REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
21-07-2006	Final Performance Report	05-01-2003 - 04-30-2006

4. TITLE AND SUBTITLE
Multiscale Modeling and Simulation of Material Processing

5a. CONTRACT NUMBER

5b. GRANT NUMBER
F46920-03-1-0281

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Komanduri, Ranga
Lu, Hongbing
Raff, Lionel
Roy, Samit

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Oklahoma State University
School of Mechanical and Aerospace Engineering
218 Engineering North, Stillwater, Oklahoma, 74078

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Capt. Brett Conner, Ph.D., Program Manager / NA
Metallic Materials Program
Air Force Office of Scientific Research
4015 Wilson Blvd., Room 713
Arlington, VA 22203-1954

10. SPONSOR/MONITOR'S ACRONYM(S)
USAF, AFRL, AFOSR

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

AFRL-SR-AR-TR-07-0307

13. SUPPLEMENTARY NOTES

14. ABSTRACT
This project is on the development of scaling laws for multiscale simulations from atomistic to continuum using material testing techniques, such as tension and indentation. The main objective is to address critical issues involved in modeling. Specific problems addressed in this project are: (1) Multiscale simulation from atomistic to continuum via mesoplasticity by coupling molecular dynamics with the material point method in microtensile testing and nanoindentation. (2) Material behavior at nano- and microlevels using an in situ microtensile stage on an atomic force microscope and a nanoindentation system, (3) Multiscale simulation using the generalized interpolation material point method and parallel processing; (4) Combined finite element method and nanoindentation tests to determine material properties of single crystal copper in different orientations, incorporating mesoplastic constitutive laws, (5) Development of accurate and robust potentials from ab initio calculations using Gaussian 03 software, novelty sampling, and feed-forward neural networks (NN); and (6) Monte Carlo-Steepest Descent simulations of nanometric cutting.

15. SUBJECT TERMS
Scaling laws, multiscale modeling, nanoindentation, material point method (MPM), molecular dynamics (MD) simulations, Neural networks (NN), mesoplasticity, multilevel refinement, ab initio potentials, Monte Carlo (MC) Simulation

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Ranga Komanduri
U	U	U	UU	223	19b. TELEPHONE NUMBER *(Include area code)* (405) 744-5900

FINAL PERFORMANCE REPORT

(F46920-03-1-0281)

MULTISCALE MODELING AND SIMULATION OF MATERIAL PROCESSING

Ranga Komanduri [1,*], Hongbing Lu[1], Samit Roy[3], and Lionel M. Raff[2]
[1]Mechanical & Aerospace Engineering
[2]Chemistry Department
[3]now at University of Missouri, Rolla
Oklahoma State University
Stillwater, OK 74078

ABSTRACT

This project deals with the development of scaling laws for multiscale simulations from atomistic to continuum, via mesoplasticity and applied for such material testing techniques as tension and indentation. The main objective is to address some of the critical issues involved in multiscale material modeling - theory and simulation. The following specific problems were addressed in this project: (1) multiscale simulation from atomistic to continuum (or vice versa) via mesoplasticity by coupling molecular dynamics (MD) with the material point method (MPM) in microtensile testing and nanoindentation. The simulations are animated so that the process under consideration can be observed *in situ* under dynamic conditions; (2) material behavior at nano- and microlevels using an *in situ* microtensile stage on an atomic force microscope (AFM) and a nanoindentation system, (3) multiscale simulation using the generalized interpolation material point (GIMP) method and parallel processing; (4) combined finite element method (FEM) and nanoindentation tests to determine the material properties of single crystal copper in different orientations, incorporating mesoplastic constitutive laws, (5) development of accurate and robust potentials from *ab initio* calculations using a commercial (Gaussian 03) software, novelty sampling, and feed-forward neural networks (NN); and (6) Monte Carlo-Steepest Descent simulations of nanometric cutting.

Keywords: Scaling laws, multiscale modeling, nanoindentation, material point method (MPM), molecular dynamics (MD) simulations, neural networks (NN), mesoplasticity, multilevel refinement, *ab initio* potentials, Monte Carlo (MC) Simulation

* voice: (405)744-5900; Fax: (405) 744-7873; e-mail: ranga@ceat.okstate.edu (Komanduri)

20070904009

1. INTRODUCTION

Despite significant developments in materials simulation techniques, the goal of reliably predicting the properties of new materials in advance of material characterization and fabrication has not yet been achieved. It is also not possible to predict the properties of the material at nanoscale knowing the properties of the material at macrolevel or vice versa. This situation exists for several reasons that include a lack of reliable potentials to describe the behavior of the material, computational limitations, inadequate modeling of the process, absence of scaling laws, and difficulties associated with the experimental measurement of properties even at the microscale, let alone at nanoscale.

Scaling laws governing the mechanical behavior of materials from atomistic (nano), via mesoplastic (micro), to continuum (macro) are very important to numerous DoD applications, such as the development of a new class of aircraft engine materials, or new steels for naval battle ships, or new tank armor materials for the army, or numerous microelectromechanical components for a myriad of applications, for the following three reasons: (1) information on the mechanical behavior of materials at nanolevel is not presently available as input to nanotechnology for the manufacturing of nanocomponents or microelectro-mechanical systems (MEMS). For example, nanostructures may possess unique properties in view of their very high surface to volume ratio, or the nanostructures might be relatively free of defects with strength approaching the theoretical values, (2) applications where two length scales of different orders of magnitude are involved. For example, one is atomistic (nano) and the other mesoplastic (micro) as in nanoindentation, or, one is mesoplastic (micro) and the other continuum (macro) as in conventional indentation, and (3) it may be possible to extend the knowledge accumulated over time on material behavior at the macro (or continuum) level to the atomistic (or nano) level, via mesoplastic (micro) level. In this investigation, multiscale modeling and simulations of material properties are addressed considering tensile testing and indentation as two examples. These two applications are chosen because of the extensive use of these techniques for determining a wide range of material properties, including modulus, yield strength, ultimate strength, strain at fracture, and hardness. It may be noted that the scaling laws, as well as modeling and simulation tools to be developed in this investigation, will be generic and can be applied to other material processing techniques and/or material transformation mechanisms involved in these processes.

Kalia *et al.* [1] outlined the following benefits of nanophase materials: (1) Unique physical properties resulting from small grain size and numerous grains; (2) Unique properties strongly influenced by the presence of a large number of atoms at the interphase or grain boundaries; (3) Ability to sinter at significantly lower temperatures than conventional coarse-grained solids leading to considerable savings in the synthesis of high temperature materials; (4) Remarkable improvements in mechanical properties, such as hardness, strength, plastic deformation as the materials are relatively free of defects; (5) Enhanced ductility, especially of some inherently brittle ceramics, due to grain-boundary sliding. Small grains also facilitate efficient impurity doping of these materials; and (6) Unique chemical properties due to high surface to volume ratio as well as large porosity that increases the adsorption area and makes them better catalysts.

Connecting atomistic mechanisms of deformation and fracture to macroscopic behavior requires an appropriate formalism that bridges the length scales of these processes over several orders of magnitude. According to Hartley [2], the essence of multiscale modeling of the properties of materials is the ability to develop a description of material behavior on a macroscopic scale from models based on atomic scale mechanisms or vice versa, applying to deformation of crystalline materials, this takes the form of translating the behavior of individual dislocations or finite groups into a description of the global response of a mass of material to externally applied loads and displacements. Any such model is subjected to restrictions imposed by the geometry of the crystal lattice, active slip systems, the orientations and shapes of grains within the material, and the laws of thermodynamics.

It is well known that many macroscopic phenomena have their origins in the microscopic or even atomistic phenomena. For example, the coalescence of microcracks leading to gross fracture has microscopic origin. Similarly, the tip of a microcrack has its origin at the atomistic level. Another example where a macro phenomenon has microscopic origin, which in turn has atomistic origin, is metal cutting. The gross deformation in the primary shear zone on a macroscale is due to severe plastic deformation (shear strain ~2–6) of a polycrystalline material across the grain boundaries. But at the microlevel, the discontinuous nature of the chip formation leads to shear lamella (a few micrometers wide) followed by a shear front (a fraction of a micrometer wide). At the atomistic level, the deformation ahead of the tool depends on the structure and orientation of the single crystal workmaterial relative to the tool motion. The main objective of this proposal is to develop a fundamental understanding of indentation and tension

at various scales and develop a computer simulation code that would bridge the gap from atomistic to continuum, via. mesoplastic or microscopic behavior. The outcome of this work will provide the required inputs to CAD so that relevant output parameters can be used by the engineer in the design of components at any level of detail, namely, large macro components, or small microcomponents, or even extremely small nanocomponents.

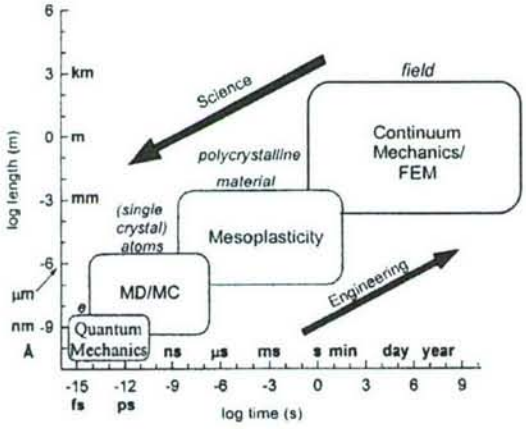


Figure 1 Schematic of simulations at various temporal and spatial levels

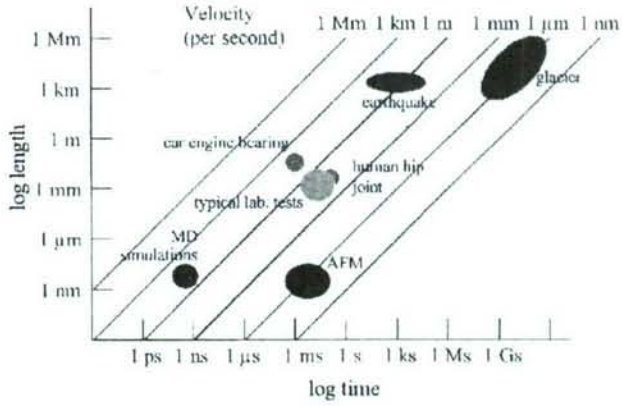


Figure 2 Schematic of various spatial and temporal scales in tribology

Figure 1 is a schematic of the various computer simulation methods used in the development of scaling laws as a function of spatial and temporal variables. At the subatomic level, we use quantum mechanics or *ab initio* calculations to develop the potential-energy functions. With increase in length (or time) scale, we model material behavior using molecular dynamics (MD) and Monte Carlo (MC) simulations, then to micro or mesoplastic, and finally to continuum mechanics using finite element methods (FEM). Figure 2 is a schematic showing various spatial and temporal scales for a typical tribological phenomenon for a wide range of applications. It may be noted at the outset that any scaling law developed would involve both temporal and spatial scales. But, generally, as the length scale increases, so does the time scale. The challenge is how to develop methods that permit simulation of a process with a fewer number of atoms (for e.g. 10^6 instead of 10^{14} atoms in a cube) or shorter time scale (for e.g., microseconds instead of seconds) but at the same time represent the process at the meso or macro scale. Alternatively stated, the problem is to develop methods that link a process at atomistic level to its neighborhood at the microlevel, and from there to the continuum level.

What are the rules linking these three levels? An attempt is made in this project to address these and other scientific and technological issues.

2. OBJECTIVES

The objective of this investigation is to contribute scientific and technological issues towards the development of scaling laws from atomistic to continuum using simulations based on such material testing methods as microtension and nanoindentation. Other objectives, in addition to the technical objective, include contribution towards the development of infrastructure and training of research personnel (or human resources) in disciplines critical to the DoD mission.

Technical Objectives:

(1) Multiscale simulation from atomistic to continuum (or vice versa) via mesoplasticity by coupling molecular dynamics (MD) with the material point method (MPM) in microtensile testing and nanoindentation. The simulations are animated so that the process under consideration can be observed *in situ* under dynamic conditions;

(2) Material behavior at nano- and microlevels using an *in situ* microtensile stage on an atomic force microscope (AFM) and a nanoindentation system,

(3) Multiscale simulation using the generalized interpolation material point (GIMP) method and parallel processing;

(4) Combined finite element method (FEM) and nanoindentation tests to determine the material properties of single crystal copper in different orientations, incorporating mesoplastic constitutive laws;

(5) Development of accurate and robust potentials from *ab initio* calculations using a commercial (Gaussian 03) software, novelty sampling, and feed-forward neural networks (NN); and

(6) Monte Carlo-Steepest Descent simulations of nanometric cutting.

Infrastructure Development:

To develop computational facilities involving massive parallel processing capable of modeling and simulation of materials properties and design as well as state-of-the art experimental facilities so as to advance OSU's research infrastructure development goals. The Advanced Computer Simulation Laboratory as well as the Advanced Materials and Processing Laboratory will be a showcase for OSU in particular and the State of Oklahoma, in general.

Training of Research Personnel:

To contribute towards the education of future scientists and engineers in disciplines critical to the DoD mission. Both graduate and undergraduate students will be trained to address the proposed research that is relevant to the defense mission with chemistry, physics, mechanical engineering, and materials engineering perspectives.

In the following, brief summaries of various problems involved in multiscale modeling from atomistic to continuum addressed in this project will be outlined. They were covered in 12 technical papers published in archival journals and included in this report for completeness.

3. ACCOMPLISHMENTS

3.1. Multiscale Simulation from Atomistic to Continuum – Coupling Molecular Dynamics (MD) with Material Point Method (MPM) (Lu *et al.* [3])

In this investigation, we introduced a new multiscale simulation approach that coupled atomistic-scale simulations using molecular dynamics (MD) with continuum-scale simulations using the material point method (MPM) originally developed by Sulsky, Zhou, and Shreyer [4] in 1995. In MPM, material continuum is represented by a finite collection of material points carrying all relevant physical characteristics, such as mass, acceleration, velocity, strain and stress. The use of material points at the continuum level provides a natural connection with the atoms in the lattice at the atomistic scale. A hierarchical mesh refinement technique in MPM is developed to scale down the continuum level to the atomistic level, so that material points at the fine level in MPM are allowed to directly couple with the atoms in MD. A one-to-one correspondence of MD atoms and MPM points is used in the transition region and non-local elastic theory is used to assure compatibility between MD and MPM regions, so that seamless coupling between MD and MPM can be accomplished.

A single crystal silicon, subjected to uniaxial tension, is used to demonstrate the viability of the technique. A Tersoff-type, three-body potential was used in the MD simulations. The coupled MD/MPM simulations showed that silicon under nanometric tension, experiences with increasing elongation in elasticity, dislocation generation and plasticity by slip, void formation and propagation, formation of amorphous structure, necking, and final rupture. Results are presented in terms of stress–strain relationships at several strain rates, as well as the rate dependence of uniaxial material properties. This new multiscale computational method has potential for use in cases where a detailed atomistic-level analysis is necessary in localized spatially separated regions whereas continuum mechanics is adequate in the rest of the material.

3.2. Two-Dimensional Mixed Mode Crack Simulation Using the Material Point Method (Wang et al. [5])

The material point method (MPM) was used by Sulsky, Zhou, and Shreyer [4] to demonstrate its capabilities in the simulation of impact/contact/penetration and interfacial crack growth problems. Because of the use of material points in the description of a continuum, consistent with the particle description (atoms) using molecular dynamics (MD), it is natural to couple MPM with MD for simulation from atomistic to continuum levels. However, in addressing plane stress/plane strain problems, the MPM algorithm and simulation examples available in literature use a regular grid mesh with uniform square cells and enforce velocity and displacement continuities through its background grid, resulting in limitations in dealing with stress concentration, inclined dislocations and inclined crack, etc.

In this investigation, we implemented an irregular mesh in MPM to eliminate the limitations resulting from the use of a regular mesh. The ray-crossing algorithm was employed to determine which cell a material point belongs to after deformation for interpolation and extrapolation of variables between material points and grid nodes. As an example to demonstrate the capability of the MPM using irregular mesh, the stress field in a continuum with an inclined crack is determined using arbitrary quadrilateral cells in the background grid mesh. The use of irregular mesh in MPM was verified by comparing MPM results with ABAQUS/Explicit simulation. The proposed method of using irregular mesh will be an essential element in using MPM to couple with atomistic scale simulation so that MPM can address inclined dislocations and cracks emanating from the atomistic simulation.

3.3. Multiscale Simulations Using Generalized Interpolation Material Point (GIMP) Method and SAMRAI Parallel Processing (Ma et al., [6])

In the simulation of a wide range of mechanics problems including impact/contact/ penetration and fracture, the material point method (MPM) has demonstrated its computational capabilities. To resolve alternating stress sign and instability problems associated with conventional MPM, Bardenhagen and Kober [7] in 2004 introduced the generalized interpolation material point (GIMP) method and implemented for one-dimensional simulations.

In this investigation, we have extended GIMP to 2D and applied to simulate simple tension and indentation problems. For simulations spanning multiple length scales, based on the continuum mechanics approach, we developed a parallel GIMP computational method using the Structured Adaptive Mesh Refinement Application Infrastructure (SAMRAI). SAMRAI is used for multi-processor distributed memory computations, as a platform for domain decomposition, and for multi-level refinement of the computational domain. Nested computational grid levels (with successive spatial and temporal refinements) are used in GIMP simulations to improve the computational accuracy and to reduce the overall computational time. The domain of each grid level is divided into multiple rectangular patches for parallel processing. This domain decomposition embedded in SAMRAI is very flexible when applied to GIMP. As an example to verify the parallel GIMP computing scheme under SAMRAI parallel computing environment, numerical simulations with multiple length scales from nanometer to millimeter were conducted on a 2D nanoindentation problem.

A contact algorithm in GIMP has also been developed for the treatment of contact pair between a rigid indenter and a deformable workpiece. GIMP results are compared with finite element results on indentation for validation. A GIMP nanoindentation problem with five levels of refinement was modeled using multi-processors to demonstrate the potential capability of the parallel GIMP computation.

3.4. Structured Mesh Refinement in Generalized Interpolation Material Point (GIMP) Method for Simulation of Dynamic Problems (Ma *et al.*, [8])

The generalized interpolation material point (GIMP) method, recently developed using a C^1 continuous weighting function, has solved the numerical noise problem associated with material points just crossing the cell borders, so that it is suitable for simulation of relatively large deformation problems. However, this method typically uses a uniform mesh in computation

when one level of material points is used, thus limiting its effectiveness in dealing with structures involving areas of high stress gradients.

In this investigation, a spatial refinement scheme of the structured grid for GIMP was used for simulations with highly localized stress gradients. A uniform structured background grid is used in each refinement zone for interpolation in GIMP for ease of generating and duplicating structured grid in parallel processing. The concept of influence zone for the background node and transitional node is introduced for the mesh size transition. The grid shape function for the transitional node is modified accordingly, whereas the computation of the weighting function in GIMP remains the same. Two other issues are also addressed to improve the GIMP method. The displacement boundary conditions are introduced into the discretization of the momentum conservation equation in GIMP, and a method is implemented to track the deformation of the material particles by tracking the position of the particle corners to resolve the problem of artificial separation of material particles in GIMP simulations. Numerical simulations of several problems, such as tension, indentation, stress concentration and stress distribution near a crack (mode I crack problem) are presented to verify this refinement scheme.

3.5. Multiscale Simulation Using Generalized Interpolation Material Point (GIMP) Method and Molecular Dynamics (MD) (Ma *et al.*, [9])

A new method for multiscale simulation bridging two scales, namely the continuum scale using the generalized interpolation material point (GIMP) method and the atomistic scale using the molecular dynamics (MD), is presented and verified in 2D. The atomistic strain from the molecular dynamics simulation is determined through interpolation of displacement field into Eulerian background grid using the same generalized interpolation functions as in GIMP method. The atomistic strain as defined is consistent with that as determined from virial theorem for interior points, but provide more reasonably stable values at the boundary of MD region or in the handshaking region between MD-GIMP so that physical quantities, such as the displacement, internal forces, and energy density are compatible in the handshaking region. A continuum material point in the continuum is split into smaller material points using multi-level refinement until it has nearly reached the atom size to couple with atoms in the MD region. Consequently, coupling between GIMP and MD is achieved by using compatible deformation and force fields in the transition region between GIMP and MD.

The coupling algorithm is implemented in the Structural Adaptive Mesh Refinement Application Infrastructure (SAMRAI) for parallel processing. Both mode I and mode II crack propagation problems are simulated using the coupling algorithm. The stress field near the crack tip was validated by comparing the results from coupled simulation with purely GIMP simulations of the same model. Coupled simulation results were also compared with pure MD simulation results. In both cases, a very good agreement was obtained.

3.6. Combined Numerical Simulation and Nanoindentation for Determining Mechanical Properties of Single Crystal Copper at Mesoscale (*Liu et al.*,[10])

Constitutive laws are critical in the investigation of mechanical behavior of single crystal or polycrystalline materials in applications spanning from microscale to macroscale. In this investigation, a combined FEM simulation and experimental nanoindentation approach was taken to determine the mechanical behavior of single crystal copper incorporating the mesoplastic constitutive model. This model was implemented in a user-defined subroutine in 3D ABAQUS/Explicit code.

Nanoindentation was modeled using the multiscale modeling technique involving mesoplasticity and elasticity, i.e., mesoplastic constitutive model was used near the local nanoindentation region (where the dislocations are generated) while elastic constitutive model was used in rest of the region in the workmaterial. The meso-mechanical behavior of the crystalline structure and the effect of the mesoplastic parameters on the nanoindentation load–displacement relationships were investigated in the FEM analysis.

Nanoindentation tests were conducted on single crystal copper to determine load–displacement relationships. Appropriate mesoplastic parameters were determined by fitting the simulated load–displacement curves to the experimental data. The mesoplastic model, with appropriate parameters, was then used to determine the stress–strain relationship of a single crystal copper at meso-scale. The effect of indenter radius (3.4–1000 mm) on material hardness under nanoindentation was simulated and found to match the experimental data for several indenter radii (3.4, 10 and 500 mm). A comparison of the topographies of nanoindentation impressions in the experiments with FEM results showed a reasonably good agreement.

3.7. Monte Carlo Simulation of Nanometric Cutting (Komanduri *et al*, [11])

Nanometric cutting of single-crystal materials at conventional cutting speeds (5 ms^{-1}) is simulated for the first time using a new Monte Carlo method that is applicable to systems that

are neither canonical nor microcanonical. This is accomplished by defining a local temperature in the cutting zone using the thermal analysis developed by Komanduri and Hou [12] for conventional machining. Extension of this method to the nanometric regime permits an accurate estimate of the local temperature in cutting. This temperature is then employed in the Boltzmann probability distribution function that is used to determine the acceptance–rejection of Monte Carlo moves in the simulation. Since cutting speed is closely related to cutting temperature, the cutting speed enters the calculation via the thermal analysis equations. The method is applied to nanometric cutting of single-crystal aluminum with the crystal oriented in the (001) plane and cut in the [100] direction. Three positive rake cutting tools, namely, 10°, 30°, and 45° are employed to investigate the effect of rake angle on the forces, the specific energy, and the nature of the chip formation. The method is evaluated by direct comparison with corresponding molecular dynamics simulations conducted under the same conditions.

3.8. Monte Carlo-Steepest Descent (MC-SD) Simulations of Nanometric Cutting (Narulkar *et al.*, [13])

In order to reduce the computational time, Monte Carlo (MC) simulations of nanometric cutting have been modified to include a combination of steepest descent (SD) and Monte Carlo procedures. This MC–SD combination is found to reduce the required computational times by a factor of at least two to three over those achieved using MC methods alone. The MC–SD method is applied to the nanometric cutting of single-crystal aluminum along the (100) plane with different rake angle tools at a cutting speed of 5 m/s. The results obtained from the MC–SD calculations are found to be almost identical to those resulting from the MC simulations.

3.9. *Ab initio* Potential-Energy Surfaces for Complex, Multi-channel Systems Using Modified Novelty Sampling and Feedforward Neural Networks (Raff *et al.*, [14])

A neural network/trajectory approach is presented for the development of accurate potential-energy hypersurfaces that can be utilized to conduct *ab initio* molecular dynamics (AIMD) and Monte Carlo studies of gas-phase chemical reactions, nanometric cutting, and nanotribology, and of a variety of mechanical properties of importance in potential microelectromechanical systems applications. The method is sufficiently robust such that it can be applied to a wide range of polyatomic systems. The overall method integrates *ab initio* electronic structure calculations with importance sampling techniques that permit the critical regions of configuration space to be determined. The computed *ab initio* energies and gradients

are then accurately interpolated using neural networks (NN) rather than arbitrary parametrized analytical functional forms, moving interpolation or least-squares methods. The sampling method involves a tight integration of molecular dynamics calculations with neural networks that employ early stopping and regularization procedures to improve network performance and test for convergence. The procedure can be initiated using an empirical potential surface or direct dynamics.

The accuracy and interpolation power of the method has been tested for two cases, the global potential surface for vinyl bromide undergoing unimolecular decomposition via four different reaction channels and nanometric cutting of silicon. The results show that the sampling methods permit the important regions of configuration space to be easily and rapidly identified, that convergence of the NN fit to the *ab initio* electronic structure database can be easily monitored, and that the interpolation accuracy of the NN fits is excellent, even for systems involving five atoms or more. The method permits a substantial computational speed and accuracy advantage over existing methods, is robust, and relatively easy to implement.

3.10. Prediction of Molecular-Dynamics Simulation Results Using Feedforward Neural Networks: Reaction of C_2 Dimer with an Activated Diamond (100) Surface (Agrawal *et al.* [15])

A new approach involving neural networks (NN) combined with molecular dynamics (MD) has been used for the determination of reaction probabilities as a function of various input parameters for the reactions associated with the chemical vapor deposition (CVD) of carbon dimers on a diamond (100) surface. The data generated by the simulations have been used to train and test neural networks. The probabilities of chemisorption, scattering, and desorption as a function of input parameters, such as rotational energy, translational energy, and direction of the incident velocity vector of the carbon dimer, have been considered. The very good agreement obtained between the predictions of NN and those provided by MD and the fact that, after training the network, the determination of the interpolated probabilities as a function of various input parameters involves only the evaluation of simple analytical expressions rather than computationally intensive algorithms show that neural networks are extremely powerful tools for interpolating the probabilities and rates of chemical reactions. We also find that a NN fits the underlying trends in the data rather than the statistical variations present in the MD results. Consequently, NN can also provide a computationally convenient means of averaging the

statistical variations inherent in MD calculations. In the present case the application of this method is found to reduce the statistical uncertainty in the MD results by a factor of ~3.5.

3.11. Monte Carlo Simulation of Void-nucleated Melting of Silicon via Modification in the Tersoff Potential Parameters (Agrawal *et al.*, [16])

Molecular dynamics (MD) simulations of silicon melting reported in the literature using Tersoff's potential give melting temperatures about 50% higher than the experimental value. To address this discrepancy, we proposed a modification to the values of the parameters of the Tersoff potential. This modification involves a change in the magnitude of 3 of the 12 parameters in the Tersoff potential as an effective means for bringing down the melting point close to the experimental value. The melting point is determined by performing Monte Carlo (MC) simulations using the empirical void-nucleated melting procedure. In addition to an agreement of the computed melting point with the experiment, the modified parameters also bring the density of the liquid and the solid at the melting point into good agreement with the experiment without significantly altering the density and energy values of the solid crystal at room temperature. The coordination number and specific heat of the liquid are also found to be in better agreement with the experiment when the modified set of parameters is used. A comparison of density over a wide range of temperatures shows that the density of the solid predicted by the Tersoff potential with the modified parameters is larger than that given by the unmodified parameters. This difference, however, is not appreciable at room temperature; it increases with temperature and is about 1% at the melting point. The computed cohesive energy of the solid, the point defect energy corresponding to a vacancy, and the surface energy values for (100) and (111) surfaces are also found to be nearly the same for the modified as well as the unmodified parameters.

3.12. Prediction of Molecular-Dynamics Simulation Results Using Feedforward Neural Networks: Reaction of C_2 Dimer with an Activated Diamond (100) Surface (Agrawal *et al* [17].)

The neural network (NN) procedure to interpolate *ab initio* data for the purpose of molecular dynamics (MD) simulations has been tested on the SiO_2 system. Unlike other similar NN studies, here, we studied the dissociation of SiO_2 without the initial use of any empirical potential. During the dissociation of SiO_2 into Si+O or Si+O_2, the spin multiplicity of the system changes from a singlet to a triplet in the first reaction and from a singlet to a pentet in the second.

In this inverstigation we employed four potential surfaces. The first is a NN fit [NN(STP)] to a database comprising the lowest of the singlet, triplet, and pentet energies obtained from density functional calculations in 6673 nuclear configurations. The other three potential surfaces are obtained from NN fits to the singlet, triplet, and pentet-state energies. The dissociation dynamics on the singlet-state and NN(STP) surfaces are reported. The results obtained using the singlet surface correspond to those expected if the reaction were to occur adiabatically. The dynamics on the NN(STP) surface represent those expected if the reaction follows a minimum-energy pathway. This study on a small system demonstrates the application of NNs for MD studies using *ab initio* data when the spin multiplicity of the system changes during the dissociation process.

3.13. A New Method for Estimating local Strains using Nanoindentation and Atomic Force Microscopy (Varghese [18])

A new method was developed for determining strains in very small regions using a micro-tensile stage and nanoindentation as tools. Indents formed by nanoindentation are used as probes to measure local deformations in a small region that can hardly be measured with most existing techniques. Strains were introduced in samples via a small *in-situ* tensile testing system, originally developed for *in-situ* testing inside an SEM, and modified and adapted here for use with the AFM. Variation of local strains in very small regions compared to global strains has also been established. Two approaches were used to estimate strains. In the first approach, length changes of a triangular indent are used to determine normal strains and the corresponding normal and shear strain components. In the second approach, displacement changes in any pair of three indents are monitored for computation of normal strains and the associated normal and shear strain components using strain transformation relations. Indents were made on a polycrystalline copper tensile specimen, and the residual indent geometry is monitored using an atomic force microscope (AFM).

The results show that the shape of the residual indents depends on the crystal orientation and the angle the slip bands make with individual grains. The technique developed is complementary to existing strain measurement techniques, and is especially suitable at micro- and nano-scales. The techniques are validated at two scales, namely, microscale using microindents formed by a Nanoindentation system, and nanoscale using nanoindents made by AFM tips. Strains were compared by measurement of strains by this technique and the

deformations determined using digital image correlation (DIC) under relatively small deformations. Further validation of this technique is made by performing a virtual "digital" tensile test consisting of simultaneous digital stretching and shrinking of an image of the nanoindents by known values and analyzing the strains. Reasonably good agreement has been obtained at both scales. The validated technique is then used to measure relative deformations of multiple grains and deformations within a single grain in the polycrystalline copper. Strains associated with a slip band were determined by this technique

4. Participation of Research Personnel

Senior personnel working on this project are Professors Ranga Komanduri (PI), Hongbing Lu (Co-PI), Lionel M. Raff (Co-PI), and Samit Roy (Co-PI). Professor M. Hagan of OSU and Dr. M. Yoshino of Tokyo Institute of Technology also participated in this project. In addition, the following visiting scholars have participated: Dr. Bo. Wang, Dr. P. M. Agrawal, and Professor Zhen Bing Hou. Drs. R. Hornung and A. Wissink of the Lawrence Livermore National Laboratory (LLNL) interacted with us on the SAMRAI parallel processing work and Dr. Mike Baskes on the development of EAM potential. Two Ph.D. students, namely, Dr. Jin Ma and Dr. Sony Varghase, have completed their Ph.D. theses. Other Ph.D. students who are working towards their theses are: Ms. Yang Liu, Milind Malshe, Rutuparna Narulkar, Nitin P. Dhphalpurkar. The following M.S. students completed their theses: M. Malshe, R. Narulkar, V. Karupapiah, A. N. A. Samadh, Bala S. Sudalayandi, Siva Mathanagopalan, Rohit Raghav, N. P. Dhphalapurkar. Two undergraduate student Ms. Emily Rivers and Thomas Hughes participated in this project.

5. List of Publications

1. H. Lu, N.P. Daphalapurkar, B. Wang, S. Roy and R. Komanduri, "Multiscale Simulation from Atomistic to Continuum – Coupling Molecular Dynamics (MD) with Material Point Method (MPM)," Philosophical Magazine, Vol. 86, No. 20 (2006) 2971-2994

2. Wang, B., Karuppiah, V., Lu, H., Roy, S., and R. Komanduri, "Two-Dimensional Mixed Mode Crack Simulation Using the Material Point Method," Mechanics of Advanced Materials and Structures 12 (2005) 471-484

3. Ma, J., Lu., H., Wang, B., Roy, S., Hornung, R., Wissink, A., and R. Komanduri, "Multiscale Simulations Using Generalized Interpolation Material Point Method (GIMPMP) and

SAMRAI Parallel Processing," Computer Modeling in Engineering & Sciences (CMES), 8 (2) (2005) 135-152

4. J. Ma, H. Lu and R. Komanduri, "Structured Mesh Refinement in Generalized Interpolation Material Point (GIMP) Method for Simulation of Dynamic Problems," Computer Modeling in Engineering & Sciences, Vol. 12, No. 3 (2006) 213-227,

5. J. Ma, H. Lu, B. Wang, R. Hornung, A. Wissink, and R. Komanduri, "Multiscale Simulation Using Generalized Interpolation Material Point (GIMP) Method and Molecular Dynamics (MD)," Computer Modeling in Engineering & Sciences (2006) (in press)

6. Liu, Y., Wang, B., Yoshino, M., Roy, S., Lu, H., and R. Komanduri, "Combined Numerical Simulation and Nanoindentation for Determining Mechanical Properties of Single Crystal Copper at Mesoscale," J of the Mechanics & Physics of Solids (JMPS) 53 (2005)2718-2741

7. Komanduri, R., Narulkar, R., and L. M. Raff, "Monte Carlo Simulation of Nanometric Cutting," Philosophical Magazine 84 (2004) 1155-1183

8. Narulkar, R., Raff, L. M., and R. Komanduri, "Monte Carlo-Steepest Descent (MC-SD) Simulations of Nanometric Cutting," Proc. of the I. Mech E. (London), Part N, J of Nanoengineering and NanoSystems, 218 (2005) 7-16

9. Raff., L. M., Malshe, M., Hagan, M., Doughan, D., Rockley, M. G., and R. Komanduri, "*Ab initio* Potential-Energy Surfaces for Complex, Multichannel Systems Using Modified Novelty Sampling and Feedforward Neural Networks," J of Chemical Physics 122, 084104 (2005) 1-16

10. Agrawal, P. M., Samadh, A. N. A., Raff, L. M., Hagan, M. T., Bukkapatnam, S. T., and R. Komanduri, "Prediction of Molecular-Dynamics Simulation Results Using Feedforward Neural Networks: Reaction of C_2 Dimer with an Activated Diamond (100) Surface," J of Chemical Physics 123 (2005) 224711-1 to 8

11. Agrawal, P. M., Raff, L. M., and R. Komanduri, "Monte Carlo Simulation of Void-nucleated Melting of Silicon via Modification in the Tersoff Potential Parameters," Physical Review B 72 (2005) 125206 1-11

12. Agrawal, P. M., Raff, L. M., Hagan, M., and R. Komanduri, "Molecular Dynamics Investigations of the Dissociation of SiO_2 on an *Ab Initio* Potential Energy Surface Obtained Using Neural Network Methods," J of Chemical Physics 124, 134306 (2006) 1-8

6. List of M.S./Ph.D Theses

1. Jin Ma, "Multiscale Simulation Using the Generalized Interpolation Material Point Method (MPM), Discrete Dislocations, and Molecular Dynamics," Ph.D. Thesis, Mechanical & Aerospace Engineering, Oklahoma State University, Stillwater, OK. (2006)

2. Varghese, S., "A New Method for Estimating Local Strains Using Nanoindentation and Atomic Force Microscopy," Ph.D. Thesis, Mechanical & Aerospace Engineering, Oklahoma State University, Stillwater, OK.(2006)

3. Rohit K. Raghav, "Three-Dimensional Generalized Interpolation Material (GIMP) Simulations in SAMRAI Environment, M.S. Thesis, Mechanical & Aerospace Engineering, Oklahoma State University, Stillwater, OK. (2005)

4. Samadh, Abdul, N. A., "Molecular Dynamics Studies of Chemical Vapor Deposition of Carbon Dimer on a Diamond (100) Surface and Application of Neural Networks for Event Probability Predictions," M.S. Thesis, Mechanical & Aerospace Engineering, Oklahoma State University, Stillwater, OK. (May 2005)

5.Sudalayandi, Bala, S., "Computation of Young's Modulus of Single Wall Carbon Nanotube Using Molecular Dynamics (MD) Simulations," M.S. Thesis, Mechanical & Aerospace Engineering, Oklahoma State University, Stillwater, OK. (April 2005)

6. Malshe, Milind, "*Ab Initio* Molecular Dynamics (AIMD) – A new approach for the development of accurate potentials," M.S. Thesis, Mechanical & Aerospace Engineering, Oklahoma State University, Stillwater, OK. (December 2004)

7. Karuppiah, Venkatesh, "2D implementation of Irregular Mesh in Materials Point Method (MPM) for Simulation of Mixed Mode Crack Propagation in Tension, Ph.D. Thesis, Mechanical & Aerospace Engineering, Oklahoma State University, Stillwater, OK. (May 2004)

8. Mathanagopalan, Siva, "A Neural Network and Molecular Dynamics (MD) Approach for Event Probability Prediction during Chemical Vapor Deposition (CVD) of a Carbon atom on a diamond (111) Surface," M.S. Thesis, Mechanical & Aerospace Engineering, Oklahoma State University, Stillwater, OK. (May 2005)

9. Narulkar, Rutupatna, "Monte Calso (MC) Simulation of Nanometric Cutting," M. S. Thesis, Mechanical & Aerospace Engineering, Oklahoma State University, Stillwater, OK. (2003)

10. Daphalapurkar, Nitin, "Multiscale Simulation From Atomistic to Continuum – Coupling molecular dynamics (MD) with material point method (MPM)," M.S. Thesis, Mechanical & Aerospace Engineering, Oklahoma State University, Stillwater, OK. (2004)

REFERENCES

1. Kalia, R.K., Campbell, T. J., Chatterjee, A., Nakino, A., Vashishta, P., and S. Ogata, *Comp. Phys. Commun.* 128 (2000) 245-259

2. Hartley, C. S., *Mater. Sci. and Eng. A* 319-321 (2001) 133-138

3. Lu, H., Daphalapurkar, N.P., Wang, B., Roy, S., and R. Komanduri, "Multiscale Simulation from Atomistic to Continuum – Coupling Molecular Dynamics (MD) with Material Point Method (MPM)," Philosophical Magazine, Vol. 86, No. 20, pp. 2971-2994, 2006

4. Sulsky, D., Zhou, S.-J., and H. L. Schreyer, *Comput. Phys. Commun.*, 87 (1995) 236-252.

5. Wang, B., Karuppiah, V., Lu, H., Roy, S., and R. Komanduri, "Two-Dimensional Mixed Mode Crack Simulation Using the Material Point Method," Mechanics of Advanced Materials and Structures 12 (2005) 471-484

6. Ma, J., Lu., H., Wang, B., Roy, S., Hornung, R., Wissink, A., and R. Komanduri, "Multiscale Simulations Using Generalized Interpolation Material Point Method (GIMPMP) and SAMRAI Parallel Processing," Computer Modeling in Engineering & Sciences (CMES), 8 (2) (2005) 135-152

7. Bardenhagen, S. G., and E. M. Kober, "The Generalized Interpolation Material Point Method," Computer Modeling in Engineering & Sciences, Vol. 5, N.6 (2004) 477-496

8. Ma, J., Lu, H., and R. Komanduri, "Structured Mesh Refinement in Generalized Interpolation Material Point (GIMP) Method for Simulation of Dynamic Problems," Computer Modeling in Engineering & Sciences, Vol. 12, No. 3, pp. 213-227, 2006

9. Ma, J., Lu, H., Wang, B., Hornung, R., Wissink, A., and R. Komanduri, "Multiscale Simulation Using Generalized Interpolation Material Point (GIMP) Method and Molecular Dynamics (MD)," Computer Modeling in Engineering & Sciences, 2006 (in press)

10. Liu, Y., Wang, B., Yoshino, M., Roy, S., Lu, H., and R. Komanduri, "Combined Numerical Simulation and Nanoindentation for Determining Mechanical Properties of Single Crystal Copper at Mesoscale," J of the Mechanics & Physics of Solids (JMPS) 53 (2005)2718-2741

11. Komanduri, R., Narulkar, R., and L. M. Raff, "Monte Carlo Simulation of Nanometric Cutting," Philosophical Magazine 84 (2004) 1155-1183

12. Komanduri, R. and Z. B. Hou, "Tribology in Metal Cutting: Some Thermal Issues," Trans ASME J of Tribology 123 (2000) 799-815

13. Narulkar, R., Raff, L. M., and R. Komanduri, "Monte Carlo-Steepest Descent (MC-SD) Simulations of Nanometric Cutting," Proc. of the I. Mech E. (London), Part N, J of Nanoengineering and NanoSystems, 218 (2005) 7-16

14. Raff., L. M., Malshe, M., Hagan, M., Doughan, D., Rockley, M. G., and R. Komanduri, "*Ab initio* Potential-Energy Surfaces for Complex, Multichannel Systems Using Modified Novelty Sampling and Feedforward Neural Networks," J of Chemical Physics 122, 084104 (2005) 1-16

15. Agrawal, P. M., Samadh, A. N. A., Raff, L. M., Hagan, M. T., Bukkapatnam, S. T., and R. Komanduri, "Prediction of Molecular-Dynamics Simulation Results Using Feedforward Neural Networks: Reaction of C_2 Dimer with an Activated Diamond (100) Surface," J of Chemical Physics 123 (2005) 224711-1 to 8

16. Agrawal, P. M., Raff, L. M., and R. Komanduri, "Monte Carlo Simulation of Void-nucleated Melting of Silicon via Modification in the Tersoff Potential Parameters," Physical Review B 72 (2005) 125206 1-11

17. Agrawal, P. M., Raff, L. M., Hagan, M., and R. Komanduri, "Molecular Dynamics Investigations of the Dissociation of SiO_2 on an *Ab Initio* Potential Energy Surface Obtained Using Neural Network Methods," J of Chemical Physics 124, 134306 (2006) 1-8

18. Varghese, S., "A New Method for Estimating Local Strains Using Nanoindentation and Atomic Force Microscopy," Ph.D. Thesis, Mechanical & Aerospace Engineering, Oklahoma State University, Stillwater, OK.(2006)

Philosophical Magazine,
Vol. 86, No. 20, 11 July 2006, 2971–2994

Taylor & Francis
Taylor & Francis Group

Multiscale simulation from atomistic to continuum – coupling molecular dynamics (MD) with the material point method (MPM)

H. LU, N. P. DAPHALAPURKAR, B. WANG,
S. ROY and R. KOMANDURI*

Mechanical and Aerospace Engineering, Oklahoma State University,
Stillwater, OK 74078, USA

(*Received 4 February 2006; in final form 9 February 2006*)

A new multiscale simulation approach is introduced that couples atomistic-scale simulations using molecular dynamics (MD) with continuum-scale simulations using the recently developed material point method (MPM). In MPM, material continuum is represented by a finite collection of material points carrying all relevant physical characteristics, such as mass, acceleration, velocity, strain and stress. The use of material points at the continuum level provides a natural connection with the atoms in the lattice at the atomistic scale. A hierarchical mesh refinement technique in MPM is presented to scale down the continuum level to the atomistic level, so that material points at the fine level in MPM are allowed to directly couple with the atoms in MD. A one-to-one correspondence of MD atoms and MPM points is used in the transition region and non-local elastic theory is used to assure compatibility between MD and MPM regions, so that seamless coupling between MD and MPM can be accomplished. A silicon single crystal under uniaxial tension is used in demonstrating the viability of the technique. A Tersoff-type, three-body potential was used in the MD simulations. The coupled MD/MPM simulations show that silicon under nanometric tension experiences, with increasing elongation in elasticity, dislocation generation and plasticity by slip, void formation and propagation, formation of amorphous structure, necking, and final rupture. Results are presented in terms of stress–strain relationships at several strain rates, as well as the rate dependence of uniaxial material properties. This new multiscale computational method has potential for use in cases where a detailed atomistic-level analysis is necessary in localized spatially separated regions whereas continuum mechanics is adequate in the rest of the material.

1. Introduction

Material failure at all length scales experiences such processes as elastic deformation, dislocation generation and propagation, cleavage, crack initiation and growth, and final rupture. In recent years, combined atomistic and continuum simulations have received increasing attention due to their potential linkage between structure–property relationships from nano to macro levels [1–11]. Various computational methods have been developed to couple spatial and temporal scales. Figure 1a is a

*Corresponding author. Email: ranga@ceat.okstate.edu

Philosophical Magazine
ISSN 1478-6435 print/ISSN 1478-6443 online © 2006 Taylor & Francis
http://www.tandf.co.uk/journals
DOI: 10.1080/14786430600625578

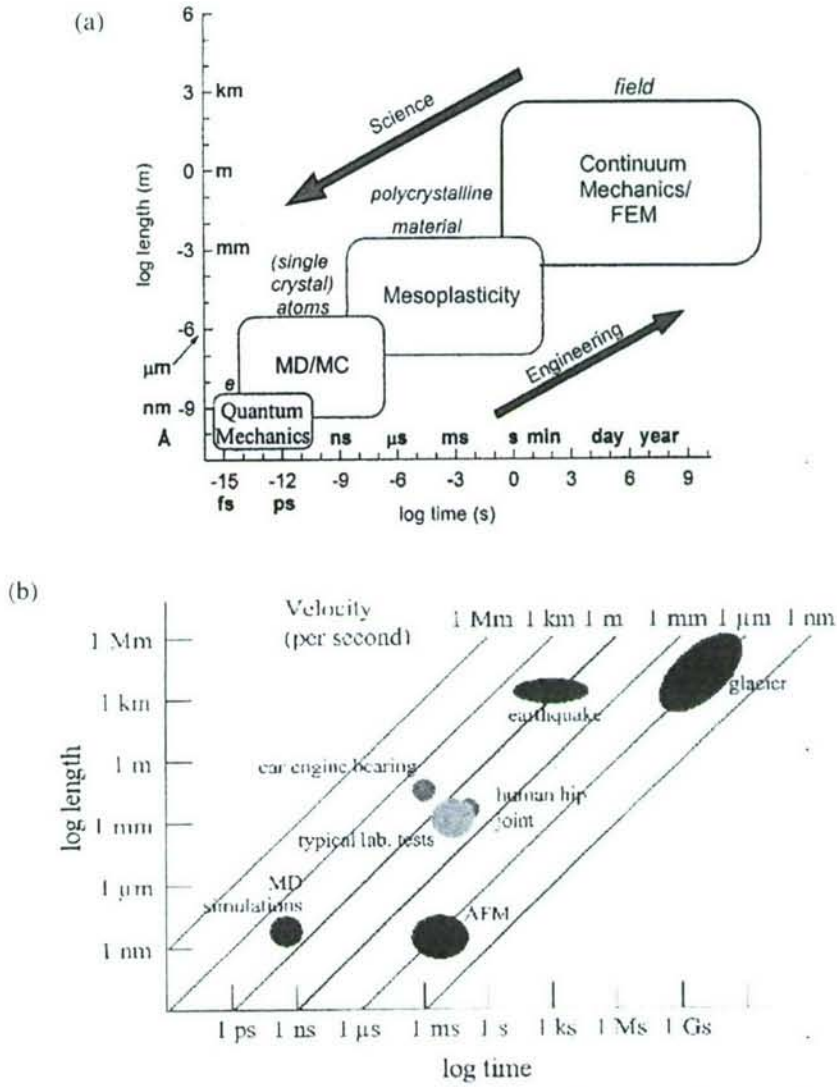


Figure 1. (a) Schematic of simulations at various levels. (b) Schematic of various length and time scales in tribology.

schematic of the various computer simulation methods used in the development of scaling laws as a function of spatial and temporal variables [12]. At the subatomic level, we use quantum mechanics or ab initio calculations to develop potential energy functions. With an increase in length (or time) scale, we model material behaviour using molecular dynamics (MD) and Monte Carlo (MC) simulations, then to micro or mesoplastic, and finally to continuum mechanics, such as finite element method (FEM). Figure 1b is a schematic showing various spatial and temporal scales for a typical tribological phenomenon for a wide range of applications [13]. It may be noted at the outset that any scaling law developed would involve both temporal and

spatial scales. At the atomistic level, MD is the preferred mode while at the continuum level, FEM is the dominant numerical technique used. Consequently, most published results on combined atomistic and continuum simulations have focused on MD/FEM coupling.

2. Brief review of literature

In combined atomistic and continuum simulations, material is divided into atomistic and continuum regions. Seamless coupling is required for the combined MD/FEM simulations, which is a key issue in multiscale simulations. Using a combined finite-element and atomistic model, Kohlhoff *et al.* [14] introduced the transition region between atomic lattice and continuum. An early suggestion for coupling the continuum and the atomistic regimes was due to Hoover *et al.* [15], and more recently due to Curtin and Miller [16] and Shiari *et al.* [17]. Ghoniem *et al.* [18], Liu *et al.* [19] and Lu and Kaxiras [20] have reviewed in detail various coupling methods linking atomistic to continuum simulations.

The hybrid MD/FEM method introduces an MD/FEM handshake region for the interface between the MD and the FEM regions, where MD atoms and FEM nodes are coupled by mutual displacement boundary conditions. The kinetic and strain (potential) energy is defined for the entire system including the handshaking interactions at the interface. The FEM mesh in the MD/FEM handshaking region is scaled down to coincide with the atomic lattice. A one-to-one correspondence of MD atoms and FEM nodes is required in the transition region. The FEM mesh is gradually made coarser in areas far away from the interested atomistic region. Additionally, the width of the transition region is taken to be equal to the cut-off distance of the interaction potential used in the MD region. This provides a complete set of neighbours within the interaction range for all atoms in the MD region. Atoms/FEM nodes that belong to the transition region not only interact via the interaction potential with the MD region but are also part of the nodes in the FEM region. The positions and velocities of these atoms/FEM nodes in the transition region (both on the MD side and on the FEM side) must be consistent with each other. The forces exerted on the atoms in the transition region, due to the interaction with the MD region, make up the external forces on the FEM nodes for the FEM simulation. However, under large deformations, the FEM elements become distorted and it is difficult to implement the entire computational process in the FEM simulations.

Recently, a new computational method, namely, the material point method (MPM) was developed by Sulsky and others [21, 22] at the University of New Mexico with support from the Sandia National Laboratories. Compared to FEM, MPM has the following advantages: (1) MPM is able to handle large deformation in a more natural manner so that mesh lock-up present in FEM is avoided; (2) MPM can easily couple with molecular dynamics (MD) simulations because of the use of material points (similar to atoms used in MD) instead of elements in FEM; (3) parallel computation is more straightforward in MPM because of the use of a grid structure that is consistent with parallel computing grids; and (4) use of the

background grid in MPM enables structured adaptive refinement for local interested region. As a result, MPM is a logical choice for coupling MD to continuum approach for multiscale simulations. However, the conventional MPM discretizes the material using a regular grid with uniform-square (two-dimensional) or cubic (three-dimensional) cells. Such a mesh is inefficient in dealing with stress concentration issues and cannot be refined to scale down the size to different levels [23]. Consequently, a hierarchical mesh refinement technique in MPM simulations has been developed in this investigation.

In this paper, a seamless MD/MPM coupling approach is presented in which MD modelling is used in the local area of interest where the variation is at an atomistic scale and MPM simulation in the rest of the material. A multiscale simulation of uniaxial tension is carried out on a silicon specimen at different strain rates to illustrate this methodology. A hierarchical mesh refinement technique is introduced in MPM to scale down from the continuum level into the atomistic level near the MD/MPM transition region where the MPM points overlap the MD atoms. A seamless coupling is realized for complete compatibility of both strain and stress for MD and MPM regions. In tensile simulations, the effects of strain rate on the nature of deformation and material properties are investigated based on the simulated stress–strain relations for silicon.

3. Computational methodology

3.1. *Molecular Dynamics*

3.1.1. Theoretical background. In molecular dynamics (MD), the Newtonian second law for the time evolution of a system is expressed by [24, 25]

$$m_i d^2\mathbf{r}_i/dt^2 = \mathbf{f}_i, \quad (i = 1, \ldots, N) \tag{1}$$

where N is the total number of atoms, m_i is the mass of the ith atom with position vector $\mathbf{r}_i$, $\mathbf{f}_i = -\nabla_i V(\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N)$ is the force acting on the ith atom due to interaction with other atoms in the system, and $V(\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N)$ is the interaction potential. The initial positions and velocities of the atoms, together with the interaction potential, define the whole set of thermodynamic, elastic and mechanical properties of the material.

The set of $3N$ second-order differential equations, equation (1), is often solved by recasting it as a set of $6N$ first-order Hamiltonian equations of motion, thus

$$\begin{aligned} d\mathbf{p}_i/dt &= \mathbf{f}_i = -\nabla_i V(\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N), \\ d\mathbf{r}_i/dt &= \mathbf{p}_i/m_i \end{aligned} \tag{2}$$

where $\mathbf{p}_i$ are momenta. Under the given initial positions and momenta of the system, integration of equation (2) yields the total trajectory of the system. With the knowledge of the trajectories of all the atoms, one can calculate spatial and temporal distributions of energy, temperature and pressure, as well as monitor the structural and phase changes in the system.

The atomistic potential for the interaction between Si–Si atoms herein is a general form of Tersoff-type three-body potential [26, 27]. The total energy as a function of atomic coordinates is given by

$$E = \frac{1}{2}\sum_{i \neq j} W_{ij}, \tag{3}$$

where i and j label the atoms of the system and W_{ij} is the bond energy of all the atomic bonds in the control volume represented by

$$W_{ij} = f_c(r_{ij})\left[f_R(r_{ij}) + b_{ij} f_A(r_{ij})\right], \tag{4}$$

where r_{ij} is the interatomic distance. The f_R and f_A functions are the repulsive and attractive forces between atoms i and j, respectively. These functions are expressed by

$$f_R(r_{ij}) = A_{ij}\exp(-\lambda_{ij} r_{ij}), \tag{5}$$

and

$$f_A(r_{ij}) = -B_{ij}\exp(-\mu_{ij} r_{ij}). \tag{6}$$

The $f_c(r_{ij})$ function acts as a cut-off function that smoothly attenuates the interaction as r_{ij} approaches the cutoff radius S_{ij}.

$$f_c(r_{ij}) = \begin{cases} 1 & \text{for } r_{ij} \leq R_{ij} \\ \dfrac{1}{2} + \dfrac{1}{2}\cos\left(\dfrac{r_{ij} - R_{ij}}{S_{ij} - R_{ij}}\pi\right) & \text{for } R_{ij} \leq r_{ij} \leq S_{ij} \\ 0 & \text{for } r_{ij} \geq S_{ij} \end{cases} \tag{7}$$

The b_{ij} function is given by

$$b_{ij} = \chi_{ij}\left(1 + \beta_i^{n_i}\zeta_{ij}^{n_i}\right)^{-(1/2n_i)}, \tag{8}$$

where

$$\zeta_{ij} = \sum_{k \neq i,j} f_c(r_{ik}) g(\theta_{ijk}), \tag{9}$$

with

$$g(\theta_{ijk}) = 1 + \frac{c_i^2}{d_i^2} - \frac{c_i^2}{d_i^2 + (h_i - \cos\theta_{ijk})^2}, \tag{10}$$

where θ_{ijk} is the bond angle between bonds ij and ik. The parameters used in the Tersoff's potentials for silicon, i.e. A, B, R, S, λ, μ, β, n, c, d and h, are given in table 1 [26, 27].

3.1.2. Strain and stress. In MD simulations, the stress–strain relation is usually determined from tensile models [28, 29]. In each increment of applied deformation, a nominal strain can be determined by the nominal length change normalized by initial length for the entire MD model in the loading direction. In thermodynamics,

Table 1. Material parameters in Tersoff's potential for silicon [26, 27].

Parameter	A (eV)	B (eV)	λ (Å^{-1})	μ (Å^{-1})	β	n	c	d	H	R (Å)	S (Å)
Silicon	1.83×103	4.71×10^2	2.48	1.73	1.1×10^{-5}	0.79	1.0×10^5	16.22	−0.6	2.7	3.0

the stress in a solid is defined as the partial derivative of the internal energy with respect to the strain per unit volume. In continuum mechanics, the components of the stress tensor for a hyperelastic material are given by

$$\sigma_{ij} = \frac{1}{V}\left(\frac{\partial E}{\partial \varepsilon_{ij}}\right)_S, \tag{11}$$

where V is the volume of the solid, E is the total internal energy, ε_{ij} are the components of the strain tensor and S is entropy. In special cases, if the internal energy is the strain energy, Hooke's law is the result of equation (11). In MD, the total energy is the summation of the energy of individual atoms, E^α, and is expressed as

$$E^\alpha = T^\alpha + U^\alpha = \frac{1}{2} M^\alpha (\mathbf{v}^\alpha)^2 + \Phi^\alpha(\mathbf{r}), \tag{12}$$

where T^α is the kinetic energy, U^α the potential energy, M^α the mass, $\mathbf{v}^\alpha$ the magnitude of its velocity and $\Phi^\alpha(\mathbf{r})$ the potential energy at the atom location $\mathbf{r}$. Thus, the stress components for a given atom are written as

$$\sigma_{ij}^\alpha = -\frac{1}{V^\alpha}\left(M^\alpha v_i^\alpha v_j^\alpha + \sum_\beta F_i^{\alpha\beta} r_j^{\alpha\beta}\right), \tag{13}$$

where V^α is the atomic volume of atom α, v_i^α and v_j^α are components of velocity. $F_j^{\alpha\beta}$ are the components of force between atoms α and β from the derivative of the potential energy, and $r_j^{\alpha\beta}$ are the components of the distance between atoms α and β. The nominal stress is computed as the average of the atomic stresses for the volume of the model. Thus, the nominal stress components of each model are expressed by

$$\sigma_{ij} = -\frac{1}{V}\sum_\alpha \left(M^\alpha v_i^\alpha v_j^\alpha + \sum_\beta F_i^{\alpha\beta} r_j^{\alpha\beta}\right), \tag{14}$$

where V is the volume of the MD model and $V = \sum_\alpha V^\alpha$.

3.2. *The Material Point Method (MPM)*

3.2.1. Theoretical background. In the material point method (MPM) [21, 22] a material continuum is discretized into a finite collection of N material points, as shown in figure 2. Each material point is assigned a mass ($m_p, p = 1, \ldots, N$) consistent with the material density and volume of the point, and all other variables,

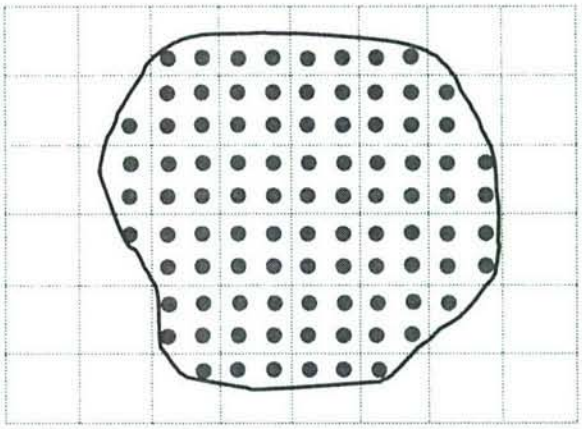


Figure 2. MPM grid cells with material points.

such as position, acceleration, velocity, strain and stress. The motion equations are solved at the points for each time step of the analysis using a background computational grid. Information is transferred from the material points to grid nodes by using standard finite element shape functions,

$$\phi(\mathbf{x}) = \sum_{i=1}^{n} \phi_i N_i(\mathbf{x}), \tag{15}$$

where n is the number of nodes in the grid and subscript i refers to the nodal values of $\phi(\mathbf{x})$ containing $\mathbf{x}$.

The Newtonian motion equations for material points are

$$\rho d^2\mathbf{u}/dt^2 = \rho\nabla\sigma^s + \rho\mathbf{f}_b, \tag{16}$$

where the mass density $\rho = \sum_{p=1}^{N} m_p\delta(\mathbf{x} - \mathbf{x}_p)$, m_p is material point mass, $\mathbf{x}_p$ is material point location, $\delta(\mathbf{x} - \mathbf{x}_p)$ is the Delta function, $\mathbf{u}$ is the displacement, the acceleration $\mathbf{a} = d^2\mathbf{u}/dt^2$, $\mathbf{f}_b$ is the specific body force and σ^s is a specific stress tensor, which is the stress tensor σ divided by the mass density. The material constitutive relation can be represented as

$$\sigma^s = \mathbf{C}^{(s)}\varepsilon \quad \text{and} \quad \varepsilon = \frac{1}{2}\left[(\nabla\mathbf{u}) + (\nabla\mathbf{u})^{\mathrm{T}}\right], \tag{17}$$

where $\mathbf{C}^{(s)}$ is the fourth-order elasticity tensor and ε is the strain tensor. This paper will consider only linear elastic materials at the continuum level.

Applying the interpolation scheme for acceleration into the weak form of Newton's equation gives

$$\sum_{j=1}^{n} m_{ij}\mathbf{a}_j = \mathbf{f}_i^{\mathrm{int}} + \mathbf{f}_i^{ext}, \tag{18}$$

where the mass matrix is given by

$$m_{ij} = \sum_{p=1}^{N} m_p N_i(\mathbf{x}_p) N_j(\mathbf{x}_p), \tag{19}$$

and the internal and the external forces on the node i are

$$\mathbf{f}_i^{\text{int}} = -\sum_{p=1}^{N} m_p \sigma^s(\mathbf{x}_p) \nabla N_i(\mathbf{x}_p) \quad \text{and} \quad \mathbf{f}_i^{ext} = \hat{\tau}_i + \sum_{p=1}^{N} m_p \mathbf{f}_b(\mathbf{x}_p) N_i(\mathbf{x}_p), \tag{20}$$

where $\hat{\tau}_i$ is the surface traction associated with grid point i.

The grid point accelerations obtained from equation (18) are then used to update the position, velocity, stress, strain and temperature of the material points.

3.2.2. Hierarchical mesh refinement. The conventional MPM method uses a regular grid cell mesh with equally spaced nodes. Adaptive mesh refinement in areas with stress concentration is not possible using such a mesh. To accommodate multiple level meshes, a hierarchical mesh refinement technique is presented herein. For the two-dimensional (2D) situation, the hierarchical mesh refinement with three levels using four-node regular cells (1, 4, 7–10) and five-node transition cells (2, 3, 5, 6) are shown in figure 3a. For the regular cell with four nodes in figure 3b, the shape functions used in equation (15) are expressed as [30]:

$$\begin{aligned} N_1 &= \frac{1}{4}(1-\xi)(1-\eta), \quad N_2 = \frac{1}{4}(1+\xi)(1-\eta), \\ N_3 &= \frac{1}{4}(1+\xi)(1+\eta), \quad N_4 = \frac{1}{4}(1-\xi)(1+\eta). \end{aligned} \tag{21}$$

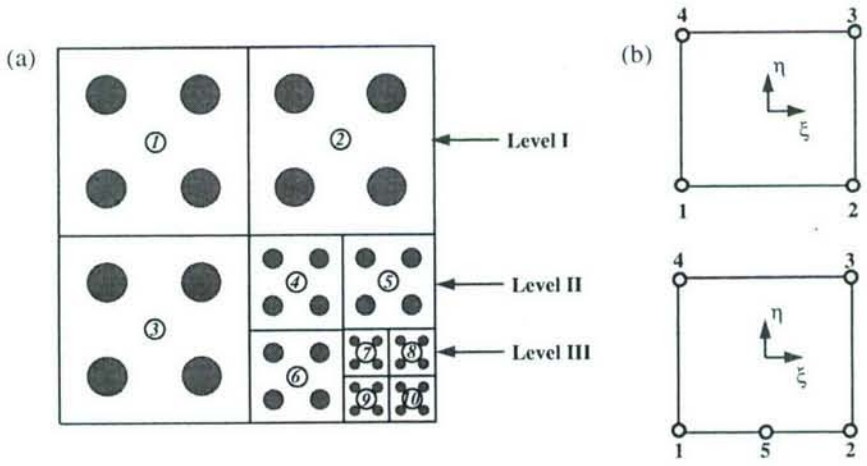


Figure 3. Hierarchical mesh refinement in 2D MPM.

where (ξ, η) are natural coordinates, as shown in figure 3b. Shape functions for transition cells with five nodes are given by:

$$N_1 = \frac{1}{4}(1-\xi)(1-\eta) - \frac{1}{2}N_5, \quad N_2 = \frac{1}{4}(1+\xi)(1-\eta) - \frac{1}{2}N_5,$$
$$N_3 = \frac{1}{4}(1+\xi)(1+\eta), \quad N_4 = \frac{1}{4}(1-\xi)(1+\eta), \tag{22}$$
$$N_5 = \frac{1}{2}(1-\xi^2)(1-\eta).$$

Similarly, for the three-dimensional (3D) situation, a hierarchical refinement mesh with two levels is shown in figure 4a using regular cells with eight nodes and transition cells with nine (cells 2, 4 and 6) and thirteen nodes (cells 1, 3 and 5). The shape functions for regular cells with eight nodes in figure 4b are given by [30]:

$$N_i = \frac{1}{8}(1+\xi_0)(1+\eta_0)(1+\zeta_0), \quad \xi_0 = \xi\xi_i,$$
$$\eta_0 = \eta\eta_i, \quad \zeta_0 = \zeta\zeta_i \ (i = 1, \ldots, 8), \tag{23}$$

where $i(\xi_i, \eta_i, \zeta_i)$: 1 (1, 1, 1); 2 (-1, 1, 1); 3 (-1, -1, 1); 4(1, -1, 1); 5(1, 1, -1); 6(-1, 1, -1); 7(-1, -1, -1); 8(1, -1, -1), and (ξ, η, ζ) are natural coordinates as shown in figure 4. The shape functions for transition cells from nine to thirteen nodes in figure 4b are expressed by

$$\hat{N}_i = \frac{1}{8}(1+\xi\xi_i)(1+\eta\eta_i)(1+\zeta\zeta_i) \quad (i = 1, \ldots, 8),$$
$$N_1 = \hat{N}_1, \quad N_2 = \hat{N}_2 - \frac{1}{2}N_{10} - \frac{1}{2}N_{11} - \frac{1}{4}N_{13},$$
$$N_3 = \hat{N}_3 - \frac{1}{2}N_{11} - \frac{1}{2}N_{12} - \frac{1}{4}N_{13}, \quad N_4 = \hat{N}_4,$$
$$N_5 = \hat{N}_5, \quad N_6 = \hat{N}_6 - \frac{1}{2}N_9 - \frac{1}{2}N_{10} - \frac{1}{4}N_{13},$$
$$N_7 = \hat{N}_7 - \frac{1}{2}N_9 - \frac{1}{2}N_{12} - \frac{1}{4}N_{13}, \quad N_8 = \hat{N}_8, \tag{24}$$
$$N_9 = \frac{1}{4}(1-\xi^2)(1-\eta)(1+\zeta)(\zeta), \quad N_{10} = \frac{1}{4}(1-\xi)(1-\eta)(1-\zeta^2)(-\xi),$$
$$N_{11} = \frac{1}{4}(1-\xi^2)(1-\eta)(1-\zeta)(-\zeta), \quad N_{12} = \frac{1}{4}(1+\xi)(1-\eta)(1-\zeta^2)(\xi),$$
$$N_{13} = \frac{1}{2}(1-\xi^2)(1-\zeta^2)(1-\eta).$$

In the 3D MPM computational formulation using the hierarchical mesh with multiple levels, equation (23) is used for regular cells at each individual level while equation (24) is employed for the transition cells between neighbouring two levels.

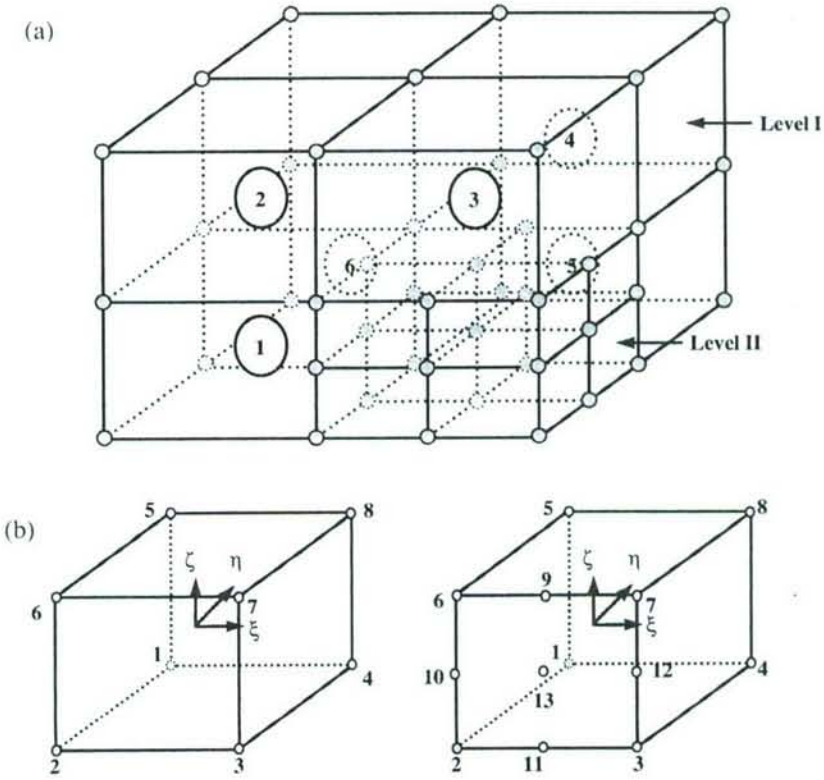


Figure 4. 3D MPM regular and transition cells.

3.3. *Seamless Coupling between MD/MPM*

Simulation of dynamic problems using both MD and MPM requires integration of the equations of motion, equations (1) and (16). The format of these two equations is similar, leading to a natural coupling of the two types of simulations. There are essentially three regions in coupled MD/MPM simulations. They are MD, MPM, and their transition regions. The MD region is modelled using the approach described in section 3.1. In the MD region, atoms are initially placed at diamond lattice points. The MPM region is modelled using the method described in section 3.2. In the MPM region, eight material points are equally positioned in each cubic cell. The coupling of the two descriptions of the media is brought about using a transition region where both MPM points and MD atoms are initially overlapped and coincide with the atomic lattice. The width of the transition region is equal to the cut-off distance of the interaction potential used in the MD region. This provides a complete set of neighbours within the interaction range for all atoms in the MD region. Atoms/points that belong to the transition region not only interact via the interaction potential with the MD region but also are part of points in the MPM region. The positions and velocities of these atoms/points in the transition region (both on the MD side and on the MPM side) must be consistent with each other.

Table 2. Material properties of single crystal α-silicon [31].

Material properties	Lattice constant (Å)	Density ($g\,cm^{-3}$)	Young's modulus (GPa)	C_{11} (GPa)	C_{12} (GPa)	C_{44} (GPa)	Poisson's ratio	Bulk modulus (GPa)	Fracture stress (MPa)
Silicon	5.43	2.33	130	165.78	63.94	79.57	0.28	100	62

The atom velocities in the transition region due to the interaction with the MD region furnish the loading boundary condition on overlapped points for the MPM simulation. To avoid a density mismatch at the boundary, the mass at each point for MPM in the transition region is set equal to the mass of each MD atom.

Using the above method to deal with the transition region, the displacement as well as strain fields are compatible, while stress fields possibly are not compatible. To ensure the complete compatibility of both strain and stress for seamless coupling between MD and MPM, we use the non-local elasticity theory [15] in this investigation. The elastic energy is expanded in terms of the strain ε in the Taylor series form as:

$$E(\varepsilon) = E(0) + \left(\partial E/\partial \varepsilon_{ij}\right)\big|_0 \varepsilon_{ij} + \frac{1}{2}\left[\partial^2 E/\left(\partial \varepsilon_{ij}\partial \varepsilon_{kl}\right)\right]\big|_0 \varepsilon_{ij}\varepsilon_{kl} + \frac{1}{6}\left[\partial^3 E/\left(\partial \varepsilon_{ij}\partial \varepsilon_{kl}\partial \varepsilon_{mn}\right)\right]\big|_0 \varepsilon_{ij}\varepsilon_{kl}\varepsilon_{mn} + \cdots. \tag{25}$$

Therefore, the material elastic constants are required to satisfy the following conditions:

$$c_{ij} = \left(\partial E/\partial \varepsilon_{ij}\right)\big|_0 \quad \text{(first-order elastic constants)},$$
$$c_{ijkl} = \left[\partial^2 E/(\partial \varepsilon_{ij}\partial \varepsilon_{kl})\right]\big|_0 \quad \text{(second-order elastic constants)},$$
$$c_{ijklmn} = \left[\partial^3 E/(\partial \varepsilon_{ij}\partial \varepsilon_{kl}\partial \varepsilon_{mn})\right]\big|_0 \quad \text{(third-order elastic constants)}.$$

Using the natural state as the reference configuration, the first-order elastic constants are zero and only the second- and higher-order elastic constants need to be assigned to the continuum. As linear elasticity theory is used in this investigation, the second-order elastic constants of the lattice and continuum regions require matching. Material properties of single crystal α-silicon are listed in table 2 [31].

4. Numerical implementation

In this paper, a tension model is used in coupled MD/MPM simulations. Using the seamless coupling algorithms described in section 3, a coupled MD/MPM code has been developed for multiscale simulations. The computational scheme or the flow chart used for the coupled MD and MPM simulation of silicon under tension is shown in figure 5. To generate the tension model, initial position data must be assigned first to all atoms and points in both MD and MPM regions, which include atoms/points in the MD/MPM transition regions. Loading and boundary conditions

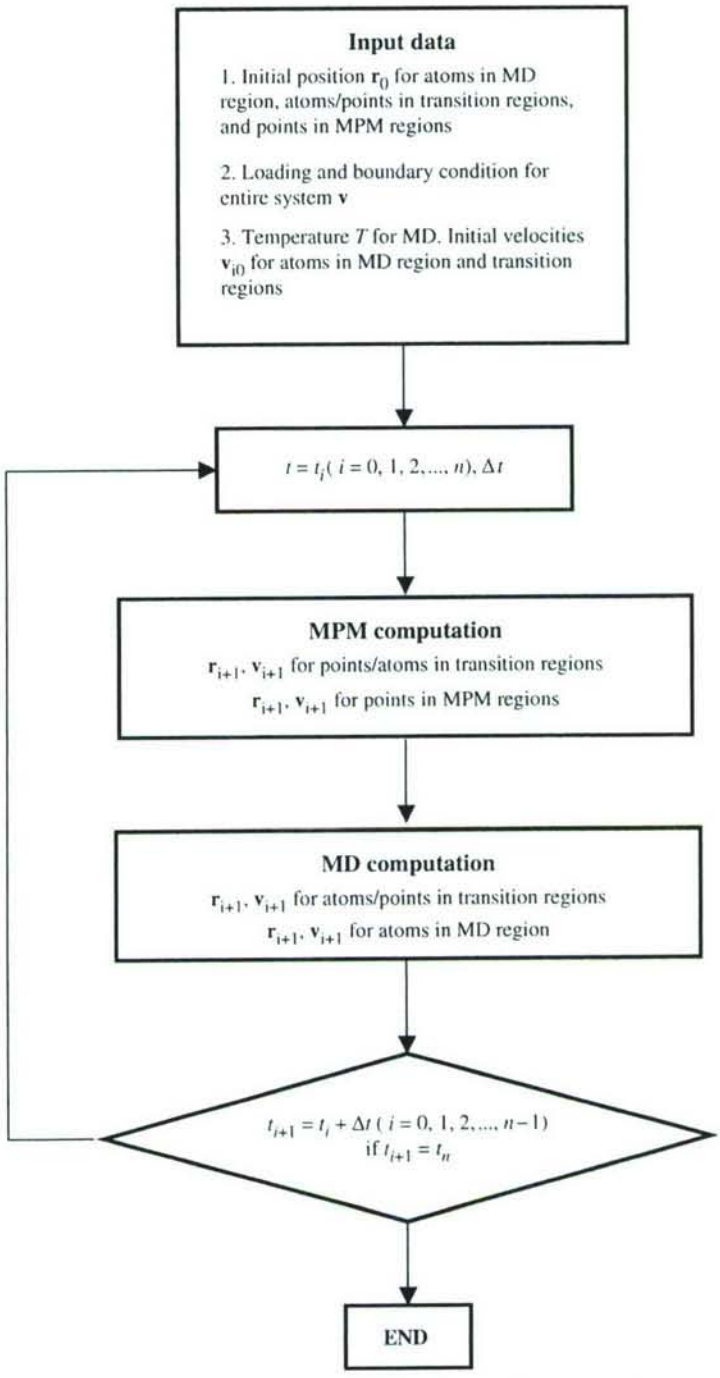


Figure 5. Coupled MD and MPM computational flow chart.

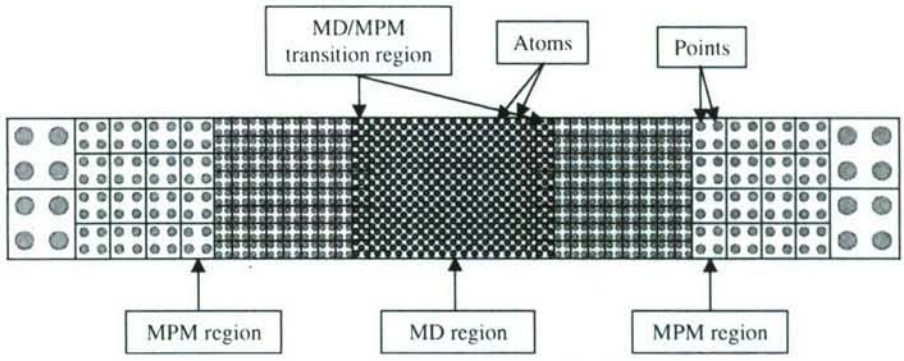


Figure 6. Schematic of tensile model for coupled MD/MPM simulations.

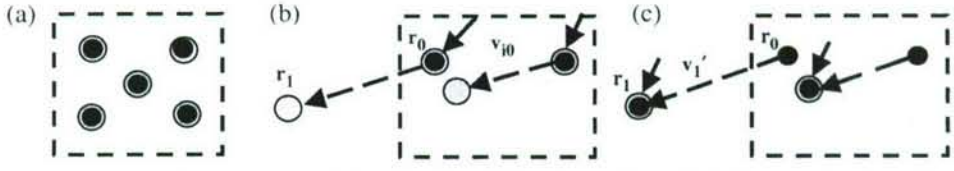


Figure 7. (a) MD atoms and MPM points initially overlapped in the transition region. (b) MPM points moving under velocity. (c) MD atoms moving same positions as MPM points (black: MD atoms; grey: MPM points; dash line: MPM grid).

are also prescribed for the entire system. Based on the given temperature, MD computation is first initiated so that the initial velocities for all atoms can be determined. The initial velocities will provide boundary conditions for the MPM region through the transition region. Then, MPM computation can be started and MPM results in the transition region provide boundary conditions for the MD region. Thus, MD computation can be iterated and MD results in the transition region provide new boundary conditions in MPM computation for the next time step. This iteration process continues until the completion of the simulation. For the purpose of illustrating the method, a schematic of simple tension model, which consists of one MD region, two transition regions, and two MPM regions with three hierarchical mesh refinement levels, is shown in figure 6.

The next item of interest is coupling details in the transition region. For the MPM cells in the MD/MPM transition regions (shown in figure 6), MPM points are arranged following MD atoms at the atomic lattice sites, which have the interaction with the MD region. In other words, MD atoms and MPM points in the transition region are in one-to-one correspondence and are initially overlapped at the same locations, as shown in figure 7a. It may be noted that the MD time step increment is normally much smaller than that in MPM, so that MD and MPM computations have to be matched well in coupling simulations. We use adaptive time step increments: in the transition region wherein the MD time step increment and MPM time step increment are the same. For continuum region, the time step increment

gradually increases for the area farther away from the atomic region. In this simple model, we chose the same time step increment for both MD and MPM computations. For time integration, the time step increment is $\Delta t = 1.018$ fs (10^{-15} s). The steps involved in coupled MD/MPM simulations are as follows:

Step 1: Initial positions $\mathbf{r}_0$ for all atoms in the MD region and all points in the MPM regions, which include atoms/points in the MD/MPM transition regions, are given according to the dimensions of the tension model and the crystal structure of the material. At the absolute temperature T, MD computation is initiated and initial velocities v_{i0} for all MD atoms are calculated. The initial velocities at atoms are applied to the overlapped MPM points in the transition regions as a boundary condition for MPM regions, as shown in figure 7b.

Step 2: MPM computation is conducted at the time step t_0 and new positions $\mathbf{r}_1$ and velocities $\mathbf{v}_1$ are obtained for all MPM points. The new positions at points are applied to the overlapped MD atoms in the transition regions as a boundary condition for MD simulation as shown in figure 7c.

Step 3: MD computation is performed at the time step t_0 and the updated positions $\mathbf{r}'_1$ and velocities $\mathbf{v}'_1$ are obtained for all atoms. The updated velocities at atoms are applied to the overlapped MPM points in the transition regions as a new boundary condition for MPM simulations at the next time step t_1.

Step 4: Repeat the same algorithm to steps 2 and 3 in turns until the entire computation for the tension model is implemented by means of coupled MD and MPM approach.

During coupling of MD with MPM simulations, MPM always provides the updated positions for MD atoms in the transition regions, while MD in turn provides the updated velocities for MPM points in the transition regions. It should be noted that following this algorithm from step 1 to step 4, both atoms for MD and points for MPM in the transition region naturally maintain displacement compatibility so that strains are compatible as well. However, the stress compatibility cannot be ensured automatically. Additionally, mass mismatch exists for both MD atoms and MPM points in the transition region, since mass determined by the mass density and the occupied volume is assigned to each point in the MPM region. To ensure complete compatibility of both strain and stress for seamless coupling between MD and MPM, non-local elasticity theory is used in this study. Based on this theory, for homogenous, linear elastic materials, only the second-order elastic constants require matching between atomistic and continuum regions. In this investigation, we take elastic constants for continuum region from the lattice theory and use the same mass for both MD atoms and MPM points in the transition regions.

5. Results and discussion

Figure 8 is a 3D coupled MD/MPM model at the initial stage of tensile testing showing the MD region, MD/MPM transition regions, and MPM regions. There are three levels in the MPM region. The dimensions of this model

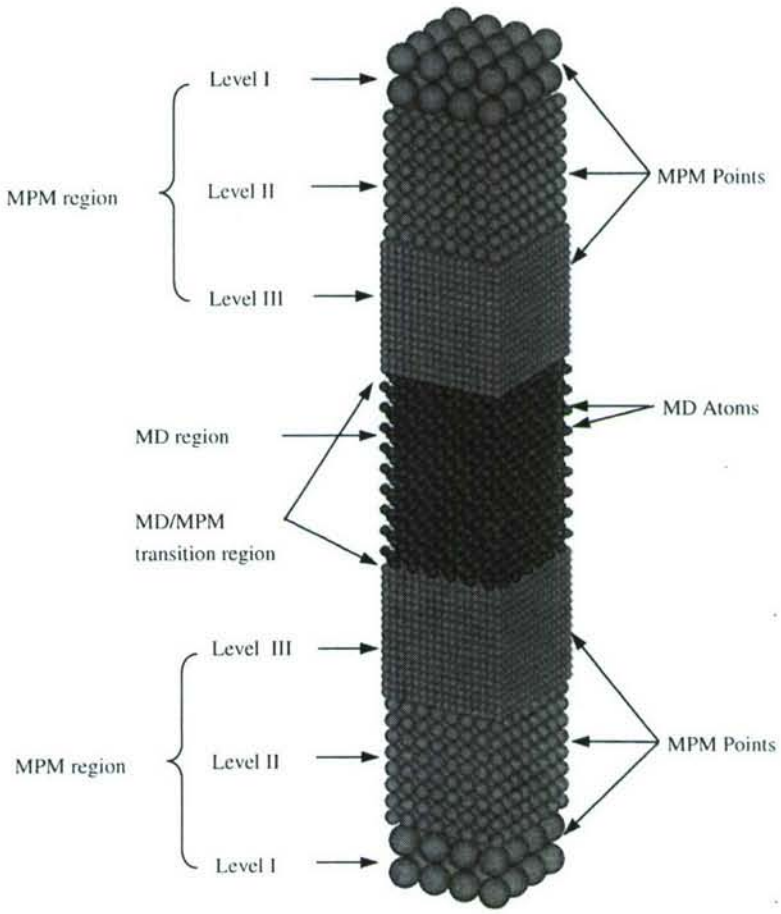


Figure 8. Coupled MD/MPM model showing the initial stage of tensile testing.

are 32.6 Å × 32.6 Å × 222.8 Å. There are 2901 atoms in the MD region and 9280 material points in the MPM regions, which include atoms/points in the MD/MPM transition regions. In the transition regions, MD atoms and MPM points are initially overlapped in one-to-one correspondence and coincide with the atomic lattice. In the MD region, the crystal is set up with cube orientation [001] and the tensile loading is applied along the [001] direction under controlled velocity. The initial temperature is 293 K for the MD simulations. The elastic constants, the Young's modulus, the Poisson's ratio, and other relevant parameters, given in table 2, are used in the MPM regions. The coupling simulations on the tension model are conducted for the silicon work material at different strain rates, i.e. 44.88 ns^{-1}, 53.86 ns^{-1}, 62.84 ns^{-1}, 67.33 ns^{-1}, 80.79 ns^{-1}, and 89.76 ns^{-1}. It may be noted that very high strain rates are used of necessity to minimize the computational time for coupling simulations. For the coupling simulation at a strain rate of 44.88 ns^{-1}, the total simulation time was 4 ps (10^{-12} s) and the time step increment used was 1.018 fs. It took ~3 h of computational time using a PC with a 2.8 GHz processor.

5.1. *Deformation and failure mechanism under tensile loading*

Figure 9 shows snapshots of coupling simulation for the entire model at various stages of tensile testing at a strain rate of 44.88 ns^{-1} and figure 10 shows close-up views of figure 9. The snapshots of coupled MD/MPM simulation show that the tensile specimen experiences with increasing elongation in elasticity, generation of dislocations and plasticity by slip, void formation and propagation, formation of amorphous structure, necking, and final fracture in the MD region while only elastic deformation is present in the MPM regions at either end. In the initial stage, there is uniform stretching with elastic deformation and no dislocations occurring in the crystal lattice (see figure 11b). The limit of elasticity will be reached until the strain reaches approximately 0.1 (see figure 15). With further increase in deformation,

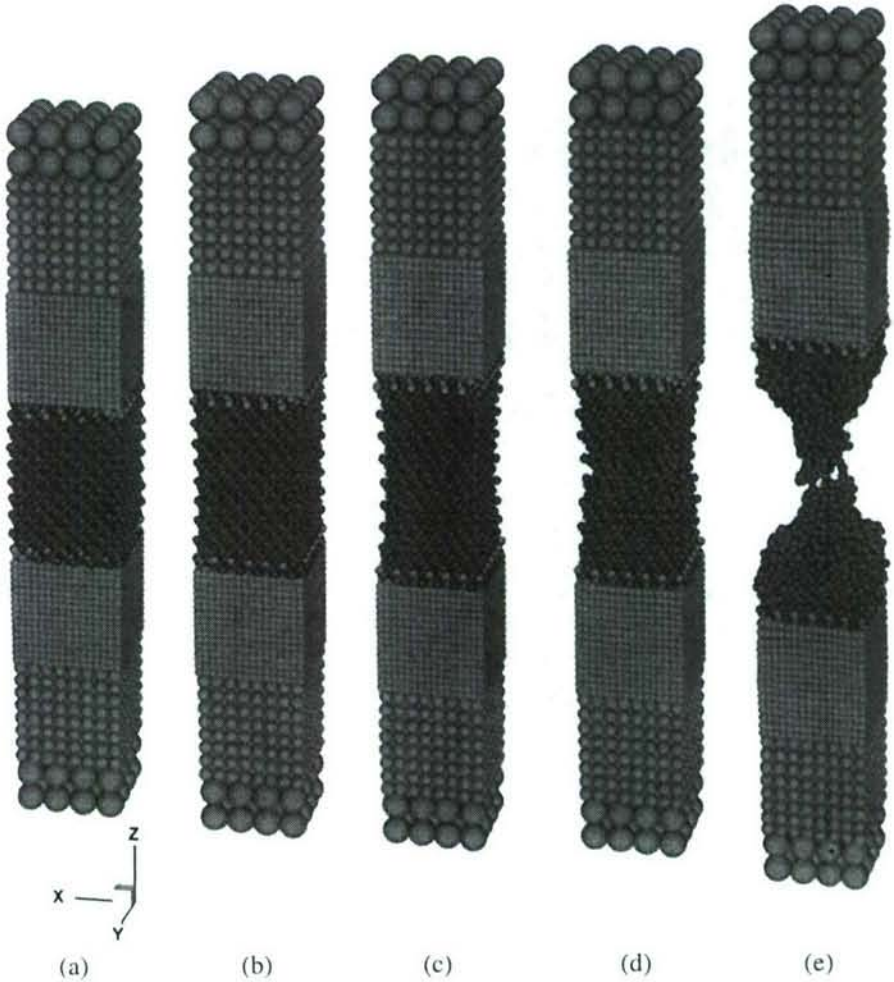


Figure 9. Snapshots of MD/MPM coupling simulation at a strain rate of 44.88 ns^{-1} and at different time steps (a) 0.01 ps, (b) 1.73 ps, (c) 3.05 ps, (d) 3.25 ps and (e) 5.5 ps.

Figure 10. Magnified view of the snapshots (figure 9) of MD/MPM coupling simulation at a strain rate of 44.88 ns^{-1} and at different time steps (a) 0.01 ps, (b) 1.73 ps, (c) 3.05 ps, (d) 3.25 ps and (e) 5.5 ps.

the dislocations are seen to be generated locally with their densities gradually increasing in the plastic deformation region. Slip takes place along [111] at about 45° to the loading direction (see figure 11c). Plasticity by slip will last until the formation of voids and semi-amorphous structure (see figure 11d). Ultimate tensile stress is reached at a strain of ~0.3 (see figure 15). Further increase in elongation results in local interactions or rearrangements of silicon atoms, resulting in somewhat amorphous structure (see figures 11e and 11f). The deformation is then concentrated in a narrow region resulting in necking (see figure 10d). Finally, separation of the tensile specimen occurs (see figure 10e).

Figures 10a–e are 3D snapshots of MD simulation at higher magnification (of figure 9) showing different stages of tensile testing at a strain rate of 44.88 ns^{-1} and at different time steps. Slip taking place at 45° to the loading direction can be clearly seen (figures 10b and 10c). The formation of amorphous structure in the central region and void formation can be clearly seen in figure 10d. Figure 10e shows concentration of amorphous structure in the central region followed by necking and separation. This corresponds to the steepest slope in figure 15 leading to final rupture of the sample.

