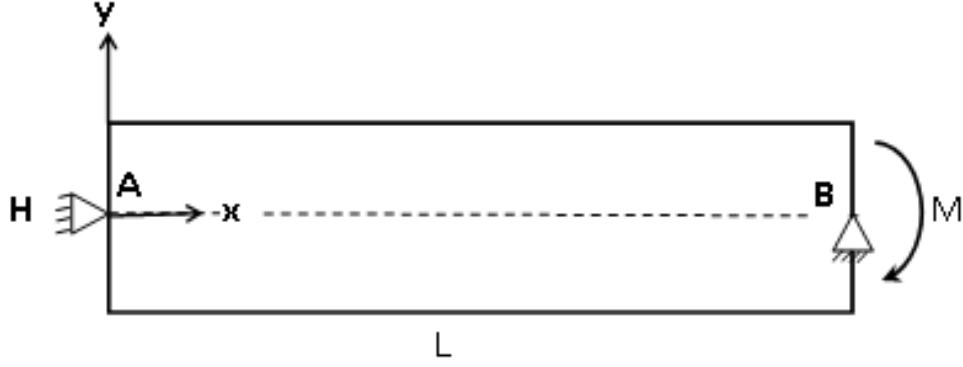
The domain is supported at hinge points  $A$  and  $B$ , and subject to a unit bending moment  $M$ . Both  $x$  and  $y$  displacements are constrained at Point  $A$ , while only the  $y$  displacements are constrained at point  $B$ . The boundary conditions for the problem are given as

$$\sigma_x = \frac{E}{1-\mu^2} \left\{ \frac{\partial u}{\partial x} + \mu \frac{\partial v}{\partial y} \right\} = \frac{My}{I} \quad \text{at } x = 0 \text{ and } L \quad (2.44)$$

$$\tau_{xy} = \frac{E}{2(1+\mu)} \left\{ \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right\} = 0 \quad \text{at } x = 0 \text{ and } L \quad (2.45)$$

$$\sigma_y = \frac{E}{1-\mu^2} \left\{ \frac{\partial v}{\partial y} + \mu \frac{\partial u}{\partial x} \right\} = 0 \quad \text{at } y = \pm H/2 \quad (2.46)$$

$$\tau_{xy} = \frac{E}{2(1+\mu)} \left\{ \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right\} = 0 \quad \text{at } y = \pm H/2 \quad (2.47)$$



**Figure 2.1:** Plane stress problem under pure bending.

with displacement constraints at point  $A$  and  $B$  given as

$$u = 0; \ v = 0 \text{ at } (0,0) \text{ and } v = 0 \text{ at } (L,0) \quad (2.48)$$

where,  $\sigma$  and  $\tau$  correspond to the normal and shear stress respectively, and  $I = H^3/12$  is the moment of inertial for the domain.

The analytical solution for this problem is given as (Lam et al., 2006)

$$u = \frac{M}{EI} \left( x - \frac{L}{2} \right) y \quad (2.49)$$

$$v = \frac{M}{2EI} (xL - x^2 - \mu y^2) y \quad (2.50)$$

and the first-order derivatives the displacements, with respect to  $x$  and  $y$  directions, are given as

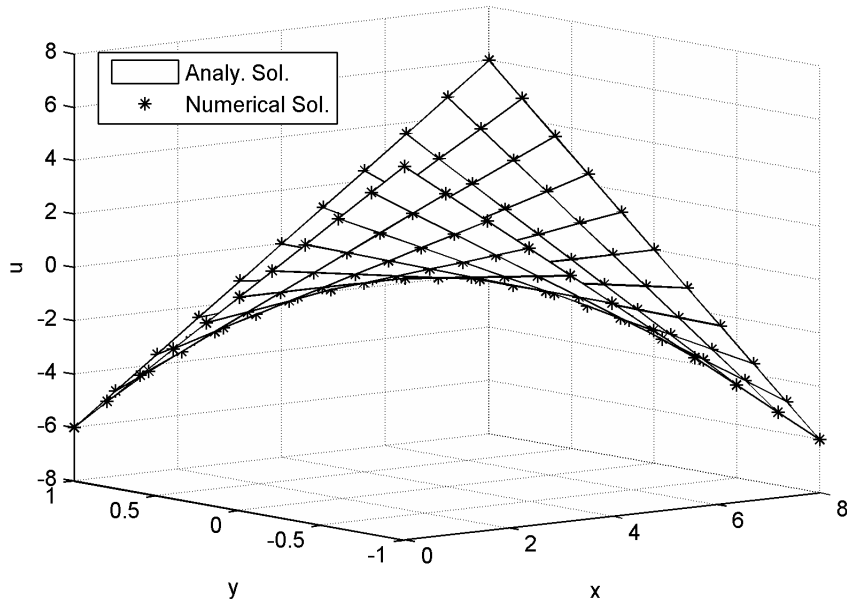
$$\frac{\partial u}{\partial x} = \frac{M}{EI} y \quad (2.51)$$

$$\frac{\partial u}{\partial y} = \frac{M}{EI} \left( x - \frac{L}{2} \right) \quad (2.52)$$

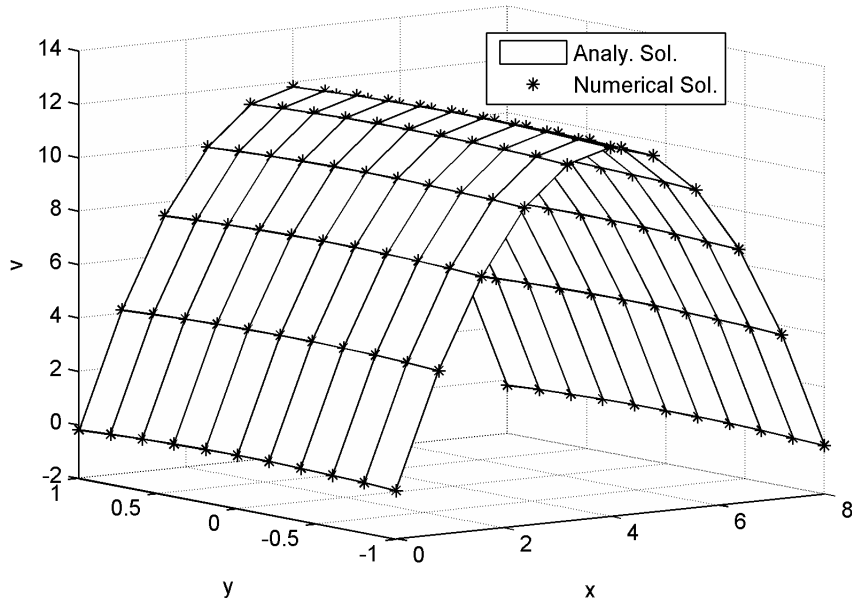
$$\frac{\partial v}{\partial x} = \frac{M}{EI} \left( \frac{L}{2} - x \right) \quad (2.53)$$

$$\frac{\partial v}{\partial y} = -\frac{\mu M}{EI} y \quad (2.54)$$

The Young's modulus and Poisson's ratio for this problem are taken as 1 and 0.25 respectively. The problem is numerically solved using the Hermite-cloud method using a uniform  $11 \times 11$  point distribution. 1. The source codes for the Hermite-Cloud algorithm were developed using Matlab version 7.4, and run on a Pentium Duo Core Machine (3GHz) with 3.25GB RAM. The results, namely the  $u$  and  $v$  displacements and their first-order derivatives, obtained using the Hermite-cloud method are compared with that of the analytical solution in Figs. 2.2 and 2.3. From the figure it can be observed that the numerical results agree well with the analytical solution. This is further supported by the global error value of less than 1%, observed for the solution. The error global error values observed for the first-order derivatives are also less than 1%.



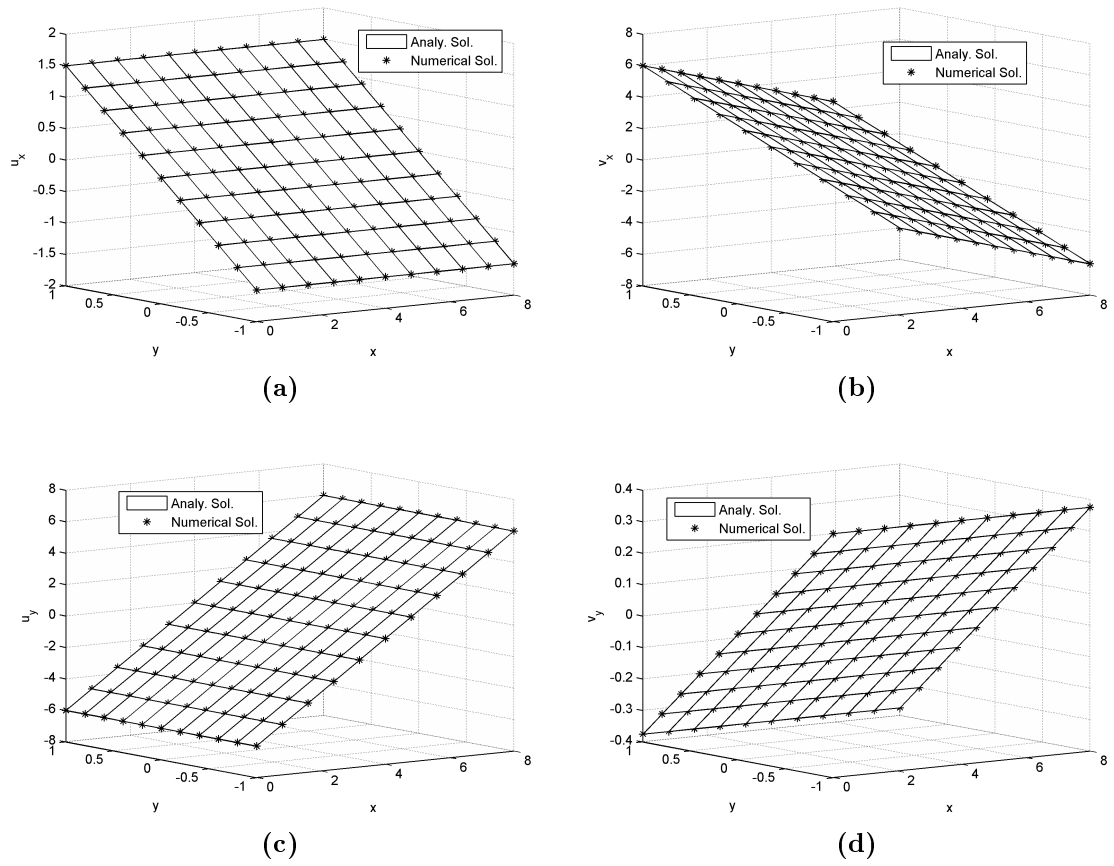
(a)



(b)

**Figure 2.2:** Plane stress problem: comparison of analytical and numerical solution  
(a)  $u$  displacement (b)  $v$  displacement.





**Figure 2.3:** Plane stress problem: comparison of analytical and numerical solution: displacement derivatives (a)  $u_x$  (b)  $v_x$  (c)  $u_y$  (d)  $v_y$  .

## 2.2 Molecular Dynamics

Molecular dynamics, is a simulation technique that follows the time evolution of a set of interacting atoms in a system, which might be in solid liquid or gaseous state. It is like a virtual microscope that provides a snapshot of the position, velocity and orientation of atoms at different time instances. The atoms in the system are thus followed at successive time intervals, as they move, interact and collide with other atoms or the system boundary. The simulation is based the following assumptions (Leach, 2001; Rapaport, 2004; Haile, 1997)

- The atoms/molecules in the system are spherically symmetric and they interact with each other through defined potentials. The trajectory of the atoms are defined by their instantaneous position and velocity. The interaction potential is a continuous function that depends on the instantaneous atomic positions as well as the inter-atomic spacing, also known as soft-sphere model.
- The system is isolated meaning that the system energy, mass, and the number of atoms remain constant with time.

The MD method thus provides solution to an  $N$ -body problem, consisting of atoms/molecules interacting with one another through defined potentials. The equations of motion of the interacting atoms can be written using the Lagrangian function as

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{r}_i} - \frac{\partial L}{\partial r_i} = 0 \quad i = 1 \dots N \quad (2.55)$$

where,  $r_i = (x_i, y_i, z_i)$  is the Cartesian coordinate of an atom  $i$ , with  $\dot{r}$  denoting the time derivative, and  $N$  is the number of atoms in the system. The Lagrangean  $L$  for the system can be computed from the kinetic and potential energies as

$$L = \sum_i \frac{m_i}{2} (\dot{x}_i^2 + \dot{y}_i^2 + \dot{z}_i^2) - U(r_1, r_2, \dots, r_n) \quad (2.56)$$

where  $m_i$  is the atomic mass and  $U$  is the potential energy of the system that is a function of atomic coordinates. Equations (2.55) and (2.56) finally lead to Newton's equation of motion that can be expressed as

$$m_i \ddot{r}_i = -\frac{\partial U(r_1, r_2, \dots, r_n)}{\partial r_i} \quad \text{or} \quad (2.57)$$

$$f_i = \sum_{j=1; j \neq i}^N f_{ij} \quad (2.58)$$

where  $f_i$  is the force on atoms  $i$  due to its interactions with other atoms in the system. If the initial position and velocities are known at a particular time instance, the phase-space trajectory defined in terms of the atomic position and momenta (momentum), can be determined at subsequent time intervals by integrating the equations of motion, Eq. (2.57). Macroscopic properties can then be computed from the atomic trajectories and momenta as time averages. The major steps in an MD simulation involves, modeling of the system by defining appropriate potentials, integrating the equations of motion by applying suitable boundary conditions and finally obtaining macroscopic properties from the atomic trajectories. Each of these steps are discussed briefly in the following sections.

### 2.2.1 Modeling the system - inter-atomic potentials

The first and the most important step in MD simulation is the choice of a potential function to model the system of interacting atoms. Macroscopic properties of a system can be accurately computed by quantum mechanical principles that account for interactions at both atomic as well as subatomic level. However, such simulations are both time consuming and computationally expensive and therefore can simulate only a small number of atoms. MD simulation thus adopts a classical approach to simulate larger number of atoms by the use of inter-atomic potentials. The inter-

atomic potentials are developed based on the Born-Oppenheimer approximation (Leach, 2001), that ignores the electronic motion within the atoms and computes the energy of the system only based on nuclear positions. This assumption is valid as the mass and the time scale with which electrons move are much smaller than that of the nuclei. The electronic degrees of freedom are thus incorporated into the potential function, which is a function of atomic coordinates.

The potential function of an atomic system is generally of the form

$$U(r_1, r_2, \dots, r_n) = \sum_i \phi(r_i) + \sum_{i,j>i} \phi(r_i, r_j) + \sum_{i,j>i,k>j} \phi(r_i, r_j, r_k) + \dots \quad (2.59)$$

where  $r_n$  is the radius of the  $n^{th}$  atom,  $\phi_m$  is an  $m$ -body potential which is the sum of energies due to external forces, pair-wise interaction, three body interaction etc. The first term that represents the energy due to external forces are usually ignored and multi-body interactions are usually restricted to three, in order to reduce computational costs.

Pair-wise interactions model the attractive and repulsive forces between atomic nuclei surrounded by negatively charged electrons, also known as the electron cloud. A popular model to describe pair-wise interactions is the Lennard-Jones (LJ) 12 – 6 potential given as

$$\phi(r_{ij}) = 4\epsilon \left[ \left( \frac{\sigma}{r_{ij}} \right)^{12} - \left( \frac{\sigma}{r_{ij}} \right)^6 \right] \quad r_{ij} = r_i - r_j \quad (2.60)$$

where  $r_{ij}$  is the distance between the atoms  $i$  and  $j$ ,  $\sigma$  represents equilibrium distance, the distance at which the energy  $\phi(r)$  is zero meaning that the attractive and repulsive forces are in equilibrium, and  $\epsilon$  is the well depth (see Fig. 2.4) that indicates how strongly the atoms attract each other. The  $1/r^{12}$  term in the potential represents short range repulsive forces and is based on Pauli's principle, while long range attractive forces represented by the  $1/r^6$  term is based on van der waals forces.

The potential has unit of  $\epsilon$ , which is electron volt (eV).

The force between atoms can be computed as the derivative of the potential function with respect to inter-atomic distance as

$$f_{ij} = -\frac{\partial U(r)}{\partial r} = \left(\frac{48\epsilon}{\sigma^2}\right) \left[ \left(\frac{\sigma}{r_{ij}}\right)^{14} - \frac{1}{2} \left(\frac{\sigma}{r_{ij}}\right)^8 \right] r_{ij} \quad (2.61)$$

Computing the forces between atoms using the above equation requires  $N^2$  computations. In order to save computational cost and make the computer implementation of the algorithm simpler, a cutoff radius  $r_c$  is introduced, and an atoms are considered to interact only within the cutoff radius. Considering only pairwise interactions the potential function in Eq. (2.59) can be written as

$$U(r) = \begin{cases} \phi(r_{ij}) & r \leq r_c \\ 0 & r > r_c \end{cases} \quad (2.62)$$

A cutoff radius of  $2.5\sigma$  and  $3.2\sigma$  are commonly used in simulations that use LJ potential. The 12 – 6 LJ potential is widely used pair-wise potential for simulating problems in the field of engineering and science.

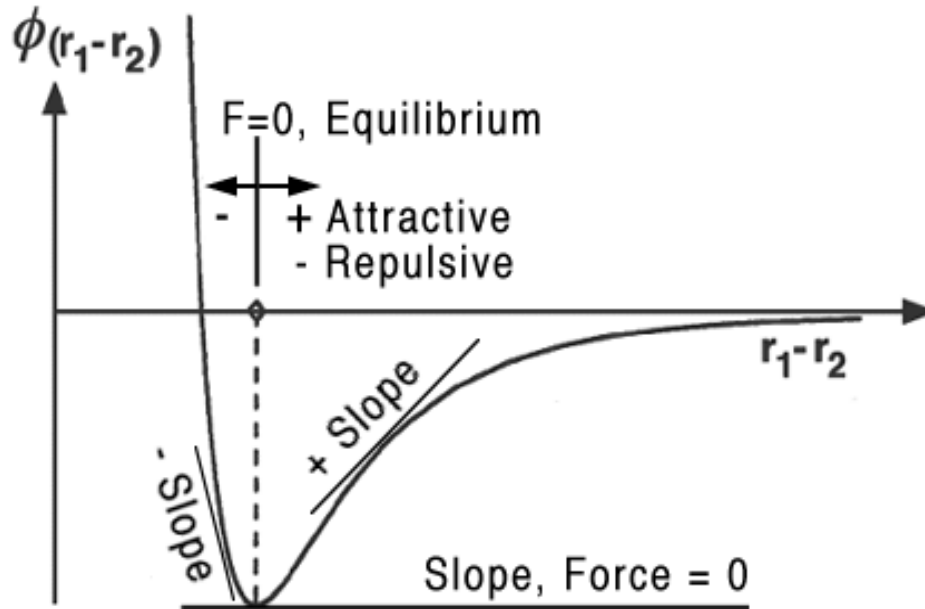
The other commonly used potential to simulate pair-wise interaction is the Morse potential (see Fig. 2.5)

$$\phi(r_{ij}) = D_o \left( e^{-2\alpha(r_{ij}-r_o)} - 2e^{-\alpha(r_{ij}-r_o)} \right) \quad (2.63)$$

$$f_{ij} = 2\alpha D_o \left( e^{-2\alpha(r_{ij}-r_o)} - e^{-\alpha(r_{ij}-r_o)} \right) \quad (2.64)$$

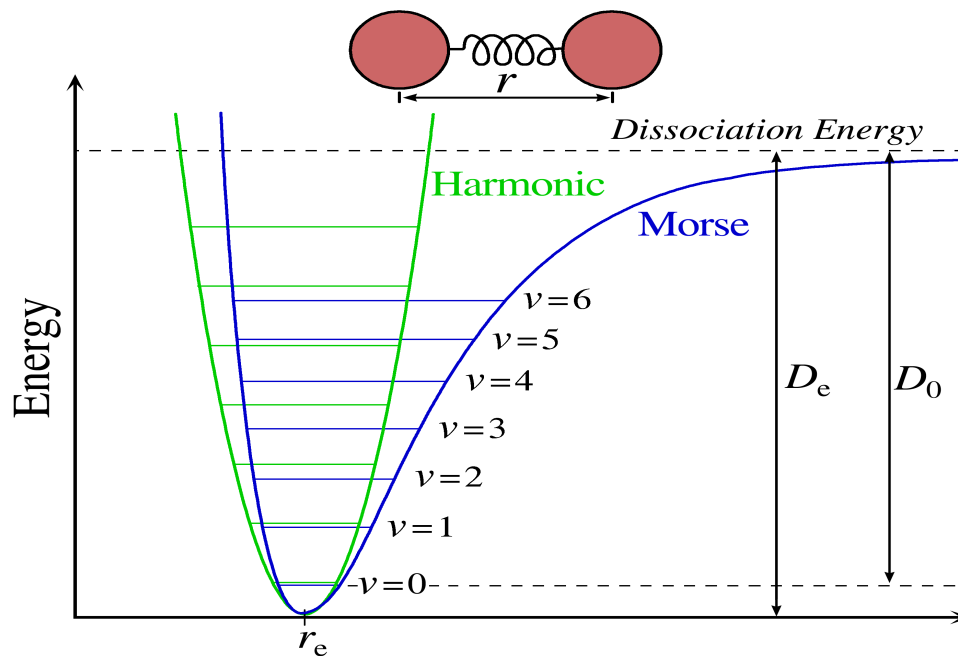
where  $\phi(r_{ij})$  is the potential energy function, and  $D_o$ ,  $\alpha$  and  $r_o$  are potential parameters that correspond to the cohesion energy, elastic modulus and equilibrium bond distance, respectively.

The higher-order interaction terms in the potential function defined in Eq. (2.59)



Source : [http://www.sv.vt.edu/classes/MSE2094\\_NoteBook/MolecDyn/lj.html](http://www.sv.vt.edu/classes/MSE2094_NoteBook/MolecDyn/lj.html)

**Figure 2.4:** 12 – 6 Lennard Jones potential.



Source : [http://www.ask.com/wiki/Morse\\_potential](http://www.ask.com/wiki/Morse_potential)

**Figure 2.5:** Morse potential.

are generally restricted to a maximum of three-body potential due to practical limitations. Commonly used multi-body potentials include the Tersoff potential, Brenner potential and the embedded atom model (EAM) potentials.

Finally, the interaction force, Eqs. (2.61) and (2.64), obtained as the derivative of the energy function, Eqs. (2.60) and (2.63), with respect to the inter-atomic distance, is substituted in the Newton's equation of motion defined in Eq. (2.57), and then numerically integrated to obtain the atomic trajectories and momenta.

## 2.2.2 Initial and boundary conditions

Before starting the simulation, the problem variables have to be initialized. This involves assigning the initial coordinates, velocities and acceleration for the atoms. The initialization is done as follows

- The initial positions are assigned by positioning the atoms at their respective lattice sites in a unit cell. The arrangement is done in such a way that required density ( $\rho = N/V$ ) is achieved, where  $N$  is the total number of atoms in the unit cell and  $V$  is the volume. Generally used lattice structures include simple cubic, face centered cubic (FCC) or body centered cubic (BCC) lattices in  $3D$  and a square or triangular lattice in  $2D$ .
- The velocity vector  $v$  is assigned a magnitude based on the Maxwell distribution at a given temperature as

$$v = \sqrt{2 * T} \quad (2.65)$$

where  $T$  is the temperature, and are assigned random velocity directions. In this work, a uniform random number generator in the interval  $[-1, 1]$  is used for the purpose. The resulting velocities are scaled to ensure that the net

momentum is zero. i.e.,

$$p = \sum_i m_i v_i = 0 \quad (2.66)$$

- The accelerations for all the atoms are initially set to zero

In a macroscopic system, the number of atoms near the system boundary can be neglected in comparison to the total number of atoms. However, the total number of atoms  $N$  in a simulated system, no matter how large the simulated system is, would be much smaller than the number of atoms contained in a macroscopic piece of matter (of the order of  $10^{23}$ ). Therefore, the ratio of the number of atoms near the system boundary to the total number of atoms would be much larger than in reality and therefore boundary effects become important. To carry out simulations that would capture the behavior of interior atoms, without boundary effects, periodic boundary conditions are used, wherein the simulation region is replicated infinitely in all three Cartesian directions.

In enforcing periodic boundary conditions, we assume that any atom/atoms that leaves the simulation region through a bounding face re-enters through the opposite face. Also, the atoms that are within the cut-off radius of a boundary interact with atoms in the adjacent copy of the system or atoms near the opposite side of the boundary, known as the wrap around effect. The periodic boundary conditions are imposed in the simulation by checking the position of the atom after each time step and making necessary adjustments to bring atoms back into the simulation cell. The use of such a boundary condition helps in modeling a system that is bounded and at the same time free of physical boundary.

### 2.2.3 Non-dimensional units

The numbers normally encountered in atomic scale simulations are very small and may introduce round off errors while programming. To avoid this difficulty, MD



simulations are carried out in reduced/dimensionless units in order to work with numbers closer to unity. The use of dimensionless units also permits scaling, wherein the dimensionless model can be used to describe a class of problems and the properties measured in dimensionless units can be appropriately scaled depending on the problem of interest. All quantities such as temperature, pressure, etc., can be expressed in terms of these reduced units.

For the LJ potential defined in Eq. (2.60), the dimensionless terms  $r^*$  and  $t^*$  corresponding to the length and time scales respectively, can be expressed in terms of the LJ potential parameters  $\sigma$  and  $\epsilon$  as

$$r^* = r/\sigma \quad (2.67)$$

$$t^* = t/\sqrt{m\sigma^2/\epsilon} \quad (2.68)$$

The mass in reduced units is taken as one, corresponding to the mass of one atom. The interaction force for the LJ potential defined in Eq. (2.61), can be expressed in reduced units as

$$f_{ij} = 48 \sum_{j=1; j \neq i}^N \left[ \left( \frac{1}{r_{ij}^*} \right)^{14} - \frac{1}{2} \left( \frac{1}{r_{ij}^*} \right)^8 \right] r_{ij}^* \quad (2.69)$$

Other physical properties such as kinetic and potential energies can also be expressed in terms of reduced units in a similar fashion.

## 2.2.4 Velocity rescaling

Thermal energy may be released into or consumed by the system due to numerical errors, choice of the time step, ambient temperature or incorrect initializations. To prevent the temperature of the system from drifting and to establish thermal equilibrium, the velocities need to be rescaled at regular time intervals, and is done as

$$v_i' = \sqrt{\frac{T_d}{T}} v_i \quad (2.70)$$

where  $T_d$  is the desired temperature. The average temperature is used instead of the instantaneous temperature. The frequency between two rescaling has to be determined for each problem. Rescaling is generally not done during the initial phase to avoid possible interference with energy conservation.

### 2.2.5 Numerical integration

The equation of motion given by Eq. (2.57) is an  $N$ -body problem for which an analytical solution is not feasible if the number of atoms exceeds two. Hence, to obtain the position and velocity of the atoms after every time step the equations of motion have to be numerically integrated. Integration algorithms used for this purpose are generally based on finite difference schemes. If the position and the velocity at time  $t$  are known, the integration scheme can be used to compute the same quantities after a time  $t + \Delta t$ . Widely used integration methods in MD simulation include the leapfrog and the predictor corrector methods (Rapaport, 2004). Choosing an appropriate integration scheme is important as it determines the total computational time. A proper choice of an integration algorithm is generally based on the following criteria; it should be computationally efficient, must permit longer time step and must satisfy energy conservation properties.

The leapfrog scheme, based on Taylor's series expansion, is widely used because it is simpler to implement and requires only one force computation per time step, greatly reducing the computational effort. The method also has good energy conservation properties. The truncation error of the algorithm is of the order of  $h^4$  and  $h^2$  in computing displacements and velocities respectively; where  $h = \Delta t$  is the time step of numerical integration. In the leapfrog method the velocities and coordinates are

updated as:

$$v_i(t + h/2) = v_i(t - h/2) + ha_i(t) \quad (2.71)$$

$$r_i(t + h) = r_i(t) + hv_i(t + h/2) \quad (2.72)$$

$$v_i(t) = v_i(t - h/2) + (h/2)a_i(t) \quad (2.73)$$

where  $r_i$ ,  $v_i$  and  $a_i$  correspond to the displacement, velocity and acceleration of the  $i^{th}$  atom respectively. The time step has to be carefully chosen as it needs to be sufficiently small to model the vibrations of atomic bonds correctly but not too small such that the computational effort becomes very high. Typical time steps used in MD simulations is in the order of one femtosecond ( $1\text{fs} = 10^{-15}\text{s}$ ).

The leapfrog algorithm computes the position and velocities at different time steps. To obtain the velocity and position at the same time instance, a two step approach is used. In this approach, the velocities are computed for the first half time step and the coordinate values are updated for one full time step based on these computed velocities. Finally, the velocity values are updated for the next half time step based on the new acceleration values. The two step approach, used in this work, is given as

$$v_i(t + h/2) = v_i(t) + (h/2)a_i(t) \quad (2.74)$$

$$r_i(t + h) = r_i(t) + hv_i(t + h/2) \quad (2.75)$$

$$v_i(t + h) = v_i(t + h/2) + (h/2)a_i(t + h) \quad (2.76)$$

## 2.2.6 Measuring physical properties

The final step in MD simulation is evaluating macroscopic properties from the microscopic information such as the atomic position and velocities, and is done using the principles of statistical mechanics. Statistical mechanics provides the

mathematical base to relate the motion and distribution of atoms in a  $N$ -body system to the macroscopic observables such as pressure, temperature, energy etc.

The thermodynamic state of a system is defined in terms of properties of the system such as temperature, pressure, etc. The microscopic state of the system is defined in terms of the atomic positions  $r$  and momenta  $p$ , that form the coordinates of phase space. The ensemble is a collection of points in the phase space corresponding to a particular thermodynamic state. An ensemble is thus a collection of all possible microscopic states of system that correspond to a thermodynamic state. There are different ensembles possible such as the micro canonical ensemble ( $NVE$ ), canonical Ensemble ( $NVT$ ), Isobaric-Isothermal Ensemble ( $NPT$ ), etc. The commonly used micro-canonical ( $NVE$ ) ensemble is characterized by a fixed number of atoms,  $N$ , a fixed volume,  $V$ , and a fixed total energy,  $E$ , corresponding to an isolated system.

The ensemble average is the average taken over different microscopic states of a system, evolving over time with different initial conditions and is given as  $\langle A \rangle_{ensemble} = A(p^N, r^N)$ , where  $A$  is the macroscopic property. Computing the ensemble averages for a macroscopic sample, containing a large number of atoms, is very difficult as all possible microscopic states of the system have to be known. Hence in MD simulations, the physical properties, which are a function of the particle coordinates and velocities, are obtained as time averages, expressed as

$$\langle A \rangle_{time} = \frac{1}{M} \sum_{t=1}^M A(p^N, r^N) \quad (2.77)$$

where  $t$  is the simulation time and  $M$  is the number of time steps. The time average thus calculated can be approximated to the ensemble average by the use of Ergodic hypothesis. As per this hypothesis, given sufficient time, the time average can be equated to the ensemble average or  $\langle A \rangle_{ensemble} = \langle A \rangle_{time}$ . As the simulation is done over a sufficient period of time, the accuracy of this hypothesis is ensured because, with indefinite time, the system will pass through all possible states. The

time averages for the potential and kinetic energy can be expressed as

$$KE = \langle KE \rangle = \frac{1}{2M} \sum_{j=1}^M \left\{ \sum_{i=1}^N m_i v_i^2 \right\} \quad (2.78)$$

$$PE = \langle PE \rangle = \frac{1}{M} \sum_{i=1}^M V_i \quad (2.79)$$

where  $V_i$  is the potential energy function and  $m_i, v_i$  are the mass and velocity of the  $i^{th}$  atom respectively.

### 2.2.7 Limitations of MD simulation

With the rapid growth in computation resources and the availability of accurate inter-atomic potentials, MD simulations have emerged as a successful tool to simulate complex physical phenomena such as fracture, defects in crystals, friction, biological systems such as DNA, RNA, and also to study the electronic properties of materials. However, the method suffers from a few shortcomings (Liu et al., 2004)

- The length and time scales that can be analyzed using MD simulations are still limited. Even with the most advanced computers available today, MD simulations can handle at most billion atoms which in spatial scale is still in the sub-microns range. Problems in the nanoscale regime that involve billions of atoms and span several microns are still very large for MD simulations. Similarly, the maximum temporal scales that can be simulated in MD simulation is only of the order of nanoseconds. The time scale can be only increased by proportionally reducing the number of atoms, and the development of parallel MD algorithms is still a great challenge. It is therefore difficult to compare the results of MD simulation with that of laboratory experiments that carried out on much larger length and times scales.
- In MD simulation, only a small representative region of the total system

is simulated, due to computational constraints. To account for the region surrounding this representative region rigid/periodic boundary conditions are used. The results obtained from such a simplification might not exactly simulate the actual macroscopic system.

- The major step in the MD simulation involves determination of the forces, which are obtained as the derivative of the potential energy function. The accuracy of the method therefore depends on the proper choice of the potential and its ability to mimic the behavior of the actual system being simulated. Also, the integration algorithm and the time step used for integrating the equations of motion, to a great extent, determine the computational requirement and accuracy of the simulation.