Figures 11–14 show snapshots in the mid-section at different stages of tensile testing at different strain rates from 44.88 ns^{-1} to 89.76 ns^{-1} and at different time steps. Initially the deformation is predominantly elastic. This corresponds to steep elastic slope in figure 15. This is followed by slip on the preferred [111] direction. Further increase in strain leads to the formation of inhomogeneous structure leading to amorphization in the entire necked region (see atoms marked by circles in figure 11d). The similar amorphization can be seen in figures 12d, 13d and 14d. Ultimately, separation of the sample takes place with drastic reduction in stress. With increase in strain rate, the crystallinity of the silicon is maintained up to larger strains. Also, with increase in strain rate, the strain at fracture increases. It may be noted that necking that occurs before final failure for silicon is significantly less as

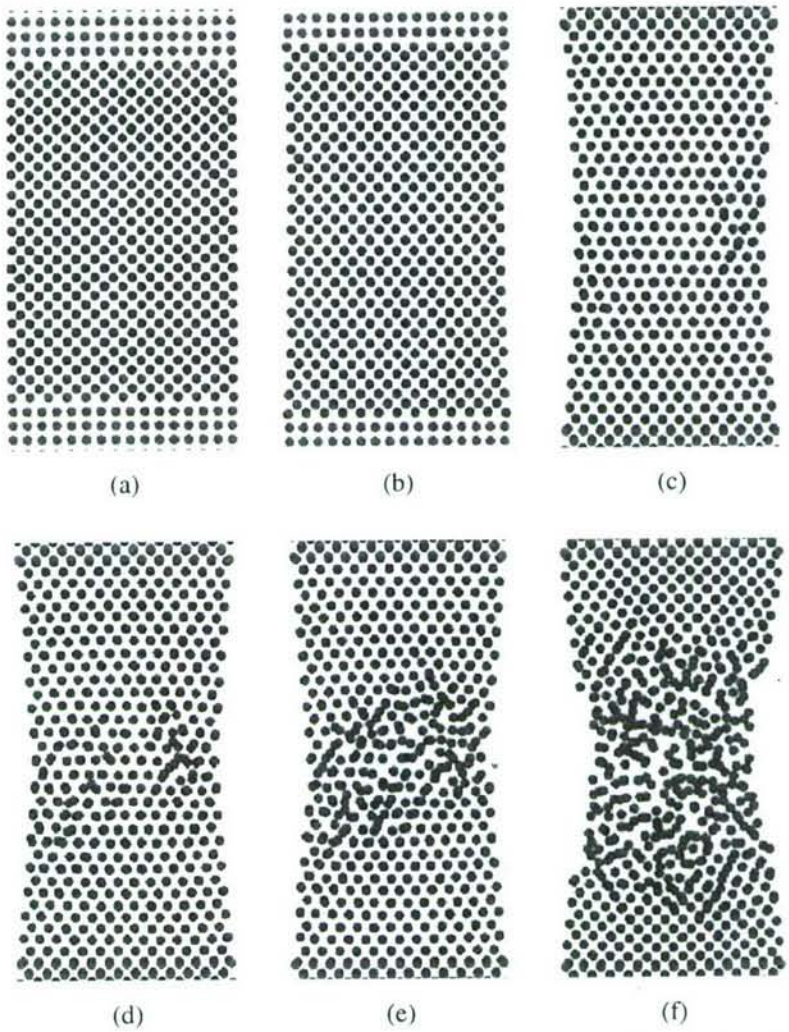


Figure 11. Snapshots of MD/MPM coupling simulation at a strain rate of 44.88 ns^{-1} at different time steps (a) 0.01 ps, (b) 1.52 ps, (c) 2.95 ps, (d) 3.05 ps, (e) 3.15 ps and (f) 3.56 ps.

compared to necking in ductile materials, such as copper, aluminium [24]. The catastrophic failure leads to a sudden drop in the stress–strain curves (see figure 15).

5.2. *Effects of strain rate on material properties*

The effects of strain rate on material properties have also been investigated for the coupled MD/MPM simulations. Under different strain rates, slip essentially takes place at about 45° with respect to the loading direction followed by the formation of amorphous structure and generation of voids in the necked region and their

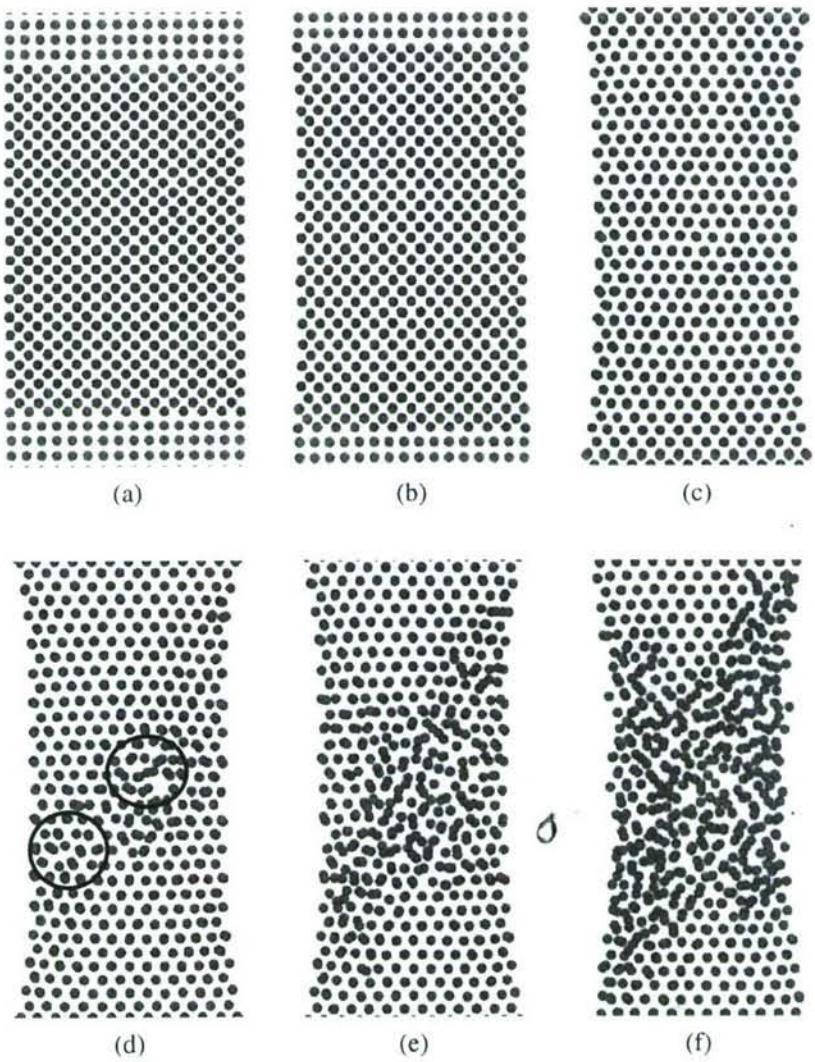


Figure 12. Snapshots of MD/MPM coupling simulation at a strain rate of $67.33\,ns^{-1}$ at different time steps (a) 0.01 ps, (b) 1.24 ps, (c) 2.58 ps, (d) 2.87 ps, (e) 2.95 ps and (f) 3.00 ps.

coalescence, leading to final failure. The engineering stress–strain curves at different strain rates are shown in figure 15. They show that material behaves differently at different stages including elastic and plastic deformation, and final rupture at different strain rates. Thus, the strain rate affects material properties significantly. However, before the strain reaches ~0.1, there is no dislocation generation and the initial part of engineering stress-strain curve is essentially elastic at different strain rates in the range from $44.88\,ns^{-1}$ to $89.76\,ns^{-1}$. From the slope of the stress strain curves, the Young's modulus is determined as ~121 GPa, which is very close to that obtained from the lattice theory [31]. As expected, the elastic modulus is nearly constant and is not affected by the strain rate.

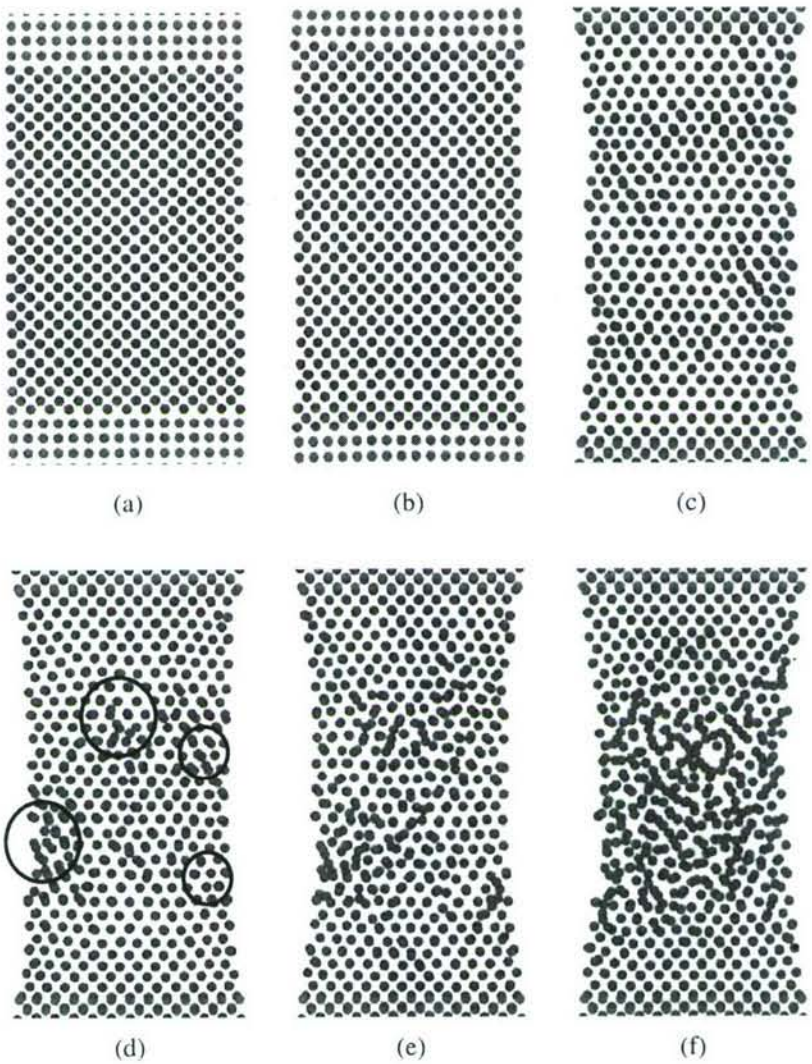


Figure 13. Snapshots of MD/MPM coupling simulation at a strain rate of 80.79 ns^{-1} at different time steps (a) 0.01 ps, (b) 1.26 ps, (c) 2.48 ps, (d) 2.79 ps, (e) 2.85 ps and (f) 2.91ps.

The curve in figure 16a shows the effect of the strain rate on the tensile strength for silicon. The tensile strength increases from ~15 to ~18 GPa with increase in the strain rate from 44.88 ns^{-1} to 89.76 ns^{-1}. Figure 16b shows the variation of the ultimate strain with strain rate. It shows that the ultimate strain increases rapidly up to ~0.54 as the strain rate increases from 44.88 ns^{-1} to 67.33 ns^{-1}, and then increases gradually up to ~0.57 as the strain rate increases from 67.33 ns^{-1} to 89.76 ns^{-1}.

The ultimate strain vs. the strain rate curve changes slope at a strain rate of ~67.33 ns^{-1}. Observation of the snapshots indicates that defect initiation and coalescence are somewhat different across the strain rate. At lower strain rates, i.e. below 67.33 ns^{-1} (e.g. at 44.88 ns^{-1}, as shown in figure 11d), voids are formed in the

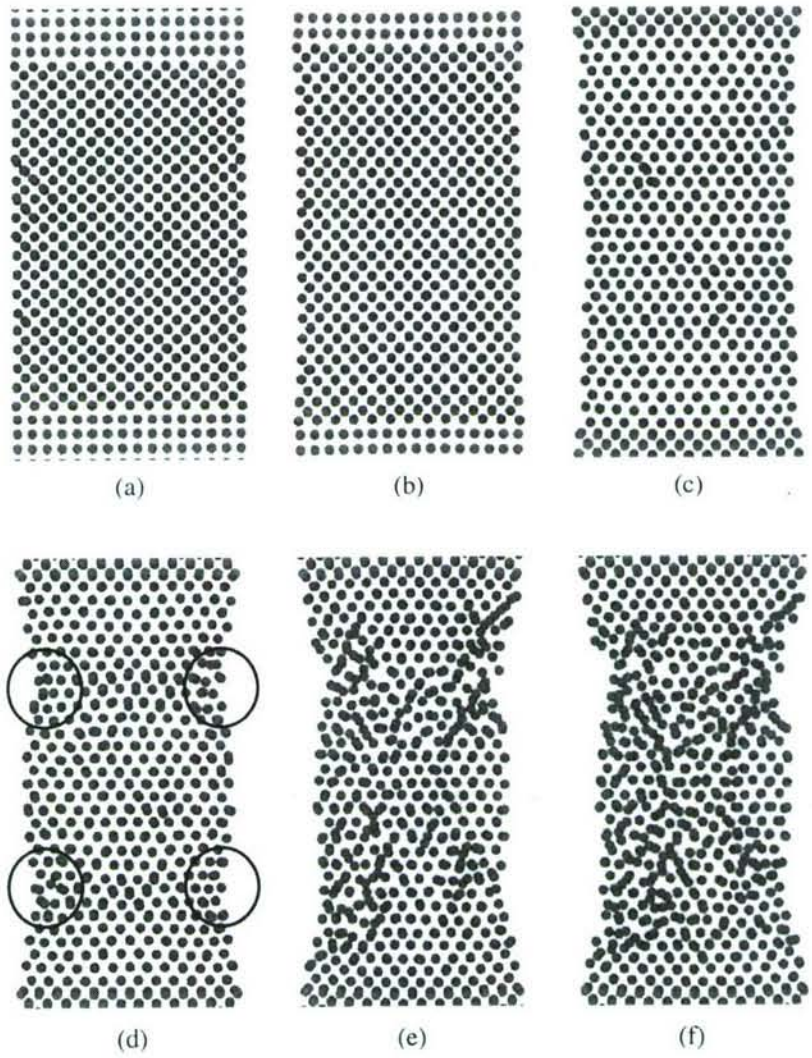


Figure 14. Snapshots of MD/MPM coupling simulation at a strain rate of 89.76 ns^{-1} at different time steps (a) 0.01 ps, (b) 1.29 ps, (c) 2.4 ps, (d) 2.434 ps (e) 2.78 ps and (f) 2.88 ps.

central region of the tensile specimen. These voids coalesce to larger defects resulting in necking and ultimate failure. While at higher strain rates (67.33 ns^{-1} or higher, e.g. at 89.76 ns^{-1}, as shown in figures 14d and 14e), defects are formed near the surface at about 1/4 and 3/4 of the MD region (see regions marked by circles in figure 14). These defects grow in size and cover the entire area, leading to decreased load bearing capacities at these locations, resulting in some bulging, as shown in figures 14d–e.

Compared with failure at the lower strain rates (e.g. figure 11d), there are more defect sites at higher strain rate (figure 14e). Once the defects at multiple sites coalesce, necking occurs, leading to a quick rupture of the material. The change in the slope of the ultimate failure strain shown in figure 16b is most likely the result of

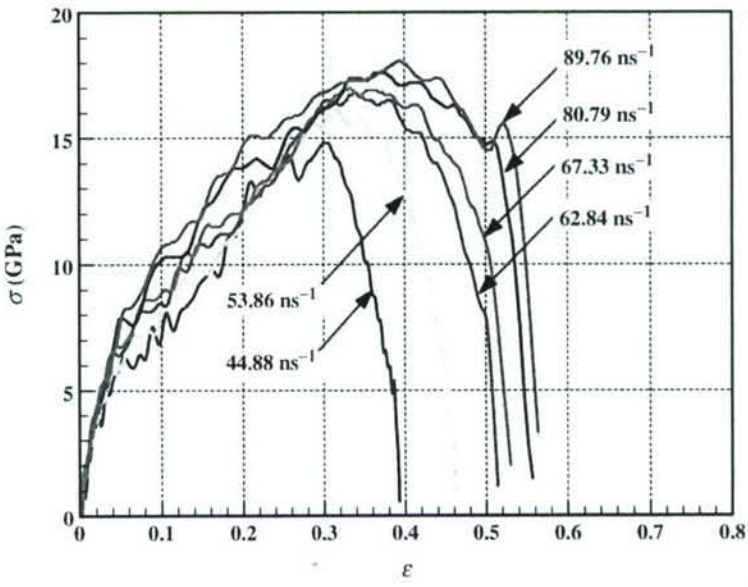


Figure 15. Engineering stress–strain curves from MD/MPM coupled simulations for silicon.

the change in defect formation mechanism from relatively low to high strain rates. However, it is not clear at this stage the underlying reason for a change in the defect formation mechanism across the strain rates. Further investigation is needed to explore this feature.

6. Conclusions

- A new computational approach has been developed for coupling molecular dynamics (MD) with material point method (MPM) for multiscale simulations from atomistic to continuum. In the MPM computation, the hierarchical mesh refinement technique is introduced to scale down from the continuum level to the atomistic level so that MPM can be coupled with MD. A Tersoff-type, three-body potential was used in MD computations for silicon. Seamless coupling has been achieved through a transition region between MD and MPM regions.
- Coupled MD/MPM simulations of uniaxial tension at multiscale were conducted on silicon at different strain rates. Numerical results show that the entire tension process involves elongation in the elastic region followed by dislocation generation and slip by plastic deformation, void formation and propagation, and finally necking and formation of an amorphous structure resulting in final fracture.
- From coupling simulations, engineering stress and strain curves for silicon were determined at different strain rates, and the effects of strain rate on material properties were also investigated. For silicon, both the tensile strength and the ultimate strain increase with increase in the strain rate.

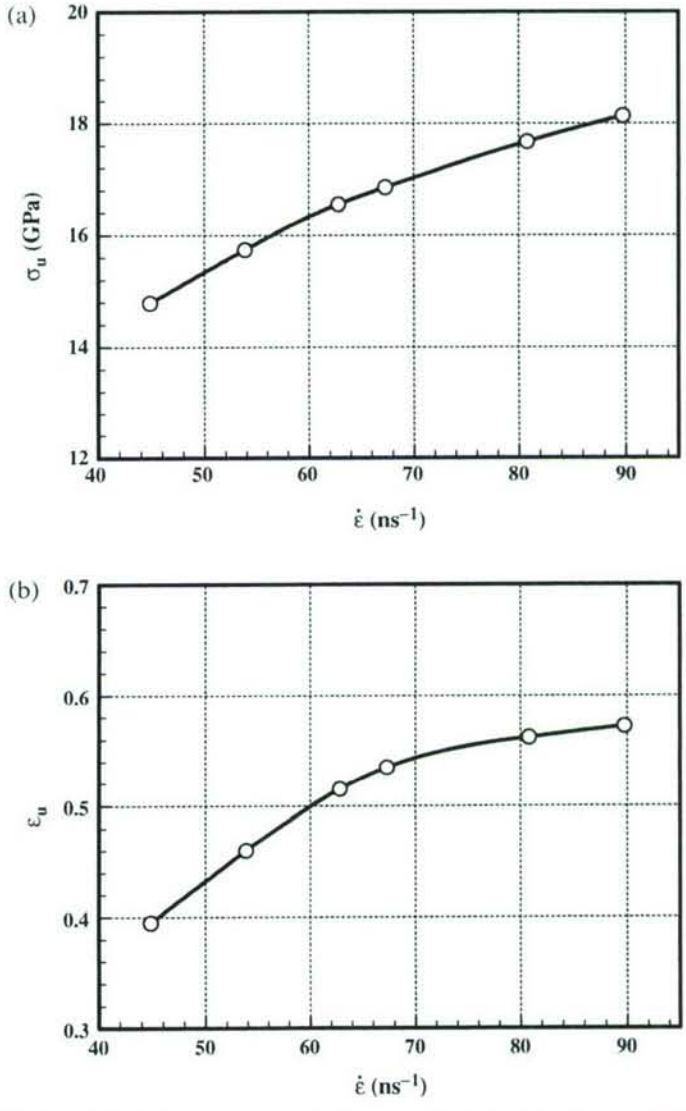


Figure 16. Effects of strain rate on material properties obtained from coupled MD/MPM simulations for silicon. (a) Tensile strength vs. strain rate. (b) Ultimate strain vs. strain rate.

Acknowledgements

The work was supported by a grant from the Air Force Office of Scientific Research (AFOSR) through a DEPSCoR grant (No. F49620-03-1-0281). The authors thank Dr. Craig S. Hartley and Dr. J. Tiley, Jr., Program Managers for the Metallic Materials Program at AFOSR for their interest and support of this work. One of the authors also thanks A. H. Nelson, Jr. Endowed Chair in Engineering for additional financial support.

References

[1] W.E. King, G. Campbell, T. Gonis, *et al.*, Mater. Sci. Engng A **191** 1 (1995).
[2] R. Phillips, Curr. Opinion Solid St. Mater. Sci. **3** 526 (1998).
[3] V.B. Shenoy, R. Miller, E.B. Tadmor, *et al.*, J. Mech. Phys. Solids **47** 611 (1999).
[4] J.A. Smirnova, L.V. Zhigilei and B.J. Garrison, Comput. Phys. Commun. **118** 11 (1999).
[5] G.S. Smith, E.B. Tadmor, N. Bernstein, *et al.*, Acta mater. **49** 4089 (2001).
[6] S. Ogata, E. Lidorikis, F. Shimojo, *et al.*, Comput. Phys. Commun. **138** 143 (2001).
[7] L.E. Shilkrot, W.A. Curtin and R.E. Miller, J. Mech. Phys. Solids **50** 2085 (2002).
[8] W.A. Curtin and R.E. Miller, Model. Simul. Mater. Sci. Engng **11** R33 (2003).
[9] E. Weinan and H. Zhongyi, J. Comput. Phys. **182** 234 (2002).
[10] R. Komanduri, H. Lu, S. Roy, *et al.*, paper presented at the AFoSR Metallic Materials Grantees Meeting, Wintergreen, VA (2004).
[11] B. Wang, N.P. Daphalapurkar, H. Lu, *et al.*, paper presented at the 2nd International Conference on Material Multiscale Modelling, Los Angeles (2004).
[12] R. Komanduri, unpublished results (2002).
[13] I.M. Hutchings, University of Cambridge (personal communication) (2002).
[14] S. Kohlhoff, P. Gumbsch and H. Fischmeister, Phil. Mag. A **64** 851 (1991).
[15] W. Hoover, A. De Groot and C. Hoover, Comput. Phys. **6** 155 (1992).
[16] W.A. Curtin and R.E. Miller, Model. Simul. Mater. Sci. Engng **11** R33 (2003).
[17] B. Shiari, R.E. Miller and W.A. Curtin, J. engng Mater. Technol. **127** 367 (2005).
[18] N.M. Ghoniem, E.P. Busso, N. Kioussis, *et al.*, Phil. Mag. **83** 3475 (2003).
[19] W.K. Liu, E.G. Karpov, S. Zhang, *et al.*, Comput. Meth. appl. mech. Engng **193** 1529 (2004).
[20] G. Lu and E. Kaxiras, in *Handbook of Theoretical and Computational Nanotechnology*, edited by M. Rieth and W. Schommers (American Scientific, 2004).
[21] D. Sulsky, S. Zhou and H. Schreyer, Comput. Phys. Commun. **87** 236 (1995).
[22] D. Sulsky and H. Schreyer, Comput. Meth. appl. mech. Engng **139** 409 (1996).
[23] H. Tan and J. Nairn, Comput. Meth. appl. mech. Engng **191** 2095 (2002).
[24] R. Komanduri, N. Chandrasekaran and L.M. Raff, Int. J. mech. Sci. **43** 2237 (2001).
[25] R. Komanduri, N. Chandrasekaran and L.M. Raff, J. Mater. Sci. Engng A **340** 58 (2003).
[26] J. Tersoff, Phys. Rev. B **37** 6991 (1988).
[27] J. Tersoff, Phys. Rev. B **39** 5566 (1989).
[28] S. Frankland, V. Harik, G. Odegard, *et al.*, Composites Sci. Technol. **63** 1655 (2003).
[29] W. Chang and T. Fang, J. Phys. Chem. Solids **64** 1279 (2003).
[30] O.C. Zienkiewicz and R.L. Taylor, *The Finite Element Method*, Vols. 1–2 [(McGraw-Hill, New York, 1989–1991).
[31] W.R. Runyan, *Concise Encyclopedia of Chemical Technology*, fourth edn (Wiley, New York, 1999), p. 996.

Mechanics of Advanced Materials and Structures, 12:471–484, 2005
Copyright © Taylor & Francis Inc.
ISSN: 1537-6494 print / 1537-6532 online
DOI: 10.1080/15376490500259293

Taylor & Francis
Taylor & Francis Group

Two-Dimensional Mixed Mode Crack Simulation Using the Material Point Method

B. Wang, V. Karuppiah, H. Lu, S. Roy, and R. Komanduri
School of Mechanical & Aerospace Engineering, Oklahoma State University, Stillwater, OK, USA

The material point method (MPM) has demonstrated its capabilities in the simulation of impact/contact/penetration and interfacial crack growth problems. Because of the use of material points in the description of a continuum, consistent with the particle description (atoms) using molecular dynamics (MD), it is natural to couple MPM with MD for simulation from atomistic to continuum levels. However, in addressing plane stress/plane strain problems, the MPM algorithm and simulation examples available in literature use a regular grid mesh with uniform square cells and enforce velocity and displacement continuities through its background grid, resulting in limitations in dealing with stress concentration, inclined dislocations and inclined crack, etc. In this article, an irregular mesh is implemented in MPM to eliminate the limitations resulting from the use of a regular mesh. The ray-crossing algorithm is employed to determine which cell a material point belongs to after deformation for interpolation and extrapolation of variables between material points and grid nodes. As an example to demonstrate the capability of the MPM using irregular mesh, the stress field in a continuum with an inclined crack is determined using arbitrary quadrilateral cells in the background grid mesh. The use of irregular mesh in MPM was validated by comparing MPM results with ABAQUS/Explicit simulation. The proposed method of using irregular mesh will be an essential element in using MPM to couple with atomistic scale simulation so that MPM can address inclined dislocations and cracks emanating from the atomistic simulation.

1. INTRODUCTION

Material point method (MPM) was pioneered by Sulsky et al. [1–4] to address dynamic problems in solid mechanics. It has demonstrated its capabilities in the simulation of impact/contact/penetration [3–6] and interfacial crack growth problems [7, 8]. In MPM, a material continuum is discretized into a finite collection of infinitesimal material points. Two descriptions are used in MPM—one based on a collection of material points (Lagrangian) and the other based on a computational background grid (Eulerian). The background rectangular grid is used solely for the calculations. The material points are followed throughout the deformation of a solid to provide a Lagrangian description and the field governing equations are solved at background grid nodes so that MPM is not subjected to mesh entanglement. As a result, MPM takes advantage of both Eulerian and Lagrangian descriptions to give it the capability of handling large deformations in a more natural manner so that mesh lockup problems present in FEM is avoided. Because MPM uses material points to represent a material continuum, and the material points are a description consistent with the particle description in molecular dynamics (MD), coupling between MPM and MD is more naturally compatible. Also, parallel computation is relatively straightforward because the use of a grid structure in MPM is consistent with parallel computing grids. Additionally, for problems involving contact [5, 6], MPM provides a no-slip contact algorithm to avoid the penetration between two bodies based on a common background mesh.

When MPM is coupled with MD for multiscale simulation, inclined dislocations, and cracks emanating from the MD simulation will need to be modeled by MPM at the continuum level. Conventional MPM usually uses regular grid mesh, which consists of uniform square cells in 2D case, leading to two limitations. One is that conventional MPM cannot address stress concentration issues. Another one is that conventional MPM cannot handle inclined crack problems. Based on conventional MPM, only horizontal or vertical crack can be generated. However, for most materials, there exists sliding after deformation so that inclined cracks are generated. Therefore, a new algorithm in MPM with irregular mesh needs to be developed to solve the above two issues. However, publications in the open literature on MPM have not addressed the issue of using irregular mesh directly. The objective of this article is to implement irregular mesh and develop a new algorithm within the framework of conventional MPM for addressing inclined dislocations and cracks for applications such as multiscale simulation using MPM/MD.

In this article, a method of using an irregular mesh in MPM is developed to enable the use of arbitrary quadrilateral cells in the background grid mesh for handling and inclined cracks and stress concentration. Transformation between local (natural)

Received 10 November 2004; accepted 13 July 2005.
Address correspondence to R. Komanduri, School of Mechanical & Aerospace Engineering, Oklahoma State University, Stillwater, OK 74078, USA; Phone: (405) 744–5900, Fax: (405) 744–7873. E-mail: ranga@ceat.okstate.edu

coordinates and global (irregular) coordinates used in the irregular mesh, similar to isoparametric elements in finite element is employed. MPM needs to identify the locations of material points with respect to cells. In the case of the use of irregular mesh, a ray crossing technique is used to identify whether the material points are within an irregular mesh or not. The use of irregular mesh is then implemented in the MPM algorithm as developed by Sulsky et al. [1–4]. Validation was first made with a tension problem using two MPM models (with regular and irregular meshes) through comparison with FEM results obtained from ABAQUS explicit code. An inclined crack problem is also solved as an example to demonstrate the capability of the arbitrary quadrilateral cells in the MPM algorithm. The stress and deformation fields for the tensile model with an inclined crack are determined as a function of time from dynamic MPM simulations. For mixed mode fracture, fracture parameters such as energy release rates and stress intensity factors are calculated using the virtual crack closure method and the displacement extrapolation method. The same problem is also solved using dynamic FEM to validate the method of using irregular mesh in MPM.

2. MATERIAL POINT METHOD

Material point method (MPM) was originally developed by Sulsky and Schreyer of the University of New Mexico and Chen of the University of Missouri, under the support of Sandia National Laboratories [1, 2] for addressing a range of solid mechanics problems. Figure 1 is a schematic of the MPM for 2D calculations in solid mechanics. The solid line is the outline of the body to be analyzed. The black dots are the material points. Each material point is given an initial mass consistent with the mass density and volume of the material. Material parameters such as mass, displacement, velocity, stress, strain, internal energy, and temperature are assigned to each material point according to the material it represents. As the numerical solution proceeds, the material points are tracked and their states updated so that they carry the complete solution. Information from the material points is transferred to background computational grid nodes. The continuum equations are discretized at grid nodes using standard finite difference or finite element approach. The solutions at grid nodes are then used to update the position and velocity of the material points. Strain increments computed on the grid are interpolated to the material points and used in conjunction with constitutive equations to update stress states for each material point.

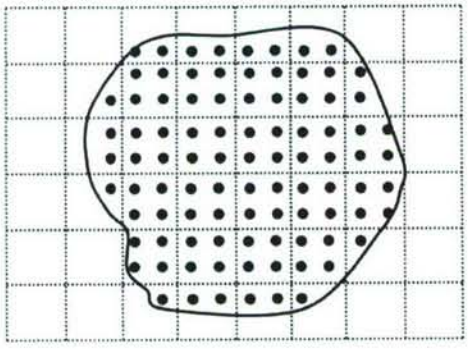


FIG. 1. Schematic of the MPM grid mesh showing cells with material points.

The governing equation in weak form is [9, 10]

$$\int_{\Omega_T} \mathbf{w} \cdot \boldsymbol{\tau}\, dS + \int_{\Omega} \rho \mathbf{w} \cdot \mathbf{b} d\Omega = \int_{\Omega} \rho \mathbf{w} \cdot \mathbf{a} d\Omega + \int_{\Omega} \rho \boldsymbol{\sigma}^s : \nabla \mathbf{w} d\Omega \tag{1}$$

where $\boldsymbol{\sigma}^s = \boldsymbol{\sigma}/\rho$ is the specific stress tensor, $\boldsymbol{\sigma}$ is the Cauchy stress tensor, $\mathbf{b}$ is the body force, $\mathbf{a}$ is the acceleration, and ρ the mass density. $\mathbf{w}$ is the test function which is an arbitrary spatial function. Differential volume and surface elements are denoted by $d\Omega$ and dS, respectively. Displacement boundary conditions are defined by $\mathbf{u} = \mathbf{g}$ on $\partial\Omega_u$, and traction boundary conditions are given by $\boldsymbol{\sigma} \cdot \mathbf{n} = \boldsymbol{\tau}$ on $\partial\Omega_T$, where $\mathbf{n}$ is the unit outward normal vector at the boundary surface and $\partial\Omega = \partial\Omega_u \cup \partial\Omega_T$.

A material continuum is divided into a finite collection of discrete infinitesimal regions $\Omega_p (p = 1, \ldots, N_p)$ called material points. Each material point is assigned a mass m_p in Ω_p, where $m_p = \int_{\Omega_p} \rho(x) d\Omega$ and $\Omega = \cup \Omega_p$. Mass density can then be approximated as a sum of material point masses using a Dirac delta function $\rho(\mathbf{x}, t) = \sum_{p=1}^{N_p} m_p \delta(\mathbf{x} - \mathbf{X}_p(t))$.

All variables represented by $\phi(\mathbf{x})$, such as the coordinates, displacements, velocities, and accelerations need to be transferred between grid nodes and material points using the shape functions [10],

$$\phi(\mathbf{x}) = \sum_{n=1}^{N_n} \phi_i N_i(\mathbf{x}) \tag{2}$$

where N_n is the number of nodes in the background grid and ϕ_i refers to the ith nodal value of $\phi(\mathbf{x})$.

The numerical solution will be obtained at a discrete set of times, t^k, where $k = 1, \ldots, K$. Based on Eq. (1) and the interpolation scheme in Eq. (2), the weak form of momentum equations reduces to

$$\sum_{j=1}^{n} m_{ij}^k \mathbf{a}_j^k = \mathbf{f}_i^{\text{int},k} + \mathbf{f}_i^{ext,k} \tag{3}$$

where the mass matrix is

$$m_{ij}^k = \sum_{p=1}^{N_p} m_p N_i(\mathbf{X}_p^k) N_j(\mathbf{X}_p^k), \tag{4}$$

the internal force and the external force on the node i are

$$\mathbf{f}_i^{\text{int}} = -\sum_{p=1}^{N_p} m_p \nabla N_i(\mathbf{X}_p^k) \cdot \sigma^s(\mathbf{X}_p^k, t^k) \quad \text{and}$$

$$\mathbf{f}_i^{ext} = \hat{\tau}_i^k + \sum_{p=1}^{N_p} m_p \mathbf{b}(\mathbf{X}_p^k, t^k) N_i(\mathbf{X}_p^k) \tag{5}$$

and the discrete applied traction is $\hat{\tau}_i^k = \int_{\partial\Omega_\tau} N_i(\mathbf{x})\tau(\mathbf{x}, t^k)dS$.

The grid nodal accelerations from Eq. (3) are then used to update the position, velocity, stress, strain, and temperature of the material points. In this investigation, the update-stress-last (USL) algorithm is used [11], i.e., the stress on the material points is updated at the end of each time step, using the strain increment calculated from the updated material point velocities interpolated to the grid.

3. THE USE OF IRREGULAR MESH IN MPM

Figure 2 shows an arbitrary quadrilateral cell with initial material points in the local (natural) and global coordinate systems in the MPM algorithm with an irregular mesh. At the initial time step, material points are uniformly distributed at quarter locations in the cell in terms of the local (natural) coordinate system. Each material point is assigned a quarter of the cell mass. Furthermore, it is necessary to determine which cell in the background grid mesh a material point belongs to after deformation so as to implement the interpolation and extrapolation of variables between material points and grid nodes. In this study, the ray-crossing algorithm [12] is employed to determine whether the material points are inside or outside of the arbitrary quadrilateral cells. Figure 2 also shows the principle of the ray-crossing algorithm for an arbitrary quadrilateral cell. To determine the status of a material point, we consider a horizontal ray emanating from this material point to the right. If the number of times this ray intersects with the sides of the polygon is even (including zero), the material point is outside of the polygon. If the number of intersections is odd, the material point lies inside the polygon. For example, rays emanating from sample material points, A and C, considered in Figure 2 intersect with the sides of polygon zero and two times, respectively, and are thus outside the polygon. Material point B intersects with the side one time and is thus inside the polygon.

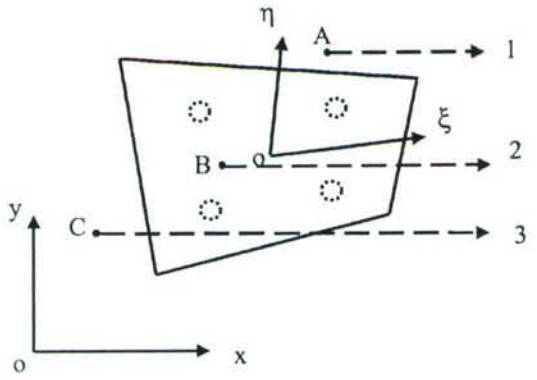


FIG. 2. An arbitrary quadrilateral cell with initial material points and schematic of the ray-crossing algorithm.

At each time step iteration in MPM computation, all variables, such as displacement, velocity, acceleration, strain, stress, and force, need to be transferred between grid nodes and material points using the shape functions. In this transformation, the local (natural) coordinates for updated positions of material points are required in the shape function used in Eq. (2) (see Eq. (7) for further details) after the particle locations in the grid cells have been determined. In a 2D situation, the shape function is of the same form as that used in the finite element analysis [10] and can be expressed as

$$N_i = \frac{1}{4}(1 + \xi_0)(1 + \eta_0) \quad \xi_0 = \xi\xi_i, \quad \eta_0 = \eta\eta_i \ (i = 1, 2, 3, 4) \tag{6}$$

and the local coordinates for four nodes of a cell $i(\xi_i, \eta_i)$ are $1(-1, -1); 2(1, -1); 3(1, 1); 4(-1, 1)$. The global coordinates of the material points (x_p, y_p) could be determined in terms of the global coordinates of cell nodes (x_i, y_i) which material points are associated with,

$$x_p = \sum_{n=1}^{N} x_i N_i(\xi_p, \eta_p) \quad y_p = \sum_{n=1}^{N} y_i N_i(\xi_p, \eta_p) \tag{7}$$

where (ξ_p, η_p) are the local (natural) coordinates of material points and $N = 4$ for four node cells. From Eqs. (6) and (7), we have

$$\begin{aligned} x_p &= \frac{1}{4}(1 - \xi_p)(1 - \eta_p)x_1 + \frac{1}{4}(1 + \xi_p)(1 - \eta_p)x_2 \\ &\quad + \frac{1}{4}(1 + \xi_p)(1 + \eta_p)x_3 + \frac{1}{4}(1 - \xi_p)(1 + \eta_p)x_4 \\ y_p &= \frac{1}{4}(1 - \xi_p)(1 - \eta_p)y_1 + \frac{1}{4}(1 + \xi_p)(1 - \eta_p)y_2 \\ &\quad + \frac{1}{4}(1 + \xi_p)(1 + \eta_p)y_3 + \frac{1}{4}(1 - \xi_p)(1 + \eta_p)y_4 \\ &(-1 \le \xi_p \le 1 \quad \text{and} \quad -1 \le \eta_p \le 1). \end{aligned} \tag{8}$$

Let

$$\begin{aligned} \alpha_1 &= -x_1 + x_2 + x_3 - x_4 & \beta_1 &= -y_1 + y_2 + y_3 - y_4 \\ \alpha_2 &= -x_1 - x_2 + x_3 + x_4 & \beta_2 &= -y_1 - y_2 + y_3 + y_4 \\ \alpha_3 &= x_1 - x_2 + x_3 - x_4 & \beta_3 &= y_1 - y_2 + y_3 - y_4 \\ \alpha_4 &= 4x_p - (x_1 + x_2 + x_3 + x_4) & \beta_4 &= 4y_p \\ & & & \quad -(y_1 + y_2 + y_3 + y_4) \end{aligned}$$

Eq. (8) becomes

$$\alpha_4 = \alpha_1\xi_p + \alpha_2\eta_p + \alpha_3\xi_p\eta_p \quad (9)$$

$$\beta_4 = \beta_1\xi_p + \beta_2\eta_p + \beta_3\xi_p\eta_p \quad (10)$$

From Eqs. (9, 10), we have

$$\xi_p = [\alpha_4\beta_3 - \beta_4\alpha_3 - (\beta_3\alpha_2 - \alpha_3\beta_2)\eta_p]/(\beta_3\alpha_1 - \alpha_3\beta_1) \quad (11)$$

and

$$[\alpha_3(\alpha_2\beta_3 - \beta_2\alpha_3)/(\beta_3\alpha_1 - \alpha_3\beta_1)]\eta_p^2 + [\alpha_1(\alpha_2\beta_3 - \beta_2\alpha_3)/(\beta_3\alpha_1 - \alpha_3\beta_1)_4\alpha_3)/ - \alpha_2 - \alpha_3(\alpha_4\beta_3 - \beta_4\alpha_3)/(\beta_3\alpha_1 - \alpha_3\beta_1)]\eta_p + [\alpha_4 - \alpha_1(\alpha_4\beta_1 - \beta_4\alpha_1)/(\beta_3\alpha_1 - \alpha_3\beta_1)] = 0 \quad (12)$$

Solving the above quadratic Eq. (12), two roots can be obtained. The absolute value of one of the roots η_{p1} and η_{p2} must be less than "1" and whichever value lies within "1" is taken as η_p. Then, the value of η_p is substituted into Eq. (11) to get ξ_p. Thus, the local (natural) coordinates for the position of material points are determined. The updated local coordinates will be used in Eqs. (2), (4) and (5) to implement the interpolation and extrapolation of all variables between grid nodes and material points.

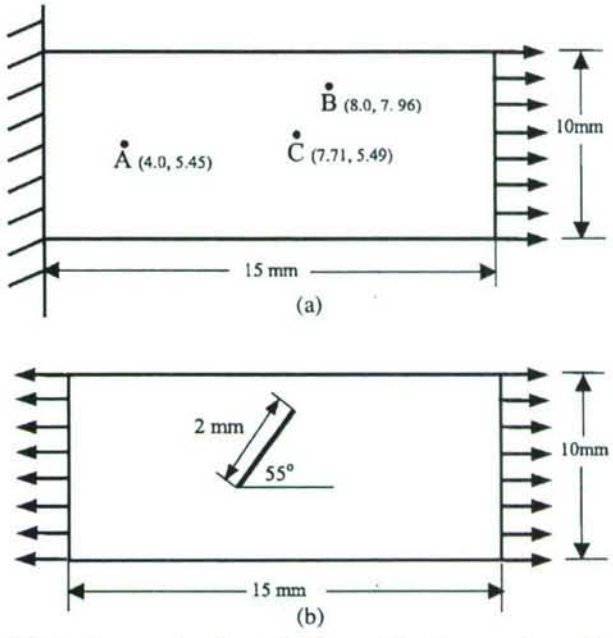


FIG. 4. Geometry, boundary and loading conditions for two tension models (uncracked and cracked plates).

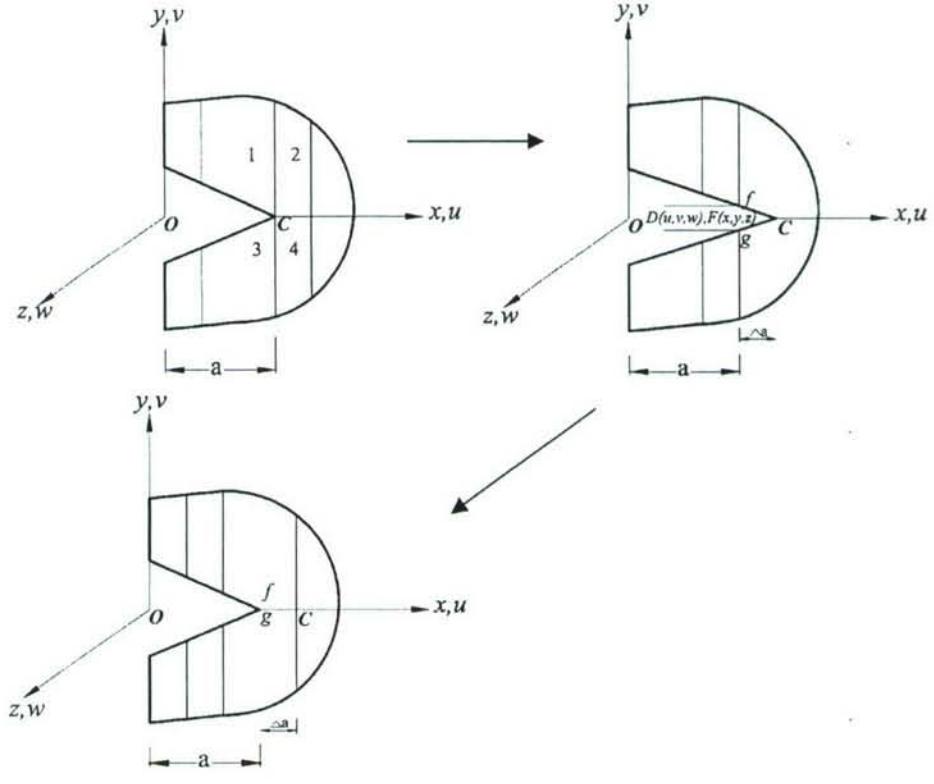
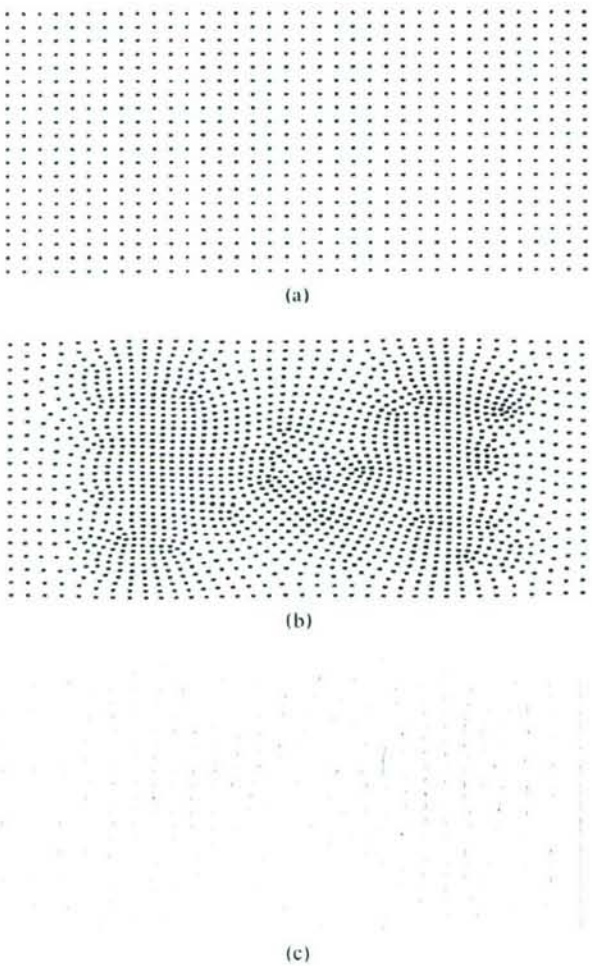


FIG. 3. Schematic of Irwin's crack closure method.

4. FRACTURE PARAMETERS FOR A MIXED MODE CRACK

To demonstrate the capability of the MPM with an irregular mesh, an inclined crack problem in mixed mode fracture is considered. The theoretical framework on the numerical analysis of mixed mode fracture is introduced in this section.

We next summarize some results on using MPM to model crack problems, as well as some work on mixed mode crack problems. Tan and Nairn [7, 8] recently applied MPM for modeling fracture. They utilized MPM to model a crack in a plane problem and computed dynamic energy release rates. They also considered the condition of no interaction of material points between two free surfaces of the crack by separating one line into two surfaces. Conventional MPM uses a regular grid mesh in which all MPM cells are uniform square ones [11, 13–20]. This leads to two major limitations: (1) It cannot address stress concentration problems as the adaptive mesh cannot be used in areas with high stress gradients; and (2) It cannot address inclined crack problems because the use of regular (uniform square) cells allows cracks only parallel and perpendicular to the grid. However, in reality, cracks can exist in any arbitrary orientation. While a single crack in a structure can be aligned with the coordinate axes, un-aligned multiple cracks and crack kinking cannot be all brought to align with the abscissa or the ordinate. To accommodate inclined cracks and to use a fine mesh near the crack and coarse mesh in the far field (for minimization of computational time while maintaining accuracy) an irregular mesh is needed in MPM, which is the subject of this article.

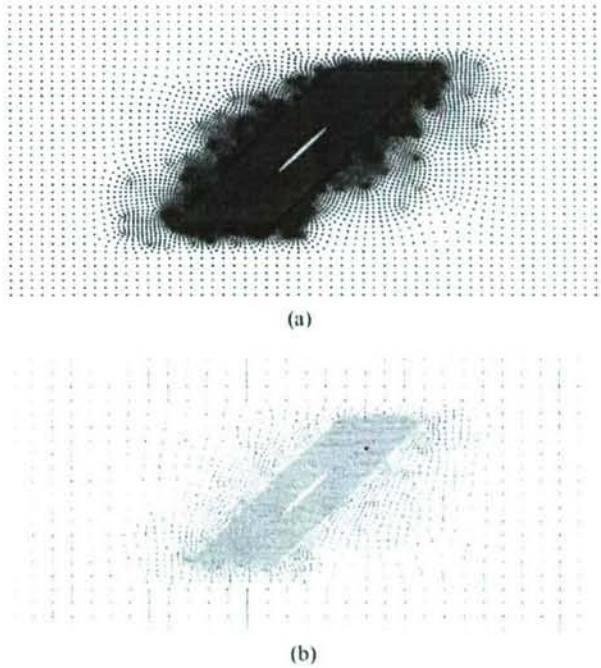


(a)

(b)

(c)

FIG. 5. MPM and FEM meshes for the tension model without a crack. (a) Regular MPM mesh, (b) Irregular MPM mesh (c) FEM mesh.



(a)

(b)

FIG. 6. MPM and FEM meshes for the tension model with an inclined crack. (a) MPM mesh with an inclined crack, (b) FEM mesh with an inclined crack.

Mixed mode fracture problems are commonly encountered in engineering [21–23]. Consequently, analysis of mixed mode

TABLE 1
Total time and time step increment for each model

Model	Total simulation time (μs)	Time step increment (ns)
MPM with regular mesh	5.1	14
MPM with irregular mesh	5.1	73
FEM	5.1	73
MPM with crack	5.1	14
FEM with crack	5.1	14

fracture problems is important in fracture mechanics for a better understanding and control of fracture leading to failure of materials. Cracks subjected to mixed mode loading can be attributed primarily to three factors: (1) mixed remote loading, i.e., remote normal and shearing forces acting on a component having a crack perpendicular to the normal loading direction, (2) deflected or inclined crack under normal/uniaxial remote loading, and (3) mechanical and/or thermal loads combined with arbitrary restraint conditions, producing a multiaxial loading condition on a crack. In computational fracture mechanics, there are at least three methods available for the computation of fracture parameters, energy release rate and the stress intensity factor (SIF) under mixed Mode I and Mode II loading conditions [24]. They are: (1) Irwin's virtual crack closure method, (2) the displacement extrapolation method, and (3) the direct J-integral computation using the equivalent domain integral together with a mode decomposition scheme. For applications of fracture mechanics in engineering structures, Dodds and coworkers [24, 25] obtained the stress intensity factors from corresponding J-integral computations. Bittencourt et al. [26] showed that for a sufficiently refined finite element meshes all three methods give essentially the same results for linear elastic materials. Usually, the virtual crack closure method can be used to calculate Mode I and Mode II energy release rates and the total energy release rate for mixed mode fracture using nodal forces and relative displacements at the crack tip. Among the three methods, the displacement extrapolation method is the most convenient method to calculate Mode I and Mode II stress intensity factors in terms of relative displacements and locations along free crack surfaces.

The energy release rate for the crack propagation can be calculated using the Irwin's virtual crack closure method [27, 28]. The principle behind Irwin's virtual crack closure technique is that if a crack is assumed to propagate an infinitesimal increment, the energy released during the crack growth will be equal to the work required to close this propagated crack to its initial crack size. Based on this assumption, the total energy release rate for a mixed crack growth increment Δa is given by

$$\begin{aligned} G &= \lim_{\Delta a \to 0} \frac{1}{2\Delta a} \int_0^{\Delta a} \sigma \cdot \Delta \vec{u} da \\ &= \lim_{\Delta a \to 0} \frac{1}{2\Delta a} \left\{ \int_0^{\Delta a} \sigma_{yy} \Delta v da + \int_0^{\Delta a} \sigma_{yx} \Delta u da \right. \\ &\quad \left. + \int_0^{\Delta a} \sigma_{yz} \Delta w da \right\} \end{aligned} \tag{13}$$

and for the 2D case, Mode I and Mode II energy release rates for a mixed crack fracture will be

$$\begin{aligned} G_I &= \lim_{\Delta a \to 0} \frac{1}{2\Delta a} \int_0^{\Delta a} \sigma_{yy} \Delta v da \\ \text{and} \quad G_{II} &= \lim_{\Delta a \to 0} \frac{1}{2\Delta a} \int_0^{\Delta a} \sigma_{yx} \Delta u da \end{aligned} \tag{14}$$

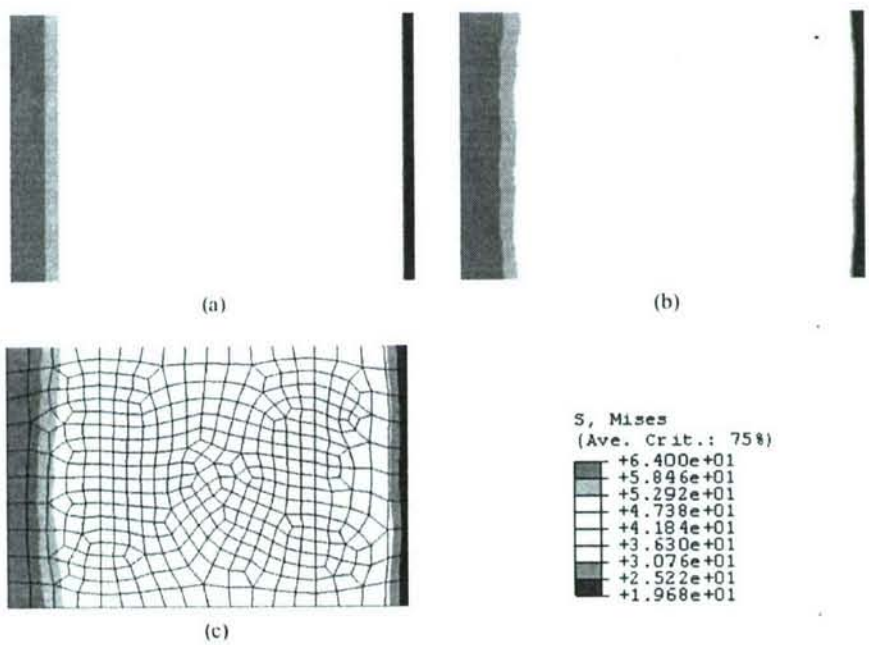


FIG. 7. von Mises stress contours for MPM (regular/irregular meshes) and FEM at $t = 5$ μs. (a) MPM model with regular mesh, (b) model with irregular mesh, (c) FEM model.

In numerical methods, the integral can be approximated using the nodal forces and nodal displacements at the crack tip. In Figure 3, the work required to close the propagated crack to its original crack size is given by

$$\Delta W = \frac{1}{2}[F_x(u_f - u_g) + F_y(v_f - v_g) + F_z(w_f - w_g)] \quad (15)$$

where u_f, u_g, v_f, v_g, w_f and w_g are displacement components and the subscripts of 'f' and 'g' denote locations belonging to the top and bottom surfaces at the crack tip, respectively, as shown in Figure 3.

Based on the definition of the energy release rate in Eq. (13), the total energy release rate can be expressed as

$$G = \frac{\Delta W}{\Delta A} \quad (16)$$

where ΔA is the crack area increment due to crack extension increment, Δa. Thus, the Irwin's virtual crack closure technique can be implemented in both dynamic FEM and MPM simulations. Generally, in the elastic regime there is a direct relationship between the stress intensity factor and the energy release rate. However, in the mixed mode situation this relation cannot be used easily. Therefore, the displacement extrapolation method is used to determine Mode I and Mode II stress intensity factors in this study. Elastic solutions for the displacements at and near the crack tip are used in this method. Paris and Sih [29] gave the displacements for linear elastic materials as

$$u = \frac{K_I}{4G}\sqrt{\frac{r}{2\pi}}\left[(2\chi - 1)\cos\frac{\theta}{2} - \cos\frac{3\theta}{2}\right] - \frac{K_{II}}{4G}\sqrt{\frac{r}{2\pi}}\left[(2\chi + 3)\sin\frac{\theta}{2} - \sin\frac{3\theta}{2}\right] + O(r) \quad (17)$$

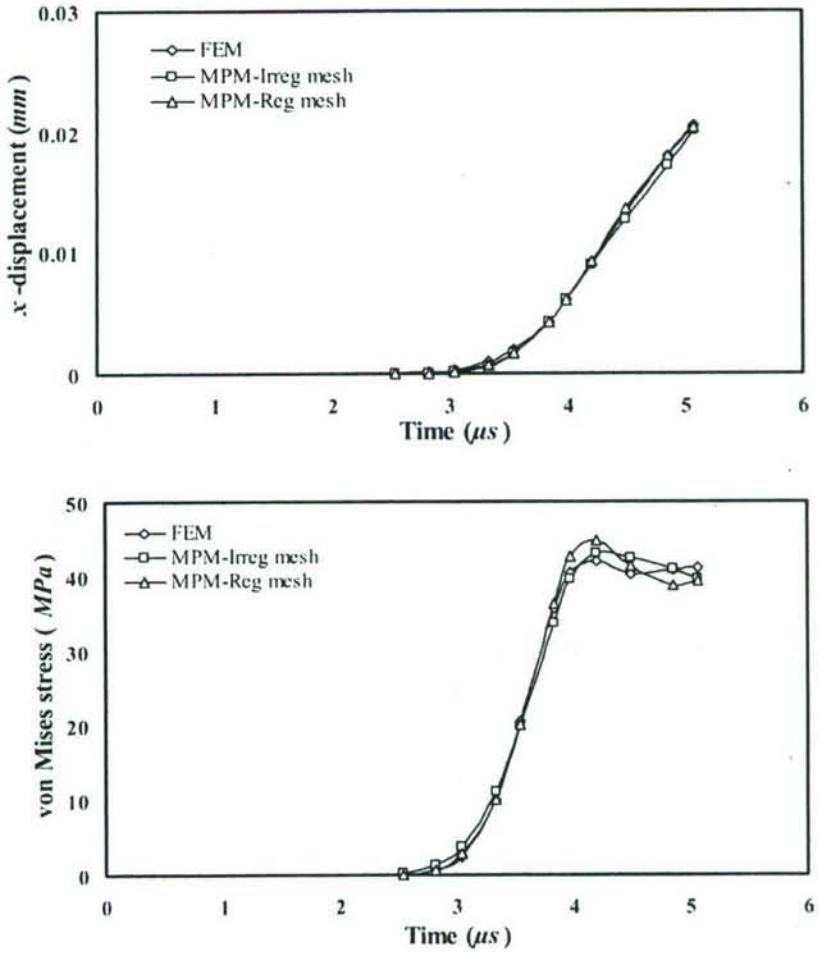


FIG. 8a. Comparison of the x-displacement and von Mises stress at location A (4.0 mm, 5.45 mm).

$$v = \frac{K_I}{4G}\sqrt{\frac{r}{2\pi}}\left[(2\chi - 1)\sin\frac{\theta}{2} - \sin\frac{3\theta}{2}\right] - \frac{K_{II}}{4G}\sqrt{\frac{r}{2\pi}}\left[(2\chi + 3)\cos\frac{\theta}{2} - \cos\frac{3\theta}{2}\right] + O(r) \quad (18)$$

$$w = \frac{2K_{III}}{G}\sqrt{\frac{r}{2\pi}}\sin\frac{\theta}{2} + O(r) \quad (19)$$

where u, v, and w are the local Cartesian displacements, (r, θ) are the local polar coordinates with its origin at the crack tip, G is the shear modulus, $\chi = 3 - 4\nu$ for plain strain or axi-symmetric conditions, $\chi = (3 - \nu)/(1 + \nu)$ for plane stress conditions, ν is the Poisson's ratio, and $O(r)$ represents terms of order r or higher.

If we neglect the higher order terms as $r \to 0$, and evaluate Eqs. (17–19) at $\theta = \pm 180°$, the displacements along the free crack faces are given as

$$u = \frac{K_{II}}{2G}\sqrt{\frac{r}{2\pi}}(1 + \chi) \quad (20)$$

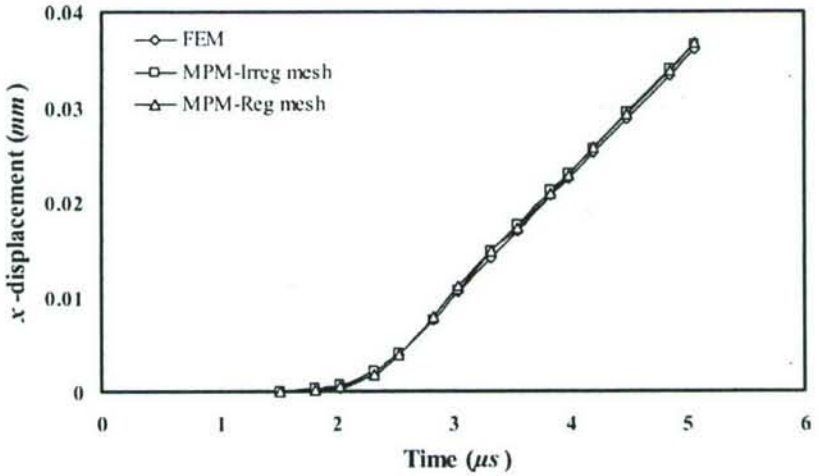
$$v = \frac{K_I}{2G}\sqrt{\frac{r}{2\pi}}(1 + \chi) \quad (21)$$

$$w = \frac{2K_{III}}{G}\sqrt{\frac{r}{2\pi}} \quad (22)$$

Equations (20–22) yield to

$$K_I = \sqrt{2\pi}\frac{G}{1 + \chi}\frac{|\Delta v|}{\sqrt{r}} \quad (23)$$

$$K_{II} = \sqrt{2\pi}\frac{G}{1 + \chi}\frac{|\Delta u|}{\sqrt{r}} \quad (24)$$



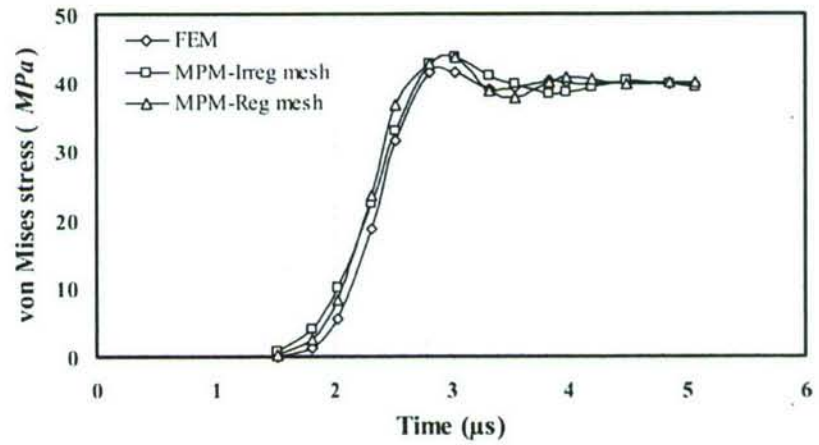


FIG. 8b. Comparison of the x-displacement and von Mises stress at location B (8.0 mm, 7.96 mm).

$$K_{III} = \sqrt{2\pi}\frac{G}{1+\chi}\frac{|\Delta w|}{\sqrt{r}} \tag{25}$$

where Δu, Δv, and Δw are the relative displacements of one crack face with respect to the other. $\frac{|\Delta v|}{\sqrt{r}}$, $\frac{|\Delta u|}{\sqrt{r}}$, and $\frac{|\Delta w|}{\sqrt{r}}$ can be calculated based on the nodal displacements and locations along free surfaces of the crack determined from the MPM computation, and subsequently K_I, K_{II}, and K_{III} are obtained using Eq. (25).

5. MPM AND FEM MODELING

Dynamic MPM and FEM analyses are conducted on two kinds of tension models (uncracked and cracked plates). (1) First, to validate the method of using an irregular mesh in MPM, a tension model without a crack is simulated using both regular and irregular MPM meshes, as well as an FEM model by ABAQUS/Explicit code. The geometry and boundary conditions of the tension model are shown in Figure 4a. A tensile step load of 400 N is applied at the right end and the left end is fixed. For the detailed comparison of MPM and FEM results, three specific locations A, B, and C are chosen in the sample to validate the MPM algorithm, shown in Figure 4a. Point A is chosen far away from points B and C along longitudinal direction to demonstrate the stress wave propagation. Points B and C are located close together in the x-direction but are separated in the y-direction to verify the uniformity of the von Mises stress distribution along the cross-section (2) To demonstrate the capability of the MPM with irregular mesh, another tension model with an inclined crack, shown in Figure 4b, is simulated using both the MPM algorithm and FEM with an adaptive mesh. Both ends of the plate are subjected to tensile loads of 2, 000 N. An inclined crack of 2 mm length oriented at an angle of 55° is located at the center of the plate. For the above two uncracked and cracked tension models, a linear elastic, isotropic, and homogeneous material is considered in this study.

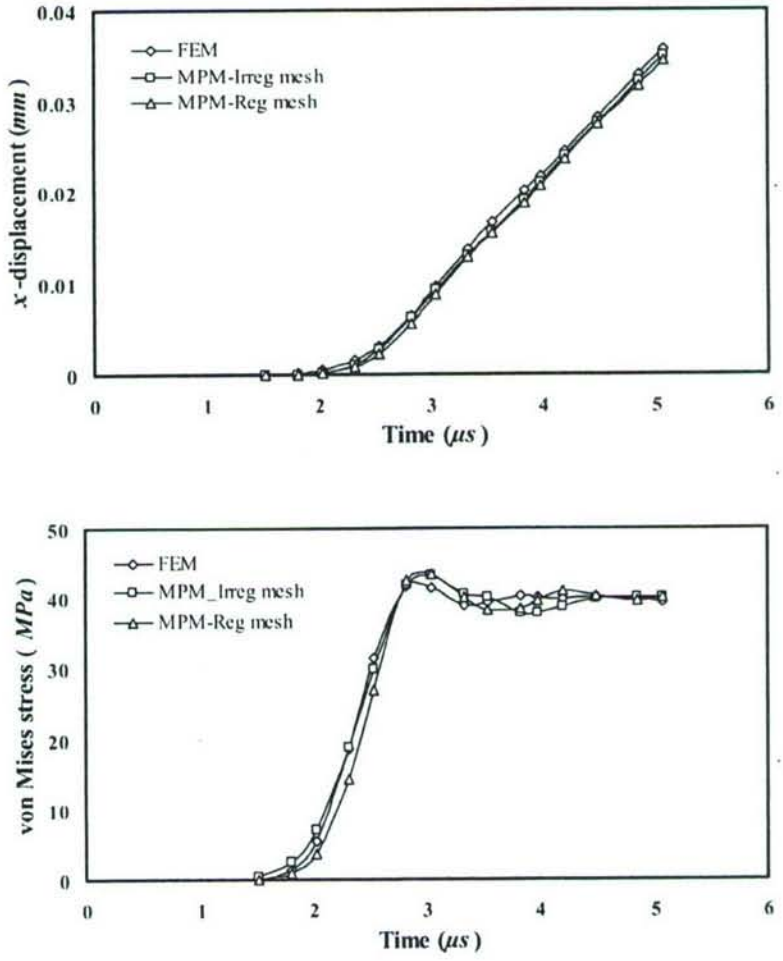


FIG. 8c. Comparison of the x-displacement and von Mises stress at location C (7.71 mm, 5.49 mm).

The linear elastic properties of the material are taken as: mass density $\rho = 1$ gm/cm^3, Young's modulus $E = 10000$ MPa, and Poisson's ratio $\nu = 0.001$ [16]. In both MPM and FEM models, free surfaces of the crack are introduced using an approach that allows the placement of two sets of nodes at the same locations along the crack line to avoid the interaction between neighboring cells/elements in the two sides of the crack line, as shown in Figures 5 and 6. To allow the direct comparison between MPM and FEM results, meshes used in both models are identical and are shown in Figures 5 and 6. To generate the MPM and FEM meshes, I-Deas software is used for preprocessing and ABAQUS/Explicit is used for dynamic FEM analysis. To characterize the stress singularity near the crack tip, very fine mesh is used around the crack and coarse mesh is used in the far field to minimize the computation time and to raise the computation capacity. This adaptive mesh will also improve the accuracy of fracture parameter calculations. During the entire computation in MPM, the background grid mesh never changes. Thus, material points can move in or out of the cells after deformation.

Figures 5 and 6 show MPM and FEM meshes for the tension models without a crack and with an inclined crack, respectively. For the uncracked plate, there are 600 material points in the MPM model with the regular mesh (Figure 5a) and 1,616 material points in the MPM model with irregular mesh (Figure 5b). In the FEM model, there are 433 nodes and 404 elements (Figure 5c). For the tension plate with an inclined crack, the MPM model with an irregular mesh has 14,628 material points (Figure 6a) and FEM model has 3,767 nodes and 3,697 elements (Figure 6b). For explicit dynamic MPM/FEM analysis, the stable time step increment depends on the length of

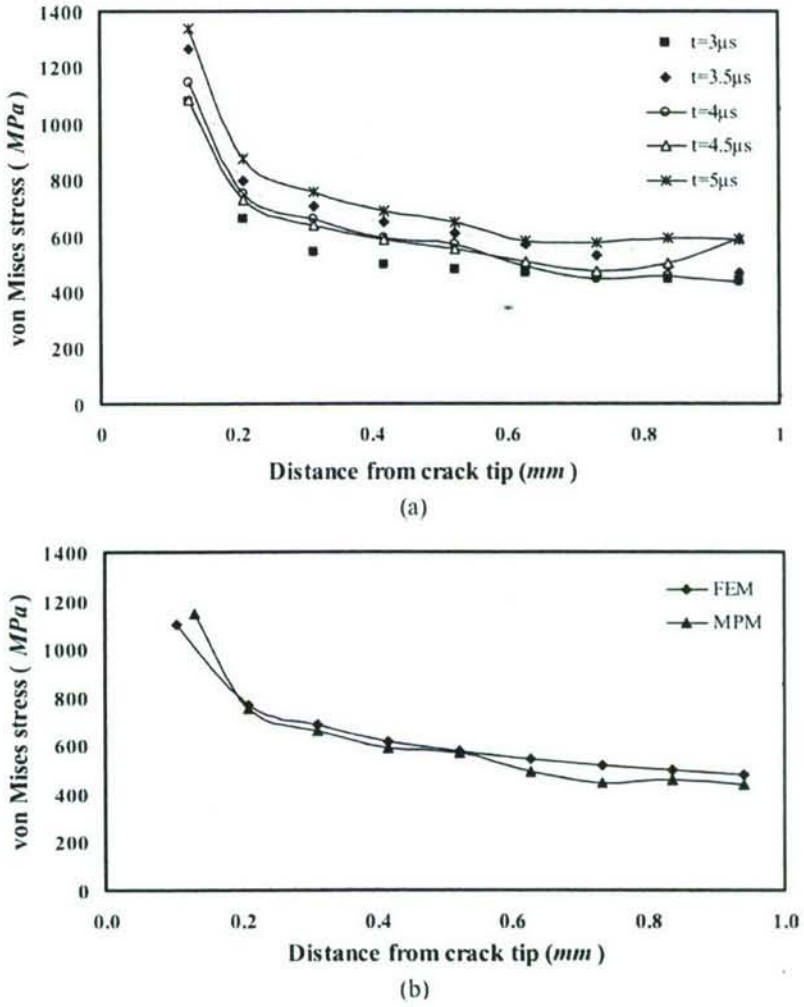
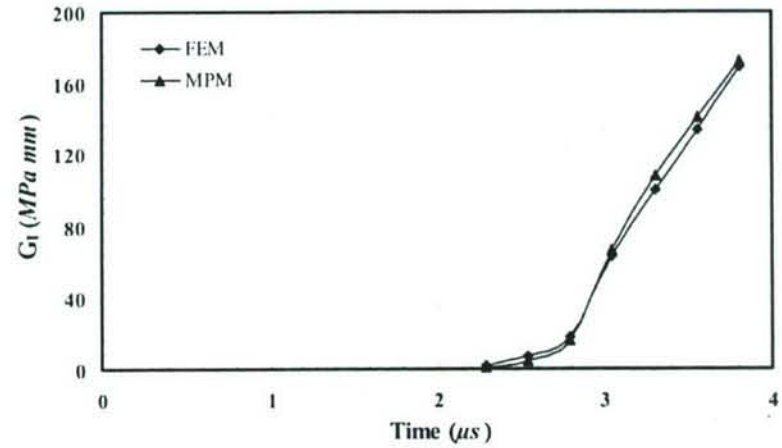
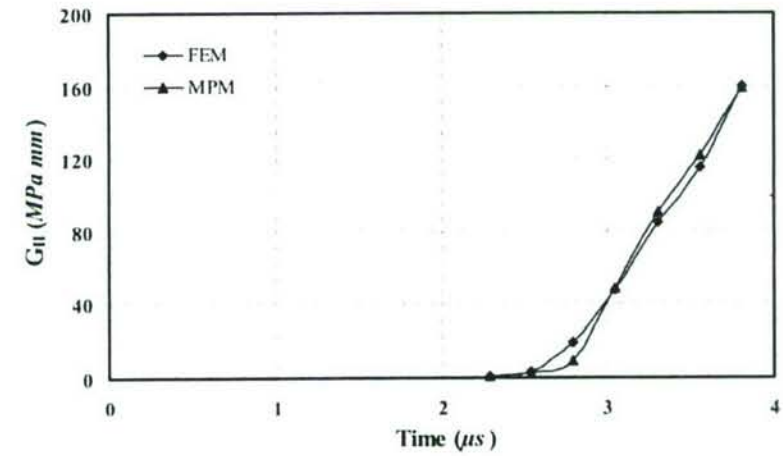


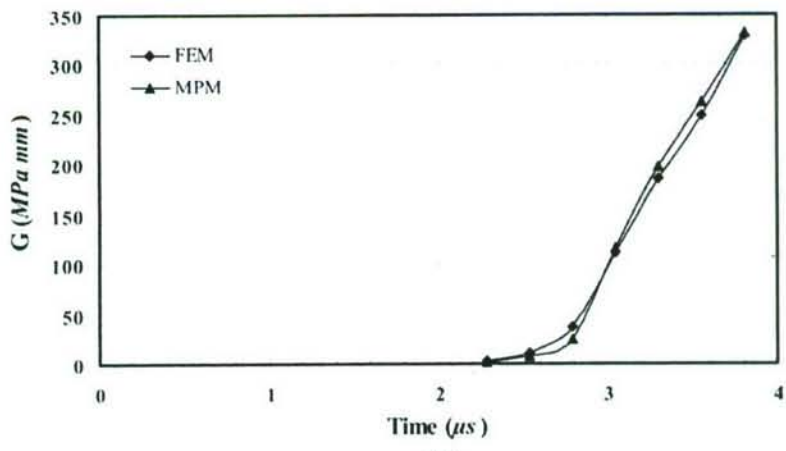
FIG. 9. von Mises stress distributions along inclined crack line near the crack tip. (a) von Mises stress distributions along inclined crack line near crack tip at different times using irregular mesh MPM analysis (b) Comparison of von Mises stress distribution along crack line near crack tip at $t = 4$ μs.



(a)



(b)



(c)

FIG. 10. (a) Mode I: Variation of energy release rate with time. (b) Mode II: Variation of energy release rate with time. (c) Variation of the total energy release rate versus time.

the cell/element and the longitudinal bar wave velocity, and the minimum time step increment was used to carry out the numerical simulations. Table 1 lists details of the total simulation time and the time step increment for each model. For uncracked tension model, a uniform force was applied at one end of the plate and the other end is constrained only along the x-direction. For cracked model, uniform forces are applied at both ends of the plate.

6. RESULTS AND DISCUSSION

Explicit dynamic analyses were conducted on the uncracked tensile specimen using two MPM models (regular and irregular meshes) and one FEM model. After running dynamic MPM and FEM simulations, displacement and stress fields with the time were obtained for these three tension models. As an equivalent scalar stress measure, von Mises stress was selected to represent the stress field ahead of the crack tip, since a mixed mode crack was introduced in the model. Figure 7 shows von Mises stress contours at t = 5 μs for the two MPM models (regular and irregular meshes) and the FEM model. The results show that the overall profiles of three dynamic stress contours are fairly consistent. Figure 8 gives a comparison of the curves for the displacements along the x-direction and von Mises stresses as a function of time at specific locations A, B, and C respectively, in the specimen as indicated in Figure 4a. Simulation results in Figure 8 also show that the stress wave travels from right to left with a longitudinal bar wave velocity c = 3,162 m/s for all three models. After the stress wave arrives at these three specific locations, material points there start moving and the von Mises

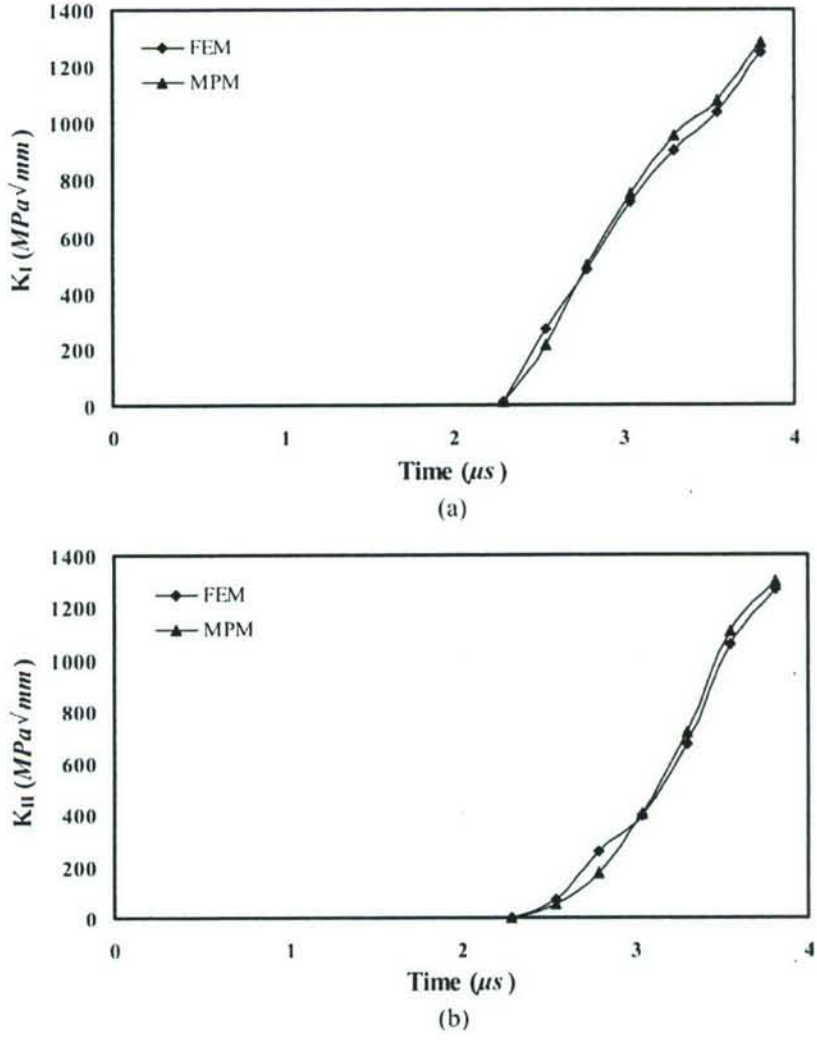


FIG. 11. (a) Mode I: Variation of stress intensity factor with time. (b) Mode II: Variation of stress intensity factor with time.

stress rapidly increases and reaches very close to the theoretical equilibrium value σ=40 MPa. It can be seen that all results for the three models in Figure 7 and Figure 8 are in good agreement indicating the validity of the MPM with an irregular grid mesh for the uncracked specimen.

For the tension model with an inclined crack in MPM analysis, the von Mises stress distributions along the inclined crack line near the crack tip at different times are shown in Figure 9a. In this figure, a high stress gradient exists due to a stress singularity near the crack tip as expected. Figure 9b depicts a comparison of irregular mesh MPM and FEM results at $t = 4\ \mu s$, indicating that the results of both MPM and FEM are in very good agreement for this example. A comparison with FEM shows that the MPM with an irregular mesh is able to accommodate the inclined crack issues accurately and implement explicit fracture mechanics computations. An adaptive mesh is used for simulating the crack in the MPM model to improve the accuracy of fracture parameter calculations. Using the virtual crack closure method, the dynamic energy release rates in Mode I and Mode II are calculated and shown in Figure 10. It can be seen from Figure 10 that MPM results are in good agreement with dynamic explicit FEM results. These MPM and FEM results indicate that the stress waves propagate from both ends to the center. After the stress wave arrives near the crack, the crack gradually starts opening and the energy release rates increase with time. It also shows that contributions from both Mode I and Mode II energy release rates are significant for the inclined crack problem, as one might expect. Additionally, the displacement extrapolation method is used to determine the Mode I and Mode II stress intensity factors for MPM and FEM models. Comparisons of the stress intensity factors as a function of time are shown in Figures 11a and 11b, indicating that irregular mesh MPM and FEM results agree very well. Based on the good agreement between MPM and FEM models with an inclined crack, it is evident that the proposed MPM algorithm with an irregular grid mesh can be used to simulate mixed mode crack problems and for implementing explicit dynamic fracture mechanics computations.

7. CONCLUSIONS

A methodology for implementing an irregular mesh in MPM has been developed to enable the use of arbitrary quadrilateral cells in MPM to model the inclined dislocations and cracks for use of multiscale simulation using MPM/MD. The use of an arbitrary quadrilateral MPM cells also makes it convenient for cell refinement in areas with high stress gradients. Dynamic MPM simulations of an uncracked tension model indicate that results from both regular and irregular meshes are consistent and agree well with the FEM results using ABAQUS/Explicit code, providing validation for the MPM with the use of irregular mesh. For mixed mode fracture analysis, a tension model with an inclined crack was also simulated using the MPM algorithm with an adaptive mesh. The adaptive mesh provides accurate fracture parameter calculations. From dynamic MPM simulations, the Mode I and Mode II energy release rates were calculated and found to be in good agreement with those obtained from dynamic FEM analysis. Mode I and Mode II stress intensity factors were also calculated from dynamic MPM simulations. Compared with FEM, MPM takes advantage of both Eulerian and Lagrangian descriptions to give it the capability of handling large deformations in a more natural manner so that locking problems present in FEM [9, 30] is avoided. Additionally, because MPM uses discrete material points to represent a material continuum, and because material points are a description consistent with the particle description (atoms) in molecular dynamics (MD), coupling between MPM and MD is compatible.

ACKNOWLEDGEMENTS

The project is supported by a grant from the Air Force Office of Scientific Research (AFOSR) through a DEPSCoR grant (No. F49620-03-1-0281). The authors are grateful to Dr. Craig S. Hartley, Program Manager, Metallic Materials Program at AFOSR for his interest and support of this work. One of the authors (B. W.) also acknowledges valuable discussions with Dr. H. Tan.

REFERENCES

1. D. Sulsky, Z. Chen, and H. L. Schreyer, A Particle Method for History-Dependent Materials, *Comput Methods in Appl. Mech. and Engng.*, vol. 118, pp. 179–196, 1994.
2. D. Sulsky, S. J. Zhou, and H. L. Schreyer, Application of a Particle-in-Cell Method to Solid Mechanics, *Computer Physics Communications*, vol. 87, pp. 236–252, 1995.
3. D. Sulsky and H. L. Schreyer, Axisymmetric Form of the Material Point Method with Applications to Upsetting and Taylor Impact Problems, *Comput Methods in Appl. Mech. and Engng.*, vol. 139, pp. 409–429, 1996.
4. Y. Shen, W. Li, D. Sulsky, and H. L. Schreyer, Localization of Plastic Deformation Along Grain Boundaries in a Hardening Material, *Int. J. of Mech. Sci.*, vol. 42, pp. 2167–2189, 2000.
5. S. G. Bardenhagen, J. U. Brackbill, and D. Sulsky, The Material Point Method for Granular Materials, *Comput Methods in Appl. Mech. and Engng.* vol. 187, pp. 529–541, 2000.
6. S. G. Bardenhagen, J. E. Guilkey, K. M. Roessig, J. U. Brackbill, W. M. Witzel, and J. C. Foster, An Improved Contact Algorithm for the Material Point Method and Aapplication to Stress Propagation in Granular Material, *Comp. Model Engng. Sci.*, vol. 2 pp. 509–522, 2001.
7. H. Tan, and J. A. Nairn. Hierarchical, Adaptive, Material Point Method for Dynamic Energy Release Rate Calculations., *Comput Methods in Appl. Mech. and Engng.*, vol. 191, pp. 2095–2109, 2002.
8. J. A. Nairn, Material Point Method Calculations with Explicit Cracks, *Tech Science Press*, CMES 2003.
9. J. N. Reddy, *An Introduction to the Finite Element Methods*, (3rd ed.), McGraw-Hill, New York, 2006.
10. J. N. Reddy, *An Introduction to Nonlinear Finite Element Analysis*, Oxford University Press, Oxford, UK, 2004.
11. S. G. Bardenhagen, Energy Conservation Error in the Material Point Method for Solid Mechanics, *J. of Comput. Phys.*, vol. 180, pp. 383–403, 2002.
12. E. Haines, *Point in Polygon Strategies in Graphic Gems IV*. AP Professional, Boston P. Heckbert (ed.), pp. 24–46, 1994.
13. S. J. Cummins and J. U. Brackbill. An Implicit Particle-in-Cell Method for Granular Materials, *J. Comput. Phys.*, vol. 180, pp. 506–548, 2002.

14. Z. Chen, W. Hu, L. Shen, X. Xin, and R. Brannon. An Evaluation of the MPM for Simulating Dynamic Failure with Damage Diffusion, *Engng. Fract. Mech.*, vol. 69, pp. 1873–1890, 2002.
15. H. L. Schreyer, D. L. Sulsky, and S. J. Zhou, Modeling Delamination as a Strong Discontinuity with the Material Point Method, *Comput. Methods in Appl. Mech. and Engng.*, vol. 191, pp. 2483–2507, 2002.
16. Z. Chen, and R. Brannon. An Evaluation of the Material Point Method, SAND Report. SAND2002-0482, February 2002.
17. W. Hu and Z. Chen, A Multi-mesh MPM for Simulating the Meshing Process of Spur Gears, *Comp. and Struct.*, vol. 81, pp. 1991–2002, 2003.
18. A. R. York, D. L. Sulsky, and H. L. Schreyer, The Material Point Method for Simulation of Thin Membranes, *Int. J. Numer. Engng.*, vol. 44, pp. 1429–1456, 1999.
19. D. Sulsky and A. Kaul, Implicit Dynamics in the Material-point Method. *Comput. Methods In Appl. Mech. And Engng.*, vol. 193, pp. 1137–1170, 2003.
20. J. E. Guilkey and J. A. Weiss, Implicit Time Integration for the Material Point Method: Quantitative and Algorithmic Comparisons with the Finite Element Method, *Int. J. Numer. Meth. Engng.*, vol. 57, pp. 1323–1328, 2003.
21. C. Ma and S. Chen, Dynamic Stress Intensity Factor for Subsurface Inclined Cracks, *J. of Engng. Mech.*, vol. 120, pp. 483–498, 1994.
22. F. Buchholz, A. Chergui, and H. Richard, Fracture Analyses and Experimental Results of Crack Growth Under General Mixed Mode Loading Conditions, *Engng. Fract. Mech.*, vol. 71, pp. 455–468, 2004.
23. A. Miranda, M. Meggiolaro, J. Castro, and L. Martha, Fatigue Life Prediction of Ccomplex 2D Components Under Mixed-Mode Variable Amplitude Loading, *Int. J. of Fatig.* vol. 25, pp. 1157–1167, 2003.
24. M. C. Walters, G. H. Paulino, and R. H. Dodds, Stress-intensity Factors for Surface Cracks in Functionally Graded Materials under Mode-I Thermomechanical Loading, *Int. J. of Solids and Struct.*, vol. 41, pp. 1081–1118, 2004.
25. C. G. Matos, and R. H. Dodds, Modeling the Effect of Residual Stresses on Defects in Welds of Steel frame Connections, *Engng. Struct.*, vol. 22, pp. 1103–1120, 2000.
26. T. Bittencourt, P. Wawrzynek, A. Ingraffea, and J. Sousa, Quasi-automatic Simulation of Crack Propagation for 2D LEFM Problems, *Engng. Fract. Mech.*, vol. 55, pp. 321–334, 1996.
27. H. Liebowitz and E. Moyer, Finite Element Methods in Fracture Mechanics, *Comput. Struct.*, vol. 31, pp. 1–9, 1989.
28. E. Rybicki and M. Kanninen, A Finite Element Calculation of Stress Intensity Factors by a Modified Crack Closure Integral, *Engng. Fract. Mech.*, vol. 9, pp. 931–938, 1997.
29. P. Paris and G. Sih, Stress Analysis of Cracks, Fracture Toughness and Testing and its Applications. STP381, ASTM, 30–83, Philadelphia, 1965.
30. J. N. Reddy and R. A. Arciniega, Shear Deformation Plate and Shell Theories: Stavsky to present, *Mechanics of Advanced Materials and Structures*, vol. 11, no. 6, pp. 535–582, 2004.

Copyright © 2005 Tech Science Press CMES, vol.8, no.2, pp.135-152, 2005

Multiscale Simulations Using Generalized Interpolation Material Point (GIMP) Method And SAMRAI Parallel Processing

J. Ma[1], H. Lu[1], B. Wang[1], S. Roy[1], R. Hornung[2], A. Wissink[2] and R. Komanduri[1,3]

Abstract: In the simulation of a wide range of mechanics problems including impact/contact/penetration and fracture, the material point method (MPM), Sulsky, Zhou and Shreyer (1995), demonstrated its computational capabilities. To resolve alternating stress sign and instability problems associated with conventional MPM, Bardenhagen and Kober (2004) introduced recently the generalized interpolation material point (GIMP) method and implemented for one-dimensional simulations. In this paper we have extended GIMP to 2D and applied to simulate simple tension and indentation problems. For simulations spanning multiple length scales, based on the continuum mechanics approach, we present a parallel GIMP computational method using the Structured Adaptive Mesh Refinement Application Infrastructure (SAMRAI). SAMRAI is used for multi-processor distributed memory computations, as a platform for domain decomposition, and for multi-level refinement of the computational domain. Nested computational grid levels (with successive spatial and temporal refinements) are used in GIMP simulations to improve the computational accuracy and to reduce the overall computational time. The domain of each grid level is divided into multiple rectangular patches for parallel processing. This domain decomposition embedded in SAMRAI is very flexible when applied to GIMP. As an example to validate the parallel GIMP computing scheme under SAMRAI parallel computing environment, numerical simulations with multiple length scales from nanometer to millimeter were conducted on a 2D nanoindentation problem. A contact algorithm in GIMP has also been developed for the treatment of contact pair between a rigid indenter and a deformable workpiece. GIMP results are compared with finite element results on indentation for validation. A GIMP nanoindentation problem with five levels of refinement was modeled using multi-processors to demonstrate the potential capability of the parallel GIMP computation.

keyword: Material point method (MPM), Generalized interpolation material point method (GIMP), Tension, Nanoindentation, Parallel computing, SAMRAI, Multilevel refinement, Contact problem.

1 Introduction

The material point method (MPM) has demonstrated its capabilities in addressing such problems as impact, upsetting, penetration, and contact (e.g. Sulsky, Zhou and Schreyer (1995); Sulsky and Schreyer (1996)). In MPM, two descriptions are used – one based on a collection of material points (Lagrangian) and the other based on a computational background grid (Eulerian), as proposed by Sulsky, Zhou and Schreyer (1995). A fixed structured mesh is generally used in the background throughout the MPM simulations. The material points are followed throughout the deformation of a solid to provide a Lagrangian description and the governing field equations are solved at the background grid nodes so that MPM is not subject to mesh entanglement. Compared to the finite element method (FEM), MPM takes advantage of both Eulerian and Lagrangian descriptions and possesses the capability of handling large deformations in a more natural manner so that mesh lock-up problems present in FEM are avoided. Additionally, for problems involving contact, MPM provides a naturally non-slip contact algorithm to avoid the penetration between two bodies based on a common background mesh (Sulsky, Zhou and Schreyer (1995), Sul-

[1] School of Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, OK 74078, U.S.A

[2] Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, CA 94551, U.S.A

[3] Correspondence author, e-mail: ranga@ceat.okstate.edu; Tel: 405-744-5900; Fax: 405-744-7873.

This work was performed under the auspices of the U.S. Department of Energy by University of California, Lawrence Livermore National Laboratory under Contract W-7405-Eng-48. UCRL-JRNL-208241.

sky and Schreyer (1996)). One drawback of the conventional MPM is that when the material points move across the cell boundaries during deformation, some numerical noise/errors can be generated, Bardenhagen and Kober (2004). To solve the instability problems associated with the conventional MPM simulations, Bardenhagen and Kober recently proposed the generalized interpolation material point method (GIMP) and implemented for one-dimensional simulations.

The present investigation extends the GIMP presented by Bardenhagen-Kober to two-dimensional simulations and applies it to simple tension and indentation problems. Furthermore, a refinement technique and a parallel processing scheme are developed so that the serial GIMP algorithm and code can be extended for parallel computation of large scale computations based on the continuum mechanics approach.

Parallel processing has been used successfully in numerical analysis using different methods, such as FEM and boundary element method (BEM), Mackerle (2003) and molecular dynamics (MD), Kalia and Nakano (1993). The computational time on parallel processors can be reduced to a small fraction of the time consumed by a single processor at the same speed. Parallel processing generally involves issues, such as domain decomposition/partitioning, load balancing, parallel solver/algorithms, parallel mesh generation, and multi-grid, Mackerle (2003). Domain decomposition has been widely applied in parallel processing in FEM, Hsien (1997). With partitioning of the overall computational domain, sequential FEM algorithm usually cannot be used directly in parallel processing without some modification, primarily due to the coupling of a large number of simultaneous linear equations. Remeshing is sometimes needed in each sub-domain. The interfaces of neighboring sub-domains must be meshed identically for subsequent communications, Mackerle (2003). These problems are intrinsic to certain numerical methods, such as FEM; however, they can be totally or partially avoided if other appropriate computational methods are used. For example, the domain decomposition is more straightforward for structured meshes, and large systems of coupled equations can be avoided, if explicit time integration is used.

Recently a platform for parallel computation, namely, the structured adaptive mesh refinement application infrastructure (SAMRAI), Hornung and Kohn (2002), has been developed by the Center for Applied Scientific Computing at the Lawrence Livermore National Laboratory. SAMRAI has provided interfaces for user-defined data types so that material points carrying physical variables (mass, displacement, velocity, acceleration, stress, strain, etc.) can be readily defined. As a result, SAMRAI is very suitable for handling material points and their physical variables in MPM or its variant, GIMP. In this investigation SAMRAI is used for parallelizing GIMP. SAMRAI has also provided a foundation for parallel adaptive mesh refinement (AMR) with the use of either dynamic or static load balancing, Wissink, Hysom and Hornung (2003). This function allows SAMRAI to process both spatial and temporal refinements in areas of interest, typically with high gradients in some physical variables (e.g., strains), and to use coarse mesh in the remaining areas. With the appropriate use of fine and coarse meshes in different regions, multiscale simulations using MPM can provide desired computational accuracy with reduced costs associated with computer memory and computational time.

Material multiscale simulations span from electronic structure, atomistic scale, crystal scale, to macro/continuum scale, Horstemeyer, Baskes, Prantil, Philliber and Vonderheide (2003); Komanduri, Lu, Roy, Wang and Raff (2004). Appropriate simulation algorithms can be used at various scales, e.g., *ab initio* computation for electronic structure, molecular dynamics at the atomistic scale, crystal plasticity or mesoplasticity at the crystal scale, and continuum mechanics at the macro scale, Horstemeyer, Baskes, Prantil, Philliber and Vonderheide (2003). At the continuum scale, FEM is generally used. Recently, the meshless local Petrov-Galerkin (MLPG) method (Shen and Atluri (2004a, 2005)) and the continuum/lattice Green function method (Tewary and Read (2004)) have been used to couple with molecular dynamics seamlessly. The MLPG method is a simple, less-costly alternative approach to FEM, Atluri and Shen (2002). For the purposes of providing the insights into the discrete atomistic system and coupling with continuum, an equivalent continuum was defined in the MD region to compute the atomic stress based on the Smoothed Particle Hydrodynamics (SPH) method, Shen and Atluri (2004b). The atomic stress tensor computed using the SPH method is more natural than other atomic stress formulations because it is in the nonvolume-averaged form and rigorously

satisfies the conservation of linear momentum. Hence, it is applicable to both homogeneous and inhomogeneous deformations. A tangent stiffness formulation was developed for both MLPG and MD regions and the displacements of the nodes and atoms are solved in one coupled set of linear equations. The MLPG/MD coupling has been demonstrated to be capable of enforcing the local balance equations in the handshaking region between continuum mechanics and molecular dynamics, Shen and Atluri (2005).

The simulation using parallel GIMP computing scheme in this investigation will focus on multiscales, e.g., from nanometer to millimeter, based on the continuum mechanics approach, namely, 2D GIMP. An example used for validating the simulation at several length scales at the continuum level is nanoindentation. It involves the contact issue between a rigid indenter and a deformable workpiece. A contact algorithm, which allows the contact interface to be located in a few computational domains, is introduced in this study. The contact pressure is determined from solving a set of equations from multiple processors. Parallel GIMP results on nanoindentation are compared with FEM results using the ABAQUS/Explicit code. A nanoindentation model with five levels will be used; this model allows simulation from nanometer to millimeter scales.

2 Generalized Interpolation Material Point (GIMP) Method

The governing equations in both conventional material point method (MPM), Sulsky, Zhou and Sheryer (1995), Hu and Chen (2003), Bardenhagen (2002) and generalized interpolation material point (GIMP) method, Bardenhagen and Kober (2004), are briefly summarized in this section. The weak form of the momentum conservation equation in the conventional MPM is given by

$$\int_\Omega \rho \mathbf{w}\cdot\mathbf{a}d\Omega = -\int_\Omega \rho \mathbf{s}^S : \nabla \mathbf{w} d\Omega + \int_{\partial\Omega} \rho \mathbf{c}^S \cdot \mathbf{w} dS + \int_\Omega \rho \mathbf{w}\cdot\mathbf{b}^s d\Omega \quad (1)$$

where $\mathbf{w}$ is the test function, $\mathbf{a}$ is the acceleration, and $\mathbf{s}^s$, $\mathbf{c}^s$ and $\mathbf{b}^s$ are the specific stress, specific traction, and specific body force, respectively. Ω is the current configuration and $\partial\Omega$ is the surface with applied traction. The material density, ρ, can be approximated as the sum of material point masses using a Dirac delta function $\rho(\mathbf{x},t) = \sum_{p=1}^{N_p} M_p \delta(\mathbf{x}-\mathbf{x}_p^t)$, where N_p is the total number of material points and M_p is the mass of the material point. Upon discretization of Eq. (1) using the shape functions $N_i(\mathbf{x}_p^t)$, the governing equations at the background grid nodes become (see Sulsky, Zhou and Schreyer (1995))

$$m_i^t \mathbf{a}_i^t = (\mathbf{f}_i^t)^{int} + (\mathbf{f}_i^t)^{ext}. \quad (2)$$

where the lumped mass matrix is given by

$$m_i^t = \sum_{p=1}^{N_p} M_p N_i(\mathbf{x}_p^t), \quad (3)$$

and the internal and external forces are given by

$$(\mathbf{f}_i^t)^{int} = -\sum_{p=1}^{N_p} M_p \mathbf{s}_p^{s,t} \cdot \nabla N_i|_{\mathbf{x}_p^t}, \quad (4)$$

$$(\mathbf{f}_i^t)^{ext} = -\sum_{p=1}^{N_p} M_p \mathbf{c}_p^{s,t} h^{-1} N_i(\mathbf{x}_p^t) + \sum_{p=1}^{N_p} M_p \mathbf{b}_p^t N_i(\mathbf{x}_p^t), \quad (5)$$

where h is the thickness of a boundary layer. At each time step, all variables for each material point, such as mass, velocity, and force are extrapolated to the grid nodes of the cell in which the material point resides. New nodal momenta are computed and used to update the physical variables carried by the material points. Thus, material points move relative to each other to represent deformation in a solid. A spatially fixed background grid is used throughout the MPM computation. MPM has already demonstrated its capabilities in solving a number of problems involving impact/contact/penetration. In case of large deformation, however, numerical noise, or errors have been observed, especially when material points have just crossed cell boundaries resulting in instability problems in the MPM simulations (see, e.g., Sulsky, Zhou and Schreyer (1995), Hu and Chen (2003), Bardenhagen and Kober (2004)). The primary cause for the problem has been attributed to the discontinuity of the gradient of the shape functions across the cell boundaries (see, e.g., Hu and Chen (2003), Bardenhagen and Kober (2004)). To resolve this problem, Bardenhagen and Kober (2004) proposed a generalized interpolation material point (GIMP) method. In GIMP, the interpolation between node i and material point p is given by the volume averaged weighting function

$$\overline{S}_{ip} = \frac{1}{V_p}\int_{\Omega\cap\Omega_p} \chi_p(\mathbf{x}) S_i(\mathbf{x}) d\mathbf{x}, \quad (6)$$

where V_p is the current volume of the material point, $\chi_p(\mathbf{x})$ is the characteristic function of the material point, and $S_i(\mathbf{x})$ is the node shape function. The role of the weighting function is the same as the shape function in conventional MPM. The modified equation of momentum conservation, Bardenhagen and Kober (2004), can be written as

$$\sum_p \int_{\Omega\cap\Omega_p} \frac{\dot{p}_p\chi_p}{V_p}\cdot\delta\mathbf{v}d\mathbf{x}+\sum_p \int_{\Omega\cap\Omega_p} \sigma_p\chi_p:\delta\mathbf{v}d\mathbf{x} = \sum_p \int_{\Omega\cap\Omega_p} \frac{m_p\chi_p}{V_p}\mathbf{b}\cdot\delta\mathbf{v}d\mathbf{x}+\int_{\partial\Omega}\mathbf{c}\cdot\delta\mathbf{v}d\mathbf{x} \quad (7)$$

where $\delta\mathbf{v}$ is an admissible velocity field, $\dot{p}_p$ is the rate of change of the material point momentum. Eq. (7) can be further discretized and solved at the grid nodes, Bardenhagen and Kober (2004). Herein, the weighting function $\overline{S}_{ip}$ is C^1 continuous under the spatially fixed background grid. Consequently the noises associated with material point crossing cell boundaries in the conventional MPM can be minimized.

In this paper, we have implemented GIMP presented by Bardenhagen-Kober for two-dimensional simulations. We have also developed a refinement technique and a parallel processing scheme to extend the serial GIMP algorithm to code large scale parallel computing. The capability of parallel GIMP computing has been demonstrated by modeling nanoindentation problem. A contact algorithm has been developed to address the contact problem between the rigid indenter and the deformable workpiece. We proceed next to describe the contact algorithm developed in this investigation.

2.1 *Contact Algorithm in GIMP*

Nanoindentation involves a contact pair of a rigid indenter and a deformable workpiece. The contact interaction between these two surfaces is governed by the Newton's third law and Coulomb's friction law as well as the boundary compatibility condition at the contact interface, Oden and Pires (1983), Zhong (1993). While MPM can prevent the penetration at the interface automatically, it uses a single mesh for the two bodies. At the contact surface, all components of the variables are interpolated to the nodes from both bodies using Eqs. (3)-(5). As a result, MPM using a single mesh tends to induce early contact in approaching and late separation when two parts move away from each other. So, MPM cannot model the contact behavior between two parts correctly. Hu and Chen (2003) proposed a multi-mesh MPM algorithm to release the no-slip constraint inherent in the MPM using a single mesh. In the multi-mesh MPM, in addition to a common mesh for all objects, there is an individual mesh for each of the objects under consideration. All meshes are identical, i.e. nodal locations are the same. The multi-mesh can be generated by creating multiple nodal fields for each node. Each nodal field corresponds to an object. In multi-mesh MPM scheme, the nodal masses and forces are mapped from the material points of each object to its own mesh. The nodal values are transferred to the corresponding nodes in the common mesh. When the values at a node of the common background mesh involve contributions from two parts, the contact between two parts occurs so that this node is defined as an overlapped node. Otherwise, two parts move independently. This multi-mesh algorithm can handle sliding and separation for the contact pair. However, in using the multi-mesh for contact problem in GIMP, the interaction at the overlapped nodes is still activated too early before the actual contact of the material points occurs.

Fig. 1 is an example illustrating early contact when Part 1 is moving toward Part 2. The four bottom particles of Part 1, labeled in hollow circles, have come into the cells of Part 2, the nodes of which cells are labeled in three dashed circles. Physical variables (e.g., normal force, and velocity component normal to the contact surface) in Part 1 will be interpolated onto these three overlapped nodes. The physical variables in the three overlapped nodes will be further interpolated into material points within the top layer of cells in Part 2, and contribute to the stress and deformation in the entire Part 2. With this treatment in the previous multi-mesh algorithm, even though Parts 1 and 2 are not in physical contact, the particles in Part 2 will contribute to the physical variables of particles in Part 1, leading to numerical early contact, and vice versa, through the overlapped nodes. Similar situation occurs when Part 1 is retracting from Part 2, resulting in late separation of two parts. Unless other measures are taken to prevent these physically incorrect early contact and late separation problems, they could cause large errors in GIMP and must be corrected in contact problems.

In this paper, a new contact algorithm is developed for GIMP simulations. Fig. 2 illustrates the contact algorithm for the contact pair between a rigid indenter and a deformable workpiece. Although circular points are

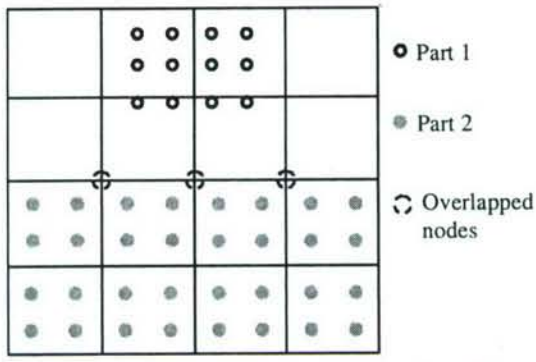


Figure 1 : Illustration of early contact in multi-mesh

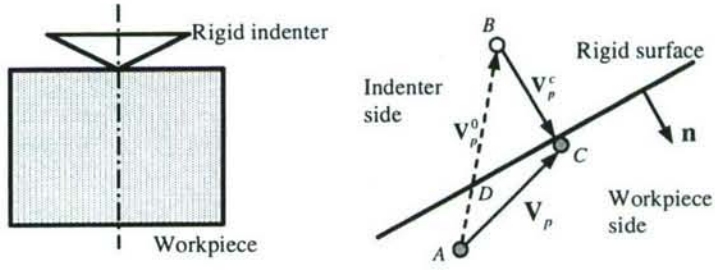


Figure 2 : Schematic of contact algorithm between the rigid indenter and the deformable workpiece

used in this schematic diagram, it should be noted that the points are representations of areas occupied by these points, based on the GIMP algorithm. A frictionless contact is assumed in this investigation. At the beginning of a time step, a material point is located at point A. At the end of this time step, the material point moves to B, if there is no contact interaction.

To satisfy the displacement compatibility condition, the material point has to be brought to the indenter surface and kept in contact with the indenter. The contact velocity correction $\mathbf{V}_p^c$ can be determined based on the rigid surface orientation indicated by its unit outward normal vector $\mathbf{n}$. The final location of the material point is set to C by a contact pressure. Hence, the velocity of a material point p under contact can be determined by

$$\begin{aligned}\mathbf{V}_p &= \sum_i^N \mathbf{V}_i \overline{S}_{ip} = \sum_i^N \frac{\mathbf{p}_i^0 + (\mathbf{F}_i^0 + \mathbf{F}_i^c)\Delta t}{m_i} \overline{S}_{ip} \\ &= \sum_i^N \frac{\mathbf{p}_i^0 + \mathbf{F}_i^0 \Delta t}{m_i} \overline{S}_{ip} + \sum_i^N \frac{\mathbf{F}_i^c \Delta t}{m_i} \overline{S}_{ip}\end{aligned} \tag{8}$$

where N is the number of nodes contributing to material point p, $\mathbf{F}_i^c$ is the contact force on node i, $\mathbf{p}_i^0$ and $\mathbf{F}_i^0$ are the nodal momentum and force without consideration of the contact, respectively. The velocity $\mathbf{V}_p^0$ of the material point without the consideration of contact is given by

$$\mathbf{v}_p^0 = \sum_i^N \frac{\mathbf{p}_i^0 + \mathbf{F}_i^0 \Delta t}{m_i} \overline{S}_{ip} \tag{9}$$

The contact force $\mathbf{F}_i^c$ on node i is the resultant of the contact pressures on the neighboring particles, and can be computed in terms of contact pressures using the approach given by Bardenhagen and Kober (2004), i.e.,

$$\mathbf{F}_i^c = \sum_{q=1}^{Q} \int_{\partial\Omega_q} S_i(\mathbf{x}) \mathbf{P}_q^c ds, \tag{10}$$

where Q is the total number of material points in contact with the indenter. If the contact pressure $\mathbf{P}_q^c$ is assumed to be constant in the contact area occupied by material point q, we have $\mathbf{F}_i^c = \sum_{q=1}^{Q} T_{iq} \mathbf{P}_q^c$, where $T_{iq} = \int_{\partial\Omega_q} S_i(\mathbf{x}) ds$. Since $\mathbf{V}_p = \mathbf{V}_p^0 + \mathbf{V}_p^c$, the contact velocity $\mathbf{V}_p^c$ for material

point p is given by

$$\mathbf{V}_p^c = \Delta t \sum_i^N \frac{\overline{S}_{ip}}{m_i} \sum_q^Q T_{iq} \mathbf{P}_q^c \tag{11}$$

Eq. (11) can be established for each material point in contact. At each material point there is an unknown contact pressure $\mathbf{P}_q^c$. Therefore, the number of unknown pressures, $\mathbf{P}_q^c$, is equal to the number of points in contact. In parallel computing, points in contact might be located in different domains processed by different processors. Consequently, a parallel solver is needed to solve Eq. (11) in this investigation. Since contact can only occur on the outer surface of an object, Eq. (11) is solved analytically under the physical contact condition $\mathbf{P}_q^c \cdot \mathbf{n} < 0$ to find the contact pressure at all material points in contact with the indenter. The contact pressure is then extrapolated to the nodes from the contact material points to update the total nodal forces.

2.2 *Numerical Implementation*

We consider the case where initially there are four material points in a cell for which the 2D weighting function is depicted in Fig. 3. To compute the weighting function, we take $\chi_p(\mathbf{x})$ to be one in the current region occupied by the material point p and zero elsewhere. In this figure, one node is at the origin and the horizontal axes give material point positions normalized by the cell size. Fig. 3 is based on the same material point characteristic function and node shape function as in Bardenhagen and Kober (2004). It is noted that the computation of the weighting function in the deformed state involves some practical difficulties because the integration boundaries in Eq. (7) can be difficult to obtain. To circumvent this problem, we assume that the shape of the region occupied by the four material points remains rectangular without rotation, so that Eq. (6) can be evaluated analytically. This assumption leads to significant saving in the computational time while introducing only small errors. Using this assumption, GIMP is extended to 2D simulations and the results are presented in Section 4.

3 Parallel Computing Scheme Using GIMP with SAMRAI

3.1 *Structured Adaptive Mesh Refinement Application Infrastructure (SAMRAI)*

The Structured Adaptive Mesh Refinement Application Infrastructure (SAMRAI), a scientific computational package for structured adaptive mesh refinement and parallel computation, is used with the GIMP for parallel computation of large-scale simulations. SAMRAI is chosen because of its similarity in grid structure with GIMP. In GIMP, the computation is usually independent of the background grid mesh so that a structured spatially-fixed mesh can be used throughout the entire simulation process. This advantage makes GIMP highly suitable for parallel computation, as the domain decomposition for structured mesh can be easily performed and no remeshing is required. Thus the complexity and inefficiency associated with parallel processing can be avoided.

In SAMRAI, the computational domain is defined as a hierarchy of nested grid levels of mesh refinement, Berger and Oliger (1984), as shown in Fig. 4. Each grid level is divided into non-overlapping, logically-rectangular patches, each of which is a cluster of computational cells. Indices are used extensively in SAMRAI to manage grid levels and patches. For example, patch connectivity is managed by the cell indices. The organization of the computational mesh into a hierarchy of levels of patches allows data communication and computation to be expressed in geometrically-intuitive box calculus operations. Communication patterns for data dependencies among patches can be computed in parallel without inter-processor communications, since the mesh configuration is replicated readily across processor memories. Inter-processor communications, i.e., data communications between patches on the same as well as neighboring levels, are pre-defined by SAMRAI communication schedules. Problem-specific communication interfaces are also provided by SAMRAI.

SAMRAI supports several data types defined in a patch, such as cell-centered data, node-centered data, and face-centered data. These data are stored as arrays to allow numerical subroutines to be separated easily from the implementation of mesh data structures. User-defined data structures over a patch, which can be accessed through cell index, are supported by SAMRAI. These characteristics make SAMRAI a very flexible parallel com-

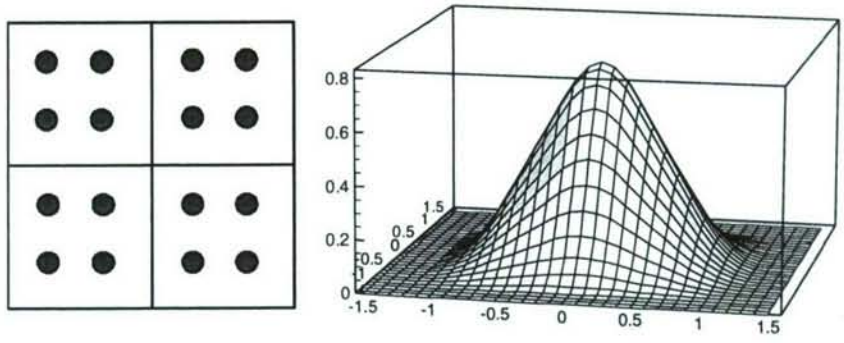


Figure 3 : Material points in cells and the weighting function in 2D GIMP

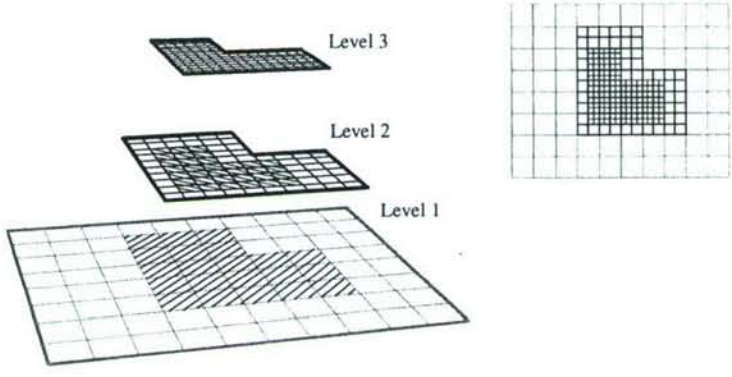


Figure 4 : Illustration of a hierarchy of three nested grid levels of mesh refinement

puting environment for numerous physics applications, Wissink, Hysom, and Hornung (2003).

3.2 Spatial and Temporal Refinements

In the application of SAMRAI to large-scale GIMP simulations, the techniques for refinement, both spatial and temporal, have to be developed to achieve high accuracy in areas of high stress/strain gradients while reducing the overall computational time by using coarse mesh in regions of low stress/strain gradients. Since a structured mesh is used in GIMP, the refinement can be implemented by imposing fine levels of sub-grids at locations of interest, using the approach adopted by Berger and Oliger (1984) in SAMRAI. The scheme for the structured grid refinement is illustrated in Fig. 4. The cell size ratio, also called the refinement ratio, of two neighboring levels is always an integer for convenience. The advantage of this refinement technique is that nesting relationships between different levels can be handled. A material point in GIMP can be split into several small material points. Tan and Nairn (2004) proposed a criterion to split material points based on local deformation gradient. If the refinement ratio is two in each direction, one coarse material point can be split into four material points in the next fine level in 2D case. However, this splitting technique can become complicated when conservation of energy and momentum have to be enforced. In this paper, a more natural refinement approach is developed to avoid direct splitting and merging processes by using material points of the same size and mass in overlapped region (called ghost region) between two neighboring levels.

Fig. 5 shows two neighboring coarse and fine grid levels in 2D GIMP computations with a refinement ratio of two. The thick line represents the physical bound-

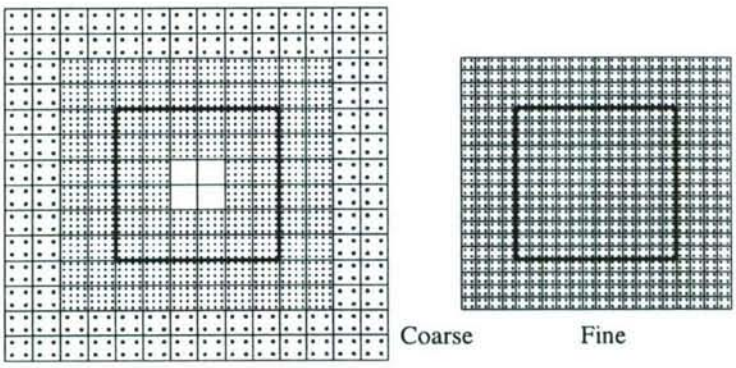
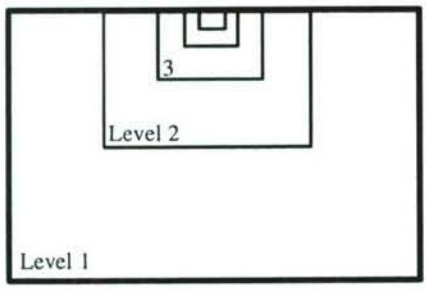


Figure 5 : Two neighboring coarse and fine grid levelsin 2D GIMP computations

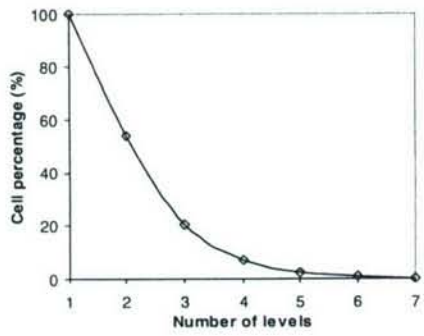
ary of the fine level with four layers of ghost cells. Initially, four material points are assigned to each cell at the fine level. At the coarse level, the portion overlapped by the fine level is assigned with 16 material points per cell. Hence, these material points have the same size and initial positions as those at the fine level. The rest of the coarse level is assigned with four material points per cell. GIMP provides a natural coupling of the material points with different sizes at the same grid level. This is because the weighting function depends on the characteristic size of the material points and cell length and the interpolation between the nodes and material points is weighed by the mass of the material point. In GIMP computation, each level is computed independently with the physical variables communicated through the ghost regions between neighboring levels. Two data exchange processes, namely, refinement and coarsening are used in the communication. Refinement process passes information from the coarse level to the immediate fine level, while coarsening process will pass information from fine level to the next coarse level. In the refinement process, physical variables at the fine material points inside the thick lines in Fig. 5 are copied directly to replace the material points at the coarse level. In the coarsening process, the physical variables at coarse material points are copied to the ghost cells of the immediate fine level.

In the refinement, the material points located in the ghost cells at the fine level are eliminated first, and the material points in the corresponding region of the coarse level (with the same size as points in the immediate fine level) are copied to ghost cells at the fine level. In the copying process, if some (small) material points in the coarse level fall into the interior (inside the square on the right of Fig. 5) of the immediate fine level, these points will not be copied to this fine level, as points in the fine level will be able to carry over all computations in the interior of the fine level already. In coarsening, the material points of the coarse level located in the region overlapped by the fine level (inside the square on the left of Fig. 5) are eliminated first, and the material points in the fine level are then copied to the immediate coarse level. With these refinement/coarsening operations, the material points can move around freely, including moving outside the original level in large deformations. To ensure that this coarsening process can still be performed reliably during deformation, sufficiently wide region of cells should be assigned with refined material points at the coarse level so that the ghost cells of the fine level always stay within the region with fine material points on the coarse level. At the coarse level, the interior cells covered by the fine level do not participate in the computation and there are no material points inside (see Fig. 5).

The refinement techniques can be applied for multiple times at the regions of interest, such as the stress concentration regions. A fixed refinement ratio of two between two neighboring levels is very effective in reducing the total number of computational cells. Fig. 5 shows nested multi-level refinement and its corresponding relation between the total number of cells and the number of grid levels. The cell percentage represents the ratio of the total number of cells with multi-level refinement mesh to the total number of cells with one-level finest mesh. If each fine level occupies one quarter of the neighboring coarser level, as shown in Fig. 6 (a), the cell percentage as a



(a)



(b)

Figure 6 : Nested multi-level refinement and reduction in the number of cells with number of levels

function of the number of grid levels can be calculated, as shown in Fig. 6 (b). For example, when totally four levels of successive refinements are used the total number of cells is about 8% of that of one uniform fine mesh. A reduction in the number of computational cells leads to a reduction in the number of material points. Hence, the total amount of computational time can be reduced significantly. However, refinement and coarsening communications will cost additional computational time, as will be discussed in Section 4.

Another advantage of the multi-level refinement is that it allows for temporal refinement. Since the computation at each grid level is conducted independently, different time step increments can be used for computation at different levels. For example, a smaller time step increment can be used for the fine level to improve computational accuracy, while a larger time step increment can be used for the coarse level. Since the refinement ratio is an integer, the time step increment ratio should also be an integer for convenience in the computation and data communication/synchronization. For example, in Fig. 5, when the refinement ratios in both directions are fixed at two, the time step increment ratio should be set to two as well. As a result, two time step computations are performed at the fine level, and results are passed over to the immediate coarse level to couple with the results at the coarse level.

3.3 Domain Decomposition

GIMP uses structured mesh, consistent with SAMRAI, so that domain decomposition is straightforward and no remeshing, in general, is necessary. Fig. 7 (a) shows a two-dimensional computational domain decomposed into two patches separated by a horizontal dash line. The elliptical solid object with different boundary conditions applied at different regions is inside this domain/grid. After discretization, there are a certain number of material points and part of the boundary in a patch, which will be computed individually. It may be noted that patch boundary does not have to coincide with the boundary of the material continuum. The patch boundary is always chosen to be larger than the region occupied by the material continuum so that there is extra space for the material to deform. This will not cause any additional computational burden as the GIMP computation is only carried out on material points inside the patch. Each patch can be processed by a single processor and the convenience in creating patches will provide great flexibility in parallel processing.

Communication between two neighboring patches is realized through information sharing in the region overlapped by the two patches. The overlapped regions are also called 'ghost' regions, as shown in Fig. 7 (b). The ghost cells are denoted by dash lines. For ease of visualization, only the ghost cells overlapped by the other patch are shown and the ghost cells along the other three sides of a patch are not shown. On one grid level, patches can communicate with each other by simply copying data from one patch to another at the same computation time (Fig. 7 (b)). Using the material point information from the previous time step, and the physical boundary conditions, each patch is ready to advance one more time step. At this time, the material point information in the outermost layer of the ghost cells becomes inaccurate. For instance, one outermost grid node in patch one, marked by the circle, obtains information from eight material points before advancing to the next step. After advancing, it

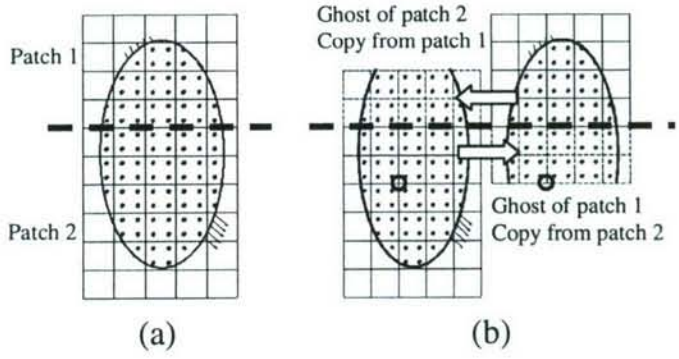


Figure 7 : A computational domain of two patches in one grid level

extrapolates to eight material points. However, in patch two, the grid node at the same location obtains information from sixteen surrounding material points. It extrapolates to these sixteen material points after advancing. Typically, after each step, the material points in the next inner layer in the ghost region become inaccurate as well.

Ghost cells and material points are attached to each patch to ensure accuracy of the interior. Each patch can be computed independently for one GIMP step since the momentum conservation equation is solved at each node and there are no coupled equations to solve. No data exchange is necessary during the GIMP step. Therefore, different patches can be assigned to different processors for parallel processing. After one GIMP step, the data in the ghost cells will be updated.

Copying material points to ghost cells involves data exchange between processors, which costs additional time. The more the number layers of ghost cells, the longer the time needed for communication, but communication can be performed less frequently. A minimum of two layers of ghost cells are necessary to ensure that computation at the material points inside a patch is always correct. If three levels of ghost cells are chosen, the communication can be performed after every two increments of each patch.

With these refinements and domain decomposition schemes for GIMP, it is possible to implement GIMP into the SAMRAI platform. In this study, the refinement ratio is chosen as two. Four layers of ghost cells are augmented to a patch such that data communications, including both data exchange on the same level and between neighboring levels, are performed every two time-steps for each fine grid level. This is critical because data exchange between levels has to be performed when the two levels are synchronized.

Fig. 8 shows the flowchart advancing all grid levels recursively starting from the coarsest level for one coarsest time step. It may be noted that the sequential GIMP algorithm can be used to advance each patch without modification.

4 Numerical Examples and Results

A Beowulf Linux cluster of 8 identical PCs were used in the simulations. Each PC has a Pentium 4 processor with a 2.4 GHz CPU, 512 MB RAM except that the master node has a memory of 1 GB. A gigabit switch is used to connect the network.

Two examples are used for validation of the 2D parallel GIMP computing under SAMRAI platform. The first example is simple tension of polycrystalline silicon under plane strain conditions. The material is assumed to be homogeneous, isotropic, and linear elastic. The Young's modulus is 170 GPa and the Poisson's ratio is 0.18. One end is constrained along the X- direction while a normal traction is ramped up on the other end. The size of the tensile model is 0.06 mm $\times$ 0.04 mm. The length of a square grid cell is 0.002 mm and the time step is 5×10^4 ps. For verification, the same problem was simulated using both conventional MPM and FEM (ABAQUS/Explicit). Fig. 9 shows GIMP, MPM and FEM simulation results of normal stresses in the X-direction at different increments from a simple tension problem. The simulation using the conventional MPM in

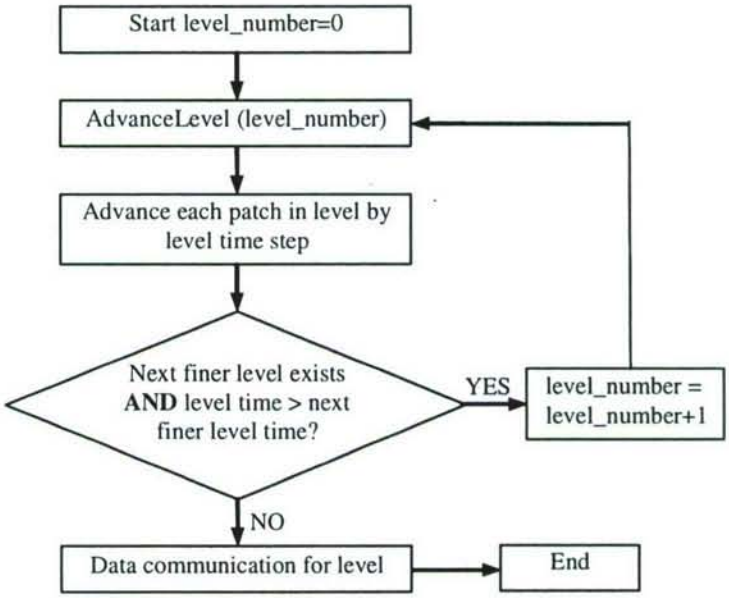
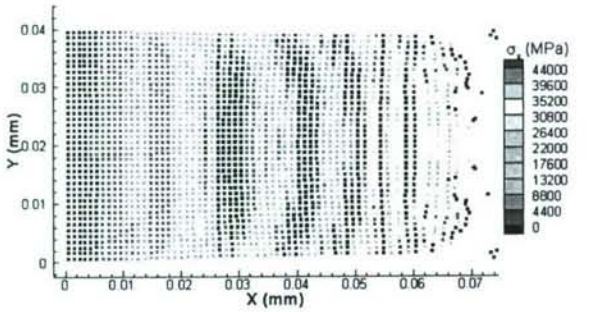


Figure 8 : Flowchart showing advancement of grid levels recursively starting from the coarsest to the finest level in GIMP

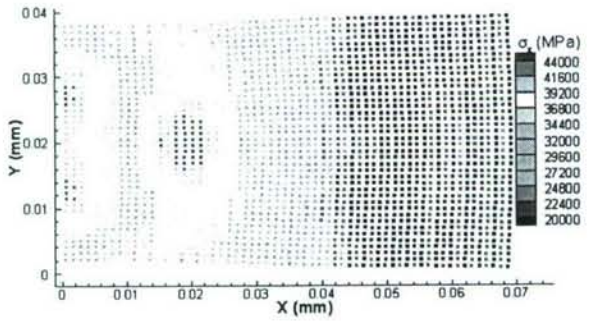
Fig. 9 (a) shows material separation close to the free end with severe numerical instability after 275 increments. Fig. 9 (b) and (c) show the normal stress distribution in the tensile direction and deformation after 500 increments from FEM and GIMP. It may be noted that FEM results show a stress contour plot on the deformed mesh while the GIMP results show a discrete scattered plot of material points. These two results are in good agreement with the difference in the maximum value being less than 10%.

In the second example, indentation on the same silicon material is simulated. The workpiece is subjected to a pressure applied in the middle of the top surface (Fig. 10 (a)) under plain strain conditions with a thickness of 0.001 mm for computing the mass and forces. The magnitude of the pressure increases linearly with time for the first 1500 increments, and is then kept constant (see Fig. 10 (b)). The cell size is 0.001 mm in both directions and the time step is 20 ps for both FEM and GIMP simulations. Due to symmetry, only one half of the workpiece is modeled. This simulation is performed with two patches in one uniform grid level. Two processors are used and one patch is assigned to each processor. Fig. 11 shows GIMP and FEM results of normal stresses in the Y-direction at different increments. The dashed line in Fig. 11 (a) is the boundary between the two patches. Fig. 11 (a) and (b) are plots of normal stresses in Y-direction at 500 time increments for GIMP and FEM simulations. The difference in stress values in Fig. 11 (a) and (b) is less than 5%. It should be noted that the FEM simulation aborted at 1348 increments due to excessive element distortion. The GIMP simulation did not encounter this problem. Fig. 11 (c) shows the GIMP stress result after 2000 increments. This demonstrates the capability of GIMP in handling excessive distortions.

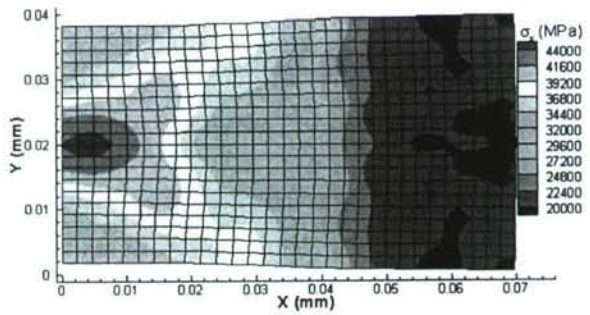
In order to validate the multi-level refinement algorithm and parallel communication, as well as the proposed contact algorithm, a simulation of nanoindentation with a wedge indenter was conducted under 2D plane strain conditions. The workpiece is aluminum and the indenter is assumed to be rigid. Fig. 11 shows the indentation model. The area below the indenter where high stress gradients are expected is refined, as shown in Fig. 12 (a). A prescribed velocity was applied on the indenter, as shown in Fig. 12 (b). The work piece dimensions are 60 μm $\times$ 40 μm. It is fixed in the Y-direction at the



(a) Conventional MPM



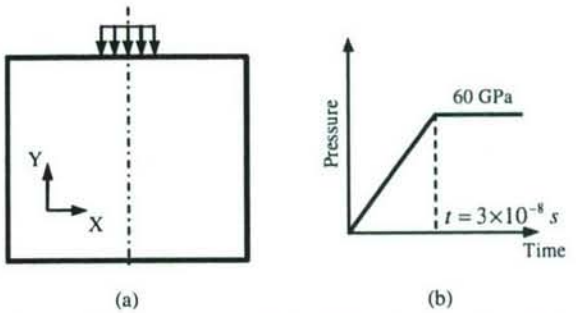
(b) GIMP



(c) FEM

Figure 9 : Simulation results of tensile stress contours for a simple tension problem

bottom. Only half of the model is simulated because of symmetry. The cell sizes are 500 nm, 250 nm and 125 nm for levels 1, 2 and 3, respectively. Each level is divided into four patches with approximately the same size. The maximum indentation depth in the simulation is about 450 nm. The dotted lines in Fig. 12 (a) illustrate the four patches in level 1. For comparison, an explicit FEM simulation (using ABAQUS/Explicit) was carried out under



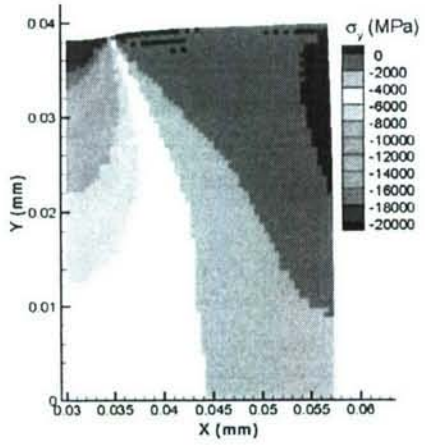
(a) (b)

Figure 10 : Loading conditions for a simple indentation problem

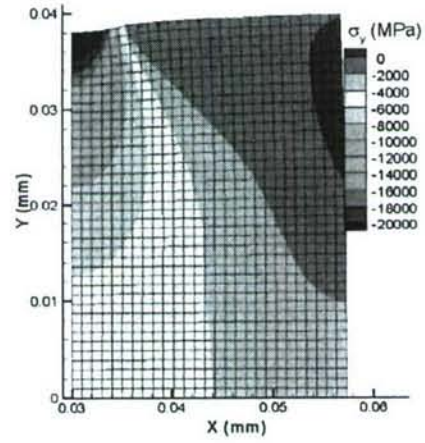
the same conditions. The FEM element size is uniform and is the same size as the finest GIMP background grid size. In this example, the maximum indentation depth (450 nm) is relatively small compared to the finest cell size, so that FEM simulation has not encountered excessive mesh distortion.

Fig. 13 shows a comparison of contours of normal stresses in the Y-direction at the maximum depth for both FEM and parallel GIMP simulations. The axis of symmetry of the workpiece is located at X=0.03 mm. For FEM, the plot is the contour of nodal stresses with deformed positions, and for GIMP, it is a discrete scattered plot of stress at deformed material points. The area below the indenter with high stress gradients is refined as shown in Fig. 12 (a) for parallel GIMP computation. The borders of grid levels 2 and 3 can barely be seen in Fig. 13 (b) due to the use of high material point density. Fig. 14 is a close-up view of shear stresses in which the three grid levels are shown. Results show that the normal and shear stresses from both parallel GIMP and FEM simulations agree very well. The difference of the normal stresses in the Y-direction for the material point in contact with the indenter tip and the stress of the FEM node at the same location is 4.4%. It may be noted that some non-smoothness in the GIMP stresses around the level boarders can be seen. This non-smoothness is caused by the refinement and coarsening and the error associated with this is negligible for these simulations.

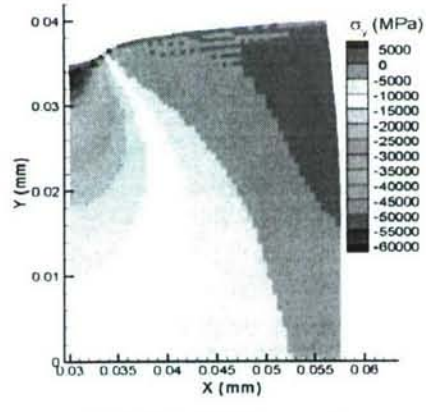
GIMP simulations using a uniform cell size of 500 nm and 125 nm were performed under the same conditions as in Fig. 12 to further verify the refinement/coarsening algorithm. Fig. 15 shows normal stresses in the Y-direction



(a) GIMP at 500 increments



(b) FEM at 500 increments



(c) GIMP at 2000 increments

Figure 11 : GIMP and FEM results of normal stress variation in the Y-direction at different increments

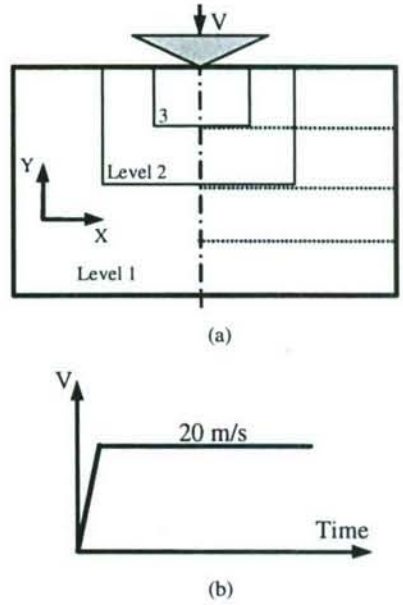
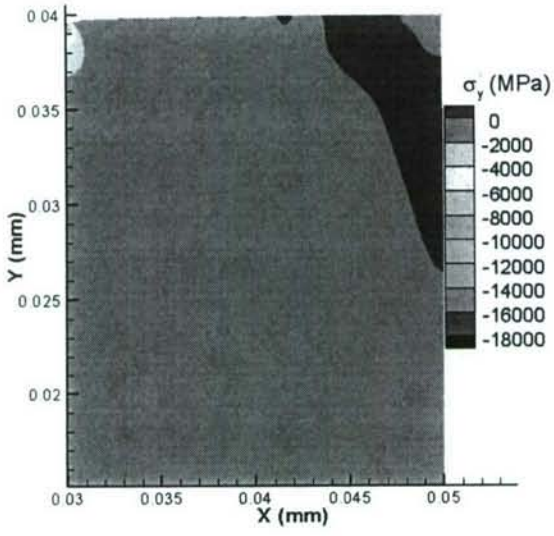


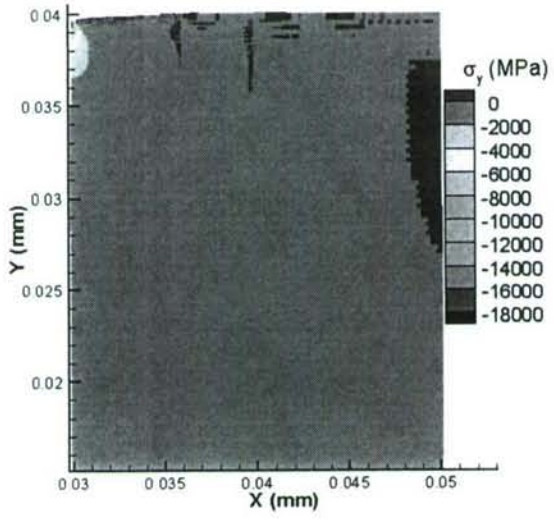
Figure 12 : Schematic of 2D indentation showing (a) three levels of refinement and (b) the indenter velocity history

and shear stresses in GIMP simulations using 500 nm uniform cells. In this case the material points in contact with the indenter are 16 times (4 times in each direction) larger than those with two levels of refinements. In general, the stress magnitudes agree with those in Fig. 13 and Fig. 14.

Fig. 16 (a) shows indentation load versus depth curves from FEM and GIMP with different grid sizes. From GIMP computations, the load versus depth curves with three levels of refinement agree very well with the results from a uniform finest mesh. The load versus depth curve from the FEM simulation with a uniform cell size of 125 nm under the same boundary conditions is plotted for comparison. It can be seen from Fig. 16 (a) that the trend of the load versus depth plots from FEM and GIMP simulations are similar. The difference in indentation load at the end of loading, which corresponds to 450 nm of indentation depth, is 5.9% between FEM and GIMP with 3 grid levels. When the depth is less than 100 nm, there is only one material point in contact with the rigid indenter. The assumption of constant pressure causes large differences under this circumstance. How-

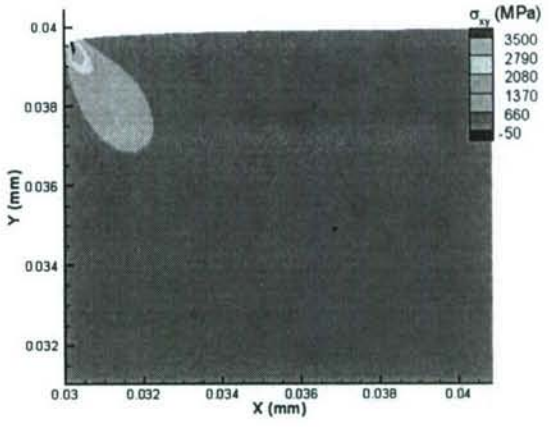


(a) FEM

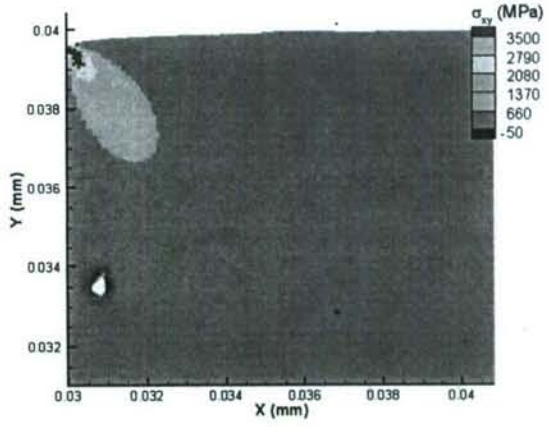


(b) GIMP

Figure 13 : Comparison of normal stresses in Y-direction from FEM and GIMP



(a) FEM

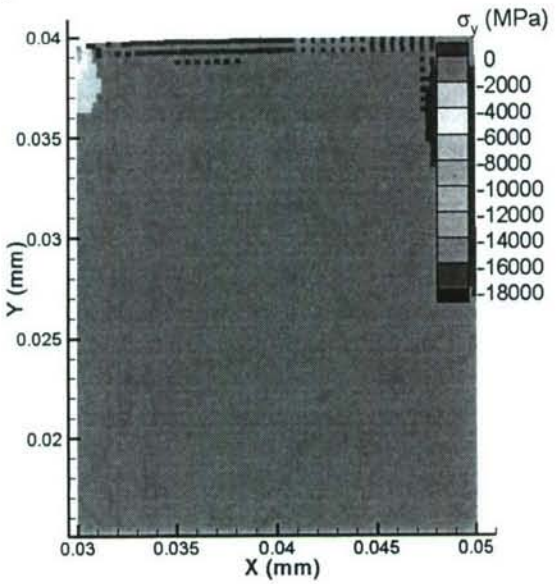


(b) GIMP

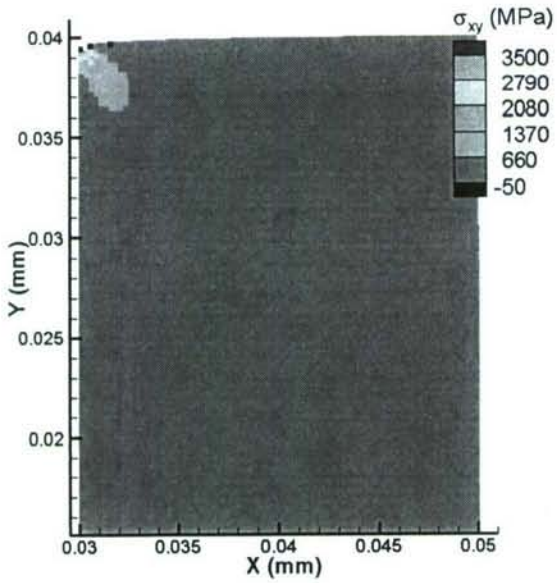
Figure 14 : Comparison of shear stresses from FEM and GIMP

ever, if the GIMP cell size is further refined to 62.5 nm and the size of the material point is 31.25 nm, the difference between GIMP and FEM becomes smaller, as can be seen in Fig. 16 (b).

Other simulations were conducted for the same problem with three levels of refinement using different number of processors to test the efficiency of parallel computing. The number of patches at each level is the same as the number of processors and the size of each patch is approximately the same. The resultant stress distribution and indentation load versus depth plots are the same as the previous results. The average time per computational step is 7.14 sec. when one processor is used and is reduced to 4.26, 3.40, 2.18 sec., respectively when two, three, and four processors are used. When four processors are used, the CPU time per step is only 30.5% of that of one processor. This gives a speed-up by a factor



(a) Normal Stress



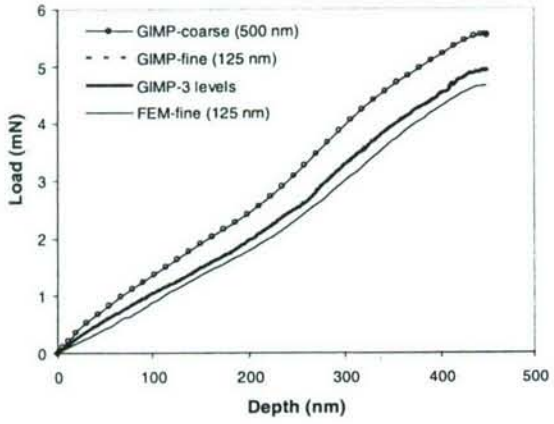
(b) Shear Stress

Figure 15 : Normal and shear stresses of GIMP simulations with a uniform cell size of 500 nm

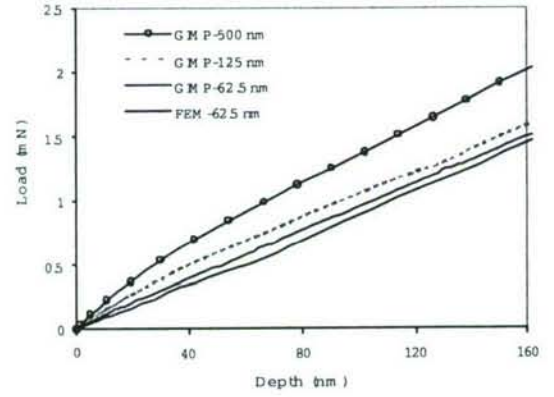
of 3.28. In the ideal case without communication overhead, the speed-up would be 4. The reduction in speed-up from the ideal number is because of the time involved in data communication between processors. It has been observed that the refinement and coarsening algorithm consume most of the communication time. Moreover, in refinement and coarsening, most of the time is taken to search for the corresponding material point in another grid level. This portion of the computational time can be reduced, if improved searching algorithm or more optimized algorithm for the storage of material points can be implemented.

The manual refinement for the indentation problem is adequate since the region of high stress gradient is known to occur below the indenter. The finest level covers the indenter and part of the specimen. With the same initial condition, the results at the finest level is identical to the results in the same area if a uniform fine mesh is used for the entire domain that requires much longer computational time. The computational load of each processor is balanced statically by assigning approximately the same number of material points to each processor. Dynamic load balance is supported by SAMRAI and can potentially improve the efficiency of the simulation.

To demonstrate the capability of the algorithm developed in this investigation for multiscale simulation, an indentation model with multiple length scales is simulated with eight processors. The dimensions of the workpiece are 0.25 mm $\times$ 0.125 mm. Initially, the velocity of the indenter increases from 0 to 150 m/s linearly with time and is then kept constant. Five successive levels of refinement are used in this simulation. The smallest material point represents an area of 64 nm $\times$ 64 nm, and the largest material point covers an area of 1 μm $\times$ 1 μm. Each level is divided into 8 equal-sized patches for best load balance. Since the contact surface can evolve into several patches, a parallel solver is implemented to solve Eq. (11) to find the contact pressure based on the Portable, Extensible Toolkit for Scientific Computation (PETSc). An aluminum workpiece is chosen with the Young's modulus and Poisson's ratio of 70 GPa and 0.33, respectively. The maximum indentation depth was 9.8 μm in this simulation (i.e., 153 times the size of the finest material point). It took nine hours to simulate this problem with eight processors. Fig. 17 (a) gives the normal stress distributions and Fig. 17 (b) shows normal stress distribution for the finest two levels. The relative large deformation in this multiscale nanoindentation problem could not be handled by FEM due to excessive distortion

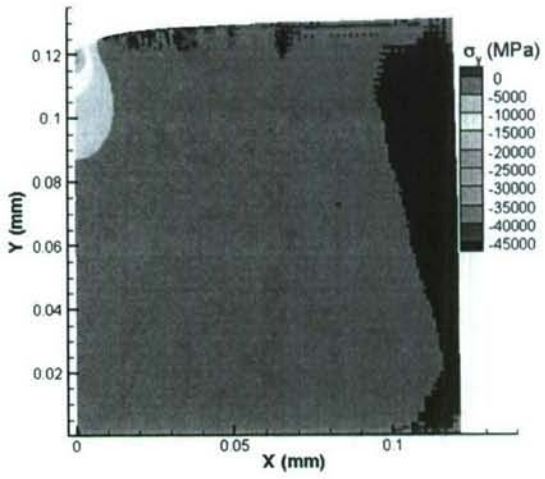


(a) FEM

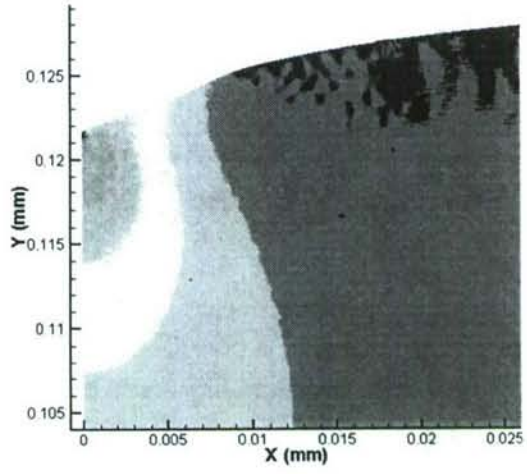


(b) GIMP

Figure 16 : Indentation load versus depth curves from FEM and GIMP with different grid sizes



(a) FEM



(b) GIMP

Figure 17 : Multiscale simulation of nanoindentation with five levels of refinement

in the FEM mesh. However, the parallel GIMP code was able to complete the entire loading/unloading processes without any difficulty. This example shows clearly the advantage of GIMP for multiscale simulations over FEM.

5 Conclusions

The following are specific conclusions based on the results of this investigation:

1. A 2D generalized interpolation material point (GIMP) method has been implemented to address problems, such as particle flying-off and alternating stress sign associated with conventional MPM in case of relatively large deformation.

2. To conduct multiple length scale simulations, a parallel computing scheme has been presented using GIMP under SAMRAI parallel computing environment in which multi-level grids are used for spatial and temporal refinements.

3. A refinement/coarsening algorithm, based on material points of GIMP in two grid levels, has been developed

for communication between neighboring grid levels of different refinements. With increase in the refinement levels, as well as decrease in the time step increments, the computational accuracy is greatly improved in the region of interest while the overall computational time is reduced. The computation at each grid level is performed recursively to ensure that the refinement and coarsening are performed when the two neighboring levels are synchronized.

4. 2D MPM and GIMP were applied to simple tension and indentation problems to validate the GIMP algorithm. GIMP results agree very well with FEM results for these two examples provided that the deformations are small. The noise and instability problems present in conventional MPM are not observed in the GIMP simulations.

5. As the deformation is increased, GIMP continued to execute while FEM aborted due to element distortion. Also GIMP results are stable. Thus GIMP is able to handle relatively large deformation problems.

6. For the nanoindentation problem, a GIMP algorithm for the contact between a rigid indenter and a deformable workpiece was developed. A reasonably good agreement between GIMP and FEM results was reached, validating the contact algorithm presented in this investigation.

7. Another nanoindentation example with multiple length scales from a few nanometers to sub-millimeters was simulated and numerical results validated the parallel GIMP computing with the use of SAMRAI.

Acknowledgement: The work was supported by a grant from the Air Force Office of Scientific Research (AFOSR) through a DEPSCoR grant (No. F49620-03-1-0281). The authors would like to thank Dr. Craig S. Hartley, Program Manager for Metallic Materials Program at AFOSR for his interest and support of this work. The authors also acknowledge Dr. Scott Bardenhagen for his introduction to the GIMP algorithm and valuable comments, and Dr. John A. Nairn for providing the 2D MPM code.

Disclaimer

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

References

Atluri, S.N.; Shen, S. (2002): The meshless local Petrov-Galerkin (MLPG) method: a simple & less-costly alternative to the finite element and boundary element methods.*CMES: Computer Modeling in Engineering & Sciences,* vol. 3, no. 1, pp. 11-51.

Bardenhagen, S.G. (2002): Energy conservation error in the material point method for solid mechanics. *Journal of Computational Physics,* vol. 180, pp. 383-403.

Bardenhagen, S.G.; Kober, E.M. (2004): The generalized interpolation material point method.*CMES: Computer Modeling in Engineering & Sciences,* vol. 5, no. 6, pp. 477-496.

Berger, M.J.; Oliger, J. (1984): Adaptive mesh refinement for hyperbolic partial differential equations. *Journal of Computational Physics,* vol. 82, pp. 484-512.

Hornung, R.D.; Kohn, S.R. (2002): Managing application complexity in the SAMRAI object-oriented framework. *Concurrency and Computation: Practice and Experience,* vol. 14, pp. 347-368.

Horstemeyer, M.F.; Baskes, M.I.; Prantil, V.C.; Philliber, J.; Vonderheide, S. (2003): A multiscale analysis of fixed-end simple shear using molecular dynamics, crystal plasticity, and a macroscopic internal state variable theory. *Modelling and Simulation in Materials Science and Engineering,* vol. 11, pp. 265-2861.

Hsien, S.H. (1997): Evaluation of automatic domain partitioning algorithms for parallel finite element analysis. *International Journal for Numerical Methods in Engineering,* vol. 40, pp. 1025-1051.

Hu, W.; Chen, Z. (2003): A multi-mesh MPM for simulating the meshing process of spur gears. *Computers & Structures*, vol. 81, pp. 1991-2002.

Kalia, R.K.; Nakano, A.; Greenwell, D.L.; Vashishta, P. (1993): Parallel algorithms for molecular dynamics simulations on distributed memory MIMD machines. *Supercomputer*, vol. 54, pp. 11-25.

Komanduri, R.; Lu, H.; Roy, S.; Wang, B.; Raff, L.M. (2004): Multiscale modeling and simulation for materials processing. *Proceedings of the AFOSR Metallic Materials (2306 AX) Grantees Meeting*, VA, USA.

Mackerle, J. (2003): FEM and BEM parallel processing: theory and applications-a bibliography. *Engineering Computation*, vol. 20, no. 4, pp. 436-484.

Oden, J.T.; Pires, E.B. (1983): Numerical analysis of certain contact problems in elasticity with non-classical friction laws. *Computers & Structures*, vol. 16, no. 1-4, pp. 481-485.

Shen, S.; Atluri, S. N. (2004a): Computational nano-mechanics and multi-scale simulation. *CMC: Computers, Materials, & Continua*, vol. 1, no. 1, pp. 59-90.

Shen, S.; Atluri, S. N. (2004b): Atomic-level stress calculation and continuum-molecular system equivalence. *CMES: Computer Modeling in Engineering & Sciences*, vol. 6, no. 1, pp. 91-104.

Shen, S.; Atluri, S. N. (2005): A tangent stiffness MLPG method for atom/continuum multiscale simulation. *CMES: Computer Modeling in Engineering & Sciences*, vol. 7, no. 1, pp. 49-67.

Sulsky, D.; Zhou, S.J.; Schreyer, H.L. (1995): Application of a particle-in-cell method to solid mechanics. *Computer Physics Communications*, vol. 87, pp. 236-252.

Sulsky, D.; Schreyer, H.L. (1996): Axisymmetric form of the material point method with applications to upsetting and Taylor impact problems. *Comput. Methods Appl. Mech. Eng.*, vol. 139, pp. 409-429.

Tan, H.; Nairn, J.A. (2002): Hierarchical, adaptive, material point method for dynamic energy release rate calculations. *Computer Methods in Applied Mechanics and Engineering*, vol. 191, pp. 2095-2109.

Tewary, V. K.; Read D. T. (2004): Integrated Green's function molecular dynamics method for multiscale modeling of nanostructures: application to Au nanoisland in Cu. *CMES: Computer Modeling in Engineering & Sciences*, vol. 6, no. 4, pp. 359-371.

Wissink, A.M.; Hysom, D.; Hornung, R.D. (2003): Enhancing scalability of parallel structured AMRA calculations. *Proc. 17th ACM International Conference on Supercomputing (ICS03)*, San Francisco, CA, pp. 336-347.

Zhong, Z.H. (1993): *Finite Element Procedures for Contact-Impact Problems*, Oxford University Press, New York.

Copyright © 2006 Tech Science Press CMES, vol.12, no.3, pp.213-227, 2006

Structured Mesh Refinement in Generalized Interpolation Material Point (GIMP) Method for Simulation of Dynamic Problems

Jin Ma, Hongbing Lu, and Ranga Komanduri[1]

Abstract: The generalized interpolation material point (GIMP) method, recently developed using a C^1 continuous weighting function, has solved the numerical noise problem associated with material points just crossing the cell borders, so that it is suitable for simulation of relatively large deformation problems. However, this method typically uses a uniform mesh in computation when one level of material points is used, thus limiting its effectiveness in dealing with structures involving areas of high stress gradients. In this paper, a spatial refinement scheme of the structured grid for GIMP is presented for simulations with highly localized stress gradients. A uniform structured background grid is used in each refinement zone for interpolation in GIMP for ease of generating and duplicating structured grid in parallel processing. The concept of influence zone for the background node and transitional node is introduced for the mesh size transition. The grid shape function for the transitional node is modified accordingly, whereas the computation of the weighting function in GIMP remains the same. Two other issues are also addressed to improve the GIMP method. The displacement boundary conditions are introduced into the discretization of the momentum conservation equation in GIMP, and a method is implemented to track the deformation of the material particles by tracking the position of the particle corners to resolve the problem of artificial separation of material particles in GIMP simulations. Numerical simulations of several problems, such as tension, indentation, stress concentration and stress distribution near a crack (mode I crack problem) are presented to verify this refinement scheme.

keyword: GIMP, Material Point Method, Mesh refinement, FEM, MPM

[1] Correspondence author, Tel: 405-744-5900; Fax: 405-744-7873; e-mail: ranga@ceat.okstate.edu. All authors are with the School of Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, OK 74078

1 Introduction

The material point method (MPM) uses a collection of material points, mathematically represented by Dirac delta functions to represent a material continuum (Sulsky, Zhou, and Schreyer (1995); Hu and Chen (2003); Guilkey and Weiss (2003)). A spatially fixed background grid, and interpolation between grid nodes and material points are introduced to track physical variables carried by the material points in the Lagrangian description. Field equations are solved on the background grid in the Eulerian description. Physical variables are interpolated from the solutions on the background grids to material points back and forth for solution and convection of physical variables. In general, the isoparametric shape functions, same as those used in the finite element method (FEM), are used. As the MPM simulation is independent of the background grid, a structured grid is usually employed for purposes of simplicity. The movement of the material points represents the deformation of the continuum. MPM has been demonstrated to be capable of handling large deformations in a natural way (Sulsky, Zhou, and Schreyer (1995)). However, primarily due to the discontinuity of the gradient of the interpolation function at the borders of the neighboring cells, artificial noise can be introduced when the material points move just across the grid cell boundaries, leading to simulation instability for MPM. The generalized interpolation material point (GIMP) method, introduced by Bardenhagen and Kober (2004) can resolve this problem. In GIMP a C^1 continuous interpolation function is used and each material point/particle occupies a region. GIMP has been demonstrated to be stable and capable of handling relatively large deformations (Ma, Lu, Wang, Roy, Hornung, Wissink, and Komanduri (2005)).

The current MPM typically uses a uniform background mesh for solving the field equations. However, this is not efficient when stress gradients are high such as stress concentrations in a plate with a hole, or the stress field of a workpiece under indentation. In contrast, transitional

mesh is effective in solving problems involving rapidly varying stress in an area. Wang, Karuppiah, Lu, Roy, and Komanduri (2005) have presented a method using an irregular background mesh to deal with problems involving rapidly varying stress, such as stress field near a crack. However, this approach does not use regular structured background mesh so that mesh generation encounters the same difficulty as FEM, and leads to the loss of some advantage of MPM on the ease of generating mesh for a complex problem.

The use of structured grid in GIMP has facilitated the implementation of GIMP in parallel processing. A refinement scheme based on splitting and merging material particles was proposed by Tan and Nairn (2002). Recently, a multilevel refinement algorithm has been developed for parallel processing using the structured adaptive mesh refinement application infrastructure (SAMRAI) (Hornung and Kohn (2002); Ma, Lu, Wang, Roy, Hornung, Wissink and Komanduri (2005)). The computational domain was divided into multiple nested levels of refinement. Each grid level is uniform but has a different cell size. Smaller material particles and smaller cell sizes are used in each finer level. Two neighboring levels are connected by overlapped material particles of the same size and data communication between levels is performed at predefined intervals. However, the refinement through material particles requires extra communication and simulation time. In this paper, a refinement for GIMP based on the transitional grid nodes is developed. This refinement is natural and does not involve extra simulation time. Moreover, the refined grid remains uniformly structured in each refinement region.

While the problem associated with artificial noise has been resolved with the use of GIMP method, it has been observed recently that material separation could occur if the deformation of the material particles was not tracked, Guilkey (2005). Tracking the deformation of material particles properly in GIMP is necessary especially when the material particles are stretched. In this paper, an approach is developed for tracking the particle deformation to resolve material point separation problem. This paper focuses on the refinement scheme for structured grid. Several numerical problems, such as tension, indentation, stress concentration and stress distribution near a crack (mode I crack problem) were simulated to verify this refinement algorithm, as well as to demonstrate the effectiveness of tracking particle deformations.

2 GIMP

For the purpose of completeness, the basic equations in GIMP (Bardenhagen and Kober (2004)) are summarized here. In dynamic simulations, the mass and momentum conservation equations are given by

$$\frac{d\rho}{dt} + \rho \nabla \cdot \mathbf{v} = 0, \text{ and} \tag{1}$$

$$\rho \mathbf{a} = \nabla \boldsymbol{\sigma} + \mathbf{b} \quad \text{in } \Omega, \tag{2}$$

where ρ is the material density, $\mathbf{a}$ is the acceleration, σ and $\mathbf{b}$ are the Cauchy stress and body force density, respectively. The displacement and traction boundary conditions are given as

$$\mathbf{u} = \bar{\mathbf{u}} \text{ on } \partial\Omega_u, \tag{3}$$

$$\boldsymbol{\tau} = \bar{\boldsymbol{\tau}} \text{ on } \partial\Omega_\tau, \tag{4}$$

where $\partial\Omega_u \subset \partial\Omega, \partial\Omega_\tau \subset \partial\Omega$ and $\partial\Omega_u \cap \partial\Omega_\tau = 0$. In variational form, the momentum conservation equation can be written as

$$\int_\Omega \rho \mathbf{a} \cdot \delta\mathbf{v} d\mathbf{x} = \int_\Omega \nabla\boldsymbol{\sigma} \cdot \delta\mathbf{v} d\mathbf{x} + \int_\Omega \mathbf{b} \cdot \delta\mathbf{v} d\mathbf{x} - \alpha \int_{\partial\Omega_u} (\mathbf{u} - \bar{\mathbf{u}}) \cdot \delta\mathbf{v} d\mathbf{x}, \tag{5}$$

where $\delta\mathbf{v}$ is an admissible velocity field, α is a penalty parameter we introduce herein to impose the essential boundary conditions and $\alpha >> 1$, Atluri and Zhu (1998), Atluri and Zhu (2000). Applying the chain rule, $\nabla \cdot \boldsymbol{\sigma} \cdot \delta\mathbf{v} = \nabla \cdot (\boldsymbol{\sigma} \cdot \delta\mathbf{v}) - \boldsymbol{\sigma} : \nabla\delta\mathbf{v}$, and the divergence theorem, Eq. (5) can be written as

$$\begin{aligned} \int_\Omega \rho \mathbf{a} \cdot \delta\mathbf{v} d\mathbf{x} + \int_\Omega \boldsymbol{\sigma} : \nabla\delta\mathbf{v} d\mathbf{x} &= \int_\Omega \mathbf{b} \cdot \delta\mathbf{v} d\mathbf{x} + \int_{\partial\Omega_\tau} \bar{\boldsymbol{\tau}} \cdot \delta\mathbf{v} dS \\ &+ \int_{\partial\Omega_u} \boldsymbol{\tau}_u \cdot \delta\mathbf{v} dS - \alpha \int_{\partial\Omega_u} (\mathbf{u} - \bar{\mathbf{u}}) \cdot \delta\mathbf{v} dS \end{aligned} \tag{6}$$

where τ_u is the resultant traction due to the displacement boundary condition on $\partial\Omega_u$. In GIMP, the domain Ω is

discretized into a collection of material particles, with Ω_p as the domain of particle p. The physical quantities, such as the mass, stress and momentum can be defined for each particle. For example, the momentum for particle p can be expressed as $\mathbf{p}_p = \int_{\Omega_p} \rho(\mathbf{x})\mathbf{v}(\mathbf{x})\chi_p(\mathbf{x})d\mathbf{x}$, where $\mathbf{v}(\mathbf{x})$ is the velocity and $\chi_p(\mathbf{x})$ is the particle characteristic function. The momentum conservation equation can be discretized as

$$\begin{aligned}\sum_p \int_{\Omega\cap\Omega_p} \frac{\dot{\mathbf{p}}_p\chi_p}{V_p}\cdot\delta\mathbf{v}d\mathbf{x} + \sum_p \int_{\Omega\cap\Omega_p} \sigma_p\chi_p : \delta\mathbf{v}d\mathbf{x} \\ = \sum_p \int_{\Omega\cap\Omega_p} \frac{m_p\chi_p}{V_p}\mathbf{b}\cdot\delta\mathbf{v}d\mathbf{x} + \sum_p \int_{\partial\Omega_\tau\cap\Omega_p} \bar{\boldsymbol{\tau}}\cdot\delta\mathbf{v}d\mathbf{x} \\ + \sum_p \int_{\partial\Omega_u\cap\Omega_p} \boldsymbol{\tau}_u\cdot\delta\mathbf{v}dS - \alpha\sum_p \int_{\partial\Omega_u\cap\Omega_p} (\mathbf{u}-\bar{\mathbf{u}})\cdot\delta\mathbf{v}dS \quad (7)\end{aligned}$$

where $V_p = \int_{\Omega\cap\Omega_p} \chi_p(\mathbf{x})d\mathbf{x}$ is the particle volume. Introducing a background grid and the grid shape function $S_i(\mathbf{x})$ that satisfies partition of unity $\sum_i^{S_i(\mathbf{x})} = 1$, the admissible velocity field can be represented by the grid nodal data as $\delta\mathbf{v} = \sum_i \delta\mathbf{v}_i S_i(\mathbf{x})$. Without the loss of generality, we take $\mathbf{u}$ in Eq. (7) to be the displacement of the boundary particles at the current time step and $\boldsymbol{\tau}_u = \sigma_p\mathbf{n}_u$ where $\mathbf{n}_u$ is the unit outward normal to $\partial\Omega_u$. The momentum conservation, Eq. (7), can eventually be written for each node i as

$$\dot{\mathbf{p}}_i = \mathbf{f}_i^{int} + \mathbf{f}_i^b + \mathbf{f}_i^\tau + \mathbf{f}_i^u, \quad (8)$$

where the time rate of change of nodal momentum $\dot{\mathbf{p}}_i = \sum_p \bar{S}_{ip}\mathbf{p}_p/\Delta t$, the nodal internal force vector $\mathbf{f}_i^{int} = -\sum_p \boldsymbol{\sigma}_p\cdot\overline{\nabla S}_{ip}V_p$, the nodal body force vector $\mathbf{f}_i^b = \sum_p m_p\mathbf{b}\bar{S}_{ip}$ and the nodal traction force vector $\mathbf{f}_i^\tau = \sum_p \int_{\partial\Omega_\tau\cap\Omega_p} \bar{\boldsymbol{\tau}}S_i(\mathbf{x})dS$. $\mathbf{f}_i^u$ is the force vector induced by the essential boundary condition given by

$$\mathbf{f}_i^u = \sum_p \int_{\partial\Omega_u\cap\Omega_p} \boldsymbol{\sigma}_p\mathbf{n}_u S_i(\mathbf{x})dS - \alpha\sum_p \int_{\partial\Omega_u\cap\Omega_p} (\mathbf{u}_p-\bar{\mathbf{u}})S_i(\mathbf{x})dS. \quad (9)$$

$\bar{S}_{ip}$ is weighting function between particle p and node i given as

$$\bar{S}_{ip} = \frac{1}{V_p}\int_{\Omega\cap\Omega_p} \chi_p(\mathbf{x})S_i(\mathbf{x})d\mathbf{x}. \quad (10)$$

The weighting function in GIMP is C^1 continuous and satisfies partition of unity. The momentum conservation, Eq. (8), can be solved at each node to update the nodal momentum, acceleration, and velocity. These updated nodal quantities can be interpolated to the material particles to update the particles, as given by Bardenhagen and Kober (2004). It may be noted that the mass of each material particle does not change, so that the mass conservation equation is satisfied automatically.

In the discretization of the weak form of the momentum conservation equation, a background grid is used. However, the computation is independent of the grid from one increment to another. Hence a spatially fixed structured grid can be used for convenience. In the background grid, no nodal connectivity is required and the integration is never performed on the element domain. Similar characteristics have been reported for other meshless methods, such as the meshless local Petrov-Galerkin (MLPG) method, Atluri and Shen (2002).

For a uniform structured grid, the grid shape function in 3D is defined as the product of three nodal tent functions (Bardenhagen and Kober (2004))

$$S_i(\mathbf{x}) = S_i^x(x)S_i^y(y)S_i^z(z), \quad (11)$$

in which the nodal tent functions are in the same form, e.g.,

$$S_i^x(x) = \begin{cases} 0 & x-x_i \le -L_x \\ 1+(x-x_i)/L_x & -L_x \le x-x_i \le 0 \\ 1-(x-x_i)/L_x & 0 \le x-x_i \le L_x \\ 0 & L_x \le x-x_i \end{cases} \quad (12)$$

Fig. 1 shows one 2D grid cell with four nodes. In this paper, the particle characteristic function of the material particle located at (x_p, y_p) is taken as

$$\chi_p(\mathbf{x}) = \chi_p^x(x)\chi_p^y(y), \quad (13)$$

where $\chi_p^x(x) = H[x-(x_p-l_x)] - H[x-(x_p+l_x)]$ and H denotes the step function.

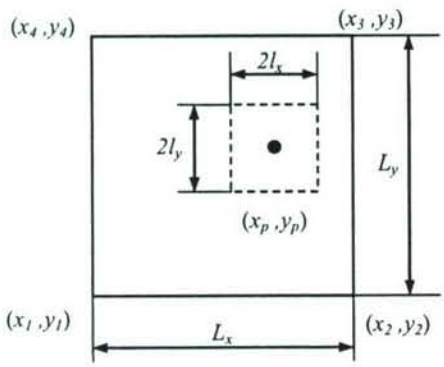


Figure 1 : 2D representation of a particle and a grid cell

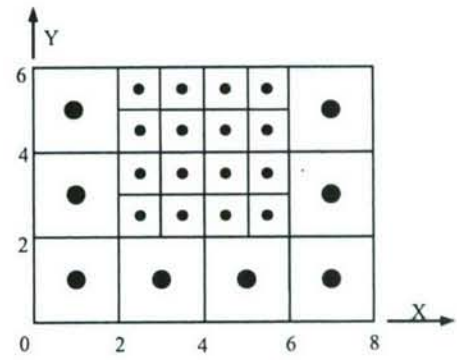


Figure 2 : Refinement of structured grid with a refinement ratio of two

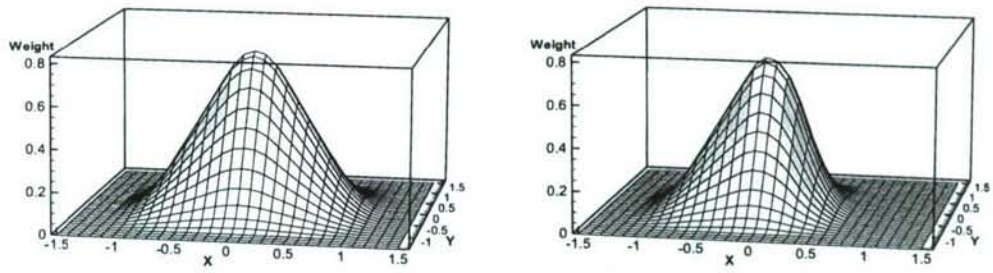
3 Structured mesh refinement

In this section, a refinement scheme for a structured mesh in GIMP is described. Since GIMP shares some characteristics with meshless methods, we expect the GIMP method to have h convergence in simulation (Atluri and Shen (2002)). The momentum conservation equation is essentially solved at each node (see Eq. (8)). Therefore, the number of equations to be solved is the same as the number of nodes. Finer grid and smaller material particles will lead to more accurate results. In some simulations, high stress gradients exist in small regions. For instance, in indentation with a sharp tip, the stress gradient is high in the workpiece beneath the indenter tip. In simulation of fracture problems, the stress gradient at the crack tip is high and of particular interest. Consequently, finer grid is needed for these regions; but away from these regions, a coarser grid can be used to reduce the computational cost. In conclusion, a uniform grid can be either too computationally expensive if it is too fine, or inaccurate, if it is coarse. A non-uniform grid with refinement can provide accurate results while minimizing the overall computational time.

Grid refinement should maintain the same characteristics of the structured grid as much as possible, in order to replicate the grid generation in parallel processing. The proposed refinement scheme is illustrated in Fig. 2 with one particle per cell assigned. The material particles that fill each cell are square in nature but denoted as circles for clarity. To understand this grid, one can consider that there are two overlapped structured grids. The coarse grid covers a rectangular region from (0, 0) to (8, 6) and the fine grid covers a region from (2, 2) to (6, 6). For each grid, the shape function can be evaluated from Eq. (12). To be consistent with any other general refinement, it is required that the region of the fine grid to be smaller than the coarse grid.

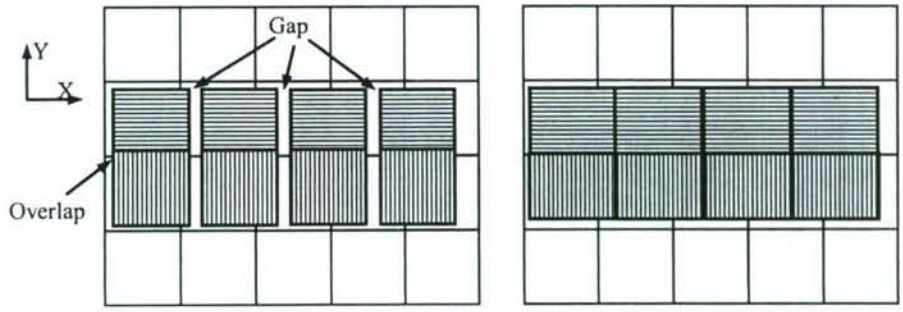
When these two grids are merged into one, the shape function and the weighting function for the nodes at the boundary of the finer grid, for example, the nodes at (2, 2) and (2, 3) should be changed. These nodes are called transition nodes. To facilitate the computation of the interpolation function, we define an influence zone for each node, denoted as $[L_x^-, L_x^+, L_y^-, L_y^+]$ in 2D or $[L_x^-, L_x^+, L_y^-, L_y^+, L_z^-, L_z^+]$ in 3D. The symbols in the square bracket define the size of the influence zone, whereas the subscript denotes the coordinate axis and the superscript denotes the direction of the axis. For example, L_x^- and L_x^+ represent the sizes in the negative and positive X direction, respectively. The influence zone for each node in 2D is rectangular and it extends to the next immediate grid line to the left, right, bottom and top of the node. If no more grid lines exist in any direction, such as the boundary nodes, the size is zero in that direction. For example, in the refined grid in Fig. 2, the influence zones for the nodes at (2, 3) and (2, 4) are both [2, 1, 1, 1]. The influence zone for the node at (2, 2) is [2, 1, 2, 1]. Based on this definition, the influence zone for this node in the coarse grid is [2, 2, 2, 2], and in the uniformly fine grid is [0, 1, 0, 1].

Based on the influence zone, the nodal tent function in each direction can be modified as, for example, in the X



(a) Influence zone: [1, 1, 1, 1] (b) Influence zone: [1, 0.5, 1, 1]

Figure 3 : Effect of influence zone on the weighting functions



(a) No particle deformation (b) Actual particle deformation

Figure 4 : Schematics showing overlaps and gaps that may occur when particle deformation is not tracked

direction,

$$S_i^x(x) = \begin{cases} 0 & x - x_i \le -L_x^- \\ 1 + (x - x_i)/L_x^- & -L_x^- \le x - x_i \le 0 \\ 1 - (x - x_i)/L_x^+ & 0 \le x - x_i \le L_x^+ \\ 0 & L_x^+ \le x - x_i \end{cases} \quad (14)$$

Eq. (14) can be substituted into the grid shape function (Eq. (11)), and the weighting function between the particle p and the node i can be evaluated as

$$\overline{S}_{ip}^x = \begin{cases} 0 & B \le -L_x^- \text{ or } A \ge L_x^+ \\ \frac{b - a + (b^2 - a^2)/(2L_x^-)}{2l_p} & b \le 0 \\ \frac{b - a - (b^2 - a^2)/(2L_x^+)}{2l_p} & a \ge 0 \\ \frac{b - a - a^2/(2L_x^-) - b^2/(2L_x^+)}{2l_p} & otherwise \end{cases} \quad (15)$$

where $A = x - x_i - l_p$, $B = x - x_i + l_p$, $a = \max(A, -L_x^-)$ and $b = \min(B, L_x^+)$. When $L_x^- = L_x^+$, Eqs. (14) and (15) are degraded to the cases for uniform grid. Without detailed proof, the grid shape function and the weighting function still satisfy partition of unity. Similarly, the gradient of the modified weighting function can be computed.

It may be noted that in Fig. 2 the refinement ratio is two, i.e., the length of a side of a coarse cell is twice that of the fine cell. To maintain the convenience of the structured grid, only integer refinement ratio should be used. All nodal positions can be computed from the domain of each grid and the cell sizes. The proposed refinement scheme can be applied to any integer refinement ratio and for multiple times for successive refinements. As an example, the weighting function between a particle of size 0.5×0.5 and a node at (0, 0) with an influence zone of [1, 1, 1, 1] is shown in Fig. 3(a). The particle is on the X-Y plane and the weighting function is computed at each particle position. For comparison, the influence zone is changed to [1, 0.5, 1, 1], representing a transitional node,

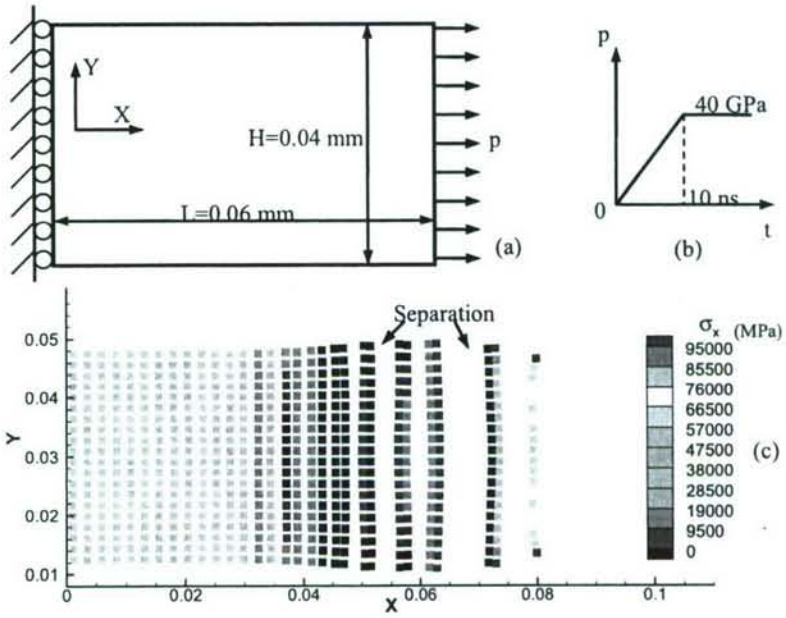


Figure 5 : Simulation showing separation when the particle deformation is tracked by strain

while other conditions are the same. The weighting function for this case is plotted in Fig. 3(b). It can be seen that the weighting function for the transitional node is still C^1 continuous.

4 Numerical simulations

4.1 Tracking particle deformation

Prior to presenting the results of structured mesh refinement, tracking particle deformation is addressed first since this is necessary in later simulations to achieve accuracy. The material particles are initialized into regular shapes, normally square and cube for 2D and 3D simulations, respectively. All the physical quantities in a particle domain are considered to be uniform. The shape of the particle changes during deformation. So, it is important to track the deformation of each particle. Fig. 4 illustrates the deformation of the particles in 2D when the particles are stretched in the X-direction. If the particle deformation is not tracked, gaps will form between neighboring particles in the X-direction, as shown in Fig. 4 (a). Due to Poisson's effect, there will be overlapping between particles in the Y-direction, if the particles do not follow the deformation of the materials properly. When the stretch and gaps are large enough, the particles would be separated. Fig. 4 (b) shows the correct deformation in which contiguous particles remain contiguous after deformation.

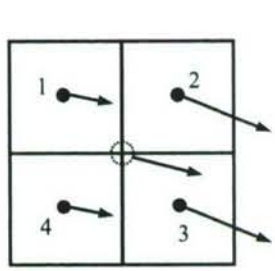


Figure 6 : Velocity field of a continuum region (the arrows represent both direction and magnitude)

To track the particle deformation, a convenient approach is to calculate the deformed particle shape based on strain history. Since the linear strain increment is computed in GIMP, the effectiveness and validity of tracking particle deformation based on strain would be limited to relatively small deformations. As an example, Fig. 5 (a) shows the simulation of a uniaxial tension problem under plane strain conditions. The material is silicon. Its mass density is 2.71 g/cm^3, Young's modulus 175.8 GPa, and Poisson's ratio 0.28. The background grid size is 0.002 $\times$ 0.002 mm^2. One particle per cell is assigned initially and the time step is 0.02 ns (1 ns = 10^{-9} s). The applied

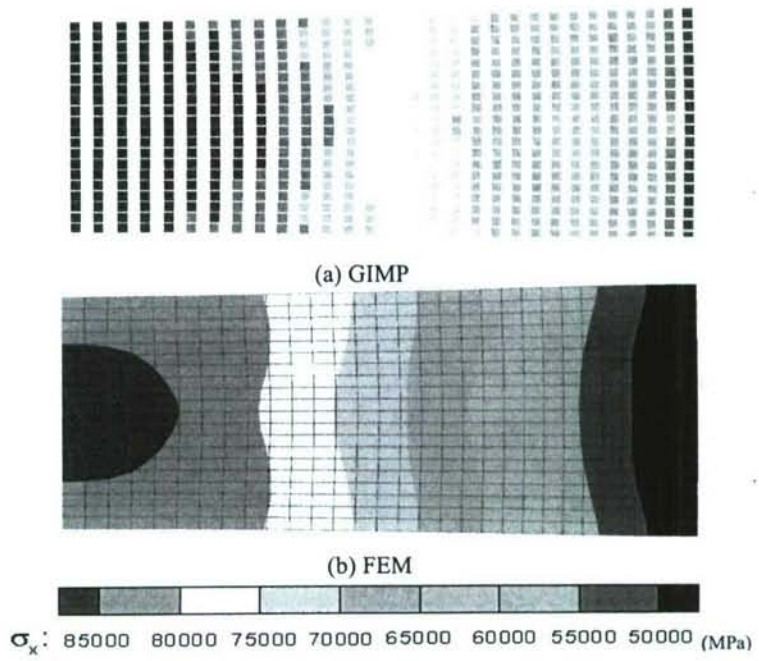


Figure 7 : GIMP results with tracking deformation of corners and their comparison with FEM

pressure increases linearly with time from 0 to 10 ns and is then maintained constant, as shown in Fig. 5 (b). The elongation in the X-direction of the particle is computed as $(1+\varepsilon_x)l_x^0$, where $l_x^0$is the initial length of the particles. Separation of the particles occurred at ~25% strain at 6 ns before the full pressure was applied, as shown in Fig. 5 (c). Similar problems have been reported by Guilkey (2005).