## 2.3 Chapter Summary

This chapter provided an overview of the molecular dynamics method and the meshless Hermite-cloud method used to model the atomistic and continuum domains. From the discussion it can be observed that though the MD method is very effective in simulating atomistic systems, the length and time scales that can be simulated by it are still somewhat constrained by the huge requirement of computational resources. The meshless Hermite-cloud on the other hand is a truly meshless method that eliminates mesh generation completely and is algorithmically simple to implement. It would therefore be wise to combine the best of both approaches, by developing a multiscale model that retains atomistic precision where required and uses the meshless framework elsewhere. This would permit the study systems with larger length and time scales, which is otherwise not feasible with a full MD model. The major issue in the development of a multiscale model is to devise an efficient coupling algorithm to couple different scales in the problem. This would be the focus of the discussion in the next chapter.

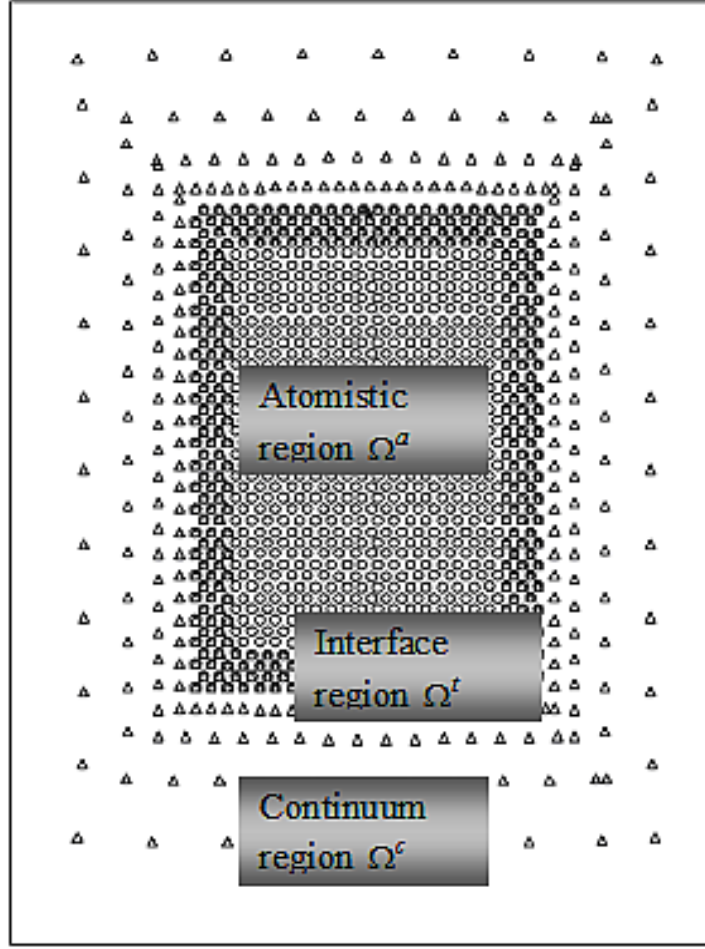
# Chapter 3

## Atomic-Continuum Coupling

The coupling algorithm to couple atomistic and continuum domains is discussed in detail in this chapter. First, the general framework of the multiscale problem and the role of the transition region in ensuring seamless transfer of information between the two domains is discussed. Domain decomposition methods, and specifically the Schwarz alternating scheme, based on which the coupling algorithm in this work is developed, is discussed in the next section. The last section presents the handshaking/coupling algorithm that provides the necessary interface between the atomic and continuum domains, which are discretized using the molecular dynamics (MD) and Hermite-cloud methods respectively, and explained in Sections 2.2 and 2.1.4. This is followed by the chapter summary.

### 3.1 General Framework and Classification of Multiscale Problems

The general framework of a multiscale problem is shown Fig. 3.1. In the figure, circles and triangles correspond to atoms and meshless nodes respectively. The problem domain is divided into atomic and continuum subdomains  $\Omega^a$  and  $\Omega^c$  such



**Figure 3.1:** General framework of multiscale problem.

that  $\Omega = \Omega^a \cup \Omega^c$ , where subscripts  $a$  and  $c$  refer to the atomistic and continuum domains respectively. The continuum domain may be either treated using the finite element or meshless approach, while the atomistic region is treated using the MD method. The transition region  $\Omega^t = \Omega^a \cap \Omega^c$ , is the region where the atomic and continuum domains overlap, and contains both atoms and meshless nodes. Compatibility and equilibrium conditions are enforced in the interface region by prescribing appropriate boundary conditions.

The interface region that couples the two domains is further subdivided into handshaking and padding regions, depending on the problem to be solved. The handshaking region is where atoms and finite element/meshless nodes co-exist. The padding region on the other hand ensures that the atoms in the atomistic region



have their complement of neighbors. The thickness of the padding region is mainly dictated by the cut-off length  $r_{cut}$ , used in the inter-atomic potential and the motion of padding atoms is governed by the continuum displacement field.

Displacement compatibility at the interface may be enforced in different ways for different models. In certain multiscale models, the continuum mesh is refined such that the finite element/meshless nodes coincide with the atoms. If the continuum region is discretized using a finite element mesh, it has to be fully refined at the interface region such that there is a one-to-one correspondence between finite element nodes and atoms. This condition is more easily achieved when using meshless nodes. A coarse mesh may be used in the regions away from the interface. The one-to-one correspondence may not be necessary in certain class of multiscale problems.

Multiscale models may be fundamentally classified, based on the modeling criteria, as follows

1. *Energy/Force based approach.* This classification is based on the method used to achieve equilibrium in the problem. In energy-based methods, an energy functional is defined for the entire problem and equilibrium is achieved by minimizing this energy functional. In force-based methods, the forces are determined at all degrees of freedom and equilibrium is achieved by forcing these forces to zero. In energy based methods spurious forces, also known as ghost forces, are generated in the transition region, due to the mismatch between the non-local and local nature of atoms and nodes in the transition region, while it is automatically taken care of in force-based methods. However, the problem with force-base methods is that there is no well defined energy functional and their convergence rates are generally slower compared to energy based methods.
2. *Boundary conditions.* Displacement compatibility is required to ensure proper coupling of atomistic and continuum regions and may be enforced in a

‘strong’ or ‘weak’ sense. In the case of strong compatibility there is a one-to-one correspondence between atoms and nodes in the padding region, and atoms and the nodes move together. In transferring continuum displacement to atomic region, the displacement of a pad atom  $U^\alpha$  is determined from nodal displacement  $U^I$  using the shape function  $N^I$  defined at the  $I^{th}$  node. Similarly, in the reverse direction, the displacement of a set of nodes are determined from the reference co-ordinates of the atoms. However, in this case the displacement compatibility may need to be enforced only on a subset of pad atoms. The major drawback of enforcing strong compatibility is that, if a finite element mesh is used in the continuum region, it has to be fully refined to coincide with that of atomic locations. To overcome this issue, some multiscale methods enforce weak-compatibility, wherein there is no direct correspondence between atoms and nodes and displacement boundary conditions are satisfied only in an average sense. However, these methods take more time for convergence and are generally less accurate.

3. *Continuum model.* The continuum region in the multiscale model can be treated using a finite element or meshless approaches as mentioned earlier. The type of continuum model used depends on the problem to be solved. In simple cases, a linear elastic model may be used with elastic constants modified to ensure compatibility with an atomistic model or a non-linear formulation may be used for more complex problems.

## 3.2 Domain Decomposition Methods

Domain decomposition methods are used in the numerical solutions of partial differential equations, wherein a boundary value problem is solved by splitting it into different boundary value problems on smaller subdomains, and the solution is obtained by iterating the solution between adjacent subdomains. As the problems

on the individual subdomains are independent, these methods are well suited for parallel computing.

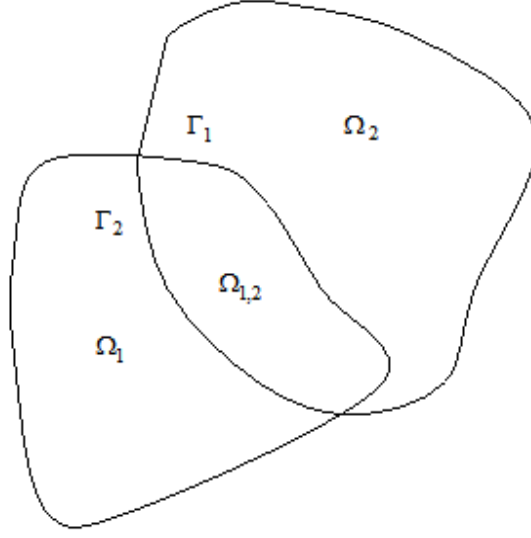
Domain decomposition methods can be classified as overlapping and non-overlapping methods, depending on the nature of the interacting subdomains. In overlapping domain decomposition methods, the subdomains overlap and intersect each other, while in non-overlapping methods, the subdomains only share a common boundary. Examples of overlapping decomposition methods include the Schwarz alternating method and the additive Schwarz method and those in the non-overlapping category include the balancing domain decomposition (BDDC) method and the finite element tearing and interconnect (FETI) method. In non-overlapping methods, the continuity of the solution across the subdomain interface is enforced by representing the value of the solution on all neighboring subdomains by the same unknown. These methods are also known as iterative sub-structuring methods.

In the Schwartz method for overlapping subdomains (Quarteroni and Valli, 2005), used in this work, the problem domain  $\Omega$  is decomposed into overlapping subdomains  $\Omega_1$  and  $\Omega_2$  such that  $\Omega = \Omega_1 \cup \Omega_2$  as shown in Fig. 3.2. Consider a boundary value problem defined in the domain  $\Omega$  as

$$Lu = f \text{ in } \Omega \tag{3.1}$$

$$u = g \text{ on } \partial\Omega \tag{3.2}$$

where  $L$  is the differential operator,  $\partial\Omega$  the problem boundary, and  $\Gamma_1$  and  $\Gamma_2$ , shown in the Fig. 3.2, the boundaries of the overlapping subdomains respectively. The problem is solved iteratively by using an initial starting guess  $u_2^0$  for the values



**Figure 3.2:** An example of overlapping subdomains.

along  $\Gamma_1$  and first solving for  $u_1^n$  as

$$\begin{cases} Lu_1^n = f & \text{in } \Omega_1 \\ u_1^n = g & \text{on } \partial\Omega_1 \setminus \Gamma_1 \\ u_1^n = u_2^{n-1} & \text{on } \Gamma_1 \end{cases} \quad (3.3)$$

and then subsequently solving for  $u_2^n$  as

$$\begin{cases} Lu_2^n = f & \text{in } \Omega_2 \\ u_2^n = g & \text{on } \partial\Omega_2 \setminus \Gamma_2 \\ u_2^n = u_1^n & \text{on } \Gamma_2 \end{cases} \quad (3.4)$$

where  $u_1^n$  and  $u_2^n$  are the solutions at the  $n^{th}$  iteration in subdomains  $\Omega_1$  and  $\Omega_2$  respectively. The iterations are repeated till convergence of solution.

If the boundary value problem in Eq. (3.1) is discretized as  $Au = f$ , the above

iterations can be expressed as

$$u^{n+1/2} = u^n + \begin{pmatrix} A_{\Omega_1}^{-1} & 0 \\ 0 & 0 \end{pmatrix} (f - Au^n) \quad (3.5)$$

$$u^{n+1} = u^{n+1/2} + \begin{pmatrix} 0 & 0 \\ 0 & A_{\Omega_2}^{-1} \end{pmatrix} (f - Au^{n+1/2}) \quad (3.6)$$

where  $A_\Omega$  is the discrete form of the differential operator. The above version is also known as the multiplicative Schwarz method. An improved version of the above method is the additive Schwarz method, that enables parallel computing, and is given as

$$u^{n+1} = u^n + \left[ \begin{pmatrix} A_{\Omega_1}^{-1} & 0 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & A_{\Omega_2}^{-1} \end{pmatrix} \right] (f - Au^n) \quad (3.7)$$

The difference between additive and multiplicative versions of the Schwarz method is similar to the difference between the Jacobi and Gauss-Seidel methods.

### 3.3 Atomic-continuum Coupling Algorithm Based on the Schwarz Alternating Method

The coupling algorithm ensures seamless coupling between the atomic and continuum models and is based on the Schwarz method for overlapping subdomains, discussed in the previous section. Handshaking/coupling is achieved by ensuring the compatibility of both the field variable and its first-order derivative, between the atomic and continuum domains and also force equilibrium in the transition region. The details of the coupling algorithm are presented in this section.

Consider a partial differential boundary value problem  $Lu(x) = P(x)$  defined over the domain  $\Omega$ , with appropriate Dirichlet and Neumann conditions defined along

the boundary. The multiscale algorithm can be explained as follows. The problem domain is first decomposed into the atomic and continuum subdomains  $\Omega^a$  and  $\Omega^c$  respectively, such that  $\Omega = \Omega^a \cup \Omega^c$ , where superscripts  $a$  and  $c$  refer to the atomistic and continuum domains respectively. The atomic domain is restricted to the strategic region where atomic physics dominates, and the remaining region is modeled using the continuum approach. The molecular dynamics and the meshless Hermite-cloud methods, (see Sections 2.2 and 2.1.4), are used to discretize the atomic and continuum domains respectively.

The transition region  $\Omega^t = \Omega^a \cap \Omega^c$ , is the region where the atomic and continuum domains overlap, and contains both atoms and meshless nodes. It is the region where compatibility and equilibrium conditions are enforced. The atomic and continuum domains fully extend into the transition domain. The atomic and continuum domains can therefore be seen as the union of the transition region  $\Omega^t$  containing both atoms and nodes, and a pure atomic  $\Omega^{pa}$  or continuum domain  $\Omega^{pc}$  containing only atoms or nodes respectively. Therefore, the atomic and continuum domains can be mathematically expressed as  $\Omega^a = \Omega^{pa} \cup \Omega^t$  and  $\Omega^c = \Omega^{pc} \cup \Omega^t$ .

Initially, the continuum domain that includes the transition region is solved using the Hermite-cloud method by enforcing appropriate boundary conditions along the transition and problem boundary. To begin with, the boundary conditions in the transition region are obtained from the initial position and velocity of the nodes in the transition region, and for subsequent iterations, the boundary conditions are generated from results of the atomistic simulation. Displacement compatibility between the atomistic and continuum domains is ensured by constraining the atoms in the transition region to follow the displacement of the nodes or

$$u_i^c(X^a) - u_i^a = 0 \quad \text{in } \Omega^t \quad (3.8)$$

where  $u_i^c(X^a)$  is the continuum displacement evaluated at an atomic location  $X^a$

and  $u_i^a$  is the atomic displacement. As the number of nodes and atoms are not equal, this condition may not be exactly satisfied at every point but may be satisfied when the average value is taken over all points in the transition region.

An interpolation function is used to evaluate the continuum displacement at an atomic location. The Hermite-cloud method provides both the field variable and derivative information that can be used to construct piecewise Hermite interpolation polynomials of order  $2n - 1$  between any two nodes in the transition region, where  $n$  is the number of nodes. If we consider two meshless nodes  $x_1$  and  $x_2$  in the transition region with the field variable  $u_1$ ,  $u_2$ , and their corresponding first-order derivatives  $u_{1,x}$  and  $u_{2,x}$ , a Hermite interpolation polynomial can be constructed between them as

$$u(x) = a_0 + a_1x + a_2x^2 + a_3x^3 \quad (3.9)$$

where the coefficients ( $a_0 - a_3$ ) of the interpolation polynomial can be evaluated from the field variable and its first-order derivative information available at both the nodes  $x_1$  and  $x_2$ . Proceeding in a similar fashion, a piecewise interpolation polynomial can be constructed between any two nodes in the transition domain. The interpolation polynomial can now be used to evaluate the value of the field variable at any atomic position in the transition domain and then generate the boundary condition for carrying out the MD simulation. As the Hermite interpolation polynomial matches both the field variable and first-order derivative at the nodal locations, first-order compatibility is ensured between the atomic and continuum domains.

The MD simulation is now carried out and the results are used to generate the boundary conditions for carrying out the continuum simulation. To transfer the information from the atoms to nodes, following an atomistic simulation, we make use of the Lagrange interpolation polynomials (Chapra and Canale, 2002). The Hermite interpolation polynomial discussed above cannot be used in this case, as the atomistic simulation results do not contain the derivative information. In this

case we only have the field variable values at the atomic locations. If we consider two atomic points  $(x_i, x_j)$  with field variable values  $(u_i, u_j)$  in the transition region, a Lagrange polynomial can be constructed between them as

$$u_n(x) = \sum_{i=0}^n L_i(x) u(x_i)$$

where  $n$  is the number of data points and  $R_i(x)$  is given as

$$R_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^n \frac{x - x_j}{x_i - x_j}$$

The interpolation polynomial can then be used to evaluate the field variable values at all the nodal locations in the transition domain.

The MD and continuum simulations are repeated iteratively until convergence is reached within a single time step. Convergence is ascertained by verifying that the norm of the field variable or the displacement values between two successive iterations lies below an acceptable error tolerance

$$||u_{k+1} - u_k||_2 \leq \delta \quad (3.10)$$

where  $u_{k+1}$  and  $u_k$  are the field variable values in the transition domain  $\Omega^t$ , at time steps  $k$  and  $k + 1$  respectively and  $\delta$  is the error tolerance. Mathematically, this is equivalent to solving two different boundary value problems that generate boundary conditions for one another, as in the Schwarz scheme, and can be expressed as

$$L(u_a(x)) = P(x) \text{ in } \Omega^a (\text{includes } \Omega^t) \quad (3.11)$$

$$L(u_c(x)) = P(x) \text{ in } \Omega^c (\text{includes } \Omega^t) \quad (3.12)$$

In addition to displacement compatibility, it is also important for the coupling



scheme to ensure equilibrium of forces in the transition domain. To ensure force equilibrium, we adopt the method used by (Fish et al., 2007), wherein the forces are gradually weakened in the transition region by introducing scaling parameters  $\alpha$  and  $\beta$  such that

$$\begin{aligned}\alpha &= 1 \text{ in } \Omega^c \\ \beta &= 1 \text{ in } \Omega^a \\ \alpha + \beta &= 1 \text{ in } \Omega^t\end{aligned}\tag{3.13}$$

The parameter  $\alpha$  and  $\beta$  linearly vary from 0 to 1 in the transition region. The force on any node  $I$  in the transition region is then computed as the weighted sum of the atomic and continuum forces as

$$F_I = \sum \alpha \{L(u(x^c))\} + \sum_{\xi \neq \eta} \beta_{\eta,\xi} f_{\eta\xi} N^I(x^a)\tag{3.14}$$

where  $f_{\eta\xi}$  is the force exerted by atoms  $\xi$  on atom  $\eta$ ,  $N$  the shape function, and the scaling parameter  $\beta_{\eta,\xi}$  between atoms  $\eta$  and  $\xi$  is given as

$$\beta_{\eta,\xi} = (\beta_\eta + \beta_\xi)/2\tag{3.15}$$

where  $\beta_\eta$  is the scaling parameter evaluated at an atomic location  $X_\eta^a$ . The use of the scaling parameter  $\beta_{\eta,\xi}$  in the atomic domain ensures that the force on atom  $\eta$  due to atom  $\xi$  is the same as the force exerted on atom  $\xi$  due to  $\eta$ , and hence Newton's third law is satisfied (Fish et al., 2007). The coupling algorithm thus ensures both first-order compatibility and force equilibrium in the transition region. The algorithmic steps for solving the problem using the coupling scheme discussed above is listed in Table 3.1.

The coupling algorithm for both static and transient problems is the same, except that in the transient case the algorithmic steps have to be repeated within every time step until the solution within that time step converges. Once convergence is achieved, the solution is marched forward to the next time step, and the procedure

is repeated until the required time.

The major advantage of using interpolation polynomials for transferring information between the nodes and atoms and vice versa is that the atoms and nodes are not required to be coincident in the transition region. In other words, the continuum nodes need not be graded down to coincide with their corresponding atomic points in the transition region, which permits more freedom in the nodal distribution.

### **Multiple time step for numerical integration**

The time scales in the atomic and continuum domains are different; with the time scales in the atomic domain generally being smaller than the continuum time scales. Therefore, it would be computationally advantageous to go for a multi-time step algorithm for the numerical integration of the equations of motion. If the time step for atomic domain is chosen as  $\Delta t^a$ , the time step in the continuum domain  $\Delta t^c$  is chosen as a multiple of the atomic time step as

$$\Delta t^c = n\Delta t^a \tag{3.16}$$

where  $n$  is an integer. Therefore, within each time step, the calculations in the atomic domain are repeated  $n$  times until they match with the continuum time step.

## **3.4 Chapter Summary**

In this chapter, a novel multiscale algorithm was proposed to couple atomistic and continuum domains, which were individually modeled using the molecular dynamics and the strong-form meshless Hermite-cloud methods described in Chapter 2, respectively. The handshaking/coupling method provides a smooth exchange

**Table 3.1:** Multiscale algorithm for atomic-continuum coupling.

---

I	Determine the positions of all the atoms and the meshless nodes in the atomic and continuum domains.
II	Do until the required number of time steps
III	Compute the force on each node in the transition region using Eq. (3.14)
IV	Solve the continuum domain, with boundary conditions enforced the transition and problem boundaries.
V	Use the field variable and first-order derivative information to construct a Hermite polynomial of order $2n - 1$ , between any two nodes nodes in the transition region. Interpolate to obtain the field variable values at all the atomic locations in the transition region.
VI	Constrain the atoms in the transition region to follow the displacement of the nodes and then solve the atomic domain, with boundary conditions enforced the transition and problem boundaries.
VII	Update the values in the transition boundary from the results in step VI and go to step II
VIII	Repeat steps II to VII till solution converges.
IX	Increment the time step
X	End Do
XI	Output simulation results
XII	Stop

---

of information by ensuring both displacement compatibility and force equilibrium in the overlapping transition region. The highlight of the proposed method is that it enables the construction of higher order interpolation polynomials in the transition domain that ensure compatibility of both the field variable and its first-order derivative in the transition region. Also, the use of interpolation polynomials in the transition domain means that nodal distribution in the transition domain need not be fully refined to coincide with the atomic locations, which permits more freedom for the nodal distribution in the continuum domain. In the next chapter, accuracy of the proposed algorithm is verified through benchmark problems in one and two space dimensions.

# Chapter 4

## Results and Discussions

In this chapter, the multiscale model is validated by solving some benchmark test problems and comparing the results with closed form analytical solutions. The problems analyzed using the multiscale model are divided into static test cases that involve only length scale coupling, and transient cases that involve coupling of both time and length scales. Further, the static test cases include one and two-dimensional Laplace and Poisson equations with different forcing functions, and those involving local high gradients. The transient test cases include the classical wave propagation problem in both one and two dimensional space.

In all the numerical examples presented, the atomic and continuum domains are respectively modeled using the harmonic potential and the Hermite-cloud method, detailed in Chapter 2, unless specified otherwise. The atomic mass ( $m$ ) in reduced units and the spring constant ( $k$ ) defined in the harmonic potential are set to unity unless specified otherwise. In the Hermite-cloud simulation, the cloud size for the cubic spline window function defined in Eq. (2.8) is taken as 1.17 times the nodal spacing, in both the  $x$  and  $y$  directions respectively, to ensure that sufficient nodes are included in the interpolation domain. The basis function vector defined in Eq. (2.13) is taken to be quadratic, and the cloud area defined in Eq. (2.13) is set to

unity. The distributions of nodes in continuum domains are uniform. The error tolerance defined in Eq. (3.10) is set at  $1 \times 10^{-5}$  for all problems. The numerical accuracy of the multiscale model is measured using a global error measure defined as (Aluru and Li, 2001)

$$\psi = \frac{1}{|f_{\max}|} \sqrt{\frac{1}{N_T} \sum_{i=1}^{N_T} (f_e - f_c)^2} \quad (4.1)$$

where  $\psi$  is the global error in the computed solution,  $f_e$  and  $f_c$  the respective exact and numerically computed solutions,  $f_{\max}$  is the maximum value of the exact solution in the domain, and  $N_T$  the total number of nodes and atoms scattered in the computational domain.

To study the results of the multiscale simulation in comparison to a full atomistic simulation, a relative error measure is used and is defined as (Gu and Zhang, 2006)

$$\xi = \frac{\sum_i^m f^{AD} - \sum_i^n f^{MM}}{\sum_i^m f^{AD}} \quad (4.2)$$

where  $\xi$  is the relative error,  $f^{AD}$  the solution obtained by a pure atomistic simulation consisting of  $m$  atoms, and  $f^{MM}$  the solution obtained from the multiscale simulation, consisting of  $n$  atoms and nodes taken together. The source codes for the current multiscale method were developed using Matlab version 7.4, and run on a Pentium Duo Core Machine (3GHz) with 3.25GB RAM.

## 4.1 Static Problems with Coupled Length Scales

### 4.1.1 1-D Problems

In this section the multiscale model is used to solve one dimensional boundary value problems. The computational model for 1-D multiscale problems is shown in Fig.

4.1. The atomic and continuum domains constitute approximately one half of the computational domain, and overlap in the transition region.

The first example considered is the simple Poisson equation, for which the governing equation and boundary conditions are

$$\frac{\partial^2 u}{\partial x^2} = \frac{105}{2}x^2 - \frac{15}{2} \quad -1 \leq x \leq 1 \quad (4.3)$$

$$\frac{\partial u}{\partial x}(x = -1) = -10 \quad (4.4)$$

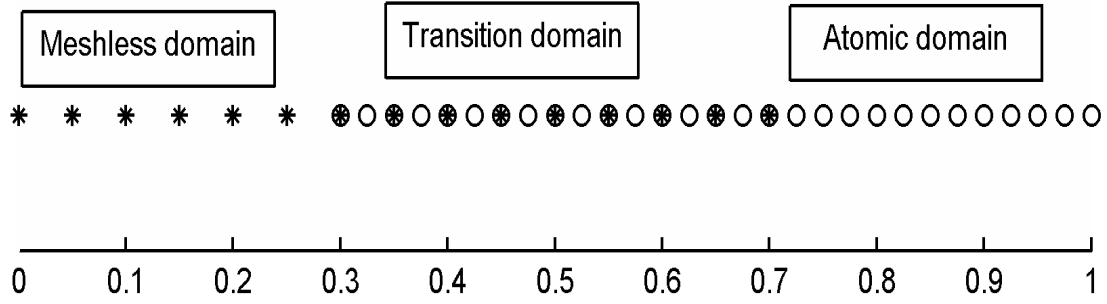
$$u(x = 1) = 1 \quad (4.5)$$

The problem domain  $[-1, 1]$  is divided into continuum, atomic and transition domains in the region  $[-1, 0.2]$ ,  $[-0.2, 1]$  and  $[-0.2, 0.2]$ , respectively. The continuum and atomic domains contain 25 nodes and 49 atoms, with an average spacing of 0.05 and 0.025 units respectively. The problem is solved as per the procedure detailed in Table 3.1.

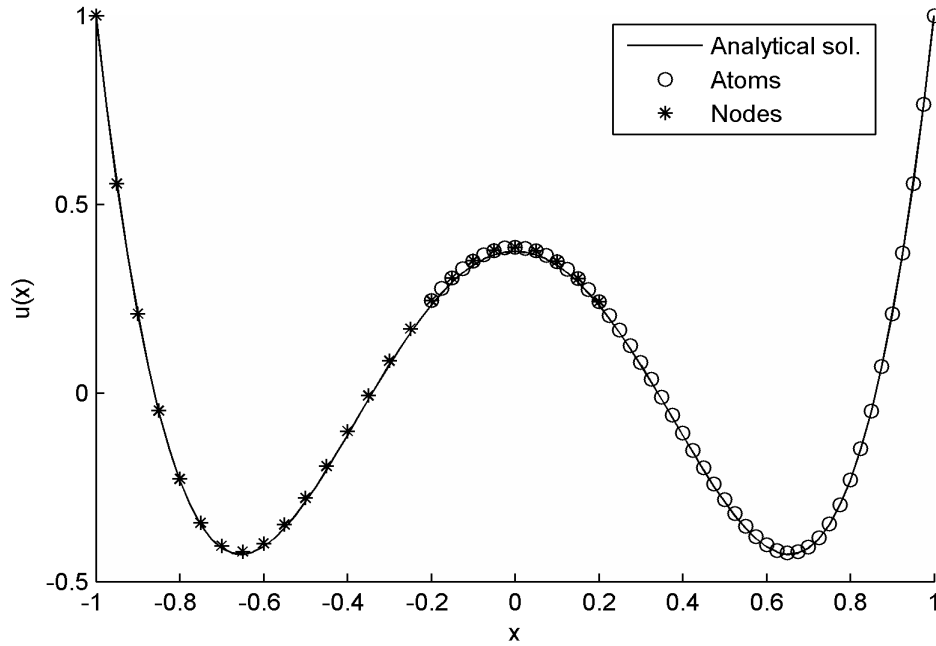
The analytical solution for this problem is given as

$$u = \frac{35}{8}x^4 - \frac{15}{4}x^2 + \frac{3}{8} \quad (4.6)$$

Figure 4.2 shows the comparison between the analytical and numerical results for this problem. The numerical results are compared with both the analytical solution and a pure atomistic simulation, using the global and relative error measures defined in Eqs. (4.1)-(4.2), and the values are tabulated in Table 4.1. From the tabulated values it can be noted that there is a good agreement between the analytical and numerical results, as seen from a small global error value of 0.0083. A relative error value of 0.0063 shows that the results of the multiscale simulation are not only accurate but also close to a pure atomistic simulation. The atomistic simulation used for comparison is carried out with the same atomic spacing as in the multiscale model.



**Figure 4.1:** Computational model for static 1-D problems.



**Figure 4.2:** Comparison of analytical and multiscale simulation results for the 1-D Poisson equation.



The next example considered is the one-dimensional heat conduction problem with an internal heat source. The problem has a steep gradient and is governed by the following differential equation and boundary conditions

$$\frac{\partial^2 T}{\partial x^2} = -2s^2 \sec h^2[s(x - 0.5)] \tanh[s(x - 0.5)] \quad (4.7)$$

$$T(x = 0) = -\tanh(3s) \quad (4.8)$$

$$T(x = 1) = \tanh(3s) \quad (4.9)$$

The corresponding exact analytical solution is

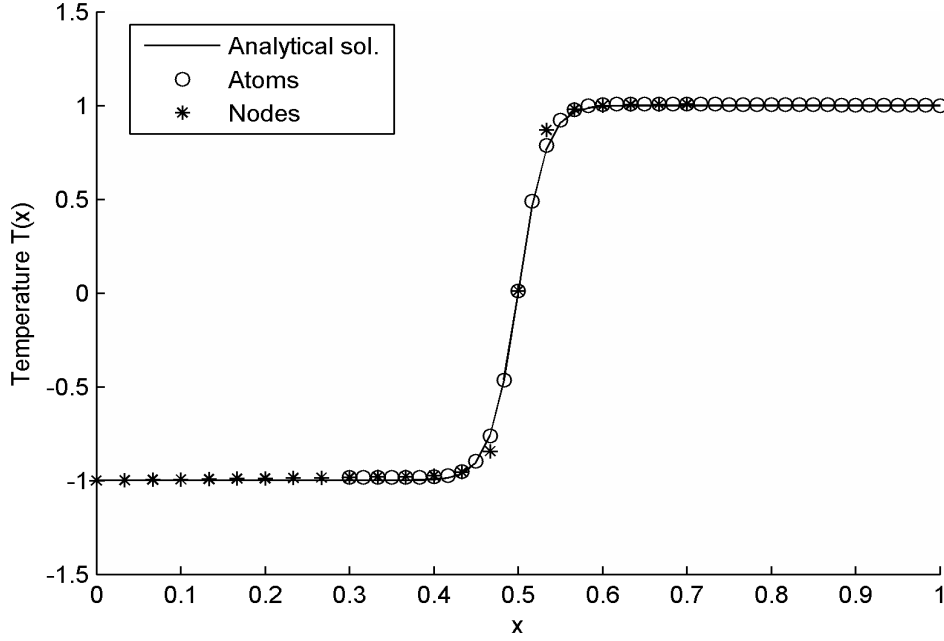
$$T(x) = \tanh[s(x - 0.5)] \quad (4.10)$$

The parameter  $s$  is taken as 30 in this problem.

The one-dimensional computational model for this problem is similar to the previous problem with the continuum, atomic and transition domains defined in the regions  $[0, 0.7]$ ,  $[0.3, 0.7]$ , and  $[0.3, 0.7]$  respectively. There are a total of 22 nodes in the continuum domain with an average nodal spacing of  $1/30$  units, and 43 atoms in the atomic domain with an average atomic spacing of  $1/60$  units. The results of the multiscale simulation are plotted against the analytical solution in Fig. 4.3, and the error values are given in Table 4.1. It is observed from the figure that the method satisfactorily captures the higher gradient with relatively few nodes/atoms, and is able to produce results that are almost identical to a pure atomistic simulation.