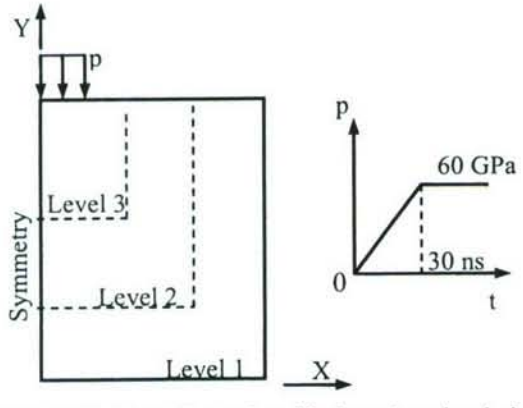


Figure 8 : Two-dimensional indentation simulation

Tracking particle deformation by strain could be more effective if nonlinear strain is used in the GIMP method. However, recovery of the deformed shape based on nonlinear strain involves additional complications. Based on the GIMP algorithm, there is another convenient approach to track the particle deformation. Numerically, the displacement and velocity of each particle in GIMP are computed at the center of the particle to represent the entire particle domain. However, in reality, the velocity and deformation at the corners of a particle can be different from the center. Fig. 6 shows four 2D contiguous particles sharing one common corner point at the middle. This corner point should have unique displacement and velocity. As a result, it is helpful to track the displacement and velocity of each corner to track the particle deformation.

It is not difficult to compute the velocity of the particle corner given its location. It is computed from the interpolation from the background grid, similar to the center of the particle. For a 2D particle, in addition to updating the position of the center of the particle, the positions of the four corners are updated at each increment. To compute the weighting function for the corners, a fictitious

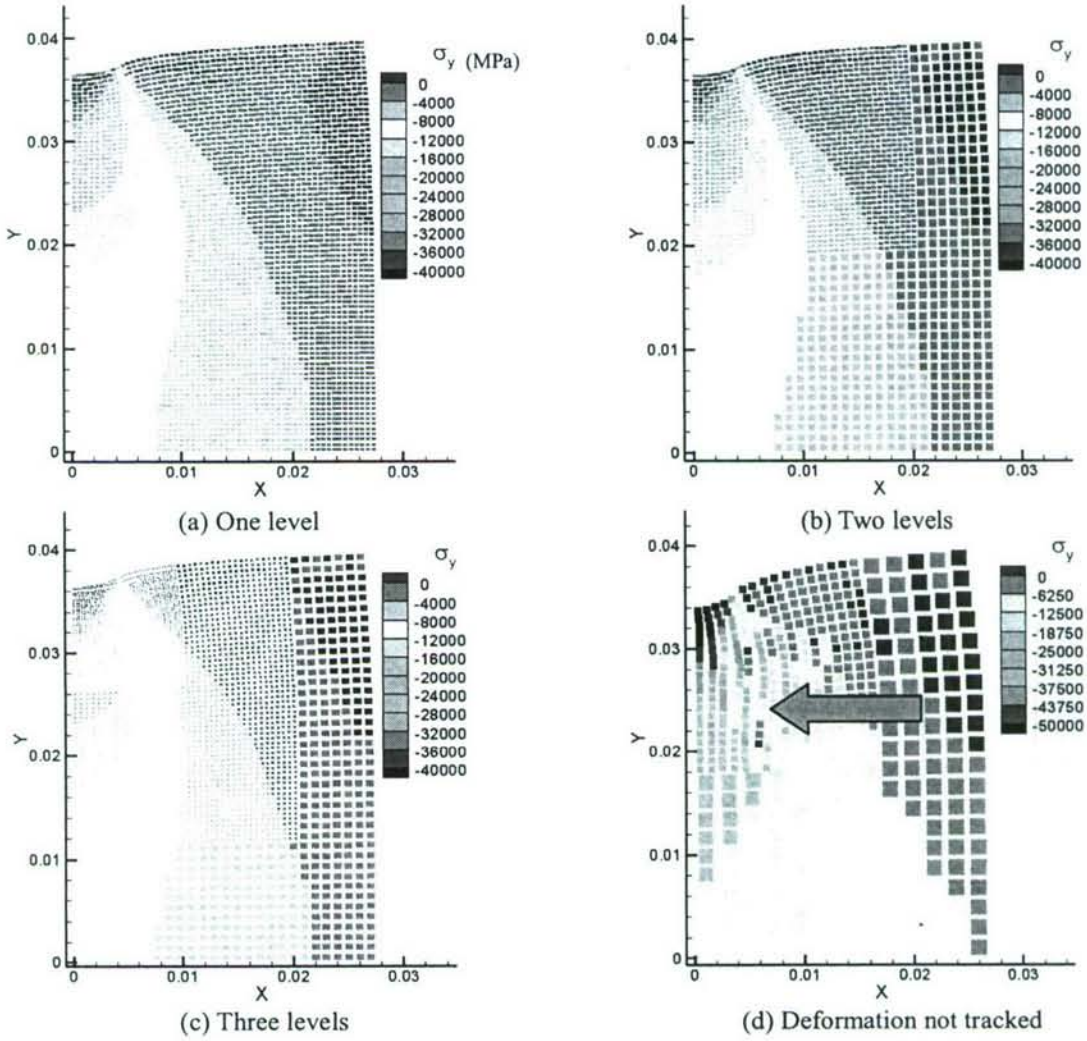


Figure 9 : Comparison of the stress distributions at different levels of refinements in force indentation

size can be assigned to each corner. Numerical simulation shows that the result is not sensitive to this size in the range of 10% to 80% of the initial particle size. The new particle shape can be obtained by connecting the four corners with straight lines. In order to avoid numerical integration of the interpolation function, it is assumed that the deformed material particle shape is rectangular with edges parallel to the coordinate axes. The size of the rectangle is, therefore, determined from the extent of the corners. As will be demonstrated later in this section, this assumption does not introduce any significant error while it can greatly improve the efficiency of the GIMP algorithm.

Using this approach to track the particle deformation, the problem in Fig. 5 (a) was simulated again with GIMP and the results at 20 ns are plotted in Fig. 7 (a). No separation of particles was seen during the entire simulation up to 50% strain. It is noted that each material particle is plotted as a square of the same size and the particle deformation is not shown due to software limitations on visualization. For comparison, the same problem is simulated using FEM (Abaqus/Explicit) and the FE result is shown in Fig. 7 (b). It can be seen that these two sets of results agree reasonably well with each other; the maximum difference in maximum tensile stress is $\sim$8%.

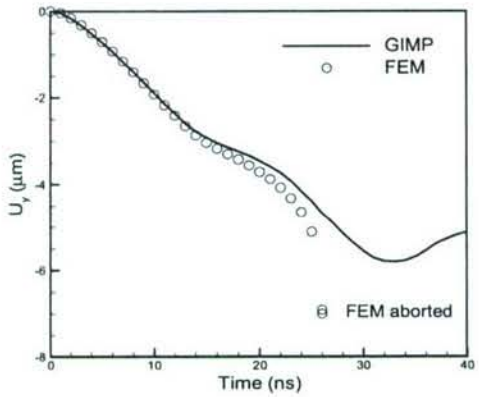


Figure 10 : Comparison of displacement history with FE

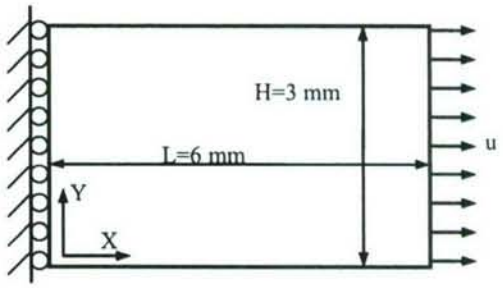
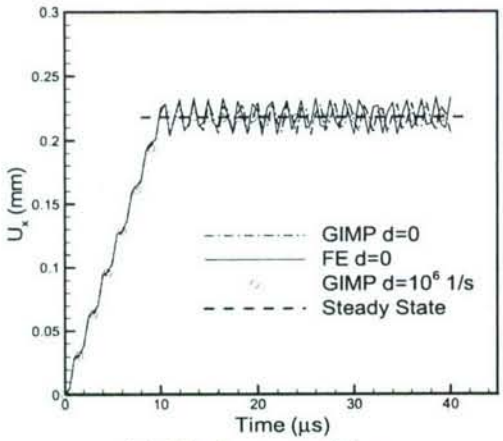
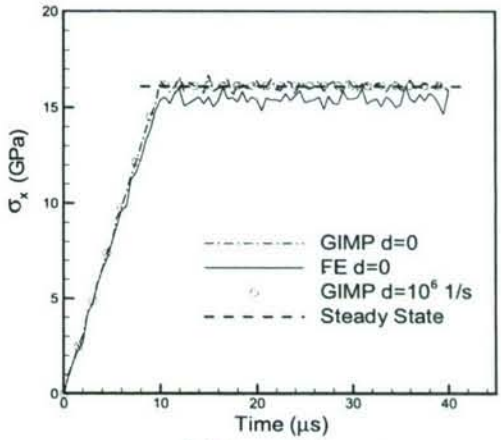


Figure 11 : A simple tension problem with displacement boundary conditions



(a) Displacement history



(b) Stress history

Figure 12 : Comparisons of the displacement and stress

4.2 Indentation problem

To verify the refinement algorithm, a 2D indentation problem was simulated. Pressure is applied at the top of the workpiece, as shown in Fig. 8. The dashed lines indicate the borders of refinement levels. The work material is silicon with the properties the same as those given in the previous section. Due to symmetry with respect to the Y-axis, only half of the workpiece is modeled. The size of the model is 0.027 × 0.04 mm^2. The pressure increases linearly with time from t = 0 - 30 ns and is then kept constant at 60 GPa.

Several simulations were performed under different settings for the purpose of comparison. In the first simulation, a uniform grid with a cell size of 0.001 × 0.001 mm^2 is used and the time step is 20 ps (1 ps = 10^{-12} s). The stress distribution in the workpiece at t = 20 ns is shown in Fig. 9(a). In this figure, the units of length and stress are mm and MPa, respectively. In the second simulation, as indicated in Fig. 9(b), two levels of refinements are used with the refinement ratio to be 2. The cell lengths are 0.002 mm and 0.001 mm for the first and second levels, respectively. The fine level covers a rectangular area of the workpiece from (0, 0.02) to (0.02, 0.04). The grid is fixed in space; the material particles initially in the fine region move to the coarse region during deformation. As shown in Fig. 9(b), some fine material particles have moved below the line Y = 0.02 mm. It

may be noted that in Fig. 9 the material particles are plotted as squares corresponding to their initial sizes. Gaps between particles are intentionally shown to depict the particle sizes.

Three levels of refinements are used in the third simulation of the same problem. In this simulation, the time step is 10 ps and the results are shown in Fig. 9(c). The stresses agree very well with the previous two simulations. Fig. 9(d) shows the result when the particle deformation is not tracked. A severe material separation was observed at t = 36 ns, as indicated by the arrow. Additionally, the displacement history of the particle in the middle of the top surface for the third simulations is shown in Fig. 10. The same result of the integration point of the element in the middle of the top surface in FE simulation using ABAQUS/Explicit is also shown in Fig. 10 for comparison. It can be seen that the displacement compares well with FE up to t = ~20 ns. After this time, FE simulation aborted due to mesh distortion. This demonstrates the capability of GIMP using a grid with structured refinement in handling large deformations.

It may be noted that if the exterior corners of the surface particles are tracked from the nodal interpolations in the same way as the interior corners, simulations tend to become unstable due to erroneous surface corner displacements. This problem was observed to be strictly and consistently associated with the particles with external tractions applied. It is caused by insufficient nodal interpolation. To eliminate this problem, the exterior corners of the surface particles were tracked by strain only, as used in these simulations.

4.3 *Validation of the displacement boundary condition*

We next describe the results on the verification of the displacement boundary conditions. For dynamic simulations, an artificial damping may be introduced. With damping, the nodal momentum can be updated as

$$\Delta \mathbf{p}_i = (\mathbf{f}_i^{int} + \mathbf{f}_i^b + \mathbf{f}_i^\tau + \mathbf{f}_i^u - d\mathbf{p}_i)\Delta t, \tag{16}$$

where d is the artificial damping coefficient.

A rectangular slab is fixed on the left and a displacement boundary condition is applied on the right, as shown in Fig. 11. The material is silicon and its properties are given in the previous section. The cell size in this simulation is 0.5 × 0.5 mm^2 and four particles are assigned to each cell initially. The time step is 5 ns. The prescribed displacement increases linearly with time to 0.5 mm at t = 10 μs, which corresponds to a velocity of 50 m/s, and remains constant thereafter. This problem is simulated in FE using Abaqus/Explicit for comparison. The displacement in the X-direction, U_x, for a particle initially centered at (2.625, 1.625) as a function of time is plotted in Fig. 12 (a). It can be seen that without damping, the vibrations of displacement from both FEM and GIMP simulation are in phase before t = 14 μs, but out of phase afterwards. The dashed line represents the steady state displacement at this point. It can be seen that when an artificial damping of 10^6 s^{-1} is used, the GIMP solution converges quickly. The error of the converged displacement is 0.46%. Fig. 12 (b) shows the comparison of the normal stress in the X-direction. Good comparison between GIMP and FE has been obtained. With damping, the GIMP solution converges to the analytical solution for static simulations.

4.4 *Stress concentration problem*

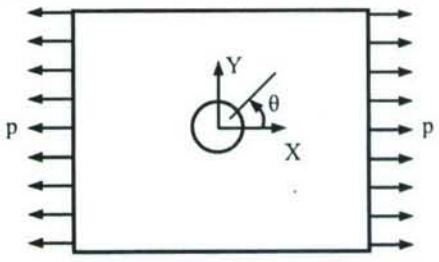
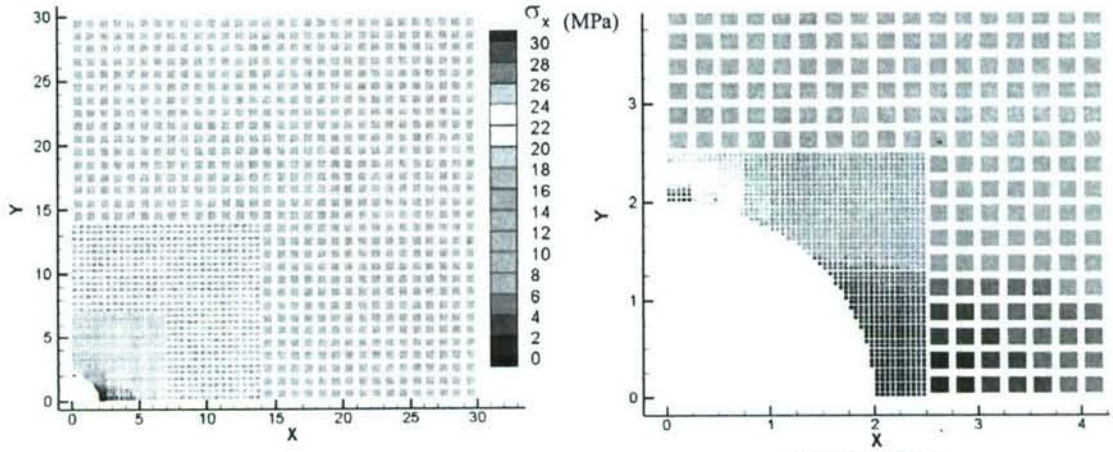


Figure 13 : A plate with a circular hole subjected to tension

Fig. 13 shows a copper plate (60 × 60 mm^2) with a central hole (4 mm diameter) subjected to a distributed load. This problem is simulated using GIMP as a dynamic problem with a damping factor of 1000 s^{-1} and three levels of refinement. The cell sizes at these three levels are 1.0 mm, 0.5 mm, and 0.25 mm, respectively. One particle is assigned to each cell not adjacent to the hole initially. The cells close to the circular hole are assigned 25 particles each to model the circular edge more accurately with the use of square particles. It may be noted that all the particles occupy square areas initially. The time step is 10 ns and the applied distributed load intensity p = 10 MPa. The stress distribution after 4000 steps is shown in Fig. 14 (a) and the area close to the



(a) Overall (b) Magnified

Figure 14 : Normal stress in the X-direction with p = 10 MPa

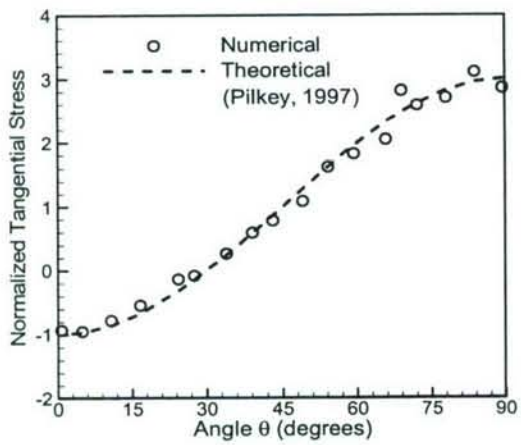


Figure 15 : Normalized tangential stress along the circumference of the hole

hole (bottom left corner in Fig. 14 (a)) is magnified in Fig. 14 (b). The normal stress of the particle at the top of the hole is 30.7 MPa when the applied tension is 10 MPa. This gives a stress concentration factor of 3.07. The tangential stress of the particles along the hole circumference, normalized by the applied pressure, is plotted in Fig. 15 in comparison with the theoretical value, Pilkey (1997). A good agreement between the numerical simulation and theoretical value is obtained. This demonstrates that the GIMP refinement algorithm is effective for problems involving significant stress gradients. Furthermore, with the use of small square particles, GIMP is capable of modeling curved surfaces.

4.5 Static stress intensity factor

We next use the GIMP refinement algorithm to determine the stress field and stress intensity factor for a mode I crack problem to determine the capability of the GIMP refinement algorithm in simulation of stress distribution near a crack. Guo and Nairn (2003) have successfully extended the MPM method to compute stress distribution in a plate with explicit cracks using multiple nodal fields along the crack surface. The physical quantities of material particles on each side of the crack were interpolated using variables in the field on that side of crack surface. In their simulation, a uniform mesh was used. Since the stress gradient at the crack tip is very high, a refined mesh near the crack tip and a coarse mesh in the far field should lead to savings in computational time while maintaining the same accuracy with the use of a uniform fine mesh. The fracture problem is thus an appropriate problem to evaluate the refined GIMP algorithm. For this purpose, we model the same fracture problem by Guo and Nairn (2003) using MPM, that is, a double cantilever beam (DCB) with a crack as shown in Fig. 16. In the area close to the crack tip, finer meshes are used, while in the area far away from the crack tip, coarse meshes are used. The thickness of the plate is 1 mm, thereby

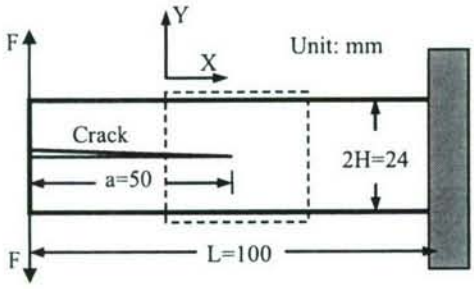


Figure 16 : Geometry of a double cantilever beam with a crack

justifying a plane stress condition for this problem. The material of the DCB is considered to be homogeneous, isotropic, and linearly elastic. Its density, Young's modulus, and Poisson's ratio are 1500 kg/m^3, 2300 GPa, and 0.33, respectively. The applied force is $F = 4 \times 10^{-4}$ N and this results in a mode I crack problem. The static stress intensity factor for the DCB can be calculated using the following equation (Kanninen 1973)

$$K_I = 2\sqrt{3}\frac{F(a+3H/2)}{H^{3/2}}. \tag{17}$$

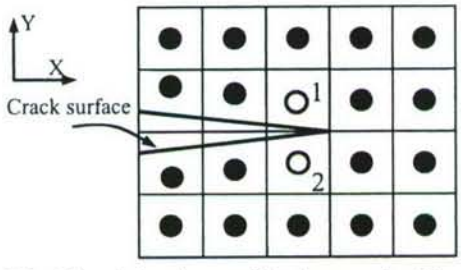
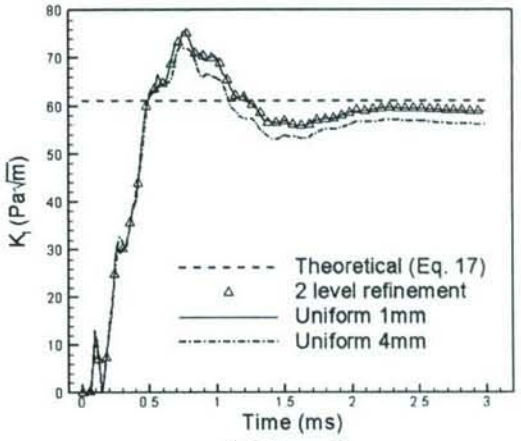
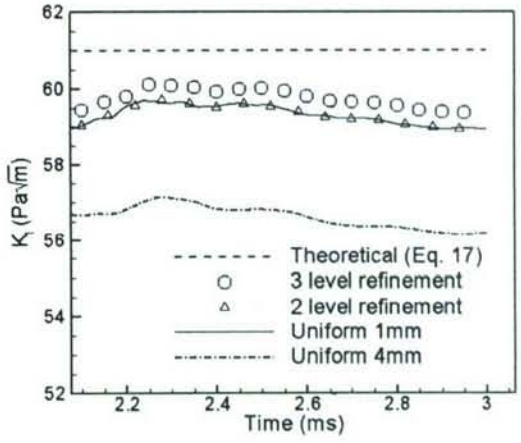


Figure 17 : Material points and background grid around the crack tip

This problem has been simulated using MPM with uniform grids of three sizes, 4 mm, 2 mm, and 1 mm (Guo, Nairn (2003)). They have computed the energy release rate and determined the stress intensity factor from J-integral. Their results indicate that the stress intensity factor determined from finer grid is more accurate and closer to the theoretical value.



(a) Overall



(b) Magnified

Figure 18 : Computed stress intensity factor as a function of time

In the mesh refinement GIMP algorithm used in this study, the energy release rate was computed using the virtual crack closure technique (Rybicki and Kanninen (1977); Wang, Karuppiah, Lu, Roy, and Komanduri (2005)). The energy released during an infinitesimal crack growth of Δa is assumed to be the energy required to close the crack to its initial size. Hence, the energy release rate G is determined by

$$G = \lim_{\Delta a \to 0} \frac{1}{2\Delta a} \int_0^{\Delta a} (\boldsymbol{\sigma}\mathbf{n}) \cdot \Delta\mathbf{u} da. \tag{18}$$

where $\mathbf{n}$ is the unit normal vector at the crack surface.

For the 2D mode I crack in the X-direction,

$$G_I = \lim_{\Delta a \to 0} \frac{1}{2\Delta a} \int_0^{\Delta a} \sigma_{yy} \cdot \Delta u_y da. \tag{19}$$

In GIMP, the energy release rate can be computed as

$$G_I = \frac{1}{2t\Delta a} F_{tip}(u_y^1 - u_y^2), \tag{20}$$

where the superscripts 1 and 2 denote the two material particles immediately to the left of the crack tip as shown in Fig. 17, Δa is the X distance between particle 1 and the crack tip, t is the thickness of the beam. F_{tip} is the nodal force to hold the crack tip together (Rybicki and Kanninen (1977)) and is computed as the crack tip nodal force from one side of the crack in this simulation.

The mode I stress intensity factor for the static crack is given by

$$K_I = \sqrt{GE}. \tag{21}$$

GIMP simulations were carried out using four material points per cell. The time step is 0.1 μs, and a damping coefficient of 4000 s^{-1} is used to allow the results to converge to the static data. The computed stress intensity factors using uniform grids of 4 mm and 1 mm, respectively, are plotted in Fig. 18 (a) and the theoretical value calculated from Eq. (17) is also shown for comparison. For simulation with refinement, two levels of refinement were used, i.e., 2 mm grid size for the coarse level and 1 mm grid size for the fine level, and the extent of the fine level is indicated by the dashed square in Fig. 16. The computed stress intensity factor using the refinement algorithm is also plotted in Fig. 18 (a). It is seen that the results from two levels of refinement are identical to the results from one uniform fine level with 1 mm cell size. Moreover, the simulation time using the refinement algorithm is 38% shorter than that of one uniform fine level. The computed stress intensity factor became even closer to the theoretical value when a third level of refinement was added for the crack tip as shown in Fig. 18 (b). For the case of using uniform grid of 2 mm, the computed stress intensity factor using GIMP and MPM (Guo and Nairn 2003) are plotted in Fig. 19. The MPM results were computed using a damping factor of 1000 s^{-1}, and therefore, more oscillations can be seen as expected.

Fig. 20 shows the distribution of σ_y at t = 2.5 ms when the force applied on the beam was changed to $F = 4$ N,

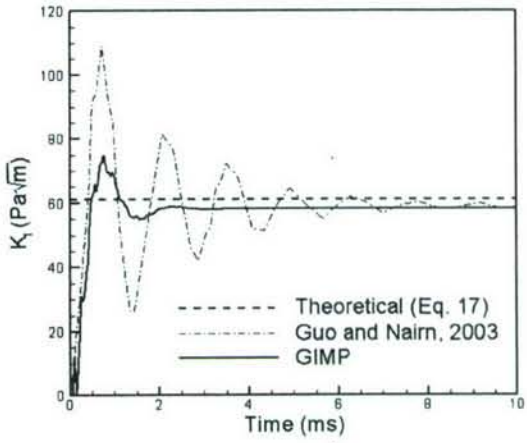


Figure 19 : Comparison of the stress intensity factor with MPM results

10000 times of the previous value, with other parameters the same. In this simulation, three levels of refinements with cell sizes of 1 mm, 0.5 mm and 0.25 mm, were used and the different density of material points in each level can be clearly seen in the figure. The computed stress intensity factor, scaled by 10000 times, still compares well with the theoretical value. It is noted that deformation near the crack surfaces has caused material points crossing cell boundaries, a situation where MPM would give numerical noise such as alternating stress signs. Despite the extent of the deformation, and material points crossing cell boundaries, GIMP method with the use of particle deformation tracking still gives correct results, further verifying the structured refinement methods developed herein.

5 Conclusions

1. A spatial refinement scheme for a structured grid was developed by adding transitional nodes and by changing the influence zone of the transition nodes in GIMP. The influence zone is square for uniform grid nodes and rectangular for transitional nodes. This influence zone affects the computation of the nodal shape function. The computation of the weighting function remains the same as for the uniform grid. The refinement scheme can be applied successively and the refined grid remains structural in each refinement level, i.e., every node can be de-

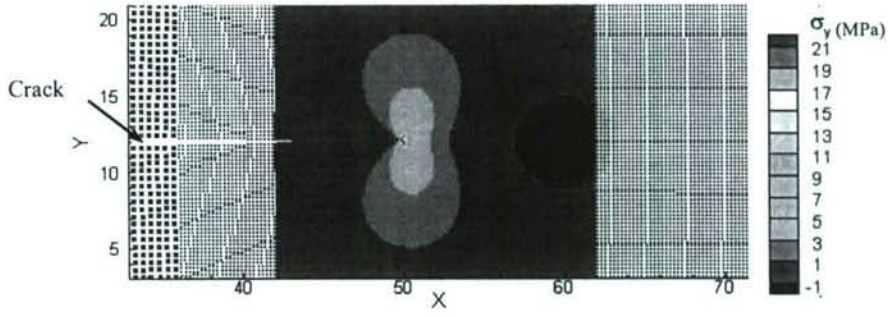


Figure 20 : Stress distribution in the beam using three levels of refinements at t = 2.5 ms

termined by the extent of the grid level and cell size. The refinement scheme was implemented and several problems such as tension, indentation, stress concentration, and stress distribution near a crack (mode I crack problem) were modeled to demonstrate its effectiveness and accuracy. A good agreement has been obtained between numerical and theoretical results, indicating the validity of the structured mesh refinement for GIMP scheme.

2. The GIMP algorithm has also been extended to include the displacement boundary condition, based on the approach used in the meshless local Petrov-Galerkin (MLPG) method, Atluri and Zhu (2000). A penalty parameter is used to impose the displacement boundary condition and a nodal force vector because the displacement boundary condition is introduced to the nodal momentum governing equation. A uniaxial tension problem with constant pulling velocity was simulated to verify the displacement boundary condition.

3. A method to track the material particle deformation was developed and verified in one example. When the particle deformation is not tracked, artificial separation was observed when the particle strain increases to a certain level. In tensile simulations, when the normal strain is ~25%, material particles tend to separate from the body. Our approach tracks the displacement of each corner of the material particles. Since neighboring particles share corners, no separation would occur during deformation using this approach.

Acknowledgement: The work was supported by a grant from the Air Force Office of Scientific Research (AFOSR) through a DEPSCoR grant (No. F49620-03-1-0281). The authors would like to thank Dr. Craig S. Hartley, and Dr. Jaimie Tiley, Program Managers for the Metallic Materials Program at AFOSR for the support of and interest in this work. Thanks are also due to Dr. Richard Hornung and Dr. Andy Wissink of the Lawrence Livermore National Laboratory for providing the code and assistance with the use of SAMRAI. One of the authors (RK) also acknowledges the A.H. Nelson, Jr., Endowed Chair in Engineering for additional support. Thanks are due to Dr. Samit Roy, formerly of OSU-Tulsa for participating in the project.

References

Atluri, S.N.; Zhu, T. (1998): A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics. *Computational Mechanics*, vol. 22, pp. 117-127.

Atluri, S.N.: Zhu, T. (2000): The meshless local Petrov-Galerkin (MLPG) approach for solving problems in elasto-statics. *Computational Mechanics*, vol. 25, pp. 169-179.

Atluri, S.N.; Shen, S. (2002): The meshless local Petrov-Galerkin (MLPG) method: A simple and less-costly alternative to the finite element and boundary element methods. *CMES: Computer Modeling in Engineering & Sciences*, vol. 3, n. 1, pp. 11-51.

Bardenhagen, S.G.; Kober, E.M. (2004): The generalized interpolation material point method.*CMES: Computer Modeling in Engineering & Sciences*, vol. 5, n. 6, pp. 477-496.

Guilkey, J.E. (2005): Use of MPM in fully Lagrangian

mode to eliminate cell crossing error, *First MPM Workshop*, March 13-14, 2005, Salt Lake City, Utah.

Guilkey, J.E.; Weiss, J.A. (2003): Implicit time integration for the material point method: Quantitative and algorithmic comparisons with the finite element method, *Int. J. Numerical Methods in Engineering*, vol. 57, pp. 1323-1338.

Guo, Y.; Nairn, J. (2003): Calculation of J-integral and stress intensity factors using the material point method. *CMES: Computer Modeling in Engineering & Sciences*, vol.6, n. 3, pp. 295-308.

Hornung, R.D.; Kohn, S.R. (2002): Managing application complexity in the SAMRAI object-oriented framework. *Concurrency and Computation: Practice and Experience*, vol. 14, pp. 347-368.

Hu, W.; Chen, Z. (2003): A multi-mesh MPM for simulating the meshing process of spur gears. *Computers & Structures*, vol. 81, pp. 1991-2002.

Kannien, M.F. (1973): An augmented double cantilever beam model for studying crack propagation and arrest. *Int. J. Fracture*, vol 9, pp. 646-664.

Ma, J.; Lu, H.; Wang, B.; Roy, S.; Hornung, R.; Wissink, A.; Komanduri, R. (2005): Multiscale simulations using generalized interpolation material point (GIMP) method and SAMRAI parallel processing. *CMES: Computer Modeling in Engineering & Sciences*, vol 8, n2, pp.135-152.

Pilkey, W.D. (1997): Peterson's Stress Concentration Factors, 2nd Edition, John Wiley & Sons, New York.

Rybicki, E.F.; Kanninen, M.F. (1977): A finite element calculation of stress intensity factors by a modified crack closure integral. *Engineering Fracture Mechanics*, vol. 9, pp. 931-938.

Sulsky, D.; Zhou, S.J.; Schreyer, H.L. (1995): Application of a particle-in-cell method to solid mechanics. *Computer Physics Communications*, vol. 87, pp. 236-252.

Tan, H.; Nairn, J.A. (2002): Hierarchical, adaptive, material point method for dynamic energy release rate calculations. *Computer Methods in Applied Mechanics and Engineering*, vol. 191, pp. 2095-2109.

Wang, B.; Karuppiah, V.; Lu, H.; Roy, S.; Komanduri, R. (2005): Two-dimensional Mixed Mode Crack Simulation Using the Material Point. *Mechanics of Advanced Materials and Structures*, vol. 12, pp. 471-484.

Multiscale Simulation Using Generalized Interpolation Material Point (GIMP) Method and Molecular Dynamics (MD)

J. Ma[1], H. Lu[1], B. Wang[1], R. Hornung[2], A. Wissink[2], and R. Komanduri[1*]

Abstract

A new method for multiscale simulation bridging two scales, namely the continuum scale using the generalized interpolation material point (GIMP) method and the atomistic scale using the molecular dynamics (MD), is presented and validated in 2D. The atomistic strain from the molecular dynamics simulation is determined through interpolation of displacement field into Eulerian background grid using the same generalized interpolation functions as in GIMP method. The atomistic strain as defined is consistent with that as determined from virial theorem for interior points, but provide more reasonably stable values at the boundary of MD region or in the handshaking region between MD-GIMP so that physical quantities, such as the displacement, internal forces, and energy density are compatible in the handshaking region. A continuum material point in the continuum is split into smaller material points using multi-level refinement until it has nearly reached the atom size to couple with atoms in the MD region. Consequently, coupling between GIMP and MD is achieved by using compatible deformation and force fields in the transition region between GIMP and MD. The coupling algorithm is implemented in the Structural Adaptive Mesh Refinement Application Infrastructure (SAMRAI) for parallel processing. Both mode I and mode II crack propagation problems are simulated using the coupling algorithm. The stress field near the crack tip was validated by comparing the results from coupled simulation with purely GIMP simulations of the same model. Coupled simulation results were also compared with pure MD simulation results. In both cases, a very good agreement was obtained.

Keywords: Incremental atomic strain, multiscale simulation, GIMP, MD, coupling, SAMRAI

---

[1] School of Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, OK 74078
*Correspondence author, Tel: 405-744-5900; Fax: 405-744-7873; e-mail: ranga@ceat.okstate.edu
[2] Center for Applied Scientific Computing, Lawrence Livermore Naticnal Laboratory, Livermore, CA 94551
This work was performed under the auspices of the U.S. Department of Energy by University of California, Lawrence Livermore National Laboratory under Contract W-7405-Eng-48. UCRL-JRNL-208241.

1. Introduction

Simulations at the continuum and atomistic levels are often used to determine the material deformation and failure at their respective length scales. The atomistic simulations are usually performed at very small length scales (?) and time scales (?) whereas continuum simulations are performed at larger scales (?). While the atomistic scale simulation can reveal the fundamental aspects of deformation and failure behavior, it cannot be scaled up for larger length scales due to limitation in computing power. However, for problems, such as nanoindentation and crack propagation, a combined atomistic and continuum simulations may provide as much information as a purely atomistic simulation would provide. This is especially so when MD simulations are used for regions encompassing high stress gradient zones, such as the crack propagation zone (Kohlhoff, Gumbsch and Fischmeister, 1991) and the workpiece just underneath the indenter as in nanoindentation (Shilkrot, Miller and Curtin, 2002) whereas continuum region is used for the rest of material.

Several techniques have been proposed for simulation bridging two or more scales (Kohlhoff, Gumbsch and Fischmeister, 1991; Shilkrot, Miller and Curtin, 2002; Curtin and Miller, 2003; Shiari, Miller and Curtin, 2005; Raffi-Tabar, Hua and Cross, 1998). A transition region, or handshake region, overlapped by the outer boundary of the atomistic region and the inner boundary of the continuum region, is usually used to transfer the physical quantities (Kohlhoff, Gumbsch and Fischmeister, 1991; Shilkrot, Miller and Curtin, 2002; Curtin and Miller, 2003). At the continuum level, in multiscale simulation, the finite element method (FEM) is often used. Waves with the wavelength larger than the element size can be transmitted from MD into continuum, and on the other hand, waves with wavelength smaller than the element size get reflected artificially (Raffi-Tabbar, Hua and Cross, 1998). Efforts were made on minimizing the wave reflection while enforcing the displacement and force continuity, as well as energy conservation in the handshake region. An the intermediate scale, namely, the mesoscale simulation, was also used in bridging the continuum and atomistic simulations (Raffi-Tabar, Hua and Cross, 1998). Recently, nonlinear deformation has been considered in coupling. The coupled atomistic/continuum discrete dislocation (CADD) method has demonstrated

superior capability in detecting dislocations from MD simulation and passing them to the continuum region, as well as handling explicit material defects and inclusions (Shiari, Miller and Curtin, 2005). For 2D simulations, the dislocations generated in the atomistic region can propagate into the continuum region by defining continuum elements with slip planes in front of the transition region to detect dislocations (Shilkrot, Miller and Curtin, 2002; Curtin and Miller, 2003).

Other simulation techniques, such as the meshless local Petrov-Galerkin (MLPG) method (Shen and Atluri, 2005), Green's function method (Cai, Koning, Bulatov and Yip, 2000), and coarse-grained molecular dynamics (Rudd, Broughton, 1998) are also used at the continuum level. These techniques show advantages in heat transfer, stress compatibility or minimizing wave reflection for the coupling between atomistic and continuum levels. Mesh distortion associated with FEM is an issue for large nonlinear deformations and dislocations. Recently, the material point method (MPM) (Sulsky, Zhou and Schreyer, 1995; Sulsky and Schreyer, 1996) was introduced for dynamic simulations and the general interpolation material point (GIMP) method (Bardenhagen and Kober, 2004) was presented with improved simulation stability. The MPM and GIMP use material points to represent a material continuum and utilizes both Lagrangian description (for material points carrying physical variables) and Eulerian description (for convection of physical variables and solution of field equations). Because the GIMP/MPM methods do not use a fixed body mesh so that mesh entanglement associated with highly nonlinear deformations can be prevented. GIMP and MPM have demonstrated the capabilities in the computation of stresses and strains in metal forming, dynamic fracture (Guo and Nairn, 2004), and impact problems (Sulsky and Shreyer, 1996). Recently, GIMP has been successfully implemented for 2D simulations covering length scales from nanometers to millimeters (Ma, Wang, Lu, Roy, Hornung, Wissink and Komanduri, 2004) using multi-level refinements with parallel computing in the framework of Structured Adaptive Mesh Refinement Infrastructure (SAMRAI) (Hornung and Kohn, 2002). The GIMP refinement technique proposed here has made it possible for coupling with MD so that the advantages of GIMP/MPM methods can be fully carried over to multiscale simulations. This paper will describe the GIMP/MD coupling techniques, and demonstrate it in the Mode I and Mode II crack propagation problems.

One aspect of the problems associated with coupling is the conversion of physical quantities computed from MD region to continuum region, and vice versa. MD simulation gives atom positions, velocities, etc., so that information can be passed over back and forth between two different regions. The stresses and strains defined in a material continuum can be computed from MD simulation using various approaches (Gao, Huang and Abraham, 2001; Buehler, Abraham and Gao, 2003; Zimmermann, 1999; Horstemeyer and Baskes, 2000). A typical, and still most effective approach thus far is based on the virial theorem (Marc and McMillan, 1985; Zimmermann, Webb, Hoyt, Jones, Klein, Bammann, 2004). However, it has some limitations such as not being able to compute stress/strain at the boundary of the MD region. Some new approaches have been developed to calculate the atomic stress more accurately. For example, Zhou and McDowell (2004) defined an equivalent continuum for the atomistic system with conserved momenta, work rates, and mass. The atomic stresses are then calculated from the internal forces and lattice constants. Hardly (see Zimmermann et al. (2004) for a summary of Hardly's approach) computed the atomic stresses from the contribution of nearby atoms with spatial averaging using a localization function.

The atomistic strain measures the deformation of the atomic lattice. It is also used in combined atomistic and continuum studies of material behavior. Buehler, Gao and Huang (2004) investigated the stress and strain fields near the crack tip in MD simulations and revealed that continuum mechanics can be used at nanoscale (no conditions stated). In another study, Bueher, Abraham and Gao, 2003 showed that in a harmonic solid, the local hyperelastic wave speed governs the crack speed of a mode-I crack without branching when the hyperelastic zone approaches the energy length scale near the crack tip.

This paper presents a method for coupling MD with GIMP simulations in 2D. A new formulation to compute the atomistic strain rate and strain increment is developed by computing the velocity gradient based on the interpolation of the velocity field on a background grid. Based on the atomic strain computation, an equivalent continuum system can be defined for the atomic region for the purpose of coupling MD with continuum computation using the GIMP method. The coupled GIMP and MD simulations use the same background mesh. The coupling scheme is based on the

atomistic strain, stress, as well as deformation while neglecting the heat transfer. Both mode I and mode II crack propagation problems are simulated to demonstrate the coupling approaches. The coupled simulation results are compared with purely GIMP simulation, and pure MD simulation for validation.

2. GIMP and refinement

In the generalized interpolation material point method (GIMP) (Bardenhagen and Kober, 2004), a continuum is discretized into a collection of material points. Each material point carries all the physical variables for a fully defined problem in solid mechanics, such as the position, mass, velocity, stress and strain, etc. The material point deforms to represent the motion of the continuum while the mass at each material point remains constant. For a dynamic problem, using the variational principle, the momentum conservation equation is given by

$$\int_{\Omega} \rho \mathbf{a} \cdot \delta \mathbf{v} d\mathbf{x} + \int_{\Omega} \boldsymbol{\sigma} : \nabla \delta \mathbf{v} d\mathbf{x} = \int_{\Omega} \mathbf{b} \cdot \delta \mathbf{v} d\mathbf{x} + \int_{\partial\Omega} \mathbf{c} \cdot \delta \mathbf{v} d\mathbf{x}, \tag{1}$$

where ρ is the mass density, $\mathbf{a}$ is the acceleration field, $\mathbf{b}$ is the body force density, $\boldsymbol{\sigma}$ is the Cauchy stress tensor, $\delta\mathbf{v}$ is an arbitrary admissible velocity field and Ω is the current configuration (Bardenhagena and Kober, 2004). To solve this equation, a background grid is introduced for interpolation between the material points and the nodes on the background grid using a weighting function that will be described later. Hence, Eq. (1) can be discretized and the equation of motion is eventually solved at each node,

$$\dot{\mathbf{p}}_i = \mathbf{f}_i^{\text{int}} + \mathbf{f}_i^{b} + \mathbf{f}_i^{ext}, \tag{2}$$

where the nodal momentum, internal force and body force are obtained by summing the contributions from each material points p to this node i as $\dot{\mathbf{p}}_i \doteq \sum_p \overline{S}_{ip} \dot{\mathbf{p}}_p$, $\mathbf{f}_i^{\text{int}} = -\sum_p \boldsymbol{\sigma}_p \cdot \overline{\nabla S}_{ip} V_p$ and $\mathbf{f}_i^{b} = \sum_p m_p \mathbf{b} \overline{S}_{ip}$, respectively. m_p and V_p are the particle mass and volume. The external force is given by $\mathbf{f}_i^{ext} = \int_{\partial\Omega_\epsilon} \mathbf{c} S_i(\mathbf{x}) dS$, where $\mathbf{c}$ is the surface traction. Finally, the position, strain and stress of the material points can be updated using the interpolation from surrounding nodes.

For GIMP simulations, in areas with high stress gradient, finer computational mesh and smaller time step should be used to improve accuracy. Both spatial and temporal refinements have been introduced in a multilevel refinement algorithm (Ma, Wang, Lu, Roy, Hornung, Wissink, and Komanduri, 2005) in which the computational domain consists of nested grid levels with successive refinement. Each finer level covers part of the next coarse level and each level is computed separately with its own time increment. Smaller time increments are used for finer levels and the communication between two grid levels are performed when these two levels are synchronized. Fig. 1 shows two neighboring coarse and fine grid levels in 2D GIMP computations with a refinement ratio of two. The thick line represents the physical boundary of the fine level with four layers of ghost cells. Initially, four material points are assigned to each cell at the fine level. At the coarse level, the portion overlapped by the fine level is assigned with 16 material points per cell. Hence, these material points have the same size and initial positions as those at the fine level. The rest of the coarse level is assigned with four material points per cell. Two data exchange processes, i.e., refinement and coarsening are used in the communication. Refinement process passes information from the coarse level to the immediate fine level, while coarsening process passes information from fine level to the next coarse level. In the refinement process, physical variables at the fine material points inside the thick lines are copied directly to replace the material points at the coarse level. In the coarsening process, the physical variables at coarse material points are copied to the ghost cells of the immediate fine level.

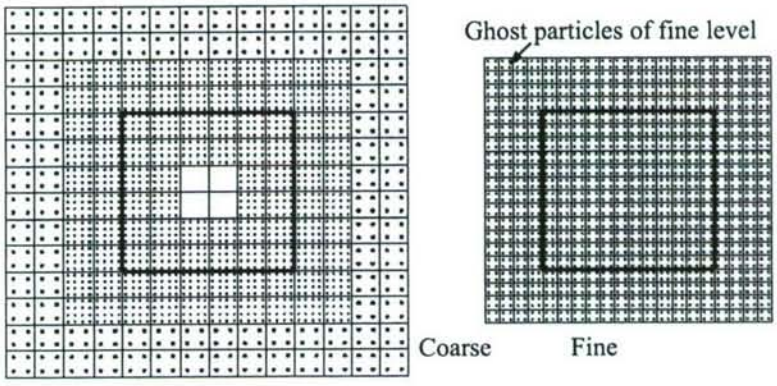


Fig. 1: Two neighboring coarse and fine grid levels in 2D GIMP computations

Using SAMRAI, each grid level can be divided into multiple patches for parallel processing. Each patch is rectangular and is assigned to a processor. The patches are overlapped by ghost cells for communication. The parallel processing scheme will be discussed in more detail in later sections.

3. MD Simulation and Atomistic strain

Molecular dynamics (MD) simulations solves the motion of the atoms by integrating the equations of motion governed by Newton's second law $\mathbf{f} = m\mathbf{a}$. In general, short range interaction, governed by an atomic potential function, among atoms within the cutoff radius is considered. With the rapid increase of computing power, MD simulation is becoming a powerful tool in simulation of material behavior. The MD code used in this paper is the LAMMPS code (Plimpton, 1995) developed at the Sandia National Laboratories.

To couple MD with continuum simulation using the GIMP method, we need to pass MD results to continuum simulation. For this purpose, variables used in MD must be consistent with those in the GIMP simulation. The most critical issue is passing the atom velocities, positions to the continuum region. This involves the computation of continuum strain fiels based on the discrete atom deformation information.

There are numerous publications in recent years on the computation of strain from MD results (?). One successful approach is the formulation to calculate the atomic virial strain developed by Zimmermann, Webb, Hoyt, Jones, Klein, and Bammann (2004). In this approach, the left Cauchy-Green deformation tensor for atom a at coordinates (x_1^a, x_2^a, x_3^a) is written as

$$B_{ij}^a = \frac{1}{\lambda}\sum_{b\neq a}^{N}\frac{\Delta x_i^{ab}\Delta x_j^{ab}}{R_{ab}^2}, \tag{3}$$

where $\Delta x_i^{ab} = x_i^a - x_i^b$, $\Delta x_j^{ab} = x_j^a - x_j^b$, R_{ab} is the undeformed distance between atoms a and b, and N is usually limited to the nearest neighbors. λ is a factor dependent on the lattice. $\lambda = 3$ for the 2D triangular lattice if the six nearest neighboring atoms are only considered (Zimmermann, 1999). Therefore, the Eulerian strain tensor for atom a is found as

$$\varepsilon_{ij}^{a} = \frac{1}{2}(\delta_{ij} - [B_{ij}^{a}]^{-1}), \tag{4}$$

where δ_{ij} is the Kronecker delta. The virial strain calculated above is meaningful instantaneously in time and space. However, at locations where the nearest neighbor list does not exist or cannot be determined, such as at the external boundary, crack surfaces and interfaces, Eq. (3) is invalid. To circumvent this drawback, we propose an alternative approach as described in the next section to calculate the atomistic strain based on the strain rate.

3.1 Incremental atomic strain

In continuum mechanics, the strain is defined as the gradient of the deformation field. To construct the deformation field from the MD simulation, an Eulerian background grid, same as that used in GIMP, is utilized. All the quantities defined at atoms, such as the mass, velocity, and forces can be projected to the background nodes through interpolation. While the nodal positions are fixed in space, the nodal quantities can vary with time. The velocity of atom a inside the grid is written as

$$\mathbf{v}(\mathbf{x}_a, t) = \sum_{i=1}^{n} \overline{S}_{ia}(\mathbf{x}_a)\mathbf{V}_i\ (t), \tag{5}$$

where n is the number of nodes and $\overline{S}_{ia}(\mathbf{x})$ is the interpolation function between node i and atom a. By definition, the strain of atom a is then

$$\boldsymbol{\varepsilon}(\mathbf{x}_a, t) = \frac{1}{2}\{\nabla\mathbf{u}(\mathbf{x}_a, t) + [\nabla\mathbf{u}(\mathbf{x}_a, t)]^T\}, \tag{6}$$

where $\mathbf{u}(\mathbf{x}_a, t)$ is the deformation field. The strain rate is given by

$$\dot{\boldsymbol{\varepsilon}}(\mathbf{x}_a, t) = \frac{1}{2}\{\nabla\mathbf{v}(\mathbf{x}_a, t) + [\nabla\mathbf{v}(\mathbf{x}_a, t)]^T\}. \tag{7}$$

The velocity gradient at atom a can be expressed as $\nabla\mathbf{v}(\mathbf{x}_a, t) = \sum_{a=1}^{n} \nabla\overline{S}_{ia}(\mathbf{x}_a)\mathbf{V}_i\ (t)$. Using the Euler integration scheme, we calculate the strain at atom a for the next time step through

$$\boldsymbol{\varepsilon}_a^{t+\Delta t} = \boldsymbol{\varepsilon}_a^{t} + \dot{\boldsymbol{\varepsilon}}_a^{t+\Delta t}\Delta t, \tag{8}$$

where $\dot{\boldsymbol{\varepsilon}}_a^{t+\Delta t} = \frac{1}{2}\sum_{i=1}^{n}(\nabla \overline{S}_{ia}\mathbf{V}_i^{t+\Delta t} + \mathbf{V}_i^{t+\Delta t}\nabla \overline{S}_{ia})$.

The nodal mass, force, and momentum are computed from neighboring atoms as $M_i^t = \sum_{a=1}^{n}\overline{S}_{ia}m_a^t$, $\mathbf{F}_i^t = \sum_{a=1}^{n}\overline{S}_{ia}\mathbf{f}_a^t$, $\mathbf{P}_i^t = \sum_{a=1}^{n}\overline{S}_{ia}(m_a^{t+\Delta t}\mathbf{v}_a^{t+\Delta t})$, respectively. The updated nodal momentum and velocity are $\mathbf{P}_i^{t+\Delta t} = \mathbf{P}_i^t + \mathbf{F}_i^t\Delta t$ and $\mathbf{V}_i^{t+\Delta t} = \mathbf{P}_i^{t+\Delta t}/M_i^t$.

3.2 Interpolation function

To ensure conservation of mass, momentum, and energy between the nodes and the atoms, the interpolation function should satisfy partition of unity. The interpolation function $\overline{S}_{ia}(\mathbf{x})$ can be chosen to be the isoparametric shape functions used in finite element analysis. However, the generalized interpolation functions with C^1 continuity in GIMP has shown better simulation stability when an Eulerian grid is used (Bardenhagen and Kober, 2004; Ma, et al. 2005) and it is used to compute the atomic strain in this investigation.

The generalized interpolation function introduced by Bardenhagen and Kober (2004) consists of two functions, i.e., the nodal shape function and the particle characteristic function. Both the nodal shape function and the material point characteristic function can be extended to 3D. In three dimensional situations, the nodal shape function, for node i, is given by

$$S_i(\mathbf{x}) = S_i^x(x)S_i^y(y)S_i^z(z), \tag{9}$$

and the particle characteristic function, for a brick particle p, is given by

$$\chi_p(\mathbf{x}) = \chi_p^x(x)\chi_p^y(y)\chi_p^z(z), \tag{10}$$

where

$$S_i^x(x) = \begin{cases} 0 & x - x_i \le -L_x \\ 1 + (x - x_i)/L_x & -L_x \le x - x_i \le 0 \\ 1 - (x - x_i)/L_x & 0 \le x - x_i \le L_x \\ 0 & L_x \le x - x_i \end{cases}, \tag{11}$$

and

$$\chi_p^x(x) = H[x - (x_p - l_x)] - H[x - (x_p + l_x)], \tag{12}$$

where *H(x)* denotes the step function. Fig. 2 shows a rectangular grid cell with a particle in it. The particle position is taken at the center of the particle. It may be noted that the grid associated with the generalized interpolation function is a structured grid, which is convenient to construct and process.

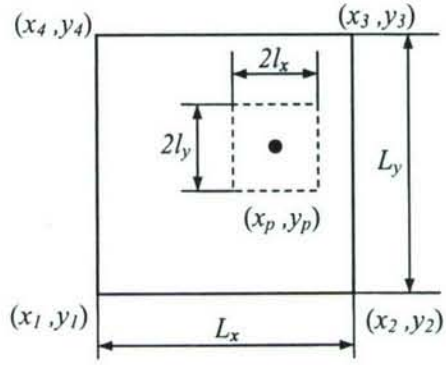


Fig. 2: 2D representation a particle and a grid cell

Both the nodal shape function and the particle characteristic function are a partition of unity, i.e., $\sum_p \chi_p(\mathbf{x}) = 1$ and $\sum_i S_i(\mathbf{x}) = 1$. The generalized interpolation function is a volume averaged weighting function between node i and material points p given by

$$\overline{S}_{ip} = \frac{1}{V_p} \int_{\Omega \cap \Omega_p} \chi_p(\mathbf{x}) S_i(\mathbf{x}) d\mathbf{x}, \tag{13}$$

where Ω is the entire computational configuration, Ω_p is the spaces occupied by particle p, V_p is the current particle volume.

The atoms are generally regarded as spherical in shape. The characteristic function of a sphere can also be found. However, to simplify the computation of the interpolation function and its gradient, we consider that the atoms are cubic in shape with the same volume of a spherical atom. We also assume that the shape and orientation of the atoms do not change in the simulation. Consequently, Eq. (13) is simplified as

$$\overline{S}_{ip} = \frac{1}{8 l_x l_y l_z} \int_{x_p - l_x}^{x_p + l_x} S_i^x(x) dx \int_{y_p - l_y}^{y_p + l_y} S_i^y(y) dy \int_{z_p - l_z}^{z_p + l_z} S_i^z(z) dz. \tag{14}$$

The final expression of the interpolation function and its gradient can be found in Bardenhagena and Kober (2004). The interpolation function is plotted in Fig. 3. The

vertical axis is the interpolation function and horizontal plane is the particle position. The node is at (0, 0). The cell size is 1×1 and the particle size is 0.5×0.5. The interpolation function is continuous and smooth. It may be noted that the volume bounded by the interpolation function and the horizontal plane is one.

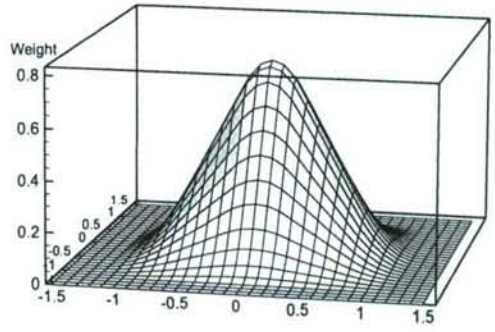


Fig. 3: The generalized interpolation function in 2D

3.3 Numerical validation

Several MD simulations are performed to calculate the atomistic strain in 2D. In these simulations, the atomic potential chosen is the Lennard-Jones potential given by

$$U(r) = 4\varepsilon_0\left(\frac{\sigma^{12}}{r^{12}} - \frac{\sigma^6}{r^6}\right) \qquad (15)$$

with $\varepsilon_0 = 1.0$ and $\sigma = 1$. Based on the unit convention in LAMMPS, the units for length and energy in this paper are σ and ε_0, respectively. This potential gives the elastic wave speeds as follows: longitudinal wave speed $c_l = 8.99$, shear wave speed $c_s = 5.19$, and the Rayleigh wave speed $c_R = 4.80$. 2D triangular lattice is used in the simulations. The model has 40 lattices in the X direction and 20 lattices in the Y direction. The size of the model is 44×39. The atom is assumed to be cubic with a volume of unity.

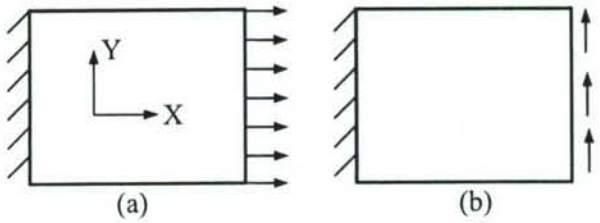
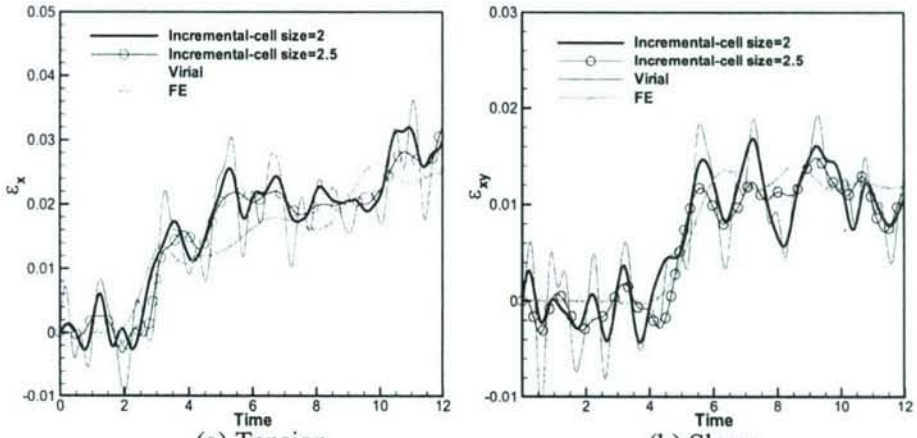


Fig. 4: Two examples to calculate the atomistic strains (a) Tension and (b) Shear

The first simulation is simple tension as shown in Fig. 4 (a). The model is constrained in the X direction on the left while a constant velocity is applied on two layers lattices on the right. The strain histories of the atom initially at (24.5, 11.2) calculated using the virial formula and the new incremental approach are plotted in Fig. 5 (a). It can be seen that in general the strains calculated from different methods agree reasonably well. The virial strain shows more oscillations than the incremental strain. The interpolation between the nodes and the atoms is also a spatial averaging process. Two cell sizes, 2 and 2.5 are used in the background grid to investigate the effect of cell size. It can be seen that oscillation of the strain are smaller when the cell size is bigger and vice versa because the nodal quantities are interpolated from a bigger number of atoms. A large content of noise, as a result of random thermal vibration of the atoms, has been suppressed due to averaging over a number of atoms. For comparison, the strain from continuum simulation using the finite element method is also plotted, assuming the model material to be homogeneous, isotropic, and linearly elastic. It is seen that all simulations show that the strain first increases at time between t = 2.5 (any unit??) and 3. The calculated time is t = 2.8 based on the longitudinal stress wave speed.

The second simulation is a shearing problem as shown in Fig. 4 (b). Constant upward velocity is applied on the two lattices on the right. The resultant shear strain at (24.5, 11.2) is plotted in Fig. 5 (b) from these simulations. The virial strain again shows largest variation, while the strain defined herein shows much less vibration.



(a) Tension (b) Shear

Fig. 5: Strain histories from different simulations

It may be noted that computation of the incremental atomic strain does not depend on the atomic potential function and the lattice structure. The deformation field is recovered to the grid nodes using interpolation. Interpolation also allows averaging on the displacement field so that the computed atomic strain shows less oscillation. It is known that the virial formula is not applicable to atoms on the surface, Zimmermann, 1999. The incremental formulation developed in this investigation, however, does not have this restriction. The strains of the atom at the top surface (in the middle of the model) from three computations are plotted in Fig. 6. It can be seen that prediction of incremental formulation is reasonably in good agreement with the finite element analysis.

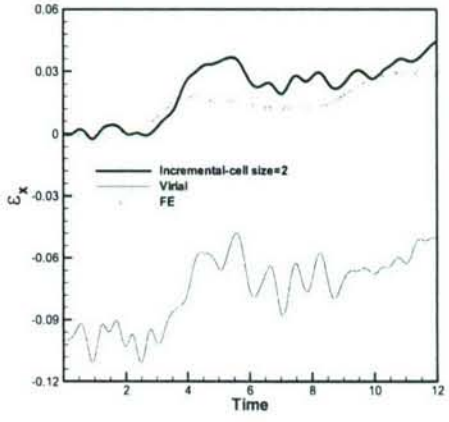


Fig. 6: Comparison of the normal strain of a surface atom in tension

The ability to compute the strains for the boundary and surface atoms is essential for coupling simulations. For example, the strain at the crack surface is required in coupling. It may be noted that the virial formulation computes a transient strain so that it does not depend on the strain history whereas the incremental strain proposed herein is computed from the strain rate so that it needs to be computed at every time increment during the entire strain history of interest.

4. Coupling of GIMP and MD simulations

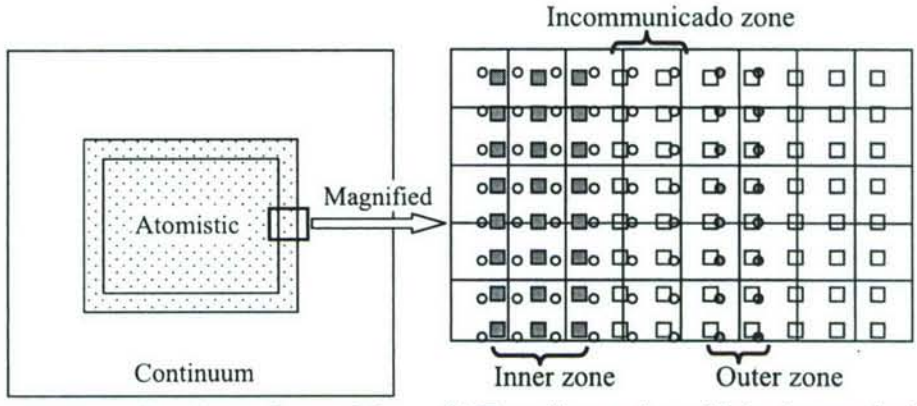
4.1 Coupling scheme

To enable coupling between GIMP and MD, the computational results must be able to exchange between the atomistic simulation from one side and the continuum simulation from the other. A successful coupling is indicated by the seamless transfer of the deformation, internal forces, heat, etc. at the interface of two different simulation regions.

In the coupling scheme presented in this paper, heat transfer between the continuum and atomistic regions are neglected. Moreover, the transition region is assumed to remain in the linear elastic regime at this stage. Fig. 7 (a) shows the overall coupling model and Fig. 7 (b) shows details of the transition zone. The MD region covers a small portion of the overall model and the continuum region covers the rest. The transition region, where the communication between GIMP and MD simulations take place, is divided into three zones, the inner zone, the incommunicado zone and the outer zone, as shown in Fig. 7 (b).

In the inner zone, the information is passed from MD to GIMP. The atomic strain is computed using the incremental scheme presented in Sections 3.1 and 3.3. The velocity and strain rate at the material points overlapped within the atomistic region are computed using Eqs. (5) and (7) by replacing the atom positions with the positions of the material points. Hence, the stresses of these points are computed based on continuum theories. Consequently, the material points inside the atomistic region can be updated by the atomistic information. These material points can participate in the overall continuum computation to provide information for the rest of the continuum region. The innermost rectangle in Fig. 7 (a) represents the region in which all material points are updated based

on the atomistic information. These material points are shown in Fig. 7 (b) as filled squares. The unfilled squares are the material points that are updated in regular continuum computations.



(a) Overall regions of a model (b) Transition region with background grid

Fig. 7: Illustration of coupled GIMP and MD simulations. The circles represent atoms and squares (smaller than physical size) material points. The material points connect each other without a gap to represent continuum.

The boundary atoms in the outer zone, shown as filled circles in Fig. 7, update their velocities based on the interpolation with the background grid node using Eq. (5). The nodal velocities of the background grid are computed from the velocities of the material points weighted by the masses of the material points, as

$$\mathbf{V}_i(t) = \frac{\sum_p \bar{S}_{ip}(\mathbf{x}_p)\mathbf{m}_p\mathbf{v}_p(\mathbf{x}_p,t)}{\sum_p \bar{S}_{ip}(\mathbf{x}_p)\mathbf{m}_p}. \tag{16}$$

Once the velocities of the boundary atoms are known, their displacements can be found from the time integration of the velocities. At this point, the interactions with interior atoms are fully defined by the inter-atomic potentials as in regular molecular dynamics simulations.

In the proposed coupling scheme, two different zones are used for communication in different directions, i.e. from GIMP to MD or from MD to GIMP. The incommunicado zone is the gap that separates the inner and outer zones. This zone serves as a buffer for the communication between the material points and the atoms. The width of this zone is 3

to 4 layers of material points. In this coupling scheme, iteration to obtain the converged solution in the transition zone is not necessary, as the source of information is guaranteed to be correct and the communication in each zone is unidirectional. This approach not only simplifies the coupling algorithm but also improves the stability and reduces the computational cost.

The time step for MD simulation is usually smaller than the time step for GIMP, and can be used as the time step in the coupled simulation to reach stability in coupled simulation. However, with the use of small MD time step as the overall time step in coupled simulation, the amount of computation is enormous. In this coupling algorithm, a temporal coupling scheme is developed. A temporal factor N is defined as the ratio of the GIMP time step to the MD time step, that is,

$$N = \frac{dt_{GIMP}}{dt_{MD}}, \tag{17}$$

where $dt_{GIMP} = cL/\sqrt{E/\rho}$ is the GIMP time step. L is the cell size; c is a constant factor (0.1 in this paper). In the coupling, MD simulation advances N time steps for each GIMP time step. For this reason, it is convenient to use an integer for the temporal factor. In computation, the GIMP time step is computed first from the cell size and stress wave speed. Next, the temporal factor N is rounded to an integer based on Eq. (17). Finally, the GIMP time step is determined from $dt_{GIMP} = N \cdot dt_{MD}$. The reduction in computation time as a result of the temporal coupling scheme is significant, as indicated by numerical examples that will follow.

We next summarize the coupling scheme with the aid of flowchart Fig. 8. The material points inside the atomistic domain are updated based on the nodal information interpolated from the atoms. These material points join the GIMP calculation to provide boundary conditions, but are not updated again. The velocity and position of the boundary atoms are updated based on the nodal information interpolated from the material points. These boundary atoms join the MD calculation to provide boundary conditions as well, but the velocities are not updated either. The concept of temporal coupling is introduced by advancing. MD simulation N steps for every GIMP step, in order to save computational time. It may be noted that the material points further inside the MD region can be dumped to reduce the amount of computation since only a few

layers of material points are needed to provide boundary conditions to the exterior material points.

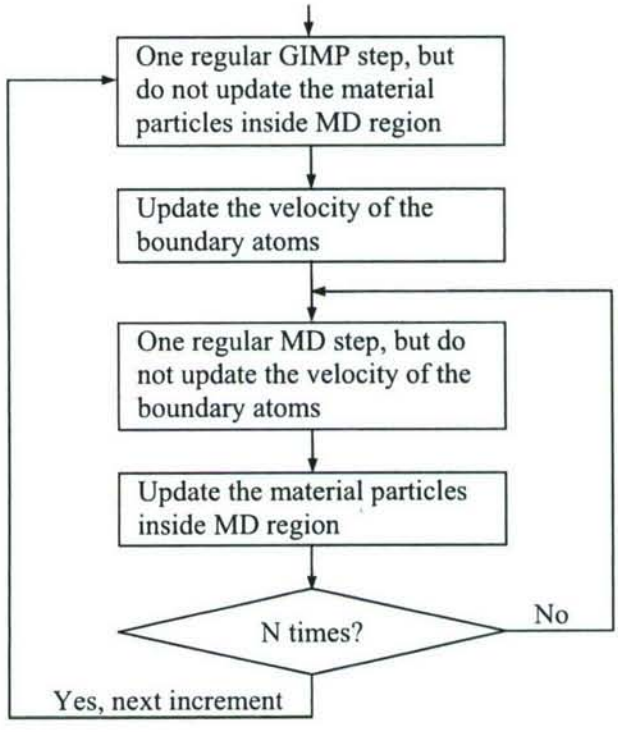


Fig. 8: Flow chart of the coupling algorithm for one increment

4.2 Parallel processing of the coupled model

The parallel processing for the GIMP with multilevel refinement using SAMRAI has been discussed in detail in (Ma, et al. 2005). The coupling between GIMP and MD is performed at the finest level only. So the finest level is always larger than the MD domain. The finest level is divided into rectangular regions called patches according to SAMRAI. Each patch is assigned to one processor and the communication between patches is performed after each patch is computed. To void data transferring in the coupling process, the MD region is decomposed into rectangular region with the same size as the patches. Each processor thus handles the same spatial region for both GIMP and MD. Fig. 9 shows three levels of refinement for GIMP with MD coupled in the finest level. The dashed lines divide the finest level as well as the MD domain into six rectangular regions. Each region is processed by a processor and the communication between the material points and the atoms for coupling is performed within the processor.

No data transferring between the processors is necessary. The disadvantage of this approach is that the load balancing of the processors is more difficult to adjust due to the change of the number of atoms and material points in computation.

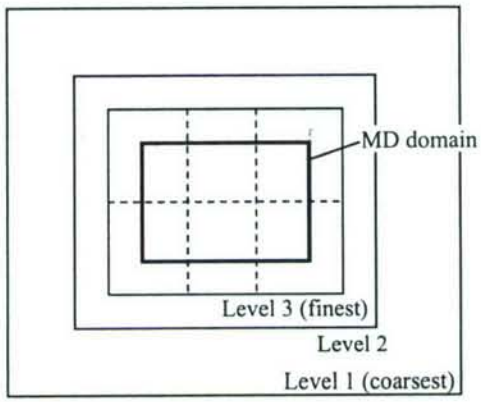


Fig. 9: Illustration of three GIMP grid refinements and the domain decomposition for coupling

5. Numerical results

5.1 Multiscale simulation of mode I crack

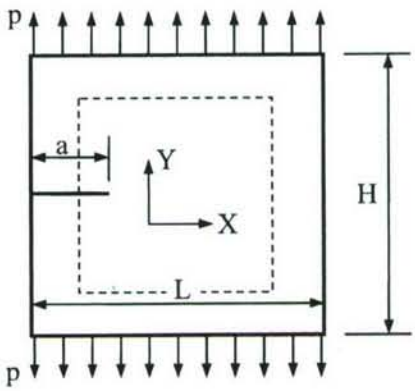


Fig. 10: Coupled GIMP/MD simulation of a 2D mode-I edge crack. The dashed lines are the boundaries of the atomistic domain.

The coupling scheme is implemented into a multiscale simulation of a mode I crack problem as shown in Fig. 10. The overall model size is 1968×1808 and the size of the MD domain is 890×600. Three levels of refinement in GIMP are applied. The finest level has a grid size of 4 and the two coarser grid sizes are 8 and 16, respectively. The finest level is coupled with MD and is divided into six patches as shown in Fig. 9. It may be

noted that the sizes of the material points and the grid cells determine the wavelength that can be transmitted from MD simulation to the continuum. In this simulation, the high frequency waves are not desirable in the continuum region. Hence, to save computational time, the minimum grid size is chosen as 4. The total number of atoms is 497,111 and the initial crack length is a = 498 at the height Y = 301. The applied pressure is 0.3 and the time step of the MD simulation is 0.002. The temporal factor N is chosen as 40, and therefore, the GIMP time step for the coupling level is 0.08. For comparison purposes, the same problem is simulated using GIMP alone without coupling. Fig. 11 shows the stress distribution in the model at simulation time t = 64 when the stress wave just traveled to the transition region. Fig. 11 (a1) corresponds to pure GIMP simulation, and Fig. 11 (b1) is with coupling. Due to thermal vibrations of the atoms, the stress in the MD region varies randomly between +/-0.15~0.2 at stress free state. The material points further inside the MD region can be dumped to reduce the amount of computation. Fig. 11 (c1) shows the result of this case in which only 3 layers of material points immediately inside transition region are kept. It is seen that the stress distribution is not affected by dumping the material points inside the MD region. It was observed that the computational time was reduced by about 50% in this case. Close to the crack tip, the stress distribution is affected by dislocations and crack propagation. In later discussions, only the results with material points dumped are presented.

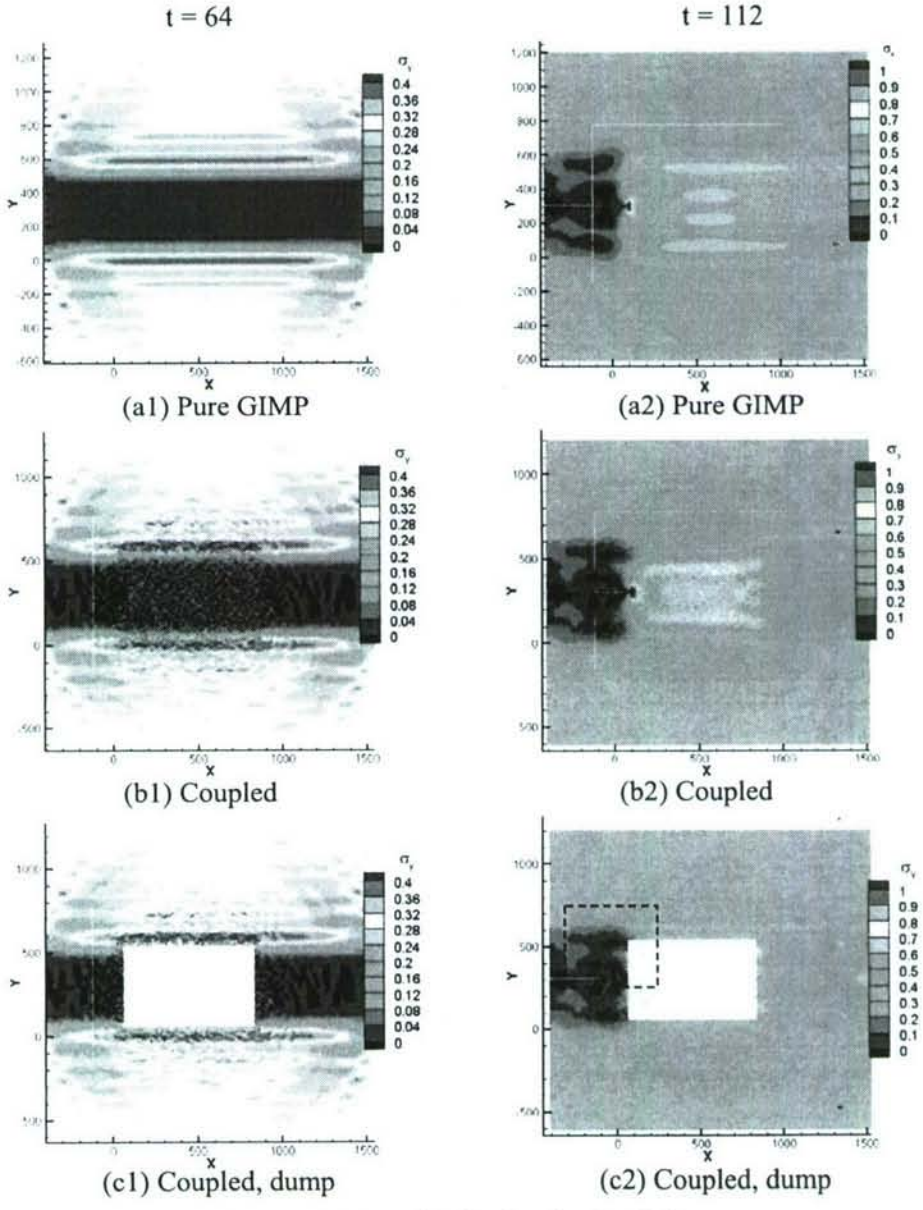


Fig. 11: Stress distribution for P = 0 .3

Fig. 11 also shows the same model at time t = 112. At this moment the stress concentration at the crack tip has developed. The crack just started to open. The stress fields from pure GIMP simulation and coupled simulation are shown in Fig. 11 (a2) and (b2), respectively. The difference of the maximum normal stress in the Y direction is 8%. The coupling region is zoomed in and shown in Fig. 12, corresponding to the region

indicated by the dashed rectangle in Fig. 11 (c2). The material points taking information from the atoms are overlapped inside the MD region. These material points cannot be seen in Fig. 12 due to the high density of the atoms.

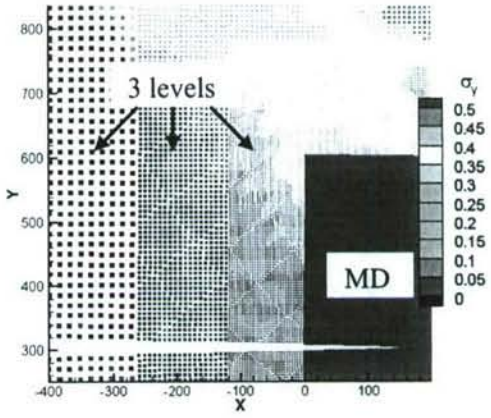


Fig. 12: Simulation model with an edge crack with P=0.3 at t=208

It may be noted that the reduction of computational time due to the temporal coupling with N = 40 is about 75%, comparing with the same simulation with N = 1 (no temporal coupling). The GIMP step involves regular GIMP calculation and communication between patches and levels (refinement and coarsening). The GIMP computation time is significantly larger than that of an MD step in this example. Temporal coupling has reduced the computational time significantly while maintaining the accuracy of coupling, as manifested in Fig. 11.

To study the energy release rate of mode I crack using the coupling algorithm, a tensile pressure of 0.15 is applied from t = 0 to t = 210. The crack started to propagate at t = ~225 and stopped at t = ~334. The external input energy, total potential energy and total energy from simulation are plotted in Fig. 13. Due to the numerical damping, the total energy decreases as time increases. The energy loss at t = 360 is 20%. Numerical simulation was conducted for this same model without damping and the difference between the input energy and the system energy gain is less than 8%. The potential energy is the sum of the potential energy of the atoms and the strain energy of the material points. It increases at first when the model is stretched. After the crack starts to propagate the potential energy drops. The energy release rate for this simulation is plotted

in Fig. 14, as well as the crack resistance. The energy release rate fluctuates as the crack propagates and its magnitude is larger than the crack resistance. After t=330 the energy release rate drops below the crack resistance, leading the crack to arrest.

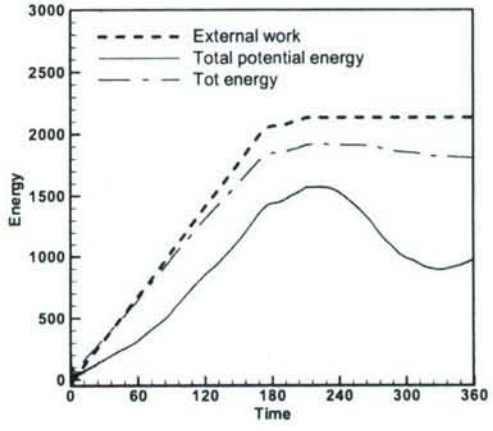


Fig. 13: Energies in the model with P = 0.15

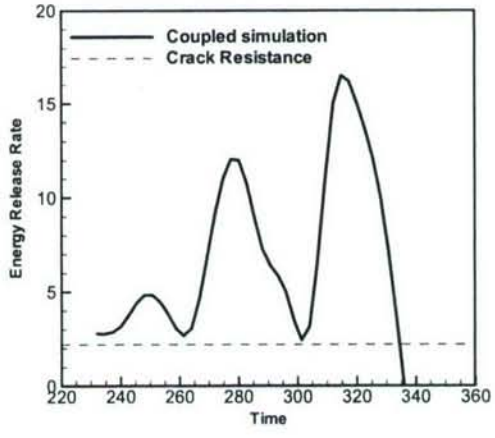


Fig. 14: Energy release rate P = 0.15

5.2 Multiscale simulation of mode II crack

The same model in Fig. 10 was simulated under mode II condition by applying velocities on the left side of the model as shown in Fig. 15. The applied shearing velocities in the X direction, are -0.1 and 0.1 for the top half and bottom half, respectively. Moreover, to avoid the contact of the initial crack surfaces, a velocity in the Y direction is applied on the left boundary, 0.03 for the top half and -0.03 for the bottom half. The

initial crack is horizontal with the tip located at (67, 301). This problem was simulated with both the coupled model and full MD model at zero temperature.

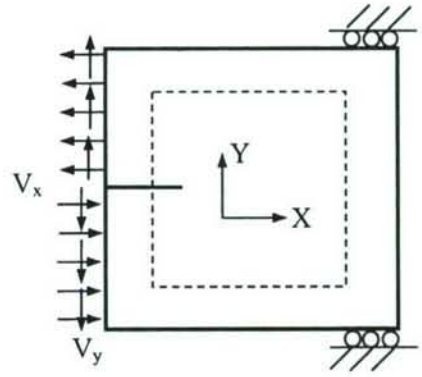
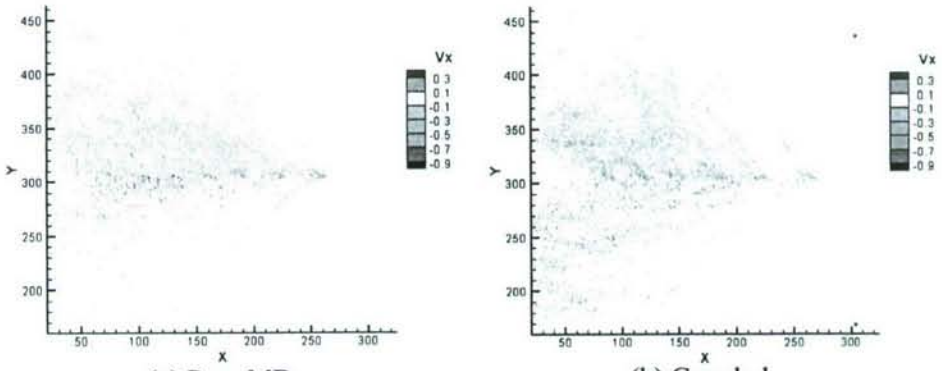


Fig. 15: Boundary conditions of the mode II crack problem

The velocities of the atoms in the X direction from pure MD simulation are shown in Fig. 16 (a). In this simulation, a crack was introduced to the left of the crack tip by eliminating the interaction of the atoms on both sides of the crack. As the velocities were applied on the boundary, the predominant deformation in the model is shearing. The path of the shearing front was observed to be straight, as an extension to the initial crack. The relative shearing motion between the two layers of atoms, which generate dislocations, occurred in the stick/slip mode, as indicated by Fig. 16 (a). The blue spots indicate atoms moving to the left at a velocity of ~0.7 and relative velocity between the atoms on the top and bottom is ~0.8. Between the blue spots, the relative velocity is very small, in the range of 0 to 0.2. Most of the shear surfaces are sticking together and slip occurs at isolated spots.



(a) Pure MD (b) Coupled

Fig. 16: Comparison of results from pure MD and coupled simulations at t = 11.2

The results from coupled simulation are plotted in Fig. 16 (b) for comparison. It can be seen that the velocity contours agree well with pure MD simulations. The stick/slip pattern also matches. This further validates the coupling algorithm. It may be noted that for this size of this model, there are 3.3 million atoms in pure MD simulation, 0.5 million atoms and three levels of material points in the coupled simulation. The computation time and memory for coupled simulation are 8% and 30% less than the pure MD simulation, respectively. As the model grows bigger, the reduction in both computation time and memory size will be more significant.

In order to track the dislocations in the atoms, the relative displacement of each atom with respect to each neighboring atom was computed and the maximum relative displacement was recorded for the atom. Fig. 17 plots the maximum relative displacement of the atoms at different simulation time when the cut-off value of the maximum relative displacement is set to 0.15, i.e., atoms with maximum relative displacement of less than or equal to 0.15 are not shown. With this cut off value, all the visible atoms are those immediately on two sides of the crack extension, as well as, those with stick/slip pattern. The left end of each plot is the crack tip and right end is the slip front. The average propagation speed of the slip frontier is ~3.5 during t = 64 and t = 160.

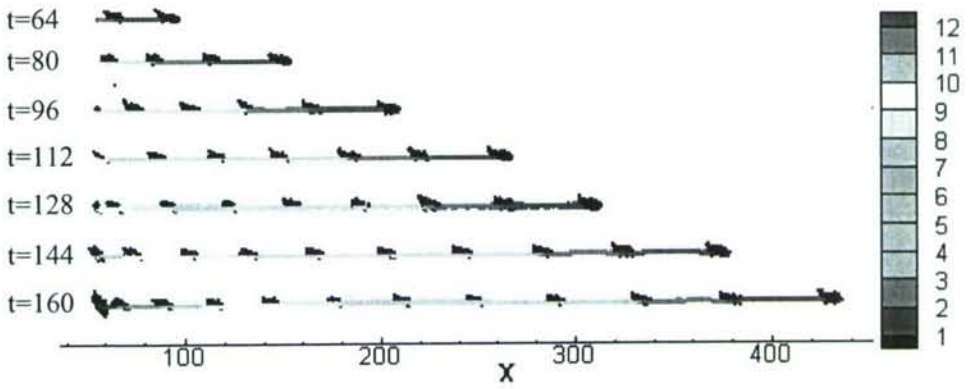


Fig. 17: Dislocation path at different simulation time

6. Conclusions

- A coupled atomistic/continuum simulation method is presented by coupling MD simulations with GIMP simulations. To enable coupling, a method for the computation of

atomistic strain, based on the computation of strain rate, is developed. The atomic strain rate is computed by first interpolating the velocities of the atoms to a background Eulerian grid, then computing the gradient of grid velocities. The computed atomic strain shows less vibration than that computed by the virial theorem due to noise reduction as a result of the interpolation process. The generalized interpolation function is chosen as the interpolation function for GIMP and the structured grid for the background grid. The atomic stress is then computed based on strain using the isotropic and homogeneous constitutive law.

The coupling algorithm uses a common background grid for MD and GIMP. The velocities of the boundary atoms are computed from the grid velocities, which are interpolated from the material points. The material points inside the MD region are updated based on the atomic information and these points join the rest of the material points in the GIMP computation. This approach ensures the compatibility of deformation and internal forces at the MD/GIMP interface region.

- A multi-level refinement scheme for GIMP is used to refine the material points close to the atomistic size. The coupling algorithm is implemented in the Structural Adaptive Mesh Refinement Application Infrastructure (SAMRAI) for parallel processing. The finest GIMP level is coupled with the MD simulation. The MD region is decomposed into domains with the same geometry as the GIMP patches. Each patch is assigned to a processor and the coupling between the atoms and material points is performed inside each processor without extra inter-processor communication.
- The coupling algorithm was validated by comparing thecoupling results with pure GIMP, and with pure MD simulation. A mode I crack propagation problem was simulated using the proposed the coupling technique. The stress filed of the coupled model is validated by comparing with pure GIMP simulations in the elastic stage. The energy release rate is computed and it is found that the crack arrests after the energy release rate is less than the crack resistance. In model II loading, stick/slip was observed in the crack front and the results compared well with pure MD simulation. While the current implementation is in 2D, the coupling algorithm can be applied to 3D.

The coupling algorithm presented here can be extended to include discrete dynamics outlined in the CADD method (Curtin and Miller, 2003; Shiari, Miller and

Curtin, 2005). A dislocation detection band can be placed inside the inner zone in Fig. 7 which in turn can be fed into the continuum region, and vice versa.

Acknowledgements

The work was supported by the Air Force Office of Scientific Research (AFOSR) through a DEPSCoR grant (No. F49620-03-1-0281). The authors would like to thank Dr. Craig S. Hartley, and Dr. Jaimie Tiley, Program Managers for the Metallic Materials Program at AFOSR for the support of and interest in this work. One of the authors (RK) also acknowledges the A.H. Nelson, Jr., Endowed Chair in Engineering for additional support. Thanks are also due to Dr. Samit Roy, formerly of OSU-Tulsa, for participating in the project.

References

Bardenhagen, S.G. and Kober, E.M. (2004): The generalized interpolation material point method. *Comp. Model. Eng. Sci.*, v5 n6, pp. 477-496.

Buehler, M.J., Abraham, F.F. and Gao (2003): H., Hyperelasticity governs dynamic fracture at a critical length scale. *Nature*, v426, pp. 141-146.

Buehler, M.J., Gao H. and Huang, Y. (2004): Atomistic and continuum studies of stress and strain fields near a rapidly propagating crack in a harmonic lattice. *Theor. Appl. Frac. Mech.*, v41, pp.21-42.

Cai, W., Koning, M. Bulatov, V. and Yip, S. (2000): Minimizing boundary reflections in coupled-domain simulations. *Phys. Rev. Lett.*, v85 n15, pp. 3213-3216.

Curtin, W.A. and Miller, R.E. (2003): Atomistic/continuum coupling in computational materials science. *Modelling Simul. Mater. Sci. Eng.*, v11, R33-R68.

Gao, H, Huang, Y and Abraham, F.F. (2001): Continuum and atomistic studies of intersonic crack propagation. *J. Mech. Phys. Solids*, v49, 2113-2132.

Guo, Y. and Nairn, J.A. (2004): Calculation of J-integral and stress intensity factors using the material point method. *CMES*, v6 n3, pp.295-308.

Hornung, R.D. and Kohn, S.R. (2002): Managing application complexity in the SAMRAI object-oriented framework. *Concurrency and Computation: Practice and Experience*, v14, pp. 347-368.

Horstemeyer, M.F. and Baskes (2000): M.I., Strain tensors at the atomistic scale. *Mat. Res. Soc. Symp. Proc.*, v578, pp.15-20.

Kohlhoff, S., Gumbsch, P. and Fischmeister, H.F. (1991): Crack propagation in b.c.c. crystals studied with a combined finite-elment and atomistic model. *Phil. Mag. A*, v64 n4, pp.851-878.

Ma, J., Wang, B., Lu, H., Roy, S., Hornung, R., Wissink, A. and Komanduri, R. (2005): Multiscale simulations using generalized interpolation material point method (GIMP) and SAMRAI parallel processing. *CMES*, v8 n2, pp. 135-152.

Marc, G. and McMillan, W.G. (1985): The virial theorem. *Adv. Chem. Phys.*, v58, pp.209-361.

Plimpton, S. J. (1995): Fast Parallel Algorithms for Short-Range Molecular Dynamics. *J Comp Phys*, 117, 1-19.

Rafii-Tabar, H., Hua, L. and Cross, L. (1998): A multi-scale atomistic-continuum modeling of crack propagation in a two-dimensional macroscopic plate. *J. Phys.: Dondens. Matter*, v10, pp.2375-2387.

Rudd, R. E. and Broughton, J. Q. (1998): Coarse-grained molecular dynamics and the atomic limit of finite elements. *Phys. Rev. B*, v58 n10, pp. R5893-5896.

Shen, S. and Atluri, S. N. (2005): A tangent stiffness MLPG method for atom/continuum multiscale simulation. *Computer Modeling in Engineering & Sciences*, v7, n1, , pp. 49-67.

Shiari, B., Miller R.E. and Curtin, W.A. (2005): Coupled atomistic/discrete dislocation simulations of nanoindentation at finite temperature. *J. Eng. Mater. And Tech.*, v127, pp.358-368.

Shilkrot, L.E., Miller, R.E. and Curtin, W.A. (2002): Coupled atomistic and discrete dislocation plasticity. *Phys. Rev. Lett.*, v89 n2, 025501.

Sulsky, D. and Schreyer, H.L. (1996): Axisymmetric form of the material point method with applications to upsetting and Taylor impact problems. *Comput. Methods Appl. Mech. Eng.*, v139, pp. 409-429.

Sulsky, D., Zhou, S.J. and Schreyer, H.L. (1995): Application of a particle-in-cell method to solid mechanics. *Comp. Phys. Comm.*, v87, pp. 236-252.

Zhou, M. and McDowell, D.L. (2002): Equivalent continuum for dynamically deforming atomistic particle systems. *Phil. Mag. A*, v82 n13, pp.2547-2574.

Zimmermann, J.A. (1999): Continuum and atomistic modeling of dislocation nucleation at crystal surface ledges. Ph.D. dissertation, Stanford University.

Zimmermann, J.A., Webb III, E.B., Hoyt, J.J., Jones, R.E. Klein, P.A. and Bammann, D.J. (2004): Calculation of stress in atomistic simulation. *Modeling Simul. Mater. Sci. Eng.*, v12, S319-S332.

Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of the Mechanics and Physics of Solids
53 (2005) 2718–2741
JOURNAL OF THE MECHANICS AND PHYSICS OF SOLIDS
www.elsevier.com/locate/jmps

Combined numerical simulation and nanoindentation for determining mechanical properties of single crystal copper at mesoscale

Y. Liu[a], B. Wang[a], M. Yoshino[b], S. Roy[a], H. Lu[a], R. Komanduri[a,*]

[a] *Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, OK 74078, USA*
[b] *Mechanical and Control Engineering, Tokyo Institute of Technology, Tokyo, Japan*

Received 25 December 2004; accepted 8 July 2005

Abstract

Constitutive laws are critical in the investigation of mechanical behavior of single crystal or polycrystalline materials in applications spanning from microscale to macroscale. In this investigation, a combined FEM simulation and experimental nanoindentation approach was taken to determine the mechanical behavior of single crystal copper incorporating the mesoplastic constitutive model. This model was implemented in a user-defined subroutine in 3D ABAQUS/Explicit code. Nanoindentation was modeled using the multiscale modeling technique involving mesoplasticity and elasticity, i.e., mesoplastic constitutive model was used near the local nanoindentation region (where the dislocations are generated) while elastic constitutive model was used in rest of the region in the workmaterial. The meso-mechanical behavior of the crystalline structure and the effect of the mesoplastic parameters on the nanoindentation load–displacement relationships were investigated in the FEM analysis. Nanoindentation tests were conducted on single crystal copper to determine load–displacement relationships. Appropriate mesoplastic parameters were determined by fitting the simulated load–displacement curves to the experimental data. The mesoplastic model, with appropriate parameters, was then used to determine the stress–strain relationship of a single crystal copper at meso-scale. The effect of indenter radius (3.4–1000 μm) on material hardness

*Corresponding author. Tel: +405 744 5900; fax: +405 744 7873.
E-mail address: ranga@ceat.okstate.edu (R. Komanduri).

0022-5096/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jmps.2005.07.003

under nanoindentation was simulated and found to match the experimental data for several indenter radii (3.4, 10 and 500 μm). A comparison of the topographies of nanoindentation impressions in the experiments with FEM results showed a reasonably good agreement.
© 2005 Elsevier Ltd. All rights reserved.

Keywords: Finite element method (FEM); Multiscale modeling; Mesoplasticity; Single crystal copper; Nanoindentation

1. Introduction

Material behavior at all scales spanning from atomistic to continuum involves such processes as elastic deformation, dislocation generation and multiplication, cleavage, void/microcrack formation and its growth into macrocracks, and final failure. In recent times, multiscale modeling has received increasing attention due to its potential linkage between structure–property relationships from nano- to macro-levels (King et al., 1995; Shenoy et al., 1999; Ogata et al., 2001; Smith et al., 2001; Shilkrot et al., 2002). Various computer simulation methods are used in the development of scaling laws as a function of spatial and temporal variables. At the subatomic level, quantum mechanics or ab initio calculations are used to develop the potential-energy functions. With increase in length (or time) scale, material behavior is modeled using molecular dynamics (MD) and/or Monte Carlo (MC) simulations, then to micro or mesoplastic, and finally to continuum mechanics (e.g. finite element method (FEM)). Mesoplasticity serves as an appropriate formalism that bridges the atomistic mechanisms of deformation and fracture to the macroscopic behavior (Hartley, 2003). This reinforces the importance of studying the material behavior under mesoscale, which is the subject of this investigation.

Ductile single crystal materials are of considerable interest in many applications including micro-electro-mechanical systems (MEMS), optical, and telecommunication systems. Single crystals possess anisotropic material properties that depend on the crystal orientation. The conventional tensile tests are difficult to conduct at nano- and micro-scales to determine orientation dependent behavior. Indentation can be an alternative approach to tension/compression to investigate the mechanical behavior of single crystals. Flom and Komanduri (2002) conducted indentation and scratching experiments on single crystal and polycrystalline materials. They showed marked anisotropies in the slip systems surrounding the indentations depending on the crystal orientation and direction of scratching. Recently, the nanoindentation technique (Oliver and Pharr, 1992) has become a useful tool to determine the mechanical properties of very small volume of materials.

Lim et al. (1998) and Lim and Chaudhri (1999) conducted nanoindentation tests to investigate the effect of indenter load (1–100 mN) and radius of the spherical indenter (7, 30, 200, and 500 μm) on the nano and microhardness of polycrystalline work-hardened and annealed oxygen-free copper (OFC) specimens. For the annealed OFC, they reported a gradual increase in normal pressure or Meyer hardness, P_m with a/R, where a and R are the radii of the indent and the indenter,

respectively. A similar variation was observed in the present investigation. They also reported that P_m versus a/R data points for indenters of radii 60, 200, and 500 μm all fall on the same curve. The equation for the best fit is given by $P_m = 0.623(a/R)^{0.498}$. However, for indenters of smaller radii used (7 and 30 μm), the P_m values were higher than for the larger indenters (60, 200, and 500 μm). Also, among the two smaller indenters, for a given a/R value, the indenter with the smaller radius gave higher values of normal pressure than for the indenter with larger radius. From this the authors concluded that the higher Meyer hardness, P_m indicates a scaling effect for the two smaller radii indenters. For the work-hardened OFC, they reported an initial increase in P_m with a/R, reaching a steady-state value beyond a/R of 0.15. They also reported the value of normal pressure, P_m, for a given value a/R, to be the same for indenters of radii 200 and 500 μm, but for 7 μm it was only slightly higher than from 200 and 500 μm. From this, the authors concluded that for nanoindentations and microindentations of work-hardened copper, no scaling effects were found as far as P_m versus a/R relationship is concerned. It will be shown in the present investigation that size effect is not present for a range of indenter radii used for the work-hardened single crystal copper through measurement and simulation. It may also be noted that Lim and Chaudhri (1999) and Lim et al. (1998) conducted nanoindentation studies on polycrystalline copper specimens (OFC), both annealed and work-hardened, using a Berkovich diamond indenter as well as spherical indenters of different radii, namely, 7, 30, 60, 200, and 500 μm. In situations where indenter radius was much smaller than the grain size and indentation was made in the middle of a grain, their results might be representative for the behavior of a single crystal copper. The grain size of OFC was reported to be in the range of 30–100 μm. While the 100 μm grains may be adequate for the smaller radii, for example, 7 and 30 μm indenters to justify that indentation was made on a single crystal, for radii larger than this, the boundaries would influence the flow of material so that indentation can no longer be considered to be on a single crystal unless the depth of indent is kept small. Also, as the sample used by Lim and Chaudhri (1999) and Lim et al. (1998) is a polycrystalline material, the orientations of grains change from one grain to another, albeit small, and it is not known what the orientation of the sample is for any grain.

Since nanoindentation tests can now be carried out with ease while maintaining accuracy as high as a few μN in load and 0.1 nm in displacement, material characterization at sub-micrometer scale is possible. However, the 3D nanoindentation problem spanning from a sub-micrometer scale to a millimeter scale needs to be solved to determine how the microstructure (crystal orientation, active slip systems, etc.) is linked to the load–displacement relationships.

Mesoplastic constitutive relationships have been investigated by many researchers, including, Hill and Rice (1972), and Asaro and Needleman (1985). Using mesoplastic theory, Peirce et al. (1982) analyzed numerically the nonuniform and localized deformation in ductile single crystals subjected to tensile loading. Huang (1991) developed a material user subroutine incorporating mesoplasticity in the ABAQUS implicit program. The theoretical framework of Hill and Rice (1972) has been implemented in the 2D subroutine. Kalidindi et al. (1992) developed an implicit

time-integration procedure mainly based on the constitutive model of Asaro and Needleman (1985). Kalidindi et al. (1992) simulated the load–displacement curve in plane-strain forging of copper that captures major features of the experimental data. They reported jumps in the calculated curve although no jumps were observed in the experimental data. Yoshino et al. (2001) applied mesoplasticity theory in 2D FEM to simulate the dislocation generation and propagation during indentation of single crystal silicon. Kysar (2001) studied the crack growth along a copper/sapphire bi-crystal interface.

With increasing computational power at reducing costs, mesoplastic constitutive relation can be implemented in some FEM codes to solve complex 3D problems. For example, Fivel et al. (1998) developed a 3D model to combine discrete dislocations with FEM for nanoindentation simulation on single crystal copper. They, however, did not include the unloading portion of the load–displacement curve, and did not compare simulation results with nanoindentation data. Wang et al. (2004) studied the dependence of nanoindentation pile-up patterns and micro-textures on the crystallographic orientation on single crystal copper using a conical indenter. They, however, reported an order of magnitude deviation of the numerical load–displacement results from the experimental data. Most of these studies used the material properties of single crystal data obtained from tensile testing at macro-scale. However, it is essential to address these parameters at the submicrometer scale when micro- or meso-scale behavior is of concern. To date, quantitative agreement between mesoplastic/elastic coupling simulation and nanoindentation/micro-tension results does not exist. It is the objective of this investigation to provide this information.

In this paper, a combined FEM/nanoindentation approach was developed to determine the material properties of single crystal copper incorporating mesoplastic constitutive model. This model was implemented in a user-defined subroutine in 3D ABAQUS/Explicit. The anisotropy of single crystal copper was considered using a linear elasticity model with cubic symmetry at the continuum scale. The 3D multiscale FEM model developed in this study involves both mesoplasticity and elasticity, and is used to simulate nanoindentation with a spherical indenter. Concurrently, nanoindentation tests were conducted on a single crystal copper. Through fitting numerical solution with the experimental load–displacement data, the parameters in the mesoscale constitutive model are determined. Using these parameters, the stress–strain relationship can be determined in the uniaxial stress state at the sub-micrometer scale, or to solve other mesoplastic problems. Using this relationship we have investigated the effect of indenter radius (3.4–1000 μm) on material hardness in nanoindentation by FEM simulations.

2. Mesoplasticity model

2.1. Theoretical background

Motivated by experimental observations, it has been postulated that plastic deformation in a face-centered cubic (FCC) crystal is confined to well-defined slip

systems. For an FCC lattice, plastic deformation occurs on the slip systems defined by the {1 1 1} family of slip planes in the ⟨1 1 0⟩ family of slip directions. These 12 slip systems govern the macroscopic distortion of the crystal.

The kinematics and constitutive relationships of mesoplasticity that follow those of Peirce et al. (1983) are used in this paper. The velocity gradient can be expressed as

$$\mathbf{L} = \mathbf{D} + \mathbf{\Omega}, \tag{1}$$

where $\mathbf{D}$ is the symmetric part of the rate of deformation and $\mathbf{\Omega}$ is the skew symmetric spin tensor. $\mathbf{D}$ and $\mathbf{\Omega}$ are additively decomposed into elastic parts ($\mathbf{D}^e$ and $\mathbf{\Omega}^e$) and plastic parts ($\mathbf{D}^p$ and $\mathbf{\Omega}^p$), respectively, as follows:

$$\mathbf{D} = \mathbf{D}^e + \mathbf{D}^p, \tag{2}$$

$$\mathbf{\Omega} = \mathbf{\Omega}^e + \mathbf{\Omega}^p. \tag{3}$$

The elastic parts correspond to elastic deformations and lattice rotations, whereas the plastic parts represent the plastic shear slips of the lattice. In the context of the deformation of FCC crystals, $\mathbf{D}^p$ and $\mathbf{\Omega}^p$ are related to the plastic shear strain rate on each slip system as

$$\mathbf{D}^p = \sum_{k=1}^{12} \mathbf{P}^{(k)} \dot{\gamma}^{(k)}, \tag{4}$$

$$\mathbf{P}^{(k)} = \tfrac{1}{2}(\mathbf{b}^{(k)} \otimes \mathbf{n}^{(k)} + \mathbf{n}^{(k)} \otimes \mathbf{b}^{(k)}), \tag{5}$$

$$\mathbf{\Omega}^p = \sum_{k=1}^{12} \mathbf{W}^{(k)} \dot{\gamma}^{(k)}, \tag{6}$$

$$\mathbf{W}^{(k)} = \tfrac{1}{2}(\mathbf{b}^{(k)} \otimes \mathbf{n}^{(k)} - \mathbf{n}^{(k)} \otimes \mathbf{b}^{(k)}), \tag{7}$$

where $\mathbf{n}^{(k)}$ denotes the normal vector of the slip plane, $\mathbf{b}^{(k)}$ is the slip direction, k is the number of slip system, $\otimes$ indicates the dyadic product of two tensors or vectors, and $\dot{\gamma}^{(k)}$ is the shear strain rate.

When a stress $\boldsymbol{\sigma}$ is applied to a material, the resolved shear stress $\tau^{(k)}$ on each slip system can be computed as

$$\tau^{(k)} = \mathbf{P}^{(k)} \boldsymbol{\sigma}, \tag{8}$$

$\dot{\gamma}^{(k)}$can be expressed as a rate-dependent power-law proposed by Hutchinson (1976)

$$\dot{\gamma}^{(k)} = \dot{\gamma}_0 \operatorname{sgn}(\tau^{(k)}) \left| \frac{\tau^{(k)}}{\tau_0^{(k)}} \right|^{1/m}, \tag{9}$$

where m is the rate sensitivity exponent, $\dot{\gamma}_0$ is the shear strain rate at a reference condition and $\tau_0^{(k)}$ denotes the current shear strength of the slip system. $\tau_0^{(k)}$ develops with the evolution of slips on active slip systems due to the accumulation of dislocations in a crystal, i.e., work hardening. Since work hardening of slip systems

depends on the shear deformation, variation of $\tau_0^{(k)}$ is estimated by

$$\Delta\tau_0^{(k)} = \sum_{i=1}^{12} h_{ki}|\Delta\gamma^{(i)}|, \tag{10}$$

where h_{ki} contains the hardening moduli. The diagonal components of h_{ki} represent the self-hardening moduli, whereas the off diagonal components represent the latent-hardening moduli. In this study, anisotropic hardening rule shown by Eq. (11) is introduced in the simulation (Yoshino et al., 1997).

$$h_{ki} = \alpha\delta_{ki} + \beta H_{ki}, \tag{11}$$

where α and β are parameters, H_{ki} denotes the hardening matrix of dislocation interaction defined by Bassani and Wu (1991). According to their study, H_{ki} has five independent parameters, taking into account, the interactions of dislocations in various slip systems. In this study, the hardening law has included both self-hardening and latent-hardening of each slip system as well as the interactions of dislocations among different slip systems.

Assuming that the crystal elasticity is unaffected by slip, the elastic constitutive equation is given by

$$\overset{\nabla}{\boldsymbol{\sigma}}{}^{e} = \mathbf{C}^{e} : \mathbf{D}^{e}, \tag{12}$$

where $\mathbf{C}^e$ is the tensor of elastic moduli having the major and minor symmetries. For cubic crystals, there are three independent components, C_{11}, C_{12} and C_{44}. $\overset{\nabla}{\boldsymbol{\sigma}}{}^{e}$ is the Jaumann rate of Cauchy stress $\boldsymbol{\sigma}$, which is the co-rotational stress rate in terms of the coordinate system that rotates with the lattice. $\overset{\nabla}{\boldsymbol{\sigma}}{}^{e}$ is determined by

$$\overset{\nabla}{\boldsymbol{\sigma}}{}^{e} = \overset{\nabla}{\boldsymbol{\sigma}} + \boldsymbol{\Omega}^{p} \cdot \boldsymbol{\sigma} - \boldsymbol{\sigma} \cdot \boldsymbol{\Omega}^{p}, \tag{13}$$

where $\overset{\nabla}{\boldsymbol{\sigma}}$ is the co-rotational stress rate on the coordinate system that rotates with the material. Combining Eqs. (4), (6), (12), and (13) leads to the constitutive equation of mesoplasticity in the form of

$$\overset{\nabla}{\boldsymbol{\sigma}} = \mathbf{C}^{e} : \mathbf{D} - \sum_{k=1}^{12} [(\mathbf{W}^{(k)} \cdot \boldsymbol{\sigma} - \boldsymbol{\sigma} \cdot \mathbf{W}^{(k)}) + \mathbf{C}^{e} : \mathbf{P}^{(k)}]\dot{\gamma}^{(k)}. \tag{14}$$

The mesoplastic constitutive law expressed by Eq. (14) to model the meso-mechanical behavior of single crystal copper is used in this investigation.

2.2. *Numerical implementation*

Based on the theory presented in Section 2.1, the mesoplastic constitutive model is implemented in a commercial finite element code (ABAQUS/Explicit), using a user-defined subroutine VUMAT (ABAQUS Manual, 2003). Details of the numerical implementation are described in the following. To account for the arbitrary crystal orientations, two coordinate systems, namely, global (specimen)

and local (crystalline) coordinate systems are introduced in the simulation. The global coordinate system is fixed to the reference configuration while the local coordinate system is aligned with the crystal lattice and rotates in the same way as the lattice. At the beginning of each time increment, the global strain increment, the time increment, global stress, and the solution-dependent state variables are given to the subroutine from the main program in ABAQUS. The user-defined subroutine transforms the global strain increment and stress into the local system.

The mesoplastic constitutive calculations are performed in the local coordinate system. In this coordinate system, the incremental stress is computed and transformed into the global system. At the end of each incremental step, the stress and state variables are updated for use in the main program in ABAQUS. Then, the next load increment is applied and a new strain increment is generated. This loop is repeated until computation at all incremental time steps is completed.

3. Nanoindentation tests

An MTS Nano Indenter (XP system) was used for nanoindentation tests on single crystal copper specimen to obtain load–displacement relationships. The system can reach a maximum indentation depth of 500 μm and a maximum load of 500 mN. The load resolution is 50 nN and the displacement resolution is 0.2 nm. All nanoindentation tests were performed in air at room temperature (23 °C). The tests do not start until a thermal equilibrium state is reached and the drift of the indenter tip drops below a set value, typically, 0.05 nm/s. After the indenter tip has made contact with the specimen surface, the indentation load is gradually applied to the surface of the specimen. The indentation load and displacement are recorded simultaneously at a sampling rate of five data points per second. Spherical diamond indenter with a radius of 3.4 μm was used in the nanoindentation tests.

A single crystal copper (10 × 10 × 1 mm) was cut from an ingot using electrical discharge machining (EDM). The surface was chemo-mechanically polished to a surface roughness, Ra of 9.08 nm. Each free surfaces in the specimen corresponds to the ⟨1 0 0⟩ direction, and the upper and lower surfaces are the (1 0 0) faces. Fig. 1 is a schematic of the nanoindentation setup.

4. Combined FEM/nanoindentation approach

This section describes the combined FEM/nanoindentation approach used to determine the material properties for single crystal copper incorporating the mesoplastic constitutive model. Various parameters in the mesoplastic model are determined through a fitting procedure that allows good correlation in the load–displacement relationship between the simulation results and the nanoindentation data. Nanoindentation provides the load–displacement curve, as well as some basic material parameters, such as Young's Modulus and surface hardness. On the other hand, for a single crystal copper, the mesoplastic constitutive model and the

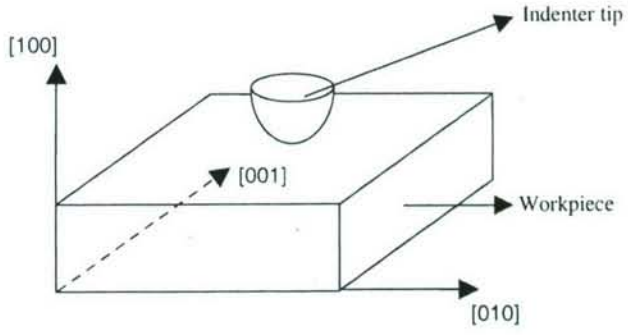
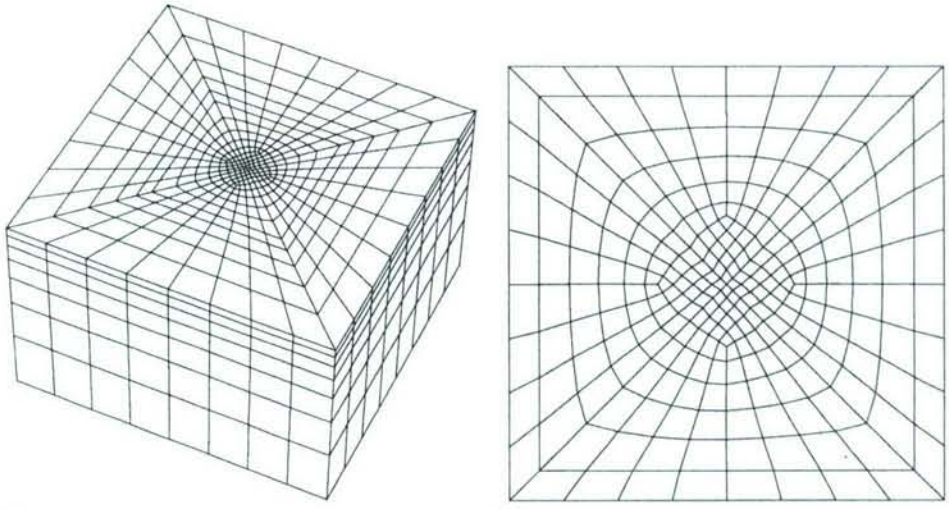


Fig. 1. Schematic of nanoindentation of a single crystal copper.

elastic model are used in the finite element analysis to simulate the nanoindentation tests and to obtain numerical load–displacement relationships. If the load–displacement relationship determined from the numerical analysis and experiment agrees, the parameters in the mesoplastic model would be suitable for the single crystal copper.

We next turn to the description of FEM analysis of nanoindentation on a single crystal copper. The size of the workpiece in the FEM model is $15 \times 15 \times 10\,\mu m$. Although this is much smaller than the actual dimensions of the specimen used in nanoindentation, it is much larger than the maximum indentation depth ($\sim$300 nm). So, the condition of indentation into a half-space is justified. The 3D finite element mesh is shown in Fig. 2(a). There are 2688 eight-node brick elements and 3208 nodes in this model. The size of the smallest elements is 140 nm. Since the primary focus in the specimen is the material directly underneath the indenter, a very fine mesh is used near the indenter tip and a coarser mesh for the remaining region. Fig. 2(b) shows the mesh in the vicinity of the indenter. The dimensions of the nanoindentation involve multiple length scales, which are reflected in the simulations. This is implemented in the simulations using constitutive equations at two length scales, namely, meso- and continuum-scales. Therefore, the workpiece is partitioned into two parts and the corresponding constitutive relationships are assigned. The dimensions of the region where mesoplastic constitutive model is applied are $6 \times 6 \times 10\,\mu m$.

The nanoindentation, as shown in Fig. 1, is simulated on a single crystal copper specimen oriented in the [100] direction. The indenter used is a rigid sphere with a radius, R of 3.4 μm. A frictionless contact pair, implemented by two contact surfaces with associated nodes between the indenter and workpiece is defined. All nodes on the bottom surface of the workpiece are constrained along the [1 0 0] direction and a constant velocity is applied to simulate the indenter motion during both loading and unloading. Since explicit dynamic finite element analysis is conducted, the time step increment is determined based on the smallest element dimension in the mesh and the dilatational wave speed. In this study, a fixed time step increment 60 ps (1 ps = 10^{-12} s) is chosen as a compromise between stability of the dynamic simulation and the overall computational time. A total time step increments of 66,967 in the simulation are used, including loading, intermediate transition, and unloading. The



(a) (b)

Fig. 2. FEM mesh used for the nanoindentation simulations: (a) overview on the mesh; (b) zoomed-in view of localized indentation area.

intermediate transition step entails ramping down the velocity to remove the numerical discontinuities in the resistance force caused by the sudden change in the direction of the indenter motion.

5. Results and discussion

5.1. Nanoindentation

The nanoindentation tests were conducted on a single crystal copper specimen. The maximum indentation displacement was 310 nm. For each indentation displacement, nanoindentation tests were conducted at four different locations to ensure repeatability of the experimental data. The actual load–displacement results are shown in Fig. 3. These four curves are very close to each other and the maximum deviation in load at the maximum displacement is 0.07 mN. Loading involves nonlinearity induced by the nonlinear material behavior and increasing contact area (geometric nonlinearity). Unloading is purely elastic. However, as the contact area decreases with decrease in indentation displacement, geometric effects lead to a nonlinear unloading curve.

5.2. Determination of mesoplastic parameters

The finite element computational scheme is implemented in the ABAQUS/Explicit code and used to model the nanoindentation problem. As stated in Section 2.1,

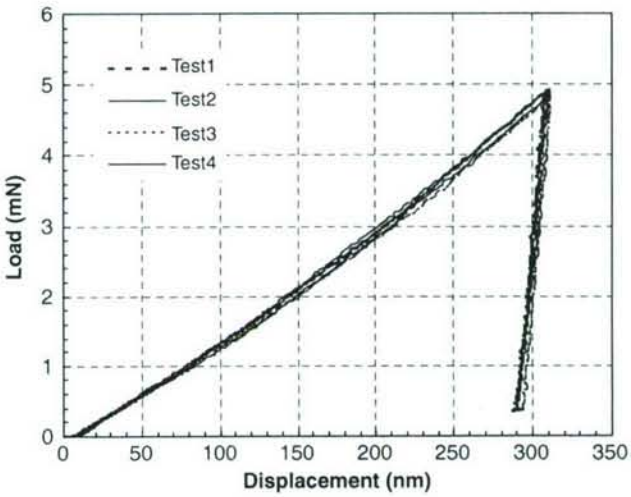


Fig. 3. Experimental load–displacement plots at maximum indentation depth of 310 nm.

mesoplastic constitutive equation used herein depends on the plastic strain rate. In the numerical algorithm, m is a numerical parameter that is adjusted to satisfy the Schmid's law. When $m = 0$, Schmid's law is satisfied accurately. However, this condition is difficult to implement in computations due to numerical instability. To resolve this problem, m should be slightly larger than zero, for e.g. 0.02. With such a small value of m, the shear stress that activates a slip system almost corresponds to the yield shear stress. The condition $m = 1$ implies that shear stress $\tau^{(k)}$ is proportional to sliding rate $\dot{\gamma}^{(k)}$, a relation identical to a Newtonian fluid. Such a situation is not realistic in crystal plasticity. Additionally, in Eq. (9), the initial value of the current strength $\tau_0^{(k)}$ is the same in all slip systems, and is given as a user input parameter τ_{in}. As τ_{in} decreases, the slip system becomes easier to activate. As a result, higher plastic deformation incurs at a given indentation force.

We turn next to finding the appropriate parameters, τ_{in} and the hardening moduli, α and β in mesoplasticity via a direct comparison between the simulation results and the corresponding experimental data. Through multiple trial runs, using the parameters given in Table 1, a reasonably good agreement between the numerical and experimental load–displacement curves can be achieved as shown in Fig. 4. It can be seen that the numerical model has captured major features as well as the magnitude of the experimental data. The resisting force on the indenter increases as the indentation depth increases and drops precipitously upon withdrawal of the indenter. The simulated curve agrees well with the experimental curve at loading/unloading.

To investigate the sensitivity of mesh size on the convergence of the numerical solution, the mesh size with the smallest element size of 140 nm was used as a benchmark. Three additional mesh cases, namely, 10% smaller, 10% larger, and 50% larger were tested for the same model and the simulation results were

Table 1
Material parameters of single crystal copper

C_{11} (GPa)	C_{12} (GPa)	C_4 (GPa)	τ_{in} (MPa)	$\dot{\gamma}_0$	α (MPa)	β (MPa)	m
168.4	121.4	75.4	34.8	1.0	4.0	0.6	0.02

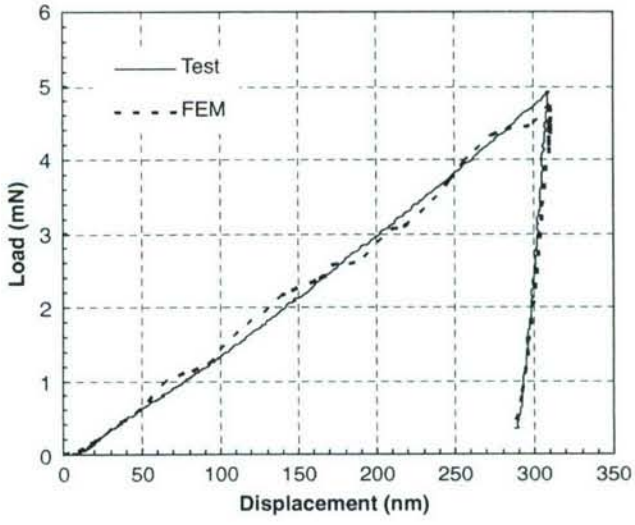


Fig. 4. Comparison between numerical and experimental load–displacement curves.

compared. For the cases of 10% larger and 10% smaller, the two load–displacement curves exhibited the same basic features as the benchmark curve, and the maximum deviation among these curves was 5.6%. For the case of 50% larger element size, more oscillations with amplitude of ~0.5 mN were observed in the load–displacement curve and the maximum deviation from the benchmark curve was 9.3%.

Additionally, since the current model size is large enough with respect to the nanoindentation depth of 310 nm, only the dimensions of the part with mesoplasticity were changed to test the numerical results. Two different cases, namely, $6 \times 6 \times 10\,\mu m$ and $12 \times 12 \times 10\,\mu m$ were simulated and a maximum error in the load–displacement curves was 6.2%. Based on the sensitivity analysis, it can be concluded that the simulation results have converged with respect to the mesh spacing for the benchmark mesh size and the model size used.

Although there is good overall agreement between the experimental and the simulation results, the loading portion of the simulation curve is not smooth, despite various attempts to obtain a smoother curve. While the dynamic characteristics of the simulation might explain partially the bumps in the curve, it cannot explain all, as most portions of the curve are still fairly smooth. It may be noted that similar nonsmooth loading curves in indentation were also reported by others

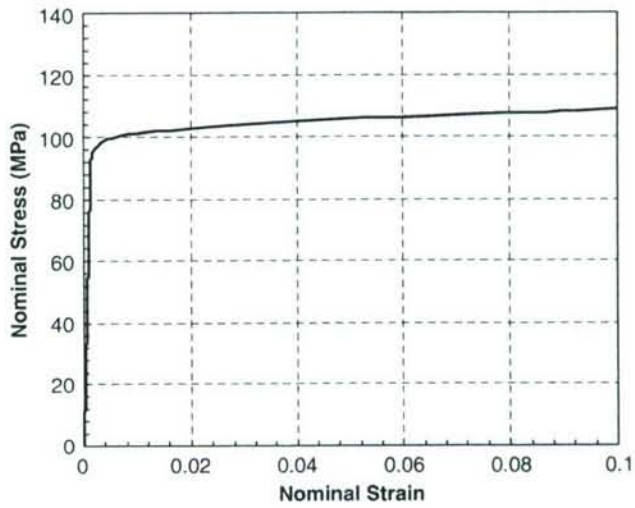


Fig. 5. Stress–strain curve for single crystal copper at meso-scale.

(Kalidindi et al., 1992; Smith et al., 2001). Smith et al. (2001) attributed this problem to the finite size of the elements. Kalidindi et al. (1992) postulated that these jumps can be minimized by using a finer mesh on the surface of the sample. The jagged load–displacement curve presented here shows a trend similar to what others have observed at the meso-scale. However, further investigation is needed to determine if there is an underlying mechanism for the jagged load–displacement curve, or, if it is only due to computational limitations.

Various parameters obtained using the combined FEM/nanoindentation approach are then employed to determine the stress–strain relationship of single crystal copper at meso-scale. Fig. 5 shows the stress–strain curve of single crystal copper at meso-scale.

5.3. *Mechanical behavior of single crystal copper at mesoscale*

The nanoindentation simulations were carried out along the $[\bar{1}\,0\,0]$ direction using a spherical indenter. The parameters in Table 1 were used for the mesoplastic constitutive model. Figs. 6(a) and (b) show the distribution of out-of-plane displacement profiles at two stages of indentation. They show the deformation on the (1 0 0) orientated surface with a four-fold symmetry. Flom and Komanduri (2002) observed the same deformation patterns in indentation experiments and Wang et al. (2004) also reported similar results using conical indenters from FEM simulations and experiments. Both positive and negative values of displacement can be seen from Fig. 6(a). The positive values occur symmetrically around the indent impression and represent the pile-up patterns of the material. With increase in indentation depth, the pile-ups become more and more pronounced (see Fig. 6(b)).

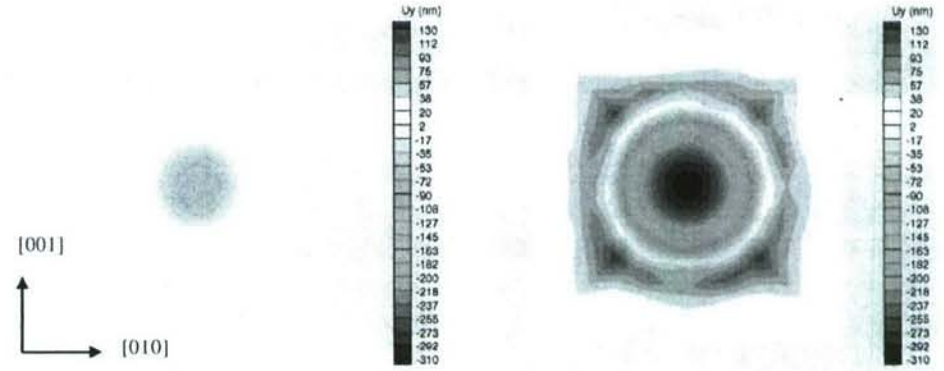


Fig. 6. Distributions of out-of-plane displacements at two stages of indentation: (a) 100 nm; (b) 310 nm.

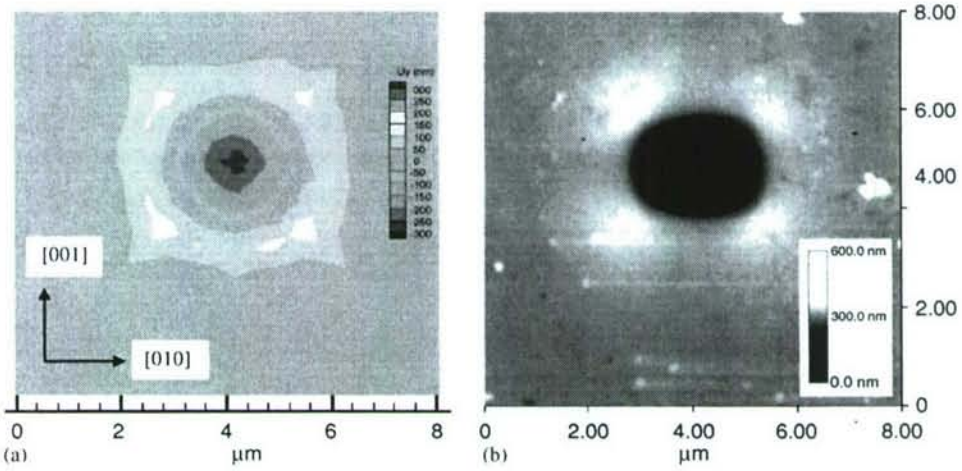


Fig. 7. Comparison of FEM and AFM images of the out-of-plane displacement at an indentation depth of 310 nm: (a) Numerical results; (b) AFM image.

Figs. 7(a) and (b) show a comparison of the numerical out-of-plane displacement plot and the AFM image after unloading the indenter. Even though, it is somewhat difficult to precisely capture in the simulations all phenomena observed in experiments, both figures show some similar features, for example, a four-fold symmetry. Since self-hardening and latent-hardening for each slip system have already been taken into account in the hardening law, the effect of other active slip systems on the dislocation activities in a lattice have also been included in the simulation. It is also possible that the radius of the styli may not be exactly spherical, especially with the smaller diameter ones.

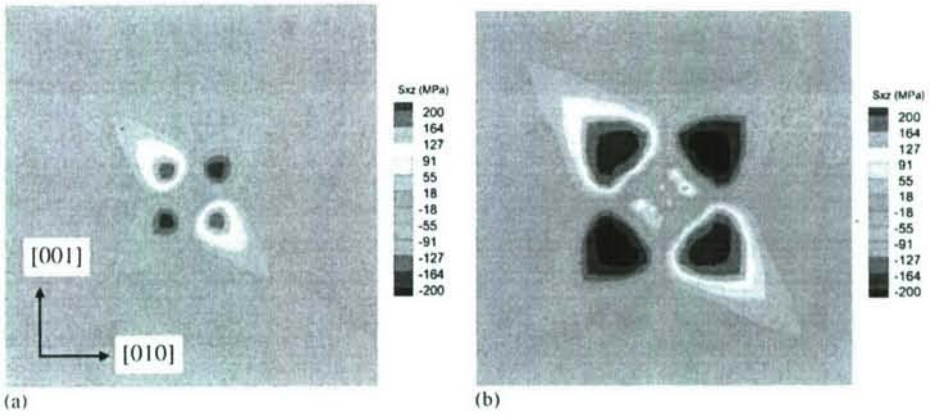


Fig. 8. Shear stress σ_{xz} distributions in the (1 0 0) plane at two stages of indentation: (a) 100 nm; (b) 310 nm.

Figs. 8(a) and (b) show the shear stress σ_{xz} distribution at two stages of indentation. Using a spherical indenter in nanoindentation, the stress waves propagate towards the four vertices in the (1 0 0) plane and the stress magnitudes increase with the indentation displacement. Figs. 9(a) and (b) show the distribution of normal stress σ_{yy} (y-axis coincides with the [1 0 0] direction) in the mid-section (the normal direction is along [0 0 1]) of the single crystal at two stages of indentation. The stress distribution is symmetric with respect to the [1 0 0] direction. These figures also indicate that maximum compressive stress occurs right underneath the indenter with its magnitude increasing with indentation depth, and the domain of the compressive stress increases with the indentation displacement. Tensile stress zones become larger and larger with increasing indentation displacement, causing more significant pile-ups.

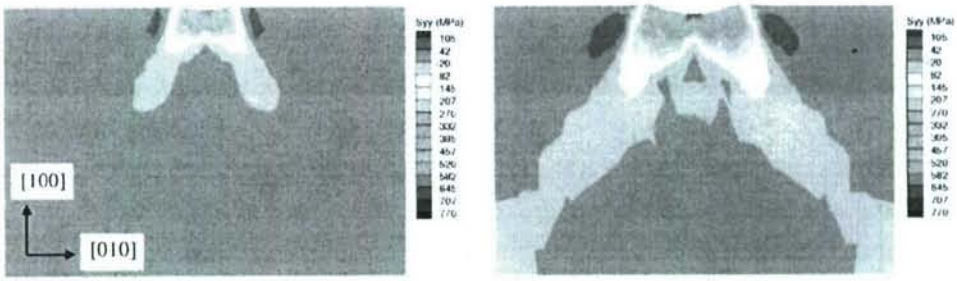
Fig. 10 shows a zoomed-in view of the pile-up on a displacement plot in the same section as that in Fig.9, after unloading the indenter. Figs. 11(a) and (b) show the distributions of the shear strain in the mid-section at two stages of indendation. The total shear strain is calculated using the following equation:

$$\gamma = \sum_k |\Delta\gamma^{(k)}|. \tag{15}$$

The shear strain is related to the status of each slip system by the Schmid's law, and is associated with the dislocation density as well. As Figs. 11(a) and (b) show, the shear strain distribution is symmetric with respect to the [1 0 0] direction.

5.4. *Effect of friction on nanoindentation behavior*

We now compare the pile-up profiles of the indent between the numerical and experimental data. The dashed line in Fig. 12 shows the surface profile scanned



(a) (b)

Fig. 9. σ_{yy} distributions in the (0 0 1) section at two stages of indentation: (a) 100 nm; (b) 310 nm.

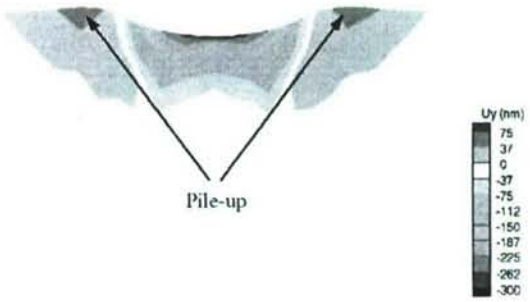
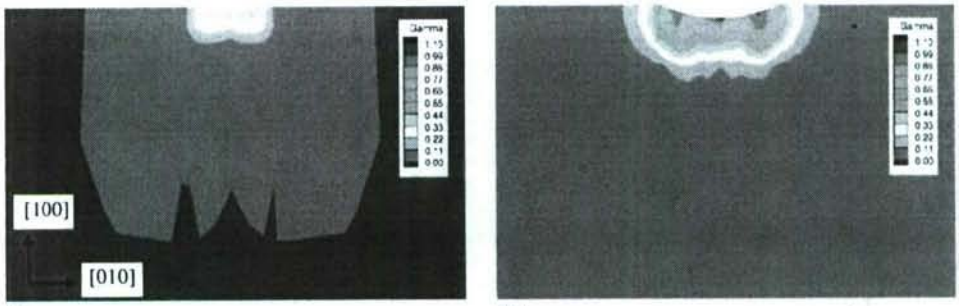


Fig. 10. Pile-up of the indentation region in the (0 0 1) section after unloading.



(a) (b)

Fig. 11. Distributions of shear strain (total shear strain on all the slip systems) in the (0 0 1) section at two stages of indentation: (a) 100 nm; (b) 310 nm.

through the center of the indentation impression along a line in the [0 1 1] direction (as determined from the AFM image in Fig. 7). Also shown is the surface profile determined from the numerical analysis for a frictionless contact condition (i.e.,

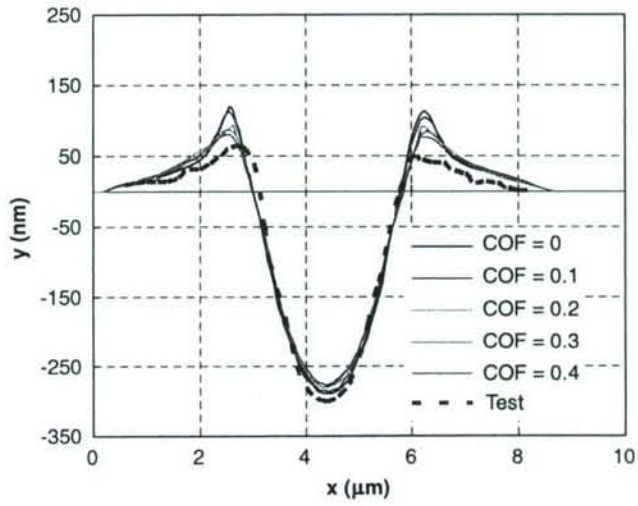


Fig. 12. Comparisons of the pile-up profiles for different COF.

coefficient of friction, COF = 0) with the highest material pile-up. It can be seen that the depth of indentation is reasonably in good agreement with the depth as determined from the AFM image (~285 nm). The material pile-up from the numerical analysis for a COF of zero is 125 nm, while the value from the experiment is 60 nm. Even though, the material pile-up is within the same order of magnitudes for both cases, the values differ by a factor of about two. This difference can be attributed partially to friction.

The analysis conducted thus far has focused on a frictionless contact between the nanoindenter and workpiece. It may, however, be noted that friction does play a role at the contact surface and can affect the surface profile. To investigate this, we introduced Coulomb frictional contact in the FEM simulation of nanoindentation. Because the COF is unknown, we used several values of COFs between $\mu = 0$ and 0.8 in the simulations, attempting to find a COF that would give us the same surface profile as measured by the AFM. A constant COF is used in each simulation. Fig. 12 shows the simulated surface profile in the [0 1 1] direction scanned through the center of the impression for COFs between 0 and 0.4. The results indicate that pile-up decreases as COF increases. We also find that as the COF reaches 0.4, further increase in the COF does not cause any additional reduction in the material pile-up. This is most likely because nonslip contact condition has been reached at a COF equals to or near 0.4, so that increasing COF simply maintains the same nonslip contact condition and does not contribute to further reduction in the material pile-up. At COF of 0.4, the simulated material pile-up is 75 nm, close to 60 nm which is the material pile-up measured from AFM. To the best of knowledge of the authors, this difference is small among the published literature at this scale lending further confirmation on the appropriateness of the mesoplastic parameters determined from the combined FEM/nanoindentation approach.

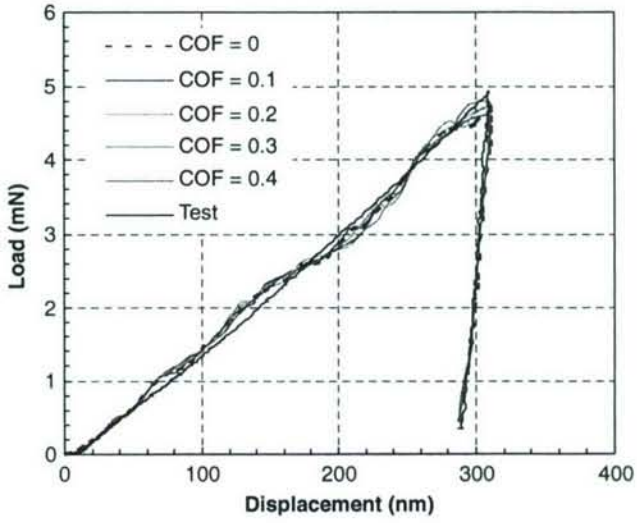


Fig. 13. Load–displacement curves for different COFs.

We next address the depth of impression as shown in Fig. 12. The indentation depth is constant and independent of the COF. Because the indentation depth is not affected by COF, the load–displacement curve should not be affected by the COF as well. It is seen that the load–displacement curves are almost identical within the range of limited numerical noise (see Fig. 13 for COFs between 0 and 0.4). As a result, this combined FEM/nanoindentation approach directed towards finding the mesoscale parameters in single crystal copper through correlating numerical load–displacement curve with the nanoindentation counterpart is not influenced by the COF at the contact surface.

5.5. *Effect of spherical indenter radius on the measured mechanical properties*

Material parameters determined from nanoindentation tests can be size-dependent, namely, indentation depth and indenter radius. It may be noted that theoretical framework for mesoplasticity used in this investigation does not encompass the size effect. Ideally, it can be applied to any scale of indentation provided the material has the same crystalline characteristics. In reality, however, the nanoindentation size effect may result in changes in the yield stress and the hardening law.

Material hardness, as one of the important mechanical properties, can be measured using the nanoindentation technique with spherical indenters of different radii. To address this problem, numerical simulations were conducted using spherical indenters with different radii (3.4, 7, 10, 25, 50, 100, 200, 500 and 1000 μm) at corresponding indentation depths (310, 638, 912, 2280, 4560, 9120, 18,240, 45,600 and 91,200 nm). FEM results indicate that pile-up on the indent

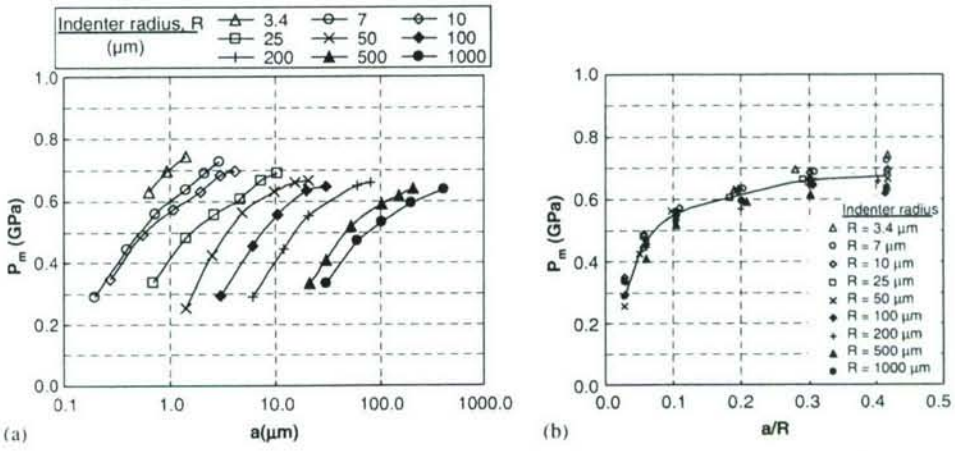


Fig. 14. Results of simulation showing the variation of mean pressure, P_m: (a) with indent radius, a for different indenter radii, R; (b) with a/R for different indenter radii.

surface using a spherical indenter with a smaller radius is more evident than with a larger radius. This is because for a given depth of indentation, smaller radii indenters are less blunt than larger radii indenters.

Fig. 14(a) shows the variation of mean pressure, P_m with indent radius, a for different indenter radii, R on a semi-log scale. It shows that normal pressure increases with increase in the indent radius. It also shows that this increase is higher for spherical indenters of smaller radii than with larger radii, especially for lower values of the indent radii. Fig. 14(b) shows the variation of mean pressure, P_m with the ratio of the indent radius to the indenter radius, (a/R) for different indenter radii, R. It is seen that a single curve can be fitted for P_m versus (a/R) for different indenter radii, when the indenter depth is normalized with respect to the indenter radius. The best fit equation is $P_m = 0.9583(a/R)^{0.2984}$. The small differences in this case are attributed to the work-hardening behavior of the sample as well as minor numerical instabilities due to mesh distortions. Lim and Chaudhri (1999) found significant differences in the variation of mean normal pressure P_m with (a/R) between the work-hardened and annealed OFC samples. For the work-hardened sample, the P_m value for a given (a/R) is only slightly greater for the indenter of 7 μm radius than for the other two indenters (200 and 500 μm) for which it is the same. The simulation results in this paper show a similar trend. To confirm the work-hardened behavior of the work material, the indents were scanned under AFM. In general, scanning along a line in a typical AFM image of an indent impression shows pile-up in a work-hardened material and sink-in in the case of an annealed specimen (McElhaney et al., 1998; Lim and Chaudhri, 1999). In the case of single crystal copper in our case, we scanned the surface along a line on an indent, and plotted the height profile in Fig. 12. A pile-up is observed around 2.5 μm. Additionally, a pile-up is observed around 4 μm along a line on an indent formed by impression of a Berkovich indenter (Fig. 15). The pile-ups in indents formed by both spherical and Berkovich indenter

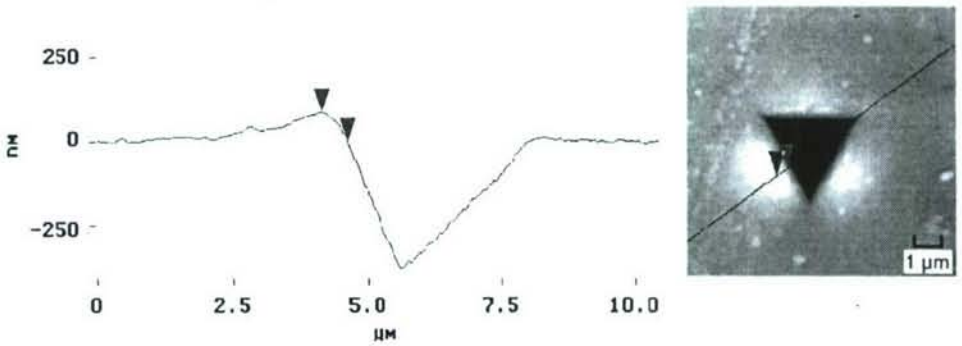


Fig. 15. AFM image of an indent impression made with a Berkovich indenter.

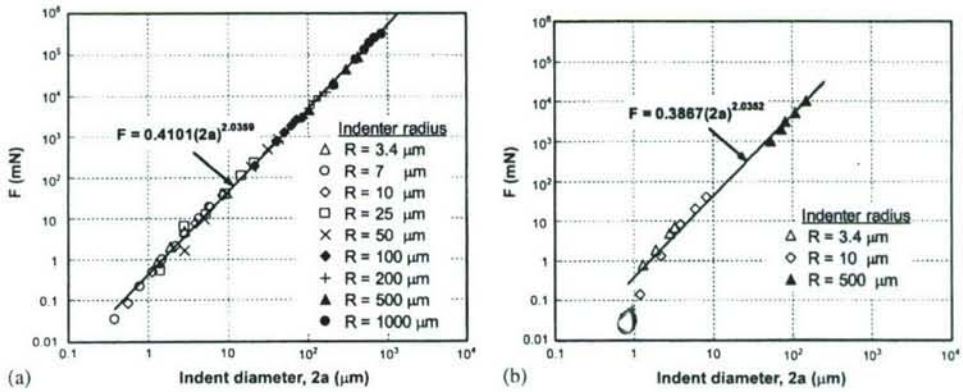


Fig. 16. (a) Results of FEM simulations showing the variation of the indentation force, F with indent diameter, $2a$ for different indenter radii, R; (b) Results of indentation experiments showing the variation of the indentation force, F with indent diameter, $2a$ for different indenter radii, R.

impressions indicate that the single crystal copper is close to work-hardened condition.

In Figs. 14(a) and (b), the variable, indent radius, is involved in both axes. To eliminate this effect, the variation of normal force with indent diameter is plotted on a log–log scale (Fig. 16(a)). It shows that most of the data points fall on a single line and independent of the indenter radius used (in this case from 3.4 to 1000 μm). A straight line can be fitted between the indenter force and the indent radius. The equation for the best fit is given by $F = 0.4101(2a)^{2.0359}$, where F is the indenter force and $2a$ is the indent diameter. It is interesting to note that this equation is very similar to the equation normally used at the macrolevel for spherical indenters, namely, $P = A(2a)^n$, where the exponent, n is called the Meyer index (1908). The value of n is reported to vary from 2 for fully strain-hardened metals to 2.5 for annealed metals (Shaw and DeSalvo, 1972). The value of the exponent, n obtained in the present investigation is 2.0359, indicating the work-hardening behavior of the single crystal copper specimen. Fig. 16(a) shows very little dependence of the

indenter radius on the normal force at the microscale, similar to the ones observed at the macroscale. This equation was shown to hold for indenters of diameters in the range of 1–30 mm by Krupkowski (1931).

In order to compare the mean normal pressure values obtained from simulations, nanoindentation and microindentation experiments were conducted for several available spherical indenters (3.4, 10 and 500 μm). The nanoindentation tests of spherical diamond indenters (radii 3.4 and 10 μm) were performed on the MTS Nano Indenter XP system, whereas the microindentation tests of spherical tungsten carbide indenter (radius 500 μm) were conducted on a Clark Micro Hardness Testers (Model CM-700AT). The experimental data are presented in Fig. 16(b). The best fit equation is $F = 0.3867(2a)^{2.0352}$. The exponent as obtained from the experiment 2.0352 is very close to the simulation, which is 2.0359. From these results, it can be seen that the approach of using combined nanoindentation/FEM simulation for determining mesoplastic model parameters works reasonably well from the microlevel to the macrolevel.

6. Discussion

The mesoplastic constitutive law used in this paper is a mechanism-based plasticity model derived from rate-dependent crystallographic theory. It is capable of capturing the effects of crystal orientations. The approach of combining numerical simulation with nanoindentation experiments allowed the determination of the stress–strain relation at mesoscale. Numerical simulations of nanoindentation using appropriate parameters for material behavior at these scales have a reasonably good agreement with the nanoindentation results on indentation mean pressure and surface pileup. It is noted, however, that the material length scale is not explicitly included in the current mesoplastic constitutive law. Therefore, it is applicable only to problems where size effects do not dominate the material behavior; it is currently not capable of modeling the size effects such as the indenter size-dependent mean pressure as reported for annealed polycrystalline copper by Lim and Chaudhri (1999). Within the framework of numerical simulations, the length scale issues can be potentially incorporated. At the continuum level, there are usually three formulations to take into consideration the size effects (Needleman, 2000): discrete dislocation plasticity, nonlocal plasticity and the coupling of matter diffusion and deformation. In discrete dislocation plasticity (Amodeo and Ghoniem, 1990; Gulluoglu and Hartley, 1993; van der Giessen and Needleman, 1995), the dislocations are modeled as line singularities in an elastic medium. The plastic flow is represented by the collective motion of a number of discrete dislocations. Dislocation-based plasticity has a characteristic length, the Burgurs vector, and is capable of capturing the size-dependent phenomena. In the nonlocal plasticity theory (Fleck et al., 1994; Nix and Gao, 1998; Shu and Fleck, 1998), the stress tensor or higher order stress tensor depends on both strains and strain gradients. Different formulations for nonlocal plasticity have been presented (see Needleman, 2000 for a review). However, there is no unified formulation in nonlocal plasticity applicable to both single crystals and polycrystalline materials at all length scales. Whether a

nonlocal plasticity theory is suitable or not depends ultimately on comparison with experimental data. Consideration of the size-dependence through the coupling of matter diffusion and deformation is usually important at high temperatures, such as the temperature encountered in the manufacturing process of semiconductor devices. In this case, surface diffusion drives the formation of islands due to the deformation of thin films. Zhang and Bower (1999) found that islands will form if the initial roughness on the surface of the film exceeds certain critical wavelength.

The work presented herein indicates that there is no size-dependence in stress–strain relation, as well as hardness on work-hardened single crystal copper. In situations where size-effects are present, as in the case of annealed copper, size effects need to be introduced in the formulation of numerical simulation. The approach used in this paper will be expanded in a future work to incorporate size-dependence in the formulation.

7. Conclusions

1. A multiscale finite element model involving meso-plasticity and elasticity was used for nanoindentation of single crystal copper. A combined FEM simulation and nanoindentation experimental approach was used to determine the parameters for the mesoplastic constitutive model. Nanoindentation tests were conducted to obtain the load–displacement characteristics. The parameters for mesoplastic constitutive model were determined by fitting the numerical load–displacement curves to the experimental data. Numerical results indicate that hardening parameters α and β and initial critical shear stress, τ_{in} have a strong effect on the nanoindentation load–displacement relationship. A reasonably good agreement between numerical and experimental results on the load–displacement curve was obtained.

2. Meso-mechanical behavior of single crystal copper was investigated. The distribution of displacements, stresses, shear strains, and shear stresses were analyzed at various stages of indentation. Numerical results show the deformation on the (1 0 0) orientated surface in nanoindentation has a four-fold symmetry similar to the AFM image of nanoindentation. This is consistent with the results reported by Flom and Komanduri (2002) and Wang et al. (2004).

3. With increase in indentation depth, material pile-up around the indenter also increases. The pile-up was more pronounced with smaller indenter radius. FEM simulation results also indicate that pile-up decreases as the coefficient of friction (COF) increases from 0 to 0.4, while the nanoindentation load–displacement relationship is found to be independent of the coefficient of friction. Simulated material pile-up at COF of 0.4 compared reasonably well with the measured pile-up. The maximum shear strain associated with the dislocation density occurs underneath the top surface which can lead to macrocrack formation. Various parameters determined from the combined FEM/nanoindentation are then used to determine uniaxial stress–strain relationships at microscale.

4. The variation of hardness, P_m with a/R shows an increasing trend in hardness with increase in (a/R) and independent of indent diameter. The equation for the best

fit is given by $P_m = 0.9583(a/R)^{0.2984}$. It is interesting to note that this equation is very similar to the equation at the macrolevel for spherical indenters, namely, $P = A(2a)^n$, where the exponent, n is the Meyer index (1908). This equation was shown to hold for indenters of diameters in the range of 1–30 mm by Krupkowski (1931). It is shown in the present investigation that this effect can be attributed to the work-hardening behavior of the material in simulations using spherical indenters of radii varying from 3.4 to 1000 μm. Similar results were reported by Lim et al. (1998) and Lim and Chaudhri (1999) for polycrystalline work-hardened copper for indenters of radii from 7 to 500 μm.

5. The variation of normal force was found to increase linearly with indent diameter, on a log–log plot, and independent of the indenter radius. The equation for the best fit is given by $F = 0.4101(2a)^{2.0359}$, where F is the normal force and $2a$ is the indent diameter. The value of n is 2.0359. The range of values of n reported to vary from 2 for fully strain-hardened metals to 2.5 for annealed metals (Shaw and DeSalvo, 1972). The value of the exponent n obtained in the present investigation is 2.0359, indicating the work-hardening behavior of the single crystal copper specimen. The exponent as determined from experiments is 2.0352 which is also very close to the simulation results.

6. The results of the present investigation (Figs. 16(a) and (b)) indicate that use of combined nanoindentation/FEM simulation approach for determining mesoplasitc model parameters works reasonably well from microlevel to the macrolevel.

7. The method presented in this study has provided an approach to determine the microstructural parameters through nanoindentation tests and numerical simulations that can be used for quantitative analysis for a wide range of problems involving mesoplastic behavior.

Acknowledgments

The work was supported by a grant from the Air Force Office of Scientific Research (AFOSR) through a DEPSCoR grant (No. F49620-03-1-0281). The authors thank Dr. Craig S. Hartley, Program Manager for Metallic Materials Program at AFOSR for his interest and support of this work. One of the authors (RK) also acknowledges the A.H. Nelson, Jr., Endowed Chair in Engineering for additional support. The authors also thank Mr. Jin Ma for nanoindentation measurements and Mr. Sony Varghese for the AFM imaging. The authors thank the reviewers for their comments and suggestions which improved the quality of the manuscript.

References

ABAQUS User's Manual, 2003. Version 6.4, ABAQUS, Inc.

Amodeo, R.J., Ghoniem, N.M., 1990. Dislocation dynamics. I. A proposed methodology for deformation micromechanics. Phys. Rev. B 41, 6958–6966.

Asaro, R.J., Needleman, A., 1985. Overview No. 42, Texture development and strain hardening in rate dependent polycrystals. Acta Metall. 33, 923–953.

Bassani, J.L., Wu, T.Y., 1991. Latent hardening in single crystals II. Analytical characterization and predictions. Philos. Trans. R. Soc. London A 435, 21–41.

Fivel, M.C., Robertson, C.F., Canova, G.R., Boulanger, L., 1998. Three-dimensional modeling of indent-induced plastic zone at a mesoscale. Acta Mater. 46, 6183–6194.

Fleck, N.A., Muller, G.M., Ashby, M.F., Hutchinson, J.W., 1994. Strain gradient plasticity: theory and experiment. Acta Metall. Mater. 42, 475–487.

Flom, D.G., Komanduri, R., 2002. Some indentation and sliding experiments on single crystal and polycrystalline materials. Wear 252, 401–429.

Gulluoglu, A.N., Hartley, C.S., 1993. Simulation of dislocation microstructures in two dimensions: II. Dynamic and relaxed structures. Model. Simul. Mater. Sci. Eng. 1, 383–402.

Hartley, C.S., 2003. A method for linking thermally activated dislocation mechanisms of yielding with continuum plasticity theory. Philos. Mag. 83, 3783–3808.

Hill, R., Rice, J.R., 1972. Constitutive analysis of elastic–plastic crystals at arbitrary strain. J. Mech. Phys. Solids 20, 401–413.

Huang, Y., 1991. A user-material subroutine incorporating single crystal plasticity in the ABAQUS finite element program. Mech. Report 178, Division of Applied Sciences, Harvard University, Cambridge, MA.

Hutchinson, J.W., 1976. Bounds and self-consistent estimates for creep of polycrystalline materials. Proc. R. Soc. London A 348, 101–127.

Kalidindi, S.R., Bronkhorst, C.A., Anand, L., 1992. Crystallographic texture evolution during bulk deformation processing of FCC metals. J. Mech. Phys. Solids 40, 537–569.

King, W.E., Campbell, G., Gonis, T., Henshall, G., Lesuer, D., Zywicz, E., Foiles, S., 1995. Theory, simulation, and modeling of interfaces in materials—bridging the length-scale gap: a workshop report. Mater. Sci. Eng. A 191, 1–16.

Krupkowski, A., 1931. A. Proprietes Mecaniques du Cuivre. Rev. Metall. 28, 641–652.

Kysar, J.W., 2001. Continuum simulations of directional dependence of crack growth along a copper/sapphire bicrystal interface. Part I: experiments and crystal plasticity background. J. Mech. Phys. Solids 49, 1099–1128.

Lim, Y.Y., Bushby, A.J., Chaudhri, M.M., 1998. Nano and macro indentation studies of polycrystalline copper using spherical indenters. Mater. Res. Soc. Symp. Proc. 522, 145–150.

Lim, Y.Y., Chaudhri, M.M., 1999. The effect of the indenter load on the nanohardness of ductile metals: an experimental study on polycrystalline work-hardened and annealed oxygen-free copper. Philos. Mag. A 79, 2979–3000.

McElhaney, K.W., Vlassak, J.J., Nix, W.D., 1998. Determination of indenter tip geometry and indentation contact area for depth-sensing indentation experiments. J. Mater. Res. 13, 1300–1306.

Meyer, E., 1908. Untersuchen ueber Haertepruefung und Haerte. Z. Ver. Deutsche Ing. 52, 645–654.

Needleman, A., 2000. Computational mechanics at the mesoscale. Acta Mater. 48, 105–124.

Nix, W.D., Gao, H., 1998. Indentation size effects in crystalline materials: a law for strain gradient plasticity. J. Mech. Phys. Solids 8, 411–425.

Ogata, S., Lidorikis, E., Shimojo, F., Nakano, A., Vashishta, P., Kalia, R., 2001. Hybrid finite-element/molecular-dynamics/electronic-density-functional approach to materials simulations on parallel computers. Comput. Phys. Commun. 138, 143–154.

Oliver, W.C., Pharr, G.M., 1992. An improved technique for determining hardness and elastic modulus using load and displacement sensing indentation experiments. J. Mater. Res. 7, 1564–1583.

Peirce, D., Asaro, R.J., Needleman, A., 1982. An analysis of nonuniform and localized deformation in ductile single crystals. Acta Metall. 30, 1087–1119.

Peirce, D., Asaro, R.J., Needleman, A., 1983. Material rate dependence and localized deformation in crystalline solids. Acta Metall. 31, 1951–1976.

Shaw, M.C., DeSalvo, G.J., 1972. The role of elasticity in hardness testing. Met. Eng. Q. 12, 1–7.

Shenoy, V.B., Miller, R., Tadmor, E.B., Rodney, D., Phillips, R., Ortiz, M., 1999. Adaptive finite element approach to atomic-scale mechanics—the quasicontinuum method. J. Mech. Phys. Solids 47, 611–642.

Shilkrot, L.E., Curtin, W.A., Miller, R.E., 2002. A coupled atomistic/continuum model of defects in solids. J. Mech. Phys. Solids 50, 2085–2106.

Shu, J.Y., Fleck, N.A., 1998. The prediction of a size effect in micro indentation. Int. J. Solids Struct. 35, 1363–1383.

Smith, G.S., Tadmor, E.B., Bernstein, N., Kaxiras, E., 2001. Multiscale simulations of silicon nanoindentation. Acta. Mater. 49, 4089–4101.

van der Giessen, E., Needleman, A., 1995. Discrete dislocation plasticity: a simple planar model. Model. Simul. Mater. Sci. Eng. 3, 689–735.

Wang, Y., Raabe, D., Kluber, C., Roters, F., 2004. Orientation dependence of nanoindentation pile-up patterns and of nanoindentation microtextures in copper single crystals. Acta Mater. 52, 2229–2238.

Yoshino, M., Shiina, T., Shirakashi, T., 1997. FEM simulation of dislocation behavior in an FCC metal. Comput. Model. Simul. Eng. 2, 218–226.

Yoshino, M., Aoki, T., Chandrasekaran, N., Shirakashi, T., Komanduri, R., 2001. Finite element simulation of plane strain elastic-plastic indentation on single crystal silicon. Int. J. Mech. Sci. 43, 313–333.

Zhang, Y.W., Bower, A.F., 1999. Numerical simulations of island formation in a coherent strained epitaxial thin film system. J. Mech. Phys. Solids 47, 2273–2297.

PHILOSOPHICAL MAGAZINE, 11 April 2004
VOL. 84, NO. 11, 1155–1183

Taylor & Francis
Taylor & Francis Group

Monte Carlo simulation of nanometric cutting

R. KOMANDURI†, R. NARULKAR

Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, Oklahoma, USA

and L. M. RAFF

Chemistry Department, Oklahoma State University, Stillwater, Oklahoma, USA

[*Received 24 June 2003 and accepted 3 November 2003*]

ABSTRACT

Nanometric cutting of single-crystal materials at conventional cutting speeds ($5\,m\,s^{-1}$) is simulated for the first time using a new Monte Carlo method that is applicable to systems that are neither canonical nor microcanonical. This is accomplished by defining a local temperature in the cutting zone using the thermal analysis developed by Komanduri and Hou for conventional machining. Extension of this method to the nanometric regime permits an accurate estimate of the local temperature in cutting. This temperature is then employed in the Boltzmann probability distribution function that is used to determine the acceptance–rejection of Monte Carlo moves in the simulation. Since cutting speed is closely related to cutting temperature, the cutting speed enters the calculation via the thermal analysis equations. The method is applied to nanometric cutting of single-crystal aluminium with the crystal oriented in the (001) plane and cut in the [100] direction. Three positive rake cutting tools, namely 10°, 30° and 45°, are employed to investigate the effect of the rake angle on the forces, the specific energy and the nature of the chip formation. The method is evaluated by direct comparison with corresponding molecular dynamics simulations conducted under the same conditions.

NOMENCLATURE

a	thermal diffusivity ($J\,cm^2$)
a/r	acceptance-to-rejection ratio
A	area
AB	length of shear plane heat source (Å)
B_{chip}	fraction of the heat conducted into the chip
ΔB	maximum compensation for heat partition
D	equilibrium dissociation energy (eV)
F	force (nN)
F_c	cutting force (nN)
F_{fr}	frictional force (nN)
F_s	shear force (nN)
F_t	thrust force (nN)

† Author for correspondence. Email: ranga@ceat.okstate.edu.

Philosophical Magazine ISSN 1478-6435 print/ISSN 1478-6443 online © 2004 Taylor & Francis Ltd
http://www.tandf.co.uk/journals
DOI: 10.1080/14786430310001646736

J joule's mechanical equivalent of heat
K Boltzmann constant
$K_0(u)$ Bessel function
L length of tool–chip interface frictional heat source (Å)
m_a mass of an atom
N_{Pe} Peclet number
N_{Tool} number of atoms in the tool
N_{WP} number of atoms in the workpiece
$P(q_0)$ Boltzmann distribution function
q_{pl} heat liberation intensity of a moving-line heat source ($J\,cm^{-1}s^{-1}$)
q_{pli} heat liberation intensity of a moving-induced plane heat source ($J\,cm^{-1}s^{-1}$)
q_{pls} heat liberation intensity of a moving-plane heat source ($J\,cm^{-1}s^{-1}$)
q_0 configuration of N particles in three-dimensional space
r interparticle distance (Å)
r_e equilibrium distance (Å)
R distance between the moving-line heat source and the point M (Å)
t depth of cut (Å)
t_{ch} chip thickness (Å)
T Temperature (K)
v velocity of moving plane heat source ($m\,s^{-1}$)
v_c cutting velocity ($m\,s^{-1}$)
v_{ch} chip velocity ($m\,s^{-1}$)
v_i velocity of the ith atom ($m\,s^{-1}$)
V Morse potential (eV)
V_{min} minimum potential (eV)
ΔV change in Morse potential (eV)
w width of cut (Å)
X, z coordinates of the point at which the temperature rise occurs in the moving coordinate system (m)

Greek letters

α rake angle (deg)
θ temperature rise (°C)
θ_M temperature rise at any point (°C)
λ thermal conductivity ($J\,cm^{-1}\,s^{-1}\,°C^{-1}$)
ξ_{iN} uniform random number between 0 and 1 of the i^{th} Markov move on the N^{th} atom
ρ density ($g\,cm^{-3}$)
ϕ shear angle (deg)
φ oblique angle (deg)

§ 1. Introduction

Molecular dynamics (MD) simulation is a powerful technique for modelling the cutting process on a nanoscale (Komanduri and Raff 2001). However, there are three significant limitations associated with this process, namely: (i) the availability of accurate potential energy surfaces; (ii) the need to use high cutting speeds (100–500 $m\,s^{-1}$) and (iii) the consideration of a fewer number of workpiece atoms (a

few thousand). The latter two are used to limit the computational time to a reasonable value. These limitations have been of serious concern to most researchers working in this field for they would like to consider larger workpieces (a few million atoms or more) and conventional cutting speeds (2–5 $m\,s^{-1}$) to simulate nanometric cutting. Currently, this is not possible to accomplish with MD simulation even using the fastest state-of-the-art computers, including massive parallel processors.

One way to overcome the latter two of the above limitations is to use a Monte Carlo (MC) method, since time or cutting speed does not enter directly into this method (Komanduri *et al.* 2002). Also, some of the detailed time-consuming calculations conducted in MD simulations are of little practical interest. For example, detailed information concerning the precise movement of the atoms at all instants of time before reaching their equilibrium positions may not be of interest. If we employ a method that does not address such issues of no significance, valuable computational time, as well as memory space, can be saved.

In this investigation, nanometric cutting is simulated at conventional cutting speeds (5 $m\,s^{-1}$) using a new MC method (Narulkar 2002) that is applicable to systems that are neither canonical nor microcanonical. Even though cutting speed does not directly enter into the MC calculations, its effect can be introduced indirectly via the temperature generated in the cutting zone, since cutting speed is related to cutting temperature. The thermal analysis method developed by Komanduri and Hou (2000, 2001a, b) for conventional cutting is extended to the nanometric regime to permit local temperatures in the cutting region to be estimated. This temperature is then employed in the Boltzmann probability distribution function (PDF) that is used to determine the acceptance–rejection of MC moves in the simulation. The method allows MC simulations of nanometric cutting to be conducted at conventional cutting speeds (5 $m\,s^{-1}$) without burdening the system with significant computational time as in MD simulations. MC simulations of nanometric cutting were conducted at a conventional cutting speed (5 $m\,s^{-1}$) with three positive rake angle tools, namely 10°, 30° and 45° to investigate the effect of the rake angle on the forces, the specific energy and the nature of the chip formation process.

§ 2. Monte Carlo simulation of nanometric cutting

MC methods have been applied extensively in such diverse fields as engineering, materials science, mathematics, medicine, chemistry and physics (Rubinstein 1981). The basis for MC methods is that the configuration of the physical system is describable by a probability density function (Metropolis and Ulam 1949, von Neumann 1951). MC methods have proven to be an effective way of reducing the total computational time. The actual time reduction depends upon many factors, including the cutting speed, the number of atoms in the workpiece N_{wp}, and the type of potential used.

At high cutting speeds of about 500 $m\,s^{-1}$, MD will generally be the method of choice, for there is not much gained by using MC simulation. As the cutting speed decreases, MC methods will quickly become significantly more efficient. Figures 1 (a) and (b) show the variation in the computational time with the number of atoms and cutting velocity v_c respectively for MD and MC simulations in uniaxial tension and nanometric cutting, respectively. When the number of atoms considered is very small, the difference in computational time between the two methods is small (Figure 1 (a)). For example, Komanduri *et al.* (2002) have shown that a combined MC–damped trajectory (DT) method for nanometric uniaxial tensile testing of fcc

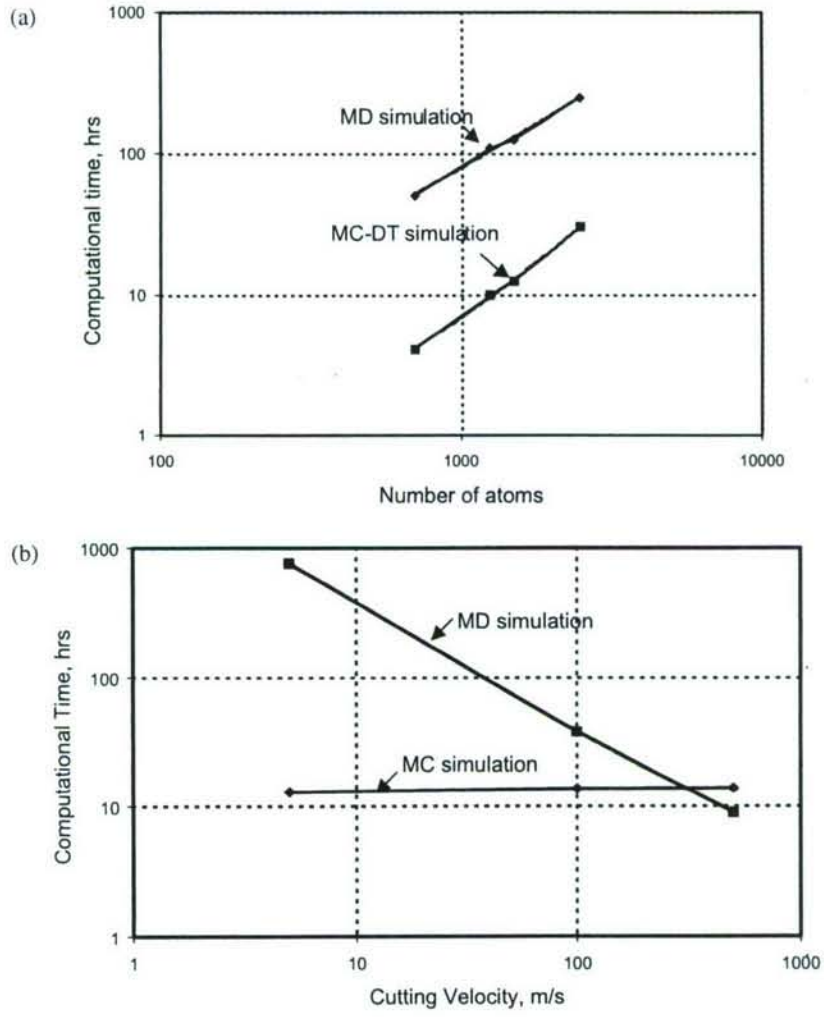


Figure 1. Variation of computational time with (a) the number of atoms in uniaxial tensile testing in MD and MC simulations and (b) the cutting velocity in MD and MC simulations.

metals takes about 16% of the time required for MD simulation for a system of 2500 atoms. This difference increases significantly as the number of atoms considered increases, for example to 10 000 atoms or higher. This trend would be the same in the case of nanometric cutting as well. This difference will be even more pronounced in nanometric cutting as the cutting speed decreases from about 500 m s^{-1} to practical cutting speed values (about 2–5 m s^{-1}) as shown in figure 1 (b). This is because, while there is not much difference between the computational times in the MC and MD simulations at a high cutting speed (500 m s^{-1}), with decrease in cutting speed (about 2–5 m s^{-1}), MD simulation times increase proportionately with decrease in cutting speed while the time required for MC simulation is nearly independent of cutting speed. Further, MC computations will generally become even more favourable as the complexity of the potential increases, for example from a simple pairwise empirical

potential to a complex potential, such as the embedded-atom potential. This is because MD calculations require the determination of the derivatives of the potential whereas MC calculations require only the potential itself.

Most MD simulations of nanometric cutting are conducted at very high cutting speeds (100–500 $m\,s^{-1}$) to reduce the computational time (Belak and Stowers 1990, Komanduri and Raff 2001). The major part of the computational time is associated with the numerical integration of the classical Newtonian equations of motion for the interacting particles to obtain new configurations in space over a certain period of time. If some of these calculations can be eliminated, a significant saving in the computational time should result.

In MC methods, the computational time can be reduced because the time variable is not directly involved. The temporal dependence of the positions of the atoms is not computed using Newtonian equations; instead, random Markov moves are used to obtain a series of equilibrium configurations. Figure 2 is a schematic diagram of the MC simulation of nanometric cutting (Narulkar 2003). The workpiece atoms are divided into moving atoms and boundary atoms. The purpose of boundary atoms is to simulate the bulk and hence these atoms are not allowed to move during the simulation. Consequently, interactions between boundary atoms are not calculated. All tool atoms are designated as boundary atoms; that is, the tool is considered to be infinitely hard. Also, the boundary atoms of the tool do not interact with the boundary atoms of the workpiece.

Unlike in MD simulations, it is not necessary to incorporate a peripheral zone in MC simulations. The purpose of the peripheral atoms in MD simulations is to dissipate the heat generated during the cutting process much like the heat carried away by the chip and the lubricant in addition to the cutting tool and the work material. This way, the effect upon the energy transfer within the solid that would be present for an extended lattice model is included in the calculations. In MC simulations, momentum and heat dissipation terms are not directly involved. Hence, there is no need for peripheral atoms in the simulation.

Since nature always drives any system towards its most stable configuration or minimum potential V_{min}, in MC simulations, we randomly select configurations that move the system towards its minimum potential. In most cases, the PDF is unknown so that this selection cannot be accomplished in a straightforward fashion. A common method to handle this type of problem involves the generation of

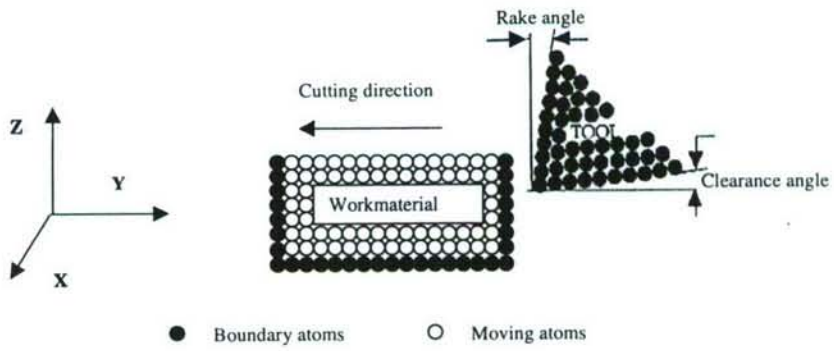


Figure 2. Schematic diagram of MC simulation of nanometric cutting showing various regions of interest (Narulkar 2003).

random configurations using Markov chains generated by the Metropolis sampling procedure (Metropolis and Ulam 1949, von Neumann 1951).

Consider an initial configuration q_0 for an N-particle system. The Boltzmann distribution function $P(q_0)$ for canonical systems is given by (for example Raff (2001))

$$P(q_0) = C \exp\left(-\frac{V(q_0)}{KT}\right), \tag{1}$$

where V and T are the potential and the temperature respectively for configuration q_0, K is the Boltzmann constant and C is a proportionality constant.

A subset of m particles from a total of N particles is now selected and the coordinates of these particles are randomly varied. Here, m is very small compared with N. For example, in MC simulations, m is typically one to five atoms in a system consisting of several thousand atoms. For the ith atom, the new configuration is obtained using the following equations (Herrmann 1994):

$$(x_i)_{\text{new}} = (x_i)_{\text{old}} + (0.5 - \xi_{i1})\Delta x, \tag{2}$$

$$(y_i)_{\text{new}} = (y_i)_{\text{old}} + (0.5 - \xi_{i2})\Delta y, \tag{3}$$

$$(z_i)_{\text{new}} = (z_i)_{\text{old}} + (0.5 - \xi_{i3})\Delta z, \tag{4}$$

where $(x_i)_{\text{new}}$, $(y_i)_{\text{new}}$ and $(z_i)_{\text{new}}$ are the new configurations in space and $(x_i)_{\text{old}}$, $(y_i)_{\text{old}}$ and $(z_i)_{\text{old}}$ are the old configurations for the ith atom. Δx, Δy and Δz are the maximum step sizes along the X, Y and Z directions respectively, and the ξ_i are randomly chosen numbers in the interval [0, 1].

For canonical systems, this new configuration point will be either accepted or rejected depending upon the change in the Boltzmann probability given by equation (1). The move is accepted, if

$$\exp\left(-\frac{\Delta V(q_0)}{KT}\right) \geqslant \xi_i, \tag{5}$$

where ξ_i is the random number generated between 0 and 1 in the ith Morkov move and $\Delta V = V_{\text{new}} - V_{\text{old}}$. However, since the system possesses large temperature gradients, it is not a canonical system and the application of equation (5) is not straightforward. This problem is addressed in § 3.

By choosing m and the maximum step size to be very large, it is possible to sample all regions of configuration space rapidly for all particles in the system. Consequently, the change ΔV in the potential will usually be very large and $\exp[-V(q_0)/KT(q_0)]$ very small. Consequently, the probability of acceptance of the given Metropolis move will be very low. Thus, a great deal of computational time is wasted because of the large number of rejected moves. On the other hand, if we take the maximum step size to be very small, ΔV will be very small and the probability of acceptance will be very high. However, it will take many accepted moves to sample the configuration space adequately and to reach convergence. These two extremes are, therefore, not beneficial. The values of Δx, Δy and Δz should be chosen such that the acceptance-to-rejection ratio a/r is in the range $0.7 \leqslant a/r \leqslant 1.3$. In this investigation, m is taken as 1 and the value of maximum step size in all directions as 0.2 Å.

Consideration of equation (1) shows that all moves leading to lower potential are accepted. In other words, if $\Delta V < 0$, the move is accepted. Thus, the system eventually moves to the minimum potential. On the other hand, if the change in the potential is positive, that is the potential rises, then the value of $\exp[-\Delta V(q_0)/KT(q_0)]$

must be greater than the random number ξ_i generated for the particular Markov move to be accepted. The move is accepted with a probability that decays exponentially with increasing ΔV. If the move is rejected, the old position of the atom is retained. This cycle is considered as one Markov move. The Markov moves are repeated until a global minimum is reached.

Two methods are generally employed in selecting a particular subset of m particles to be moved in a given set. In the first method, the m particles in the subset are selected randomly in each move from the N particles in the system. In the second method, some systematic method of selection is used that ensures that all particles are moved before any particle is moved again. For an infinite number of Markov moves, both methods become equivalent but, for a finite number, one method may produce more rapid convergence than the other. In the present investigation, one atom is chosen randomly at any time and is given a Markov move. This process is repeated until the local potential is conserved.

In cutting, a considerable amount of plastic deformation takes place in the shear zone ahead of the cutting tool. So, the atoms in the workpiece at or near the tool in the depth-of-cut region are deformed much more than the atoms further down in the workpiece, which remain in configurations similar to the bulk material. In other words, the change in potential of the atoms at or near the tool tip in the workpiece is more than for the atoms further down in the workpiece. The workpiece atoms at the depth-of-cut region are removed in the form of a chip from the bulk and are moved to an elevated potential. Hence, a larger number of Markov moves are required to reach convergence for the atoms in the cutting region.

A MC step consists of two parts. In the first part, an atom is chosen from the atoms from the cutting region (zone 1), as shown in figure 3. A Markov move is then executed on this atom. This process is repeated until the local potential converges to a minimum. In the second step, an atom is chosen randomly from the workpiece (zone 2), including those interacting with the tool and a Markov move is executed. This entire process is continued until the overall system potential is minimized. At this point, the tool is advanced by a small increment and the entire process is repeated at the new tool position.

The empirical potential used for the MC simulation is a pairwise Morse potential as it is found to be simple and more suitable for fcc materials (Komanduri *et al.* 2001). It can be represented as (Girifalco and Weizer 1959)

$$V(r) = D\{1 - \exp[-\alpha(r - r_e)]\}^2$$

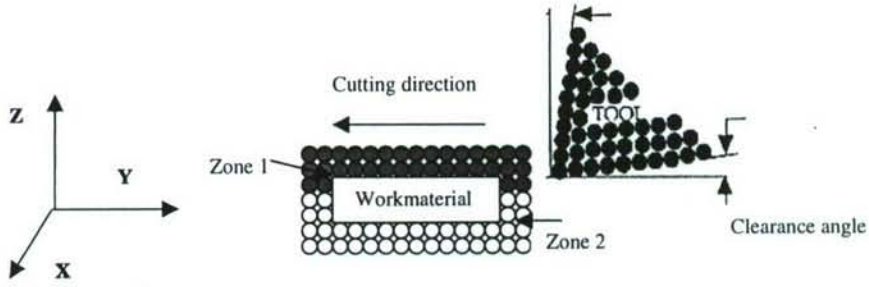


Figure 3. Schematic diagram showing two zones in the MC simulation of nanometric cutting.

or

$$V(r) = \begin{cases} D\{\exp[-2\alpha(r - r_e)] - 2\exp[-\alpha(r - r_e)]\} & \text{for } r \leqslant R^*, \\ 0 & \text{for } r > R^*, \end{cases} \tag{6}$$

where r is the distance between any two atoms and r_e is the equilibrium distance, D is the equilibrium dissociation energy, R^* is the cut-off radius and α denotes the range of interactions. The three parameters r_e, D and α are obtained from the closest spacing between the atoms (equilibrium lattice spacing), the Debye temperature and the sublimation energy, respectively. The cut-off radius is adjusted so that $V_{(R=R^*)}/V_{(r=r_e)} = 0.04$. The accuracy of the function as well as the stability of the crystal for a given material are checked for various properties, including the cohesive energy, the lattice constant, the compressibility and the elastic constants as well as the equation of state and the stability of the crystal. The potential of the entire tool–workpiece system is the pairwise summation of potential terms between the lattice atoms of the workpiece and a second summation of pairwise Morse potentials between the atoms of the tool and the atoms of the workpiece. It should be noted that, in this investigation, we are not primarily concerned with reproducing the experimental potential. The focus is on the development of MC methods for machining applications at practical cutting speeds.

§3. Application of the thermal model of metal cutting to nanometric cutting

In metal cutting with a sharp tool, there are two principal heat sources, namely the shear plane heat source at the primary shear zone and the frictional heat source at the tool–chip interface. In this investigation, the thermal model of conventional metal cutting developed by Komanduri and Hou (2000, 2001a, b) on a macroscale is adopted to estimate the temperature in the cutting region in nanometric cutting.

Komanduri and Hou (2000) first considered the temperature distribution in the chip and the work material caused by the shear plane heat source in machining using Hahn's (1951) model of a band heat source moving obliquely in an infinite medium. Komanduri and Hou (2001a) then considered the frictional heat source at the chip–tool interface, based on the work of Chao and Trigger (1955) that considers non-uniform distribution of the heat partition fraction along the tool–chip interface for the purpose of matching the temperature distribution on the chip side and the tool side. Komanduri and Hou (2001a) used the functional analysis approach, first proposed by Chao and Trigger (1955), to address the variable heat partition problem. Komanduri and Hou (2001b) then combined the effect of shear plane heat source and tool–chip interface frictional heat source to determine the temperature distribution in the workpiece, the chip and the tool. As this model is incorporated in the present investigation, it will be briefly presented in the following.

3.1. *Combined effect of shear plane heat source and tool–chip interface frictional heat source*

Figure 4 is a schematic diagram of the heat transfer model showing the two principal heat sources, namely the shear plane heat source AB and the chip–tool interface frictional heat source OA operating simultaneously on a common coordinate system. By considering that part of the heat flow from the shear plane through the chip and the tool–chip interface into the tool to be continuous, Komanduri and Hou (2001b) found the effects of the shear plane heat source on the temperature rise

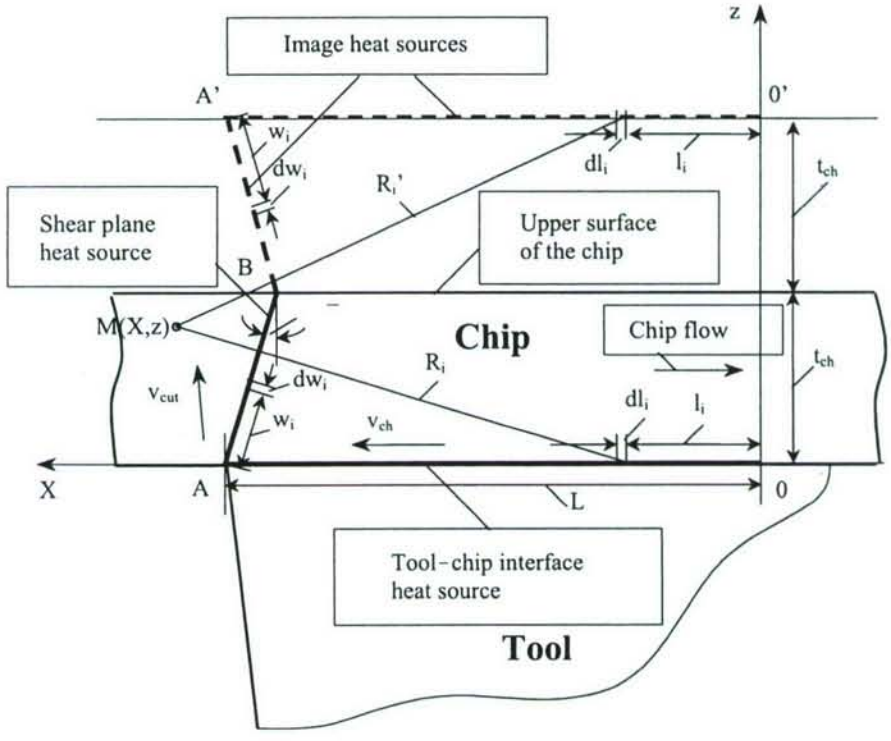


Figure 4. Schematic diagram of the heat transfer model with a common coordinate system for the combined effect of two principal heat sources, namely the heat source in the primary shear plane and the frictional heat source at the chip–tool interface (Komanduri and Hou 2001b).

at the tool–chip interface on the chip side as well as on the tool side to be the same. Thus, the contribution of the shear plane heat source on the temperature rise at the tool–chip interface was included in the total temperature rise for both sides.