The final 1-D problem considered is the Poisson equation with a local high gradient, for which the governing equation and boundary conditions are

$$\frac{\partial^2 u}{\partial x^2} = -6x - \left[ \frac{2}{\alpha^2} - 4 \left( \frac{x - \beta}{\alpha^2} \right)^2 \right] \exp \left[ - \left( \frac{x - \beta}{\alpha} \right)^2 \right], \quad 0 \leq x \leq 1 \quad (4.11)$$



**Figure 4.3:** Comparison of analytical and multiscale simulation results for the 1-D heat conduction equation with a heat source.

$$u(x=0) = \exp\left(\frac{-\beta^2}{\alpha^2}\right) \quad (4.12)$$

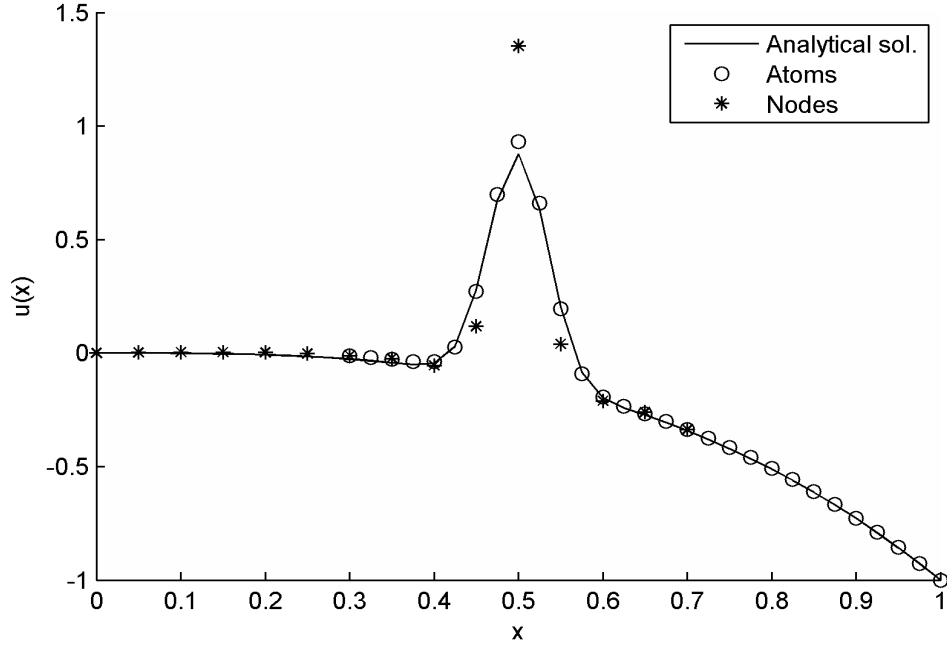
$$u(x=1) = -1 + \exp\left(-\left(\frac{1-\beta}{\alpha}\right)^2\right) \quad (4.13)$$

and the analytical solution is

$$u = -x^3 + \exp\left(-\left(\frac{x-\beta}{\alpha}\right)^2\right) \quad (4.14)$$

The computational model for this problem is same as that of the previous problem. It has 15 nodes and 29 atoms with an average nodal and atomic spacing of 0.05 and 0.025 units, respectively. In the present simulation,  $\alpha$  and  $\beta$  values are taken as 0.05 and 0.5 respectively. Figure 4.4 shows the comparison between the analytical and numerical results, and the error values presented in Table 4.1. The results once again validate the multiscale model and its capability to simulate problems with local high gradient accurately.

The results of the 1-D problems provide us with some valuable insight on the



**Figure 4.4:** Comparison of analytical and multiscale simulation results for the Poisson equation with a local high gradient.

characteristics of the multiscale model. Figure 4.5 shows the variation of the global error from the start of the simulation until its convergence, as per the criteria defined in Eq. (3.10). The results show that the convergence rate is rapid in the initial phase, after which it ceases quite abruptly. The width of the transition domain affects the global error and the convergence rate of the solution, as seen in Figs. 4.6 and 4.7. Increasing the width of the transition region, in multiples of the nodal spacing, ensures a better exchange of information between the domains, as evident from the reduced global error values and the number of iterations required for solution convergence. An optimum width for the transition domain has to be chosen based on the accuracy requirements and also taking into account the additional computations that would be required in the transition domain.

In general, for engineering problems involving atomic-continuum coupling, the strategic region is discretized using an atomistic approach. This region generally constitutes a localized region where accuracy of the solution is of importance. Therefore, it would be beneficial to study the performance of the multiscale model

with an increased atomic density. Figure 4.8 shows the variation of global error for an increasing number of atoms and a fixed number of nodes. As expected, the global error decreases with an increase in the number of atoms. However, the decrease in error is rapid only in the initial phase after which the accuracy does not increase significantly. This may be attributed to an increase in the interpolation errors in the transition region.

**Table 4.1:** Comparison of global and relative errors for static 1-D problems.

Problem type	Atoms/node distribu- tion	Global error ( $\psi_1$ ) (multi- scale simula- tion)	Global error ( $\psi_2$ ) (atomic simulation)	Relative error $\xi = \psi_1 - \psi_2$	No. of iterations for con- vergence
Poisson equation with forcing function	$49 \times 25$	0.0083	0.0020	0.0063	17
1-D Heat conduction equation	$43 \times 22$	0.0200	0.0040	0.0160	08
Poisson equation with local high gradient	$29 \times 15$	0.0807	0.0095	0.0712	08

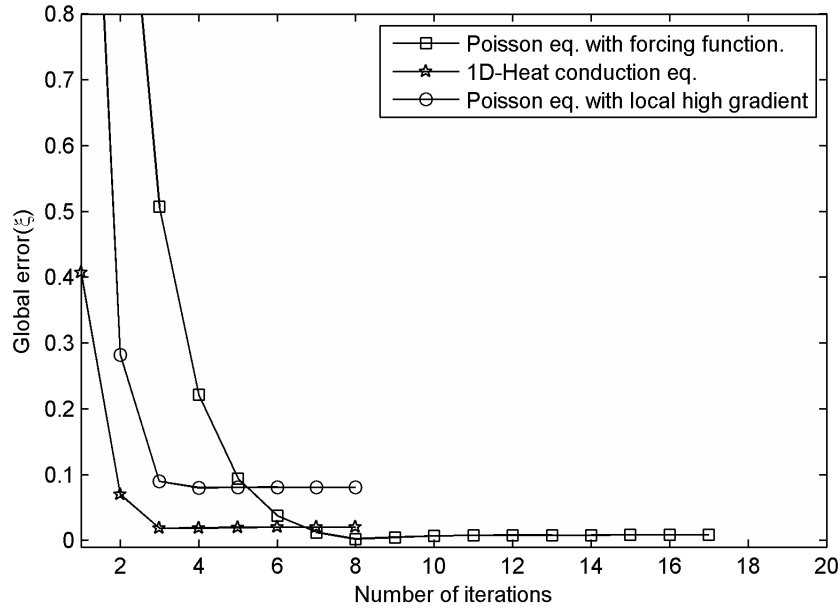


Figure 4.5: Variation of global error with the number of iterations.

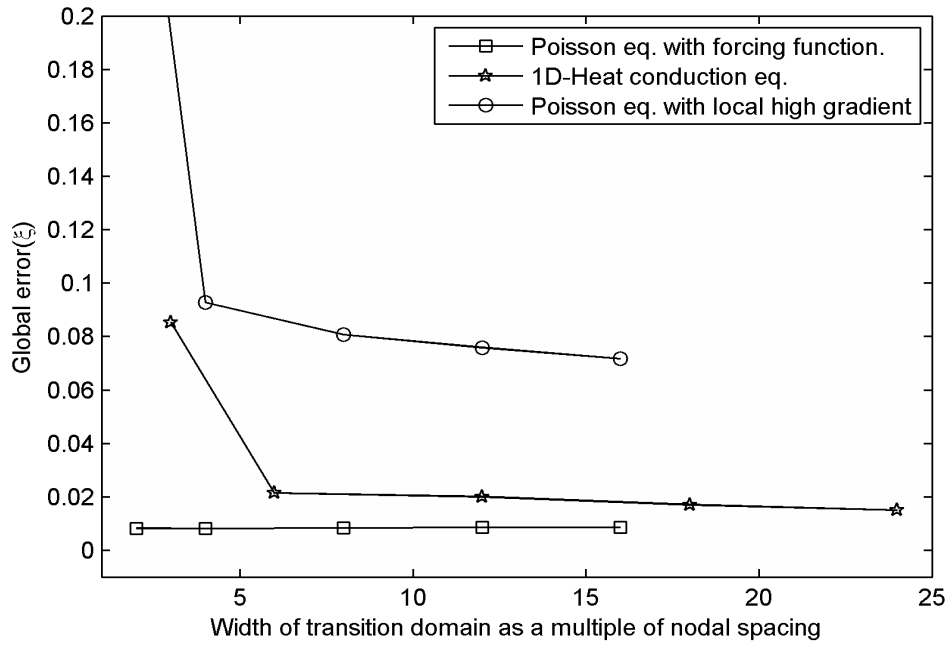


Figure 4.6: Effect of the transition region width on the global error.

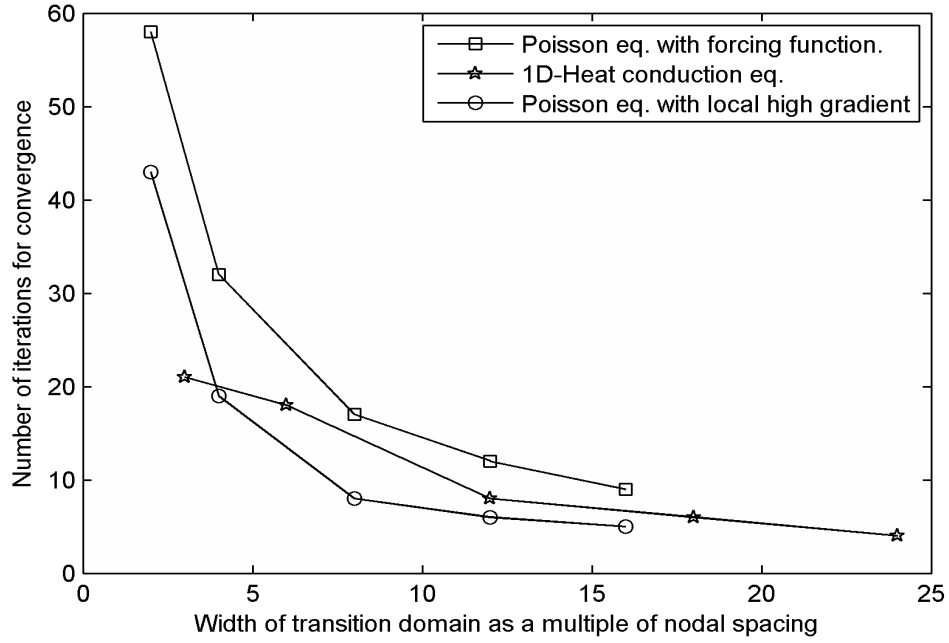


Figure 4.7: Effect of the transition region width on the solution convergence.

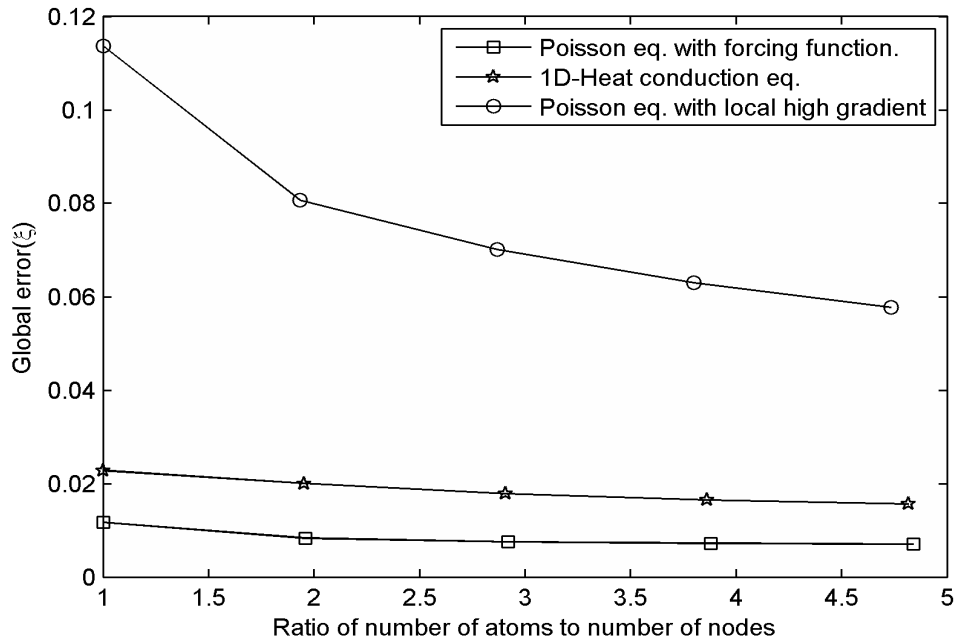


Figure 4.8: Effect of density ratio on the global error.

### 4.1.2 2-D Problems

In the two-dimensional domain, we begin with the Poisson equation over an unit square, governed by the following equation

$$\begin{aligned} \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = & -4\pi^2 \sin(2\pi x)y^2(1-y)^2 + \\ & \sin(2\pi x)(2-12y+12y^2) \quad 0 \leq x \leq 1, 0 \leq y \leq 1 \end{aligned} \quad (4.15)$$

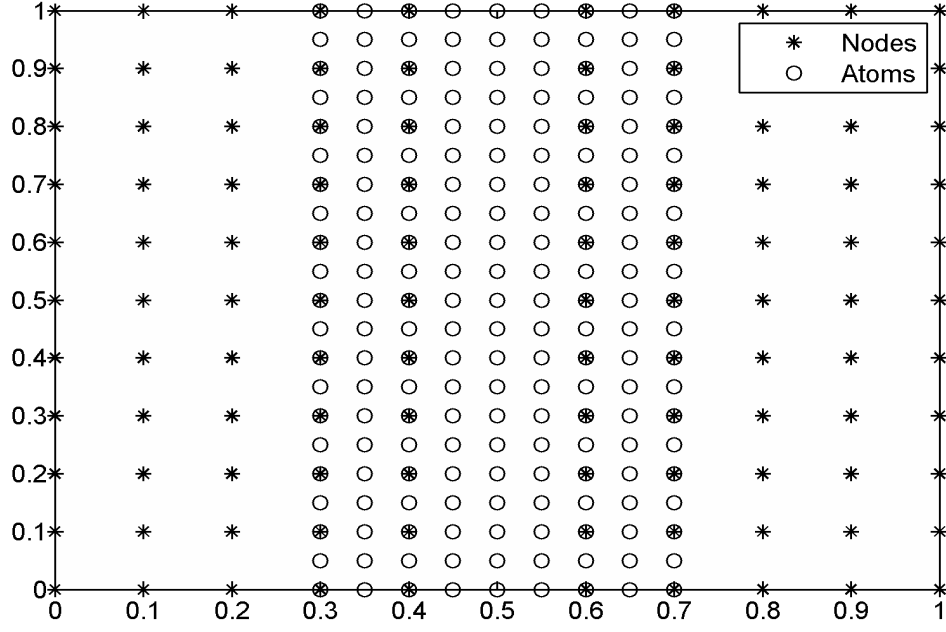
subjected to uniform Dirichlet boundary conditions along its edges

$$u(x=0) = u(x=1) = u(y=0) = u(y=1) = 0 \quad (4.16)$$

The multiscale computational model for this problem, shown in Fig. 4.9, is a unit square containing 110 nodes and 189 atoms. The atomic and nodal spacing is uniform in both  $x$  and  $y$  directions and is equal to 0.05 and 0.1 units, respectively. The atomic domain is at the center with continuum domain on both sides and the transition region providing the necessary coupling at the overlapping regions. The exact solution for this problem is

$$u(x, y) = \sin(2\pi x)y^2(1-y)^2 \quad (4.17)$$

Figure 4.10 compares the multiscale simulation results with the exact analytical solution, and the results of a pure atomistic simulation are presented in Fig. 4.11. As with 1-D problems, the atomistic simulation is carried out with the same atomic spacing, in both the  $x$  and  $y$  directions, as in the multiscale model. The results of the multiscale simulation are in good agreement with both the analytical results and the atomistic simulation, which is evident from the global and relative error values of 0.0690 and 0.0652 respectively, as shown in Table 4.2. This further demonstrates the capability of the multiscale model in ensuring a smooth coupling in two-dimensional



**Figure 4.9:** Computational model of the first type for 2-D problems.

domains.

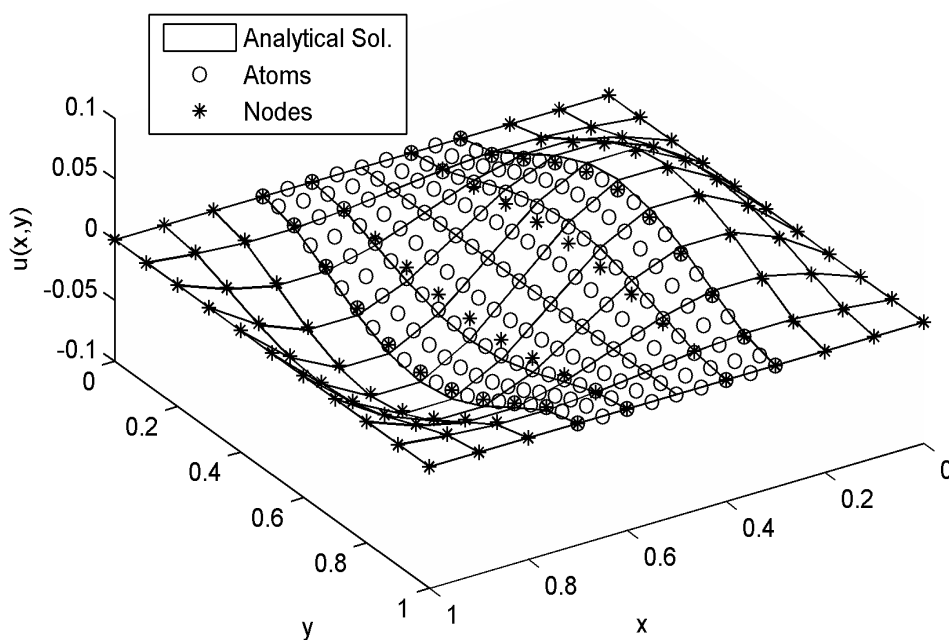
In the computational model shown in Fig. 4.9, the transition region is primarily oriented along the axis which leads to minimal information exchange along the  $y$  axis. This type of coupling is simpler to implement and is well suited for simple problems or those that have a steeper gradient along a particular axis, in which case, the primary direction of coupling may be chosen accordingly. A better version of this computational model is shown in Fig. 4.12, where the atomic domain is completely within the continuum domain and the transition region is of uniform width in both the  $x$  and  $y$  directions. This computational model is used in subsequent 2-D problems.

The next problem considered is the Laplace equation in a square domain with mixed Dirichlet and Neumann Boundary conditions

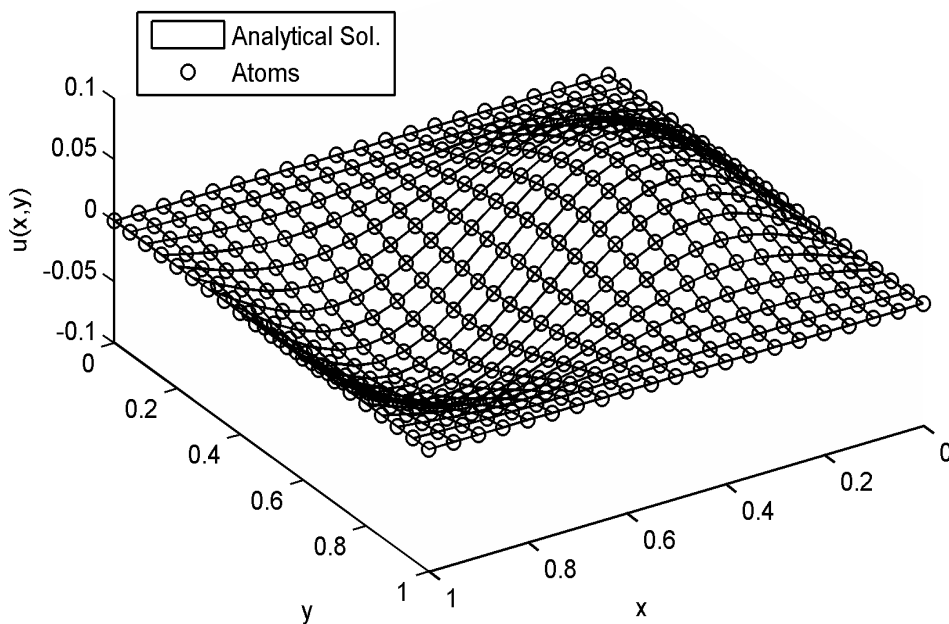
$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \quad 0 \leq x \leq 1, 0 \leq y \leq 1 \quad (4.18)$$

$$u(x=0) = -y^3 \quad (4.19)$$

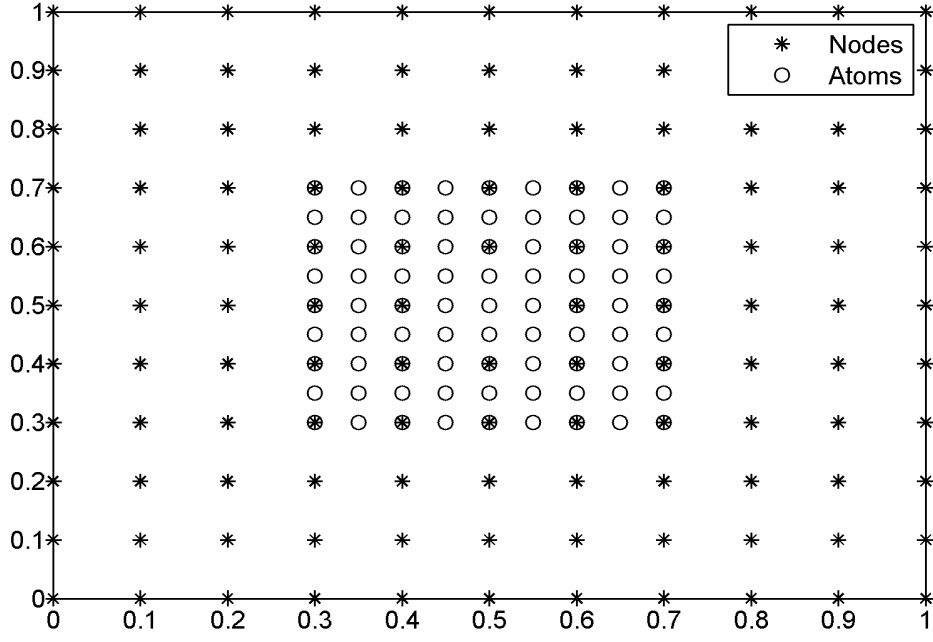




**Figure 4.10:** Comparison of analytical and multiscale simulation results for the 2-D Poisson equation.



**Figure 4.11:** Comparison of analytical and atomistic simulation results for the 2-D Poisson equation.



**Figure 4.12:** Computational model of the second type for 2-D problems.

$$u(x = 1) = -1 - y^3 + 3y^2 + 3y \quad (4.20)$$

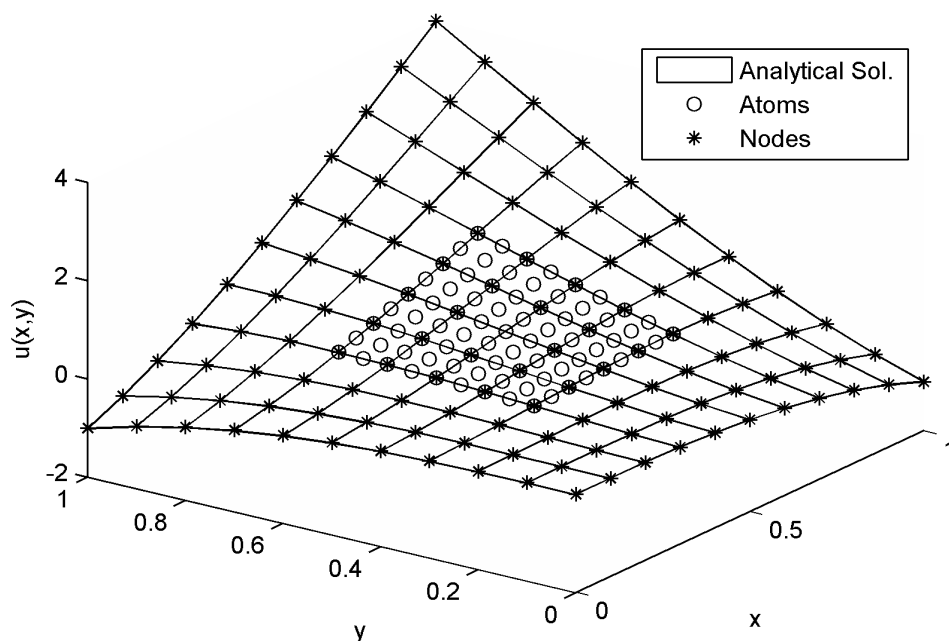
$$\frac{\partial u}{\partial y}(y = 0) = 3x^2 \quad (4.21)$$

$$\frac{\partial u}{\partial y}(y = 1) = -3 + 6x + 3x^2 \quad (4.22)$$

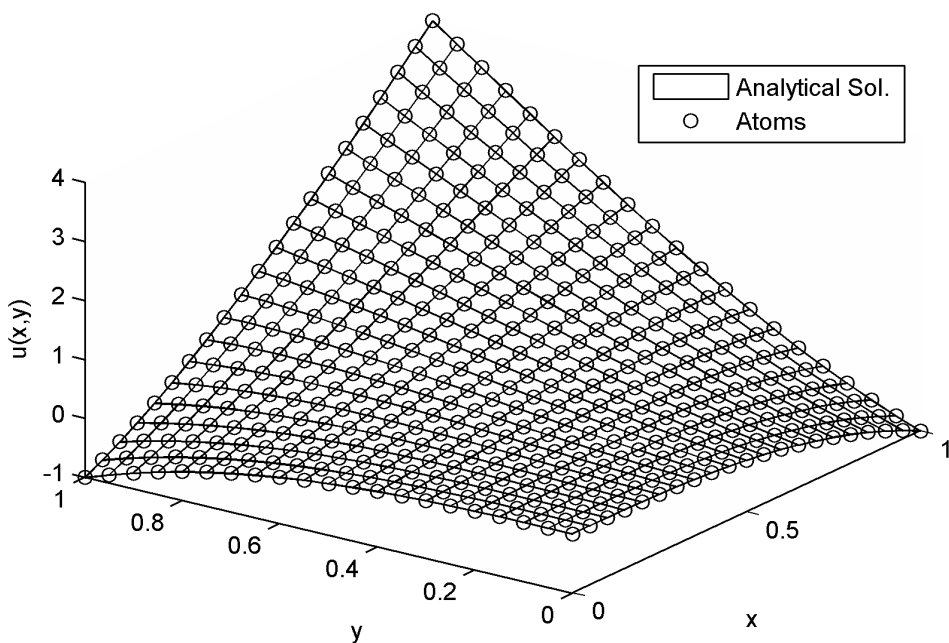
The exact solution of this problem is

$$u(x, y) = -x^3 - y^3 + 3x^2y + 3xy^2 \quad (4.23)$$

The computational model for this problem contains 81 atoms and 120 nodes with uniform spacing of 0.05 and 0.1 units in the atomic and continuum domains, respectively. The spacing is constant in both the  $x$  and  $y$  directions. The width of the transition region is equal to the nodal spacing or twice the atomic spacing in all directions. The results of the multiscale simulation and a pure atomistic simulation are shown in Figs. 4.13 and 4.14. A relative error of 0.0008 supports the conclusion that the present multiscale ensures an accurate and efficient coupling,



**Figure 4.13:** Comparison of analytical and multiscale simulation results for the 2-D Laplace equation.



**Figure 4.14:** Comparison of analytical and atomistic simulation results for the 2-D Laplace equation.

which is evident when comparing the total number of nodes and atoms (201) in the multiscale model against the total number of atoms (441) in the atomistic model.

The final problem considered is the Poisson equation with a local high gradient over a square domain for which the governing equation and boundary conditions are

$$\begin{aligned} \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = & -6x - 6y - \left[ \frac{4}{\alpha^2} - 4 \left( \frac{x - \beta}{\alpha^2} \right)^2 - 4 \left( \frac{y - \beta}{\alpha^2} \right)^2 \right] \\ & \exp \left[ - \left( \frac{x - \beta}{\alpha} \right)^2 - \left( \frac{y - \beta}{\alpha} \right)^2 \right], \quad 0 \leq x \leq 1, 0 \leq y \leq 1 \end{aligned} \quad (4.24)$$

$$u(x = 0) = -y^3 + \exp \left[ - \left( \frac{\beta}{\alpha} \right)^2 - \left( \frac{y - \beta}{\alpha} \right)^2 \right] \quad (4.25)$$

$$u(x = 1) = -1 - y^3 + \exp \left[ - \left( \frac{1 - \beta}{\alpha} \right)^2 - \left( \frac{y - \beta}{\alpha} \right)^2 \right] \quad (4.26)$$

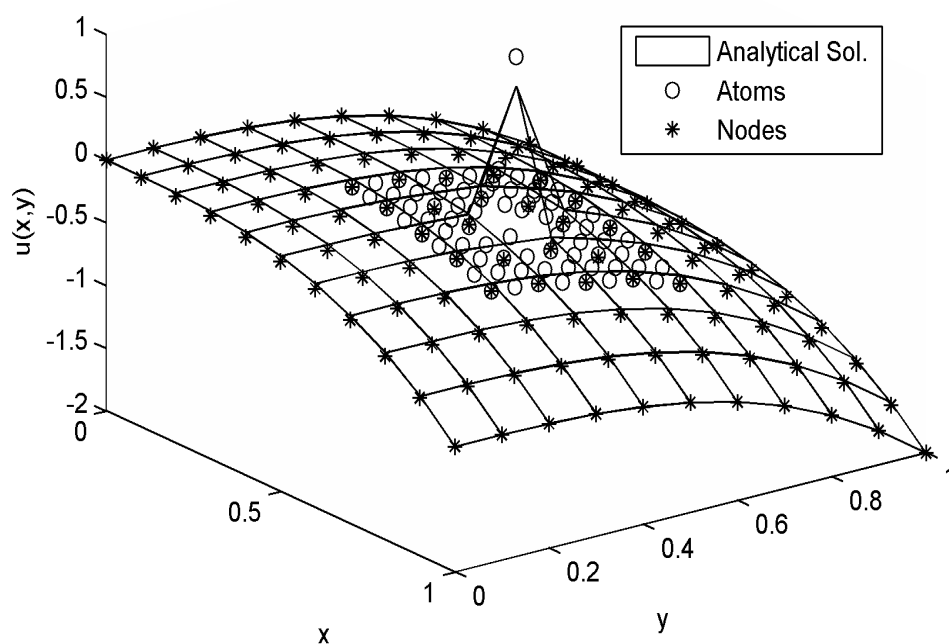
$$\frac{\partial u}{\partial y}(y = 0) = \frac{2\beta}{\alpha^2} \exp \left[ - \left( \frac{\beta}{\alpha} \right)^2 - \left( \frac{x - \beta}{\alpha} \right)^2 \right] \quad (4.27)$$

$$\frac{\partial u}{\partial y}(y = 1) = -3 - 2 \frac{(1 - \beta)}{\alpha^2} \exp \left[ - \left( \frac{x - \beta}{\alpha} \right)^2 - \left( \frac{1 - \beta}{\alpha} \right)^2 \right] \quad (4.28)$$

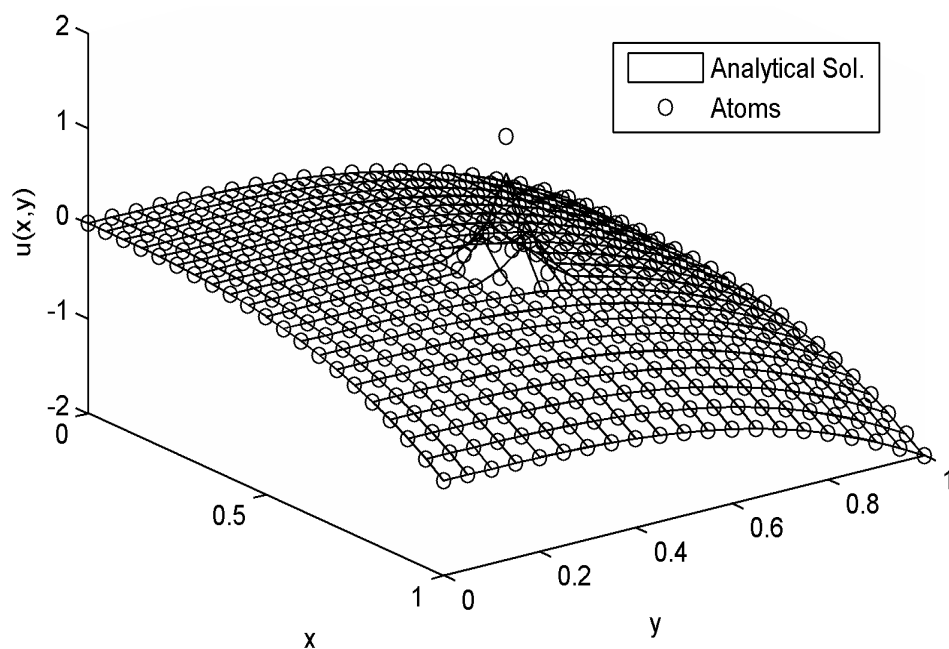
The analytical solution is

$$u(x, y) = -x^3 - y^3 + \exp \left[ - \left( \frac{x - \beta}{\alpha} \right)^2 - \left( \frac{y - \beta}{\alpha} \right)^2 \right] \quad (4.29)$$

The  $\alpha$  and  $\beta$  values for this problem are taken as 0.05 and 0.5 respectively. The problem has a local high gradient near the point (0.5, 0.5). The computational model and the atomic/nodal distributions for this problem are the same as that of the previous problem. The results of the multiscale and pure atomistic simulations are shown in Figs. 4.15 and 4.16, which clearly indicate that the multiscale model is able to capture the local high gradient satisfactorily and as effectively as a pure atomistic simulation, with small global and relative error values of 0.0393 and 0.0123 respectively. The multiscale simulation is particularly suitable for these types of



**Figure 4.15:** Comparison of analytical and multiscale simulation results for the 2-D Poisson equation with local high gradient.



**Figure 4.16:** Comparison of analytical and atomistic simulation results for the 2-D Poisson equation with local high gradient.

**Table 4.2:** Comparison of global and relative errors for static 2-D problems.

Problem type	Atoms/node distribu- tion	Global error ( $\psi_1$ ) (multi- scale simula- tion)	Global error ( $\psi_2$ ) (atomic simulation)	Relative error $\xi = \psi_1 - \psi_2$	No. of iterations for con- vergence
Poisson equation	$189 \times 110$	0.0690	0.0038	0.0652	06
Laplace equation	$81 \times 120$	0.0008	$1 \times 10^{-14}$	0.0008	18
Poisson equation with local high gradient	$81 \times 120$	0.0393	0.0270	0.0123	34

problems involving local high gradients, wherein the atomic domain can be strictly restricted to the zone with the high gradient and a continuum description can be used elsewhere, thereby substantially reducing the computational effort.

## 4.2 Transient Problems with Coupled Length and Time Scales

### 4.2.1 Wave propagation problems

In this section the accuracy of the multiscale model is validated by solving transient problems which involve the coupling of both length and time scales. The transient problems analyzed include the classical wave propagation problem in both one and two dimensional space, and the vibration of a stretched elastic string and membrane.

### Case-1 Smooth harmonic wave

The first problem considered is the homogeneous wave propagation problem in an infinite domain without boundaries. The governing equation and initial conditions for this problem are given by

$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2} \quad -\infty < x < \infty, \quad t > 0 \quad (4.30)$$

$$u(x, 0) = f(x) \quad -\infty < x < \infty, \quad (4.31)$$

$$\frac{\partial u}{\partial t}(x, 0) = g(x) \quad -\infty < x < \infty \quad (4.32)$$

where  $u(x, t)$  is the displacement of the wave at any time  $t$ .

The problem being second-order in time requires two initial conditions. In this problem, the velocity field  $(\partial u / \partial t)_{(x,0)}$ , at initial time is taken as zero. The problem is solved for two different initial conditions; a continuous Gaussian pulse and a piecewise linear ‘hat function’. The initial conditions for the two different test cases are given as

$$u(x, 0) = u_0 \exp\left(\frac{-(x-b)^2}{2c^2}\right) \quad - \text{Gaussian pulse} \quad (4.33)$$

$$u(x, 0) = u_0 \begin{cases} 1 - |x| & |x| \leq 1 \\ 0 & \text{otherwise} \end{cases} \quad - \text{Hat function} \quad (4.34)$$

where  $u_0$  is the amplitude of the wave, and parameters  $b$  and  $c$  control the center position of the wave peak and its width respectively. In this work, the constants are chosen such that the initial displacement is of the form  $u(x, 0) = u_0 e^{-10x^2}$ . The solution of the wave equation at any time  $t$  for the initial conditions defined above can be obtained using the d’Alembert’s solution of the wave equation as (Gockenbach, 2002)

$$u(x, t) = \frac{1}{2} (f(x-ct) + f(x+ct)) + \frac{1}{2c} \int_{x-ct}^{x+ct} g(s) ds \quad (4.35)$$

The computational model for this problem is shown in Fig. 4.17. A finite domain  $[-5, 5]$  is considered for numerically solving the problem. However, the simulation is stopped sufficiently before the wave reaches the boundary, to avoid any boundary effects. The atomic domain is located in the central region  $[-2, 2]$  and contains 201 atoms with an average spacing of 0.02 units. The continuum domain on either side, in the regions  $[-5, -1]$  and  $[1, 5]$  contains 81 nodes with an average nodal spacing of 0.1 units. The atomic and continuum regions overlap each other in the transition regions  $[-2, -1]$  and  $[1, 2]$ .

The atomic domain in this problem is modeled using a spring-mass system defined by a harmonic potential as

$$\varphi(u) = \frac{1}{2}k \sum_n (u_{n+1} - u_n)^2 \quad (4.36)$$

where  $k$  is the spring constant. The governing equation for the atomic domain, given in Eq. (2.57), can be discretized by only considering nearest neighbor interactions as

$$mu'' = u_{n-1} - 2u_n + u_{n+1} \quad (4.37)$$

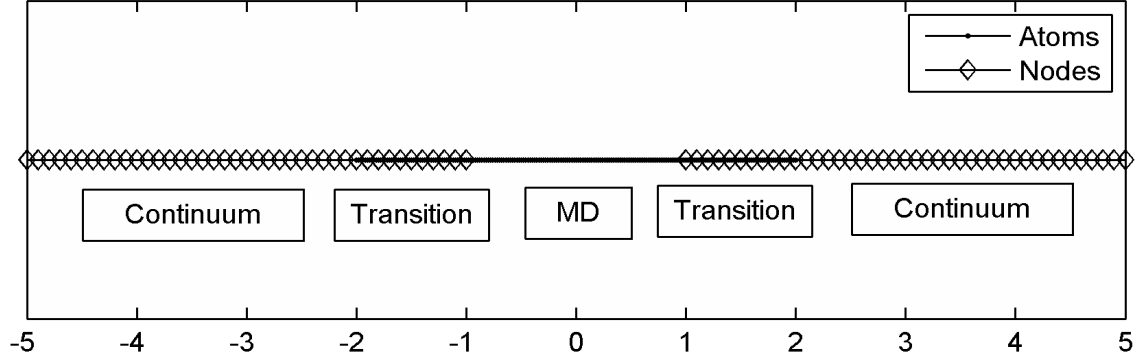
The solution to this equation at any given time step can then be obtained using the leap frog scheme detailed in Section 2.2.5.

In the continuum domain, the Hermite-cloud method and the leap frog scheme are used for space and time discretization respectively. Using the Hermite-cloud method, the displacement  $u(x)$  and the governing PDE in Eq. (4.30) can be spatially discretized as

$$u^h(x, t) = \sum_{n=1}^{NP} N_n(x)u_n(t) + \sum_{m=1}^{NT} \left( x - \sum_{n=1}^{NP} N_n(x)x_n \right) M_m(x)u_x(t) \quad (4.38)$$

$$\frac{\partial^2 u(x, t)}{\partial t^2} = \frac{\partial^2 u(x, t)}{\partial x^2} = \sum_{n=1}^{NP} N_{n,xx}(x)u(t) \quad (4.39)$$





**Figure 4.17:** Computational model for 1-D wave propagation problems.

Using the spatially discretized Eqs. (4.38)-(4.39), and the leap frog scheme defined in Eqs. (2.74)-(2.76), the PDE can be discretized in both space and time domains, and the displacement at any time  $t + 1$  can be obtained as

$$\frac{\partial u^{t+1/2}}{\partial t} = \frac{\partial u^t}{\partial t} + \frac{\Delta t^c}{2} \left( \sum_{n=1}^{NP} N_{n,xx}(x) u(t) \right) \quad (4.40)$$

$$u^{t+1} = u^t + \Delta t^c \frac{\partial u^{t+1/2}}{\partial t} \quad (4.41)$$

$$\frac{\partial u^{t+1}}{\partial t} = \frac{\partial u^{t+1/2}}{\partial t} + \frac{\Delta t^c}{2} \left( \sum_{n=1}^{NP} N_{n,xx}(x) u^{t+1} \right) \quad (4.42)$$

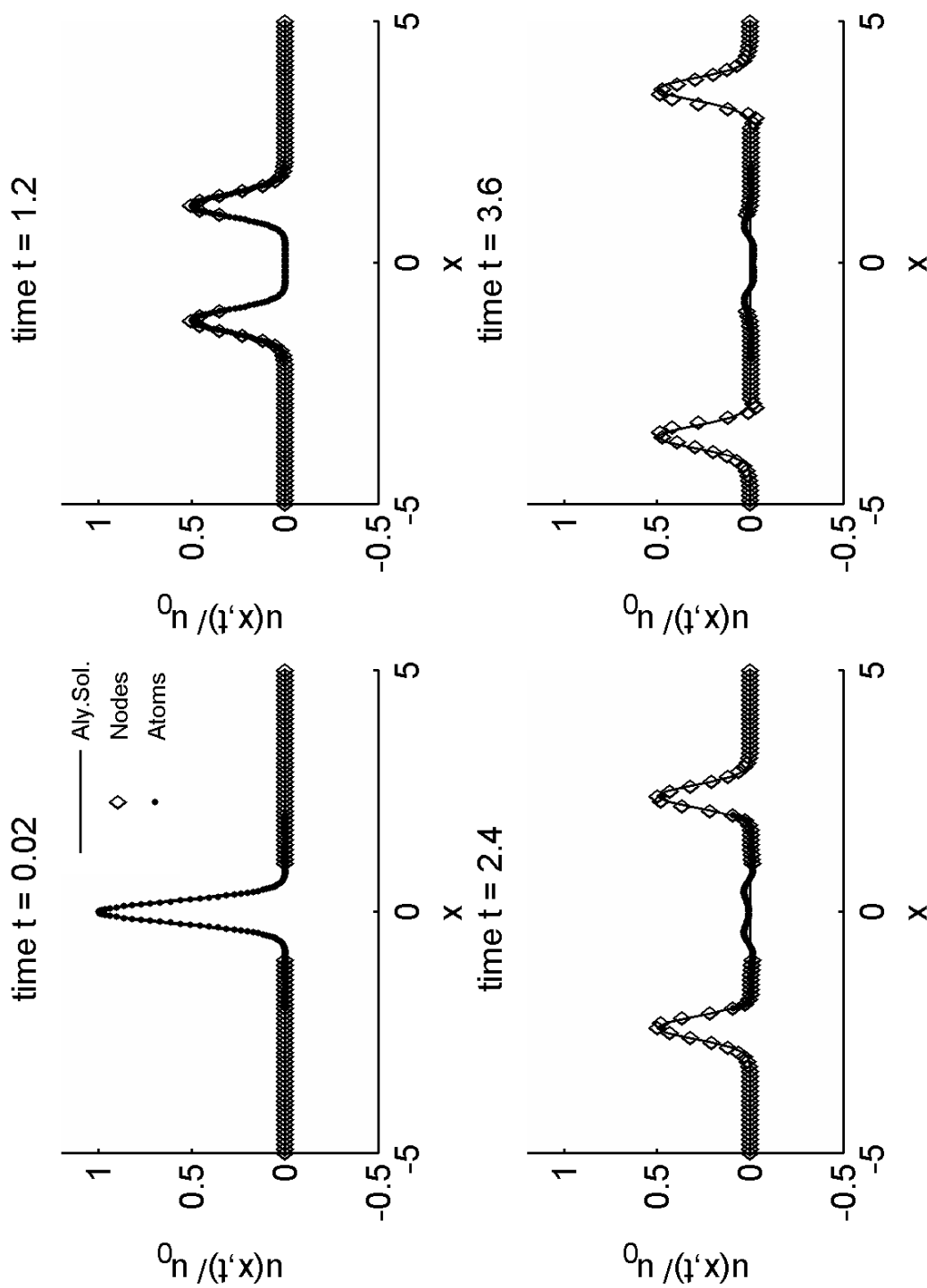
With the atomic and continuum domains discretized, the multiscale problem is now solved as per the procedure detailed in Table 3.1. A time step of  $\Delta t^c = 0.01$  units is used in the continuum domain and the multiplication factor  $n$  defined in Eq. (3.16) is taken as 5. The global and relative error measures defined in Eqs. (4.1)-(4.2) are used for the comparison of the results.

The results of the multiscale simulation at different time instances, for the two different initial conditions considered are presented in Figs. 4.18 and 4.19, and the maximum global and relative error values observed are tabulated in Table 4.3. The results of the pure atomistic simulation for the Gaussian input wave, used as a benchmark for comparing the multiscale simulation results, is shown in Fig. 4.20.

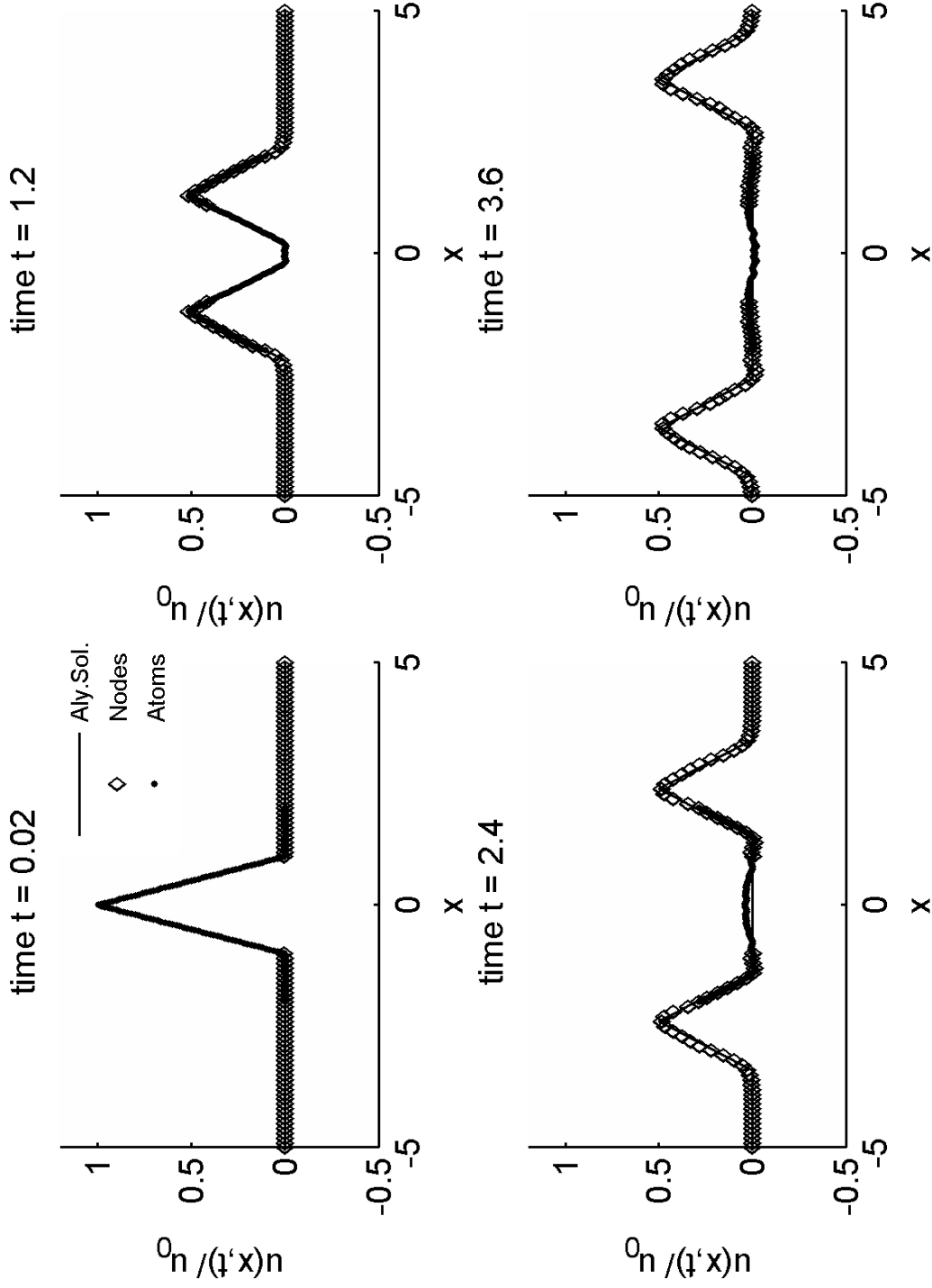
The results show good agreement between the analytical and numerical solutions, supporting the accuracy of the multiscale model and the relative errors less than 5%, as observed from the table, indicates that the multiscale model is able to match the performance of the pure MD simulation. By adopting a multiscale approach, the atomic detail is only retained wherever essential, reducing the number of atoms in the domain and thereby the computational expense. In computing the global error of the computed solution, only the atoms and nodes within the domain of influence of the wave are considered (Gockenbach, 2002).

The two different wave patterns considered in the study present two different types of inputs. The Gaussian wave corresponding to a smooth continuous input and the piecewise linear “hat” function corresponds to a discontinuous wave. The multiscale model performs well for both test cases which can be verified from the error values in the Table 4.3. Also, the wave amplitude is preserved in both the cases, except for some smoothing effect observed in the case of the ‘hat’ function that might be due to the damping effects of the numerical scheme. However, the wave is able to pass through the transition region into the continuum domain. Some wave reflections are also observed as the wave leaves the atomistic region and crosses the interface. These reflections are caused by the atomistic information that cannot be completely represented by the continuum, and similar traits have also been observed by other researchers (Wagner and Liu, 2003; Tang et al., 2006a;b).

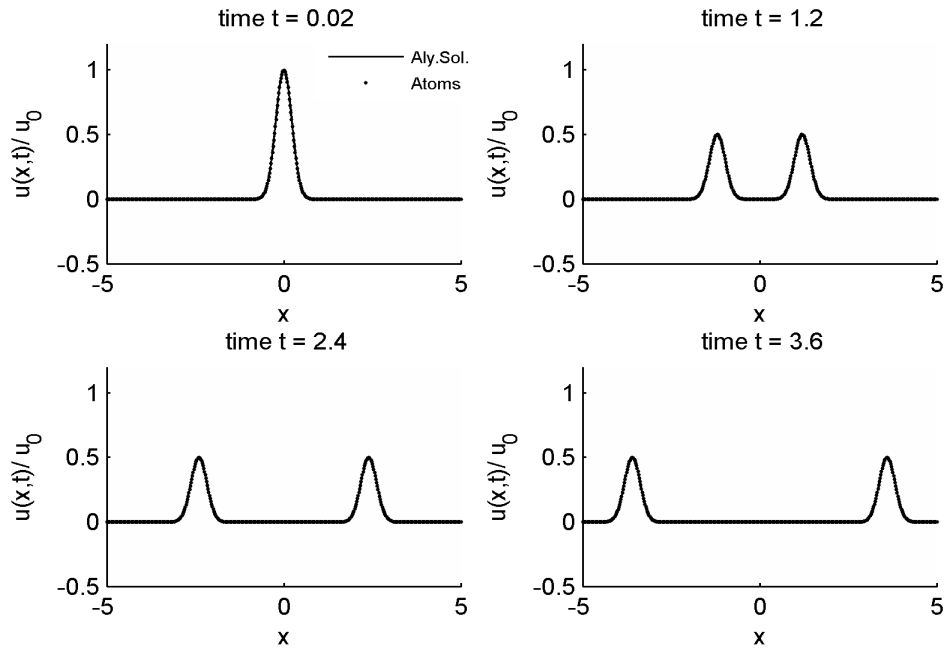
The wave propagation problem with a Gaussian input pulse can also be solved in a 2-D domain, for which the computational model is similar to the one shown in Fig. 4.9. The problem is solved with 852 nodes and 4221 atoms, with all other parameters remaining the same. The results of the simulation are presented in Fig. 4.21. The maximum global error in the observed in the solution is 0.1467. A problem that requires 10521 atoms if discretized fully using the atomistic approach, requires only 5073 points (atoms plus nodes) when solved using the multiscale approach, indicating a reduction of almost 50%.



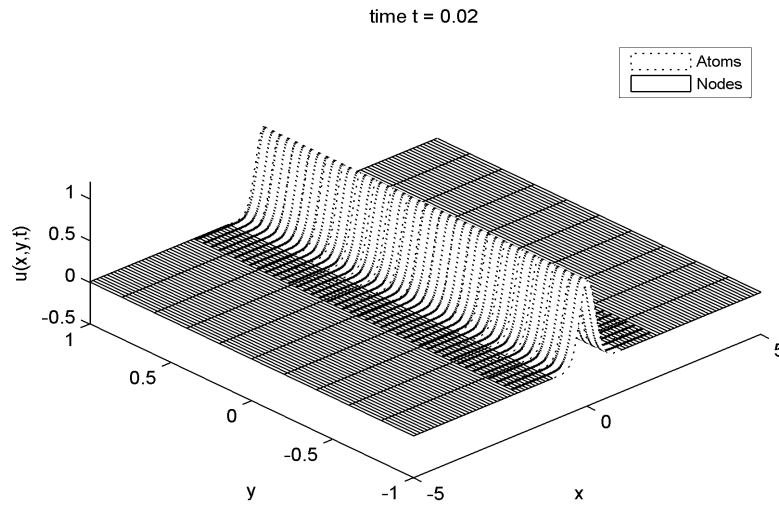
**Figure 4.18:** Snapshots of the analytical and multiscale simulation results at different time instances for 1-D wave propagation problem with a Gaussian input wave (case-1).

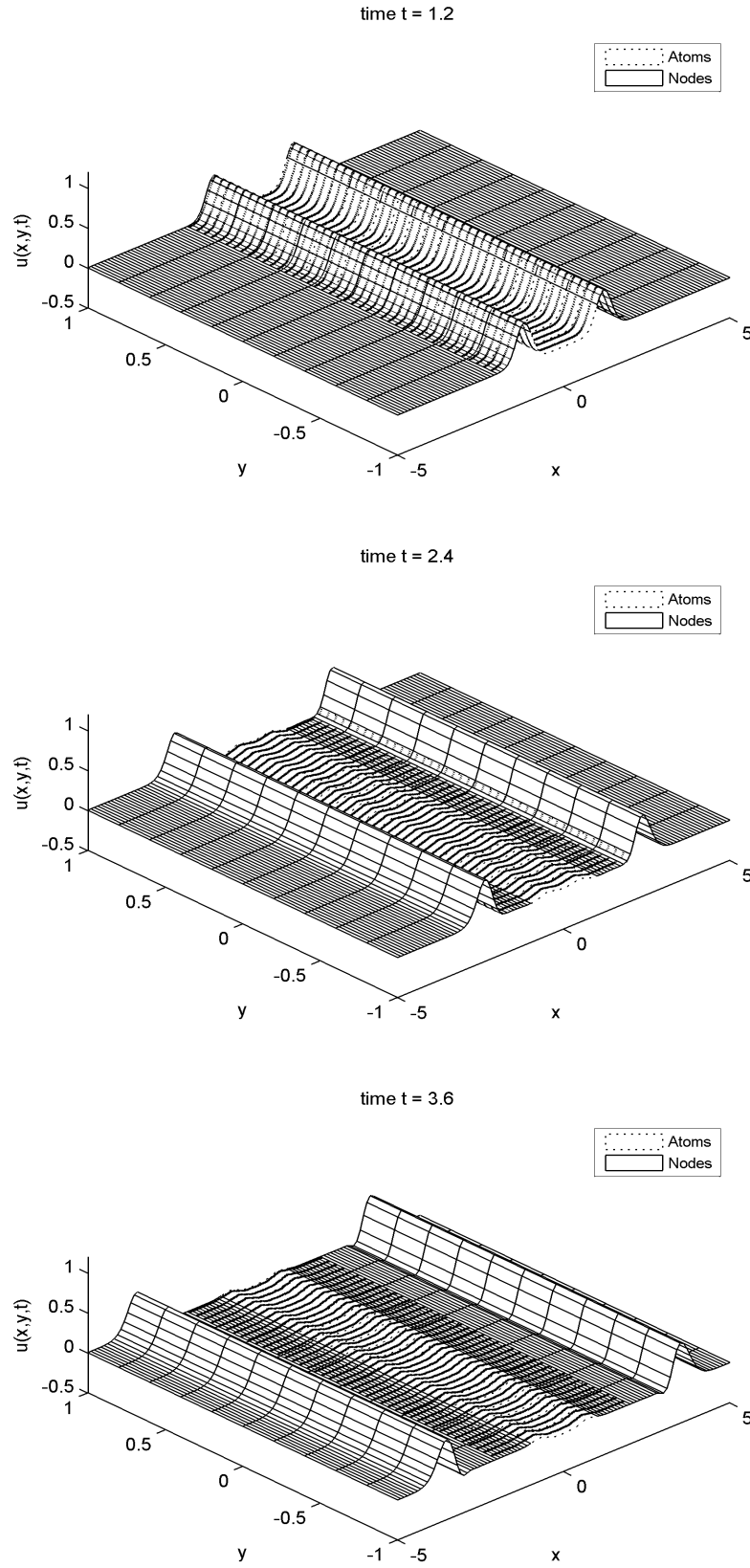


**Figure 4.19:** Snapshots of the analytical and multiscale simulation results at different time instances for 1-D wave propagation problem with a 'Hat function' (case-1).



**Figure 4.20:** Snapshots of pure MD simulation results for 1-D wave propagation with a Gaussian input wave (case-1).





**Figure 4.21:** Snapshots of analytical and multiscale simulation results at different time instances for 2-D wave propagation problem with a Gaussian input wave (case-1).

### Case-2 Harmonic wave with an oscillatory component

Next, we consider an input wave with initial displacement at time  $t = 0$ , as a Gaussian pulse of the form (Wagner and Liu, 2003; Tang, 2008),

$$u(x, 0) = \begin{cases} 0.02 \left[ \frac{e^{-10x^2} - e^{-6.25}}{1 - e^{-6.25}} \right] (1 + 0.1 \cos(80\pi x)) & |x| \leq 1.25, \\ 0 & \text{otherwise} \end{cases} \quad (4.43)$$

The term  $(1 + 0.1 \cos(80\pi x))$  introduces an oscillatory component in the input wave that is contained in the atomic region.

The computational model for this problem is similar to the one shown in Fig. 4.17, except that the atomic domain located in the region  $[-3, 3]$  contains 301 atoms. Time integration in the atomic and continuum domains is carried out using the Leap frog and the implicit Newmark integration schemes, respectively. The Newmark scheme used for time integration in the continuum domain is given as (Reddy, 1993)

$$u^{t+1} = u^t + \Delta t \frac{\partial u^t}{\partial t} + \frac{\Delta t^2}{2} \frac{\partial^2 u^{t+\alpha}}{\partial t^2} \quad (4.44)$$

$$\frac{\partial u^{t+1}}{\partial t} = \frac{\partial u^t}{\partial t} + \Delta t \frac{\partial^2 u^{t+\beta}}{\partial t^2} \quad (4.45)$$

where

$$\frac{\partial^2 u^{t+\gamma}}{\partial t^2} = (1 - \gamma) \frac{\partial^2 u^t}{\partial t^2} + \gamma \frac{\partial^2 u^{t+1}}{\partial t^2} \quad (4.46)$$

Using the Hermite-cloud method detailed in Chapter 2, the approximate solution  $u^h(x, t)$  for the displacement  $u(x, t)$  and the corresponding auxiliary equation are written as

$$u^h(x, t) = \sum_{n=1}^{NP} N_n(x) u_n(t) + \sum_{m=1}^{NT} \left( x - \sum_{n=1}^{NP} N_n(x) x_n \right) M_m(x) u_{xm}(t) \quad (4.47)$$

$$\sum_{n=1}^{NP} N_{n,x}(x) u_n(t) - \sum_{m=1}^{NT} \left( \sum_{n=1}^{NP} N_{n,x}(x) x_n \right) M_m(x) u_{xm}(t) = 0 \quad (4.48)$$

The acceleration component in Eq. (4.30), is discretized spatially as

$$\frac{\partial^2 u(x, t)}{\partial t^2} = \frac{\partial^2 u(x, t)}{\partial x^2} = \sum_{n=1}^{NP} N_{n,xx}(x) u(t) \quad (4.49)$$

Substituting Eqs. (4.47) and (4.49) into Eq. (4.44), while taking  $\alpha$  and  $\beta$  values as 0.5, and finally combining with the auxiliary equation Eq. (4.48), the displacement at any time  $(t + 1)$  is discretized in both space and time domains as

$$\begin{aligned} & \left( \begin{bmatrix} [A_{ij}^{L1}]_{NP \times NP} & [A_{ij}^{L2}]_{NP \times NT} \\ [A_{ij}^{L4}]_{NT \times NP} & [A_{ij}^{L4}]_{NT \times NT} \end{bmatrix} \right) \begin{Bmatrix} \{u_i^{t+1}\}_{NP \times 1} \\ \{u_{xi}^{t+1}\}_{NT \times 1} \end{Bmatrix} \\ &= \left( \begin{bmatrix} [A_{ij}^{R1}]_{NP \times NP} & [A_{ij}^{R2}]_{NP \times NT} \\ [A_{ij}^{R4}]_{NT \times NP} & [A_{ij}^{R4}]_{NT \times NT} \end{bmatrix} \right) \begin{Bmatrix} \{u_i^t\}_{NP \times 1} \\ \{u_{xi}^t\}_{NT \times 1} \end{Bmatrix} + \Delta t \begin{Bmatrix} \{v_i^t\}_{NP \times 1} \\ \{v_{xi}^t\}_{NT \times 1} \end{Bmatrix} \end{aligned} \quad (4.50)$$

where

$$v = \frac{\partial u_i}{\partial t} (\text{velocity}) \quad (4.51)$$

$$[A_{ij}^{L1}] = N_j(x_i) - \frac{\Delta t^2}{4} N_{j,xx}(x_i); \quad (4.52)$$

$$[A_{ij}^{R1}] = N_j(x_i) + \frac{\Delta t^2}{4} N_{j,xx}(x_i) \quad (4.53)$$

$$[A_{ij}^{L2}] = [A_{ij}^{R2}] = \left( x_i - \sum_{n=1}^{NP} N_n(x_i) x_n \right) M_j(x_i) \quad (4.54)$$

$$[A_{ij}^{L2}] = [A_{ij}^{R3}] = N_{j,x}(x_i) \quad (4.55)$$

$$[A_{ij}^{L4}] = [A_{ij}^{R4}] = \left( - \sum_{n=1}^{NP} N_{n,x}(x_i) x_n \right) M_j(x_i) \quad (4.56)$$

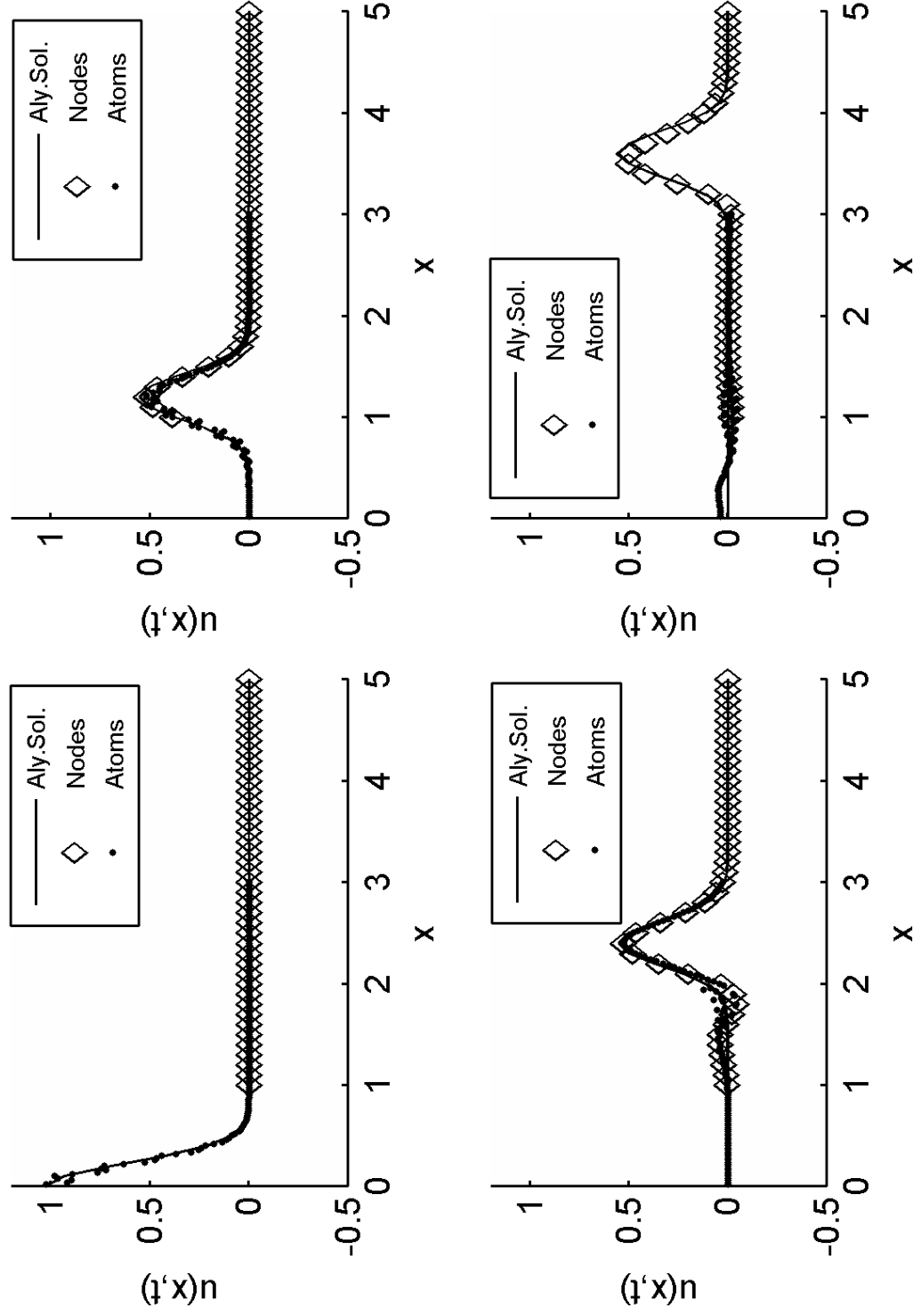
The Leap frog scheme, detailed in Section 2.2.5, is for time integration in the atomic domain and a time step of  $\Delta t^c = 0.01$  units is used in the continuum domain, with the multiplication factor  $n$  defined in Eq. (3.16) taken as 5.