Referring to figure 4, both the shear plane heat source and the tool–chip interface frictional heat source move relative to the chip at the chip velocity v_{ch} but in the opposite direction to the chip flow. As the entire shear plane heat source is under the upper boundary surface of the chip except for point B (which is at the boundary surface all the time), an image heat source A′B was considered. Komanduri and Hou (2001b) used an oblique moving band heat source for an infinite medium for both the primary shear plane heat source and its image heat source. Similarly, the total temperature rise at any point in the tool caused by the two principal heat sources also consists of two parts. The first is the frictional heat source at the tool–chip interface from the tool side and the second is due to the shear plane heat source. However, a different heat transfer model is needed to address this problem. They considered that part of the heat coming from the shear plane heat source through the chip and the tool–chip interface into the tool, acting as a stationary heat source located at the tool chip interface. Thus, the heat source in this case is considered as an induced stationary rectangular heat source caused by the shear plane heat source. The analysis of thermal modelling developed by Komanduri and Hou (2000, 2001a, b) is extensive and

somewhat involved. Hence, here only the final results of that investigation are given. For details, the reader is referred to those references.

The temperature rise due to frictional heat source at the tool–chip interface is given by (Komanduri and Hou 2001b)

$$\theta_{\mathrm{M}} = \frac{q_{\mathrm{pl}}}{\pi\lambda}\left\{(B_{\mathrm{chip}} - \Delta B)\int_{l_i=0}^{L} \exp\left(\frac{-(X-l_i)v}{2a}\right)\left[K_0\left(\frac{R_i v}{2a}\right)+K_0\left(\frac{R_i' v}{2a}\right)\right]\mathrm{d}l_i \right.$$
$$+ 2\,\Delta B\int_{l_i=0}^{L}\left(\frac{l_i}{L}\right)^m \exp\left(\frac{-(X-l_i)v}{2a}\right)\left[K_0\left(\frac{R_i v}{2a}\right)+K_0\left(\frac{R_i' v}{2a}\right)\right]\mathrm{d}l_i$$
$$\left. + C\Delta B\int_{l_i=0}^{L}\left(\frac{l_i}{L}\right)^k \exp\left(\frac{-(X-l_i)v}{2a}\right)\left[K_0\left(\frac{R_i v}{2a}\right)+K_0\left(\frac{R_i' v}{2a}\right)\right]\mathrm{d}l_i\right\}. \quad (7)$$

The temperature rise at any point in the chip, including all points along the tool–chip interface, caused by the shear plane heat source is given by (Komanduri and Hou 2001b)

$$\theta_{\mathrm{M}} = \frac{q_{\mathrm{pls}}}{\pi\lambda}\int_{l_i=0}^{L}\exp\left(\frac{-(X-l_i)v}{2a}\right)\left[K_0\left(\frac{v}{2a}\left[(X-X_i)^2+(z-z_i)^2\right]^{1/2}\right)\right]$$
$$+ K_0\left(\frac{v}{2a}\left[(X-X_i)^2+(2t_{\mathrm{ch}}-z-z_i^2)\right]^{1/2}\right)\mathrm{d}w_i. \quad (8)$$

The total rise in the temperature at any point in the workpiece is given by Komanduri and Hou (2001b),

$$\theta_{\mathrm{M}} = \frac{q_{\mathrm{pl}}}{\pi\lambda}\left\{(B_{\mathrm{chip}} - \Delta B)\int_{l_i=0}^{L}\exp\left(\frac{-(X-l_i)v}{2a}\right)\left[K_0\left(\frac{R_i v}{2a}\right)+K_0\left(\frac{R_i' v}{2a}\right)\right]\mathrm{d}l_i\right.$$
$$+ 2\Delta B\int_{l_i=0}^{L}\left(\frac{l_i}{L}\right)^m\exp\left(\frac{-(X-l_i)v}{2a}\right)\left[K_0\left(\frac{R_i v}{2a}\right)+K_0\left(\frac{R_i' v}{2a}\right)\right]\mathrm{d}l_i$$
$$\left. + C\Delta B\int_{l_i=0}^{L}\left(\frac{l_i}{L}\right)^k\exp\left(\frac{-(X-l_i)v}{2a}\right)\left[K_0\left(\frac{R_i v}{2a}\right)+K_0\left(\frac{R_i' v}{2a}\right)\right]\mathrm{d}l_i\right\}$$
$$+ \frac{q_{\mathrm{pls}}}{\pi\lambda}\int_{w_i=0}^{t_{\mathrm{ch}}/\cos(\phi-\alpha)}\exp\left(\frac{-(X-X_i)v}{2a}\right)\left[K_0\left(\frac{v}{2a}\left[(X-X_i)^2+(z-z_i)^2\right]^{1/2}\right)\right.$$
$$\left. + K_0\left(\frac{v}{2a}\left[(X-X_i)^2+(2t-z-z_i)^2\right]^{1/2}\right)\right]\mathrm{d}w_i. \quad (9)$$

Komanduri and Hou (2001b) applied the thermal model to two cases of metal cutting at the macroscopic level, namely conventional machining of steel with a carbide tool at a high Peclet number ($N_{\mathrm{Pe}} \approx 5$–$20$) (Chao and Trigger 1953) and ultra-precision machining of aluminium with a single-crystal diamond tool at a low Peclet number ($N_{\mathrm{Pe}} \approx 0.5$) (Ueda *et al.* 1998) and found good agreement between the analytical and the experimental results.

3.2. *Application of the thermal model of metal cutting to Monte Carlo simulation of nanometric cutting*

In order to apply MC methods to the non-canonical systems present in machining applications, we assume that a local temperature can be meaningfully defined for ensembles of about 100 atoms. This assumption permits us to replace the temperature T in equation (5) with one that is dependent upon the spatial coordinates of the

atoms being moved in each Markov step. That is, we replace T in equation (5) with a spatially dependent temperature $T(q_0)$ that is computed using equation (9).

As pointed out, the thermal model developed by Komanduri and Hou (2001b) was verified with the experimental results reported in the literature at the macroscopic level. In this investigation, this model is applied for the first time to nanometric cutting. To check the viability of this model, MD simulation of nanometric cutting of single-crystal aluminium was conducted with 10° and 30° rake tools at a cutting speed of $500\,\mathrm{m\,s^{-1}}$ using the pairwise potential previously described and the results compared with the thermal analysis of the metal-cutting process on a nanoscale.

For estimating the temperature in the MD simulation, a small region ahead of the tool in the cutting zone (figure 5) was selected and observed continuously during the entire simulation. The width of this region is the same as the width of the workpiece. The temperature in this region was calculated at each tool movement by assuming an equipartitioning of the total energy between kinetic and potential energy as would be present for an ensemble of independent harmonic oscillations (figure 6). This assumption leads to a temperature T in the cutting region equal to

$$T = \sum_{i=0}^{P} \frac{1}{3} \frac{m_a v_i^2}{K}, \tag{10}$$

where P is the total number of atoms falling in the region considered, m_a is the mass of the atom and v_i is the velocity of the ith atom in the same region.

The size of the region is chosen so as to ensure that there is a sufficient number of atoms in the region to obtain a good statistical average value of the temperature. Usually, some 50–100 atoms are taken for P.

The cutting and thrust forces are obtained from the MD simulation of nanometric cutting. Figures 7 (a) and (b) show the variations in cutting and thrust forces, respectively, with cutting time for nanometric cutting with a 10° rake angle tool and figures 8 (a) and (b) show the corresponding variations for a 30° rake angle tool. These forces are then resolved along the shear plane and the tool face using Merchant's (1945) force circle diagram (see figure 9 for details) to determine the

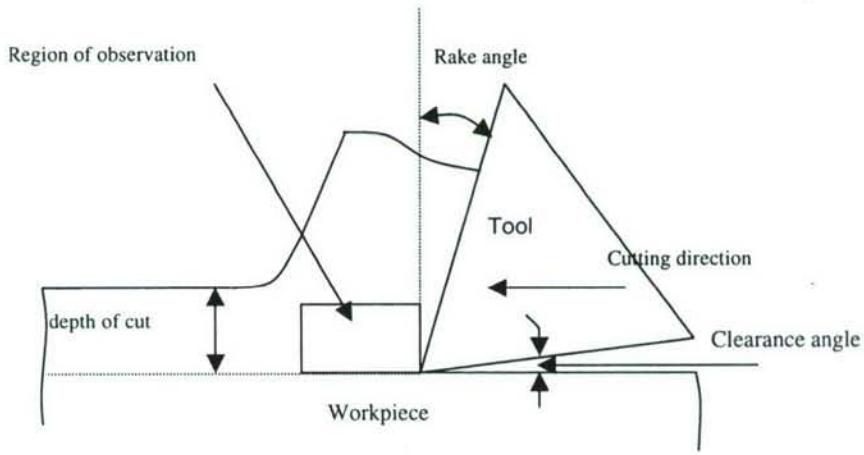


Figure 5. Schematic diagram showing the region of observation in the cutting zone for temperature estimation in nanometric cutting.

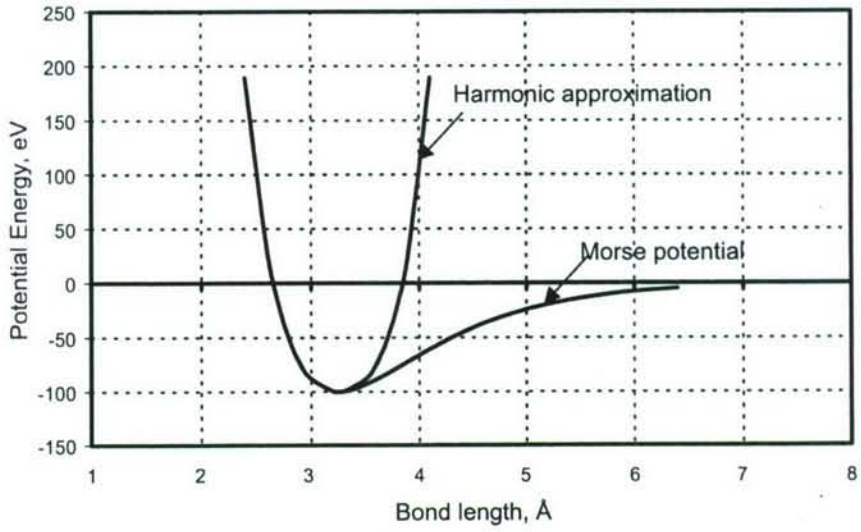


Figure 6. Comparison of the Morse potential and harmonic oscillation.

intensity of the heat sources at the shear plane and the tool–chip frictional interface. The resolved forces are then used to calculate the heat generated at the shear plane and at the tool–chip interface, which are the quantities required by equations (9) and (13).

The heat generated at the chip–tool interface is given by

$$q_{\mathrm{pl}} = \frac{F_{\mathrm{fr}} v_{\mathrm{ch}}}{Lw} \ \mathrm{J\,cm^{-1}\,s^{-1}}. \tag{11}$$

Similarly, the heat generated at the shear plane is given by

$$q_{\mathrm{pls}} = \frac{(F_{\mathrm{c}} v_{\mathrm{c}} - F_{\mathrm{fr}} v_{\mathrm{ch}})}{\mathrm{AB} \times w} \ \mathrm{J\,cm^{-1}\,s^{-1}} \tag{12}$$

where F_{fr} is the frictional force acting along the chip–tool surface, F_{c} is the cutting force, L is the length of frictional heat source, v_{ch} is the chip velocity, w is the width of cut and AB is the length of the shear heat source ($t/\sin\phi$).

In the MD simulation, the tool is considered infinitely hard. So, the atoms do not vibrate at their respective equilibrium positions; that is, there is no molecular kinetic energy in the tool atoms, which is reflected in the fact that the temperature rise in the tool is zero throughout the entire simulation. In other words, when we apply the thermal model, the tool does not share the heat generated at the chip–tool interface and as a first approximation, the chip carries away all the heat. Therefore, equation (9) is simplified by setting $\Delta B = 0$ and $B_{\mathrm{chip}} = 1.0$. Thus,

$$\begin{aligned}\theta_{\mathrm{M}} = {} & \frac{q_{\mathrm{pl}}}{\pi\lambda}\left\{\int_{l_i=0}^{L} \exp\left(\frac{-(X-l_i)v}{2a}\right)\left[K_0\left(\frac{R_i v}{2a}\right) + K_0\left(\frac{R_i' v}{2a}\right)\right] \mathrm{d}l_i\right\} \\ & + \frac{q_{\mathrm{pls}}}{\pi\lambda}\int_{w_i=0}^{t_{\mathrm{ch}}/\cos(\phi-\alpha)} \exp\left(\frac{-(X-X_i)v}{2a}\right)\left[K_0\left(\frac{v}{2a}\left[(X-X_i)^2 + (z-z_i)^2\right]^{1/2}\right)\right. \\ & \left. + K_0\left(\frac{v}{2a}\left[(X-X_i)^2 + (2t_{\mathrm{ch}} - z - z_i)^2\right]^{1/2}\right)\right] \mathrm{d}w_i. \end{aligned} \tag{13}$$

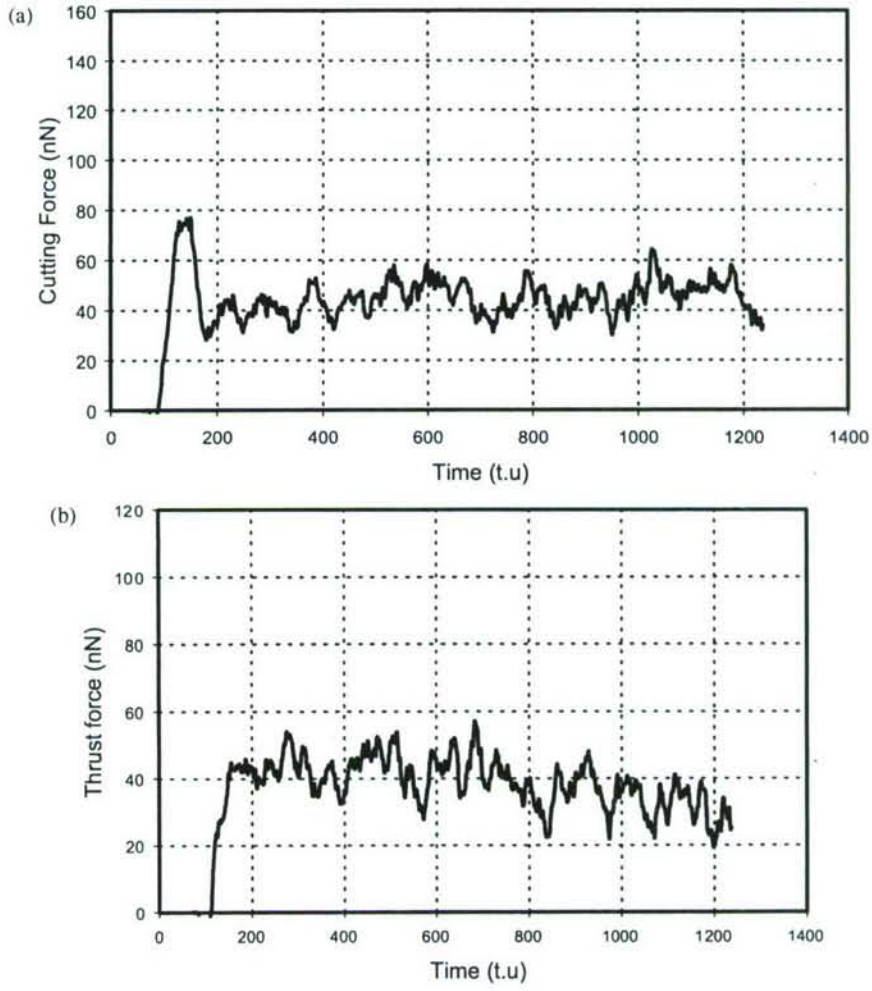


Figure 7. Variations in (a) cutting and (b) thrust forces (moving-average values) with cutting time in MD simulation of nanometric cutting of a single-crystal aluminium workpiece (rake angle, 10°; cutting velocity, 500 m s^{-1}; depth of cut, 5.1 Å; width of cut, 43 Å). In this and subsequent figures, t.u indicates time units.

§4. Results and discussion

The main objective of this investigation is to conduct comparison studies of MC simulations of nanometric cutting of a single-crystal aluminium work material oriented in the (001) plane and cut in the [100] direction with three positive rake angle tools, namely 10°, 30° and 45°, at a cutting speed of 5 m s^{-1} and to determine the effect of rake angle on the cutting and thrust forces, force ratio, frictional force on the tool face, shear force on the shear plane and specific energy, as well as the subsurface deformation in the work material.

4.1. *Comparison of the temperature in the cutting zone by thermal analysis and molecular dynamics simulation of nanometric cutting at 500 m s^{-1}*

The temperature in the cutting region was calculated using equation (10) and compared with the results obtained using equation (13). Figures 10 and 11 show the

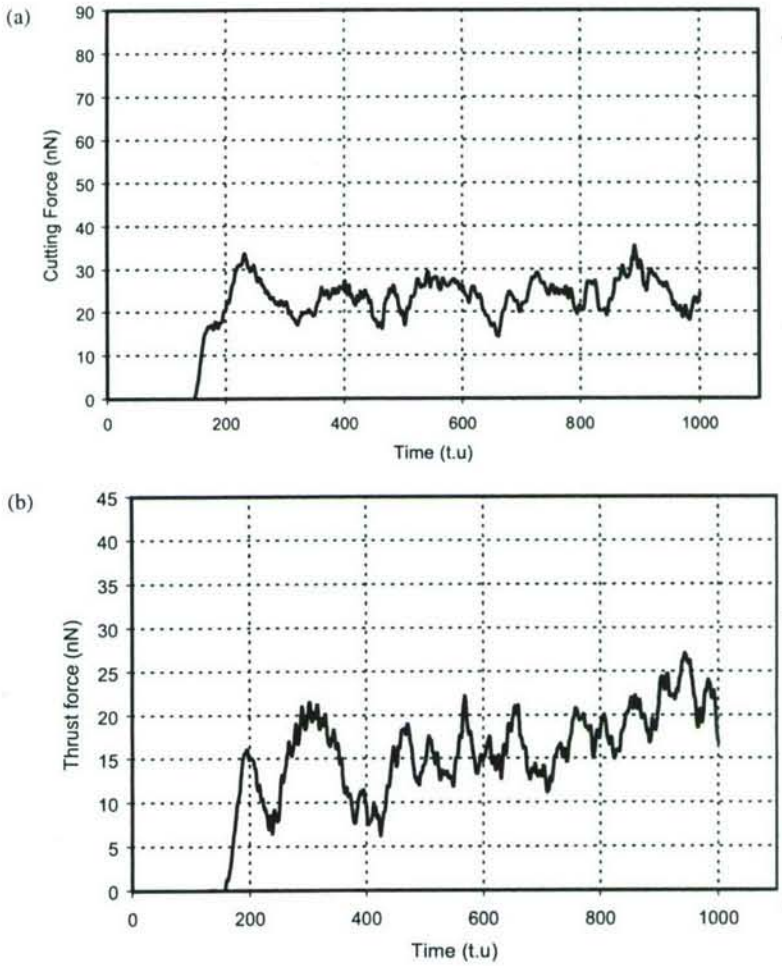


Figure 8. Variations in (a) cutting and (b) thrust forces (moving-average values) with cutting time in MD simulation of nanometric cutting of a single-crystal aluminium workpiece (rake angle, 30°; cutting velocity, 500 m s^{-1}; depth of cut, 5.1 Å; width of cut, 43 Å).

variation in the temperature in the cutting zone with cutting time from the MD simulation and the thermal model for cutting at 500 m s^{-1} with 10° and 30° rake angle tools, respectively. It can be seen that the agreement is very good. Once the local temperatures in the cutting region are known, they can be substituted into equation (5) and used in MC calculations. This method permits the effects due to cutting speed to be indirectly introduced into MC simulations of nanometric cutting. This procedure, therefore, permits MC simulation of nanometric cutting to be conducted at conventional cutting speeds of about 5 m s^{-1}. A similar simulation by the MD method would take an inordinate amount of time, even with the fastest computers available.

4.2. *Force analysis for molecular dynamics and Monte Carlo simulations of cutting*

In a cutting experiment, the X, Y and Z components of the cutting force are measured at the tool edge. These components are then decomposed into cutting

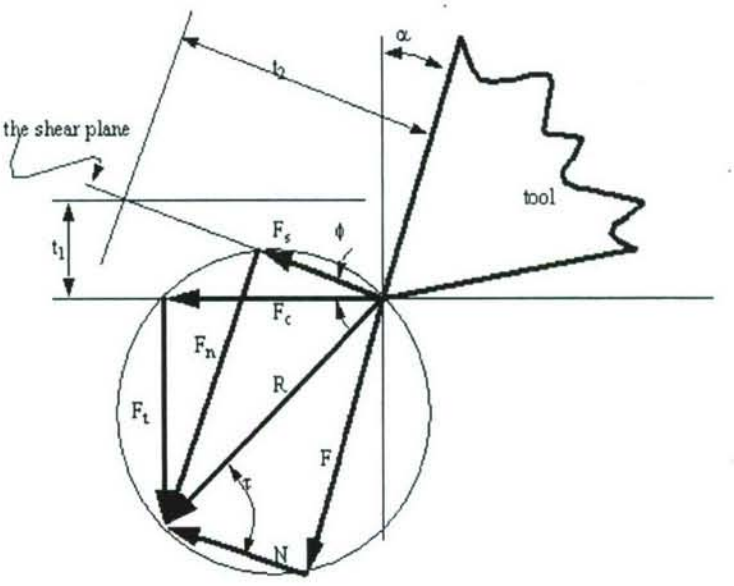


Figure 9. Merchant's (1945) force circle diagram for orthogonal cutting.

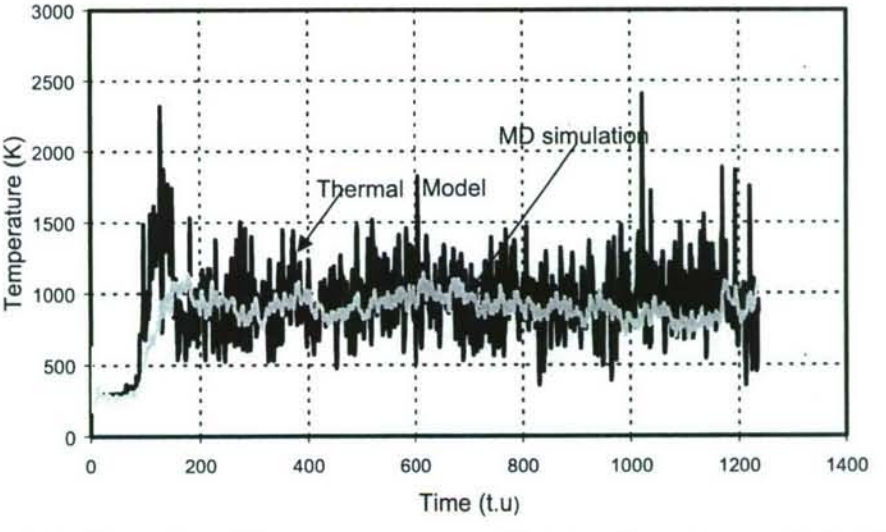


Figure 10. Comparison of the temperature generated in the cutting region near the tool tip, between the thermal model after Komanduri and Hou (2001b) and the MD simulation results (rake angle, 10°; cutting velocity, 500 m s^{-1}).

force, thrust force, shear force and frictional forces, as shown in figure 9. In this section, we examine the relationship between these forces and those normally computed in MD and MC simulations.

The total force F^{total} exerted by the cutting tool, is the sum of the static forces that are present owing to the interactions between the workpiece atoms and the workpiece–tool atoms plus the inertial terms for each atom due to acceleration effects. The static forces are given by the sum of the negative gradients of the

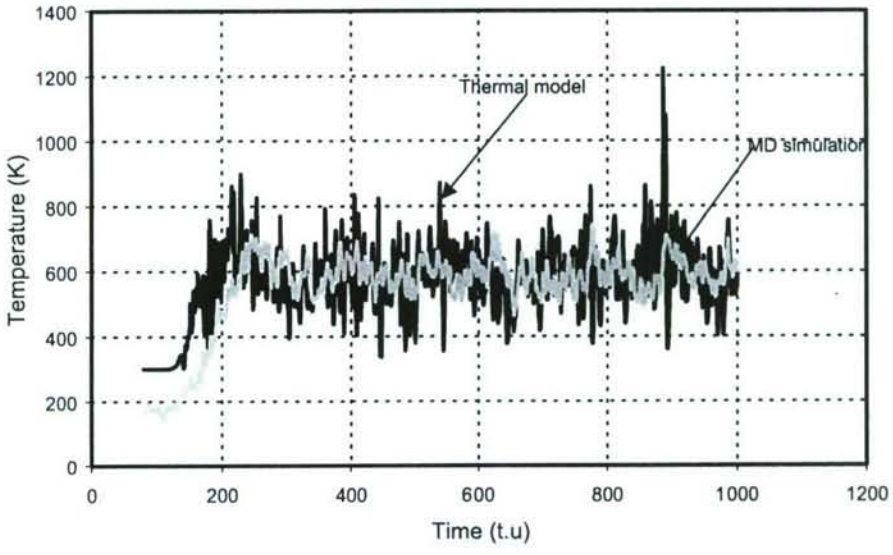


Figure 11. Comparison of the temperature generated in the cutting region near the tool tip, between the thermal model after Komanduri and Hou (2001b) and the MD simulation results (cutting velocity, $500\,\mathrm{m\,s^{-1}}$; rake angle, 30°).

potential in the direction of the force while the inertial forces are given by the sum of the time derivatives of the appropriate momentum components of each of the atoms. Thus, the total cutting force is given by

$$F_Y^{\text{total}} = -\sum_{i=1}^{N}\sum_{j=i+1}^{N}\frac{\partial V}{\partial Y_{ij}} - \sum_{i=1}^{N}\sum_{k=1}^{N_{\text{tool}}}\frac{\partial V}{\partial Y_{ik}} + \sum_{i=1}^{N}\frac{\mathrm{d}P_{Yi}}{\mathrm{d}t}, \tag{14}$$

since the cutting direction is taken to be the Y axis (see figures 2 and 3). The first double summation in equation (14) is the sum of all static forces between the N workpiece atoms. The second double summation incorporates the static forces between the N workpiece atoms and the N_{tool} atoms in the cutting tool. The final summation includes all the Y components of the inertial forces on the workpiece atoms. Since the tool atoms are fixed in position, these terms are absent for the tool. Consequently, the static forces obtained in the MD simulations are given by

$$\left(-\sum_{i=1}^{N}\sum_{j=i+1}^{N}\frac{\partial V}{\partial Y_{ij}} - \sum_{i=1}^{N}\sum_{k=1}^{N_{\text{tool}}}\frac{\partial V}{\partial Y_{ik}}\right)_{\text{MD}} = F_Y^{\text{total}} - \sum_{i=1}^{N}\frac{\mathrm{d}P_{Yi}}{\mathrm{d}t}. \tag{15}$$

In the case of MC calculations, the method converges the atoms to the metastable equilibrium positions that characterize the system at each position of the tool during the cutting process. In these equilibrium positions, the sum of the inertial terms will be zero. Therefore, the static forces obtained in the MC calculations will be given by

$$\left(-\sum_{i=1}^{N}\sum_{j=i+1}^{N}\frac{\partial V}{\partial Y_{ij}} - \sum_{i=1}^{N}\sum_{k=1}^{N_{\text{tool}}}\frac{\partial V}{\partial Y_{ik}}\right)_{\text{MC}} = F_Y^{\text{total}}. \tag{16}$$

Since most of the inertial terms will have the same sign as F_Y^{total}, we expect to have

$$F_Y^{\text{total}} - \sum_{i=1}^{N} \frac{dP_{Yi}}{dt} < F_Y^{\text{total}}. \tag{17}$$

Therefore, the static forces obtained in MC studies will exceed those for MD simulations under corresponding conditions. As the cutting speed decreases, the sum of the inertial forces will also decrease. Consequently, at low cutting speeds, we expect to see the forces computed in MD simulations approach those obtained in the corresponding MC calculations.

4.3. *Comparison between molecular dynamics and Monte Carlo simulations of nanometric cutting at 500 m s^{-1}*

4.3.1. *Molecular dynamics simulation of nanometric cutting*

MD simulations of nanometric cutting of single-crystal aluminium work material were conducted in the (100) plane with cutting in the [001] direction at a cutting velocity of 500 m s^{-1} with a 10° rake tool. Table 1 gives the workpiece and tool dimensions as well as the Morse potential parameters used in the simulation.

Figures 12 (a)–(d) show various stages of MD simulation of nanometric cutting at 500 m s^{-1}. In the initial stages of cutting, very little deformation ahead of the tool is observed. As cutting progresses, plastic deformation in the shear zone can be seen. As the tool proceeds further, dislocation generation and propagation in the workpiece as well as subsurface deformation can be seen (figure 12 (d)).

Figures 7 (a) and (b) show the variation in the static cutting and thrust forces with cutting time for a 10° rake tool at 500 m s^{-1} in MD simulations of nanometric

Table 1. Workpiece and tool dimensions, cutting parameters and Morse potential parameters used in MD and MC simulation of nanometric cutting of single-crystal aluminium at a cutting velocity of 500 m s^{-1}.

Work material	Single-crystal aluminium
Crystal structure	Fcc
Morse potential parameters for aluminium	
α	1.16460 Å^{-1}
r_e	3.253 Å
D	0.2703 eV
Configuration	Two-dimensional or orthogonal cutting
Work material dimension	$11a$, $50a$ and $11a$ along the X, Y and Z directions respectively, where a is the lattice constant of the workpiece
Tool material	Infinitely hard (tungsten)
Tool dimensions	$22a$, $10a$ and $10a$ along the X, Y and Z directions respectively, where a is the lattice constant of the tool
Depth of cut	4.05–5.1 Å

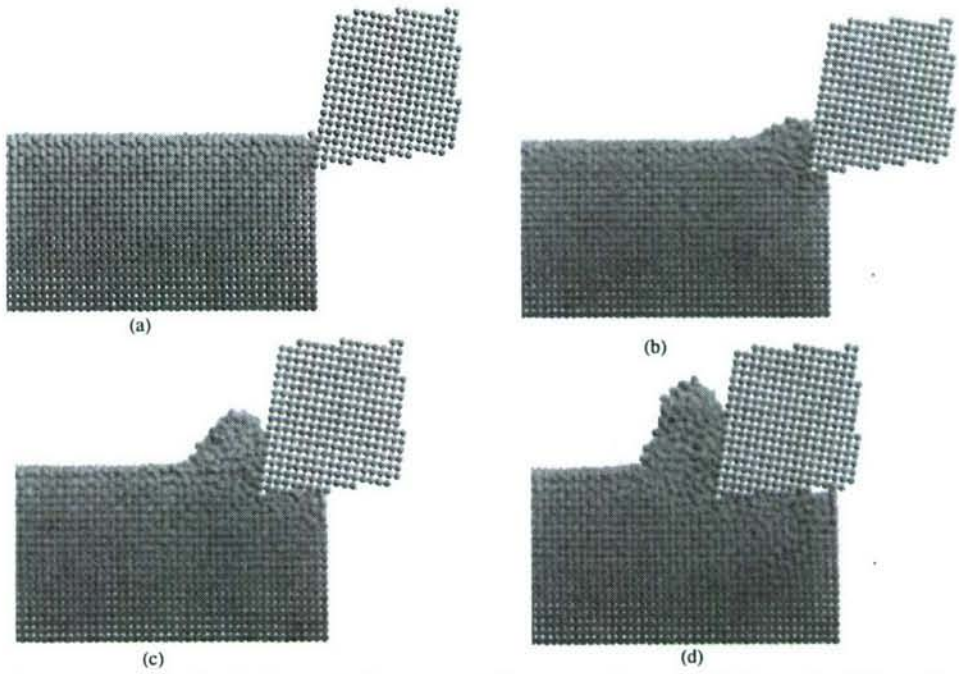


Figure 12. MD simulation at various stages of nanometric cutting (rake angle, 10°; cutting velocity, 500 m s^{-1}).

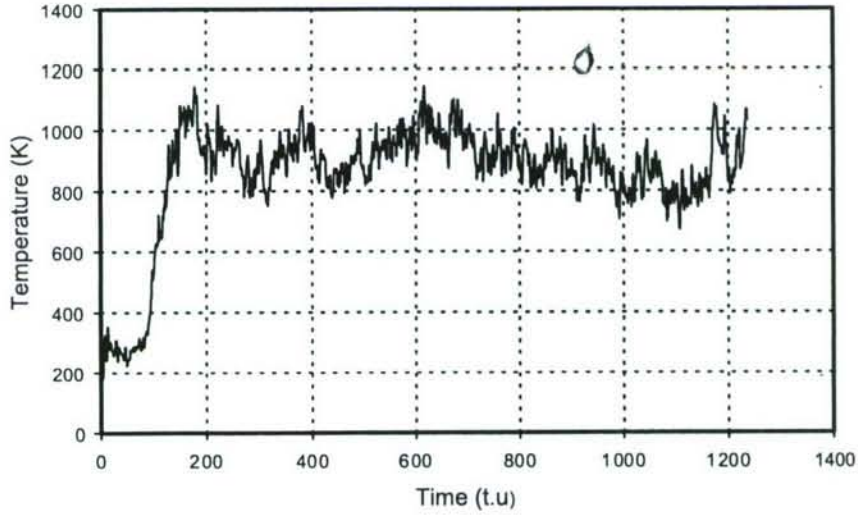


Figure 13. Variation in the temperature in the cutting zone in MD simulation of nanometric cutting of aluminium (rake angle, 10°; cutting velocity, 500 m s^{-1}).

cutting. The temperature in the cutting zone is calculated using equation (10) as described in § 3.2. Figure 13 shows the variation in the temperature in the cutting region with cutting time (same plot as in figure 10). From this, the mean temperature is estimated to be about 900 K at 500 m s^{-1} with a 10° rake tool.

4.3.2. *Monte Carlo simulation of nanometric cutting at 500 m s^{-1}*

MC simulations of nanometric cutting were conducted on a single-crystal aluminium work material at the same cutting speed of 500 m s^{-1} with a 10° rake tool. Figures 14 (a)–(d) show the results of the MC simulation at various stages of nanometric cutting. On comparing figures 12 and 14, it can be seen that the nature of deformation ahead of the tool as well as the subsurface deformation, in both MC and MD simulations are very similar. Figures 15 (a) and (b) show the variations in the static cutting and thrust forces respectively with cutting time in MC simulation with a 10° rake tool at a cutting speed of 500 m s^{-1}. After each tool advancement, the temperature in the cutting region was calculated using the thermal model of Komanduri and Hou (2001b). Figure 16 shows the variation in temperature in the cutting zone with cutting time in MC simulation. A mean temperature of about 1000 K was obtained during the cutting process. This is in good accord with the MD simulation results shown in figure 13.

Table 2 gives a comparison between MD and MC simulation of nanometric cutting of aluminium work material at the same cutting speed of 500 m s^{-1} with a 10° rake tool. Comparison of figures 7 (a) and (b) with figures 15 (a) and (b) shows that, as expected, the static cutting and thrust forces are higher for the MC simulation than those obtained in the MD calculations. The MC static thrust force is about 35% higher than the MD result. The static cutting force obtained from the MC calculation exceeds the MD result by amounts varying up to 120% at various times during the simulation. These differences are attributed here to

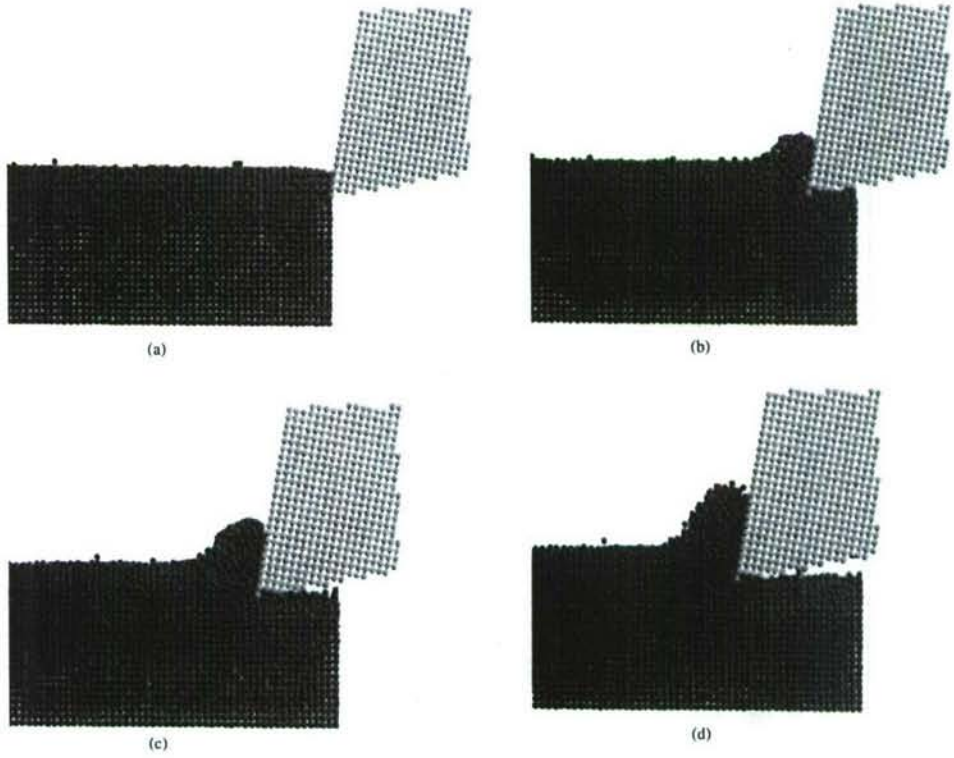


Figure 14. MC simulation at various stages of nanometric cutting of aluminium (rake angle, 10°; cutting velocity, 500 m s^{-1}).

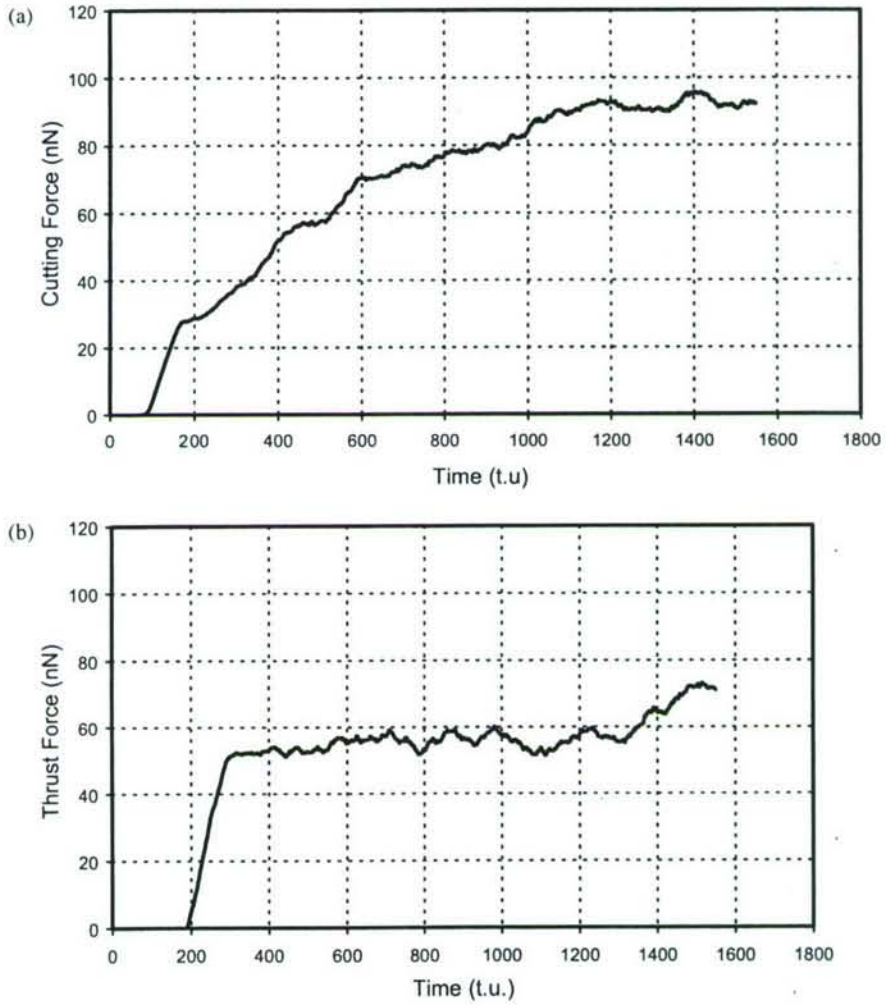


Figure 15. Variation in (a) cutting and (b) thrust forces with cutting time in MC simulation of aluminium (rake angle, 10°; cutting velocity, 500 m s^{-1}).

the presence of a net inertial force term in the MD calculations that is absent for MC simulations (see equations (14)–(17)) and to the larger cut depth employed in the MC calculations. The somewhat large magnitude of these differences is due primarily to the large cutting speed of 500 m s^{-1}, which produces a very large net inertial term. At experimental cutting speeds of about 2 m s^{-1} with equal depths of cut, the differences are expected to be much smaller. The larger static forces present in the MC calculations will also lead to larger specific energies than those obtained using MD methods.

4.4. *Monte Carlo simulation of nanometric cutting at a cutting speed of 5 m s^{-1}*

MC simulations of nanometric cutting were conducted on a single-crystal aluminium work material oriented in the (001) plane and cut in the [100] direction using three positive rake angle tools, namely 10°, 30° and 45°. Table 3 gives the computational parameters used in the MC simulation of nanometric cutting at

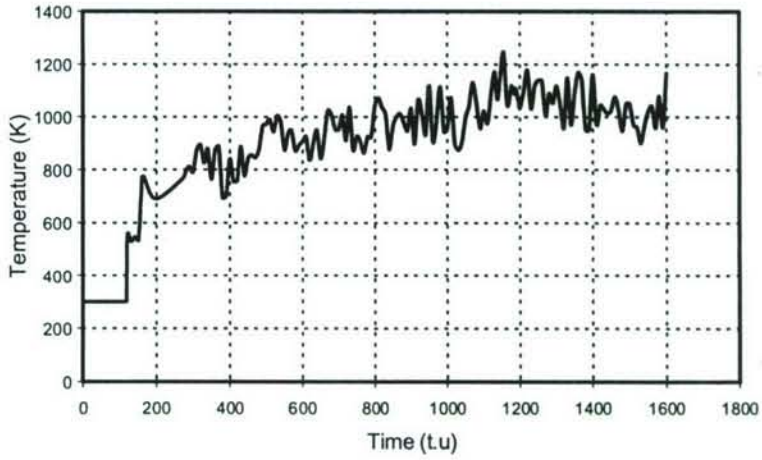


Figure 16. Variation in temperature in the cutting zone with cutting time in MC simulation of nanometric cutting (rake angle, 10°; cutting velocity, 500 m s^{-1}).

Table 2. Comparison between MD and MC simulation for nanometric cutting of aluminium at 500 m s^{-1} and 10° rake angle (workpiece dimensions are given in table 1).

Cutting conditions and output parameters	MD	MC
Cutting force per unit width (nN Å^{-1})	1.255	2.53
Thrust force per unit width (nN Å^{-1})	0.958	1.17
Width of cut (Å)	40.5	43.1
Depth of cut (Å)	4.1	5.1
Force ratio	0.764	0.462
Specific energy (10^5 N mm^{-2})	0.306	0.496
Computational time (h)	8–10	14–16

Table 3. Computational parameters used in the MC simulation of nanometric cutting of single-crystal aluminium at a cutting velocity 5 m s^{-1}.

Configuration	Two-dimensional or orthogonal cutting
Work material dimensions	$5a \times 25a \times 15a$, where a is the lattice constant of the workpiece
Tool dimensions	$20a \times 11a \times 11a$, a is the lattice constant of the tool
Depth of cut	5.1 Å

5 m s^{-1}. Figures 17 (a)–(d), 18 (a)–(d) and 19 (a)–(d) show various stages of the MC simulations with 10°, 30° and 45° rake tools respectively. As the tool proceeds into the workpiece, we observe the formation of a primary shear zone and some subsurface deformation in the machined surface. We also observe elastic recovery of the atoms in the machined surface as the tool moves away from the machined surface just generated (figure 17 (d)). The high elastic recovery is attributed to the basic principle underlying the MC method, namely, that the system attempts to reach minimum potential. The same reason is also applicable for the increased crystalline structure observed in the chip during nanometric cutting instead of the highly deformed structure in the chip in conventional cutting and/or in MD simulations.

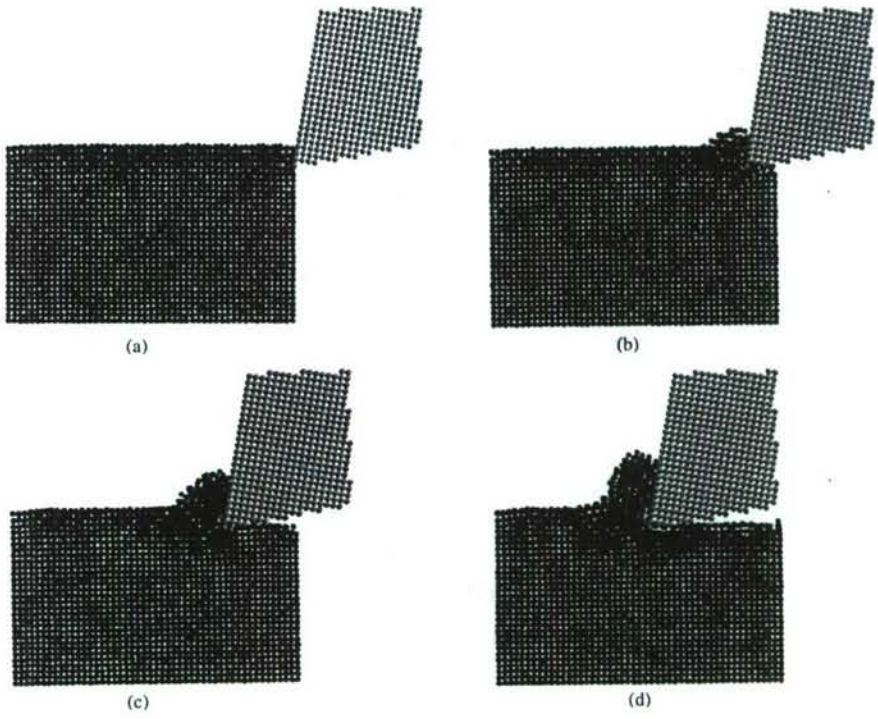


Figure 17. MC simulation at various stages of nanometric cutting (rake angle, 10°; cutting velocity, $5\,m\,s^{-1}$).

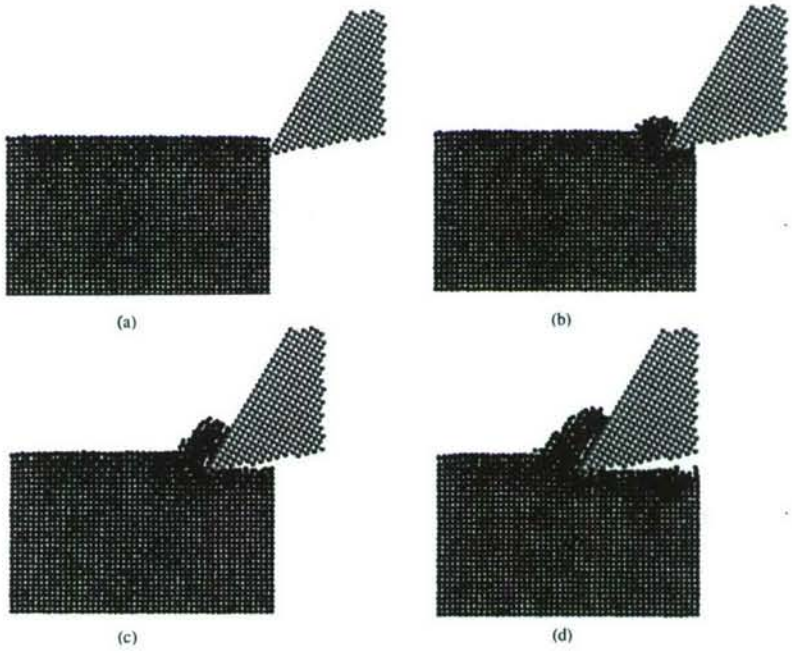


Figure 18. MC simulation at various stages of nanometric cutting (rake angle, 30°; cutting velocity, $5\,m\,s^{-1}$).

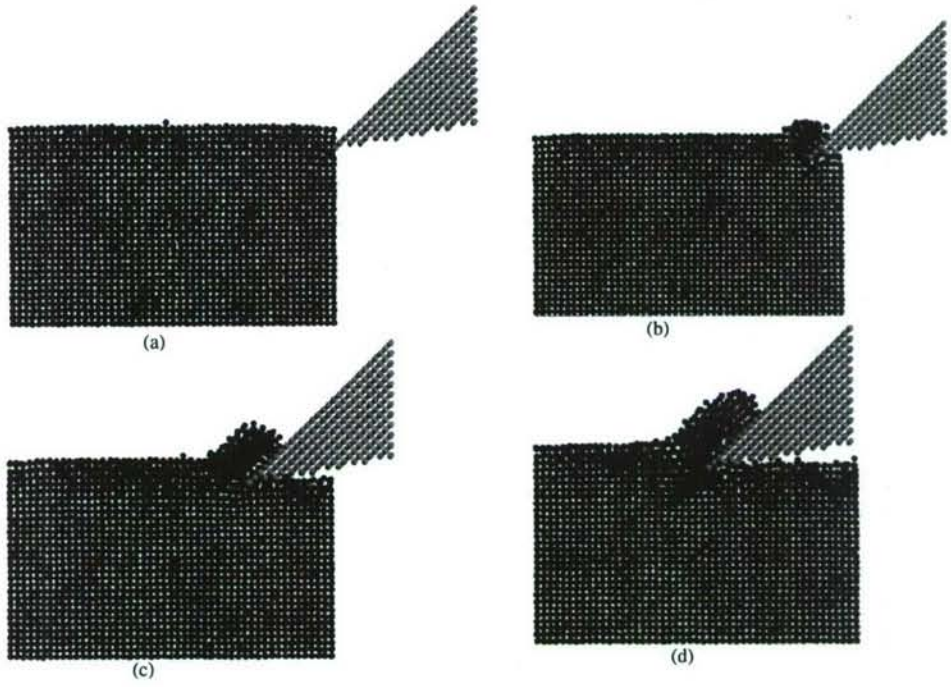
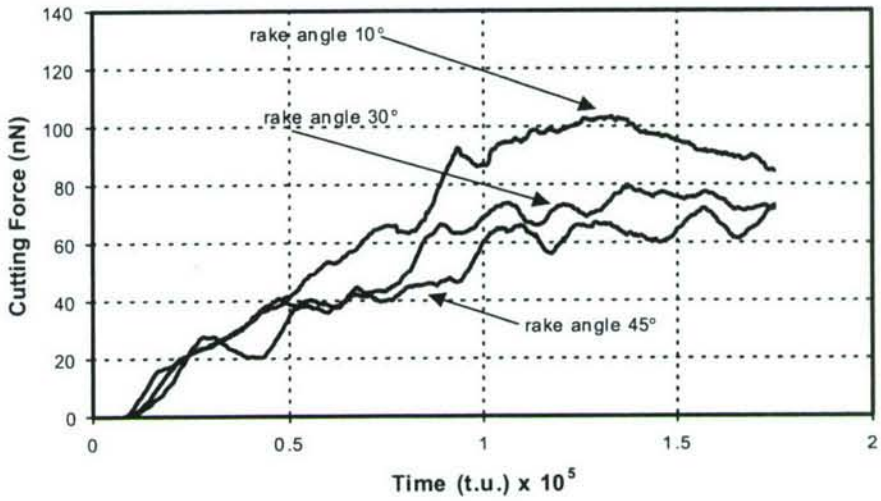


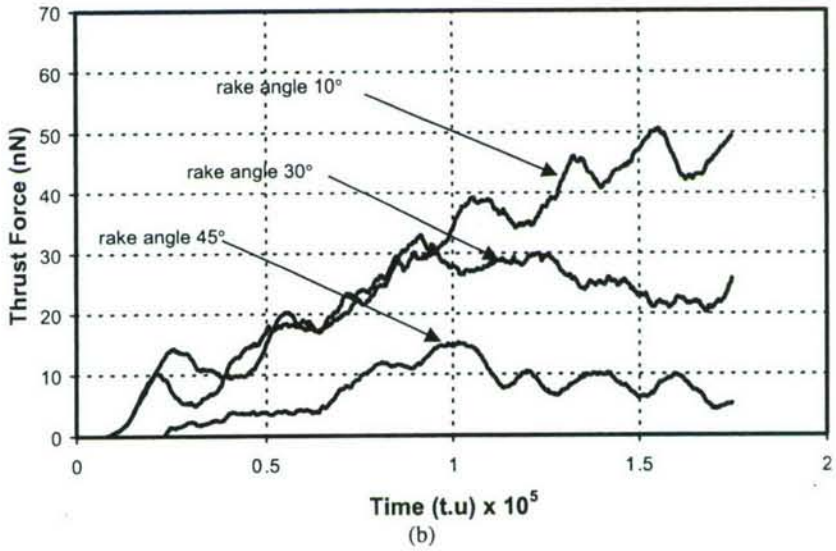
Figure 19. MC simulation at various stages of nanometric cutting (rake angle, 45°; cutting velocity, $5\,m\,s^{-1}$).

Figures 20 (a) and (b) show the variations in the cutting and the thrust forces respectively with cutting time for three positive rake angles, namely 10°, 30° and 45°, in MC simulations at a cutting speed of $5\,m\,s^{-1}$. It can be seen that both cutting and thrust forces decrease as the rake angle increases. Figure 21 shows the variations in mean cutting and thrust forces per unit width with rake angle in MC simulations at a cutting speed of $5\,m\,s^{-1}$. It can be seen that both the forces decrease with increase in rake angle. Figure 22 shows the variation in the force ratio (the ratio of the thrust force to the cutting force) with rake angle in MC simulations at a cutting speed of $5\,m\,s^{-1}$. The force ratio decreases as the rake angle increases from 10° to 45° as shown in figure 22 and table 4.

Figure 23 shows the variations in frictional force per unit width on the tool face and the shear force per unit width on the shear plane, respectively, with rake angle in MC simulations at a cutting speed of $5\,m\,s^{-1}$. It can be seen that both forces decrease as the rake angle increases (see table 5 for details). Figure 24 shows the variation in shear angle with rake angle at a cutting speed of $5\,m\,s^{-1}$. It can be seen that shear angle increases with increase in rake angle. This is because the chip thickness decreases with increase in rake angle, as can be seen from the MC simulations of nanometric cutting with different rake angle tools (figures 17–19). Figure 25 shows the variation in specific energy (energy required for removing unit volume of work material) with rake angle in MC simulations at a cutting speed of $5\,m\,s^{-1}$. It can be seen that the specific energy also decreases as the rake angle increases from 10° to 45° (see table 4 for details).



(a)



(b)

Figure 20. Variation in (a) cutting and (b) thrust forces with cutting time in MC simulation of nanometric cutting for different rake angles at a cutting speed of $5\,m\,s^{-1}$.

Figure 26 shows the variation in temperature in the cutting zone with cutting time. It can be seen that temperature in the cutting zone at $5\,m\,s^{-1}$ is about 400 K. This is in contrast with the temperature in the cutting zone of about 1000 K at $500\,m\,s^{-1}$.

§ 5. Conclusions

(1) MC simulation of nanometric cutting was conducted for the first time at a practical cutting speed of $5\,m\,s^{-1}$. The MC method incorporates a Markov random walk and the Metropolis acceptance–rejection criterion

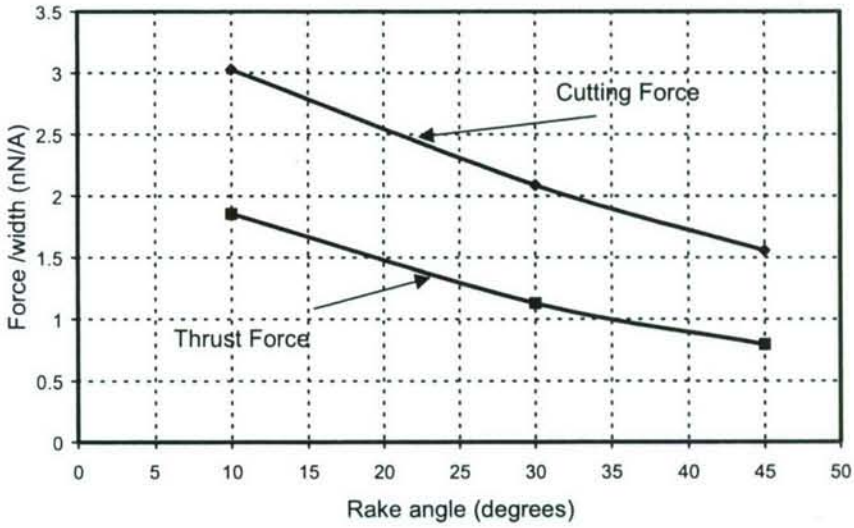


Figure 21. Variation in cutting force per unit width and thrust force per unit width with rake angle in MC simulation of nanometric cutting at a cutting velocity of $5\,m\,s^{-1}$.

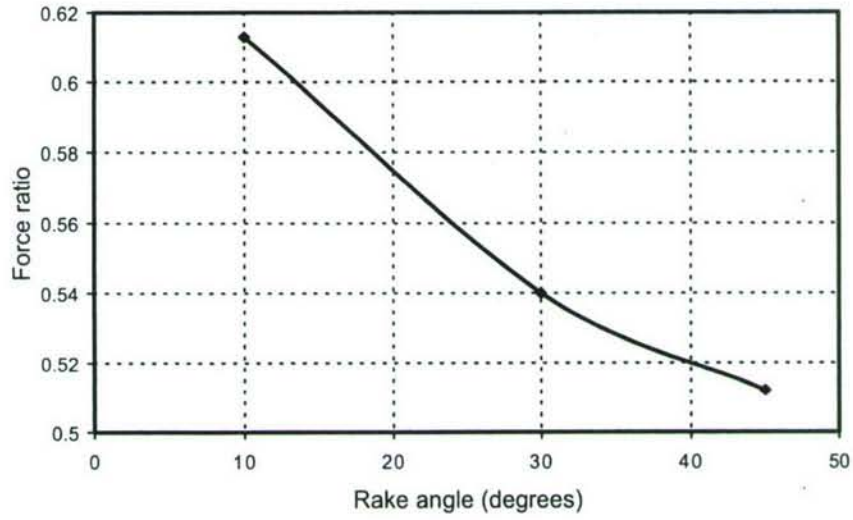


Figure 22. Variation in the force ratio (thrust force to cutting force) with rake angle in MC simulation of nanometric cutting at a cutting velocity of $5\,m\,s^{-1}$.

adopted for use with temperature gradients to determine the new configurations of the atoms.

(2) The temperature in the cutting zone in MD and MC simulations of nanometric cutting was estimated using the thermal model of the cutting process developed by Komanduri and Hou (2001b) on the macroscopic scale. Reasonably good agreement was found between the results of the thermal model and MD simulation results obtained using a harmonic oscillator energy partitioning. This local temperature is employed in the Boltzmann

Table 4. Results of MC simulation of nanometric cutting of aluminium at $5\,m\,s^{-1}$ for various rake angles, namely 10°, 30° and 45° (see table 3 for workpiece dimensions).

Rake angle (deg)	Cutting force per unit width ($N\,mm^{-1}$)	Thrust force per unit width ($N\,mm^{-1}$)	Force ratio	Specific energy ($10^5\,N\,mm^{-2}$)
10	3.0253	1.854	0.613	0.593
30	2.0885	1.128	0.540	0.384
45	1.5556	0.796	0.512	0.305

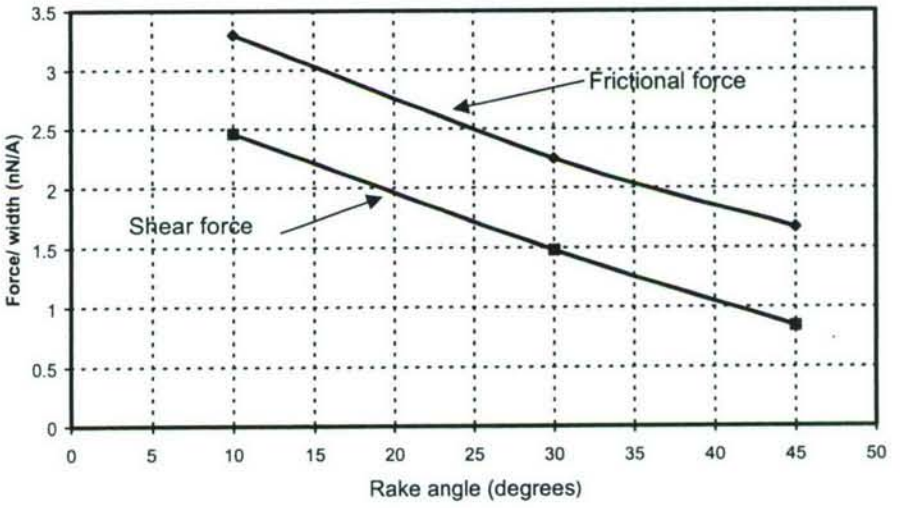


Figure 23. Variation in the frictional force per unit width and the shear force per unit width with rake angle in MC simulation of nanometric cutting at a cutting velocity of $5\,m\,s^{-1}$.

Table 5. Results of MC simulation of nanometric cutting of aluminium at $5\,m\,s^{-1}$ for various rake angles (see table 3 for workpiece dimensions).

Rake angle (deg)	Shear force per unit width ($N\,mm^{-1}$)	Shear angle (deg)	Frictional force per unit width ($N\,mm^{-1}$)
10	2.46	14.54	3.30
30	1.48	23.84	2.25
45	0.84	34.24	1.67

acceptance–rejection criterion. Thus, cutting speed is indirectly incorporated into the MC simulation via the temperature in the cutting zone since cutting speed and cutting velocity are related.

(3) A comparison of MD and MC simulations of nanometric cutting conducted at a cutting velocity of $500\,m\,s^{-1}$ yielded reasonably good agreement under similar cutting conditions. As expected, static forces obtained from MC calculations exceed the corresponding MD results due to the presence of

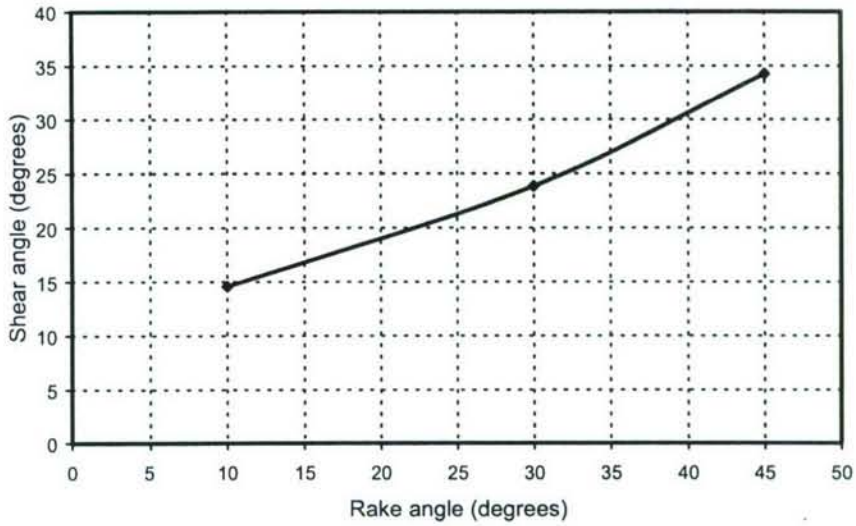


Figure 24. Variation in the shear angle with rake angle in MC simulation of nanometric of aluminium at a cutting speed of $5\,m\,s^{-1}$.

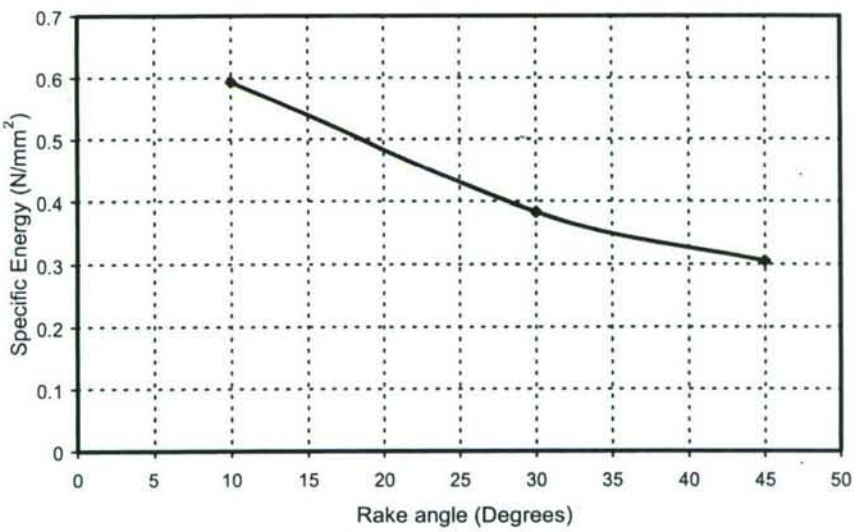


Figure 25. Variation in specific energy with rake angle in MC simulation of nanometric cutting of aluminium at a cutting velocity of $5\,m\,s^{-1}$.

inertial forces in the MD simulations and to a longer cut depth for the MC studies.

(4) MC simulations of nanometric cutting were conducted on a single-crystal aluminium work material in the (001) plane and cut in the [100] direction using three positive rake angle tools, namely 10°, 30° and 45°, at a cutting velocity of $5\,m\,s^{-1}$ to investigate the effect of rake angle on the cutting and thrust forces, force ratio, frictional force on the tool face, shear force on the shear plane, shear angle and specific energy. The cutting and thrust forces, the force ratio, the frictional force per unit width on the tool face, the shear

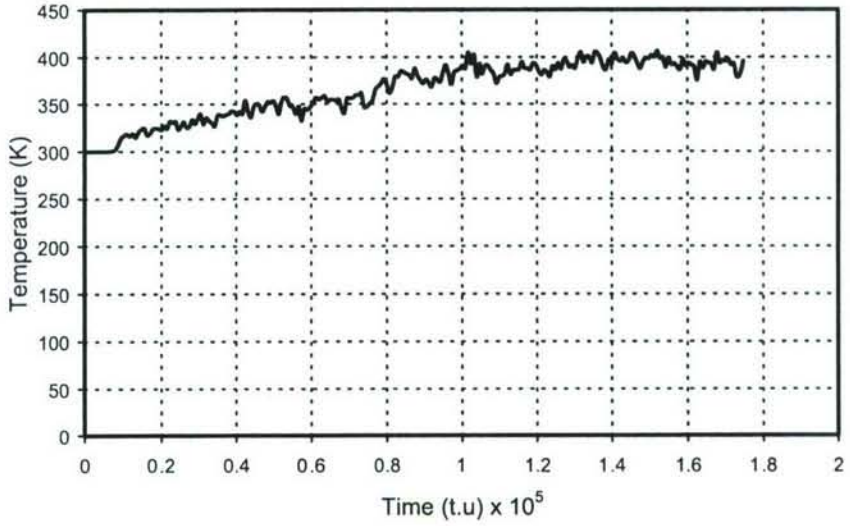


Figure 26. Variation in temperature in the cutting zone with cutting time in MC simulation of nanometric cutting (rake angle, 10°; cutting velocity, $5\,m\,s^{-1}$).

force per unit width on the shear plane, and the specific energy all decreased as the rake angle increases, as found in practice. Only the shear angle, as expected, was found to increase with increase in rake angle owing to the decrease in the chip thickness with increase in rake angle. These variations with rake angle are consistent with practical cutting experience.

(5) The temperature in the cutting zone at a cutting velocity of $5\,m\,s^{-1}$ was estimated to be about 400 K with a 10° rake tool. This is in contrast with the temperature in the cutting zone of about 1000 K with a 10° rake tool at $500\,m\,s^{-1}$.

(6) Although this is not shown expressly in the present paper there appears to be ample scope for optimization of the MC simulation of nanometric cutting by combining MC simulation with the DT or the steepest-descent technique as well as with the use of massively parallel processors to reduce significantly the computational time. Work in this area is currently under way with initial indications of some promising results.

Acknowledgements

This project is funded by grant DMI-0200327 from the National Science Foundation. Thanks are due, in particular, to Dr W. DeVries, Dr G. Hazelrigg, Dr J. Chen and Dr D. Durham of the Division of Design, Manufacturing, and Industrial Innovation, to Dr B. M. Kramer, of the Engineering Centers Division, and to Dr J. Larsen Basse of Tribology and Surface Engineering program for their interest in and support of this work. One of the authors (R.K.) also thanks the A. H. Nelson, Jr. Endowed Chair in Engineering for additional support for this activity. This project was also sponsored by a DEPSCoR grant on the Multiscale Modeling and Simulation of Material Processing (F49620-03-1-0281). The authors thank Dr Craig S. Hartley of the AFOSR for his interest in and support of this work.

References

Belak, J., and Stowers, I. F., 1990, *Proceedings of the ASPE Annual Conference*, Rochester, New York (Raleigh, NC: American Society of Precision Engineers), p. 76.

Chao, B. T., and Trigger, K. J., 1953, *Trans. ASME*, **75**, 109; 1955, *ibid.*, **75**, 1107.

Girifalco, L. A., and Weizer, V. G., 1959, *Phys. Rev.*, **114**, 687.

Hahn, R. S., 1951, *Proceedings of the First US National Congress of Applied Mechanics* (New York: American Society of Mechanical Engineers), p. 661.

Herrmann, D. W., 1994, *Computer Simulation Methods in Theoretical Physics* (Berlin: Springer).

Komanduri, R., and Hou, Z. B., 2000, *Int. J. Mech. Sci.*, **42**, 1715; 2001a, *ibid.*, **43**, 57; 2001b, *ibid.*, **43**, 89.

Komanduri, R., and Raff, L. M., 2001 *Proc. Instn. Mech. Engrs, Part B: J. Engng Mf.*, **215**, 1639.

Komanduri, R., Chandrasekaran, N., and Raff, L. M., 2001, *Int. J. Mech. Sci.*, **43**, 2237.

Komanduri, R., Raff, L.M., and Chandrasekaran, A., 2002, *Phil. Mag. Lett.*, **82**, 247.

Merchant, M. E., 1945, *J. Appl. Phys.*, **216**, 267, 318.

Metropolis, N., and Ulam, S. M., 1949, *J. Am. statist. Assoc.*, **44**, 335.

Narulkar, R., 2003, MS Thesis, Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, Oklahoma, USA.

Raff, L.M., 2001, *Principles of Physical Chemistry*, (Upper-Saddle River, New Jersey: Prentice-Hall), chapter 17, pp. 966–987.

Rubinstein, R. Y., 1981, *Simulation and the Monte Carlo Method* (New York: Wiley).

Ueda, T., Sato, M., and Nakayama, K., 1998, *Ann. CIRP*, **47**, 41.

von Neumann, J., 1951, *US natn. Bur. Stand. appl. Math. Ser.*, **12**, 36.

Monte Carlo–steepest descent (MC–SD) simulations of nanometric cutting

R Narulkar[1], **L M Raff**[2], and **R Komanduri**[1]*

[1]Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, Oklahoma, USA

[2]Department of Chemistry, Oklahoma State University, Stillwater, Oklahoma, USA

The manuscript was received on 5 August 2004 and was accepted after revision for publication on 11 November 2004.

DOI: 10.1243/174034905X35351

Abstract: In order to reduce the computational time, Monte Carlo (MC) simulations of nanometric cutting have been modified to include a combination of steepest descent (SD) and Monte Carlo procedures. This MC–SD combination is found to reduce the required computational times by factors of at least two to three over those achieved using MC methods alone. The MC–SD method is applied to the nanometric cutting of single-crystal aluminium along the (100) plane with different rake angle tools at a cutting speed of 5 m/s. The results obtained from the MC–SD calculations are found to be almost identical to those resulting from the MC simulations.

Keywords: nanoelectronics, nanowires, electronic assemblies/devices

1 INTRODUCTION

Molecular dynamics (MD) simulations are powerful tools for modelling the machining process at nanoscale [**1**]. However, if such simulations are carried out using experimental speeds for cutting, grinding, or tensile pulling, the required computational times become prohibitively large. For this reason, most MD simulations are conducted at high cutting speeds (100–500 m/s). The major part of this computational time is devoted to numerical integration of the classical Newtonian equations of motion for the interacting particles to obtain the temporal behaviour of the workpiece atoms. If, at any given moment, the cutting process is stopped, continued integration of the equations of motion will give the final, metastable structure of the workpiece atoms. Moreover, it will be possible not only to have this final structure, but also detailed information concerning the mechanism by which the individual atoms attain that structure. In many cases, it is only the final metastable configurations that are of interest. When this is the situation, MD calculations provide far more information than that is required, and that information is obtained at the expense of extremely large quantities of computational time.

Corresponding author: Mechanical and Aerospace Engineering, Oklahoma State University, 218 Engineering North, Stillwater, OK 74078, USA. email: ranga@ceat.okstate.edu

In many applications, the above problems have been avoided by employing Monte Carlo (MC) methods. Such procedures allow the metastable or equilibrium states for the system of interest to be determined without the necessity of solving large sets of coupled differential equations that provide unnecessary information about the dynamics through which these states are attained. In the large majority of cases, MC methods are applied to canonical systems that are characterized by phase–space distributions that are Boltzmann so that the system has a well-defined temperature [**2**, **3**]. MC procedures have also been successfully applied to microcanonical systems in which the total system energy is fixed and all system configurations that have a given total energy have equal probability of existence. Nyman, *et al.* [**4**] and Severin, *et al.* [**5**] have developed procedures for executing efficient sampling of such microcanonical systems. These methods have been employed by Schranz, *et al.* [**6**] in studying simple bond scission reactions of 1,2-difluoroethane [**7**] and both silylene and disilane [**8**].

The principal problem that must be addressed to develop an MC applicable to cutting experiments is to find a procedure that will incorporate the effects of cutting speed. Since there is no time variable in an MC calculation, it is not immediately obvious how this might be done for systems that are neither

canonical nor microcanonical. This can be accomplished [**9**, **10**] by defining a local temperature in the cutting zone using the thermal analysis developed by Komanduri and Hou [**11–13**] for conventional machining. Extension of this method to the nanometric regime permits an accurate estimate of the local temperature in cutting as a function of the spatial coordinates of the workpiece. This temperature is then employed in the Boltzmann probability distribution function that is used to determine the acceptance–rejection ratio of the MC moves in the simulation. Since cutting speed is closely related to cutting temperature, the cutting speed enters the calculation via the thermal analysis equations.

At cutting speeds around 500 m/s, it was found that MC simulations for cutting of an aluminium workpiece with a 10° rake angle require somewhat more time than the corresponding MD calculations [**9**, **10**]. This is due to the very large number of MC moves that must be executed to converge to the desired metastable structures throughout the simulation. However, as the cutting speed decreases, MD calculations become increasingly demanding of computational time whereas MC methods are nearly independent of cutting speed. Therefore, at low cutting speeds, MC calculations require two orders of magnitude less time than MD simulations.

In this paper, an improved MC method is presented that decreases the required computational time by at least a factor of two to three depending upon the experimental conditions, the number of atoms in the model, and – presumably – upon the nature of the interaction potential employed. The modification is based on the fact that MC methods always drive the configuration of the system toward a potential minimum. Therefore, the number of MC moves required to attain this minimum can be reduced by using a damped trajectory (DT) procedure [**14**] or steepest descent (SD) method to compute a configuration close to the potential minimum before executing MC moves. Because DT and SD methods converge to the nearest local potential minimum, the above methods alone will usually not correctly determine the final metastable configuration attained by the system at a given point during a simulation. To circumvent this problem, a series of MC moves obtained from a Markov chain [**2**, **3**] are executed after completion of the DT or SD calculation. By alternating MC moves and either SD or DT calculations, the system rapidly converges to the desired metastable configuration. For the systems investigated in this research, it was found that the MC–SD computational times are reduced by factors ranging between two to three over those required for purely MC simulations.

2 BRIEF REVIEW OF LITERATURE

MD simulations were initiated in the late 1950s at the Lawrence Radiation Laboratory (LRL) in the United States by Alder and Wainwright [**15**] in the field of statistical mechanics. Since then, MD simulations have been extensively applied to many diverse fields, including, crystal growth [**16**], indentation [**17**], tribology [**18**], low-pressure diamond synthesis [**19–24**], and laser interactions [**25**], to name a few. MD simulation of nanometric cutting was pioneered by Belak, Boercker, and Stowers [**26**, **27**] at the Lawrence Livermore National Laboratory (LLNL). This work was followed by some research groups in Japan [**28–35**] and in the US by the authors of the present paper at Oklahoma State University [**36**]. For details, the readers are referred to an extensive review article on this subject by the authors containing some 124 references [**1**].

MC methods utilize random procedures that allow simple solutions to be obtained for physical problems of a complex nature [**37**, **38**]. The term 'Monte Carlo' is used in connection with the probability sampling techniques that approximate the solutions of a wide range of mathematical and physical problems. Metropolis *et al.* [**39**] of the Los Alamos National Laboratory in New Mexico were the first to introduce the MC method for the study of a system of interacting particles in 1953. Since then it has been used extensively in physics, chemistry, economics, weather forecasting, and financial transactions, to name a few. The literature on these topics is quite extensive, including several textbooks [**37**, **38**, **40**, **41**]. However, virtually all of these are on systems that are either canonical or microcanonical. In these cases, there are usually no external forces present and the system energy is conserved.

There are, however, very few, if any, MC papers dealing with such systems as cutting where there are large external forces that cause the system energy to grow rapidly in such a manner that the system is neither canonical nor microcanonical. The present authors were, perhaps, the first to apply MC simulations to nanometric cutting [**9**]. This approach enabled simulation of the cutting process at conventional speeds (5 m/s) instead of extremely high speeds (100–500 m/s) used in conventional MD simulations. As already pointed out, this is accomplished by defining a local temperature in the cutting zone using the thermal analysis developed by Komanduri and Hou [**11–13**]. This temperature was then employed in the Boltzmann probability distribution that is used to determine the acceptance–rejection ratio of the Monte Carlo moves in the simulations. Since cutting speed is closely related to cutting temperature, the cutting speed thus enters the calculation via the thermal analysis equations. The present investigation

is a follow up of that work with the objective of reducing the computational time by using a combination of SD and MC procedures.

3 LOCAL MINIMUM METHODS

3.1 Damped trajectories (DTs)

In a DT procedure, the kinetic energy of all the lattice atoms is set to zero and the Hamiltonian equations of motion for these atoms are integrated to a point at which the total potential energy attains a minimum. This is defined as one DT cycle. Once the system passes through a potential minimum, the total kinetic energy is reset to zero and subsequent cycles are executed until convergence to a local minimum potential is reached. Since the system potential is independent of the nuclear masses, the time required for the execution of the DT procedure can be decreased by setting the mass of all the workpiece atoms equal to that of hydrogen. Moreover, since energy conservation requirements are of no concern, larger integration step sizes may be employed. Komanduri, *et al.* [**14**] previously employed a combination of MC and DT procedures in their studies of uniaxial tension of face-centred cubic (FCC) metals. However, when the metastable structures become very amorphous as is the case in cutting simulations, the convergence rate for damped trajectories becomes very slow. In such cases, SD methods were found to offer superior computational speeds.

3.2 Steepest descent

A SD method makes use of the fact that the gradient of a hypersurface is a vector, the direction of which corresponds to the direction of the maximum rate of increase of the value of the hypersurface [**42–44**]. Thus, a move is made in the direction opposite to the surface gradient, that will be a movement towards the surface minimum. The method can be represented by

$$V_{i+1} = V_i - \alpha \vec{\nabla} V \tag{1}$$

where, α is the step size in the direction of the gradient and $\vec{\nabla} V$ is the surface gradient at point i. As α increases, the convergence rate to the potential minimum increases, but the resolution of the precise position of the minimum decreases. One SD cycle is defined as being one movement of every atom of the workpiece material according to equation (1). Figure 1 shows a flow diagram for the execution of one SD cycle. Convergence is monitored by examination of the magnitude of change of the total potential in successive SD cycles.

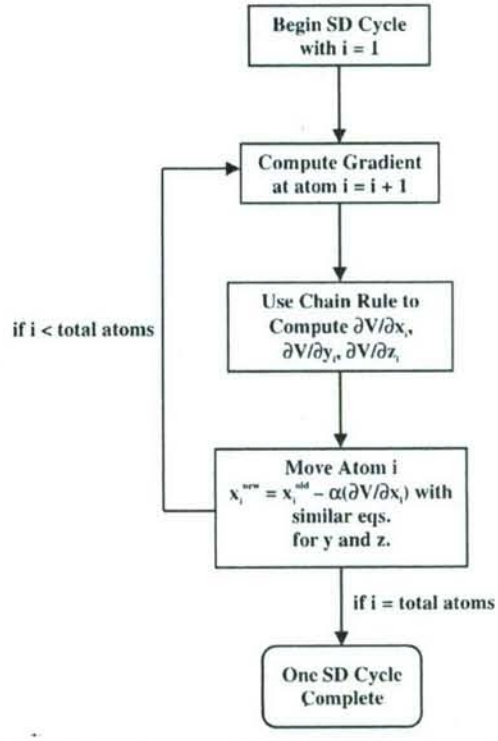


Fig. 1 Flow diagram for the execution of one SD cycle

In the present application, the workpiece and workpiece–tool interaction potentials are assumed to be a summation of pairwise Morse functions, i.e.

$$V_{\text{Total}} = \sum_{i=1}^{N_{\text{WP}}} \sum_{j>i}^{N_{\text{WP}}} V_{\text{WP}}^{M}(r_{ij}) + \sum_{i=1}^{N_{\text{WP}}} \sum_{j=1}^{N_{\text{T}}} V_{\text{L}}^{M}(r_{ij}) \tag{2}$$

In equation (2), $V_{\text{WP}}^{M}(r_{ij})$ and $V_{\text{L}}^{M}(r_{ij})$ are the pairwise interaction potentials between workpiece atoms and between workpiece–tool atoms, respectively. N_{WP} and N_{T} are the number of atoms in the workpiece and tool, respectively. The pairwise terms are given by

$$V_{\text{WP}}^{M}(r_{ij}) = D_{\text{WP}}\{\exp[-2a_1(r_{ij} - r_1^{eq})] - 2\exp[-\alpha_1(r_{ij} - r_1^{eq})]\} \quad \text{for } r_{ij} \leqslant r_{\text{WP}}^{c}$$

$$V_{\text{WP}}^{M}(r_{ij}) = 0 \quad \text{for } r_{ij} > r_{\text{WP}}^{c}$$

$$V_{\text{L}}^{M}(r_{ij}) = D_{\text{L}}\{\exp[-2\alpha_2(r_{ij} - r_2^{eq})] - 2\exp[-\alpha_2(r_{ij} - r_2^{eq})]\} \quad \text{for } r_{ij} \leqslant r_{\text{L}}^{c}$$

and

$$V_{\text{L}}^{M}(r_{ij}) = 0 \qquad \text{for } r_{ij} > r_{\text{L}}^{c} \tag{3}$$

In equation (3), r_{WP}^{c} and r_{L}^{c} are the cut-off radii for workpiece–workpiece and workpiece–tool interactions, respectively. The gradients required for the

SD cycles are easily computed using equations (2) and (3) combined with the chain rule to convert the derivatives to Cartesian coordinates.

4 MC–SD SIMULATION MODEL

The MC–SD method combines MC and SD calculations in a manner designed to achieve rapid convergence to the equilibrium metastable structures that characterize various stages of the cutting process. By alternating sets of MC moves with SD cycles, the desired metastable structures can be rapidly determined, without the necessity to integrate thousands of differential equations. Consequently, the need to compute millions of potential–surface derivatives whose functional forms are generally much more complex than the potential itself can be avoided. In addition, the MC calculations require the change in potential rather than the potential itself. Therefore, those potential terms that change upon execution of Markov step need to be considered. This will be a small subset of the total set of terms comprising the total potential. As a result, the computational time required for the combined MC–SD procedure is greatly reduced over that needed for an MD method.

Figure 2 shows a schematic model used for MC and MC–SD cutting simulations of a single-crystal aluminium workpiece. The workpiece is divided in two zones, a moving zone and a boundary or stationary zone. Since nature always drives any system towards its most stable configuration or minimum potential, in MC simulations configurations are randomly selected that move the system towards this configuration. In most cases, the probability density function for configuration points is unknown so that this selection cannot be accomplished in a straightforward fashion. A common method to handle this type of problem involves the generation of random configurations using Markov chains generated by the Metropolis sampling procedure [**2**, **3**].

Consider an initial configuration q_0 for an N-particle system. The Boltzmann distribution function $P(q_0)$ for canonical systems is given by

$$P(q_0) = C \exp\left[-\frac{V(q_0)}{kT}\right] \tag{4}$$

where V and T are the potential and temperature, respectively, for configuration q_0. k is the Boltzmann constant and C is, in effect, a normalization constant. A subset of m particles from a total of N particles is now selected and the coordinates of these particles are randomly varied. Here, m is very small compared with N. For the ith atom, the new configuration is obtained using the following equations

$$x_i^{new} = x_i^{old} + (0.5 - \xi_{i1})\Delta x \tag{5}$$

$$y_i^{new} = y_i^{old} + (0.5 - \xi_{i2})\Delta y \tag{6}$$

$$z_i^{new} = z_i^{old} + (0.5 - \xi_{i3})\Delta z \tag{7}$$

where x_i^{new}, y_i^{new}, and z_i^{new} are the new configuration coordinates for particle i and x_i^{old}, y_i^{old}, and z_i^{old} are those for the old configuration. Δx, Δy, and Δz are the maximum step sizes along the X, Y, and Z directions, respectively, and the ξ_i are randomly chosen numbers in the interval [0, 1].

For canonical systems, this new configuration point will be either accepted or rejected depending upon the change in the Boltzmann probability given by equation (4). The move is accepted, if

$$\exp\left(-\frac{\Delta V}{kT}\right) \geqslant \xi \tag{8}$$

where ξ is a random number in the interval [0, 1] and $\Delta V = V_{new} - V_{old}$. If the move is rejected, the configuration q_0 is returned to and the procedure is repeated until a move is accepted. More details of

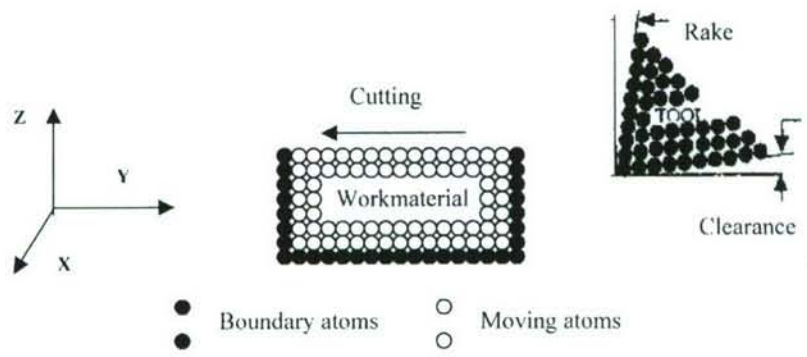


Fig. 2 Schematic model used for MC and MC–SD cutting simulations of a single-crystal aluminium workpiece

the procedure are provided in Komanduri, *et al.* [9].

The principal problem encountered when applying MC methods to cutting simulations is that the system is neither canonical nor microcanonical. Large temperature gradients are present and hence, the temperature appearing in equation (8) is undefined. In the initial work of Komanduri, *et al.* [9], this problem was addressed by defining a local temperature and then employing the thermal analysis equations developed by Komanduri and Hou [11–13] to obtain this local temperature as a function of the spatial coordinates of the atoms on which Markov moves are being executed. This thermal analysis considers the combined effect of two principal heat sources, the shear plane and the tool–chip interface frictional heat sources. Since one of the variables that enters the analysis is the cutting speed, this procedure permits the incorporation of the effects of experimental cutting speed into the simulations.

The MC–SD calculations are conducted by repeated application of the following steps:

(a) translation of the tool in the cutting direction;
(b) calculation of the temperature rise using the thermal model of Komanduri and Hou [13] in the chip, the shear zone, and beneath the tool;
(c) calculation of M SD cycles to move the atoms of the system close to the configuration corresponding to the nearest local minimum;
(d) execution of a Markov walk of S_w and S_c steps for atoms in the workpiece and in the chip, respectively, using the temperature distribution calculated in step (b).

Steps (a)–(d) are repeated until the desired cutting is achieved.

5 APPLICATION AND RESULTS

The MC–SD procedure was tested against pure MC simulations using pairwise potentials to model the cutting process along the (100) plane of a single-crystal aluminium workpiece with an infinitely hard tool. The potential parameters are given in Table 1.

Table 1 Morse parameters used in the MC–SD and MC simulations of nanometric cutting of single-crystal aluminium

Parameter	Value
D_{WP}	0.27030 eV
α_1	1.16460 Å^{-1}
r_1^{eq}	3.253 Å
D_1	0.087 eV
α_2	5.14 Å^{-1}
r_2^{eq}	2.05 Å

Table 2 MC and MC–SD parameters employed in the calculations

Parameter	Value
Δx	0.1 Å
Δy	0.1 Å
Δz	0.1 Å
α	0.01 Å
M	4
$(S_w)_{MC}$	40
$(S_c)_{MC}$	400
$(m)_{MC}$	1
$(S_w)_{MCSD}$	20
$(S_c)_{MCSD}$	100
$(m)_{MCSD}$	1

The MC and SD parameters employed in the study are given in Table 2.

The MC–SD method was first tested by simply relaxing a $(5 \times 40 \times 15)$ aluminium workpiece using MC methods alone and comparing the system potential at each iteration with that obtained when the relaxation was executed using the MC–SD procedure. The comparison is shown in Fig. 3. It can be seen that in both cases, the potential initially drops rapidly and then fluctuates about a common mean value in both methods. At any given point in the process, the percent difference between the potentials computed by the two methods is always less than 0.12 per cent. Thus, it can be seen that the same relaxed state is achieved using both MC and MC–SD methods. However, significantly fewer MC moves are required for the MC–SD calculations. The results also demonstrate that SD methods attain convergence more rapidly than purely MC procedures.

In the second test, cutting simulations were conducted with a 10° rake angle tool at a cut depth corresponding to three layers of aluminium atoms at a cutting speed of 5 m/s. The change in potential of the metastable configuration obtained at the completion of each movement of the tool was computed for each of the simulations. The total cutting distance was 80 Å. This distance was covered by 800 tool movements, each covering a cutting distance of 0.10 Å. The results for both MC and MC–SD simulations are shown in Fig. 4. As can be seen, there is virtually no difference in the potentials of the computed final metastable configurations. The maximum absolute percent difference is about 0.18 per cent. Since this difference is less than the statistical error present in the MC method, it is not expected to exert any effect on the final results. Clearly, the two methods give equivalent results.