The simulation is run for 360 time steps corresponding to  $t = 3.6$  units in the continuum domain. Figure 4.22 shows four snap shots of the analytical and numerical solution of the normalized displacements at the beginning of the simulation and after every 120 time steps. The analytical solution is obtained using the d'Alembert's solution of the wave equation (Gockenbach, 2002). As the wave is symmetrical about the origin, the results are plotted only for half of the computational domain.

The results of the multiscale simulation are compared with the analytical solution using the global error measure within every time step. Only the atoms and nodes within the domain of influence of the wave are considered for computing the global error, defined in Eq. (4.1). The domain of influence of a point  $x_0$  is defined as the interval  $[x_0 - ct, x_0 + ct]$  over which it has an influence on the solution  $u(x, t)$ , where  $c$  is the wave speed. It is observed from the Fig. 4.22 that the results of the multiscale simulation are in line with the analytical solution. The maximum global error observed during the entire simulation is 0.0545. It is also observed that the wave is able to successfully pass through the transition domain into the continuum region.

The results of the simulation are comparable with those obtained by other multiscale models such as the bridging scale method (Tang et al., 2006a), the pseudo-spectral multiscale method (Tang et al., 2006b) and the finite difference approach with velocity inter-facial conditions (Tang, 2008). In the bridging scale method and the pseudo-spectral multiscale method, the reflections are reduced by using the time history kernel technique to develop inter-facial boundary conditions for atoms at the atomic-continuum interface. The finite difference approach for multiscale simulations uses the velocity inter-facial conditions to reduce the reflections in nonlinear lattices. Although the reflections observed in the proposed method is slightly higher when compared to the other methods mentioned, it possesses other inherent merits, such as the use of a meshless technique and a simple handshaking



**Figure 4.22:** Snapshots of the analytical and multiscale simulation results at different time instances for 1-D wave propagation problem (case-2).

algorithm. The perturbations observed in the numerical solution are due to the reflection of the oscillatory wave, as it passes from the atomic domain into the continuum region, and is caused by the differences in the atomic and nodal spacing.

### Case-3 Anharmonic wave

The same wave problem is now analyzed with a non-linear potential function in the atomic domain. The problem definition, boundary and initial conditions, and the computational model for this problem are same as in case-1, except that in this case the atomic domain is modeled using an anharmonic potential function of the form

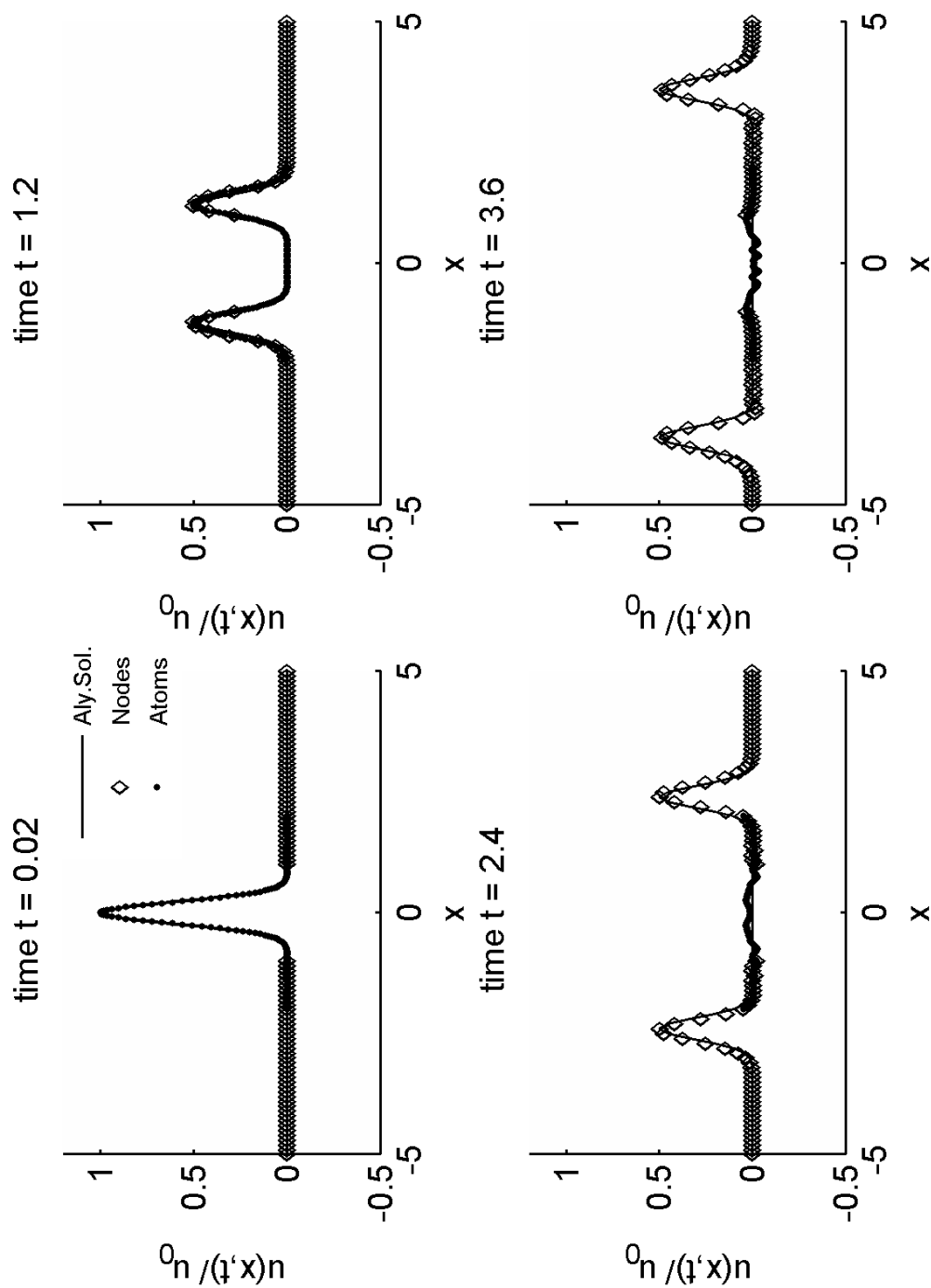
$$\phi(u) = \frac{1}{2}k_1 \sum_n (u_{n+1} - u_n)^2 + \frac{1}{4}k_2 \sum_n (u_{n+1} - u_n)^4 \quad (4.57)$$

The governing equation in the MD domain, taking nearest neighbor interactions, can be written as

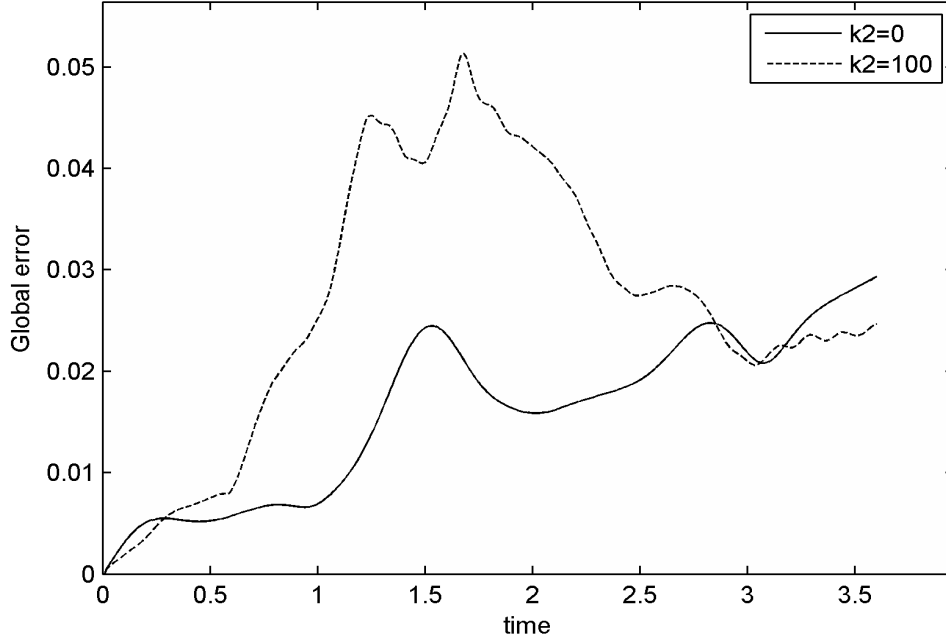
$$mu'' = k_1(u_{n-1} - 2u_n + u_{n+1}) + k_2((u_{n+1} - u_n)^3 - (u_n - u_{n-1})^3) \quad (4.58)$$

where  $k_1$  and  $k_2$  are the constants for the linear and non-linear terms. Taking  $k_2$  as zero would result in case-1. In this problem, the constants  $k_1$  and  $k_2$  are assigned the values 1 and 100 respectively.

The numerical results at different time instances obtained from the multiscale model are presented in Fig. 4.23. Figure 4.24 compares the global error variation during the simulation for both the harmonic (case-1) and an harmonic (case-3) cases and the maximum error values are tabulated in Table 4.3. The non-linearity considered in the atomic region has resulted in greater wave reflections and an increase in error values, as observed from the figure and table. However, it can be observed that the wave is able to pass through the transition region into the continuum region and the global and relative errors observed are still low at 5%.



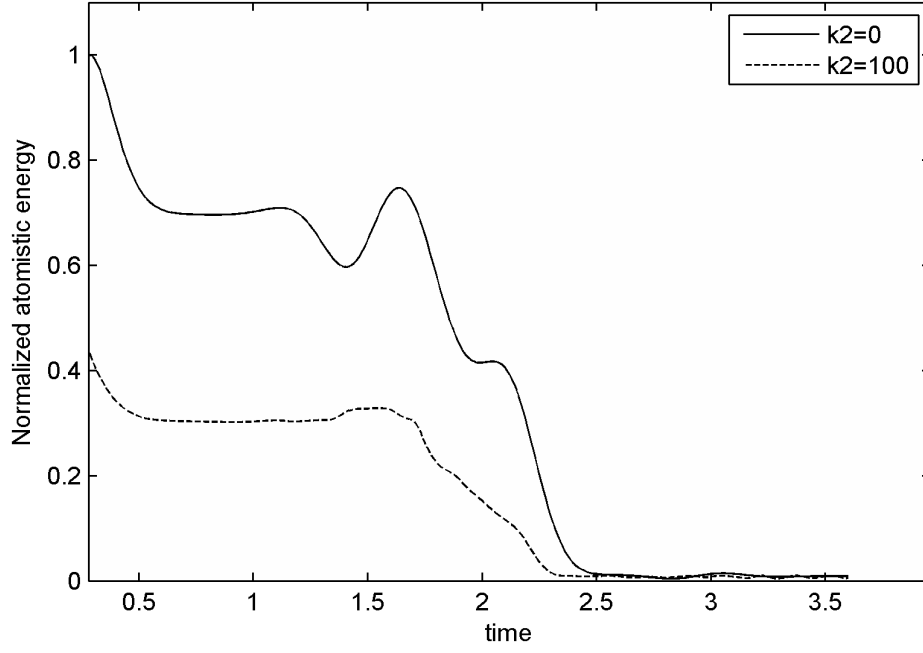
**Figure 4.23:** Snapshots of the analytical and multiscale simulation results at different time instances for 1-D wave propagation problem (case-3).



**Figure 4.24:** Global error variation as a function time for wave propagation problems.

The energy transfer between the atomic and continuum domain can be observed by plotting the total energy of the atomistic region with time. The total energy in the atomistic region is computed as the sum of potential and kinetic energy of all atoms in the atomic and transition regions. As the wave propagates from the atomic region to the continuum region its energy is also transferred. A good coupling scheme must ensure a complete transfer of energy. The energy plot for the atomistic region for both the linear and non-linear test cases is presented in Fig. 4.25.

It can be observed from the figure that the coupling algorithm is able to transfer almost all of the energy out of the atomistic region except for the energy lost due to reflections. The presently developed multiscale model is therefore successful in coupling a linear as well a non-linear atomistic model with a continuum model. The results are also in line with the results obtained by other researchers for such benchmark problems (Xiao and Belytschko, 2004; Tang et al., 2006b). It has to be noted that the atomistic models used in this work assume nearest neighbor interaction and is therefore not strictly global, unlike the potentials encountered



**Figure 4.25:** Energy of the atomistic region as a function of time for wave propagation problems.

in real applications. However, the algorithm can be extended in a straight forward manner to include global effects.

#### 4.2.2 Vibration of a stretched elastic string

Next, we consider the transverse vibration of a stretched elastic string of length  $L$  aligned along the  $x$  axis and fixed at both ends. The string is displaced from its initial position by giving it a small initial displacement. The aim is to obtain the deflection of the string at any time  $t$ . The mass of the string is assumed to be constant and the gravitational forces are neglected. The displacement is strictly limited to the transverse direction. The transverse vibration of the string is governed by the second-order one-dimensional wave equation, with governing equations and boundary conditions given as

$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2} \quad 0 \leq x \leq 1 \quad (4.59)$$

$$u(0, t) = 0 \quad \forall t \quad (4.60)$$

$$u(1, t) = 0 \quad \forall t \quad (4.61)$$

where  $u = f(x, t)$  is the transverse displacement. The initial conditions are defined in terms of the initial displacement and velocity which are taken as

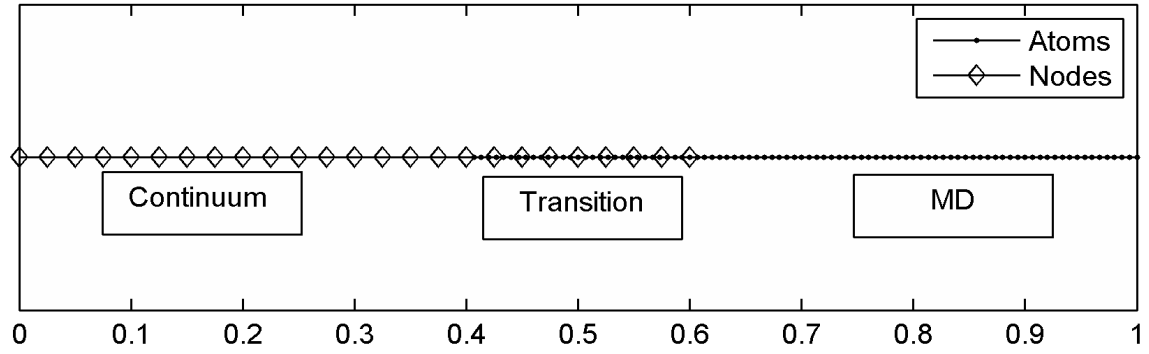
$$u(x, 0) = u_0 \sin(2\pi x) \quad \frac{\partial u}{\partial t}(x, 0) = 0 \quad (4.62)$$

The multiscale model for this problem is set up as shown in Fig. 4.26, and contains 24 nodes and 90 atoms in the atomic and continuum domains, with an average atomic and nodal spacings of 0.00667 and 0.025 units respectively. A time step of 0.01 units is used for time integration in the continuum domain. The exact solution for the problem is

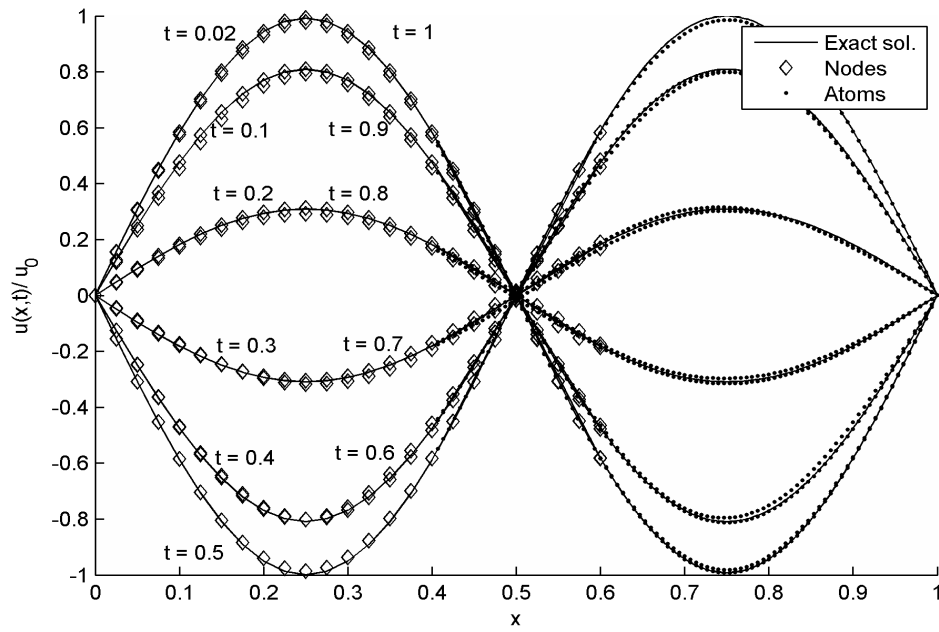
$$u(x, t) = u_o \sin(2\pi x) \cos(2\pi t) \quad (4.63)$$

The normalized displacements at different time instances obtained from the multiscale model are plotted in Fig. 4.27 and the global error values are shown in Table 4.3. The results once again demonstrate the ability of the present multiscale model to effectively couple atomic and continuum domains.

The time step used for numerical integration and the multiplication factor  $n$  used for computing the time step in the continuum domain varies with the problem and has to be chosen judiciously. A smaller value of  $n$  would mean that the time steps in the atomic and continuum domains are not significantly different and hence will not lead to any substantial saving in the computational cost. On the contrary, a large value of  $n$  might lead to a decrease in accuracy of the solution and if not meeting



**Figure 4.26:** Computational model for transverse vibration of a stretched string



**Figure 4.27:** Comparison of analytical and multiscale simulation results at different time instances for the transverse vibration of a stretched string.

the stability requirements in the atomic domain, and may also cause the numerical scheme to fail.



### 4.2.3 Vibration of stretched elastic membrane

Finally, we consider the vibration of a stretched elastic membrane over a square domain of length 5 units. The motion of the membrane is governed by the two-dimensional wave equation

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = \frac{\partial^2 u}{\partial t^2} \quad 0 \leq x \leq 5, \quad 0 \leq y \leq 5, \quad t \geq 0 \quad (4.64)$$

where  $u(x, y, t)$  is the displacement of the membrane. The membrane is fixed along the boundaries, with the following boundary conditions

$$u(0, y, t) = u(5, y, t) = u(x, 0, t) = u(x, 5, t) = 0 \quad (4.65)$$

The membrane is given an initial displacement in the form of a Gaussian wave,

$$u(x, y, 0) = u_0 \exp \left\{ - \left[ (x - 2.5)^2 + (y - 2.5)^2 \right] \right\} \quad (4.66)$$

where  $u_0$  is the amplitude of the Gaussian wave. The initial velocity field is taken as zero. The analytical solution for this problem is obtained by separation of variables as

$$u(x, y, t) = u_0 \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} B_{mn} \varphi_{mn}(x, y) \cos(2\pi\omega_{mn}t) \quad (4.67)$$

where

$$B_{mn} = \left( \frac{25}{4} \pi \right) \sin \left( \frac{m\pi}{2} \right) \sin \left( \frac{n\pi}{2} \right) \exp \left( - \left( \frac{m^2 + n^2}{100} \right) \pi^2 \right) \quad (4.68)$$

$$\varphi_{mn}(x, y) = \sin \left( \frac{m\pi x}{5} \right) \sin \left( \frac{n\pi y}{5} \right) \quad (4.69)$$

$$\omega_{mn} = (1/2) \sqrt{m^2 + n^2} \quad (4.70)$$

The computational model used in this problem is same as that used in the static 2-D problems shown in Fig. 4.12, with the number of atoms and nodes in the atomic,

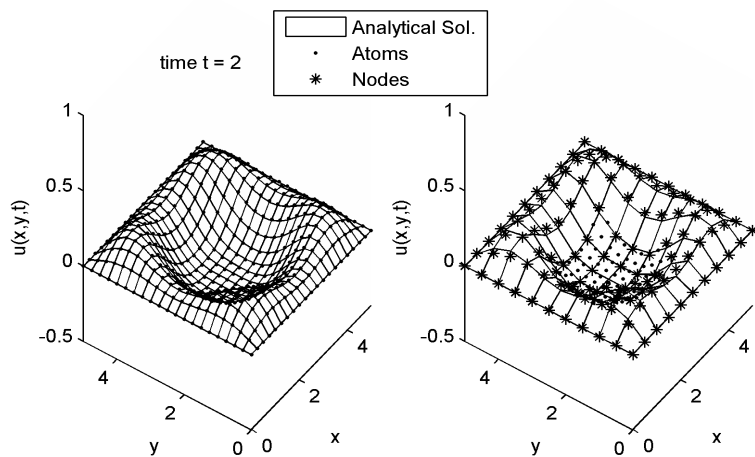
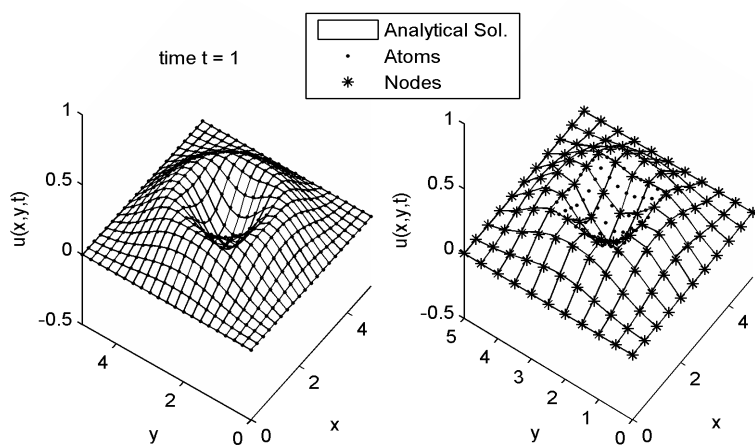
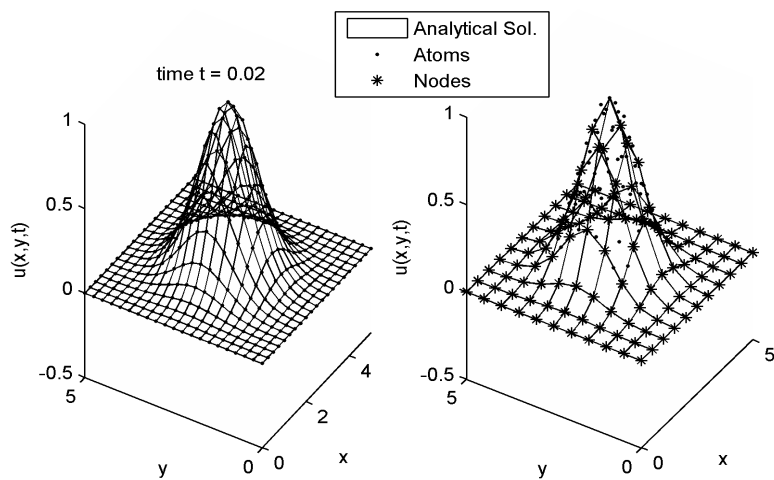
transition, and continuum domains being similar. A time step of  $\Delta t^a = 0.004$  units is used in the atomic domain with the multiplication factor  $n$  taken as 5. The simulation is run for 200 time steps corresponding to  $t = 4$  units in the continuum domain.

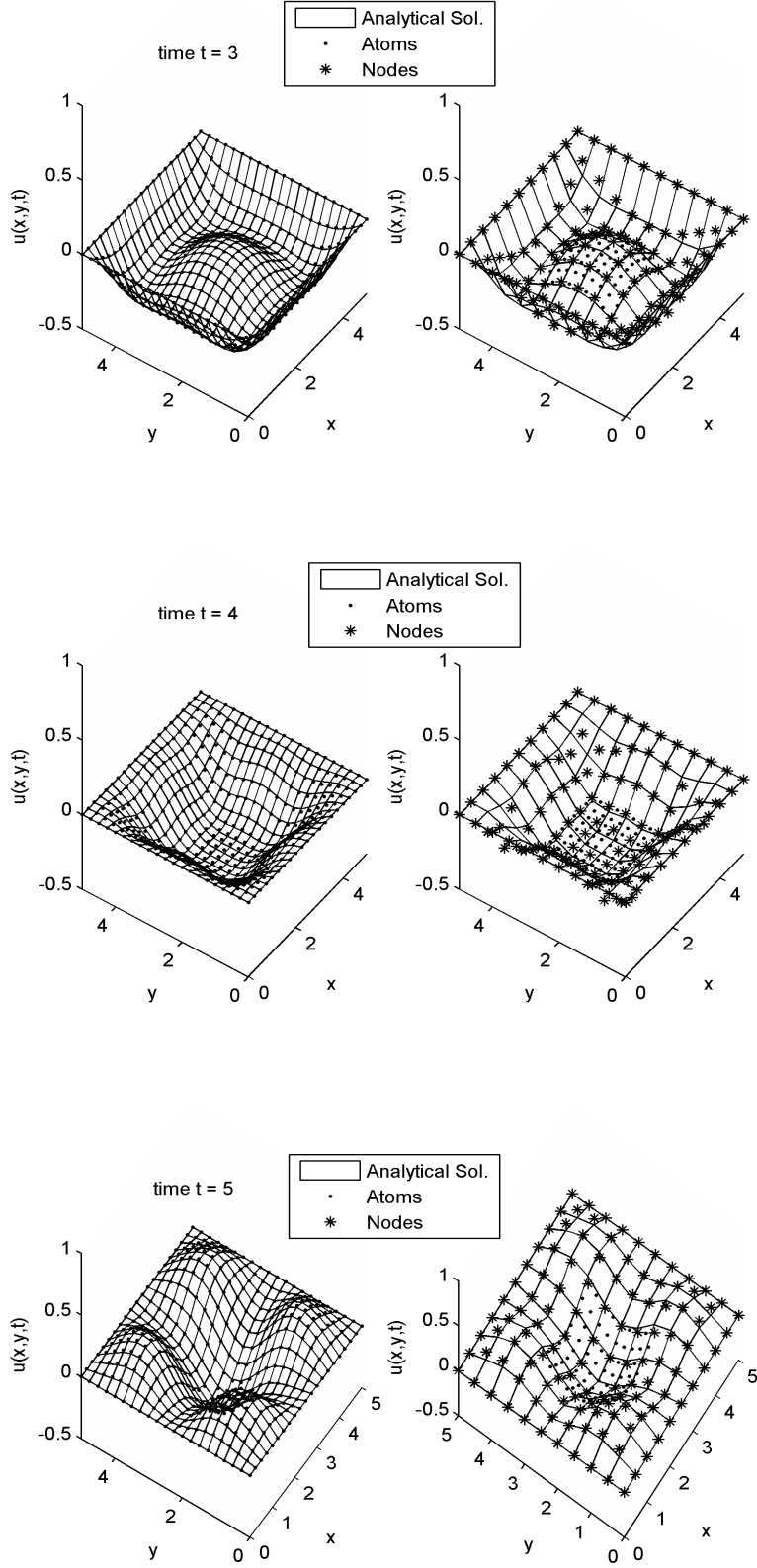
Figure 4.28 shows the snapshots of the MD and numerical results plotted after every 50 time step interval, starting with time  $t = 0$ . The multiscale simulation results are comparable with the results obtained by (Zong and Lam, 2002), with a maximum global error of 0.0726 observed during the entire simulation.

In conclusion, the multiscale method not only fares well in coupling length scales, but also ensures accurate and efficient coupling of both time and length scales, which is well corroborated by the results, displaying smooth propagation of waves from the atomic to continuum domains. The inter-facial reflection observed in the proposed method would form the focus in the future works, wherein the ways to reduce can be explored.

**Table 4.3:** Comparison of global and relative errors for transient 1-D problems.

Problem type	Atoms/node distribu- tion	Global error ( $\psi_1$ ) (multiscale simulation)	Global error ( $\psi_2$ ) (atomic simulation)	Relative error $\xi = \psi_1 - \psi_2$
Wave propagation -Gaussian Wave and harmonic potential.	$81 \times 201$	0.02994	0.002313	0.02767
Wave propagation -‘Hat’ function	$81 \times 201$	0.024948	0.002644	0.0223
Wave propagation - non-linear potential.	$81 \times 201$	0.05641	0.04816	0.00825
Vibration of a stretched string	$24 \times 90$	0.02413	0.00033	0.0238





**Figure 4.28:** Snapshots of analytical and multiscale simulation results at different time instances for 2-D wave propagation problem.

### 4.3 Chapter Summary

In this chapter the present multiscale numerical scheme, developed to couple the atomic and continuum domains, was validated through benchmark test problems in one and two dimensional space. The static examples involving the Poisson and Laplace equations with different boundary conditions and high gradients show that this multiscale method is highly capable of coupling length scales and is able to deliver results that are accurate and comparable to those obtained from pure atomistic simulations. The ability of the current method in coupling both time and lengths scales is further validated through transient wave propagation problems in both one and two dimensional space. The present multiscale method thus fulfills its objective of providing accurate results at a reduced computational expense by employing multiple time/length scales, while ensuring a seamless interface between the two domains. In the next chapter, nanoindentation and nanoscratch experiments on a copper thin film will be investigated using the presently developed multiscale model.

## Chapter 5

# Multiscale Simulation of Nanoindentation and Nanoscratching

With rapid advances in the field nanotechnology, nano-parts and components are increasingly finding applications in the field of semi-conductor, optical and mechanical industries. Nanotribology has therefore emerged as a new field for the characterization of materials at these scales, and is gaining more research focus in recent years. Nanoindentation and nanoscratching are two very popular techniques used for determining the mechanical and wear properties of nanostructures, nanoscale systems such as MEMS/NEMS, thin films, coatings and nanocomposites.

Numerical simulations of nanoindentation and nanoscratching experiments have been successfully performed and compared with experimental data. Molecular dynamics (MD) simulation has been the preferred choice for modeling materials at nanoscales, and nanoindentation and nanoscratching experiments have been successfully simulated using full atomistic models using empirical inter-atomic potentials (Mulliah and et al., 2004; Lee et al., 2005; Komanduri et al., 2000a; Peng

et al., 2010). However, as discussed in earlier chapters, the MD approach is bounded by an upper limit on the number of atoms that can be included in the study. The advantages of a multiscale approach have already been highlighted in the previous chapters, and to support this claim a multiscale model was successfully developed and validated.

In this chapter, nanoindentation and nanoscratching experiments on copper thin films are simulated using the developed multiscale model. In the following sections, the multiscale model for nanoindentation and nanoscratching simulations are explained in detail. Each section begins with a review of the method, followed by a description of the mathematical model, and finally presents the findings of the numerical simulations.

## 5.1 Multiscale Simulation of Nanoindentation

### 5.1.1 Review of instrumented indentation

The term hardness may be defined as the ability of a material to resist permanent indentation or deformation, when in contact with an indenter under load (ASM, 2000). Conventional hardness testing generally involves the pressing of an indenter of known geometry into the test material and measuring the hardness directly or indirectly from the contact pressure. Most commonly used hardness tests include the Brinell, Rockwell, Vickers and Knoop tests. In these tests, the hardness value may be expressed as the contact pressure (load by supporting area in  $\text{kgf/mm}^2$ ) or in the form of a hardness number, as in the case of Rockwell tests. Micro-indentation hardness testing is used to measure very low hardness values. The test uses a diamond indenter of a known geometry, which is forced into the surface, with applied forces in the range of 1 to 1000gf (ASM, 2000). The hardness is found from the resulting indent on the specimen. Microhardness testing is very similar to

conventional hardness tests, except that it is done on a microscopic scale using high precision instruments.

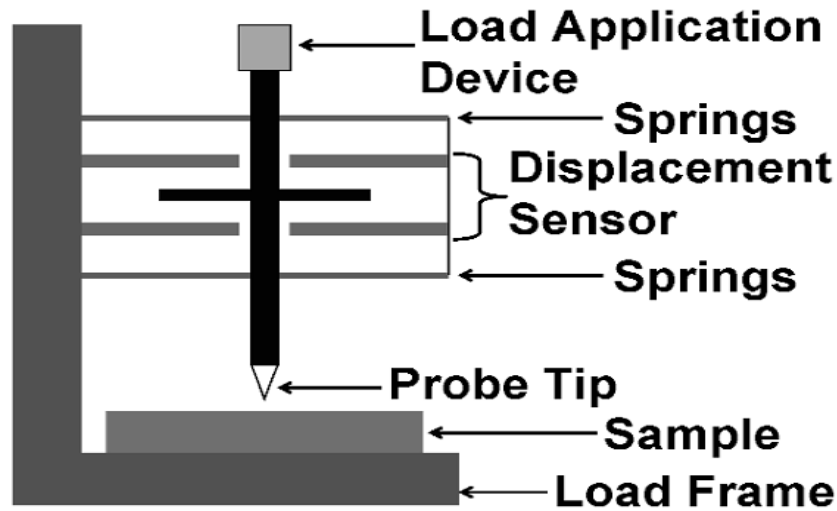
On the other hand, instrumented indentation or nanoindentation is a relatively new technique used for characterization of materials at nanoscales. Unlike conventional hardness testing, nanoindentation test involves continuous monitoring of load and displacement information over the entire loading cycle to evaluate hardness (Oliver and Pharr, 1992; Pharr, 1998; Oliver and Pharr, 2004). Modern instruments allow loads as small as 1nN to be applied, and displacements in the range of 0.1nm to be measured (ASM, 2000).

The major advantage with the nanoindentation technique is that there is no need to measure the resulting indent, greatly reducing the possibility of errors. Instead, the method uses load-displacement data to obtain the contact area, and thereby evaluate the hardness of the material. The load-displacement data can in addition be used to obtain a lot of other useful information such as the Young's modulus, fracture toughness, strain hardening index, stain hardening rate, yield strength, residual stresses, etc. (Fischer-Cripps, 2006). Due to their inherent advantages, the nanoindentation technique is increasingly being used to study the tribological properties of metals, thin films and coatings (VanLandingham, 2003).

### **Testing method**

A schematic of the nanoindentation test set-up is shown in Fig. 5.1. The experiment involves pressing an indenter of known geometry into the test specimen, with a known load applied using actuators, and measuring the displacement through displacement sensors. At maximum load the indenter is held for some time to minimize creep effects, which is then followed by an unloading cycle. The experiment is carried out in a controlled environment to avoid thermal drift, which may occur due to changes in the dimensions of the specimen or the indenter due to temperature





**Figure 5.1:** Schematic of nanoindentation test setup (VanLandingham, 2003).

fluctuations. The unloading rate is constant and elastic recovery takes place during the unloading cycle. However, the unloading curve is assumed to be purely elastic and any plastic recovery during the unloading phase is assumed to be small and therefore neglected (Oliver and Pharr, 2004).

Typical indenters used in nanoindentation tests include the Berkovich, Vicker's, Knoop, and Cube corner indenters. The Berkovich indenter, with a face half-angle of  $\theta = 65.3^\circ$ , has the same projected area to depth ratio as that of a four sided Vickers indenter, and is mostly preferred in nanoindentation experiments. The Berkovich indenter can be ground to have a sharp tip and its face angle can be made accurately, it is therefore often used for hardness and Young's modulus measurements.

### **Interpreting the load-displacement curve**

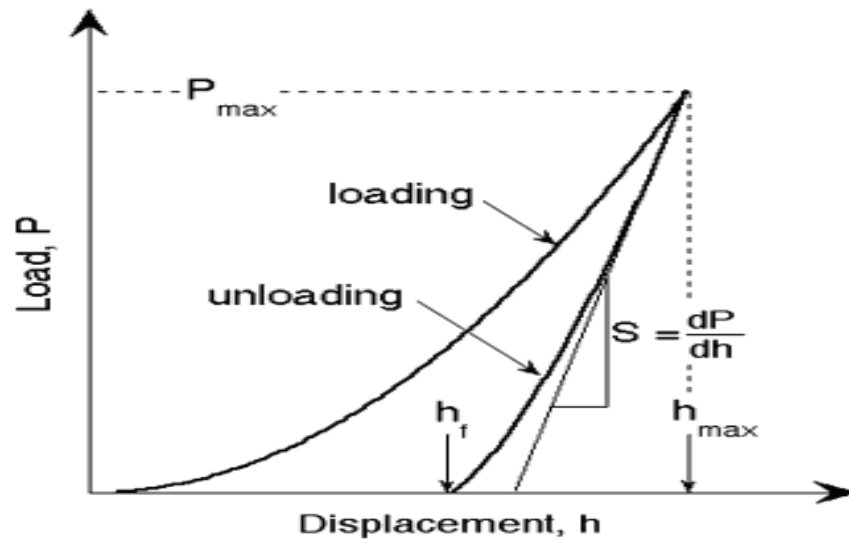
Hardness value in an nanoindentation experiment is given by the contact pressure at the maximum load, which is chosen such that it produces permanent plastic strain in the material. In this state of permanent plastic strain, a further increase in load produces a corresponding increase in the contact area, thereby keeping the contact

pressure constant. When this state is reached, the contact pressure can be equated to the hardness of the material and the resulting hardness value is referred to as the indentation hardness (Fischer-Cripps, 2006).

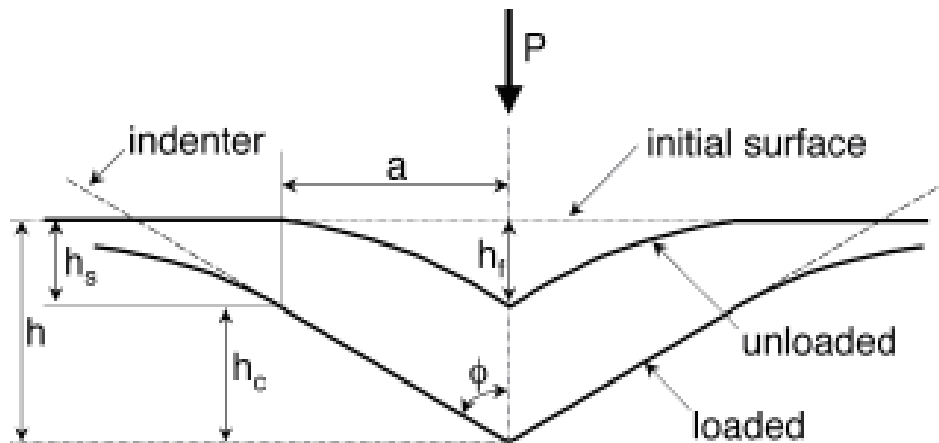
A typical load-displacement graph obtained from a nanoindentation test is shown in Fig. 5.2. The loading phase is assumed to be both elastic and plastic, while only elastic recovery is assumed to take place during the unloading phase. In the figure,  $P$  is the applied load and  $h$  is the displacement, measured with respect to a reference point or datum. The load-displacement ( $P$ - $h$ ) curve contains the following information; the maximum load  $P_{max}$  and the corresponding displacement  $h_{max}$ , the elastic unloading stiffness,  $S = dP/dh$ , defined as the slope of the linear portion of the unloading curve at the beginning of the unloading cycle, and the final depth of the resulting indent  $h_f$ , measured after the removal of the indenter. The accuracy of the properties determined from the load-displacement graph depends on how accurately these values are measured. The final displacement is less than the maximum displacement because of elastic recovery that takes place during the unloading process. The most important parameter to note from the load-displacement curve is the unloading stiffness that is used to calculate the area of contact.

A schematic of the deformation pattern observed during an indentation test is shown in Fig. 5.3, where  $h$  is the total depth of penetration,  $h_f$  the final depth after removal of load,  $h_c$  the contact depth, which is the depth until which the indenter and the material are in contact, and  $h_s$  the sink-in depth that denotes the amount by which the material sinks in along the line of contact of the indenter and the specimen, due to elastic recovery.

The unloading stiffness and contact area have to be determined to evaluate the hardness and elastic modulus of the material. The contact stiffness is found from the slope of the initial portion of the unloading curve. This is done by fitting



**Figure 5.2:** Schematic of load-displacement curve from an indentation experiment (Oliver and Pharr, 2004).



**Figure 5.3:** Schematic of deformation pattern observed during an indentation test (Oliver and Pharr, 2004).

a curve from the unloading data and then finding the slope of the resulting fit. Earlier, Doerner and Nix (1986) used a linear fit for the unloading curve based on a flat punch approximation. Later, Oliver and Pharr (1992) improved this, by using a power law relation of the form

$$P = \alpha(h - h_f)^m \quad (5.1)$$

where  $\alpha$  and  $m$  are the fitting constants that are determined empirically for different materials. The value of  $m$  is typically in the range of 1.2 to 1.6. The unloading stiffness is then found as the slope of the resulting fit, obtained by differentiating the above equation at the maximum depth  $h_{max}$  as

$$S = \left( \frac{dP}{dh} \right)_{h=h_{max}} = \alpha m (h_{max} - h_f)^{m-1} \quad (5.2)$$

The projected contact area is found using an area function  $A = f(h_c)$ , which relates the contact depth and the projected cross sectional area as

$$A = c_0 h_c^2 + c_1 h_c + c_2 h_c^{1/2} + c_3 h_c^{1/4} + \dots c_8 h_c^{1/128} \quad (5.3)$$

where  $c_0 \dots c_8$  are constants whose values depend on the indenter geometry. For a perfect Berkovich tip, the constant  $c_0$  takes a value of 24.56 (for an ideal Berkovich indenter, the projected area is given as  $3\sqrt{3} \tan^2 \theta h_c^2$ , and substituting  $\theta = 65.3^\circ$  we obtain  $A = 24.5 h_c^2$ ). The constants  $c_1 \dots c_8$  are used to account for the non-ideal indenter geometry and blunting of the tip (Oliver and Pharr, 1992).

The contact depth in Eq. (5.3) is determined from the load-displacement graph as

$$h_c = h_{max} - \epsilon \frac{P_{max}}{S} \quad (5.4)$$

where  $\epsilon$  is a constant that depends on the indenter geometry, and takes a value of 0.72 for a conical punch or 0.75 for a Berkovich indenter. The second term in the above equation is included to account for sink-in effects, when a flat elastic half space is indented by a rigid punch (Oliver and Pharr, 2004). The above equation however does not account for material pile-up at the periphery. This assumption works well in testing of hard materials, where pile-up is negligible, but may overestimate the hardness values when indenting soft materials at high loads. Finally, the hardness of the material is determined from the maximum load and projected contact area as

$$H = \frac{P_{max}}{A_c} \quad (5.5)$$

The Young's modulus, obtained from the load-displacement curve is the effective Young's modulus that accounts for displacements in both the indenter and the specimen, and is given as

$$\frac{1}{E_{eff}} = \frac{1 - \nu^2}{E} + \frac{1 - \nu'^2}{E'} \quad (5.6)$$

where  $E_{eff}$  is the effective Young's modulus of the indenter and the specimen, and  $E$ ,  $\nu$ ,  $E'$  and  $\nu'$  are the Young's modulus and Poisons ratio of the specimen and indenter respectively. The effective Young's modulus can be related to the projected contact area and unloading stiffness through the relationship developed by Sneddon (1965), for the indentation of an elastic half space by a solid of revolution as

$$E_{eff} = S \frac{\sqrt{\pi}}{2\beta\sqrt{A}} \quad (5.7)$$

where  $\beta$  is a dimensionless parameter that depends on the indenter geometry and is introduced to account for stiffness variation caused by non axi-symmetric indenters. The  $\beta$  value for a standard Berkovich indenter is 1.034.

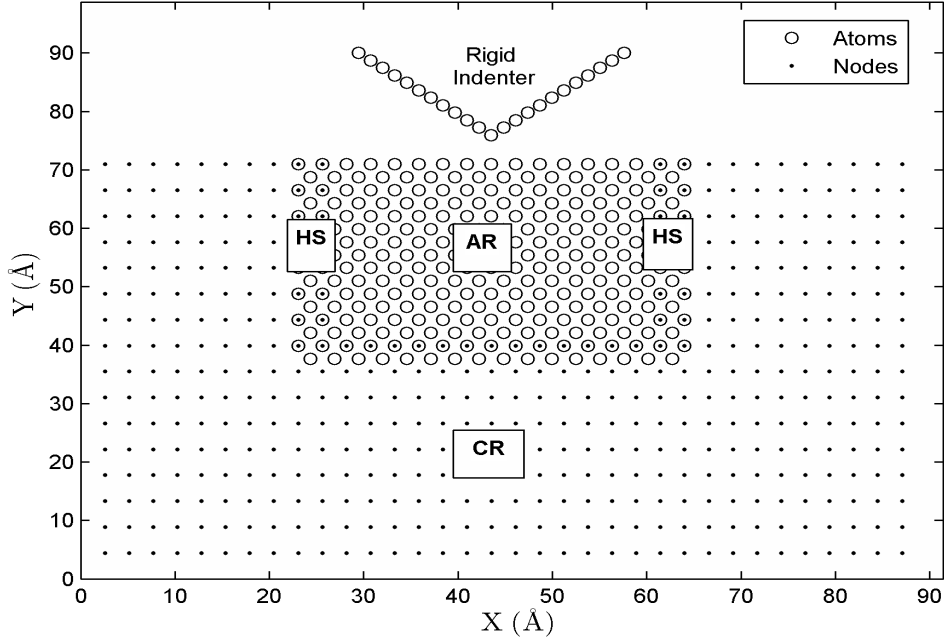
### Error sources

Although nanoindentation testing is a well established technique, it is prone to errors which have to be properly accounted to obtain accurate results. The major sources for errors in nanoindentation experiments are as follows:

- Establishing the datum or reference point before the start of indentation can be a possible source of error. The initial contact depth has to be determined accurately and added to the final depth measured from the experiment to avoid errors when computing the hardness.
- Load frame compliance might cause the displacement of the load frame to be added to the total depth of indentation. Therefore, the load frame has to be calibrated accurately and the displacement due to load frame compliance has to be subtracted from the final displacement.
- Errors might be introduced due to the inherent nature of the specimen leading to sink-in or pile up effects, causing an increase or decrease in measured area of contact, and resulting in incorrect hardness.

#### 5.1.2 Multiscale model

The 2D multiscale model for the nanoindentation problem considered in this study is shown in Fig. 5.4. The model consists of a copper thin film and a rigid triangular indenter. The copper film is of dimensions  $87.04\text{\AA} \times 70.95\text{\AA}$ , and is divided into three regions viz. the atomistic, continuum and handshake regions. The handshaking region that bridges the atomistic and continuum domains consists of both atoms and nodes. A triangular indenter of size  $28.16\text{\AA} \times 14.08\text{\AA}$ , and an included angle of  $90^\circ$  is used in the simulation. The indenter is assumed to be rigid, which means that there are no interactions between the atoms of the indenter.



**Figure 5.4:** Multiscale model of nanoindentation showing atomic (AR), continuum (CR) and handshaking (HR) regions.

The copper film is modeled with both atomistic and continuum approaches, while only an atomistic approach is used to model the indenter. The region close to the indentation site, where the deformation gradient and strains rates are large, is modeled with atomistic precision, and molecular dynamics simulation is used to obtain the trajectories of the atoms in this region. The continuum approach is used in the regions that are far from the indentation site, where the deformation gradient and the strains rates are much lower and deformations are mostly elastic. The continuum region is discretized using meshless nodes and the displacements are obtained using the Hermite-cloud method.

The rigid indenter is initially placed at a distance of  $5\text{\AA}$  from the surface of the film to avoid any attractive forces to be developed between the indenter and the film, before the start of the simulation. The displacements along the three edges of the film are constrained in both the  $x$  and  $y$  directions, and the top surface facing the indenter is set free. The simulation is displacement controlled, meaning that the indenter is incrementally moved towards the film from its initial position at a

constant speed of 100m/s until it reaches a depth of 10Å into the film, after which it is held for 100 times steps before being retracted back at the same speed.

The indentation speed, which is set at 100m/s is higher than the speeds encountered in indentation experiments ( $10^{-9}$  to  $10^{-6}$ m/s) (Komanduri et al., 2000a). However, due to the limitations on the computational resources, it is not possible to perform MD simulations at lower speeds. In general indentation simulations use indentation speeds in the range of 1-100m/s (Lee et al., 2005). Though the indentation speed used in the simulation is on the high side, it is still sufficiently slow for the simulation to be considered quasi-static, within each time step (Zhu et al., 2005).

As the copper film is modeled in 2D, plane strain conditions are assumed for continuum calculations. The Young's modulus and Poisson's ratio for the copper substrate are taken as 128GPa and 0.34 respectively (ASM, 1990). The calculations are carried out iteratively, using the multiscale algorithm detailed in Table 3.1, until convergence is attained. For the purpose of comparison, the simulation is also repeated with a full atomistic model, solved using the MD approach.

### **Atomistic model**

In the atomistic region, the molecular dynamics method is used to obtain the atomic trajectories, defined in terms of the atomic position and momentum as mentioned in Section 2.2. Initially, the face centered cubic atoms of copper are placed in their respective lattice positions along the closely-packed  $\langle 111 \rangle$  plane. The indenter is modeled in the form of a simple triangle and is assumed to be rigid, which implies that there is no change in the relative position of the indenter atoms during the simulation. A pair-wise Morse potential is used to model the interactions between the copper atoms in the film, and also the interactions between copper atoms in the



film and the atoms of the indenter, as

$$\varphi(r_{ij}) = D_o \left( e^{-2\alpha(r_{ij}-r_o)} - 2e^{-\alpha(r_{ij}-r_o)} \right) \quad (5.8)$$

where  $\varphi(r_{ij})$  is the potential energy function,  $r_{ij}$  the inter-atomic distance between the atoms  $i$  and  $j$ , and  $D_o$ ,  $\alpha$  and  $r_o$  are the constants that correspond to the cohesion energy, elastic modulus and equilibrium bond distance, respectively. The potential parameters for the copper-copper interactions in the film, and between the copper atoms in the film and those of the indenter, are given in Table 5.1 (Inamura et al., 1992). To reduce computational effort, only nearest neighbor interactions are considered with a cut-off radius set at  $2.5r_o$ . All calculations are performed in reduced units.

The interactive force  $F_i$ , on an atom  $i$ , is obtained as the negative derivative of the interaction potential defined above as

$$F_i = - \sum_{j=1, j \neq i} \nabla \varphi(r_{ij}) = 2\alpha D_o \left( e^{-2\alpha(r_{ij}-r_o)} - e^{-\alpha(r_{ij}-r_o)} \right) = m_i \frac{d^2 r_i}{dt^2} \quad (5.9)$$

where  $m_i$  is the mass of atom  $i$  and  $r$  is its atomic vector position. The Newton's equations of motion are then numerically integrated to obtain the atomic position and velocities.

The Leap-Frog scheme, explained in Section 2.2.5, is used for integrating the

**Table 5.1:** Morse potential parameters.

Parameter	Copper-Copper	Copper-Indenter
D	0.3429	0.1
$\alpha$	1.3588	1.7
$r_o$	2.7202	2.2

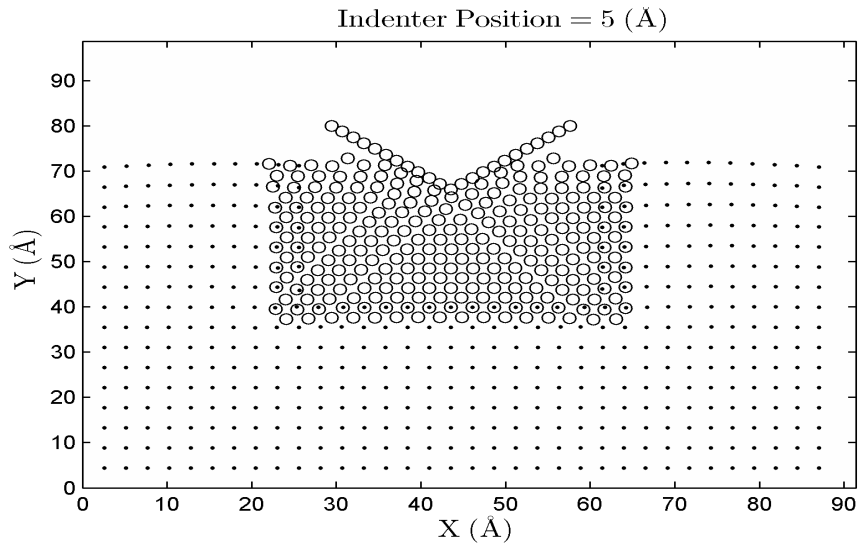
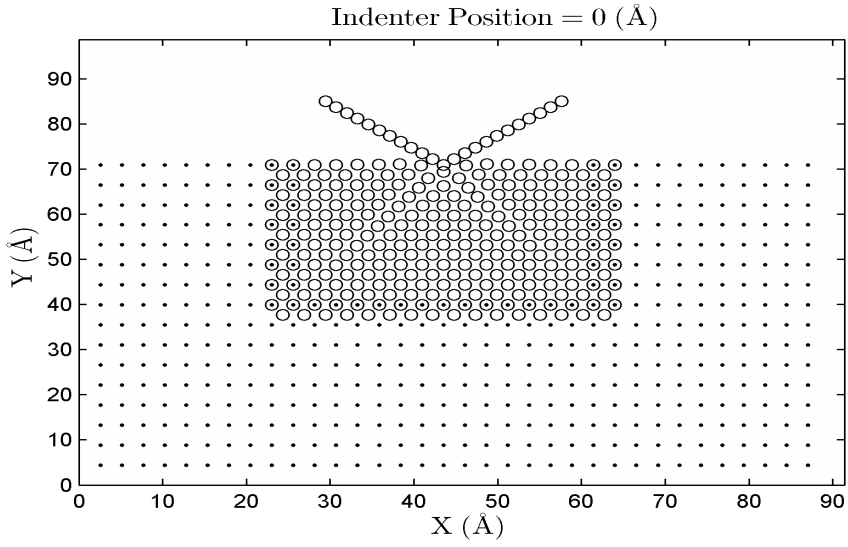
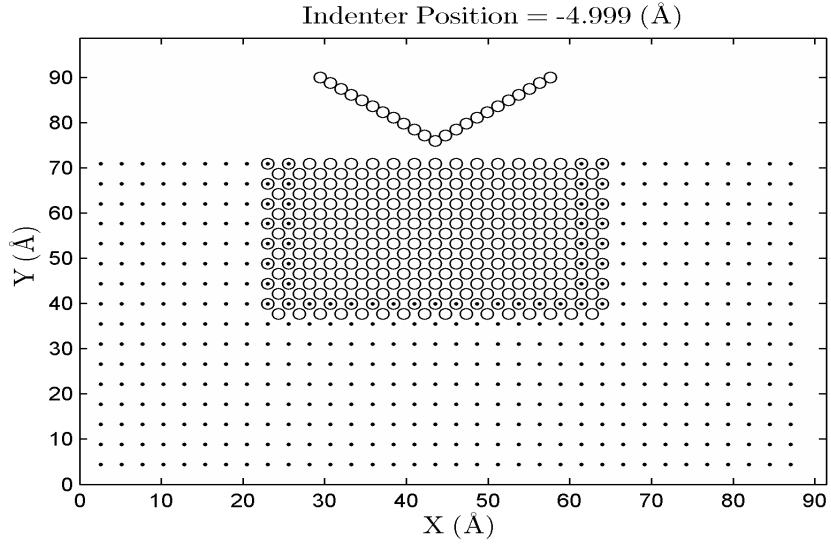
equations of motion in the atomic domain, with the time step  $\Delta t$  chosen such that it is smaller than the time period of thermal oscillations in the system but sufficiently large to lead to achieve a reasonable computational time (Wang et al., 2008). Based on this criteria, the time step for integration is set at  $\Delta t = 1\text{fs}$ .

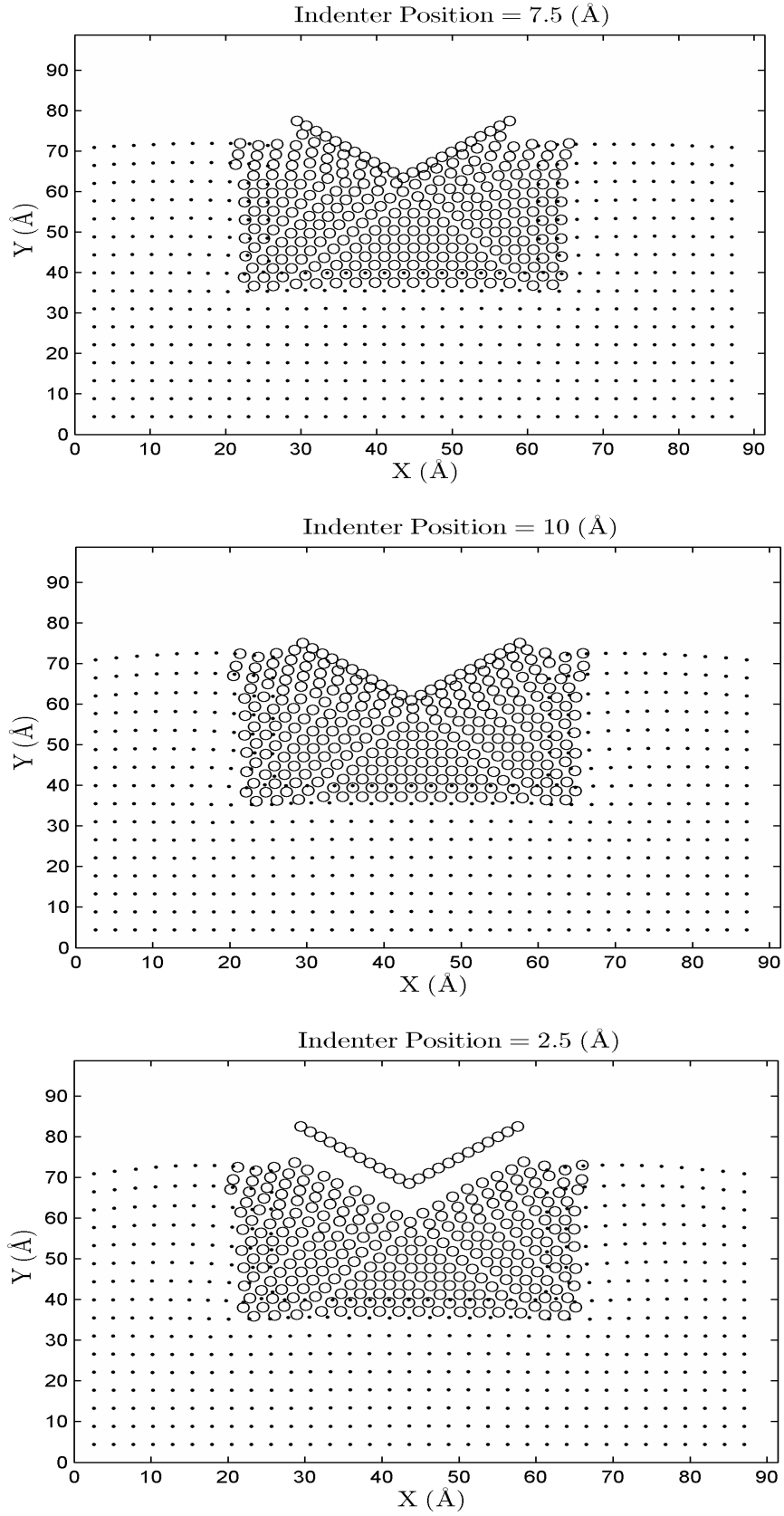
### 5.1.3 Numerical results

In this section, we present the results of the nanoindentation problem simulated using the developed multiscale model. The results of the simulation are also compared with a full atomistic model solved using the MD approach. The simulation begins with the rigid indenter initially positioned at a distance of  $5\text{\AA}$  above the copper film. Figure 5.5 shows the snapshots of the simulation at different stages of the indentation process, starting from the initial position, progressing to the touch-down phase on the film, after which the indenter moves further into the film until it reaches a maximum depth of  $10\text{\AA}$ , and it is held for  $100\text{fs}$  and finally retracted back at the same speed.

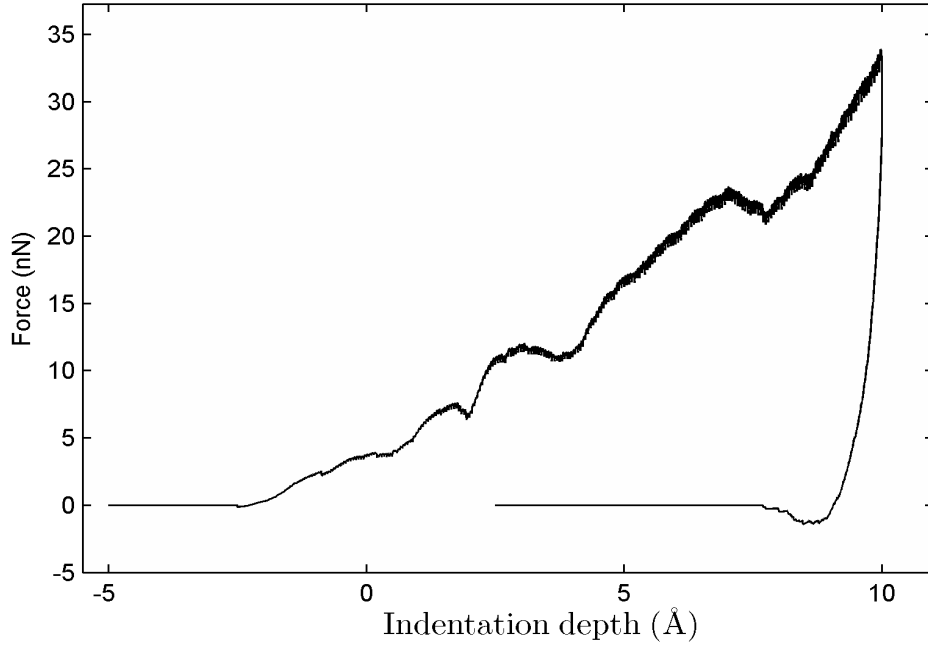
Within each time step, the force on the indenter is found by summing up the forces on all indenter atoms. The load-displacement (force vs. indentation depth) curve obtained from the simulation is shown in Fig. 5.6. In the figure, the vertical axis corresponds to the net force experienced by the indenter during indentation, with positive and negative values corresponding to repulsive and attractive forces respectively. Similarly, the horizontal axis corresponds to the indentation depth, with a negative value indicating that the indenter position is above the copper film.

A maximum normal force of  $34.01\text{nN}$  was observed just before the indenter reaches its maximum depth of  $9.97\text{\AA}$ . Small negative forces of  $0.13\text{nN}$  and  $1.4133\text{nN}$  respectively were observed just before the indenter touches the film and immediately after it was retracted from the film, and were caused by the attractive forces developed between the atoms of the indenter and the film. The load-displacement





**Figure 5.5:** Snapshots of the simulation at different stages of indentation.

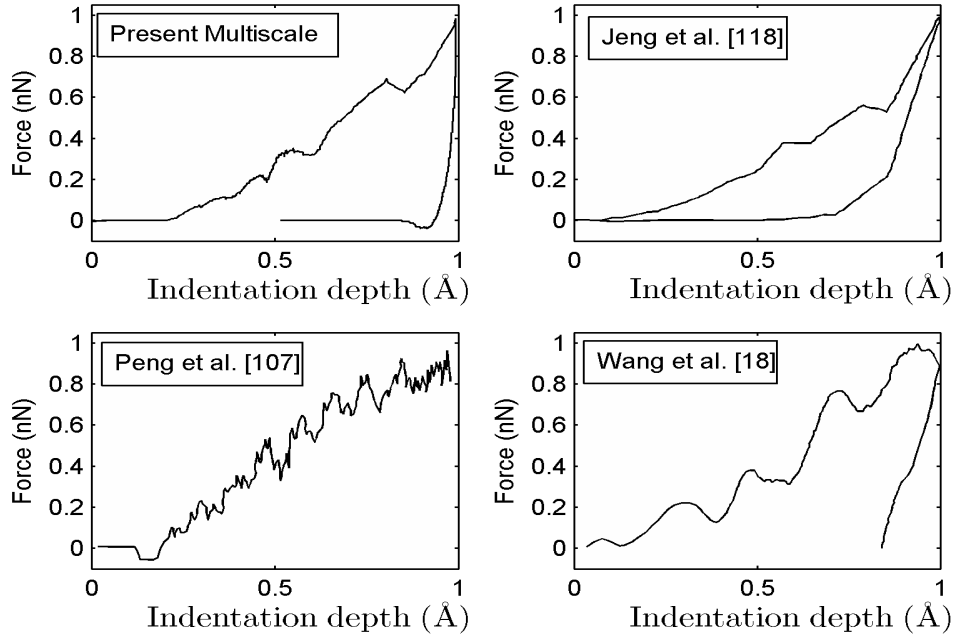


**Figure 5.6:** Load-displacement curve from multiscale simulation.

curve exhibits the general traits observed in nanoindentation experiments. The initial elastic deformation is characterized by an increase in the force with an increase in indentation depth. Distinct peaks observed in the indentation curve, followed by subsequent drop in forces indicate plastic deformation. Also, the forces are higher at successive peaks, which show that with an increase in indentation depth a higher force is required to cause plastic deformation, indicating some strain-hardening in the material.