Figure 5 shows the computed change in cutting force during the cutting simulations used to obtain the results shown in Fig. 4. The time on the abscissa of Fig. 5 is computed by simply dividing the cutting speed by the total distance moved by the tool. It is

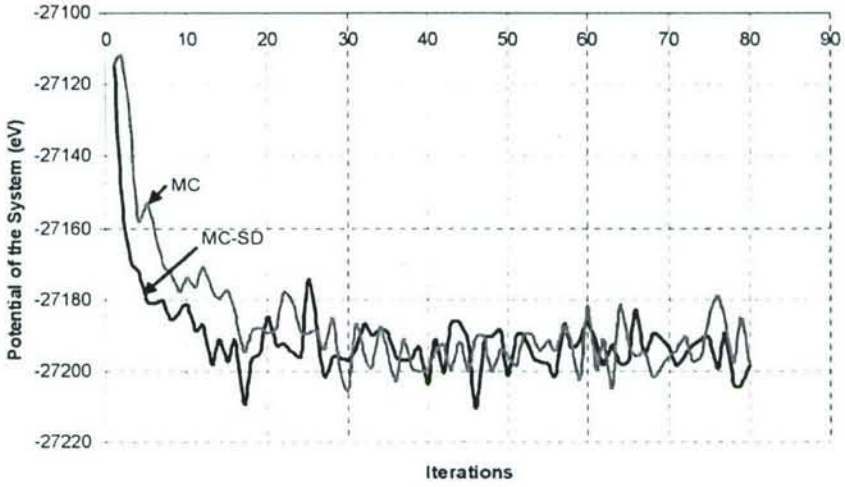


Fig. 3 Potentials during relaxation of a (5 × 40 × 15) aluminium workpiece whose potential is described by equations (2) and (3) with the parameters given in Table 1. The two curves show close agreement of the results obtained using MC and MC–SD methods

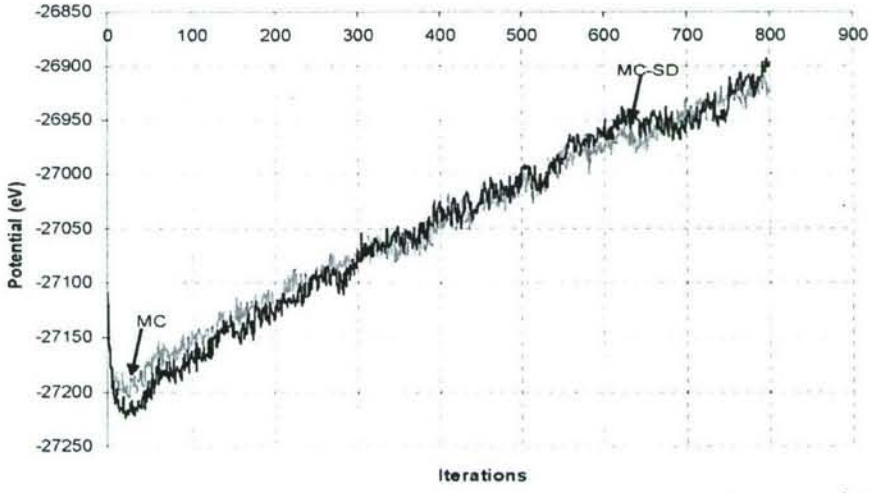


Fig. 4 Changes in the potential of metastable configurations obtained by MC and MC–SD simulations of aluminium cutting at 5 m/s, 10° rake angle, at a cut depth corresponding to three layers of lattice atoms

clear that, except for statistical difference produced by the stochastic nature of MC calculations, the two results are essentially identical.

Figure 6 shows snapshots of the configuration of the system at three different stages of cutting for the two simulations. Again, it can be seen that the two methods are in excellent agreement.

Finally, four different comparison calculations of the two methods were conducted using the same cutting parameters and potential parameters as those employed to obtain the results shown in Figs 3–6. The calculations differ only in the total number of lattice atoms included in the aluminium workpiece. The criterion for the selection of the number of atoms in the simulation is size convergence, i.e. when the results become insensitive to further increase in the number of atoms considered, the model is adequate. It was typically found in the nanometric cutting studies that the size convergence is obtained with a few thousand atoms (<10 000) [1].

Figure 7 shows a comparison of the computational times required by both MC and MC–SD simulations. It may be noted that selected 8 700 lattice atoms have been selected, which is typical of MD

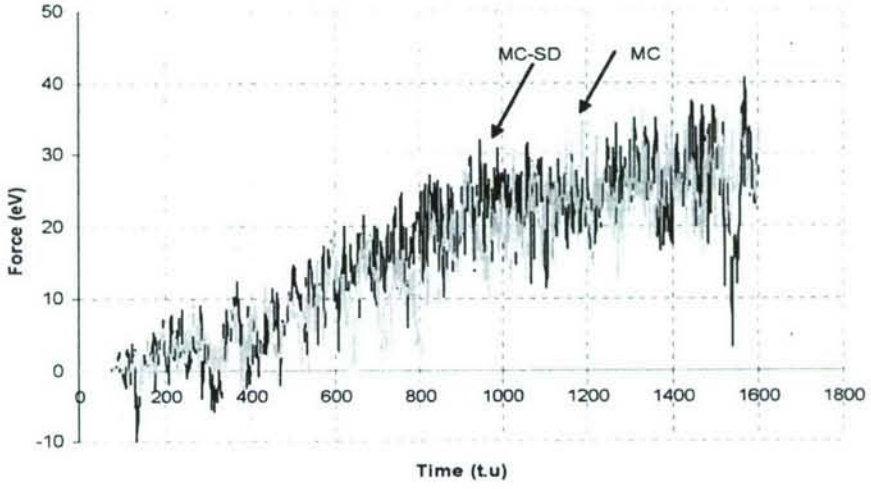


Fig. 5 Computed changes in cutting force during the MC and MC–SD simulations used to obtain the results shown in Fig. 4. Time is computed from the ratio of the cutting speed and total distance moved by the tool

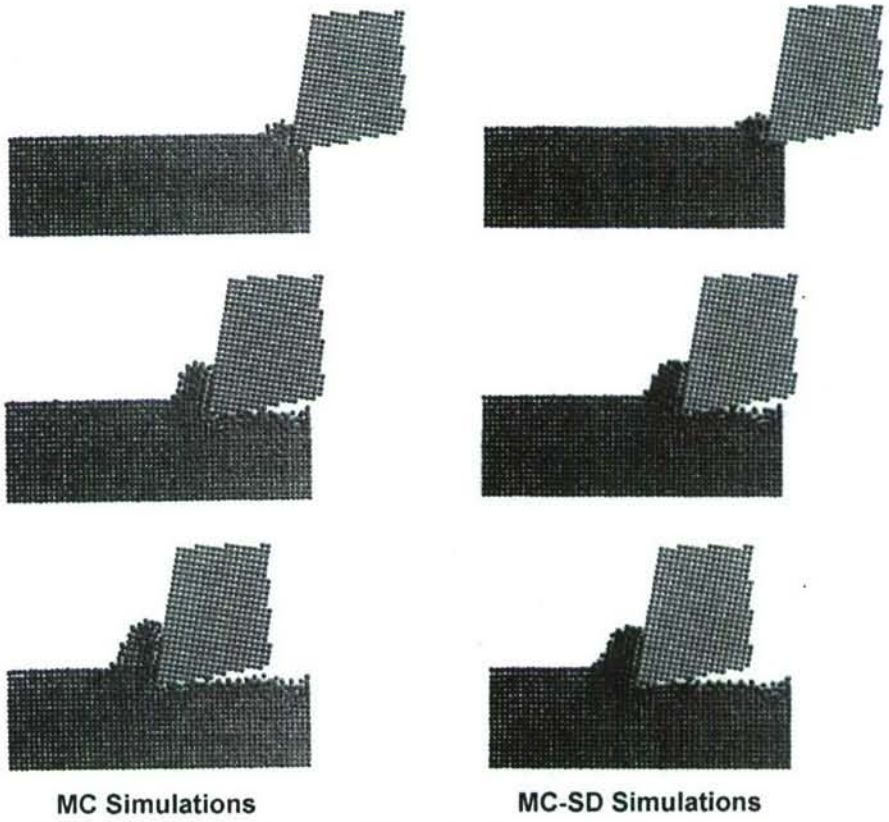


Fig. 6 MC and MC–SD simulations of cutting of a ($5 \times 40 \times 15$) aluminium workpiece with 10° rake angle at a cut depth corresponding to three layers of lattice atoms with a cutting speed of 5 m/s. The system potential is given by equations (2) and (3) with the parameter values listed in Table 1. The snapshots on the left are those obtained from the MC calculations. Those on the right are snapshots at the same moment of time determined by MC–SD methods

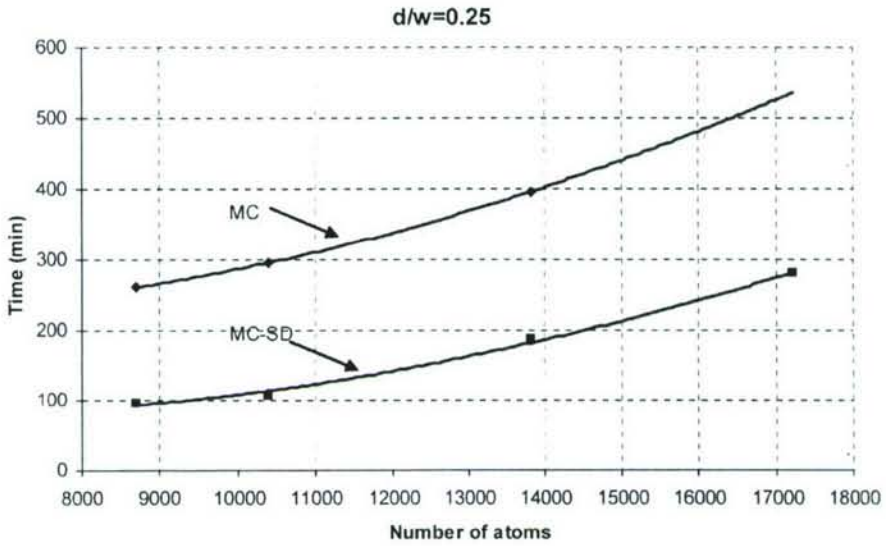


Fig. 7 Computational times for MC and MC–SD simulations of cutting of an aluminium workpiece at a cutting speed of 5 m/s at a cut depth corresponding to three layers of lattice atoms as a function of the size of the simulated workpiece. The system potential is given by equations (2) and (3) with the parameter values listed in Table 1

simulations of nanometric cutting, and 17 200 as an upper limit. For a system containing 8700 lattice atoms, the computational time ratio for MC and MC–SD calculations is about 2.7. When the number of lattice atoms increases to 17 200, this ratio decreases to about 1.9. Consequently, a computational time advantage between a factor of two to three using the MC–SD method can be achieved. It should be recognized that these ratios represent lower limits to the possible computational advantage that might be obtained for the test system considered here. No effort was made to optimize the MC–SD method by adjustment of α, M, $(S_w)_{MC-SD}$, and $(S_c)_{MC-SD}$. If such an optimization is done, further savings in processing time are anticipated.

6 CONCLUSIONS

1. To reduce the computational time, MC simulations of nanometric cutting have been modified to include a combination of SD and MC procedures.
2. Simulations involving MC–SD combinations are found to reduce the required computational times by a factor of two to three over those achieved using MC methods alone, thus offering computational time advantages with this method. As the number of atoms in the model decreases, the reduction in computational time increases. The results obtained from the MC–SD calculations are found to be almost identical to those resulting from the MC simulations.
3. It is anticipated that optimization of the MC–SD method by adjustment of α, M, $(S_w)_{MC-SD}$, and $(S_c)_{MC-SD}$ may provide additional computational time advantages.

ACKNOWLEDGEMENTS

This project is funded by a grant from the National Science Foundation (DMI-0200327). The authors thank Dr W. DeVries, Dr G. Hazelrigg, Dr J. Cao, and Dr D. Durham of the Division of Design, Manufacturing, and Industrial Innovation, Dr B. M. Kramer, Engineering Centers Division, and Dr J. Larsen Basse, Tribology and Surface Engineering Program, for their interest and support of this work. This project was also sponsored by a Department of Defence Experimental Study to Stimulate Competitive Research (DEPSCoR) grant on the Multiscale Modeling and Simulation of Material Processing (F49620-03-1-0281). The authors thank Dr Craig S. Hartley of the Air Force Office of Scientific Research (AFOSR) for his interest in and support of this work.

REFERENCES

1 **Komanduri, R.** and **Raff, L. M.** A review on the MD simulation of nanometric cutting. *Proc. Instn Mech. Engrs, Part B: J. Engng Manufacture*, 2001, 1639–1672.
2 **Metropolis, N.** and **Ulam, S. M.** The Monte Carlo method. *J. Am. Statist. Ass.*, 1949, **44**, 335.

3. **von Neumann, J.** Various techniques used in connection with random digits. *US Natn. Bur. Stand. Appl. Math. Ser.*, 1951, **12**, 36.
4. **Nyman, G., Nordholm, S.,** and **Schranz, H. W.** Efficient microcanonical sampling for a preselected total angular momentum. *J. Chem. Phys.* 1990, **93**(9), 6767.
5 **Severin, E. G., Freasier, B. C., Hamer, N. D., Jolly, D. L.,** and **Nordholm, S.** An efficient microcanonical sampling method. *Chem. Phys. Lett.* 1978, **57**, 117.
6 **Schranz, H. W., Raff, L. M.,** and **Thompson, D. L.** Correspondence of canonical and microcanonical rate constants using phase space theory for simple bond fissions. *Chem. Phys. Lett.* 1990, **171**, 68.
7 **Schranz, H. W., Raff, L. M.,** and **Thompson, D. L.** Non-statistical effects in bond fission reactions of 1,2-difluoroethane. *Chem. Phys. Lett.* 1991, **182**, 455.
8 **Schranz, H. W., Raff, L. M.,** and **Thompson, D. L.** Statistical and non-statistical effects in bond fission reactions of SiH_2 and Si_2H_6. *J. Chem. Phys.* 1991, **94**, 4219.
9 **Komanduri, R., Nuralkar, R.,** and **Raff, L. M.** Monte Carlo simulation of nanometric cutting. *Phil. Mag.*, 2004, **84**, 1155–1183.
10 **Nuralkar, R.** Mechanical and aerospace engineering. MS Thesis, Oklahoma State University, USA.
11 **Komanduri, R.** and **Hou, Z. B.** Thermal modeling of the metal cutting process, Part I – temperature rise distribution due to shear plane heat source. *Int. J Mech. Sci.*, 2000, **42**, 1715–1752.
12 **Komanduri, R.** and **Hou, Z. B.** Thermal modeling of the metal cutting process, Part II – temperature rise distribution due to frictional heat source at the chip–tool interface. *Int. J. Mech. Sci.*, 2001, **43**, 57–88.
13 **Komanduri, R.** and **Hou, Z. B.** Thermal modeling of the metal cutting process, Part III – temperature rise distribution due to the combined effect of shear plane heat source and frictional heat source at the chip–tool interface. *Int. J. Mech. Sci.*, 2001, **43**, 89–107.
14 **Komanduri, R., Raff, L. M.,** and **Chandrasekaran, A.** A combined Monte Carlo/damped trajectory (MC/DT) simulation of nanometric testing of FCC metals under uniaxial tension. *Phil. Mag. Lett.*, 2002, **82**(5), 247–256.
15 **Alder, B.** and **Wainwright, T.** Studies in molecular dynamics, I – general method, *J. Chem. Phys.*, 1959, **31**, 459. *Ibid*, Studies in molecular dynamics, II – behavior of a small number of elastic spheres, 1960, **33**, 1439.
16 **Dodson, B. W.** Molecular dynamics modeling of vapor-phase and very-low-energy ion-beam crystal growth processes. *Solid St. Mater. Sci.*, 1990, **16**(2), 115–120.
17 **Landman, U., Luedtke, W., Burnham, N.,** and **Colton, R.** Atomistic mechanisms and dynamics of adhesion, nanoindentation, and fracture. *Science*, 1990, **248**(17), 454–461.
18 **Kim, D. E.** and **Suh, N. P.** Molecular dynamics investigation of two-dimensional atomic scale friction. *Trans ASME, J. Tribology*, 1994, **116**, 225–231.
19 **Peploski, J., Thompson, D.,** and **Raff, L.** Molecular dynamics studies of elementary surface reactions of C_2H_2 and C_2H in low-pressure diamond-film formation. *J. Phys. Chem.*, 1992, **96**, 8538–8544.
20 **Chang, X., Perry, M., Peploski, J., Thompson, D.,** and **Raff, L.** Theoretical studies of hydrogen-abstraction reactions from diamond and diamond-like surfaces. *J. Chem. Phys.*, 1993, **99**, 4748–4758.
21 **Chang, X., Thompson, D.,** and **Raff, L.** Minimum-energy reaction paths for elementary reactions in low-pressure diamond-film formation. *J. Chem. Phys.*, 1993, **97**, 10112–10118.
22 **Perry, M.** and **Raff, L.** Theoretical studies of elementary chemisorption reactions on an activated diamond ledge surface. *J. Phys. Chem.*, 1994, **98**, 4375.
23 **Perry, M.** and **Raff, L.** Theoretical studies of elementary chemisorption reactions on an activated diamond (111) terrace. *J. Phys. Chem.*, 1994, **98**, 8128–8133.
24 **Chang, X., Thompson, D.,** and **Raff, L.** Hydrogen-atom migration on a diamond (111) surface. *J. Chem. Phys.*, 1994, **100**, 1765–1766.
25 **Stillinger, F. H.** and **Weber, T. A.** Computer simulation of local order in condensed phases of silicon. *Phys. Rev. B*, 1985, **31**(8), 5262–5271.
26 **Belak, J.** and **Stowers, I. F.** A molecular dynamics model of the orthogonal cutting process. *Proc. ASPE Annual Conference*, New York, 1990, p. 76.
27 **Belak, J., Lucca, D. A., Komanduri, R., Rhorer, R. L., Moriwaki, T., Okuda, K., Ikawa, N., Shimada, S., Tanaka, H., Dow, T. A., Drescher, J. D.,** and **Stowers, I. F.** Molecular dynamics simulation of the chip forming process in single crystal copper and comparison with experimental data. *Proc. of the ASPE Annual Conference*, New York, 1991, pp. 100–103.
28 **Shimada, S., Ikawa, N., Tanaka, H., Ohmori, G., Uchikoshi, J.,** and **Yoshinaga, H.** Feasibility study on ultimate accuracy in microcutting using molecular dynamics simulation. *Ann. CIRP*, 1993, **42**(1), 91–94.
29 **Ikawa, N., Shimada, S., Tanaka, H.,** and **Ohmori, G.** An atomistic analysis of nanometric chip removal as affected by tool–work interaction in diamond turning. *Ann. CIRP*, 1991, **40**(1), 551–554.
30 **Shimada, S., Ikawa, N., Ohmori, G.,** and **Tanaka, H.** Molecular dynamics analysis as compared with experimental results of micromachining. *Ann. CIRP*, 1992, **41**(1), 117–120.
31 **Shimada, S., Ikawa, N., Tanaka, H.,** and **Uchikoshi, J.** Structure of micromachined surface simulated by molecular dynamics analysis. *Ann. CIRP*, 1994, **43**(1), 51–54.
32 **Inamura, T., Suzuki, H.,** and **Takezawa, N.** Cutting experiments in a computer using atomic models of a copper crystal and a diamond tool. *Int. J. Jap. Soc. Precis. Engng*, 1991, **25**(4), 259–266.
33 **Inamura, T., Takezawa, N.,** and **Taniguchi, N.** Atomic-scale cutting in a computer using crystal models of copper and diamond. *Ann. CIRP*, 1992, **41**(1), 121–124.
34 **Inamura, T., Takezawa, N.,** and **Kumaki, Y.** Mechanics and energy dissipation in nanoscale cutting. *Ann. CIRP*, 1993, **42**(1), 79–82.
35 **Inamura, T., Takezawa, N., Kumaki, Y.,** and **Sata, T.** On a possible mechanism of shear deformation in nano-scale cutting. *Ann. CIRP*, 1994, **43**(1), 47–50.
36 **Chandrasekaran, N., Noori-Khajavi, A., Raff, L. M.,** and **Komanduri, R.** A new method for MD simulation of nanometric cutting. *Phil. Mag.*, 1997, **77**(1), 7–26.

37 **Binder, K.** and **Heermann, D. W.** *Monte Carlo Simulation in Statistical Physics – An Introduction*, third edition, 1997 (Springer Verlag, Berlin).

38 **Hammersley, J. M.** and **Handscomb, D. C.** *Monte Carlo Methods*, 1983 (Chapman and Hall, London).

39 **Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H.,** and **Teller, E.** Equation of state calculations by fast computing machines. *J. Chem.l Phy.*, 1953, **21**, 1087–1092.

40 **Rubinstein, R.** *Simulation and the Monte Carlo Method*, 1981 (John Wiley & Sons, New York).

41 **Heermann, D. W.** *Computer Simulation Methods in Theoretical Physics*, 1986 (Springer-Verlag, Berlin).

42 **Arfken, G.** The methods of steepest descents. *Mathematical Methods for Physicists*, third ed., 1985, pp. 428–436 (Academic Press, Orlando).

43 **Morse, P. M.** and **Feshbach, H.** Asymptotic series: method of steepest descent. *Methods of Theoretical Physics, Part I*, 1953, pp. 434–443 (McGraw-Hill, New York).

44 **Reiter, C. A.** Visualizing steepest descent. *The Visual Computer*, 1991, **8**(1), 64–67.

HTML ABSTRACT + LINKS

THE JOURNAL OF CHEMICAL PHYSICS 122, 084104 (2005)

Ab initio potential-energy surfaces for complex, multichannel systems using modified novelty sampling and feedforward neural networks

L. M. Raff
Department of Chemistry, Oklahoma State University, Stillwater, Oklahoma 74078

M. Malshe
Department of Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, Oklahoma 74078

M. Hagan
Department of Electrical and Computer Engineering, Oklahoma State University, Stillwater, Oklahoma 74078

D. I. Doughan and M. G. Rockley
Department of Chemistry, Oklahoma State University, Stillwater, Oklahoma 74078

R. Komanduri
Department of Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, Oklahoma 74078

(Received 3 September 2004; accepted 30 November 2004; published online 15 February 2005)

A neural network/trajectory approach is presented for the development of accurate potential-energy hypersurfaces that can be utilized to conduct *ab initio* molecular dynamics (AIMD) and Monte Carlo studies of gas-phase chemical reactions, nanometric cutting, and nanotribology, and of a variety of mechanical properties of importance in potential microelectromechanical systems applications. The method is sufficiently robust that it can be applied to a wide range of polyatomic systems. The overall method integrates *ab initio* electronic structure calculations with importance sampling techniques that permit the critical regions of configuration space to be determined. The computed *ab initio* energies and gradients are then accurately interpolated using neural networks (NN) rather than arbitrary parametrized analytical functional forms, moving interpolation or least-squares methods. The sampling method involves a tight integration of molecular dynamics calculations with neural networks that employ early stopping and regularization procedures to improve network performance and test for convergence. The procedure can be initiated using an empirical potential surface or direct dynamics. The accuracy and interpolation power of the method has been tested for two cases, the global potential surface for vinyl bromide undergoing unimolecular decomposition via four different reaction channels and nanometric cutting of silicon. The results show that the sampling methods permit the important regions of configuration space to be easily and rapidly identified, that convergence of the NN fit to the *ab initio* electronic structure database can be easily monitored, and that the interpolation accuracy of the NN fits is excellent, even for systems involving five atoms or more. The method permits a substantial computational speed and accuracy advantage over existing methods, is robust, and relatively easy to implement.
© *2005 American Institute of Physics.* [DOI: 10.1063/1.1850458]

I. INTRODUCTION

Molecular dynamics and Monte Carlo simulations are the most powerful existing methods for the investigation of dynamical behavior of atomic and molecular motions of complex systems. To date, such studies have been used to study chemical reaction mechanisms, energy transfer pathways, reaction rates, and product yields in a wide array of polyatomic systems. In addition, molecular dynamics (MD) and Monte Carlo (MC) methods have been successfully applied to the investigation of gas-surface reactions, diffusion on surfaces and in the bulk, membrane transport, the synthesis of diamond using chemical vapor deposition techniques, nanometric cutting, scratching, and grinding, as well as uniaxial tensile properties. The structure of vapor deposited rare gas matrices has been studied using trajectories procedures.

Such studies are becoming increasingly important for the investigation of microelectromechanical systems (MEMS) devices such as sensors, actuators, and structural members for biological, medical, and consumer products.[1] While mechanical properties for most materials at bulk or macro level are well documented, the corresponding properties at microlevels and nanolevels are not easy to obtain. In addition, the relationship between nanoproperties and microproperties is not well known in many cases. In order to design and fabricate MEMS devices, the design engineer must know the

 © 2005 American Institute of Physics

Downloaded 10 Mar 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

material properties at these levels. Consequently, atomic simulation methods are becoming a necessity.

With the advent of relatively inexpensive, powerful workstations, molecular dynamics simulations have become routine. Once the potential-energy hypersurface for the system has been obtained, the MD or MC computations are straightforward. In the majority of cases, the computation time required is on the order of hours to a few days. However, the accuracy of these simulations depends critically on the accuracy of the potential hypersurface used. In this paper, we present an integrated approach for the determination of accurate potentials from *ab initio* quantum mechanical calculations using novelty sampling procedures and a feedforward neural network approach. The method is ideally suited for addressing systems containing five atoms or more which exhibit multiple open reaction channels.

II. PREVIOUS APPROACHES TO THE POTENTIAL ENERGY SURFACE PROBLEM

The major problem associated with MD/MC investigations is the development of a potential-energy surface whose topographical features are sufficiently close to those of the true, but unknown, surface that the results of the calculations are experimentally meaningful. Once the potential surface is chosen or computed, all the results from any quantum mechanical, semiclassical, or classical scattering or equilibrium calculation are determined. The only purpose of the dynamics calculations is to ascertain what these results are. Therefore, the most critical part of any MD or MC study is the development of the potential-energy hypersurface and associated force field. Surprisingly, this is often the portion of the investigation to which the least effort is devoted. This situation arises because of the inherent difficulty associated with this part of the overall problem.

A. Empirical and semiempirical potential surfaces

When four or more many-electron atoms are present, global potentials have usually been obtained using empirical methods which rely upon *ad hoc* parametrized functional forms suggested by physical and chemical considerations. The parameters contained in these forms are generally fitted to equilibrium thermochemical, spectroscopic, and structural data. If experimental activation energies are known, the barrier heights predicted by the empirical potential are adjusted to these values. Such methods have been employed by numerous investigators since the mid 1970s. For example, Sewell and Thompson[2] used this method to study C–Cl and C–H bond scission reactions in activated two-chloroethyl radicals. Similar calculations have been reported by Marks[3] who studied $-N{=}C{\rightarrow}-C{\equiv}N$ isomerization in HD_2CNC. Rice and Thompson[4] have examined the unimolecular decomposition reactions of CH_3NO_2 to yield CH_3+NO_2 and CH_3O+NO by writing the potential surface as a summation of channel potentials in which each term represents one of the asymptotic limits CH_3NO_2, CH_3+NO_2, or CH_3O+NO. Alimi, Apkarian, and Gerber[5] have used empirical methods to obtain the potential surface for their studies of pressure effects of Cl_2 dissociation in Xe matrices. Lendray and Schatz[6,7] fitted spectroscopic data to study energy transfer in CS_2+X ($X=H_2,CO,HCl,CS_2,CH_4$) and SF_6+X ($X=He,Ar,Xe,CO,SF_6$) systems, respectively. Budenholzer, Chang, and Huang[8] have investigated the dissociation and intramolecular vibrational relaxation of dimethylperoxide with three potential surfaces developed with these techniques. We employed empirical potentials to conduct extensive calculations on the gas-phase and matrix-isolated reaction dynamics for the unimolecular decomposition of 1,2-difluoroethane and the bimolecular reaction of F_2 with ethylene.[9–15]

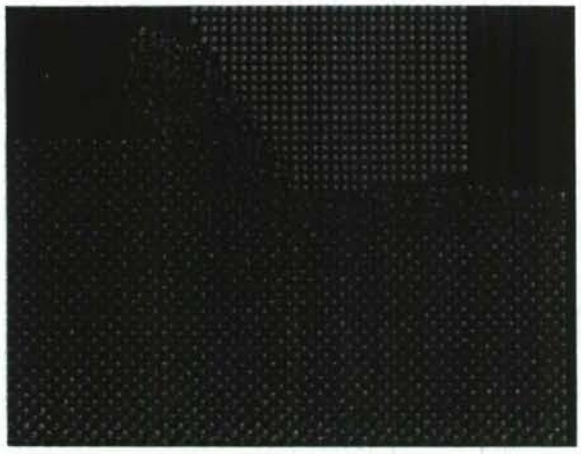


FIG. 1. Snapshot during the MD simulation of nanometric cutting of silicon.

All machining, indentation, uniaxial tension, and tribological studies to date have relied on empirical potentials[16–36] such as pairwise sums of Morse[18] or Lennard-Jones[19] potentials for metals. Baskes and his colleagues have developed an embedded-atom method (EAM)[32] and modified embedded-atom methods (Refs. 33–35) for application to fcc and bcc metals, and even to hcp metals. For covalently bonded, semiconductor materials, such as Si, Ge, and diamond, many-body potentials, such as Tersoff's potential,[36] the Bolding–Anderson[31] potential, and the Brenner[30] potential have been developed. These potentials provide approximate descriptions of most materials at or near their equilibrium configuration. However, when the lattice atoms are in configurations far removed from equilibrium, the accuracy of any empirical potential can be expected to decrease significantly.

A typical example can be seen in the simulation of nanometric cutting of silicon. In this case, large energy input during cutting and extensive plastic deformation ahead of the tool often produces a variety of amorphous clusters that are far from equilibrium. Examples of such configurations are seen in Fig. 1 both beneath and directly in front of the cutting tool as well as in the chip. Since the parameters contained in empirical potentials are adjusted using primarily equilibrium data, it is very unlikely that the energies and gradients for these amorphous structures are accurately represented by the potential surface. A similar situation exists in other plastic deformation processes, such as nanotension and nanoinden-

Downloaded 10 Mar 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

tation. It also occurs in many chemical reactions that sample regions of configuration space that are far removed from equilibrium.

The limitations and liabilities of the use of empirical potential surfaces in MD calculations may be summarized as follows.

(1) Since the *ad hoc* empirical functional form is simply concocted with little or no theoretical foundation, it is highly unlikely that it will accurately predict the correct experimental force field.

(2) The parameters contained in an empirical potential are generally adjusted to fit equilibrium and stationary-point data, such as bond lengths, bond angles, vibrational frequencies, reactant and product equilibrium energies, elastic moduli, sublimation energies, and potential barrier heights. As a result, the potential usually fits the stationary points on the surface acceptably well. However, MD and MC simulations are primarily concerned with the energies and forces at nonstationary points. It is very unlikely that these quantities are accurately represented in configurations far from equilibrium.

(3) If the database contains data from *ab initio* quantum mechanical calculations at nonstationary points as well as experimental data, the problem of fitting the empirical parameters becomes very difficult. Such fitting must be accomplished using iterative adjustment methods that often require months of effort. For example, nine months of laborious iterative fitting was required to obtain an acceptable fit of the 100 parameters contained in the empirical function used to fit the global potential surface for vinyl bromide.[39]

(4) Once an empirical potential is developed for a particular complex system, there is no straightforward means by which it can be improved. Any alteration of the potential requires that the entire fitting process be repeated.

(5) Empirical potentials are specific for the system under consideration. The procedures used to obtain such potentials cannot be easily automated and the methods applied to all systems. Every system must be treated individually. Therefore, the research is tedious and labor intensive.

Each of the above problems is essentially intractable. Because of this, the accuracy of molecular dynamics simulations on complex systems has, for practical purposes, reached its limit. If we wish to advance beyond this natural limit for empirical and hybrid potentials, we must develop ways to effectively utilize quantum mechanical methods that have the power to produce more accurate surfaces and force fields.

B. *Ab initio* potential-energy surfaces

In principle, *ab initio* methods can produce much more accurate potential surfaces for molecular dynamics. The fundamental problem is that electronic structure calculations become computationally intractable when the number of atoms and electrons present in the system becomes large. Nevertheless, excellent results have been obtained for numerous three- and four-body systems. In this section, we present a brief review of some of systems studied and the methods employed to obtained the required *ab initio* potentials.

In the case of simple, three-body reactions, such as $H+D_2 \rightarrow HD+D$ or $H_2+F \rightarrow HF+H$, accurate *ab initio* potential-energy surfaces can be obtained with a minimum of difficulty.[38] The total configuration space is sufficiently small that electronic structure calculations of the energies and gradients can be carried out in all regions. Since only three internal coordinates are required, interpolation between the computed points presents no formidable obstacles. For example, this approach has been employed by Bodo *et al.*[40] to obtain potential surfaces for the reactions

$$LiH(X\,^1\Sigma^+) + H(^2S) \rightarrow Li(^2S) + H_2(X\,^1\Sigma^+)$$

and

$$LiH^+(X\,^2\Sigma^+) + H(^2S) \rightarrow Li(^1S) + H_2(X\,^1\Sigma^+).$$

The *ab initio* energies were computed for 11 341 different geometries using configuration interaction methods. These data were then fitted using Aguado-Paniagua (AG) type functions and a simple multipolar expansion at long range with switching functions to smoothly join the two forms.

When the system of interest contains four atoms, it is usually no longer possible to conduct electronic structure calculations in all regions of configuration space. Consequently, importance sampling procedures must be employed to identify the critical regions of configuration space. In addition, interpolation between the computed points on the surface becomes increasingly difficult.

Ischtwan and Collins[40] have developed a moving interpolation technique in which the potential energy in the neighborhood of any point is approximated by a Taylor series using inverse bond length coordinates as the expansion variables. Initially, a set of system configurations along the reaction path is selected using chemical intuition. *Ab initio* electronic structure calculations of the system energy and gradients in these configurations are then executed. The data thus produced are employed to obtain the set of initial Taylor series expansions. A Shepard method[41,42] is used to effect the required interpolations. This procedure expresses the potential at any configuration as a weighted average of the Taylor series about all N points in the data set, or, alternatively, those points within a specified cutoff radius. Subsequently, the results are iteratively improved by computing trajectories on the Taylor series fitted surface and recording the internal coordinates at a series of successive configurations encountered in the trajectories. These new points are added to the data set according to a weight factor that is determined by the relative density of points in the data set in the region of the new points. The weight is large if the configuration is close to a large number of trajectory points which are themselves not close to the current points in the data set. Ischtwan and Collins[40] applied the method to the $NH+H_2 \rightarrow NH_2+H$ reaction. Comparison of the fitted surface to the exact *ab initio* energies at a set of new points showed that the average absolute error in the potential after three iterations was 12.13 kJ mol^{-1}. The average relative errors in the surface gradient at this point was 56%.

Jordan, Thompson, and Collins[43] used the moving interpolation-trajectory sampling method to investigate the dynamics of the $OH+H_2 \rightarrow H_2O+H$ reaction. The criterion

Downloaded 10 Mar 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

used to assess convergence of the iteratively improved potential to the final surface was some computed dynamic variable such as the distribution of product angular momentum, distribution of product relative translational energy, or reaction probability.

Numerous four-atom reactions have been investigated using the moving interpolation-trajectory method. In some of these studies, efficient microcanonical sampling[44,45] and Monte Carlo methods are employed to replace the arbitrary choice of initial configurations along the reaction coordinate. Collins and Bettens have studied hydrogen-atom abstraction[46] in BeH_2, $BeH_2+H \rightarrow BeH+H_2$, and the reaction of atomic nitrogen with H_3^+,[47] $N(^4S)+H_3^+(^1A_1') \rightarrow NH_2^+ + H$, using these methods. The $OH+H_2 \rightarrow H_2O+H$ reaction has been investigated by Thompson and Collins.[48] Bettens, Hansen, and Collins[49] have examined the reaction $H_3^+ + O \rightarrow H_2O^+ + H$ and Bettens and Collins[50] have reported results for the $C(^3P)+H_3^+$ system.

The moving interpolation-trajectory method with Shepard interpolation has also been employed in studies of quantum reaction dynamics. Collins and Zhang[51] reported such a study on the reaction $BeH+H_2 \rightarrow BeH_2+H$. Nguyen, Rossi, and Truhlar[52] used a dual level approach that combined a small number of high-level electronic structure calculations with a lower-level neglect-of-diatomic-differential-overlap (NDDO) global surface obtained with a modified Shepard interpolation[41,42] in an effort to determine the extent to which useful dynamics information can be obtained with a minimum of effort devoted to electronic structure calculations. Bettens and Collins[53] have reported a study of the accuracy of the potential surfaces obtained using this method for nine four-body systems that include BeH_3, CH_3^+, NH_3^+, OH_3^+, $NeCOH^+$, $ArCOH^+$, and $KrCOH^+$. For a review of these methods, see Ref. 54.

Other methods for obtaining *ab initio* potential surfaces for three- and four-body systems have been suggested. One of these employs the criterion of reproducing kernel Hilbert space (RKHS) as the means to effect the required interpolation between the computed *ab initio* energies. Ho *et al.*[55] have employed this procedure to investigate the $O(^1D)+H_2 \rightarrow OH+H$ reaction. When 1280 *ab initio* data points were employed to define the surface, the interpolation yielded a surface whose absolute rms error compared to *ab initio* energies was 1.26 kJ mol^{-1}. Pederson *et al.*[56] have used the same method to study the dynamics of the $N(^2D)+H_2$ reaction.

Maisuradze *et al.*[57] have introduced an interpolating moving least squares (IMLS) method to effect the interpolation between the computed *ab initio* points. When the method is unrestricted, the least-squares coefficients are obtained from the solution of a large matrix equation that must be solved repeatedly during a trajectory study. In this form, the computational time required increases with NM^2, where N is the number of data points to be fitted and M is the number of bases functions used in the linear combination that provides the fit. The method was initially tested for a one-dimensional Morse oscillator and a one-dimensional slice of the HN_2 surface.[57] In later work, Guo *et al.*[58] evaluated the IMLS method using the analytic surface of Kuhn *et al.*[59] for simple O–O bond rupture of HOOH. With $N=300$, the results showed the interpolation errors to lie in the range 8.2–16.5 kJ mol^{-1}. Kawano *et al.*[60] have compared the accuracy of the IMLS method with that of the modified Shepard procedure for the HOOH bond scission reaction. With $N=6489$, the rms errors for IMLS and modified Shepard were 4.52 and 1.95 kJ mol^{-1}, respectively.

TABLE I. Comparison of interpolation accuracy of *ab initio* energies obtained using various methods for three- and four-body systems.

System	N[a]	Method	rms or absolute average energy error (kJ mol^{-1})	Reference
OH_2	1,280	RKHS	1.26	55
H_3	550	MSI[b]	0.154	61
$C+H_3^+$	406	MSI[b]	0.6	50
$H_3^+ + O$	1,000	MSI[b]	0.5	49
$BeH+H_2$	1,300	MSI[b]	0.1%–0.4%	51
$OH+H_2$	31	MSI[b]	34.2	52
BeH_2+H	438	MSI[b]	0.35	46
$OH+H_2$	400	MSI[b]	1.0	48
$NH+H_2$	600	MSI[b]	12.1	40
$N+H_3^+$	699	MSI[b]	0.05	47
BeH_3	500–1000	MSI[b]	0.03–2.39	53
CH_3^+	500–1000	MSI[b]	0.03–2.39	53
NH_3^+	500–1000	MSI[b]	0.03–2.39	53
OH_3^+	500–1000	MSI[b]	0.03–2.39	53
OH_3	500–1000	MSI[b]	0.03–2.39	53
$NeCOH^+$	500–1000	MSI[b]	0.03–2.39	53
$ArCOH^+$	500–1000	MSI[b]	0.03–2.39	53
$KrCOH^+$	500–1000	MSI[b]	0.03–2.39	53
HOOH	6489	IMLS[c]	4.52	60
HOOH	300	IMLS[c]	8.24–15.6	58
HOOH	6489	MSI[b]	1.95	60

[a]N is the number of *ab initio* points computed.
[b]MSI is the modified Shepard interpolation procedure.
[c]IMLS is the interpolating moving least squares method.

Table I provides a convenient comparison of the interpolation accuracy obtained with the various methods that have been employed for three- and four-body systems.

When the number of atoms in the system becomes five or more, the difficulty of the problem increases significantly. The dimensionality of the system increases, and as a result, the size of the configuration space over which interpolation must be executed is much larger. If more than one reaction channel is open, these difficulties become even greater. Thompson, Jordan, and Collins[62] have investigated hydrogen-atom abstraction in the CH_4+H system. When the configuration sampling is done by initiating trajectories in the $H+CH_4$ region of configuration space, the interpolation errors produced by the Shepard moving method are about 1.6 kJ mol^{-1}. This increases to 3.5 kJ mol^{-1} if the trajectories are initiated in the H_2+CH_3 region of the surface. The associated errors in the relative gradients are reported to be about 8% and 12%–13%, respectively.

Similar investigations reported by Collins and Radom[63] for the exchange reaction of HF with HOC^+ gave interpolation errors of 1.5 kJ mol^{-1} when 1000 *ab initio* points were included in the database. When the size of the system was increased to six atoms for study of the exchange reaction H_2O+HOC^+, the rms error increased to 3.5 kJ mol^{-1}. When

Downloaded 10 Mar 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

TABLE II. Interpolation accuracy of *ab initio* energies obtained using the modified Shepard interpolation (MSI) for five- and six-body systems.

System	N^a	rms or absolute average energy error (kJ mol^{-1})	Relative % error in gradient	Reference
$H+CH_4$	1250	1.6 (reactant valley sampling)	8%	62
$H+CH_4$	1250	3.5 (product valley sampling)	12%–13%	62
$HF+HOC^+$	1357	1.5	Not reported	63
H_2O+HOC^+	3047	3.5	Not reported	63
$H_3^+ + HD$	728	0.17	Not reported	64

[a]N is the number of *ab initio* points computed.

a modified Shepard interpolation was employed to study the reactions

$$H_3^+ + HD \rightarrow H_2D^+ + H_2$$

and

$$H_3^+ + D_2 \rightarrow HD_2^+ + HD,$$

the average absolute interpolation error was 0.34 kJ mol^{-1}. Crepos *et al.*[65] have employed the Shepard interpolation method to study surface reactions of H_2 on Pt(111). Table II summarizes the results for five- and six-atom systems.

In order to avoid the extremely difficult task of interpolating *ab initio* electronic structure data, some investigators have employed a method usually called "direct dynamics." In this approach, trajectories are computed by direct calculation of the force field at each integration point using some *ab initio* quantum mechanical method. Because of the huge number of computations required for each trajectory, the *ab initio* method chosen must usually be some form of density functional theory. For example, it may be necessary to follow the dynamics for several picoseconds. With an integration step size on the order of a femtosecond, 1000 or more integration steps will be required. If the forces are computed at four points in each step, something on the order of 4000 gradients will have to be computed for each trajectory. As a result, only a very limited number of trajectories can be obtained. As the system under investigation increases in size, the computational requirements will quickly overwhelm the available computational resources. The basic problem is that all the information obtained about the potential-energy surface and the corresponding force field during the integration of the trajectory is discarded after the completion of the calculation. Therefore, each subsequent trajectory fails to profit from all the computational effort expended in obtaining the previous trajectories.

The conventional wisdom is that *ab initio* molecular dynamics calculations for complex systems containing five or more atoms with several open reaction channels are presently beyond our computational capabilities. The rationales for this view are (a) the inherent difficulty of high-level *ab initio* quantum calculations on complex systems that may make numerous, large-scale computations impossible, (b) the large dimensionality of the configuration space for such systems that makes it necessary to examine prohibitively large numbers of nuclear conformations, and (c) the extreme difficulty associated with accurate interpolation of numerical data obtained from electronic structure calculations when the dimensionality of the system is nine or greater. In this paper, we present general methods to handle each of these difficulties for many complex systems. Furthermore, as our computational resources increase, the range of systems that can be so treated will expand without the need to significantly modify the methods. Therefore, the theoretical approach is robust and powerful.

III. THE NEURAL NETWORK APPROACH

In this paper, we present an integrated approach for obtaining *ab initio* quantum mechanical potential-energy surfaces and force fields for use in molecular dynamics simulations for systems of five or more atoms in which several reaction channels may be energetically open. The method involves (1) electronic structure calculations of energies and force fields for an ensemble of atoms whose interatomic distances are such that all atoms fall within a previously specified distance from some central point, generally called the cutoff radius, (2) the implementation of sampling methods that utilize MD calculations and novelty-sampling to permit the regions of configuration space that are important in the dynamics to be accurately identified, and (3) the use of neural networks (NN), early stopping, and regularization methods to provide rapid and accurate interpolation of the *ab initio* database and convenient tests for convergence that do not require MD calculation of dynamical results.

A. Electronic structure calculations

The solution to this portion of the problem is already available in the form of several software programs that will execute different levels of electronic structure calculations. The GAUSSIAN (Ref. 66) suite of programs is an example. With such software, it is possible to tailor the computational accuracy to be in accord with the complexity of the system under study and the extent of computational resources available to the user. Extremely complex systems might be treated using density functional theory with a smaller basis set. Moller–Plesset perturbation theory at an appropriate level (MP2, MP3, MP4, MP5) can be used with larger basis sets for systems whose complexity is not as great. For even smaller systems, configuration interaction or multireferenced methods can be employed. As our computational resources increase, appropriate adjustments can be made in the type of electronic structure calculation employed. This flexibility makes the overall procedure very robust.

The calculations reported in this paper utilize the Linux version of GAUSSIAN98 and a multiprocessor containing 26 CPU nodes each with a clock speed of 1.7 GHz. Our results show that with only one node in the CPU cluster, we can compute the potential and associated forces for one configuration of a five-atom silicon cluster at the MP4(SDQ) level of accuracy using a 6-31G** basis set in about 15 min. A seven-atom silicon cluster containing 98 electrons requires about 60 min. Smaller clusters require significantly less computational time since the time scales approximately with N^4, where N is the number of electrons present. A MP4(SDQ)

Downloaded 10 Mar 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

calculation of the potential and associated force field for one configuration of vinyl bromide (C_2H_3Br) takes about 6.5 min of CPU time. If only the potential is computed, the required time decreases to 1.3 min. These calculations employ a 6-31G(d,p) basis set for the carbon and hydrogen atoms and Huzinaga's (4333/433/4) basis set augmented with split outer s and p (43321/4321/4) orbitals for the bromine atom. A polarization f orbital is also added with an exponent of 0.5 to provide a more accurate description. Consequently, we are able to compute the energies and forces for 96 five-atom silicon clusters per day per node. With 26 CPU units, each with its own dedicated RAM and hard drive, the energies and forces for about 2500 clusters can be computed per day. For a smaller system, such as vinyl bromide, about 5760 *ab initio* energies and associated force fields can be obtained per day. Thus, the electronic structure calculations for 10^5 configurations can be completed in about 18 days for vinyl bromide and about 40 days for five-atom silicon clusters.

As the clock speeds for computers increase (machines operating at over 3.0 GHz are already available), the computational times required will decrease correspondingly. A 52-node cluster with CPU clock speeds of 3.4 GHz will be able to complete the computation of the vinyl bromide and five-atom silicon cluster potentials and force fields in about 4.5 and 10 days, respectively. Relative to the time generally required for the execution of a research project, such numbers are small.

If we are concerned with a gas-phase chemical reaction, the number of atoms and electrons that must be included in the electronic structure calculations is usually equal to the total number in the system of interest. For machining experiments, the situation is not so clear. The computational power required now depends upon the nature of the clusters that occur in a machining experiment and the cutoff radius used in the calculations. For example, consider the machining of silicon. If we are using a Tersoff potential[36] with a 3.0 Å cutoff radius, MD calculations show that we observe clusters containing from two to seven atoms within this radius but none with more than seven atoms. Therefore, the electronic structure calculations must be able to handle seven-atom clusters or less. If we are machining a fcc metal such as aluminum, the cluster sizes are much larger. Therefore, unless the available computational power is correspondingly increased, it will be necessary to substitute something like density functional calculations for the MP4(SDQ) methods discussed above.

B. Sampling procedures

Although we can compute MP4(SDQ) energies and forces for thousands of configurations per day (and about 10^5 per month) for systems containing between 50 and 100 electrons, this number is still very small compared to the totality of the configuration space available to the system. For example, consider the three-body LiH_2 system whose configurations are determined by three internal coordinates. If reasonable ranges are taken for each coordinate and each range divided into N equal segments, the energies and force fields for N^3 configurations would need to be computed to characterize the system. Bodo, Martinazzo, and Yianturco[40] used 11 341 configuration points for this purpose. This corresponds to $N \approx 23$. While such brute force calculations can be executed for a three-atom system containing only five electrons, larger systems rapidly move completely out of range. The configurations of the 50-electron, six-atom vinyl bromide system require the specification of 12 internal coordinates. Thus, a complete characterization would require about 23^{12} or 2.19×10^{16} points. If we have the computational power to compute the energies for 28 000 configurations per day, as suggested by our results, it will still require about 2.14×10^9 years to obtain the energies for each of these configurations. Obviously, this is far beyond our reach.

If all of the 10^{16} configurations of vinyl bromide are important in the MD calculations, *ab initio* molecular dynamics (AIMD) calculations are indeed beyond our present capabilities and probably beyond those that will exist for decades or centuries to come. However, simple chemical logic tells us that most of the 10^{16} configurations are of no interest in the dynamical studies. For example, we know that configurations such as H–H–H–C–Br–C are of no interest. In any system, the number of important configurations forms a small subset of the total configuration space. Therefore, the key to success is to devise an importance sampling method that adequately samples this subset rather than the entire configuration space.

As soon as we direct our attention to the sampling problem, it becomes evident that molecular dynamics does precisely what we require. Such methods were first introduced by Collins and co-workers.[54] The nuclear configurations that are generated in an ensemble of trajectories comprise the subset of configuration space that we wish to sample. All we need to do is to devise efficient procedures for (a) the incorporation of configuration points into the database, (b) interpolation between the points, and (c) testing for convergence of the potential surface that are well suited to use for complex systems involving multiple reaction channels. The data listed in Table II suggest that the accuracy of moving interpolation methods is decreasing rapidly as the dimensionality of the system increases. This is particularly the case for the surface gradients, which are of paramount importance in dynamics simulations. In addition, Monte Carlo methods or methods based on chemical intuition for initiating the process are much less likely to work well in complex systems. We also need convergence tests that are independent of the MD results.

The method we employ follows that used by Collins and co-workers.[54] We sample the important regions of configuration space in iterative fashion using trajectories. However, the present procedure initiates the process by employing an empirical potential for the system of interest[9–29,40,39] rather than using Monte Carlo methods, microcanonical sampling,[44,45] or chemical intuition. This is a much more effective starting procedure since it is unlikely that random sampling or intuition will provide an initial sampling of the important regions of configuration space in complex systems with many open reaction channels.

One of the advantages of the procedure described in the current work is that the iteration is not sensitive to the choice

Downloaded 10 Mar 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

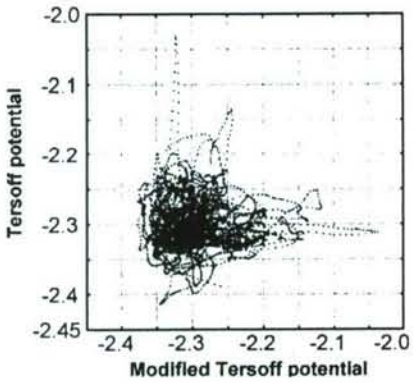


FIG. 2. Energies (eV) computed using two different Tersoff-type potentials for a set of several thousand Si_5 configurations generated using MD methods in a cutting simulation on a silicon workpiece. If the two potentials gave the same energies for a given configuration, all points would lie along the 45° line.

of the empirical surface employed in the initial step. So long as the choice is reasonable, the iterative procedure will converge. As an illustration, we have employed two potentials proposed by Tersoff[36] for the initial step in the investigation of the potential surface for five-atom silicon clusters. These two potentials have different functional forms and different parameters. Figure 2 shows a plot of the energies computed using both potentials for a set of several thousand Si_5 configurations generated using MD methods on a silicon workpiece in nanometric cutting similar to that shown in Fig. 1. If the two potentials gave the same energies for a given configuration, all points would lie along a 45° line. Obviously, the two empirical potentials predict very different energies. Nevertheless, when neural nets are fitted to each set of *ab initio* energies obtained from the configurations generated by MD calculations using the two empirical potentials, Fig. 3 shows that the predictions of the two nets for an independent testing set are in excellent accord.

It should also be noted that the need to have a reasonable empirical potential can be obviated by using direct dynamics as the starting procedure. In this case, the interative procedure is initiated by computing several trajectories using direct dynamics and storing the system configurations at equally spaced intervals of time. The energies and gradients at these points obtained in the direct dynamics calculations can then be used without modification to begin the procedure or, if desired, higher-level electronic structure calculations can be executed on the stored configurations. In either case, the method now becomes self-starting. In addition, the combination of direct dynamics with neural networks provides a procedure that makes direct dynamics more powerful. This combination permits future trajectory calculations to profit from the computational effort expended in the direct integration of previous trajectories.

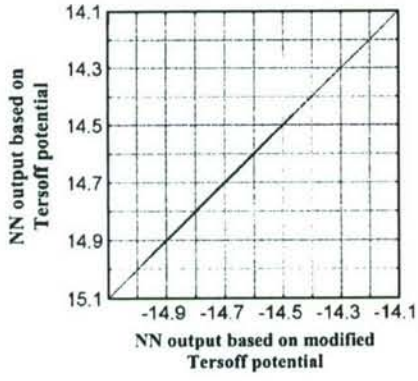


FIG. 3. Predictions of two different neural nets fitted to *ab initio* energies for Si_5 configurations generated by MD calculations using the two potentials whose results are shown in Fig. 2. The plotted points are for an independent testing set. If the two NN produce the same interpolated energies, all results will lie along the 45° line. All energies are in eV.

A set of trajectories using this empirical potential is now computed using standard MD methods.[67] As the trajectories proceed, we store the nuclear configurations at equally spaced time intervals. This is continued until we have several thousand such configurations. The energies and forces present in these configurations are then computed using an appropriately chosen *ab initio* method. This data set comprises our approximation for the *ab initio* system potential and force field.

In the second step, feedforward NN are fitted to the *ab initio* database for the energies and forces obtained in the first step. The fitting procedure is described in Sec. III C. A second set of MD trajectories is now computed using the NN force field rather than the original empirical potential. During these calculations, additional nuclear configurations are stored and their energies and forces subsequently computed. As explained below, novelty sampling algorithms based on the current density of configuration points in the database guide our selection of configurations. Regions with a low density are preferentially included in our sample. The energies and forces obtained for the newly selected configurations are added to the overall database characterizing the potential-energy hypersurface and force field. New neural networks are trained to fit to this expanded database and the entire procedure repeated. Since the configurations obtained from the MD calculations involve structures in which all of the internal coordinates are simultaneously varying, the computed force fields will properly represent all the coupling terms that influence intramolecular energy transfer and reaction dynamics.

After several iterations, we expect the *ab initio* potential and force field obtained from the trained NN to converge to something close to the experimental potential and force field, at least to the extent that our choice of *ab initio* computational method is capable of yielding energies and forces that approach the experimental values. As described below, the novelty sampling itself provides a convenient test for convergence that is independent of results obtained from MD calculations.

To ensure that the NN accurately represents the potential and associated force field for the system, it is essential that the training data be spread throughout the subset of configuration space that is important in the dynamics. No interpolation method, including neural networks, can be expected to extrapolate accurately. While there is little that we can do during training to improve the network performance outside

Downloaded 10 Mar 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

the range of the training data, we can improve its ability to interpolate between data points, and we can work to ensure that the training data cover the relevant range of the input space.

To guide our selection of new configurations to be added to the database, we employ a modification of the method of novelty detection developed by Pukrittayakamee and Hagan.[68] Let $\mathbf{q_i}$ be a $[(3N-6)\times 1]$ column vector of the internal coordinates of the ith cluster in the data set currently being used to determine the NN force field. By computing trajectories using this force field, we generate a configuration $\mathbf{q_n}$. Qualitatively, we wish to incorporate the forces and energies of this configuration into the database if $\mathbf{q_n}$ lies in a region of configuration space where the density of points in the database is low. However, if this region already contains a high density of configuration points, we wish the probability of incorporating point $\mathbf{q_n}$ to be much lower.

Novelty detection is based on the two-norm difference between vector $\mathbf{q_n}$ and vector $\mathbf{q_i}$:

$$\|\mathbf{q_i}-\mathbf{q_n}\| = [(\mathbf{q_i}-\mathbf{q_n})^T(\mathbf{q_i}-\mathbf{q_n})]^{1/2}, \tag{1}$$

where $(\mathbf{q_i}-\mathbf{q_n})^T$ is the transpose of $(\mathbf{q_i}-\mathbf{q_n})$. A vector $\mathbf{d}$ is now defined whose elements are all the distances computed using Eq. (1) for each of the N configuration points in the database. The minimum distance separating point $\mathbf{q_n}$ from other points in the database is the minimum element of $\mathbf{d}$ that we denote as $d_m=\min(\mathbf{d})$. The mean separation distance, $\langle d\rangle$, is the average value of the elements of $\mathbf{d}$. The basic concept is to incorporate configuration point $\mathbf{q_n}$ into the database with high probability if d_m or $\langle d\rangle$ is large, but low probability if both d_m and $\langle d\rangle$ are small.

A selection algorithm based solely upon the magnitude of d_m will often result in the failure to incorporate configuration points that are needed in the database or the incorporation of points that are not critical. This occurs because a minimum distance criterion does not incorporate consideration of the gradient of the function being fitted. In regions where the gradient is large, many more points will be required to obtain an accurate NN fit than is the case in regions characterized by smaller gradients. Unfortunately, the underlying potential function is unknown. Therefore, it is impossible to calculate the gradient at an arbitrary configuration point without executing electronic structure calculations. Pukrittayakamee and Hagan[68] have found that if the minimum distance between a set of modified configuration vectors is used in place of d_m, the effect of the function gradient can be incorporated into the selection algorithm. We create the set of modified vectors Γ using

$$\Gamma_i = [\mathbf{q_i}O_i], \tag{2}$$

where O_i is the output from the NN at configuration point $\mathbf{q_i}$. When the minimum distance, Γ_m, between Γ_i and Γ_n for all N configuration points in the database is used in place of d_m, the effect of function gradient is incorporated. This occurs because if the gradient between points $\mathbf{q_i}$ and $\mathbf{q_n}$ is large, the corresponding NN outputs, O_i and O_n, will be very different and Γ_m will increase, as desired.

The first half of our sampling algorithm is based on the normalized probability distribution function of Γ_m values, $P(\Gamma_m)d\Gamma_m$, for all N configuration points in the current database. This distribution function and its maximum value, $P(\Gamma_m)_{\max}$, are computed at the beginning of each iterative cycle in the sampling procedure. As new configuration points are generated during the trajectories computed using the current NN force field, they are added to the database if

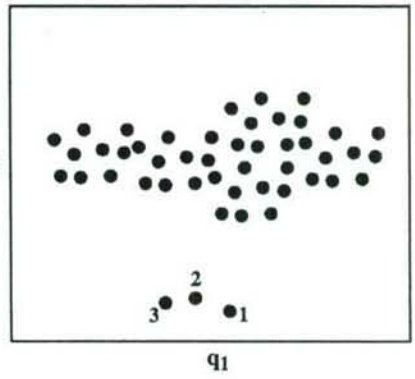


FIG. 4. Hypothetical distribution of configuration points in a data base for a system with two configuration variables, q_1 and q_2. The points labeled 1, 2, and 3 are outliers.

$$P(\Gamma_m) \leq T_1 P(\Gamma_m)_{\max}$$

and

$$\Gamma_m \geq \Gamma_m^*$$

or if (3)

$$\frac{P(\Gamma_m)}{P(\Gamma_m)_{\max}} \leq \xi(1-T_1).$$

Otherwise the new configuration point will be rejected by the first test. In Eq. (3), T_1 is an empirically determined, acceptance threshold in the range $(0\leq T_1\leq 1)$ and ξ is a random number whose distribution is uniform on the interval [0,1]. Γ_m^* is a small value of Γ_m chosen so as to exclude points that are very near other points in the database. Examination of Eq. (2) shows that the new point will be accepted if very few points in the database have Γ_m values near that for the new point. It will also be accepted with a probability that decreases as $P(\Gamma_m)$ approaches its maximum value $P(\Gamma_m)_{\max}$.

To be certain that our sampling does not exclude configuration points that are outliers, we must augment the first half of the sampling algorithm with an additional test that will include outliers. An outlier configuration point is usually defined as one that is inconsistent with the remainder of the database and one that does not follow the majority's underlying correlation.[69]

Suppose we have two configuration points in the database that are far removed from the remainder of the points. This situation is illustrated in Fig. 4 for the simple case in which the configuration depends only upon two coordinates, q_1 and q_2. The points denoted as 1, 2, and 3 are outliers. Now suppose one of the iterative cycles generates the configuration point denoted by Point 3 in Fig. 4. Since Γ_m for Point 3 is about the same as that for most of the members of the

Downloaded 10 Mar 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

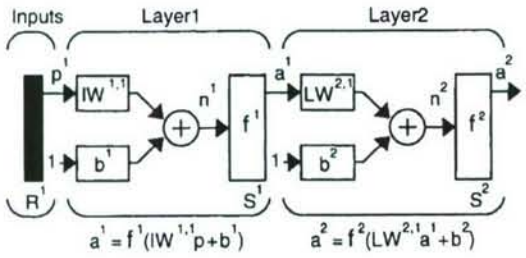


FIG. 5. Schematic of a two-layer neural network.

database, the first selection criterion would very likely result in rejection of this point. However, if we are to adequately sample configuration space, it is clear that Point 3 needs to be included in the database.

We can ensure that an outlier such as Point 3 will be incorporated into the database by using a second selection criterion. At the beginning of each iterative cycle, we compute the normalized probability distribution function of $\langle\Gamma\rangle$ values, $P(\langle\Gamma\rangle)d\langle\Gamma\rangle$, for all N configuration points in the current database. From this distribution, we obtain the maximum value, $P(\langle\Gamma\rangle)_{\max}$. If the first half of the selection algorithm results in rejection of a configuration point, we still accept the point if

$$P(\langle\Gamma\rangle) \leqslant T_2 P(\langle\Gamma\rangle)_{\max}$$

or if (4)

$$\frac{P(\langle\Gamma\rangle)}{P(\langle\Gamma\rangle)_{\max}} \leqslant \xi(1-T_2).$$

Proper selection of T_2 will result in the addition of most outlier configurations generated in the MD iterative calculations.

C. Feedforward neural network fitting

To make the above procedure a practical method for MD calculations, we must avoid compromising the accuracy of the *ab initio* database by concocting *ad hoc*, parametrized functional forms for fitting and interpolation of the potential and force field. Not only would such a procedure significantly degrade the accuracy of the potential, it would also be so tedious to execute that the overall method would become essentially useless.[70]

Our procedure is to employ multilayer, feedforward neural networks to fit the potential and the associated force field. These networks are then used in the MD calculations to provide the forces for the instantaneous configuration of the system. A detailed discussion of such networks can be found in Hagan *et al.*[68,71] and Haykin.[72]

Figure 5 is an example of a layered feedforward network (two layers in this case). For a full description of the notation, see Demuth and Beale.[73] The input vector to the network is represented by $\mathbf{p}^1$, which has R elements. The superscript represents the input number, since it is possible to have more than one input vector (although we use only one input vector in this research). In our case, the elements are the interparticle distances, three-body angles, and dihedral angles required to specify the configuration. The input is connected to Layer 1 through the input weight $\mathrm{IW}^{1,1}$, where the first superscript denotes the layer number and the second superscript represents the input number. The bias for the first layer is represented by b^1. The net input vector to Layer 1 is denoted by $\mathbf{n}^1$. It is computed as

$$\mathbf{n}^1 = \mathrm{IW}^{1,1}\mathbf{p} + b^1. \tag{5}$$

The output of Layer 1, $\mathbf{a}^1$, is computed by passing the net input through a transfer function, according to $\mathbf{a}^1 = f^1(\mathbf{n}^1)$. The output vector has S^1 elements. The output of the first layer is input to the second layer through the layer weight, $\mathrm{LW}^{2,1}$. A network can have any number of layers, although two- and three-layer networks are generally used. The overall output of the network is typically chosen to be the output of the last layer in the network, although it could be the output of any layer.[73]

For our investigations, the R elements of the input vector $\mathbf{p}^1$ comprise interatomic distances, bond angles, and dihedral angles. If the system under study contains N atoms, $\mathbf{p}^1$ will be a $(3N-6)\times 1$ column vector. The output vector of the nth layer, $\mathbf{a}^n$, will contain only a single element if the NN is trained to fit the potential energy of the system. If the NN is trained to fit the force field, $\mathbf{a}^n$ will be a $(3N-6)\times 1$ column vector. If we employ a NN trained to fit both the force field and the potential energy, $\mathbf{a}^n$ will be a $(3N-5)\times 1$ column vector.

The power of the multilayer network for function approximation has been well studied. Hornik, Stinchcombe, and White[74] have shown that a two-layer network with sigmoid functions in the first hidden layer and linear transfer functions in the output layer are universal approximators. Pinkus[75] gave a more recent review of the approximation capabilities of neural networks. Niyogi and Girosi[76] developed bounds on function approximation error when the network is trained on noisy data. In this research, our task is to choose the best neural network architecture to predict the required potentials and force fields from the supplied internal coordinates of the configuration and then to train the appropriate network to find the optimal set of weights and biases.

NN training algorithms are generally gradient-based optimization procedures that are designed to minimize the mean square error between the outputs of the network and the target outputs over the appropriately chosen training set. That is, the algorithm is provided with a set of examples of proper network behavior: $\{(\mathbf{p}_1,t_1),(\mathbf{p}_2,t_2),\ldots,(\mathbf{p}_k,t_k)\}$, where $\mathbf{p}_q$ is an input vector of internal coordinates to the network and t_q is the corresponding target output (e.g., potential energy). As each input $\mathbf{q}_q$ is applied to the network, the corresponding network output $\mathbf{a}_q$ is compared to the target. The training algorithm adjusts the network parameters in order to minimize the sum-squared error:

$$F(\mathbf{w}) = \sum_{q=1}^{Q} (\mathbf{t}_q - \mathbf{a}_q)^T(\mathbf{t}_q - \mathbf{a}_q), \tag{6}$$

where $\mathbf{w}$ is a vector containing all network weights and biases.

Downloaded 10 Mar 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

Some training algorithms are incremental, in which the weights are updated after each randomly selected input is presented to the network. The extended Kalman filter algorithms[77] are examples. Other algorithms have a batch form, in which all inputs are presented to the network before an update is made, as in the Levenberg–Marquardt algorithm.[78] The results reported here have utilized primarily the Levenberg–Marquardt procedure.

Before training the network, we must select the number of neurons and the number of layers necessary to achieve an accurate approximation. We also need to determine how the training data set should be selected. The key in selecting the network architecture is to use enough neurons to capture the complexity of the underlying function without having the network overfit the training data, in which case it will not generalize accurately to new situations. If the number of neurons is too large, the network will overfit the training data. In this case, the network response perfectly matches the training data, but interpolation between the training points is poor because the network has too many independent parameters that have not been constrained in any way.

Network interpolation can be improved by a variety of techniques. In one method, called "early stopping,"[79] we place a portion of the training data into a validation data set. The performance of the network on the validation set is monitored during training. During the early stages of training, the validation error will decrease. If overfitting begins, the validation error will begin to increase. At this point, training is stopped. Another method to improve the interpolation performance of the network is called "regularization." With this method, the performance index is modified to include a term that penalizes network complexity. The common penalty term is the sum of squares of the network weights. That is, we replace Eq. (6) with

$$F(\mathbf{w}) = \sum_{q=1}^{Q} (\mathbf{t}_q - \mathbf{a}_q)^T (\mathbf{t}_q - \mathbf{a}_q) + \rho \sum_{i=1}^{N} (w_i)^2. \tag{7}$$

This performance index forces the weights to be small, which produces a smoother (less complex) network response. The key to this method is choosing the best regularization parameter ρ. If the value is too large, the network response will be too smooth and will not accurately approximate the underlying function. If the value is too small, the network will overfit. There are several procedures for selecting the optimal value of ρ. One of the most successful is Bayesian regularization.[80,81]

In both early stopping and regularization, we gradually increase the number of neurons in the neural network until it is more than large enough to provide an accurate approximation. However, we constrain the size of the weights and the number of iterations of the training algorithm. These constraints reduce the effective number of parameters in the network and prevent overfitting. In this research, we use a combination of regularization and early stopping to produce the best generalization.[82]

A very important consideration in network performance is to have enough training points appropriately spaced throughout the configuration space that is important in the process under study. Neural networks do not extrapolate accurately. Therefore, we must have adequate training data in all important regions of configuration space. This will be accomplished using novelty detection.[68]

Since feedforward neural networks can be easily differentiated to yield the gradients of the function being fitted, the method provides us with a choice of three procedures to execute MD calculations. We can use a NN to fit the computed potentials and then obtain the force fields required by the MD calculations by differentiation of the network function. Alternatively, we can use a separate neural network to fit the force fields computed by the *ab initio* calculations. This network can then be used in the MD calculations to produce the forces directly. A third option is train one network to fit the computed potentials and at the same time, have the derivatives of the network function with respect to its inputs fit the force fields. The performance index for training this network would be a combination of the squared error on the computed potentials and the squared error on the force fields. To implement a gradient-based algorithm on the combined performance index, we would need to compute the derivative with respect to the network weights of the network function derivative with respect to its inputs. This requires backpropagation operations that are similar to those required for Hessian calculations.[83] If the NN fits are exact, the three methods are equivalent. However, since this will not be the case, all three procedures can be investigated to determine the one best suited to the MD calculations.

The fact that NN procedures can be employed to interpolate the computed derivatives of the potential directly provides a significant advantage over Shepard interpolation methods. To interpolate the force fields directly using Taylor series expansions truncated after the quadratic terms would require the evaluation of mixed third derivatives of the potential with respect to the expansion variables. These derivatives are generally not available from electronic structure calculations.

It should be noted that the interpolation accuracy of feedforward NN can be easily increased by employing more hidden layers and additional neurons. This is another major advantage over moving interpolation methods using Shepard's procedure and moving least-squares methods. To increase the accuracy of the Taylor series expansions employed in Shepard interpolations, additional terms must be included in the expansions. These terms require the evaluation of mixed third, fourth, etc. derivatives of the potential function, which are generally not available from electronic structure calculations.[40,47] To increase the accuracy of the moving least-squares procedures,[57,58,60] the number of basis functions M must be increased and the computational requirements of the method increase with M^2.

Neural networks have been used successfully by Blank *et al.*[84,85] to fit data derived from an empirical potential model for CO chemisorbed on a Ni(111) surface. These same investigators also examined the interaction potential of H_2 on a Si(100)-2 × 1 surface using a data set comprising 750 energies computed from local density functional theory. Hobday, Smith, and BelBruno[86] have used similar methods to investigate the energies of C–H systems with a NN compris-

Downloaded 10 Mar 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

ing five input nodes, one hidden layer with six nodes and an output layer. The computed equilibrium structures were almost identical with energies within 0.08%–0.14% of the Brenner[30] energies. Similar success with NN has been reported by Tafeit *et al.*[87] and by Gassner *et al.*[88] for torsional energies in large biological systems and for the $H_2O-Al^{3+}-H_2O$ system, respectively. Brown, Gibbs, and Clary[89] have employed a NN with 32 hidden nodes to describe the *ab initio* potential surface the $(HF)_2$ van der Waals complex. The error in the complex dissociation energy compared to the *ab initio* results was 0.048 kJ mol^{-1}.

In each of these cases, the dimensionality of the system was six or less and only one reaction channel was investigated. These factors significantly decrease the difficulty of the fitting problem. The results of the present study show that even when the dimensionality of the system becomes large and many reactions channels are open, neural networks can still be effectively employed to represent *ab initio* potential hypersurfaces provided they are combined with appropriate sampling and initiation methods.

IV. RESULTS

We have tested the feasibility of our proposed methods for gas-phase, chemical reaction dynamics and machining applications using the unimolecular, gas-phase dissociation of vinyl bromide and orthogonal cutting of a silicon workpiece as our prototypes. Each of these systems tests our proposed methods with its own distinct set of computational difficulties. The vinyl bromide potential is a 12-dimensional surface. Dissociation at an internal energy of 6.44 eV involves four distinct dissociation channels, bromine atom dissociation, hydrogen-carbon-atom bond scission, three-center HBr elimination and three-center H_2 elimination. The four-center H_2 and HBr dissociation channels have previously been found to be of minor importance.[37,70] The NN fits to the *ab initio* energies and force fields must, therefore, represent each of the channels accurately if the product yields, dissociation rates, and vibrational energy distributions for the various reaction products are to be correctly predicted. Silicon machining presents a very different set of problems. The energies and force fields for clusters containing up to seven silicon atoms with 98 electrons must be accurately represented. Moreover, the high-energy, violent nature of the cutting experiment ensures that many of these clusters will be in configurations that are very far from equilibrium. Therefore, the methods must be able to handle a very wide range of structures with sufficient accuracy to permit meaningful MD and MC calculations to be executed.

A. Vinyl bromide system

To execute the first step in the iterative sampling process described in Sec. III B, we have employed the empirical hybrid potential surface previously developed by Rahaman and Raff.[70] An ensemble comprising nonreactive trajectories and ones that result in three-center HBr or H_2 dissociation, hydrogen-atom, and bromine-atom dissociation were selected for the initial sampling. Configuration points along these trajectories were recorded at equally spaced time intervals to obtain 1400 points. The MP4(SDQ) energies and associated force fields for these configurations were computed on an AMD Athlon™ XP 2100+ machine with a 1.87 GHz clock speed. The calculations employed a 6-31G(d,p) basis set for the carbon and hydrogen atoms and Huzinaga's (4333/433/4) basis set augmented with split outer s and p (43321/4321/4) orbitals and a polarization f orbital with an exponent of 0.5 for the bromine atom. The atom labels used in the study are shown in Fig. 6.

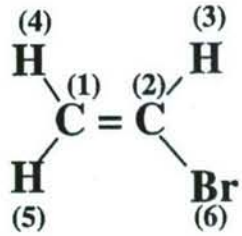


FIG. 6. Atom labels for vinyl bromide.

Prior to NN fitting, the data obtained from the MP4(SDQ) calculations were scaled to lie in the range $[-1,+1]$ using

$$K_s = \frac{2(K - K_{\min})}{K_{\max} - K_{\min}} - 1, \tag{8}$$

where K_s is the scaled result and $K_{\min}$ and $K_{\max}$ are the minimum and maximum values of the potential energy or the associated derivatives in the database. After dividing the data into training, validation, and testing sets, a NN with 12 inputs, 20 hidden neurons, and one output (12-20-1) was fitted to the scaled data using the Levenberg–Marquardt algorithm.[78] The NN employed sigmoid transfer functions and linear output functions. The scaled input data are much more closely approximated by the NN sigmoid transfer functions, which range from −1 to +1, than the unscaled values. Early stopping[79] was used to avoid overfitting. The fitting process required approximately six minutes of computational time. This is in sharp contrast to the months of effort needed to fit complex, multidimensional empirical surfaces.[70]

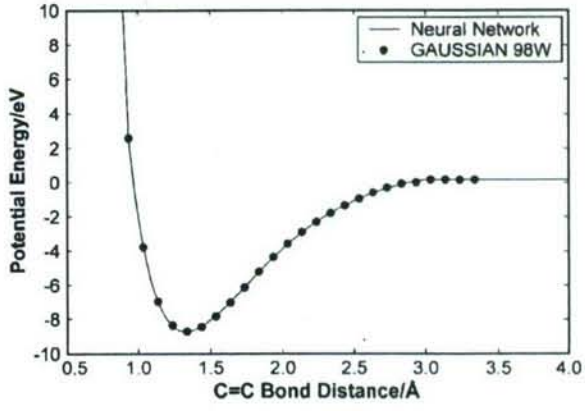


FIG. 7. Comparison of *ab initio* MP4(SDQ) calculations (plotted points) with the predictions of the NN (smooth curve) for the C=C stretching potential of vinyl bromide.

Downloaded 10 Mar 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

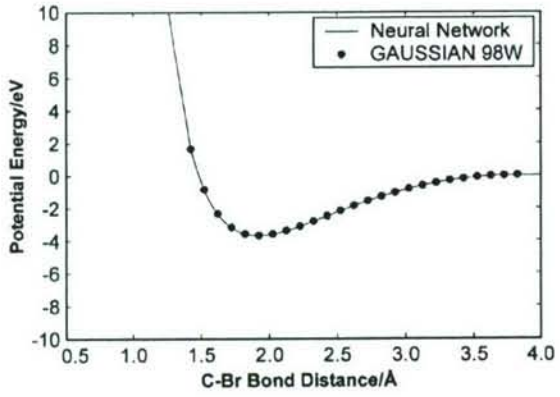


FIG. 8. Comparison of *ab initio* MP4(SDQ) calculations (plotted points) with the predictions of the NN (smooth curve) for the C–Br stretching potential of vinyl bromide.

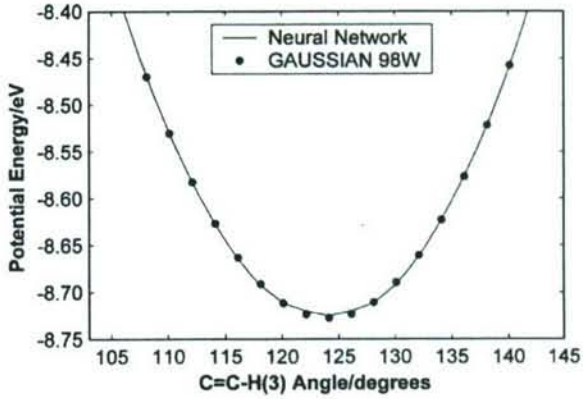


FIG. 10. Comparison of *ab initio* MP4(SDQ) calculations (plotted points) with the predictions of the NN (smooth curve) for the C=C–$H^{(3)}$ bending potential of vinyl bromide.

Figures 7 and 8 show the accuracy of the NN fits for C=C and $C^{(2)}$–Br, bond stretching, respectively. Figure 9 shows the variation of the potential for $C^{(2)}$–$H^{(3)}$ stretching motion. The plotted points in the figures are the MP4(SDQ) results; the curve is the prediction from the neural network. As can be seen, the agreement is excellent. Figures 10 and 11 show the results for variations in the C=C–$H^{(3)}$ and C=C–Br angles, respectively. Figures 12 and 13 illustrate the accuracy of the NN fitting for variations of the $H^{(4)}$–C=C–Br and $H^{(5)}$–C=C–$H^{(3)}$ dihedral angles, respectively. In each case, the neural network provides an excellent fit to the *ab initio* database.

Figure 14 shows a comparison of the MP4(SDQ) scaled energies and the corresponding NN results for the test set of vinyl bromide configurations. If the NN fit were perfect, all points would fall on the 45° line shown in the figure. The computed rms deviation of the NN and quantum mechanical energies is 0.092 kJ mol^{-1}. Clearly, the fit is nearly exact. Comparison with the data given in Table II for the six-atom H_2O+HOC^+ system shows the rms error using the Shepard interpolation method was 38 times greater than that for the vinyl bromide system, which has four open reaction channels as opposed to two for the H_2O+HOC^+ surface. Figure 15 shows the distribution of the deviations between the *ab initio* and NN results. The points give the computed distribution. The smooth curve is a least-squares fit of the Lorentzian function

$$P(\Delta E) = \frac{\alpha}{(\Delta E - \beta)^2 + \sigma^2}, \tag{9}$$

where ΔE is the difference between the scaled *ab initio* and NN potential energies, and α, β, and σ are fitting parameters. The fit shown in Fig. 15 has $\alpha = 8.01 \times 10^{-6}$, $\beta = 1.43 \times 10^{-6}$, and $\sigma = 1.67 \times 10^{-4}$.

B. Si_5 system

We have also obtained an *ab initio* potential-energy hypersurface for five-atom silicon clusters that occur in the simulation of a cutting experiment with a diamond tool and silicon workpiece. As previously noted, the very high energies present in a cutting experiment significantly increase the size of the subset of configuration space that must be sampled to obtain an accurate potential-energy hypersurface. In effect, there are multiple "reaction channels" open. There-

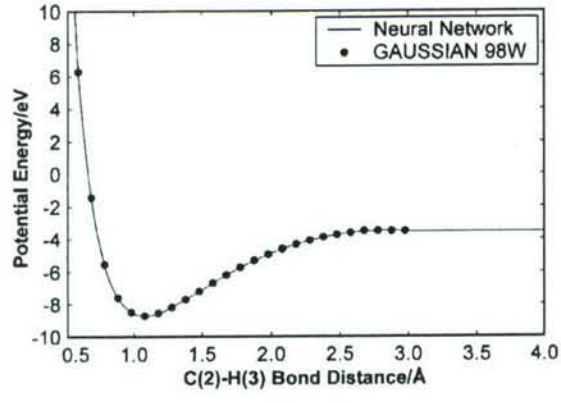


FIG. 9. Comparison of *ab initio* MP4(SDQ) calculations (plotted points) with the predictions of the NN (smooth curve) for the $C^{(2)}$–$H^{(3)}$ stretching potential of vinyl bromide.

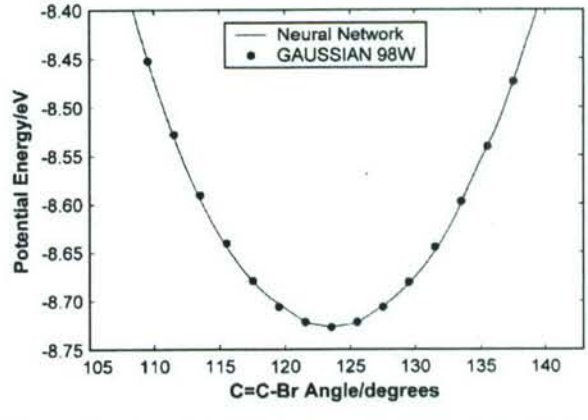


FIG. 11. Comparison of *ab initio* MP4(SDQ) calculations (plotted points) with the predictions of the NN (smooth curve) for the C=C–Br bending potential of vinyl bromide.

Downloaded 10 Mar 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

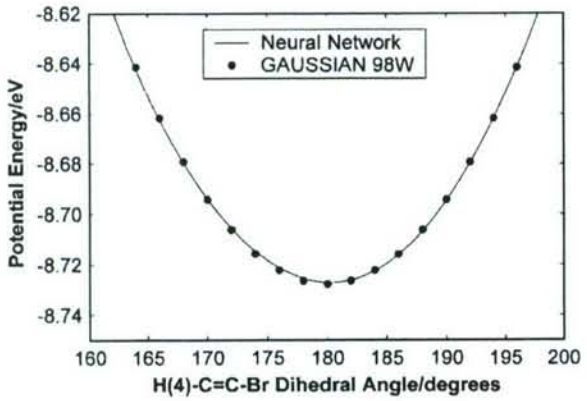


FIG. 12. Comparison of *ab initio* MP4(SDQ) calculations (plotted points) with the predictions of the NN (smooth curve) for variations of the $H^{(4)}-C=C-Br$ dihedral angle of vinyl bromide.

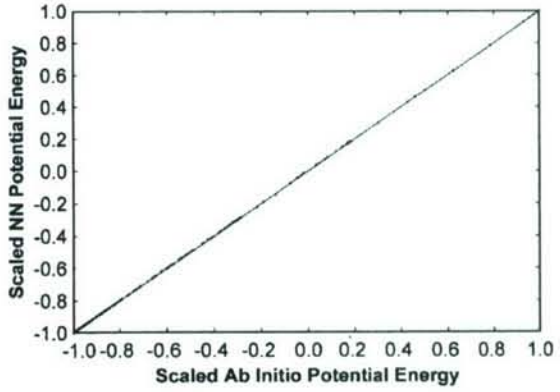


FIG. 14. Comparison of scaled [Eq. (8)] *ab initio* MP4(SDQ) energies with the predictions of the NN for a test set of about 140 configurations. If the fit were perfect, all points would fall on the 45° line shown in the figure. The standard deviation of the points from this line corresponds to 0.092 kJ mol^{-1}.

fore, this system is an extremely demanding test of our methods.

A Tersoff potential[36] was employed in the initial step of the iterative process. MD cutting simulations with a +15° rake angle, 10° clearance angle, 1 Å cutting depth, 54.3 Å cutting width, at a cutting speed of 491.2 m s^{-1} were carried out. The total number of atoms in the workpiece was 1675. Throughout the MD simulations, the configurations of the five-atom silicon clusters that are present in front of the tool, in the chip, and within a few unit cell distances in the workpiece beneath the tool are stored.

Subsequent to the MD calculations, the energies and forces fields for each of the 18 000 stored configurations are computed using density functional theory (DFT) with a 6-31G** basis set and the B3LYP procedure for incorporating correlation energy.[66] In executing these calculations, it is important to take symmetry into account. The energy and force fields for any given Si_5 cluster are obviously unaltered if two or three silicon atoms are exchanged. However, the neural network does not know this. Therefore, we must either train the NN to recognize symmetry by including all permutations of identical atoms in the database or adopt a procedure that makes this unnecessary. Since there are 5! permutations for the Si_5 system, we have adopted the latter approach. The *Z*-matrix input describing the Si_5 clusters comprise four Si–Si bond distances, three Si–Si–Si angles, and two dihedral angles. Before each calculation, we list the four bond distances in increasing order of their deviation from the Si–Si equilibrium bond length. This listing determines the *Z*-matrix input vector for the bond angles and dihedral angles. When the neural net is utilized to interpolate energies or force fields, the same ordering of the input vector is used. In effect, this procedure obviates the need to train the NN to recognize symmetry and simplifies the overall training process.

The database obtained from the cutting simulations was augmented with an additional 10 000 five-atom Si_5 configurations near equilibrium. These configurations were obtained by placing thermal energy corresponding to a temperature of 300 K in the silicon workpiece and then following the vibrational motions of the lattice using molecular dynamics with the Tersoff empirical potential for silicon.[36] Five-atom configurations were stored at equally spaced time intervals dur-

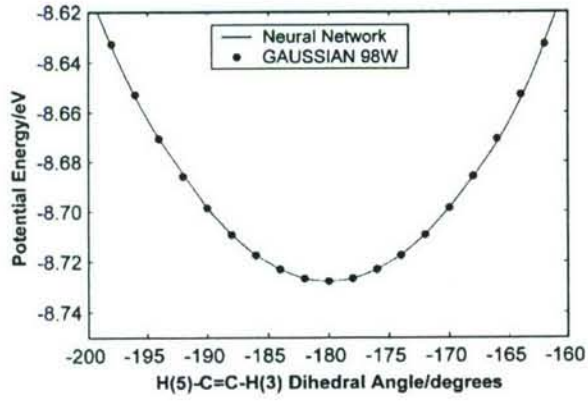


FIG. 13. Comparison of *ab initio* MP4(SDQ) calculations (plotted points) with the predictions of the NN (smooth curve) for variations of the $H^{(5)}-C=C-H^{(3)}$ dihedral angle of vinyl bromide.

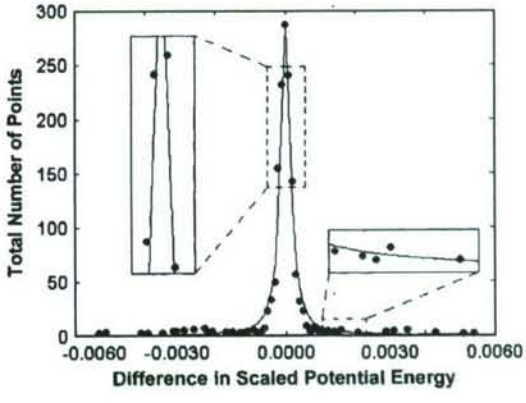


FIG. 15. Distribution of the deviations of the predictions of the NN from the *ab initio* MP4(SDQ) energies for vinyl bromide. A total of 140 testing configurations are included. The computed results are shown as plotted points. The smooth curve is a least squares fit of Eq. (9) to these data.

Downloaded 10 Mar 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

ing the calculations. The energies and force fields for these configurations were obtained using DFT and the *Z*-matrix ordering previously described.

In the second phase of the procedure, the *ab initio* potential energies for each of the 28 000 configurations are scaled using Eq. (8). The scaled database is then divided into training, validation, and testing sets. Approximately 10% of the data are used for the validation set, 10% for the testing set, and the remainder is the training set. A (9-45-1) NN is fitted to the data in the training set using the Levenberg–Marquardt algorithm.[78] The NN employed sigmoid transfer functions and linear output functions.

To avoid overfitting, it is essential to employ an early stopping[79] procedure. In this method, the rms deviation of the NN output from the data in the validation set is computed after each iterative training cycle in the NN fitting process. At first, this rms deviation will decrease as the number of training cycles increases because the accuracy of the fit is improving. At some point, however, overfitting of the training set will begin. While overfitting will improve the apparent accuracy of the fit to the training data, it will increase the rms deviation of the NN output to the validation data since these data are not included in the training. The procedure, therefore, is to continue the training process until the rms deviation of the NN output from the validation data attains a minimum value. The training process is stopped at that point. We have also incorporated Bayesian regularization[80,81] to improve network performance.

The entire training process for Si_5 with a database of 22 400 configurations required about 16 h of CPU time. We have found that this can be reduced to about 8 h by using a conjugate gradient fitting method[71] initially and then switching to the Levenberg–Marquardt algorithm[78] about halfway through the fitting. Comparison of the networks obtained by the two procedures showed that they produce nearly the same interpolated energies and force fields for Si_5. It is important to note that these fitting processes require little or no human intervention. The computational effort needed is at least two orders of magnitude less than that required to fit multidimensional empirical surfaces containing over 100 adjustable parameters.[70] The reduction in human time is even greater.

Figure 16 shows a comparison of the DFT scaled energies and the corresponding NN results for the testing subset of the 28 000 Si_5 configurations. If the NN fit were perfect, all points would fall on the 45° line shown in the figure. The computed rms deviation of the NN and DFT energies is 0.36 kJ mol^{-1}. This level of accuracy is comparable to that achieved using Shepard interpolation methods for three- and four-atom systems and significantly better than the corresponding accuracy reported for a six-atom system with two open reaction channels using Shepard interpolation.[63]

The next step in the procedure is to utilize the NN to execute AIMD calculations during which we employ novelty sampling techniques to obtain additional configurations to iteratively improve the potential-hypersurface until an acceptable degree of convergence to the final potential and force fields is obtained. To illustrate the method, we have examined the vibrational motions of a silicon lattice at 800 K.

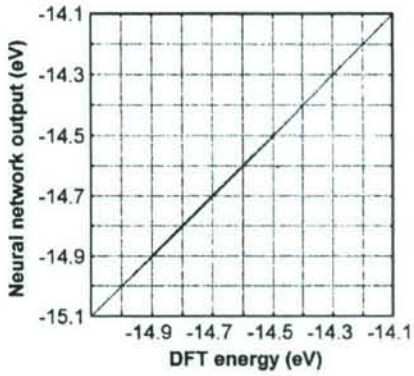


FIG. 16. Comparison of scaled [Eq. (8)] *ab initio* DFT energies with the predictions of the NN for the testing subset of 28 000 Si_5 configurations. If the fit were perfect, all points would fall on the 45° line shown in the figure. The standard deviation of the points from this line corresponds to 0.36 kJ mol^{-1}.

Figure 17(a) shows the distribution of minimum distances, $P(\Gamma_m)$, for the initial set of configurations. When Eq. (3) with $T_1=0.5$ and $\Gamma_m^*=0.06$ is applied to the configurations resulting from trajectories computed using the NN fit to the original data set, several thousand additional configurations are identified whose distribution of Γ_m values is that shown in Fig. 17(b). Comparison of this distribution with Fig. 16 for the original configurations shows that the new configurations lie in a region of configuration space not adequately sampled by the original data set. These configurations are, therefore, incorporated into the overall database and a new NN is fitted to the computed DFT energies. The new distribution $P(\Gamma_m)$ for the concatenated set of configurations is shown in Fig. 17(c). During the second iteration, essentially no new configurations are found whose Γ_m values do not lie underneath the distribution shown in Fig. 17(c). The process has, therefore, converged for the test system. The advantage of this procedure is that it is unnecessary to compute dynamics results in order to test for convergence.

V. SUMMARY

A feedforward NN-novelty sampling approach for the development of accurate *ab initio* potential-energy hypersurfaces that can be utilized to conduct molecular dynamics and Monte Carlo studies of large systems with multiple open reaction channels has been described. The method integrates *ab initio* electronic structure calculations with trajectory/novelty-sampling techniques that permit the critical regions of configuration space to be determined. The modified novelty-sampling method involves the tight integration of MD calculations with neural networks that employ early stopping and regularization procedures to improve network performance and test for convergence.