Figure 5.7 compares the load-displacement curve of the present simulation with those obtained from numerical simulations of indentation carried out on FCC metals, obtained from open literature (Wang et al., 2008; Peng et al., 2010; Jeng and Tan, 2004). As the simulation parameters are different in each case, the load-displacement curves are all normalized for comparison. From the figure it can be observed that the results of the present multiscale simulation exhibit similar trends with those reported in the literature.

The displacement profile at the end of the simulation is shown in Fig. 5.8, and it can be observed that the maximum displacement occurs below the indenter and



**Figure 5.7:** Comparison of the load-displacement curves.

the displacement gradually reduces away from the indentation site. There are no abrupt changes or discontinuities observed in the displacement gradient supporting the fact that the handshaking algorithm in the present multiscale method has been successful in ensuring the seamless exchange of information between the atomic and continuum domains. Some material pile-up is observed near the indentation site, as evident from the displacement profile. Such pile-up phenomena have been observed in indentation testing of metallic materials, specifically while testing copper specimens (Beegan et al., 2003; Suresh et al., 1999; McElhaney et al., 1998).

Figure 5.9 compares the results of the multiscale simulation against a full MD simulation, where the entire film is modeled using the atomistic approach. Though the two indentation curves follow each other closely in the initial stages, they proceed to take slightly different paths thereafter, with a maximum variation of 17% observed in the force-values. While the forces computed at every time instance, using the MD and multiscale approaches do show variations, both curves have a similar profile, indicating the adequacy of the present multiscale method for nanoindentation simulation. Such variations have also been reported by other researchers, where

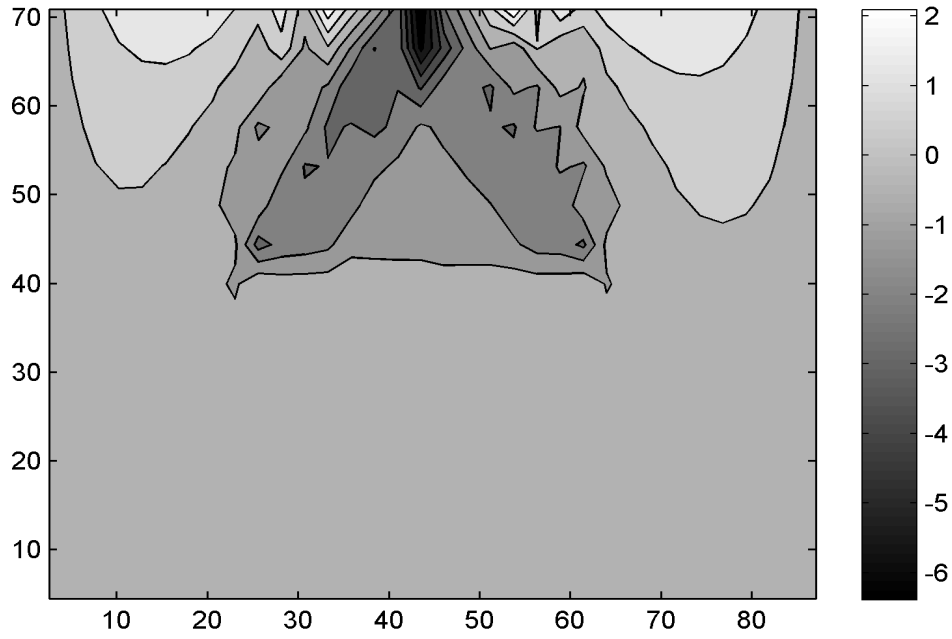
they have reasoned it to the differences in precision level of MD and continuum simulation, the differences in boundary conditions (Picu, 2000), the inconsistencies in the multiscale method (Liu et al., 2004), the linear elasticity assumption in the continuum region, and the presence of spurious forces at the interface (Shilkrot et al., 2002b). The linear assumption in the continuum region renders the multiscale method stiffer, which explains why the maximum force obtained from the multiscale method is lower than the atomistic model.

Comparing the results of the present multiscale method with the pure MD results, we find that the present multiscale method is accurate and able to achieve reasonably comparable results as that of a full atomistic simulation, at a fraction of the computational cost. For simulations done on a Pentium Duo Core Machine (3GHz) with 3.25GB RAM, the clock time noted for the full MD simulation was 3,038s against 235s for the multiscale method, a speed-up of more than one order, albeit with some reduction in accuracy. The present multiscale method has however been able to capture the load-displacement curve with reasonable accuracy, and its performance and accuracy can be further improved by addressing some of the issues mentioned earlier.

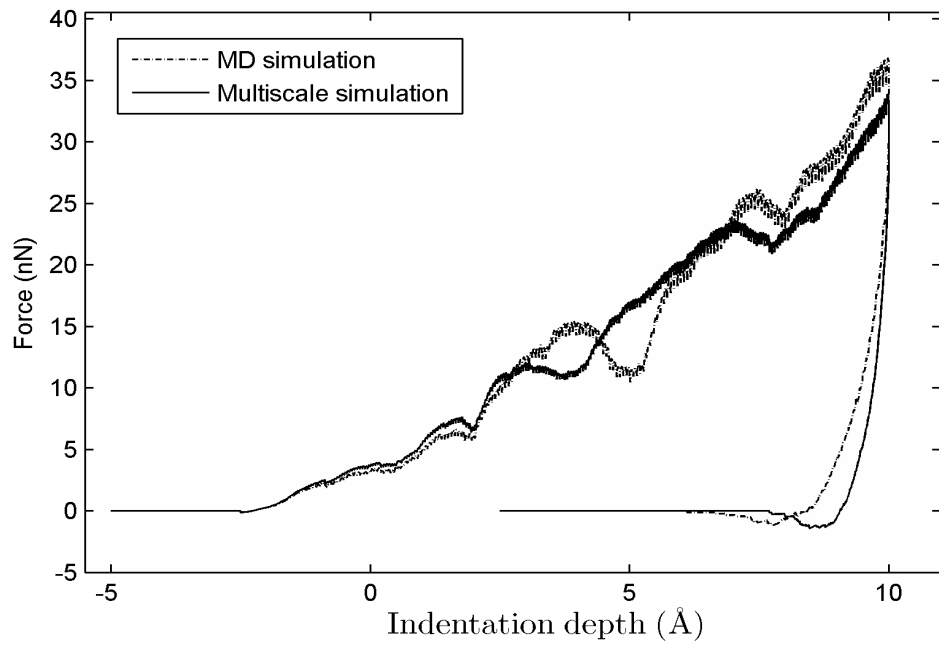
As the current simulation is done in 2D, the contact area defined in Eq. (5.5) is

**Table 5.2:** Force/unit length values from MD and multiscale simulations.

Method	Included Angle of Indenter	Maximum Force (nN)	Contact Perimeter ( $\text{\AA}$ )	Force per unit Length (N/m)
Multiscale model	$90^\circ$	34.0127	14.142	2.405
	$75^\circ$	25.0051	12.606	1.983
	$105^\circ$	38.5471	16.426	2.346
MD simulation	$90^\circ$	36.8133	14.142	2.603



**Figure 5.8:** Displacement profile at the end of simulation.

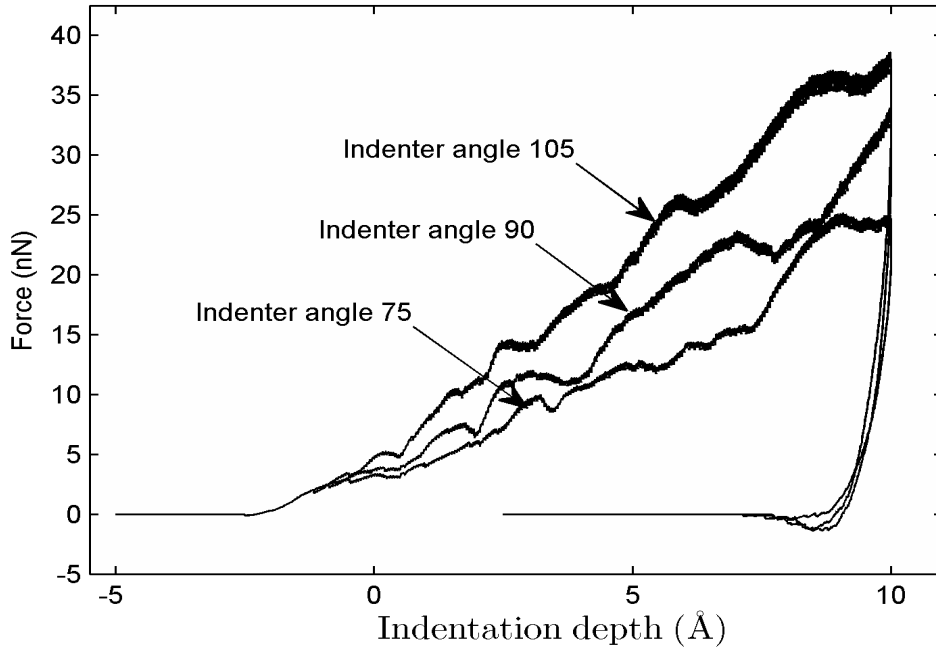


**Figure 5.9:** Comparison of load-displacement curves from MD and multiscale simulation.

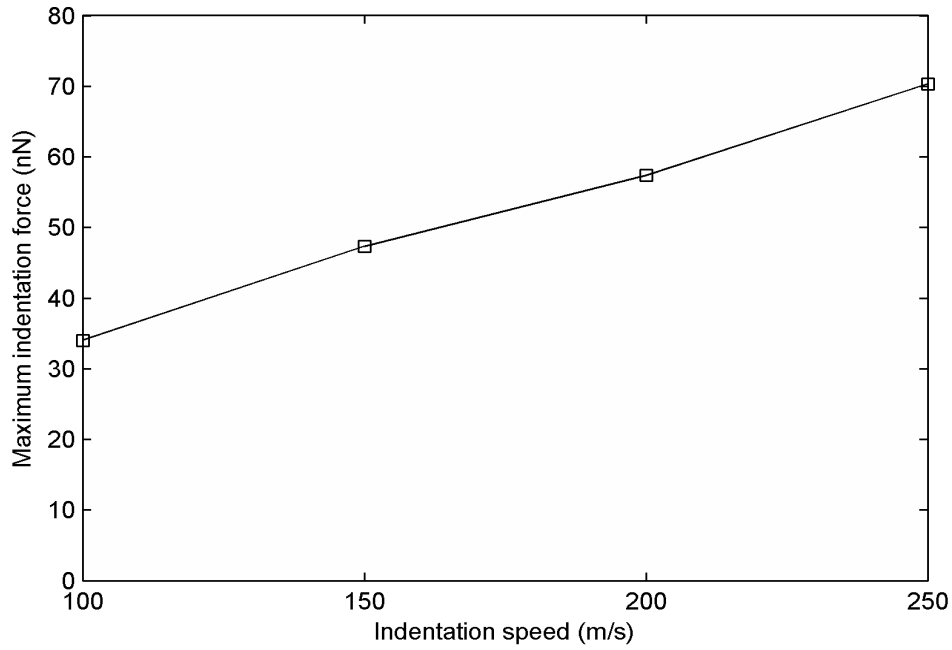


substituted with the contact perimeter, and the force per unit length values are reported, instead of the hardness values. The force per unit length values obtained by both the multiscale method and full MD simulation are reported in Table 5.2. To study the impact of the indenter profile on the measured properties, the simulations are repeated with different included angles of the indenter and the results are also presented in Table 5.2. From the table, we find that for a constant indentation depth, the force per unit length is almost the same for included angles of  $90^\circ$  and  $105^\circ$ . However, for the case of  $75^\circ$  angle, the force per unit length is lower by about 16%. This drop may be attributed to the fact that the sharper indenter, with a reduced contact perimeter, results in a significant reduction of the indentation force. The load-displacement curves for the three test cases are shown in Fig. 5.10, and it can be observed that the curves have a similar profile, with the maximum force increasing for larger indenter angles.

The multiscale simulation was also carried out at varying indentation speed starting at 100m/s and increasing till 250m/s. The maximum force observed in each case is



**Figure 5.10:** Comparison of load-displacement curves for different indenter angles.



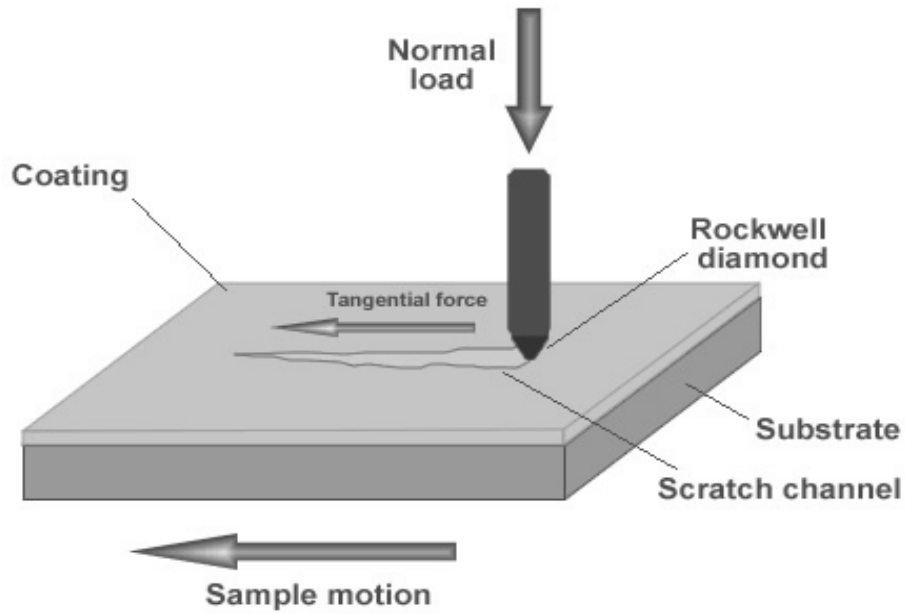
**Figure 5.11:** Plot of maximum indentation force at different indentation speeds.

plotted in Fig. 5.11. From the figure, we find that the maximum force gradually increases from 34.01nN at 100m/s to 70.32nN at 250m/s, which is consistent with the other experimental work (Peng et al., 2010; Saha and Nix, 2002) that have noted an increase in hardness at higher indentation speeds.

## 5.2 Multiscale Simulation of Nanoscratching

### 5.2.1 Review of nanoscratch testing

Nanoscratch testing is an experimental technique used to evaluate the adhesion and wear properties of nanoscale systems, thin films and coatings. A typical nanoscratching test set-up is shown in Fig. 5.12. During the test, scratches are made on the test specimen with a indenter of known geometry, which is drawn across the specimen at a constant speed and a constant or progressively increasing load. The testing involves indentation followed by scratching along the scratch direction, and finally unloading of the specimen. The critical load in a scratching test is defined



Source: <http://www.pvd-coatings.co.uk/pvd-coating-technology/testing-equipment/scratch-tester/>

**Figure 5.12:** Schematic of nanoscratching test setup.

as the load at which failure occurs, when a constant load is used. When using an increasing load, it is defined as the smallest load at which failure occurs. The load at which the material fails is indicative of its wear resistance. Nanoscratch testing can be carried out in the same machine used for nanoindentation testing.

A  $60^\circ$  conical diamond indenter, with a tip radius of  $1\mu\text{m}$  is generally used in scratching tests as its easy to align it along the scratch direction. Typical scratch distance in these tests is of the order of  $500\mu\text{m}$ , with the scratch velocity and loads in the range of  $5\mu\text{m/s}$  and  $0\text{--}2.5\text{mN}$  respectively (Li and Bhushan, 2002). The main variables in nanoscratch testing are the indentation depth, scratch velocity, scratch direction and the indenter shape, which have influence on the friction coefficient, scratch hardness, friction force and abrasive wear properties measured from the test.

### 5.2.2 Multiscale model

The multiscale model for the nanoscratching problem is somewhat similar to that used in the nanoindentation problem (see Fig. 5.4). In this case, the copper thin film is of dimensions  $176.64\text{\AA} \times 66.51\text{\AA}$ , with the MD region of size  $94.72\text{\AA} \times 33.27\text{\AA}$ , positioned a distance of  $64\text{\AA}$  and  $33.25\text{\AA}$  from the left and bottom faces of the copper film, respectively. A rigid triangular indenter of size  $28.16\text{\AA} \times 14.08\text{\AA}$ , with an included angle of  $90^\circ$  is used in the experiment. The indenter is assumed to be rigid, which means that there are no interactions between the atoms of the indenter.

The copper film is modeled with both atomistic and continuum approaches, while only an atomistic approach is used to model the indenter. The atomic and continuum regions are modeled using the MD and Hermite-cloud methods. The face centered cubic atoms of copper are initially placed in their respective lattice positions, along the closely-packed  $\langle 111 \rangle$  plane, and modeled using the Morse potential, mentioned in Eq. (5.8), with the potential parameters given in Table 5.1, and the forces in the atomistic region are obtained using Eq. (5.9). The calculations are carried out iteratively within each time step, using the multiscale algorithm detailed in Table 3.1, until convergence is attained.

The rigid indenter is initially placed at a distance of  $121.61\text{\AA}$  and  $5\text{\AA}$  from the left and top surfaces of the film respectively. A vertical distance of  $5\text{\AA}$  is provided between the film and indenter to avoid any attractive forces to be developed between the indenter and the film, before the start of the simulation. The displacements along the three edges of the film are constrained in both the  $x$  and  $y$  directions, and the top surface facing the indenter is set free. The indenter is incrementally moved towards the film from its initial position at a constant speed of  $100\text{m/s}$  until it reaches a depth of  $5\text{\AA}$ , after which scratching is performed along the  $x$  direction, for a distance of  $35\text{\AA}$  with a scratch velocity of  $100\text{m/s}$ . Plane strain conditions are assumed in the calculations.

### Adaptive node distribution scheme

To further demonstrate the capability of the multiscale algorithm, an adaptive node distribution scheme is used in this problem. This is done by translating the atomistic region by a distance of  $5.12\text{\AA}$ , in the  $x$  direction, every time the indenter exceeds a scratching length of  $4\text{\AA}$ . By doing so, the size of the atomistic region can be kept constant during the simulation, instead of progressively increasing its size as the scratch length increases. During each translation, atoms are added along the scratch direction and subtracted in the region behind the indenter. The nodes in the surrounding continuum region are also modified accordingly. While doing so, the displacements and velocities of the added/deleted atoms and nodes are transferred to one another using the same interpolation functions used in the coupling algorithm (see Section 3.3).

### 5.2.3 Numerical results

In this section, the results of nanoscratching experiments are presented. The simulation begins with the rigid indenter initially positioned at a distance of  $5\text{\AA}$  above the copper film, which is then gradually moved to a depth of  $5\text{\AA}$ , followed by scratching for a distance of  $35\text{\AA}$ . For the purpose of the present discussion, the duration from the start of the simulation till the indenter reaches a depth of indentation  $5\text{\AA}$  will be denoted as the indentation phase, and the duration from the start to the end of scratching will be denoted as the scratch phase. The start and end positions for scratching is chosen such that there is a sufficient distance between the indenter and the sides of the copper film, so as to avoid boundary effects. A scratch velocity of  $100\text{m/s}$ , though somewhat higher than typical experimental values, is used in the simulation to achieve a reasonable computational time.

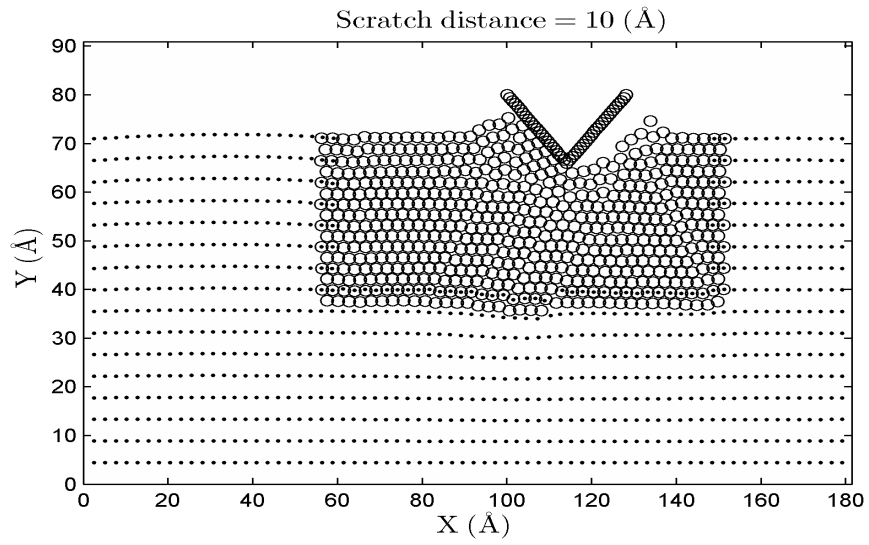
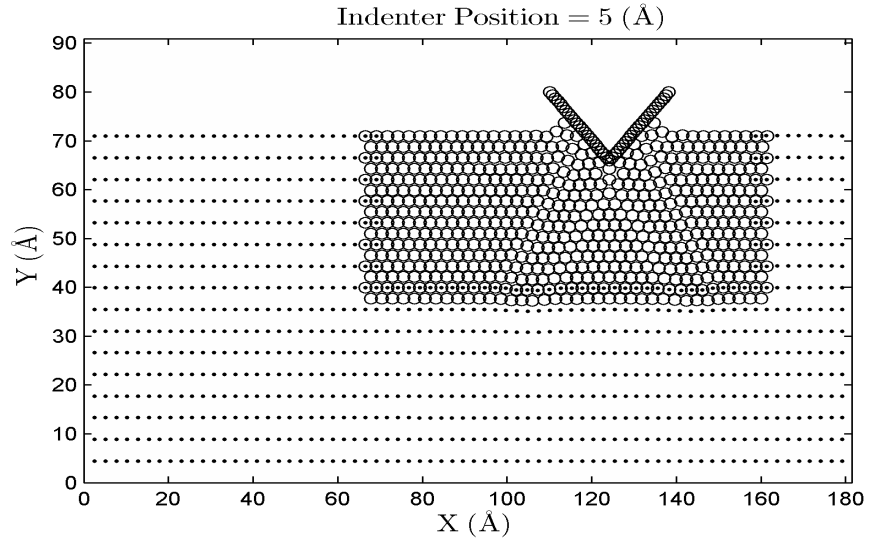
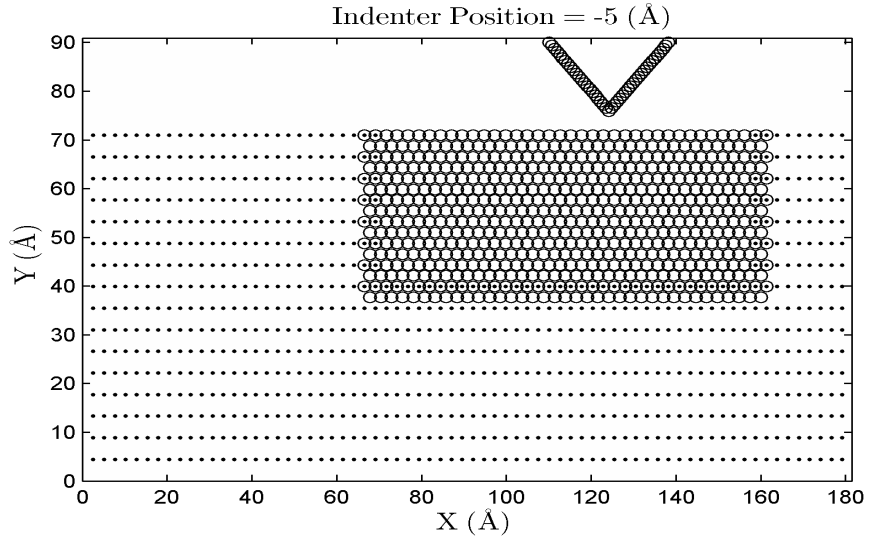
Figure 5.13 shows the snapshots of the simulation at different time instance. The

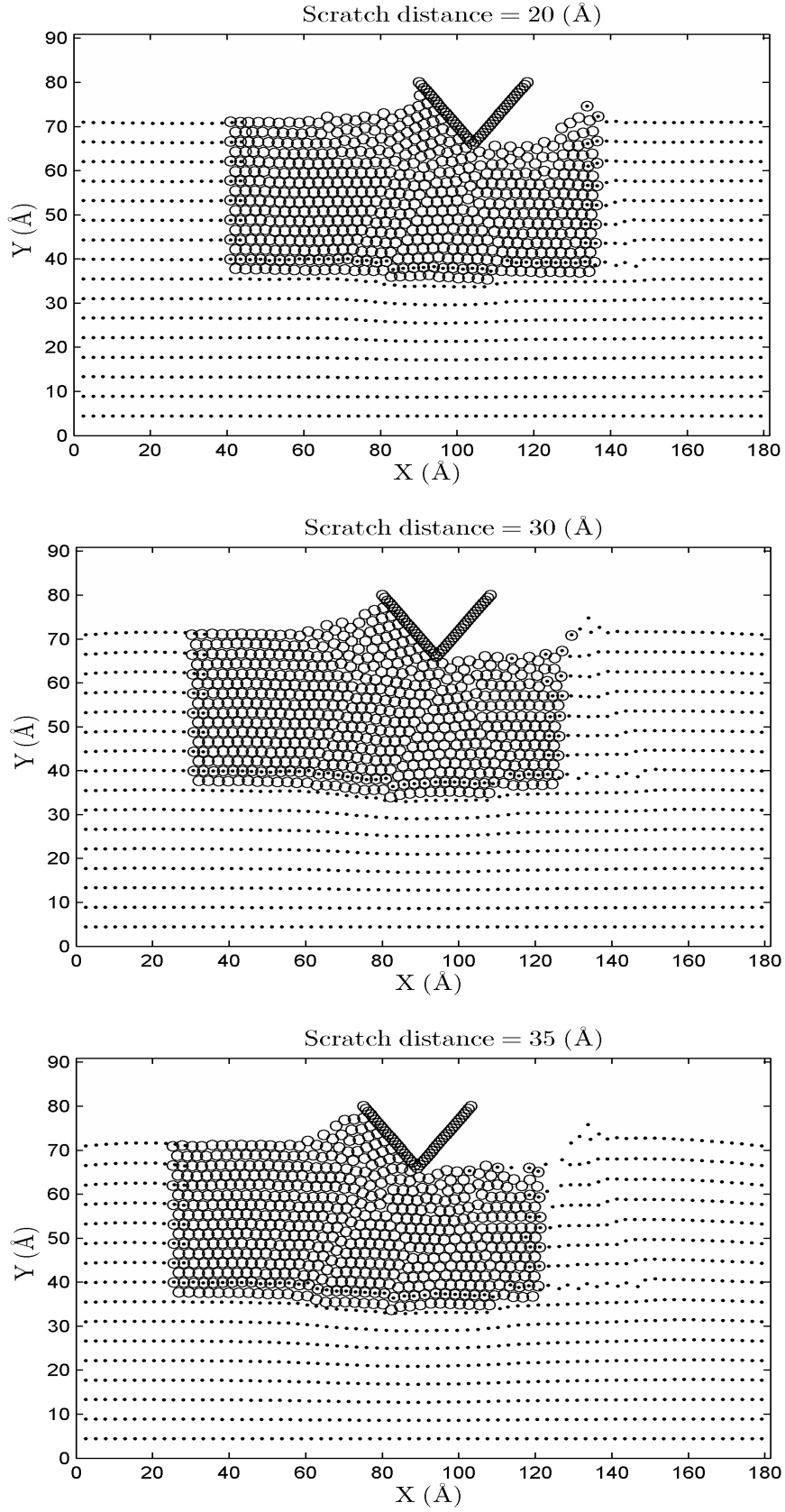
first two figures show the atomic and nodal position during the indentation phase, and the remaining figures show the same at different scratch lengths. The scratching action can be compared to that of ploughing, with atoms being piled-up before the indenter face. The pile-up is caused by the large negative rake angle, which is the angle between the indenter face and the direction of scratching (Komanduri et al., 2000b).

The forces measured in the scratching experiment are the normal force or the indentation force and then tangential or the scratch force. Figure 5.14 show the variation of normal and tangential forces during the indentation phase. From the figure, it can be observed that the normal force increases with indentation depth reaching a maximum of 10.46nN at 5Å, and its shape resembles the load-displacement curve obtained from nanoindentation simulation (see Fig. 5.6). The scratch force however remains close to zero during the indentation phase.

The variation of the normal and tangential forces during the scratching phase is shown in Fig. 5.15. The tangential force gradually builds up during the scratch phase before reaching a value of 8.37nN at a scratch length of 1.5Å, after which it fluctuates about a mean value of 7.17nN, over the remaining scratch length. During the scratch phase, the normal force initially drops from its peak value after which it exhibits fluctuations similar to the tangential force. However, the normal force is greater than the tangential force for most part of the scratching phase, with the average normal force (7.4337nN) being higher than the average value of tangential force (7.1773nN). The discontinuous force fluctuation observed during the scratching process, is caused by the dislocation nucleation and propagation, and has been observed in MD/multiscale simulations carried out by other researchers (Cao et al., 2010; Shiari et al., 2007).

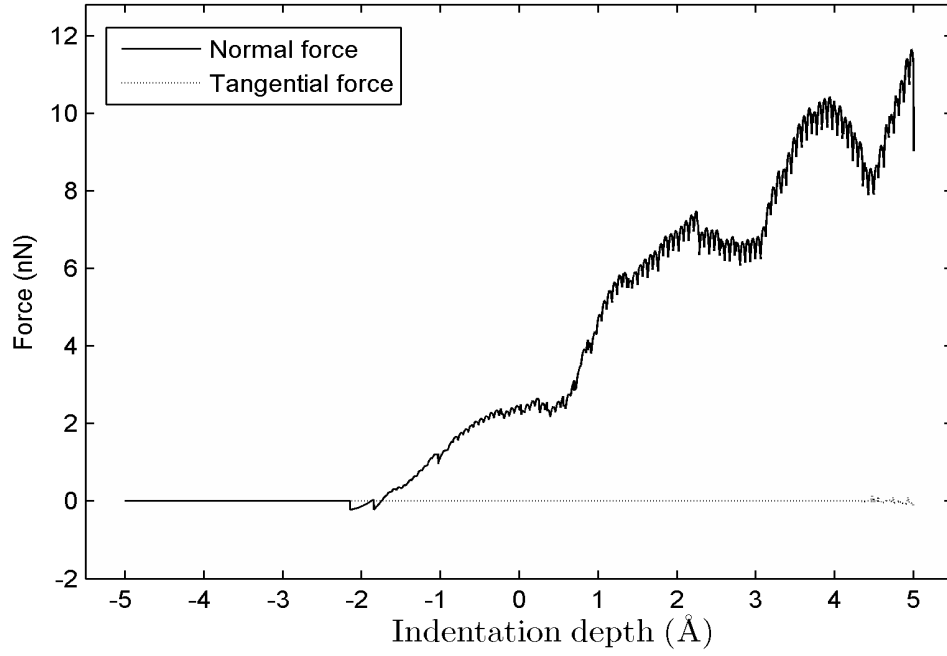
The tribological property of interest from the nanoscratching experiment is the coefficient of friction, which is obtained as the ratio of tangential force to the



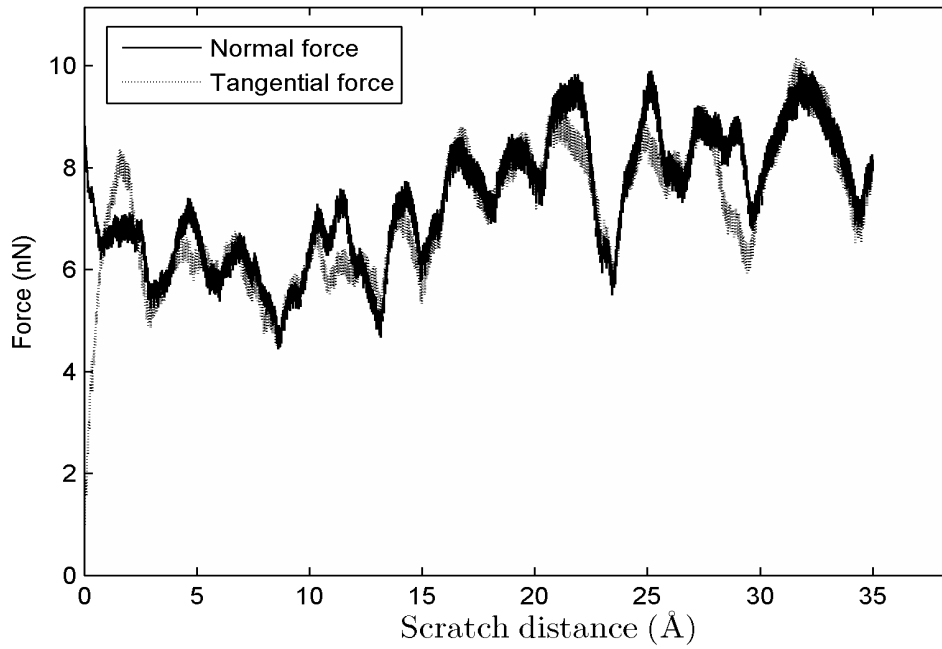


**Figure 5.13:** Snapshots of the nanoscratch simulation during the indentation and scratch phases.



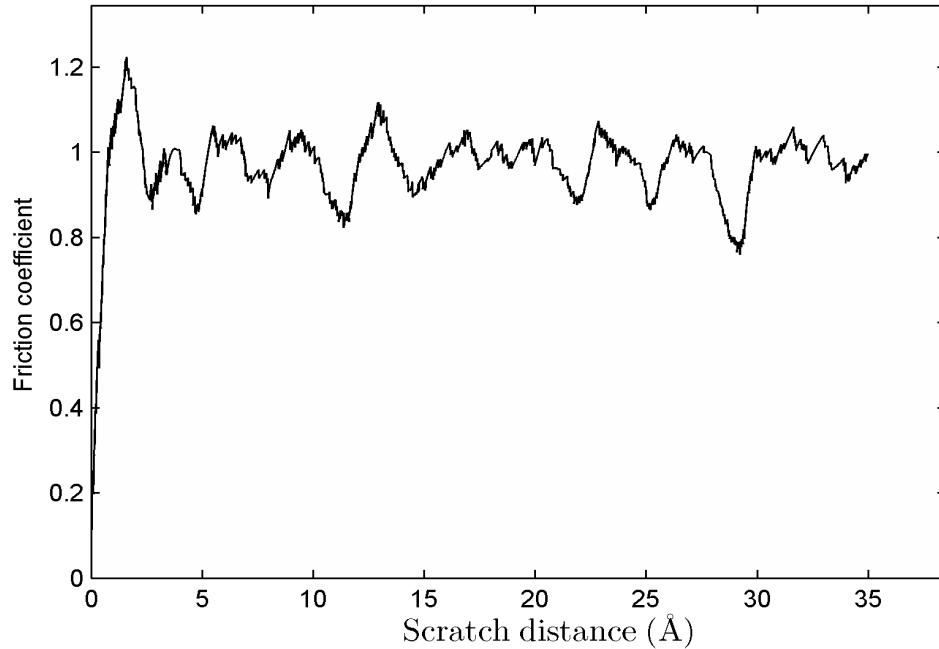


**Figure 5.14:** Variation of normal force during the indentation phase.



**Figure 5.15:** Variation of tangential force as a function of scratch length.

normal force during the scratching phase. Figure 5.16 shows the plot of the friction coefficient as a function of scratch length. The coefficient of friction obtained from the simulation is in the range of 0.76-1.22 with an average value of 0.974. In computing the average, only the values obtained after a scratch distance of  $1.5\text{\AA}$  are used, which is the distance after which the tangential force reaches a steady state. A direct comparison with experimental data is not possible as the simulation is performed in 2D. Moreover, the coefficient of friction obtained experimentally depends on a number of factors, such as load, environmental conditions, type of lubrication used, etc. However, the results of the present simulation are in reasonable agreement with values obtained from multiscale/MD simulation of nanoscratching experiments, reported in open literature (Mulliah and et al., 2004; Akabane et al., 2007; Cao et al., 2010; Noreyan and Amar, 2008; Shiari et al., 2007; Komanduri et al., 2000b).



**Figure 5.16:** Coefficient of friction as a function of scratch length.

### 5.3 Chapter Summary

In this chapter, 2D concurrent multiscale simulations were carried out to study nanoindentation and nanoscratching tests on a copper thin film. The multiscale method is able to accurately simulate nanoindentation, which is evident by comparing the load-displacement profile of the multiscale simulation with a full atomistic simulation and also with MD/multiscale simulation reported in literature. Seamless exchange of information between the atomic and continuum domains is indicated by the smooth displacement profile that does not exhibit any distinct discontinuities. By using the contact perimeter in 2D, the force per unit length values are extracted from the simulation and also compared for different included angles of the indenter. The force per unit length exhibits an increasing trend with an increase in indentation speed. The capability of the multiscale model was further validated by the use of an adaptive node distribution scheme for the nanoscratching problem, which maintains a constant size of the atomistic region. The forces and coefficient of friction obtained from the nanoscratching simulations also compare well with the values in open literature. In conclusion, the present multiscale model is able to provide an efficient and sufficiently accurate solutions to the nanoindentation and scratching problems, by restricting the atomic scale detail to a small region.

# Chapter 6

## Conclusions and Recommendations

The major contribution from this work is the development of a new multiscale algorithm to concurrently couple atomistic and continuum scales. The main challenge in developing the multiscale model is the development of a handshaking algorithm to ensure a seamless interface between the different length/time scales. In the following sections the major achievements and conclusions of this work are highlighted, and some possible avenues for future work in this direction are identified.

### 6.1 Major Theoretical Achievements

The main achievement from this work is the development of a novel multiscale technique based on the molecular dynamics and the strong-form meshless Hermite-cloud methods. The major achievements from this work are listed below:

1. The first major contribution is the development of multiscale algorithm to concurrently couple atomistic and continuum scales by ensuring displacement compatibility and force equilibrium in the overlapping transition or handshaking region. The multiscale method further uses a meshless method for continuum discretization, which addresses some of the limitations of other

continuum approaches such as the FEM. The highlight of the developed multiscale algorithm is its simplicity and elegance, which eliminates the need for complex mesh generation activity in the continuum domain.

2. The multiscale method uses a strong-form approach, instead of the more prevalent weak-form approaches, for the discretization of the governing PDE, making it a truly meshless multiscale scale method that does not require a background mesh for numerical integration. The source code implementation of the this multiscale algorithm is therefore more compact and abridged compared with those based on traditional finite elements.
3. The use of the meshless Hermite-cloud method for continuum discretization enables the construction of higher order interpolation polynomials in the overlapping transition region, and therefore compatibility of both the field variable and its first-order derivative is ensured in the transition region. Further, the use of interpolation polynomials permit more freedom in the nodal distribution in the continuum domain, and the continuum nodes need not be fully refined to coincide with the atomic locations in the transition region.
4. The proposed multiscale model was validated numerically by solving static and transient benchmark problems in one and two-dimensional domains. The static problems solved include the Poisson and Laplace equations with different boundary conditions and high gradients, and transient problems include wave propagation problems in both one and two dimensional spaces. The numerical accuracy of the multiscale model is measured by using both a global error measure, and also by comparing the results with a full atomistic simulation. The numerical results show that the proposed method is efficient and accurate, and also provides a seamless coupling between the two domains.

## 6.2 Tribology Studies via the Currently Developed Multiscale Method

The present multiscale model was employed to simulate solve two engineering problems of interest, namely nanoindentation and nanoscratching on a two-dimensional copper thin film.

The results from the nanoindentation simulation, namely the load-displacement graph obtained from the multiscale method shows only slight quantitative variation from that of the full atomistic model. More importantly, the graphs from both simulations show a similar trend thus validating the multiscale method. The displacement profile without discontinuities further supports the efficiency of the multiscale method in ensuring smooth exchange of information between the atomistic and continuum domains. The material properties extracted from the simulation include the force/unit length obtained by dividing the maximum load on the indenter by its contact perimeter. The capability of the multiscale model is also validated by the use of an adaptive node distribution scheme for solving the nanoscratch problem, wherein the size of the atomistic region is maintained constant. The forces and the coefficient of friction obtained from the simulation compare well with the values reported in literature.

## 6.3 Final Remarks

In conclusion, a multiscale model that provides accurate and efficient solutions to couple different length/time scales has been successfully developed, and substantially validated through benchmark problems and tribological problems of interest. In addition, this multiscale method is completely mesh free. By restricting the atomic scale detail to a small region, the method is able to save on computational resources,

achieving computational speed-up of more than one-order.

## 6.4 Recommendations

In this work, the developed multiscale model was used to solve two tribological problems of interest, namely nanoindentation and nanoscratching. This work can be definitely extended to include more complex problem, and a few possibilities are listed below:

- The method can be used to solve more challenging problems such as analysis of MEMS/NEMS systems, crack propagation in crystals, other tribological problems such as friction and wear, nanoscale metal cutting, material characterization, biological systems at nanoscale, etc.
- The multiscale model developed in this work is based on certain assumptions; namely, the empirical atomic potentials used in the atomistic domains are relatively simple potentials, temperature effects have not been included in the study, the continuum model is a linear elastic model, and the movement of dislocations from atomistic to continuum regions has not been considered. The main focus in this work was the development and validation of the multiscale model. In the future, some of the issues mentioned above can be advanced and/or addressed to achieve a more refined model.

## Publications Arising from this Thesis

- V. Pandurangan, H. Li and T. Y. Ng, A concurrent multiscale method based on the alternating schwarz scheme for coupling atomic and continuum scales with first-order compatibility, *Computational Mechanics*, 47(1): 1-16, 2011
- P. Venkataraman, T. Y. Ng and H. Li, Development of a novel multi-scale numerical technique, *Computational Materials Science* 49(1 SUPPL.): S131-S134, 2010
- V. Pandurangan, H. Li and T. Y. Ng, A novel multiscale method for atomic–continuum coupling with continuity of both field function and corresponding 1st-order differential. *Fifth international conference on Multiscale Materials Modeling (MMM 2010)*, Freiburg (Germany), 2010.
- T. Y. Ng, V. Pandurangan and H. Li, Multiscale modeling of nanoindentation in copper thin films via the concurrent coupling of the meshless Hermite-Cloud method with molecular dynamics, *Applied Surface Science*, doi:10.1016/j.apsusc.2011.07.059.



# References

- Metals Handbook Vol. 2: Properties and Selection: Nonferrous Alloys and Special-Purpose Materials*. American Society for Metals, 10th edition, 1990.
- ASM Handbook Vol.8, Mechanical Testing and Evaluation*. ASM International, Ohio, USA, 2000.
- T. Akabane, Y. Sasajima, and J. Onuki. Nanoscratching of metallic thin films on silicon substrate: A molecular dynamics study. *Journal of Electronic Materials*, 36(9):1174–1180, 2007.
- N. R. Aluru and G Li. Finite cloud method: a true meshless technique based on a fixed reproducing kernel approximation. *International Journal for Numerical Methods in Engineering*, 50(10):2373–2410, 2001.
- S. N. Atluri and T. Zhu. A new Meshless Local Petrov-Galerkin (MLPG) approach in computational mechanics. *Computational Mechanics*, 22(2):117–127, 1998.
- G. S. Ayton, W. G. Noid, and G. A. Voth. Multiscale modeling of biomolecular systems: in serial and in parallel. *Current Opinion in Structural Biology*, 17(2):192–198, 2007.
- S. Badia, P. Bochev, R. Lehoucq, M. L. Parks, J. Fish, M. A. Nuggehally, and M. Gunzburger. A force-based blending model for atomistic-to-continuum coupling. *International Journal for Multiscale Computational Engineering*, 5(5):387–406, 2007.

- S. Badia, M. L. Parks, P. Bochev, M. Gunzburger, and R. Lehoucq. On atomistic-to-continuum coupling by blending. *Multiscale Modeling and Simulation*, 7(1): 381–406, 2008.
- D. Beegan, S. Chowdhury, and M. T. Laugier. A nanoindentation study of copper films on oxidised silicon substrates. *Surface and Coatings Technology*, 176(1): 124–130, 2003.
- T. Belytschko, Y. Y. Lu, and L. Gu. Element-free galerkin methods. *International Journal for Numerical Methods in Engineering*, 37(2):229–256, 1994.
- T. Belytschko, Y. Krongauz, D. Organ, M. Fleming, and P. Krysl. Meshless methods: An overview and recent developments. *Computer Methods in Applied Mechanics and Engineering*, 139(1-4):3–47, 1996.
- P. Berczik and I. G. Kolesnik. Smoothed particle hydrodynamics and its application to astrophysical problems. *Kinematics and Physics of Celestial Bodies*, 9:1–11, 1993.
- J. Q. Broughton, F. F. Abraham, N. Bernstein, and E. Kaxiras. Concurrent coupling of length scales: Methodology and application. *Physical Review B*, 60(4):2391, 1999.
- Y. Cao, J. Zhang, Y. Liang, F. Yu, and T. Sun. Mechanical and tribological properties of Ni/Al multilayers - a molecular dynamics study. *Applied Surface Science*, 257(3):847–851, 2010.
- H. J. Chang, H. N. Han, and M. C. Fivel. Multiscale modelling of nanoindentation. *Key Engineering Materials*, 345-346 II:925–930, 2007.
- S. C. Chapra and R. P. Canale. *Numerical Methods for Engineers: With Software and Programming Applications*. Tata McGraw-Hill Publications, New Delhi, 2002.

- A. J. Chorin. Numerical study of slightly viscous flow. *Journal of Fluid Mechanics*, 57(04):785–796, 1973.
- E. Clementi and S. F. Reddaway. Global Scientific and Engineering Simulations on Scalar, Vector and Parallel LCAP-Type Supercomputers [and Discussion]. *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 326(1591):445–470, 1988.
- W. A. Curtin and E. Miller Ronald. Atomistic/continuum coupling in computational materials science. *Modelling and Simulation in Materials Science and Engineering*, 11(3):R33, 2003.
- J. Demongeot, J. Bezy-Wendling, J. Mattes, P. Haigron, N. Glade, and J. L. Coatrieux. Multiscale modeling and imaging: The challenges of biocomplexity. *Proceedings of the IEEE*, 91(10):1723–1737, 2003.
- M. F. Doerner and W. D. Nix. A method for interpreting the data from depth-sensing indentation instruments. *J. Mater. Res.*, 1(4):601–609, 1986.
- B. Eidel and A. Stukowski. A variational formulation of the quasicontinuum method based on energy sampling in clusters. *Journal of the Mechanics and Physics of Solids*, 57(1):87–108, 2009.
- A. C. Fischer-Cripps. Critical review of analysis and interpretation of nanoindentation test data. *Surface and Coatings Technology*, 200(14-15):4153–4165, 2006.
- J. Fish, M. A. Nuggehally, M. S. Shephard, C. R. Picu, S. Badia, M. L. Parks, and M. Gunzburger. Concurrent AtC coupling based on a blend of the continuum stress and the atomistic force. *Computer Methods in Applied Mechanics and Engineering*, 196(45-48):4548–4560, 2007.
- L. Gavete, M. L. Gavete, and J. J. Benito. Improvements of generalized finite difference method and comparison with other meshless method. *Applied Mathematical Modelling*, 27(10):831–847, 2003.

- R. A. Gingold and J. J. Monaghan. Smoothed particle hydrodynamics - theory and application to non-spherical stars. *Royal Astronomical Society*, 181:375–389, 1977.
- M. S. Gockenbach. *Partial Differential Equations: Analytical and Numerical methods*. Society for Industrial and Applied Mathematics, Philadelphia, USA, 2002.
- Y. T. Gu and L. C. Zhang. A concurrent multiscale method based on the meshfree method and molecular dynamics analysis. *Multiscale Modeling and Simulation*, 5(4):1128–1155, 2006.
- J. M. Haile. *Molecular Dynamics Simulation*. Wiley & Sons, Newyork, 1997.
- T. Inamura, N. Takezawa, and N. Taniguchi. Atomic-scale cutting in a computer using crystal models of copper and diamond. *CIRP Annals - Manufacturing Technology*, 41(1):121–124, 1992.
- S. Izumi, T. Kawakami, and S. Sakai. Study of a combined FEM-MD method for silicon. *JSME International Journal, Series A: Solid Mechanics and Material Engineering*, 44(1):152–159, 2001.
- Y. R. Jeng and C. M. Tan. Study of nanoindentation using FEM atomic model. *Journal of Tribology*, 126(4):767–774, 2004.
- P. S. Jensen. Finite difference techniques for variable grids. *Computers and Structures*, 2(1-2):17–29, 1972.
- T. E. Karakasidis and C. A. Charitidis. Multiscale modeling in nanomaterials science. *Materials Science and Engineering: C*, 27(5-8):1082–1089, 2007.
- J. Knap and M. Ortiz. An analysis of the quasicontinuum method. *Journal of the Mechanics and Physics of Solids*, 49(9):1899–1923, 2001.

- 
- S. Kohlhoff, P. Gumbsch, and H. F. Fischmeister. Crack propagation in b.c.c. crystals studied with a combined finite-element and atomistic model. *Philosophical Magazine A*, 64(4):851 – 878, 1991.
- R. Komanduri, N. Chandrasekaran, and L. M. Raff. MD simulation of indentation and scratching of single crystal aluminum. *Wear*, 240(1-2):113–143, 2000a.
- R. Komanduri, N. Chandrasekaran, and L. M. Raff. Molecular dynamics simulation of atomic-scale friction. *Physical Review B*, 61(20):14007, 2000b.
- K. Y. Lam, Hua Li, Y. K. Yew, and T. Y. Ng. Development of the meshless Hermite-Cloud method for structural mechanics applications. *International Journal of Mechanical Sciences*, 48(4):440–450, 2006.
- Andrew R. Leach. *Molecular Modelling Principles and Applications*. Pearson Education Limited, Singapore, 2001.
- J. D. Lee, X. Q. Wang, and Y. P. Chen. Multiscale material modeling and its application to a dynamic crack propagation problem. *Theoretical and Applied Fracture Mechanics*, 51(1):33–40, 2009.
- Y. Lee, J. Y. Park, S. Y. Kim, S. Jun, and S. Im. Atomistic simulations of incipient plasticity under Al(1 1 1) nanoindentation. *Mechanics of Materials*, 37(10):1035–1048, 2005.
- Hua Li, T. Y. Ng, J. Q. Cheng, and K. Y. Lam. Hermite cloud: a novel true meshless method. *Computational Mechanics*, 33(1):30–41, 2003.
- S. Li and W. K. Liu. Meshfree and particle methods and their applications. *Applied Mechanics Reviews*, 55(1):1–34, 2002.
- X. Li and B. Bhushan. A review of nanoindentation continuous stiffness measurement technique and its applications. *Materials Characterization*, 48(1):11–36, 2002.

- 
- E. Lidorikis, M. E. Bachlechner, R. K. Kalia, A. Nakano, P. Vashishta, and G. Z. Voyiadjis. Coupling length scales for multiscale atomistics-continuum simulations: Atomistically induced stress distributions in Si/Si<sub>3</sub>N<sub>4</sub> nanopixels. *Physical Review Letters*, 87(8):861041–861044, 2001.
- T. J. Liszka, C. A. M. Duarte, and W. W. Tworzydło. hp-Meshless cloud method. *Computer Methods in Applied Mechanics and Engineering*, 139(1-4):263–288, 1996.
- G. R. Liu and Y. T. Gu. A local point interpolation method for stress analysis of two-dimensional solids. *Structural Engineering and Mechanics*, 11(2):221–236, 2001a.
- G. R. Liu and Y. T. Gu. A point interpolation method for two-dimensional solids. *International Journal for Numerical Methods in Engineering*, 50(4):937–951, 2001b.
- G. R. Liu and Y. T. Gu. A meshfree method: Meshfree weak-strong (MWS) form method, for 2-D solids. *Computational Mechanics*, 33(1):2–14, 2003.
- G. R. Liu, B. B. T. Kee, and L. Chun. A stabilized least-squares radial point collocation method (LS-RPCM) for adaptive analysis. *Computer Methods in Applied Mechanics and Engineering*, 195(37-40):4843–4861, 2006a.
- P. Liu, Y. W. Zhang, and C. Lu. A three-dimensional concurrent atomistic/continuum analysis of an epitaxially strained island. *Journal of Applied Physics*, 94(10):6350–6353, 2003.
- W. K. Liu and S. Jun. Multiple-scale reproducing kernel particle methods for large deformation problems. *International Journal for Numerical Methods in Engineering*, 41(7):1339–1362, 1998.
- W. K. Liu, S. Jun, S. Li, J. Adee, and T. Belytschko. Reproducing kernel particle

- 
- methods for structural dynamics. *International Journal for Numerical Methods in Engineering*, 38(10):1655–1679, 1995a.
- W. K. Liu, S. Jun, and Y.F. Zhang. Reproducing kernel particle methods. *International Journal for Numerical Methods in Fluids*, 20(8-9):1081–1106, 1995b.
- W. K. Liu, Y. Chen, C. T. Chang, and T. Belytschko. Advances in multiple scale kernel particle methods. *Computational Mechanics*, 18(2):73–111, 1996a.
- W. K. Liu, Y. Chen, S. Jun, J. S. Chen, T. Belytschko, C. Pan, R. A. Uras, and C. T. Chang. Overview and applications of the reproducing kernel particle methods. *Archives of Computational Methods in Engineering*, 3(1):3–80, 1996b.
- W. K. Liu, E. G. Karpov, S. Zhang, and H. S. Park. An introduction to computational nanomechanics and materials. *Computer Methods in Applied Mechanics and Engineering*, 193(17-20):1529–1578, 2004.
- W. K. Liu, H. S. Park, D. Qian, E. G. Karpov, H. Kadowaki, and G. J. Wagner. Bridging scale methods for nanomechanics and materials. *Computer Methods in Applied Mechanics and Engineering*, 195(13-16):1407–1421, 2006b.
- W. K. Liu, S. Jun, and D. Qian. Computational nanomechanics of materials. *Journal of Computational and Theoretical Nanoscience*, 5(5):970–996, 2008.
- L.B. Lucy. A numerical approach to the testing of fission hypothesis. *The Astron J*, 8(12):1013–1024, 1977.
- K. W. McElhaney, J. J. Vlassak, and W. D. Nix. Determination of indenter tip geometry and indentation contact area for depth-sensing indentation experiments. *Journal of Materials Research*, 13(5):1300–1306, 1998.
- R. Miller, M. Ortiz, R. Phillips, V. Shenoy, and E. B. Tadmor. Quasicontinuum models of fracture and plasticity. *Engineering Fracture Mechanics*, 61(3-4):427–444, 1998a.

- R. Miller, E. B. Tadmor, R. Phillips, and M. Ortiz. Quasicontinuum simulation of fracture at the atomic scale. *Modelling and Simulation in Materials Science and Engineering*, 6(5):607–638, 1998b.
- R. E. Miller and E. B. Tadmor. The quasicontinuum method: Overview, applications and current directions. *Journal of Computer-Aided Materials Design*, 9(3):203–239, 2002.
- R. E. Miller and E.B Tadmor. A unified framework and performance benchmark of fourteen multiscale atomistic/continuum coupling methods. *Modelling and Simulation in Materials Science and Engineering*, 17(5):053001, 2009.
- J. J. Monaghan. Why particle methods work. *SIAM Journal on Scientific and Statistical Computing*, 3(4):422–433, 1982.
- J. J. Monaghan. An introduction to SPH. *Computer Physics Communications*, 48(1):89–96, 1988.
- J. J. Monaghan. SPH without a tensile instability. *Journal of Computational Physics*, 159(2):290–311, 2000.
- D. Mulliah and et al. Molecular dynamic simulations of nanoscratching of silver (100). *Nanotechnology*, 15(3):243, 2004.
- B. Nayroles, G. Touzot, and P. Villon. Generalizing the finite element method: Diffuse approximation and diffuse elements. *Computational Mechanics*, 10(5):307–318, 1992.
- V. P. Nguyen, T. Rabczuk, S. Bordas, and M. Duflot. Meshless methods: A review and computer implementation aspects. *Mathematics and Computers in Simulation*, 79(3):763–813, 2008.
- A. Noreyan and J. G. Amar. Molecular dynamics simulations of nanoscratching of 3C SiC. *Wear*, 265(7-8):956–962, 2008.



- W. C. Oliver and G. M. Pharr. Improved technique for determining hardness and elastic modulus using load and displacement sensing indentation experiments. *Journal of Materials Research*, 7(6):1564–1580, 1992.
- W. C. Oliver and G. M. Pharr. Measurement of hardness and elastic modulus by instrumented indentation: Advances in understanding and refinements to methodology. *Journal of Materials Research*, 19(1):3–20, 2004.
- E. Onate, S. Idelsohn, O. C. Zienkiewicz, and R. L. Taylor. A finite point method in computational mechanics. applications to convective transport and fluid flow. *International Journal for Numerical Methods in Engineering*, 39(22):3839–3866, 1996.
- H. S. Park and W. K. Liu. An introduction and tutorial on multiple-scale analysis in solids. *Computer Methods in Applied Mechanics and Engineering*, 193(17-20):1733–1772, 2004.
- H. S. Park, E. G. Karpov, P. A. Klein, and W. K. Liu. Three-dimensional bridging scale analysis of dynamic fracture. *Journal of Computational Physics*, 207(2):588–609, 2005a.
- H. S. Park, E. G. Karpov, W. K. Liu, and P. A. Klein. The bridging scale for two-dimensional atomistic/continuum coupling. *Philosophical Magazine*, 85(1), 2005b.
- M. L. Parks, P. B. Bochev, and R. B. Lehoucq. Connecting atomistic-to-continuum coupling and domain decomposition. *Multiscale Modeling and Simulation*, 7(1):362–380, 2008.
- P. Peng, G. Liao, T. Shi, Z. Tang, and Y. Gao. Molecular dynamic simulations of nanoindentation in aluminum thin film on silicon substrate. *Applied Surface Science*, 256(21):6284–6290, 2010.

- N. Perrone and R. Kao. A general finite difference method for arbitrary meshes. *Computers & Structures*, 5(1):45–57, 1975.
- G. M. Pharr. Measurement of mechanical properties by ultra-low load indentation. *Materials Science and Engineering A*, 253(1-2):151–159, 1998.
- R. C. Picu. Atomistic-continuum simulation of nano-indentation in molybdenum. *Journal of Computer-Aided Materials Design*, 7(2):77–87, 2000.
- D. Qian, G. J. Wagner, and W. K. Liu. A multiscale projection method for the analysis of carbon nanotubes. *Computer Methods in Applied Mechanics and Engineering*, 193(17-20):1603–1632, 2004.
- Alfio Quarteroni and Alberto Valli. *Domain Decomposition Methods for Partial Differential Equations*. Oxford University Press, Newyork, 2005.
- H. Rafii-Tabar, L. Hua, and M. Cross. Multiscale numerical modelling of crack propagation in two-dimensional metal plate. *Materials Science and Technology*, 14(6):544–548, 1998.
- P. W. Randles and L. D. Libersky. Smoothed particle hydrodynamics: Some recent improvements and applications. *Computer Methods in Applied Mechanics and Engineering*, 139(1-4):375–408, 1996.
- D. C. Rapaport. *The Art of Molecular Dynamics Simulation elementary methods*. Cambridge University Press, Cambridge, UK., 2004.
- J. N. Reddy. *An Introduction to the finite element method*. McGraw Hill International Edition, Singapore, 1993.
- R. E. Rudd and J. Q. Broughton. Concurrent coupling of length scales in solid state systems. *physica status solidi (b)*, 217(1):251–291, 2000.
- R. D. Russell and L. F. Shampine. A collocation method for boundary value problems. *Numerische Mathematik*, 19(1):1–28, 1972.

- R. Saha and W. D. Nix. Effects of the substrate on the determination of thin film mechanical properties by nanoindentation. *Acta Materialia*, 50(1):23–38, 2002.
- S. Shen and S. N. Atluri. Multiscale simulation based on the meshless local petrov-galerkin (MLPG) method. *CMES - Computer Modeling in Engineering and Sciences*, 5(3):235–255, 2004.
- V. B. Shenoy, R. Miller, E. B. Tadmor, R. Phillips, and M. Ortiz. Quasicontinuum models of interfacial structure and deformation. *Physical Review Letters*, 80(4):742–745, 1998.
- V. B. Shenoy, R. Miller, E. B. Tadmor, D. Rodney, R. Phillips, and M. Ortiz. An adaptive finite element approach to atomic-scale mechanics—the quasicontinuum method. *Journal of the Mechanics and Physics of Solids*, 47(3):611–642, 1999.
- V. B. Shenoy, R. Phillips, and E. B. Tadmor. Nucleation of dislocations beneath a plane strain indenter. *Journal of the Mechanics and Physics of Solids*, 48(4):649–673, 2000.
- B. Shiari, R. E. Miller, and D. D. Klug. Multiscale simulation of material removal processes at the nanoscale. *Journal of the Mechanics and Physics of Solids*, 55(11):2384–2405, 2007.
- B. Shiari, R. E. Miller, and D. D. Klug. Multiscale modeling of solids at the nanoscale: Dynamic approach. *Canadian Journal of Physics*, 86(2):391–400, 2008.
- L. E. Shilkrot, W. A. Curtin, and R. E. Miller. A coupled atomistic/continuum model of defects in solids. *Journal of the Mechanics and Physics of Solids*, 50(10):2085–2106, 2002a.
- L. E. Shilkrot, R. E. Miller, and W. A. Curtin. Coupled atomistic and discrete dislocation plasticity. *Physical Review Letters*, 89(2):255011–255014, 2002b.

- 
- L. E. Shilkrot, R. E. Miller, and W. A. Curtin. Multiscale plasticity modeling: Coupled atomistics and discrete dislocation mechanics. *Journal of the Mechanics and Physics of Solids*, 52(4):755–787, 2004.
- J. A. Smirnova, L. V. Zhigilei, and B. J. Garrison. A combined molecular dynamics and finite element method technique applied to laser induced pressure wave propagation. *Computer Physics Communications*, 118(1):11–16, 1999.
- G. S. Smith, E. B. Tadmor, and E. Kaxiras. Multiscale Simulation of Loading and Electrical Resistance in Silicon Nanoindentation. *Physical Review Letters*, 84(6):1260–1263, 2000.
- I. N. Sneddon. The relation between load and penetration in the axisymmetric boussinesq problem for a punch of arbitrary profile. *International Journal of Engineering Science*, 3(1):47–57, 1965.
- V. Springel. Smoothed particle hydrodynamics in astrophysics. *Annual Review of Astronomy and Astrophysics*, 48:391–430, 2010.
- X. Sun, S. Chen, K. Cheng, D. Huo, and W. Chu. Multiscale simulation on nanometric cutting of single crystal copper. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 220(7):1217–1222, 2006.
- S. Suresh, T. G. Nieh, and B. W. Choi. Nano-indentation of copper thin films on silicon substrates. *Scripta Materialia*, 41(9):951–957, 1999.
- E. B. Tadmor, M. Ortiz, and R. Phillips. Quasicontinuum analysis of defects in solids. *Philosophical Magazine A: Physics of Condensed Matter, Structure, Defects and Mechanical Properties*, 73(6):1529–1563, 1996.
- E. B. Tadmor, R. Miller, R. Phillips, and M. Ortiz. Nanoindentation and incipient plasticity. *Journal of Materials Research*, 14(6):2233–2250, 1999.

- S. Tang. A finite difference approach with velocity interfacial conditions for multiscale computations of crystalline solids. *Journal of Computational Physics*, 227(8):4038–4062, 2008.
- S. Tang, T. Y. Hou, and W. K. Liu. A mathematical framework of the bridging scale method. *International Journal for Numerical Methods in Engineering*, 65(10):1688–1713, 2006a.
- S. Tang, T. Y. Hou, and W. K. Liu. A pseudo-spectral multiscale method: Interfacial conditions and coarse grid equations. *Journal of Computational Physics*, 213(1): 57–85, 2006b.
- E. Van Der Giessen and A. Needleman. Discrete dislocation plasticity: A simple planar model. *Modelling and Simulation in Materials Science and Engineering*, 3(5):689–735, 1995.
- M. R. VanLandingham. Review of instrumented indentation. *Journal of Research of the National Institute of Standards and Technology*, 108(4):249–265, 2003.
- D. D. Vvedensky. Multiscale modelling of nanostructures. *Journal of Physics: Condensed Matter*, 16(50):R1537, 2004.
- G. J. Wagner and W. K. Liu. Coupling of atomistic and continuum simulations using a bridging scale decomposition. *Journal of Computational Physics*, 190(1): 249–274, 2003.
- C. T. Wang, S. R. Jian, J. S. C. Jang, Y. S. Lai, and P. F. Yang. Multiscale simulation of nanoindentation on Ni (1 0 0) thin film. *Applied Surface Science*, 255(5 PART 2):3240–3250, 2008.
- Q. X. Wang, T. Y. Ng, Hua Li, and K. Y. Lam. Multiscale Simulation of Coupled Length-Scales via Meshless Method and Molecular Dynamics. *Mechanics of Advanced Materials and Structures*, 16(1), 2009.

- S. Xiao and W. Yang. A temperature-related homogenization technique and its implementation in the meshfree particle method for nanoscale simulations. *International Journal for Numerical Methods in Engineering*, 69(10):2099–2125, 2007.
- S. Xiao, D. R. Andersen, R. P. Han, and W. Hou. Studies of carbon nanotube-based oscillators using molecular dynamics. *Journal of Computational and Theoretical Nanoscience*, 3(1):142–147, 2006.
- S. P. Xiao and T. Belytschko. A bridging domain method for coupling continua with molecular dynamics. *Computer Methods in Applied Mechanics and Engineering*, 193(17-20):1645–1669, 2004.
- X. Zhang, X. H. Liu, K. Z. Song, and M. W. Lu. Least-squares collocation meshless method. *International Journal for Numerical Methods in Engineering*, 51(9):1089–1100, 2001.
- C. Zhu, W. Guo, T. X. Yu, and C. H. Woo. Radial compression of carbon nanotubes: Deformation and damage, super-elasticity and super-hardness. *Nanotechnology*, 16(8):1035–1039, 2005.
- Z. Zong and K. Y. Lam. A localized differential quadrature (LDQ) method and its application to the 2d wave equation. *Computational Mechanics*, 29(4-5):382–391, 2002.