The electronic structure calculations utilize existing software that permit the accuracy of the quantum mechanical calculations to be scaled up as computational resources permit. The method is, therefore, robust.

Downloaded 10 Mar 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

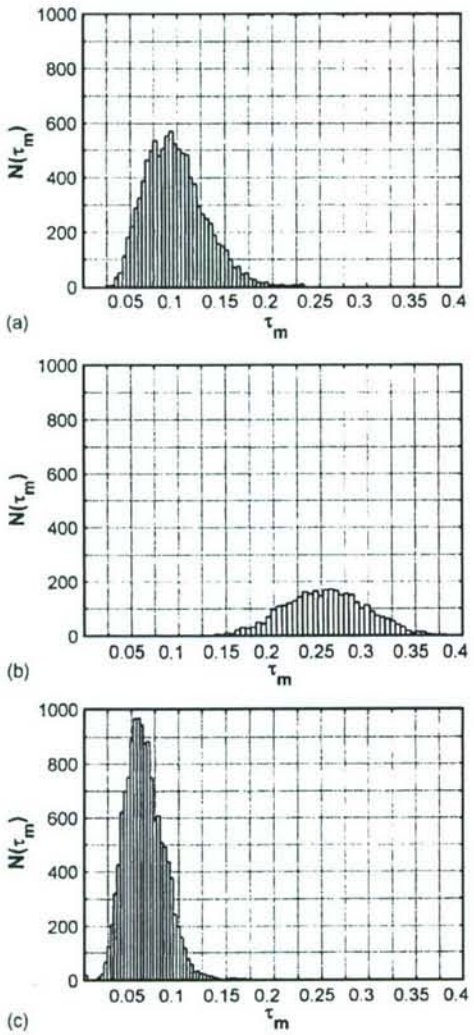


FIG. 17. (a) Distribution of minimum distances $P(\Gamma_m)$ for the initial set of 28 000 Si_5 configurations. (b) Distribution of Γ_m values for the new configurations resulting from the applications of Eqs. (3) and (4) to Si_5 structures computed using the NN fit to the original Si_5 data set. (c) New distribution of Γ_m values for the concatenated set of configurations resulting from the Si_5 structures obtained in the original trajectories using the Tersoff potential and those obtained in the first iteration on the NN surface.

The accuracy of the neural net interpolation is excellent with rms deviations of NN predictions from quantum mechanically computed energies on the order of 0.09–0.36 kJ mol^{-1} for five- and six-atom systems with multiple open reaction channels. The required computational time for NN fitting is on the order of 10 min to 16 h for the systems considered here. There is virtually no human time involved in the procedure, which is in contrast to the large human effort required to fit arbitrarily chosen, parameterized functional forms. Moreover, the NN fit is analytic so that there are no spurious discontinuities in higher derivatives that are often produced by switching functions that are employed to connect channel potentials.[70]

The sampling procedures easily allow new configurations in the iterative MD calculations to be identified as such. The energies and force fields for these configurations are computed and the results incorporated into the total database after which a new NN is fitted. Moreover, the sampling procedures serve as an effective convergence test that does not require the MD computation of the reaction dynamics. When subsequent MD calculations produce very few new configurations, convergence to the final potential surface is essentially assured.

The NN methods provide several significant advantages over moving interpolation procedures[54] employing Shepard's interpolation[41,42] or least-squares methods.[57,58,60] These include (1) increased interpolation accuracy, (2) procedures to increase interpolation accuracy that are easily implemented, (3) the flexibility to fit the force fields directly with no implementation problems, and (4) more accurate procedures for initiating the method with self-starting methods readily available.

ACKNOWLEDGMENTS

This project is funded by (Grant No. DMI-0200327) the National Science Foundation. We thank Dr. W. DeVries, Dr. G. Hazelrigg, Dr. J. Chen, and Dr. D. Durham of the Division of Design, Manufacturing, and Industrial Innovation, Dr. B. M. Kramer, Engineering Centers Division, and Dr. J. Larsen Basse, Tribology and Surface Engineering program, for their interest and support of this work. This project was also sponsored by a DEPSCoR grant on the Multiscale Modeling and Simulation of Material Processing (Grant No. F49620-03-1-0281). The authors thank Dr. Craig S. Hartley of the AFOSR for his interest in and support of this work.

[1]B. Bhushan, *Handbook of Micro/Nano Tribology* (CRC, New York, 1995); Y. Yan, M. Yoshino, T. Kuriagawa, T. Shirakashi, K. Syoji, and R. Komanduri, Mater. Sci. Eng., A **297**, 230 (2001).
[2]T. D. Sewell and D. L. Thompson, J. Chem. Phys. **93**, 4077 (1990).
[3]A. J. Marks, J. Chem. Phys. **100**, 8096 (1994).
[4]B. M. Rice and D. L. Thompson, J. Chem. Phys. **93**, 7986 (1990).
[5]R. V. Alimi, V. A. Apkarian, and R. B. Gerber, J. Chem. Phys. **98**, 331 (1993).
[6]G. Lendvay and G. C. Schatz, J. Chem. Phys. **96**, 4356 (1992).
[7]G. Lendvay and G. C. Schatz, J. Chem. Phys. **98**, 1034 (1993).
[8]F. E. Budenholzer, M. Y. Chang, and K. C. Huang, J. Phys. Chem. **98**, 12501 (1994).
[9]L. M. Raff, J. Chem. Phys. **93**, 3160 (1990).
[10]L. M. Raff, J. Chem. Phys. **95**, 8901 (1991).
[11]L. M. Raff, J. Chem. Phys. **97**, 7459 (1992).
[12]L. M. Raff, J. Chem. Phys. **90**, 6313 (1989).
[13]L. M. Raff, J. Phys. Chem. **91**, 3266 (1987).
[14]L. M. Raff, J. Phys. Chem. **92**, 141 (1988).
[15]L. M. Raff and R. W. Graham, J. Phys. Chem. **92**, 5111 (1988).
[16]For examples of uniaxial tension calculations, see, R. L. B. Selinger, Z. G. Wang, and W. M. Gelbart, Phys. Rev. A **43**, 4396 (1991); R. L. B. Selinger, R. M. Lynden-Bell, and W. M. Gelbard, J. Chem. Phys. **98**, 9808 (1993); R. M. Lynden-Bell, J. Phys.: Condens. Matter **4**, 2172 (1992); **7**, 4603 (1995); T. Kitamura, K. Yashiro, and R. Ohtani, JSME Int. J., Ser. A **40**, 430 (1997); P. Heino, H. Hakkinen, and K. Kaski, Phys. Rev. B **58**, 641 (1998); Europhys. Lett. **41**, 273 (1998); R. Komanduri, N. Chandrasekaran, and L. M. Raff, Int. J. Mech. Sci. **43**, 2237 (2001); Mater. Sci. Eng., A **340**, 58 (2003).
[17]For examples of MD cutting calculations, see the review article, "A review on the molecular dynamics simulation of machining at the atomic scale," R. Komanduri and L. M. Raff, Proc. Inst. Mech. Eng., Part C: J. Mech. Eng. Sci. **215**, 1639 (2001), and the 124 references cited therein.
[18]N. Chandrasekaran, A. Noori Khajavi, L. M. Raff, and R. Komanduri, Philos. Mag. B **77**, 7 (1998).

Downloaded 10 Mar 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

[19]R. Komanduri, N. Chandrasekaran, and L. M. Raff, Wear **219**, 84 (1998).
[20]R. Komanduri, N. Chandrasekaran, and L. M. Raff, Philos. Mag. B **79**, 955 (1999).
[21]R. Komanduri, N. Chandrasekaran, and L. M. Raff, Wear **242**, 60 (2000).
[22]R. Komanduri, N. Chandrasekaran, P. R. Mukund, and L. M. Raff, Wear **240**, 113 (2000).
[23]R. Komanduri, N. Chandrasekaran, and L. M. Raff, Phys. Rev. B **61**, 14007 (2000).
[24]R. Komanduri, N. Chandrasekaran, and L. M. Raff, Ann. ICRP **48**, 1 (1999).
[25]R. Komanduri, N. Chandrasekaran, and L. M. Raff, Mater. Sci. Eng., A **311**, 1 (2001).
[26]R. Komanduri and L. M. Raff, *Proceedings of Third International Conference on Metal Cutting and High-Speed Machining*, June 27–29, 2001, Metz, France.
[27]R. Komanduri, N. Chandrasekaran, and L. M. Raff, Philos. Mag. B **81**, 1989 (2001).
[28]R. Komanduri, N. Chandrasekaran, and L. M. Raff, Mater. Sci. Eng., A **340**, 58 (2003).
[29]A. Chandrasekaran, L. M. Raff, and R. Komanduri, Philos. Mag. Lett. **82**, 247 (2002).
[30]D. W. Brenner, Phys. Rev. B **42**, 9458 (1990).
[31]B. C. Bolding and H. C. Anderson, Phys. Rev. B **41**, 10568 (1990).
[32]M. S. Daw and M. I. Baskes, Phys. Rev. B **29**, 6443 (1984).
[33]M. I. Baskes, Phys. Rev. B **46**, 2727 (1992).
[34]M. I. Baskes, *Sandia Report* 94-8704, 1995.
[35]M. I. Baskes and H. C. Johnson, Modell. Simul. Mater. Sci. Eng. **2**, 147 (1994).
[36]J. Tersoff, Phys. Rev. B **37**, 6991 (1988); **39**, 5566 (1989).
[37]S. A. Abrash, R. W. Zehner, G. J. Mains, and L. M. Raff, J. Phys. Chem. **99**, 2959 (1995).
[38]See, for example, B. Liu, J. Chem. Phys. **58**, 1925 (1973); P. Siegbahm and B. Liu, *ibid.* **68**, 2457 (1978); B. Liu, *ibid.* **80**, 581 (1984).
[39]E. Bodo, R. Martinazzo, and F. A. Gianturco, *High Accuracy Potentials for Quantum Dynamics*, edited by A. Miani, J. Tennyson, and T. van Mourik (ChemReact, London, 2003), CCP1/CCP6, p. 43.
[40]J. Ischtwan and M. A. Collins, J. Chem. Phys. **100**, 8080 (1994).
[41]R. Farwig, *Algorithms for Approximation*, edited by J. C. Mason and M. G. Cox (Clarendon, Oxford, 1987).
[42]P. Lancaster and K. Salkauskas, *Curve and Surface Fitting, An Introduction* (Academic, London, 1986).
[43]M. J. T. Jordan, K. C. Thompson, and M. A. Collins, J. Chem. Phys. **102**, 5647 (1995).
[44]G. Nyman, S. Nordholm, and H. W. Schranz, J. Chem. Phys. **93**, 6767 (1990).
[45]E. G. Severin, B. C. Freasier, N. D. Hamer, D. L. Jolly, and S. Nordholm, Chem. Phys. Lett. **57**, 117 (1978).
[46]M. A. Collins and R. P. A. Bettens, Phys. Chem. Chem. Phys. **1**, 939 (1999).
[47]M. A. Collins and R. P. A. Bettens, J. Chem. Phys. **109**, 9728 (1998).
[48]K. C. Thompson and M. A. Collins, J. Chem. Soc., Faraday Trans. **93**, 871 (1997).
[49]R. P. A. Bettens, T. A. Hansen, and M. A. Collins, J. Chem. Phys. **111**, 6322 (1999).
[50]R. P. A. Bettens and M. A. Collins, J. Chem. Phys. **108**, 2424 (1998).
[51]M. A. Collins and D. H. Zhang, J. Chem. Phys. **111**, 9924 (1999).
[52]K. A. Nguyen, I. Rossi, and D. G. Truhlar, J. Chem. Phys. **103**, 9728 (1998).
[53]R. P. A. Bettens and M. A. Collins, J. Chem. Phys. **111**, 816 (1999).
[54]M. A. Collins, Theor. Chem. Acc. **108**, 313 (2002).
[55]T. Ho, T. Hollebeek, H. Rabitz, L. B. Harding, and G. C. Schatz, J. Chem. Phys. **105**, 10472 (1996).
[56]L. A. Pederson, G. C. Schatz, T. Hollebeek, T. Ho, H. Rabitz, and L. B. Harding, J. Phys. Chem. **104**, 2301 (2000).
[57]G. C. Maisuradze, D. L. Thompson, A. F. Wagner, and M. Minkoff, J. Chem. Phys. **119**, 10002 (2003).
[58]Y. Guo, A. Kawano, D. L. Thompson, A. F. Wagner, and M. Minkoff, J. Chem. Phys. **121**, 5091 (2004).
[59]B. Kuhn, T. R. Rizzo, D. Luckhaus, M. Quack, and M. A. Suhm, J. Chem. Phys. **111**, 2565 (1999).
[60]A. Kawano, Y. Guo, D. L. Thompson, A. F. Wagner, and M. Minkoff, J. Chem. Phys. **120**, 6414 (2004).
[61]T. Ishida and G. C. Schatz, J. Chem. Phys. **107**, 3558 (1997).
[62]K. C. Thompson, M. J. T. Jordan, and M. A. Collins, J. Chem. Phys. **108**, 8302 (1998).
[63]M. A. Collins and L. Radom, J. Chem. Phys. **118**, 6222 (2003).
[64]G. E. Morzano and M. A. Collins, J. Chem. Phys. **119**, 5510 (2003).
[65]C. Crepos, M. A. Collins, E. Pijper, and G. J. Kroes, J. Chem. Phys. **120**, 2392 (2004).
[66]M. J. Frisch, G. W. Trucks, H. B. Schlegel *et al.*, GAUSSIAN 98 (Revision A.1), Gaussian, Inc., Pittsburgh, PA, 2001.
[67]L. M. Raff and D. L. Thompson, in *The Classical Trajectory Approach to Reactive Scattering*, Theory of Chemical Reaction Dynamics, Vol. III, edited by M. Baer (CRC Press, Boca Raton, FL, 1985), p. 1.
[68]A. Pukrittayakamee and M. T. Hagan (unpublished).
[69]I. T. Jolliffe, *Principal Component Analysis* (Springer-Verlag, New York, 1986).
[70]A. Rahaman and L. M. Raff, J. Phys. Chem. A **105**, 2156 (2001); J. Phys. Chem. **105**, 2147 (2001).
[71]M. T. Hagan, H. B. Demuth, and M. Beale, *Neural Network Design* (PWS Publishing Company, Boston, MA, 1996).
[72]S. Haykin, *Neural Networks: A Comprehensive Foundation* (Prentice Hall, Saddle River, NJ, 2nd Ed., 1999).
[73]H. B. Demuth and M. Beale, *Users' Guide for the Neural Network Toolbox for* MATLAB, ver. 4.0, (The Mathworks, Natick, MA, 2000).
[74]K. M. Hornik, M. Stinchcombe, and H. White, Neural Networks **2**, 359 (1989).
[75]A. Pinkus, Acta Numerica **8**, 143 (1999).
[76]P. Niyogi and F. Girosi, Adv. Comput. Math. **10**, 51 (1999).
[77]G. V. Puskorius and L. A. Feldkamp, *Proceedings of the International Joint Conference on Neural Networks (IJCNN'91)*, Seattle, Washington, 1991.
[78]M. T. Hagan and M. Menhaj, IEEE Trans. Neural Netw. **5**, 989 (1994).
[79]W. S. Sarle, *Proceedings of the 27th Symposium on the Interface*, 1995.
[80]D. J. C. MacKay, Neural Comput. **4**, 448 (1992).
[81]F. D. Foresee and M. Hagan, *Proceedings of the 1997 International Conference on Neural Networks*, Houston, Texas, 1997.
[82]D. Chen and M. Hagan, *Proceedings of the 1999 International Joint Conference on Neural Networks*, Washington, D.C., 1999.
[83]C. M. Bishop, Neural Comput. **4**, 494 (1992).
[84]T. B. Blank and S. D. Brown, Anal. Chem. **277**, 273 (1993).
[85]T. B. Blank, S. D. Brown, A. W. Calhoun, and D. J. Doren, J. Chem. Phys. **103**, 4129 (1995).
[86]S. Hobday, R. Smith, and J. BelBruno, Nucl. Instrum. Methods Phys. Res. B **153**, 247 (1999).
[87]E. Tafeit, W. Estelberger, R. Horejsi, R. Moeller, K. Oettl, K. Vrecko, and G. Reibnegger, J. Mol. Graphics **14**, 12 (1996).
[88]H. Gassner, M. Probst, A. Lauenstein, and K. Hermansson, J. Phys. Chem. A **102**, 4596 (1998).
[89]D. F. R. Brown, M. N. Gibbs, and D. C. Clary, J. Chem. Phys. **105**, 7597 (1996).

Downloaded 10 Mar 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

THE JOURNAL OF CHEMICAL PHYSICS 123, 224711 (2005)

Prediction of molecular-dynamics simulation results using feedforward neural networks: Reaction of a C_2 dimer with an activated diamond (100) surface

Paras M. Agrawal and Abdul N. A. Samadh
Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, Oklahoma 74078

Lionel M. Raff[a)]
Chemistry Department, Oklahoma State University, Stillwater, Oklahoma 74078

Martin T. Hagan
Electrical and Computer Engineering, Oklahoma State University, Stillwater, Oklahoma 74078

Satish T. Bukkapatnam
Industrial and Management Engineering, Oklahoma State University, Stillwater, Oklahoma 74078

Ranga Komanduri
Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, Oklahoma 74078

(Received 2 August 2005; accepted 6 October 2005; published online 13 December 2005)

A new approach involving neural networks combined with molecular dynamics has been used for the determination of reaction probabilities as a function of various input parameters for the reactions associated with the chemical-vapor deposition of carbon dimers on a diamond (100) surface. The data generated by the simulations have been used to train and test neural networks. The probabilities of chemisorption, scattering, and desorption as a function of input parameters, such as rotational energy, translational energy, and direction of the incident velocity vector of the carbon dimer, have been considered. The very good agreement obtained between the predictions of neural networks and those provided by molecular dynamics and the fact that, after training the network, the determination of the interpolated probabilities as a function of various input parameters involves only the evaluation of simple analytical expressions rather than computationally intensive algorithms show that neural networks are extremely powerful tools for interpolating the probabilities and rates of chemical reactions. We also find that a neural network fits the underlying trends in the data rather than the statistical variations present in the molecular-dynamics results. Consequently, neural networks can also provide a computationally convenient means of averaging the statistical variations inherent in molecular-dynamics calculations. In the present case the application of this method is found to reduce the statistical uncertainty in the molecular-dynamics results by about a factor of 3.5. © *2005 American Institute of Physics*. [DOI: 10.1063/1.2131069]

I. INTRODUCTION

The use of neural networks (NNs) to predict an outcome (or the output results as a function of a set of input parameters) has been gaining wider acceptance with the advance in computer technology as well as with an increased awareness of the potential of NN.[1–7] A neural network is first trained to learn the underlying functional relationship between the output and the input parameters by providing it with a large number of data points, where each data point corresponds to a set of output and input parameters. The data required to train a NN may be obtained from experimental results or by computer simulations. For example, one can perform *ab initio* calculations to compute the interaction potential for a system of atoms as a function of internal coordinates for a large number of configurations, and then these data can be used to train a NN to provide the interpolated potential energy for any other unknown configuration.[8–15] Recently, for multiatomic systems such as vinyl bromide and a Si_5 cluster, Raff *et al.*[8] have shown that such a procedure leads to interpolated values of the potential with a high level of accuracy.

Sumpter and Noid[16] demonstrated the use of NN to map the vibrational motion derived from the vibrational spectra onto a potential-energy surface (PES) with relatively high accuracy. In another application, Sumpter *et al.*[17] trained a NN to learn the relation between the phase-space points along a trajectory and the mode energies for stretching, torsion, and bending vibrations of H_2O_2. Likewise, Nami *et al.*[18] demonstrated the use of NN to determine the TiO_2 deposition rates in a chemical-vapor deposition (CVD) process from the knowledge of a range of deposition conditions.

In view of the success achieved in obtaining interpolated values of the PESs for multiatomic systems using a NN trained by the *ab initio* energy values for a large number of configurations, it is reasonable to ask if we can successfully compute the interpolated value of a reaction rate or reaction probability for a chemical reaction using a NN trained by the data obtained by molecular-dynamics (MD) simulations. To

a)Electronic mail: lionelraff@hotmail.com

 © 2005 American Institute of Physics

Downloaded 19 Dec 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

be specific, we consider here an example of the interaction of a carbon dimer C_2 with a diamond (100) surface. Due to its importance in the growth of nanocrystalline diamond, the study of the C_2-diamond surface system has been an area of considerable interest.[19–30] In the present investigation, we explore the validity of the results provided by a trained NN. For the training of the NNs, we obtain data by performing MD simulations to compute chemisorption, scattering, and desorption probabilities as a function of the direction of the initial velocity vector (θ, ϕ), impact parameter (b), translational energy (E_{trans}), and rotational energy (E_{rot}) of C_2 for a given lattice temperature (see the next section for a definition of these parameters). We then test the trained NN by comparing the predictions provided by the NN with MD simulations for various input conditions.

In Secs. II and III, we present the details of the MD simulations and the NN fitting procedure, respectively. The results are presented and discussed in Sec. IV and are summarized in Sec. V.

II. MOLECULAR-DYNAMICS SIMULATIONS

Various aspects of the MD simulations presented in this investigation are the same as those used in our previous gas-surface scattering studies.[31–37] A total of 324 atoms are used to model the system. Out of these, 282 atoms of diamond substrate are used to model the (100) crystalline face with 40 atoms of hydrogen on the top layer of the diamond surface and 2 atoms in the C_2 dimer. The BRENNERMD code used employs the potential given by Brenner *et al.*[38] with van der Waals interactions incorporated using a Lennard-Jones (LJ) potential. The (x, y) coordinates of the atoms on the top three layers are as shown in Fig. 1(a) (for atoms of top layer, $z = 0$). Each of the carbon atoms on the top layer, except the central atom and boundary atoms, is capped by a hydrogen atom. The equations of motion are integrated using Gear's predictor-corrector method[39] with a time step size Δt of 0.5 fs.

The results of a given trajectory depend upon a multitude of input variables. These include b, θ, ϕ, E_{rot}, E_{trans}, the initial orientation of the C_2 dimer, the angle defining the C_2 rotational plane, the initial C_2 vibrational energy and its phase, the temperature of the system, and all of the variables that define the vibrational phases of the diamond surface. Here, θ denotes the angle of incidence, i.e., the angle between the direction of the initial translational velocity vector of the C_2 dimer and the perpendicular on the surface (Z direction). The impact parameter b is defined as the distance between the location of the central atom C [see Figs. 1(a) and 1(b)] and the point of intersection P of the initial velocity vector and the diamond surface, as shown in Fig. 1(b). ϕ represents the angle between the line CP and X axis, as shown in Fig. 1(b), and E_{rot} and E_{trans} refer to the rotational and translational kinetic energies of the dimer, respectively.

In this investigation, we have focused our efforts on the determination of the dependence of probabilities for chemisorption, scattering, and desorption upon b, θ, ϕ, E_{rot}, and E_{trans}. The initial C_2 vibrational energy is set equal to the zero-point energy and the temperature of the lattice is maintained constant at 600 K using the Berendsen thermostat procedure.[40] For a given input set of b, θ, ϕ, E_{rot}, and E_{trans}, the probabilities of different reactions are determined by running 50 trajectories to effect the averaging over the remaining variables.

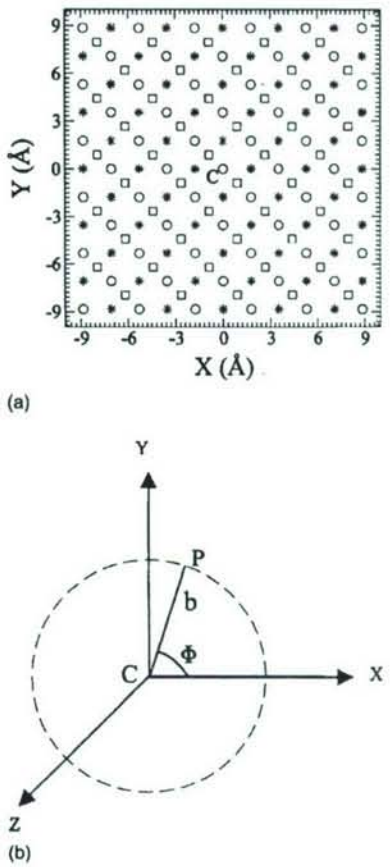


FIG. 1. (a) The projection of the top three layers of carbon atoms of diamond (100) lattice on XY plane: first layer (○), second layer (□), and third layer (*). The central atom C denotes the radical site. (b) A schematic diagram to depict parameters b and ϕ; C denotes the radical site and P represents the point of intersection of the initial C_2 velocity vector and the diamond surface.

III. NEURAL NETWORKS

Using MD simulations, we first computed the reaction probabilities, $P_x(\alpha;\mathrm{MD})$ ($x = C$, S, and D which correspond to chemisorption, scattering, and desorption, respectively), for a given set of input parameters, $\alpha = \alpha(b, \theta, \phi, E_{rot}, E_{trans})$. By running such calculations for different values of α, we have computed N sets of values of $P_x(\alpha;\mathrm{MD})$. These N sets of data points are used to train the two-layer neural network,[1] as shown in Fig. 2.

For training, i.e, for the determination of $W_1^{(x)}$, $b_1^{(x)}$, $W_2^{(x)}$, and $b_2^{(x)}$ matrices, we used the same procedure as described in our previous work[8] except for the use of the tan-sigmoid function in the present study in place of log-sigmoid function. Further, in this study, to obtain the best fit, we chose the number of neurons (n) in the hidden layer to be 50. From such training, for each reaction, we determined $W_1^{(x)}$, $b_1^{(x)}$,

Downloaded 19 Dec 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

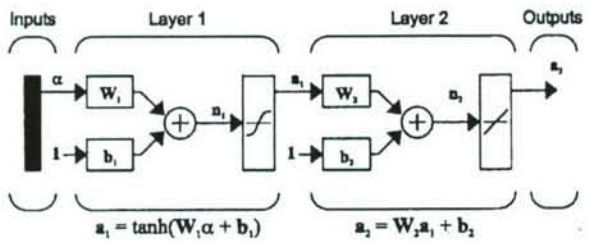


FIG. 2. A schematic illustration of the two-layer neural network used in the present study.

$W_2^{(x)}$, and $b_2^{(x)}$ matrices of dimension $(n\times 5)$, $(n\times 1)$, $(1\times n)$, and (1×1), respectively. These matrices can be used[1] to compute P_x for any desired value of α using the following equations:

$$P_x(\alpha;\mathrm{NN}) = W_2^{(x)} a_1^{(x)} + b_2^{(x)}, \tag{1}$$

where

$$a_1^{(x)} = \tanh[n_1^{(x)}], \tag{2}$$

with

$$n_1^{(x)} = W_1^{(x)}\alpha + b_1^{(x)}. \tag{3}$$

Here, $n_1^{(x)}$, $a_1^{(x)}$, and α represent matrices of dimensions $(n\times 1)$, $(n\times 1)$, and (5×1), respectively, and the elements of matrix α are the input parameters b, θ, ϕ, E_{rot}, and E_{trans}. It may be added that for the sake of better accuracy of the neural network at the time of training, we have linearly scaled[8] the input and output parameters in the range of -1 to 1; after computing $P_x(\alpha;\mathrm{NN})$ using Eq. (1), we performed the reverse scaling procedure to obtain the actual $P_x(\alpha;\mathrm{NN})$.

IV. RESULTS AND DISCUSSION

A. Various reactions

We noted the occurrence of the following reactions:

$$* + \mathrm{C_2} \rightarrow * - \mathrm{C_2} \quad \text{(chemisorption)} \quad [\text{reaction } (\mathbf{R_1})],$$

$$* + \mathrm{C_2} \rightarrow * + \mathrm{C_2} \quad \text{(scattering)} \quad [\text{reaction } (\mathbf{R_2})],$$

and

$$* + \mathrm{C_2} \rightarrow * \text{-----} \mathrm{C_2} \rightarrow * + \mathrm{C_2} \quad \text{(adsorption and desorption)}$$

$$[\text{reaction } (\mathbf{R_3})].$$

The bonds not specified in these equations remain unchanged. Here * refers to the central carbon atom on the top surface. It may be noted that no hydrogen atom is bonded with this central carbon atom. It is, therefore, a radical site.

In addition to these channels, we have also noticed a reaction in which C_2 is inserted into two surface carbon atoms:

$$\mathrm{H} - *' + \mathrm{C_2} + *'' - \mathrm{H} \rightarrow \mathrm{H} - *' - \mathrm{C} - \mathrm{C} - *'' - \mathrm{H}$$

$$[\text{reaction } (\mathbf{R_4})].$$

Here $*'$ and $*''$ denote two neighboring carbon atoms on the diamond surface, and H denotes a hydrogen atom bonded with the carbon atom. All other bonds remain unchanged. It has been noted that reaction (R_4) occurs with very low probability. Therefore, we confine our attention to reactions (R_1)–(R_3) only.

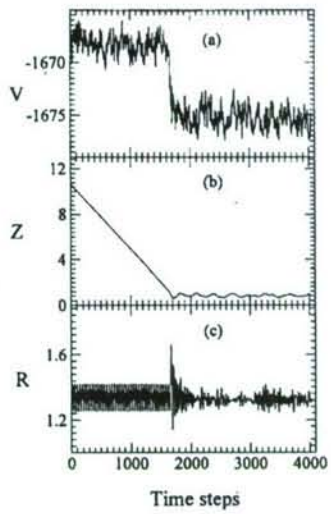


FIG. 3. (a) The potential energy (V) of the system (in eV), (b) Z coordinate of the center of mass of carbon dimer (in Å), and (c) the internuclear distance (R) between the carbon atoms of the dimer (in Å), as a function of the integration time step for a typical trajectory leading to chemisorption. One time step corresponds to 0.5 fs.

The potential energy of the system, the Z coordinate of the center of mass of C_2, and the dimer bond length as a function of time for a trajectory undergoing reaction (R_1) are shown in Fig. 3. We note a drop in the potential energy due to chemisorption, a drop in the value of Z, and a modification in the vibrational frequency and amplitude of C_2 as the chemisorption takes place. The plots for a trajectory undergoing reaction (R_3) are shown in Fig. 4. From Fig. 4(b) we note that Z first decreases, reaches a minimum, remains nearly constant for some time, and then begins to increase due to reverse journey of C_2. If we compare the potential variation given by Figs. 3(a) and 4(a), we clearly note the difference that in the case of reaction (R_3) the change in the potential energy of the system is very small compared to

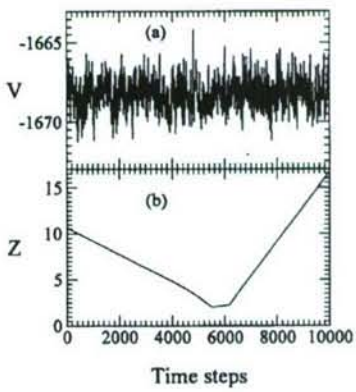


FIG. 4. (a) The potential energy (V) of the system (in eV) and (b) Z coordinate of the center of mass of carbon dimer (in Å), as a function of the integration time step for a typical trajectory leading to reaction (R_3). One time step corresponds to 0.5 fs.

Downloaded 19 Dec 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

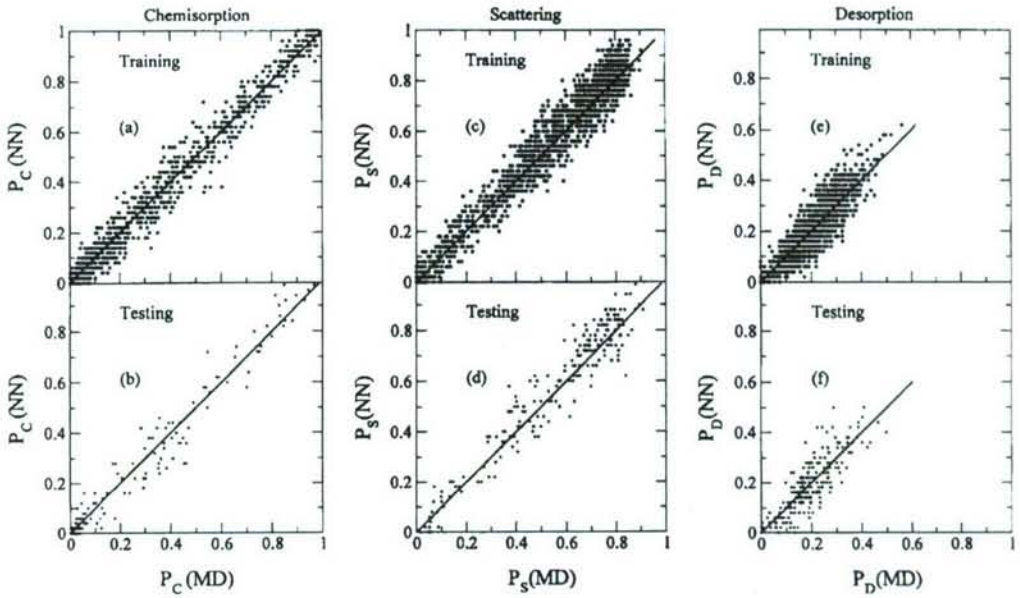


FIG. 5. The probabilities of an event (chemisorption/scattering/desorption) given by neural network vs the corresponding probabilities given by molecular dynamics for training and testing data sets: [(a) and (b)] chemisorption, [(c) and (d)] scattering, and [(e) and (f)] desorption.

reaction (R_1). For reaction (R_2), we also obtain curves (not shown) similar to those in Fig. 4, except that Z starts rapidly increasing after reaching its minimum value.

B. Training and testing of the neural networks

1. Training by a single set of data

We have computed the probability, $P_x(\alpha;\mathrm{MD})$ [$x=C$, S, and D corresponding to reactions (R_1)–(R_3)], of a reaction for a given set of input parameters, $\alpha=\alpha(b,\theta,\phi,E_{\mathrm{rot}},E_{\mathrm{trans}})$, by running 50 trajectories during which we average over other variables including the initial phases of vibration of C_2 and the lattice, and the initial orientation of C_2 bond vector. The vibrational energy of C_2 and temperature of the lattice are kept fixed in all the calculations reported in this work. We computed $P_x(\alpha;\mathrm{MD})$ for 1900 different values of α. Out of these 1900 data points, 1500 data points are used for training, 200 for validation, and 200 for testing. We used the training set to train a NN to predict the probabilities of reactions for all values of α, within the range of α of the training set. We also employed an early stopping procedure and validation data set to prevent overfitting.[41]

The initial success of the NN training is shown in Fig. 5 (top plots) where we plot 1500 values of $P_x(\alpha;\mathrm{NN})$ given by the NN versus the corresponding probabilities, $P_x(\alpha;\mathrm{MD})$, given by the trajectory calculations which were used to train the neural network. Next, for 200 values of α, we compute $P_x(\alpha;\mathrm{NN})$ using Eqs. (1)–(3) and compare these values with the corresponding values given by MD simulations (see bottom plots of Fig. 5). The results of the testing set are very similar to those for the training set, which show good generalization capability of the network.

For the purpose of the following discussion, it is appropriate to describe the relationships among three related reaction probabilities: $P_x(\alpha;\mathrm{MD})$, $P_x(\alpha;\mathrm{NN})$, and $P_x(\alpha)$. (For brevity, when the specification of some labels is not essential, we may ignore one or more of the labels among x, α, MD, and NN). $P_x(\alpha;\mathrm{MD})$ is the reaction probability for process x computed from the trajectory results using

$$P_x(\alpha;\mathrm{MD}) = N_x/N_T, \tag{4}$$

where N_x is the number of trajectories that resulted in reaction x and N_T is the total number of trajectories computed. $P_x(\alpha)$ is the underlying true, but unknown, classical reaction probability for process x on the potential-energy surface used in the calculations. These two probabilities are related by

$$P_x(\alpha) = \lim_{N_T\to\infty} P_x(\alpha;\mathrm{MD}). \tag{5}$$

When N_T is finite, there will be a random statistical fluctuation present in $P_x(\alpha;\mathrm{MD})$ that will cause its value to deviate from $P_x(\alpha)$. This fluctuation will be such that ~95% of the time, we expect to have

$$P_x(\alpha) - 2\Delta \leq P_x(\alpha;\mathrm{MD}) \leq P_x(\alpha) + 2\Delta, \tag{6}$$

where[42]

$$\Delta = [P_x(\alpha)\{1 - P_x(\alpha)\}/N_T]^{1/2}. \tag{7}$$

Finally, $P_x(\alpha;\mathrm{NN})$ is the reaction probability predicted by the NN for process x [see Eq. (1)].

As mentioned earlier, Figs. 5(a)–5(f) show plots of $P_x(\mathrm{MD})$ vs $P_x(\mathrm{NN})$ for chemisorption, scattering, and desorption, respectively, with $N_T=50$ in the training set (top plots) and testing set (bottom plots). If the agreement between $P_x(\mathrm{MD})$ and $P_x(\mathrm{NN})$ were to be perfect, all points would lie on the 45° lines shown in the plots. In the analysis that follows, we shall demonstrate that most of the deviation of the plotted points from the 45° line is due to the statistical variations present in $P_x(\mathrm{MD})$ rather than to inaccurate inter-

Downloaded 19 Dec 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

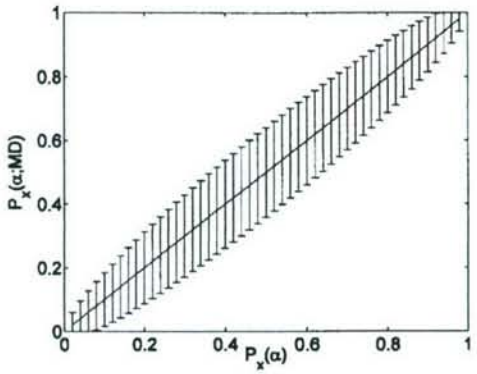


FIG. 6. The expected statistical spread of the probabilities $P_x(\alpha;\mathrm{MD})$ as given by Eq. (6) vs the true probabilities $P_x(\alpha)$.

polation of the neural net. This conclusion is based on the variation of $P_x(\alpha;\mathrm{MD})$ with $P_x(\alpha)$ computed using Eqs. (6) and (7). This result is shown in Fig. 6.

A comparison of the spread of data from the ideal 45° line in Fig. 5(a) or Fig. 5(b) with that shown in Fig. 6 shows that they match; 95% of the points in these curves are within distances of 0.08, 0.10, and 0.11, respectively, from the 45° line. This shows that the expected statistical spread of $P_x(\alpha;\mathrm{MD})$ values is very close to that observed in the plots of $P_x(\alpha;\mathrm{MD})$ vs $P_x(\alpha;\mathrm{NN})$ shown in Fig. 5. In effect, $P_x(\alpha;\mathrm{NN})$ is fitting $P_x(\alpha)$ and not the statistical fluctuations in $P_x(\alpha;\mathrm{MD})$. This observation indicates that neural net fitting of the MD results provides a rapid and effective means of averaging out the statistical fluctuations present in $P_x(\alpha;\mathrm{MD})$ when N_T is not very large.

The NN with an early stopping procedure[41,43] tends to fit the trend and not the statistical fluctuations. This observation is not a new one. It has been shown that the number of training iterations corresponds to the complexity of the neural network function.[44] Therefore, as training continues, the complexity of the network function increases and it becomes more likely that the network will begin to fit statistical fluctuations, rather than the underlying true probability function. By determining when the error on the validation set begins to increase, we can stop the training at an optimal point, before overfitting occurs. It is trivial to visualize that with overfitting, in Fig. 5, the spread in the training plots would shrink and that in the testing plots would increase.

To check further that the spreads in the plots of Fig. 5 are mainly due to the statistical variability associated with the results of the MD calculations, we compute $P_x(\alpha;\mathrm{MD})$ by running 500 trajectories instead of 50, for each value of α. This is expected to decrease the statistical fluctuations in $P_x(\alpha;\mathrm{MD})$ by a factor of 3.2. The plots of such probabilities, $P_x(\mathrm{MD})$ vs $P_x(\mathrm{NN})$, are shown in Fig. 7. It may be mentioned that the $P_x(\mathrm{NN})$ values have been determined here by using the same sets of W_1, W_2, b_1, and b_2 as we used to obtain Fig. 5. The decreased spread of the points around the 45° line suggests that the same NN, which was obtained by fitting the less accurate MD data, is in better agreement with the more accurate MD data. This point again indicates that the NN fitting procedure involving early stopping fits the

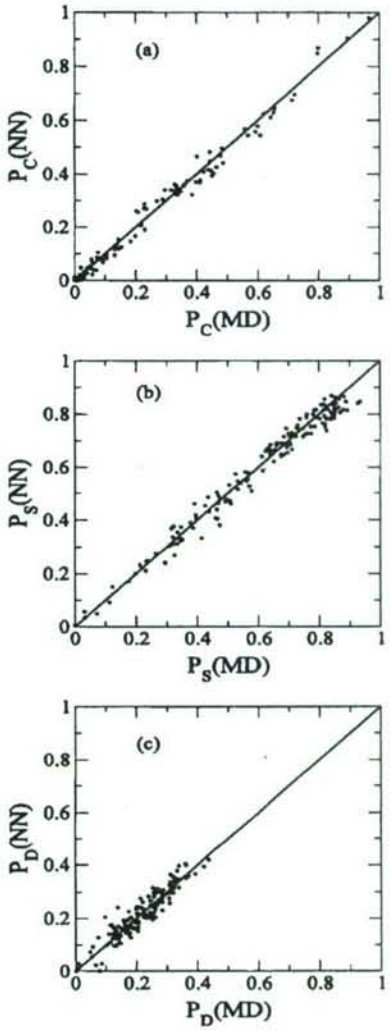


FIG. 7. The probabilities of an event (chemisorption/scattering/desorption) given by neural networks vs the corresponding probabilities given by molecular dynamics using sets of 500 trajectories: (a) chemisorption, (b) scattering, and (c) desorption.

trend of the input data and not the statistical fluctuations associated with the input data. This implies that statistical error present in the MD results may be most conveniently reduced using NNs rather than by computing additional trajectories.

A comparison of the spread in Figs. 5(b) and 7(a) shows that there has been a decrease in the spread by a factor of 2.0. The reduction in the spread by a factor of 2.0, instead of 3.2, indicates the effect of error ($\varepsilon_{\mathrm{NN}}$) associated with the neural network fitting.

To estimate the deviation associated with the NN fitting, we employed the same 1900 data points 50 times to get 50 different trained NN such that each network has been trained with different random initial weights. In two-sigma limit, the variation of such 50 values of $P_x(\alpha;\mathrm{NN})$ from the mean value for each of the 1900 values of α has been determined. The average of such variations is found to be equal to 0.04.

Thus, we may consider two limiting cases. When the

Downloaded 19 Dec 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

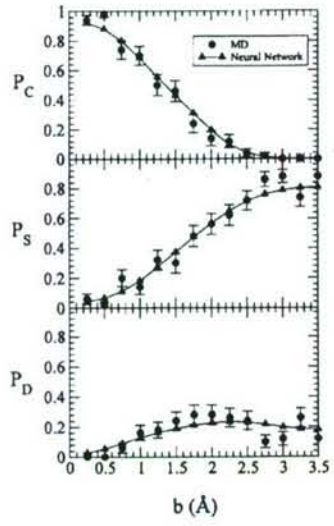


FIG. 8. The variation of chemisorption probability (P_C), scattering probability (P_S), and desorption probability (P_D) as a function of impact parameter (b), for E_{trans}=0.06 eV, E_{rot}=0.052 eV, θ=17°, and ϕ=310°. The triangles (▲) joined by the line denote the NN ensemble predictions and the circles (●) represent the MD data. The error bars associated with the MD points correspond to one-sigma limit.

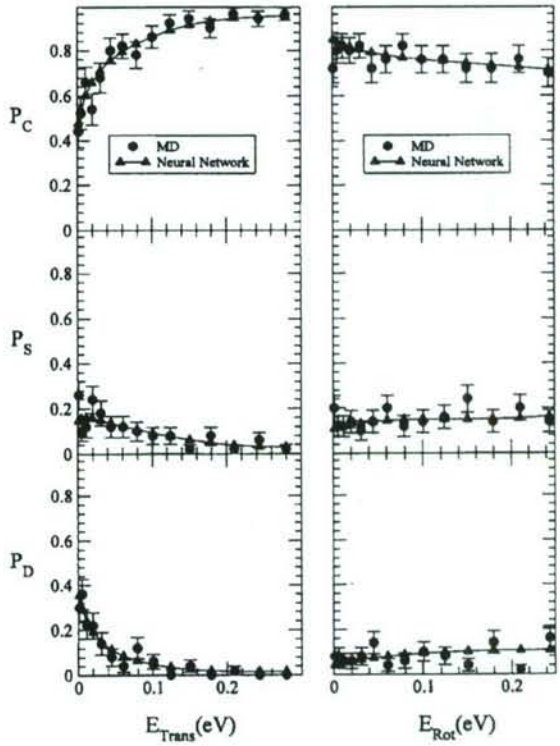


FIG. 9. Same as Fig. 8 except that these curves show variation with E_{trans} for E_{rot}=0.052 eV, b=1.0 Å, θ=11°, and ϕ=110° (left curves), and that with E_{rot} for E_{trans}=0.06 eV, b=1.0 Å, θ=17°, and ϕ=110° (right curves).

statistical error (ε_{MD}) in the input MD data is very small as compared to ε_{NN}, we would expect some spread in figures similar to Figs. 5(a)–5(f) due to error ε_{NN}. When ε_{MD} is very large as compared to ε_{NN}, then the probabilities predicted by the NN are expected to be more accurate than those given by MD, and the spread of points from the 45° line in a figure similar to Fig. 5 would be close to ε_{MD}. The results shown in Figs. 5(a)–5(f), 6, and 7 show that, for the present application, we have $\varepsilon_{MD} \gg \varepsilon_{NN}$.

2. Ensemble (committee) of networks

To improve the network accuracy, we used the 1900 data points previously described to train an ensemble of f networks.[45] In our computations, we used f=50. Each network in the ensemble was trained using a different random partitioning of the data into 1600 training data and 300 validation data. In addition, each network was trained with different random initial weights. Thus, for f networks in the ensemble, we obtain f sets of $W_1^{(x)}$, $b_1^{(x)}$, $W_2^{(x)}$, and $b_2^{(x)}$ matrices. After all networks in the ensemble have been trained, we apply the same input to each network and average all the responses. The results obtained using this method are given in Sec. IV B 3. The ensemble average produces a more accurate response than any one member of the ensemble.[45] It averages out the variations in individual network responses due to randomness in sampling of the data and randomness in the choice of initial weights and biases provided by training.

The results obtained for the C_2-diamond (100) surface system are examples of the advantage provided by analysis of the data using NN ensembles. Equations (6) and (7) show that the two-sigma limit statistical uncertainty in values of $P_x(\alpha;\text{MD})$ is ±28.3% when $P_x(\alpha)$ is 0.50 and N_T is 50. In contrast, the corresponding statistical uncertainty present in the results obtained using the NN ensemble is about ±8%. This statistical error reduction of a factor of about 3.5 is achieved with very little additional computational effort.

3. Dependence of reaction probabilities on input parameters

A major advantage of NNs can be realized by determining the probability of $P_x(\alpha;\text{NN})$ corresponding to any value of α within the range of b, θ, ϕ, E_{rot}, and E_{trans} for which the NN has been trained. For example, in almost negligible time (as compared to the time required to run tens of thousands of trajectories) we can find the dependence of $P_x(\alpha;\text{NN})$ on any variable, say b, for any fixed value of θ, ϕ, E_{rot}, and E_{trans}. As an example, Figs. 8(a)–8(c) show such variations of P_C, P_S, and P_D as a function of b for θ=17°, ϕ=310°, E_{rot} =0.052 eV, and E_{trans}=0.06 eV given by the NN ensemble. For comparison, the results obtained by trajectory calculations are also shown in the figures. We note that the agreement between the NN and MD results is very good. Similarly, the computed MD results and NN predictions of variation of P_C, P_S, and P_D as a function of E_{trans} and E_{rot} for fixed values of other parameters are compared in Fig. 9, and the results shown in Fig. 10 give the corresponding curves when θ and ϕ are varied. All these curves show that the agreement between the MD results and NN predictions is very good.

It is trivial to exhibit the utility of such a trained NN. As an example, in Fig. 11 we show the variation of P_C, P_S, and

Downloaded 19 Dec 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

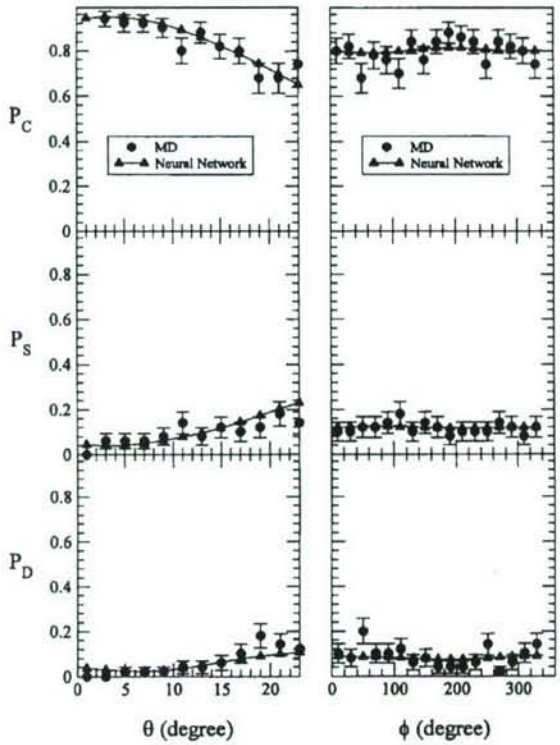


FIG. 10. Same as Fig. 8 except that these curves show variation with θ for E_{trans}=0.124, E_{rot}=0.052 eV, b=1.0 Å, and ϕ=110° (left curves), and that with ϕ for E_{trans}=0.06 eV, E_{rot}=0.052 eV, b=1.0 Å, and θ=11° (right curves).

P_D with angle θ predicted by the NN ensemble for different values of impact parameters for some fixed values of ϕ, E_{rot}, and E_{trans}. It is easy to visualize that any number of such curves for the dependence of P_C, P_S, and P_D on any of the parameters can similarly be obtained in a CPU time negligible compared to the time required to obtain the same results by explicit MD calculations.

V. SUMMARY AND CONCLUSIONS

We have computed the probabilities of various reactions associated with the chemical-vapor deposition (CVD) of carbon dimers on a diamond (100) surface using MD simulations. The probabilities of chemisorption, scattering, and desorption as a function of input parameters (rotational energy, translational energy, angles θ and ϕ associated with the incident direction of the initial velocity vector of the carbon dimer, and impact parameter) have been employed to train and test the neural networks (NNs). The results of the study can be summarized as follows.

(1) By using a large number of data, the values predicted by the trained NN are found to be in very good accord with those provided by MD.
(2) It has been noted that after training the NN, the determination of the interpolated probabilities as a function of the input parameters involves only the evaluation of simple analytical expressions. Thus, the present study shows that interpolation by using a NN can significantly reduce the burden of performing a large number of expensive MD calculations.
(3) It has been observed that the NN fitting procedure involving an early stopping procedure fits the trend of the input data and not the statistical fluctuations associated with the input data. Thus, NNs can also provide a computationally convenient method of averaging out statistical variations inherent in the MD calculations.

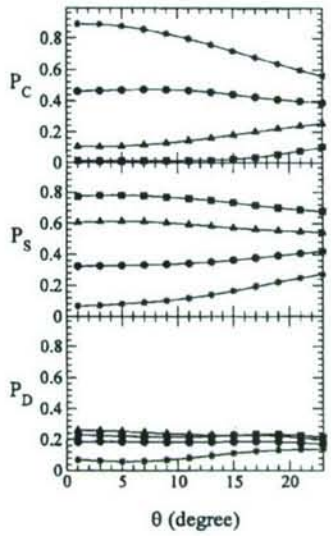


FIG. 11. The variation of chemisorption probability (P_C), scattering probability (P_S), and desorption probability (P_D) predicted by the NN ensemble as a function of angle θ for different values of impact parameter: 1.0 (*), 1.5 (●), 2.0 (▲), and 2.5 Å (■), for E_{trans}=0.06, E_{rot}=0.052 eV, and ϕ=110°.

These observations thus show that NN can be an extremely powerful tool for interpolating the probabilities and rates as a function of different parameters in various chemical reactions.

ACKNOWLEDGMENTS

This project is funded by grants from the National Science Foundation (DMI-0200327) and (DMI-0457663). We thank Dr. W. DeVries, Dr. G. Hazelrigg, Dr. J. Cao, and Dr. D. Durham of the Division of Design, Manufacturing, and Industrial Innovation, Dr. B. M. Kramer, Engineering Centers Division, and Dr. J. Larsen Basse, Tribology and Surface Engineering program for their interest and support of this work. This project was also sponsored by a DEPSCoR grant on the Multiscale Modeling and Simulation of Material Processing (F49620-03-1-0281). The authors thank Dr. Craig S. Hartley and Dr. J. Tiley of the AFOSR for their interest in and support of this work. One of the authors (R.K.) also thanks, A. H. Nelson, Jr. Endowed Chair in Engineering for additional support.

[1] M. T. Hagan, H. B. Demuth, and M. Beale, *Neural Network Design* (PWS, Boston, MA, 1996).
[2] S. Haykin, *Neural Networks: A Comprehesive Foundation*, 2nd ed. (Prentice-Hall, Saddle River, NJ, 1999).
[3] J. Hertz, A. Krogh, and R. G. Palmer, *Introduction to the Theory of*

Downloaded 19 Dec 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

Neural Computation (Addison-Wesley, Reading, MA, 1996).
[4] *Advances in Neural Information Processing Systems* 7, edited by G. Tesauro, D. S. Touretzky, and T. K. Leen (MIT, Cambridge, MA, 1995).
[5] A. Maren, C. Harsten, and R. Pap, *Handbook of Neural Computing Applications* (Academic, New York, 1990).
[6] P. D. Wasserman, *Advanced Methods in Neural Computing* (Van Nostrand Reinhold, New York, 1993).
[7] K. M. Hornik, M. Stinchcombe, and H. White, Neural Networks **2**, 359 (1989).
[8] L. M. Raff, M. Malshe, M. Hagan, D. I. Doughan, M. G. Rockley, and R. Komanduri, J. Chem. Phys. **122**, 084104 (2005).
[9] S. Lorenz, A. Groß, and M. Scheffler, Chem. Phys. Lett. **395**, 210 (2004).
[10] S. Hobday, R. Smith, and J. BelBruno, Nucl. Instrum. Methods Phys. Res. B **153**, 247 (1999).
[11] H. Gassner, M. Probst, A. Lauenstein, and K. Hermansson, J. Phys. Chem. A **102**, 4596 (1998).
[12] D. F. R. Brown, M. N. Gibbs, and D. C. Clary, J. Chem. Phys. **105**, 7597 (1996).
[13] E. Tafeit, W. Estelberger, R. Horejsi, R. Moeller, K. Oettl, K. Vrecko, and G. Reibnegger, J. Mol. Graphics **14**, 12 (1996).
[14] T. B. Blank, S. D. Brown, A. W. Calhoun, and D. J. Doren, J. Chem. Phys. **103**, 4129 (1995).
[15] T. B. Blank and S. D. Brown, Anal. Chem. **277**, 273 (1993).
[16] B. G. Sumpter and D. W. Noid, Chem. Phys. Lett. **192**, 455 (1992).
[17] B. G. Sumpter, C. Getino, and D. W. Noid, J. Chem. Phys. **97**, 293 (1992).
[18] Z. Nami, O. Misman, A. Erbil, and G. S. Mary, IEEE Trans. Semicond. Manuf. **10**, 288 (1997).
[19] R. Rabeau, P. John, and J. I. B. Wilson, J. Appl. Phys. **96**, 6724 (2004).
[20] D. J. Mann, J. Peng, R. A. Freitas, Jr., and R. C. Merkle, J. Comput. Theor. Nanosci. **1**, 71 (2004).
[21] J. Peng, R. A. Freitas, Jr., and R. C. Merkle, J. Comput. Theor. Nanosci. **1**, 62 (2004).
[22] M. Sternberg, P. Zapol, and L. A. Curtiss, Phys. Rev. B **68**, 205330 (2003).
[23] S. W. Yang, X. Xie, P. Wu, and K. P. Loh, J. Phys. Chem. B **107**, 985 (2003).
[24] M. Sternberg, M. Kaukonen, R. M. Nieminen, and T. Frauenheim, Phys. Rev. B **63**, 165414 (2001).
[25] D. M. Gruen, P. C. Redfern, D. A. Horner, P. Zapol, and L. A. Curtiss, J. Phys. Chem. B **103**, 5459 (1999).
[26] D. Zhou, D. M. Gruen, L. C. Qin, T. G. McCauley, and A. R. Krauss, J. Appl. Phys. **84**, 1981 (1998).
[27] P. C. Redfern, D. A. Horner, L. A. Curtiss, and D. M. Gruen, J. Phys. Chem. **100**, 11654 (1996).
[28] D. M. Gruen, C. D. Zuiker, A. R. Krauss, and X. Pan, J. Vac. Sci. Technol. A **13**, 1628 (1995).
[29] D. M. Gruen, S. Liu, A. R. Krauss, and X. Pan, J. Appl. Phys. **75**, 1758 (1994).
[30] M. D. Perry and L. M. Raff, J. Phys. Chem. **98**, 4375 (1994).
[31] S. K. Kim, J. M. White, P. M. Agrawal, and D. L. Thompson, J. Chem. Phys. **115**, 7657 (2001).
[32] M. D. Perry and L. M. Raff, J. Phys. Chem. **98**, 4375 (1994).
[33] P. M. Agrawal, D. L. Thompson, and L. M. Raff, Surf. Sci. **195**, 283 (1988).
[34] P. M. Agrawal, L. M. Raff, and D. L. Thompson, Surf. Sci. **188**, 402 (1987).
[35] B. M. Rice, I. NoorBatcha, D. L. Thompson, and L. M. Raff, J. Chem. Phys. **86**, 1608 (1987).
[36] L. M. Raff, I. NoorBatcha, and D. L. Thompson, J. Chem. Phys. **85**, 3081 (1986).
[37] I. NoorBatcha, L. M. Raff, and D. L. Thompson, J. Chem. Phys. **83**, 1382 (1985).
[38] D. W. Brenner, O. A. Shenderova, J. A. Harrison, S. J. Stuart, B. Ni, and S. B. Sinnott, J. Phys.: Condens. Matter **14**, 783 (2002).
[39] M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids* (Clarendon, Oxford, 2002).
[40] H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsterer, and J. R. Haak, J. Chem. Phys. **81**, 3684 (1984).
[41] W. S. Sarle, *Proceedings of the 27th Symposium on the Interface of Computing Science and Statistics*, Pittsburgh, PA, 1995, p. 352 (unpublished).
[42] L. M. Raff and D. L. Thompson, in *Theory of Chemical Reaction Dynamics*, edited by M. Baer (CRC, Boca Raton, FL, 1985), Vol. III.
[43] S. Amari, N. Murata, K. Muller, M. Finke, and H. Yang, IEEE Trans. Neural Netw. **8**, 985 (1997).
[44] J. Sjoberg and L. Ljung, Int. J. Control **62**, 1391 (1995).
[45] A. Krogh and J. Vedelsby, in *Advances in Neural Information Processing System*, edited by G. Tesauro, D. Touretzky, and T. Leen (MIT, Boston, MA, 1995), Vol. 7.

Downloaded 19 Dec 2005 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

PHYSICAL REVIEW B 72, 125206 (2005)

Monte Carlo simulations of void-nucleated melting of silicon via modification in the Tersoff potential parameters

Paras Mal Agrawal,[1] Lionel M. Raff,[2] and Ranga Komanduri[1,*]
[1]*School of Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, Oklahoma 74078, USA*
[2]*Chemistry Department, Oklahoma State University, Stillwater, Oklahoma 74078, USA*
(Received 2 December 2004; revised manuscript received 11 May 2005; published 20 September 2005)

Molecular dynamics simulations of silicon melting reported in the literature using Tersoff's potential give melting temperatures about 50% higher than the experimental value. To address this discrepancy, we have proposed a modification to the values of the parameters of the Tersoff potential. This modification involves a change in the magnitude of 3 of the 12 parameters in the Tersoff potential as an effective means for bringing down the melting point close to the experimental value. The melting point is determined by performing Monte Carlo simulations using the empirical void-nucleated melting procedure. In addition to an agreement of the computed melting point with the experiment, the modified parameters also bring the density of the liquid and the solid at the melting point into good agreement with the experiment without significantly altering the density and energy values of the solid crystal at room temperature. The coordination number and specific heat of the liquid are also found to be in better agreement with the experiment when the modified set of parameters is used. A comparison of density over a wide range of temperatures shows that the density of the solid predicted by the Tersoff potential with the modified parameters is larger than that given by the unmodified parameters. This difference, however, is not appreciable at room temperature; it increases with temperature and is about 1% at the melting point. The computed cohesive energy of the solid, the point defect energy corresponding to a vacancy, and the surface energy values for (100) and (111) surfaces are also found to be nearly the same for the modified as well as the unmodified parameters.

DOI: 10.1103/PhysRevB.72.125206 PACS number(s): 34.20.Cf, 64.70.Dv, 65.20.+w, 65.40.−b

I. INTRODUCTION

The exploration of empirical intermolecular interaction potentials for a system of silicon atoms has been a subject of considerable interest.[1–22] One of the most popular potentials for silicon is the Tersoff potential.[1–4] It has been shown to explain various properties of silicon crystals. However, Cook and Clancy[23] have found, and Yoo *et al.*[24] have recently confirmed, that the Tersoff potential[4] predicts the melting point to be nearly 50% higher than experiment. This observation suggests the necessity of introducing some modification in the parameters of the Tersoff potential that will provide better agreement of the predicted melting point with experiment.

It is well known that molecular dynamics (MD) simulations of melting of a perfect crystal with periodic boundary conditions produce superheating due to the lack of free surface and defects. A large number of computational studies show that the mechanical melting temperature T_s at which a crystal without any defects and free surface melts is about 20% higher than the true thermodynamic melting point T_m. The ratio,

$$f = T_m/T_s \tag{1}$$

is found to be ~0.8 for a large number of systems. For example, the computed values of f for Ar over a wide range of pressure (0–53 GPa) is found to be 0.85 ± 0.03,[25–27] and in the range 0.77–0.81 for Ne,[28] Cu,[29] CH_3NO_2,[30] ammonium nitrate,[31] and ammonium dinitramide.[32] Using kinetic analysis of the homogeneous nucleation behavior for melting of superheated crystals, Lu and Li[33] found f in the range 0.82–0.85 for Ag, Al, Au, Co, Cu, Fe, Mn, Ni, Pd, Pt, and Sb, and lower values of f for metals with low melting points, i.e., 0.78 for Sn and 0.77 for Pb. It is, therefore, of interest to determine if the computed value of f for a covalent crystal, such as Si also falls in the same range.

Another point of interest in connection with the melting of silicon is the testing of an empirical method, namely, void-nucleated melting, for the determination of the melting point. The void-nucleated method for determining T_m is the result of empirical observations in various studies.[25–34] Detailed investigations of defect-nucleated melting of covalent and metallic solids have been reported by Phillpot *et al.*[34] and Lutsko *et al.*[29] The MD simulations of Lutsko *et al.* of a lattice containing nearly 1000 atoms of copper show that an ideal crystal melts at a temperature over 200 K above the thermodynamic melting temperature. Solca *et al.*[25,28] studied the void-nucleated melting method to determine the melting point of Ar and Ne. They noted that with the increase in the void size, the melting-temperature-versus-void-size curve first exhibits a decrease and then attains a plateau region that corresponds to the true melting temperature, and for a void size larger than that corresponding to the plateau region, the melting temperature plummets.

For Ar, at various values of pressure ranging from 0 to 53 GPa, Agrawal *et al.*[26] have confirmed this observation by comparing the melting point so determined with those computed by Zha *et al.*[35] using a thermodynamic method on the same potential. By simulating three types of voids in this study,[26] it has also been noted that melting temperature depends on the number of atoms removed and not upon the number of voids or the shape of the voids. In a

 ©2005 The American Physical Society

subsequent study on the melting of nitromethane, Agrawal *et al.*[30] obtained good agreement between the melting point computed from the plateau region in the melting-temperature-versus-void-size curve and that determined by the method based on the equilibration of a mixture of liquid and solid.[36]

Besides Ar, Ne, and CH_3NO_2, the void-nucleated melting method has also been tested for CH_3COOH,[37] cyclotrimethylene trinitramine (RDX),[38] 1,3,3-trinitroazetidine (TNAZ),[38] ammonium nitrate,[31] and ammonium dinitramide.[32] The method, however, is empirical in that there is no theoretical basis to validate the assumption that the melting temperature in the plateau region is the true melting point. Therefore, it may be interesting to see if this void-nucleated melting method is applicable for a covalent system, such as Si where we have the results of other studies[23,24] for comparison.

In this paper, we report the results of computations of the melting point of silicon using the Tersoff potential[4] and the void-nucleated method. The results provide additional information related to the validity of the void-nucleated method and the value of f for Si. In addition, the study results in a modification of the parameters of the Tersoff potential[4] that gives a melting point in agreement with experiment.

II. COMPUTATIONAL DETAILS

A. Force field

We have investigated the Tersoff potential,[4] which is expressed in terms of 12 parameters as follows:

$$V_{\text{Tersoff}} = (1/2)\sum_{i\neq j} V_{ij}, \tag{2}$$

where

$$V_{ij} = f_c(r_{ij})[A\exp(-\lambda r_{ij}) - b_{ij}B\exp(-\mu r_{ij})], \tag{3}$$

$$b_{ij} = \chi(1+\beta^n\zeta_{ij}^n)^{-1/2n}, \tag{4}$$

$$\zeta_{ij} = \sum_{k\neq i,j} f_c(r_{ik})g(\theta_{ijk}), \tag{5}$$

$$g(\theta_{ijk}) = 1 + c^2/d^2 - c^2/[d^2 + (h-\cos\theta_{ijk})^2], \tag{6}$$

and

$$f_c(r_{ij}) = \begin{cases} 1 & \text{for } r_{ij} < R, \\ \frac{1}{2} + \frac{1}{2}\cos[\pi(r_{ij}-R)/(S-R)] & \text{for } R < r_{ij} < S, \\ 0 & \text{for } r_{ij} > S. \end{cases} \tag{7}$$

The parameters of the potential, as given in Ref. 4 are listed in Table I.

For the purpose of discussion, we shall denote the above-mentioned potential as V_{IV} or the Tersoff-4 potential to distinguish it from other similar potential functions with different sets of parameters given by Tersoff.[1–3] We denote those given by Tersoff in Refs. 1–3 by V_{I}, V_{II}, and V_{III} or Tersoff-1, Tersoff-2, and Tersoff-3 potential, respectively. The notations, Tersoff-2 and Tersoff-3 are consistent with those used by Balamane *et al.*[39]

TABLE I. The modified and unmodified parameters of the Tersoff potential.

Tersoff parameters	Tersoff-4[a]	Tersoff-ARK
A (eV)	1.8308×10^3	1.8308×10^3
B (eV)	4.7118×10^2	4.7118×10^2
λ (Å^{-1})	2.4799	2.4799
μ (Å^{-1})	1.7322	1.7322
β	1.1×10^{-6}	$\mathbf{1.15\times10^{-6}}$
n	0.78734	**0.988**
c	1.0039×10^5	1.0039×10^5
d	16.217	16.217
h	−0.598 25	**−0.745 25**
R (Å)	2.7	2.7
S (Å)	3.0	3.0
χ	1.0	1.0

[a]Reference 4.

B. Monte Carlo simulations and melting

1. NPT simulations

Monte Carlo (MC) simulations at constant pressure (P) and constant temperature (T) for a fixed number of atoms (N) as described in Ref. 40 are performed to simulate melting. A $3\times3\times3$ supercell of Si crystal having 216 atoms has been considered. The cubic simulation box has an edge length of L. Periodic boundary conditions have been used. At every cycle during the MC walk, $N+1$ moves were attempted. These correspond to a random move of each of N atoms and a random change in the size of the simulation box. The coordinates and box size have been varied according to the following standard equations given in Ref. 40:

$$q_i(\text{new}) = q_i(\text{old}) + (2\xi_1 - 1)(\Delta q)_{\max}, \tag{8a}$$

$$L(\text{new}) = L(\text{old}) + (2\xi_2 - 1)(\Delta L)_{\max}, \tag{8b}$$

where ξ_1 and ξ_2 are random numbers chosen uniformly in the interval 0 to 1, and $(\Delta q)_{\max}$ and $(\Delta L)_{\max}$ are the maximum step sizes for the translation of the Cartesian coordinate (q) of an atom and box length (L), respectively. These maximum displacements have been determined to obtain an acceptance ratio of about 50% from the total number of attempted moves and their values were $(\Delta q)_{\max}=0.15$ Å and $(\Delta L)_{\max}=0.13$ Å.

A walk or a change made in the box size is accepted, if either $W<0$ or

$$\exp(-W/k_BT) > \xi, \tag{9}$$

otherwise the move is rejected. In the above relation, k_B is Boltzmann's constant, ξ is a random number chosen uni-

formly in the interval 0 to 1, and W is given by

$$W = P(v_{new} - v_{old}) + (V_{new} - V_{old}) + Nk_BT \ln(v_{old}/v_{new}). \tag{10}$$

Here, the subscripts new and old have the usual meaning, v denotes the volume of the box, V represents the total interaction energy of the system, and P and T signify the external pressure and temperature at which the calculations were made. When a move is rejected, the properties corresponding to the previous configuration were included in the averages. A new configuration was then generated from this previous configuration.

2. Creation of voids and gradual heating

Starting from the ideal configuration at $T=0$ K, an equilibrated configuration at $T=500$ K and $P=1$ atm is obtained by running 10 000 cycles of Monte Carlo simulations. By successive simulations, each for 10 000 cycles, the equilibrated configurations at temperatures 1000, 1500,..., and 3000 K have been obtained. All the simulations have been run at $P=1$ atm. From this set of configurations, the closest configuration at a temperature lower than T_o is simulated for 10 000 cycles to obtain an equilibrated configuration at temperature T_o. After equilibrating the system at temperature T_o, N_{void} atoms have been removed out of the total N_o atoms. The locations of these N_{void} atoms have been chosen randomly with the constraint that the minimum distance between any two is greater than r_{min}, where r_{min} is chosen nearly as large as possible. We choose this number N_{void} to signify the void size.

After the creation of voids, this system of $N=N_o-N_{void}$ atoms is again equilibrated at temperature T_o and pressure P by running MC simulations for n_o cycles. The heating of the system from initial temperature T_o is achieved by uniformly changing the temperature by $\Delta T=0.001$ K, after each cycle of $N+1$ moves. Thus, the temperature at the nth cycle of moves is

$$T = T_o + (n - n_o)\Delta T. \tag{11}$$

The gradual heating of the system in the MC simulations has been achieved by redefining the desired temperature in Eqs. (9) and (10) during constant number-pressure-temperature (NPT) simulations. This method of gradual heating based on redefining the desired temperature in NPT simulations has also been employed earlier for the melting studies of Ar[26] and nitromethane[30] by MD simulations. The success of this procedure has been tested in those studies[26,30] by monitoring the temperature and pressure as a function of integration time; there it has been found that pressure remains constant and the temperature changes according to the desired input temperature. Here in MC simulations we do not have the kinetic energy to monitor the temperature, but we have tested the increase in temperature by monitoring the variation in the potential energy with the MC cycle number during the heating process. Figure 1 depicts this variation in the potential energy per atom with T given by Eq. (11) in a simulation corresponding to $T_o=1500$ K and $\Delta T=0.001$ K/cycle. Each point in the figure has been obtained by averaging the potential energy over 1000 cycles. The straight line in the figure corresponds to the least-squares fit. As expected, we note that the slope of the straight line is equal to $\frac{3}{2}k_B$ that shows that the system is being heated according to the temperature given by Eq. (11).

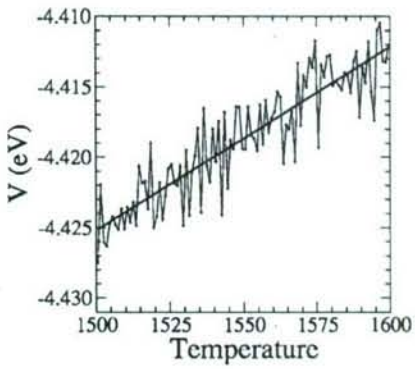


FIG. 1. Variation of potential energy per atom as a function of temperature T given by Eq. (11). The straight line shows the least squares fit.

3. Order parameter

The melting of the system has been identified by monitoring the energy and the order parameter (ζ)[26,30] as a function of temperature during the heating process. At the melting temperature, these parameters change abruptly. Since the diamond structure can be considered as a combination of two fcc structures, we can write ζ as an average of the order parameters of the two structures, i.e.,

$$\zeta = \frac{1}{2}\sum_{i=1}^{2} \zeta_i, \tag{12}$$

where

$$\zeta_i = (1/N_c)\left|\sum_{j=1}^{Nc} \exp(i\mathbf{k}\cdot\mathbf{r_j})\right|. \tag{13}$$

In these equations $N_c \approx N/2$, $\mathbf{r_j}$ is the position vector of the jth atom of ith fcc structure and $\mathbf{k}$ is the reciprocal lattice vector. For the fcc lattice,[26,40]

$$\mathbf{k} = (2\pi/a_o)(-1,1,-1), \tag{14}$$

where a_o is the length of the unit cell. It is trivial to see that for an ordered crystal $\zeta \to 1$ and $\zeta \to 0$ for the liquid state.

C. MD simulations

We employed a larger system, a $5\times5\times5$ supercell of Si having 1000 atoms, to compute density, configurational energy, specific heat, coordination number, and radial distribution function. Such a system has been simulated using isothermal-isobaric molecular dynamics (NPT-MD) method. The Melchionna modification[41] of the Nosé-Hoover equations of motion was used to achieve the constant temperature and pressure. The thermostat, as well as barostat relaxation

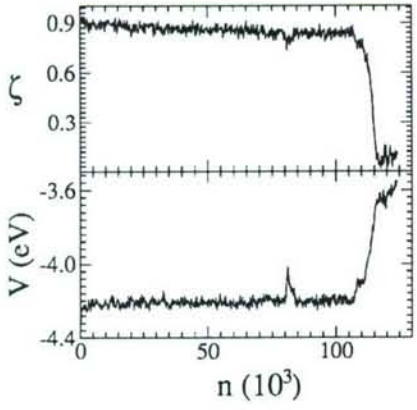


FIG. 2. Variation of order parameter (ζ) and potential energy per atom (V) as functions of Monte Carlo simulation cycle number (n) for the Tersoff-4 potential with initial temperature (T_o)=2950 K, void size (N_{void})=0, and $n_o=10\times10^3$ cycles.

time, has been chosen as 50.0 t.u. (1 t.u.=1.018×10^{-14} s), and the Verlet leapfrog procedure[40] (with time step 0.05 t.u.) has been used to integrate the equations of motion. Computed results are given in Figs. 7–9 and Tables III and V.

III. RESULTS

A. Melting

Figure 2 shows the variation of order parameter ζ and the configurational energy per atom V as a function of the Monte Carlo cycle number n for a system of 216 atoms at P =1 atm initially at temperature T_o=2950 K with no voids. The heating according to Eq. (11) has been started after n_o =10 000 cycles. A rapid change in ζ and V depicted by these curves at $n=107\times10^3$ cycles corresponds to the melting at T=3047 K.

Figure 3 gives the variation of the transition temperature as a function of the void size, N_{void}, defined by the number of atoms randomly removed from the crystal. Multiple points at a value of N_{void} in the figure demonstrate the effect of varying either the initial phase or temperature of the system. This figure shows that the transition temperature decreases as N_{void} increases. A horizontal line in the figure has been drawn to mark the average temperature in the plateau region corresponding to $11\leq N_{void}\leq14$. The average temperature in this region is 2509±44 K. With the assumption that the transition temperature in the plateau region is the melting temperature that corresponds to the thermodynamic melting point, we obtain 2509 K as the melting point given by this void-nucleated method, which is in good agreement with the computed values, 2567 K and 2584 K, given by Yoo *et al.*[24] using the same potential and the method of coexisting solid-liquid phases and 2547±22 K given by Cook and Clancy.[23]

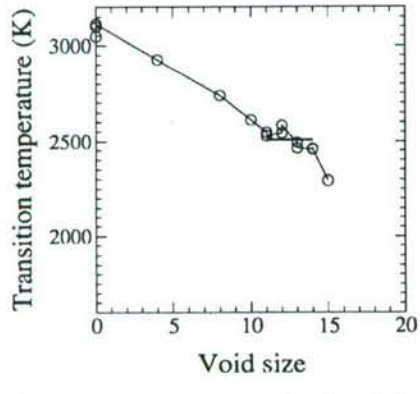


FIG. 3. Transition temperature as a function of the void size (N_{void}) for the Tersoff-4 potential. The horizontal line shows the average temperature (2509 K) in the plateau region.

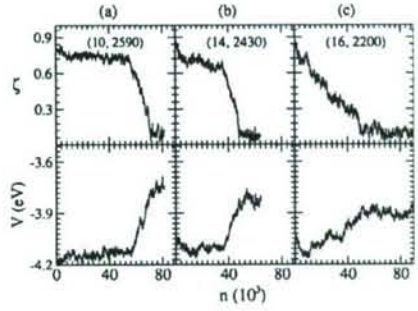


FIG. 4. Same as Fig. 2, but for different values of void size (N_{void}) and initial temperature (T_o): (a) N_{void}=10, T_o=2590 K; (b) N_{void}=14, T_o=2430 K; (c) N_{void}=16, T_o=2200 K.

A few typical curves showing the variation of order parameter (ζ) and configurational energy (V) for N_{void}=10, 14, and 16, respectively, are shown in Figs. 4(a)–4(c). For all these curves, n_o=10 000 cycles and T_o values are 2590, 2430, and 2200 K, respectively, as indicated in the figure. A comparison of the curves corresponding to N_{void}=14 and 16 shows that melting point decreases from 2456 K for void size N_{void}=14 to less than 2200 K for void size N_{void}=16.

For the sake of further discussion, following the previous studies,[25,26,28,30] we introduce a term, "critical void size," to refer to the maximum value of the void size in the plateau region. It may be noted that the critical void size depends on the substance, the interaction potential function, pressure, and the number of atoms (N) considered in the simulation box. For example, the studies of Agrawal *et al.*[26] on the melting of Argon reveal that at a given pressure (P =44.56 kbar) the critical void size increases from 50 to 120 as N increases from 864 to 2040.

In view of various simulation studies on Ar, Ne, CH_3NO_2, RDX, TNAZ, ammonium nitrate, and ammonium dinitramide, it is not surprising to note that the value of the melting point of Si given by the void-nucleated method employed here agrees with the previously reported[23,24] melting point values.

However, one needs to make a rigorous analysis of the appropriateness and limitations of this method. Due to the lack of understanding of the mechanism of melting, the relationship between the existence of the above-mentioned plateau region and the mechanism of melting is not yet known. There have been various studies to explore the mechanism of melting. The Lindemann criterion[42] of melting and the Born

TABLE II. A comparison of melting temperatures given by the modified (V_V) and unmodified (V_{IV}) Tersoff potential parameters with the experiment.

	T_s (K)	T_m (K)	$f=T_m/T_s$
Tersoff-4[a]	3075±29	2509±44	0.82
Tersoff-ARK[b]	2097±21	1711[c]	
Experiment		1687[d]	

[a]Reference 4.
[b]Present modification.
[c]Obtained by using Eq. (1) and $f=2509/3075$.
[d]Reference 49.

instability criterion[43] of melting have been of importance in explaining the mechanism of melting. The investigations of Jin *et al.*[27] show a strong correlation between the Lindemann and the Born instability criteria. The Lindemann criterion based on the mechanical approach of melting has gained more attention as a result of the recent investigations of Burakovsky *et al.*[44] regarding the melting of elements based on the dislocation mechanism.[44–47] Considering melting as a dislocation-mediated phase transition, Burakovsky *et al.*[44] computed melting curves for 24 elements in good agreement with the available data. They also note that their dislocation-based melting approach leads to melting points that are very close to those given by the Lindemann criterion. Gomez *et al.*[47] showed that defects which occur in the solid phase as the transition temperature is approached are responsible for melting. However, we are still in the learning phase in the area of the mechanism of melting. One would like to rigorously learn if the existence of the plateau region in the void-size-versus-melting-temperature curve is universal and whether the melting point in the plateau region corresponds to the thermodynamic melting point.

B. Superheating and *f* factor

The extent of superheating of a crystal can be measured by the ratio f given by Eq. (1). Table II gives the value of the melting point without any void, $T_s=3075$ K, determined by averaging over four values of the computed melting points with different initial conditions. The true or thermodynamic melting temperature $T_m=2509$ K given by the average of the melting temperatures in the plateau region, as discussed in Sec. III A, leads to $f=0.82$.

It is interesting to note that this value of $f=0.82$ for a covalent Si crystal is comparable to that determined by the homogeneous nucleation model of Lu and Li[33] for Al, Au, Co, Cu, Fe, Mn, Ni, Pb, Pd, Pt, Sb, and Sn, and those computed for Ar, Ne, Cu, CH_3NO_2, RDX, TNAZ, ammonium nitrate, and ammonium dinitramide. For all these crystals f is found to be in the range 0.77 to 0.85.

A near constant value of f for a large number of systems suggests the possibility of $f\sim0.80$ for many other substances. Thus, the prediction of f in this range may be an important criterion to judge the validity of a theoretical model. Further, the rigorous determination of f by some theoretical model would also make the determination of T_m easy.

As regards the effect of the size of the simulation box on the factor f that signifies the degree of superheating, Agrawal *et al.*[26] have found that the value of f for a system of argon crystals does not change when the periodic system containing 864 atoms is replaced by that having 2048 atoms. A simulation of 6912 atoms of Ar by Jin *et al.*[27] also gave almost the same value of f. The agreement between the computed values of f and those given by Lu and Li[33] as discussed in the above paragraphs also suggest the possibility of insignificant change in the value of f as the number of atoms in the periodic system considered is increased by a large extent. Also, based on the Lindemann criterion,[42] namely, a solid melts when the amplitude of atomic vibrations exceeds a certain fraction of the lattice spacing, it appears that the melting temperature or superheating of an ideal crystal with the periodic boundary conditions would not change when the system size is increased although the limits of this generalization have not been established.

C. Modification in the potential

1. Addition of long-range interactions

In the Tersoff potential, $f_c(r_{ij})$ occurring in Eq. (3) is assumed to be zero for $r_{ij}>3.0$ Å for a silicon crystal. To examine the effect of long-range interaction, we have added a Lennard-Jones (12-6) interaction,

$$V_{LJ}(r_{ij})=4\varepsilon[(\sigma/r_{ij})^{12}-(\sigma/r_{ij})^{6}], \tag{15}$$

to the Tersoff potential V_{IV} and then determined the melting point. The values of parameters, $\sigma=3.826$ Å and $\varepsilon=0.017\,44$ eV have been taken from Refs. 48 and 49. To avoid strong repulsions due to this term, we considered this interaction only when $r_{ij}>4.0$ Å.

By running a Monte Carlo simulation with the addition of such a long-range interaction, we found that melting occurs at 3116 K when void size=0. If we compare this value of T_s with $T_s=3075\pm29$ K, given by the Tersoff-4 potential alone, we infer that the effect of the addition of van der Waals interactions in the Tersoff potential on the melting is very small. Therefore, we need to look for some more significant modification to obtain a melting point in agreement with the experiment.

2. Choice of new parameters

After running a few sets of simulations by varying the magnitude of 3 of the 12 parameters in the Tersoff potential, it has been observed that a combination of β, n, and h parameters occurring in Eqs. (4) and (6) would give values of the melting point and density of liquid Si in agreement with experiment without introducing appreciable change in the density and configurational energy of a solid Si crystal at low or room temperatures. We have not attempted any change in parameters A, B, λ, μ, and χ as the properties of the solid even at low temperatures may be very sensitive to these parameters. Similarly, we did not investigate the effect of varying R, S, c, and d as these parameters may not be as effec-

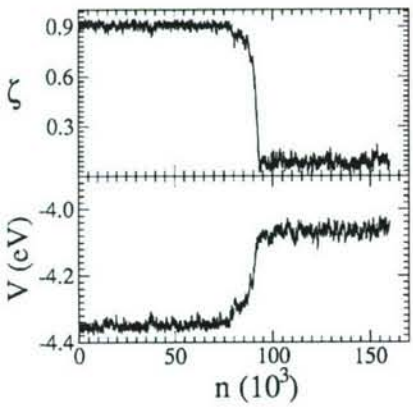


FIG. 5. Variation of order parameter (ζ) and potential energy per atom (V) as functions of Monte Carlo simulation cycle number (n) for Tersoff-ARK potential with initial temperature (T_o)=2000 K, void size (N_{void})=0, and $n_o=10\times10^3$ cycles.

tive, in modifying the melting point as well as the density of liquid without introducing appreciable changes in the density and cohesive energy of the solid at low or room temperatures, as β, n, and h. To achieve this, we first tested various combinations of n and h in a coarse grid without changing the value of β. The values of n and h ranging from ~0.75 to 1.5 times their original values have been attempted to reproduce the melting point. A finer grid of β, n, and h is then explored to reproduce the melting temperature, liquid density, and cohesive energy of the solid. The values of β, n, and h so determined are 1.15×10^{-6}, 0.988, and −0.745 25, respectively (see Table I). For further discussion, let us denote the Tersoff potential with these modified parameters as V_V or Tersoff-ARK (Agrawal-Raff-Komanduri) potential. The results obtained with Tersoff-ARK are discussed in Secs. III C2*a*–III C2*e*.

a. Melting point. Similar to the results shown in Fig. 2 for the Tersoff-4 potential, we get Fig. 5 for the Tersoff-ARK potential. In this simulation T_o=2000 K, n_o=10 000 and, melting occurs at n=79 000 that correspond to melting at 2069 K for a crystal without any void. Figures 6(a)–6(i) show the configurations of the system of 216 silicon atoms at various stages of this simulation. Figures 6(a)–6(c) depict the crystalline phase. Figure 6(d) corresponds to n=79 000 when the crystal starts melting and the order parameter (ζ) just begins to drop (see Fig. 5 for n versus ζ curve for this simulation). Figures 6(e) and 6(f) exhibit the configurations at n =85 000 and 90 000, respectively; these belong to the region of the rapid drop in the order parameter (see Fig. 5). Figures 6(g)–6(i) represent configurations at n=95 000, 100 000, and 105 000, respectively; this region corresponds to the liquid characterized by low values of the order parameter (see Fig. 5).

By averaging the results on a set of four such simulations with different initial conditions, we get the mechanical melting temperature as $T_s=2097\pm21$. Using Eq. (1) and assuming the value of $f=T_m/T_s=2509/3075$, as described in Sec. III B, we obtain the melting temperature, T_m=1711 K (see Table II), which is in good agreement with the experimental value 1687 K.[49]

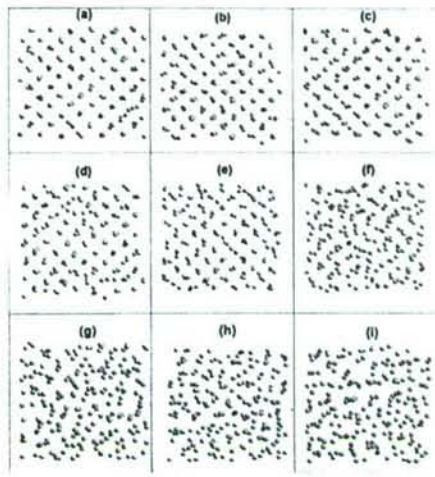


FIG. 6. The configuration of silicon atoms at various stages of a MC simulation at n=(a) 1000, (b) 40 000, (c) 75 000, (d) 79 000, (e) 85 000, (f) 90 000, (g) 95 000, (h) 100 000, and (i) 105 000 for Tersoff-ARK potential with initial temperature (T_o)=2000 K, void size (N_{void})=0, and $n_o=10\times10^3$ cycles.

b. Density. Figure 7 compares the density of a solid and a liquid for the Tersoff-ARK (solid curve) and the Tersoff-4 (dotted curve) potentials. We note that at low temperatures, the densities of the solids given by V_{IV} and V_V are in excellent agreement. However, the density values predicted by V_V are found to be larger than those given by V_{IV} at higher temperatures. The difference increases with temperature. At the melting point, this difference is about 1%. For the liquid, we see a large difference in the densities predicted by these two potentials. At 1687 K, the density of liquid predicted by V_V is found to be 2.589±0.013 g cm^{-3}, which is in excellent agreement with the experimental density 2.583 g cm^{-3}.[50] V_{IV}, however, gives a density of 2.225 g cm^{-3} for the supercooled liquid at this temperature. It may be noted that for the Tersoff-ARK potential at 1687 K, the ratio, $\Delta V/V_s$, of change in volume Δv to the volume of solid V_s at the phase transition is equal to −11.8%, which is in excellent agreement with the experimental value −11.9%.[51]

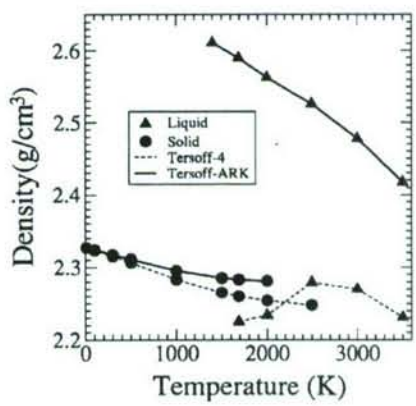


FIG. 7. The variations in densities of solid (●) and liquid (▲) as a function of temperature: solid curves for Tersoff-ARK potential and dotted curves for the Terosoff-4 potential.

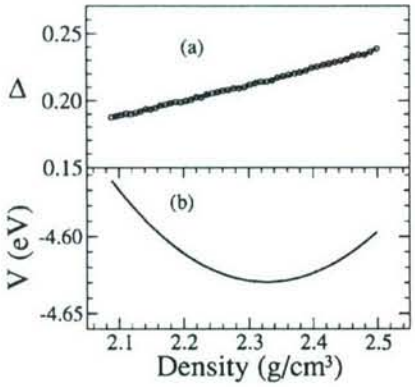


FIG. 8. (a) The difference in energy Δ as defined by Eq. (16); (b) the value of the configurational energy per atom, as a function of density for silicon in the ideal tetrahedral configuration at $T=0$ K.

c. Energy. We have compared configurational energy of Si in the solid phase in two ways: (i) At $T=0$ for different values of pressure or density, and (ii) at $P=0$ for different values of temperatures. In Fig. 8(a), we have shown the difference in the two potentials by a parameter Δ defined as

$$\Delta = 10^5(V_V - V_{IV})/V_{IV}, \tag{16}$$

where V_V and V_{IV} are computed for an ideal tetrahedral geometry at $T=0$ at different values of pressure, which can be specified by the density of the crystal. We note that for $\Delta < 0.25$, i.e., the two potentials differ by less than 0.000 25% over a wide range of pressure, at $T=0$. In this connection, it may also be interesting to see the behavior and absolute value of V_V per atom in this configuration; Fig. 8(b) gives these data.

Figure 9 compares the configurational energy given by the two potentials at various temperatures. It shows the variation of configurational energy per atom as a function of temperature for Si in solid and liquid phases for the Tersoff-ARK (solid curve) and the Tersoff-4 (dotted curve) potentials at $P=0$. In the case of the solid, the results of V_{IV} and V_V differ by less than 0.001 eV and, therefore, the dotted curve corresponding to the potential V_{IV} coincides with the solid curve. The slope of the kinetic-energy-versus-temperature curve for the classical model is trivially 1.5 k_B. By summing the slope of the potential and kinetic energies, we can obtain the specific heat C_p. It is interesting to note that in this classical simulation, the average slope of the potential energy curve for the solid is nearly equal to that given by the kinetic energy curve. The average value of C_p in the temperature range 100–2000 K for the Tersoff-ARK potential as well as the Tersoff-4 potential is found to be 25.4 J mol^{-1} K^{-1}, which is in good agreement with the experimental values[52] at high temperatures: 25.359, 27.196, and 28.870 J mol^{-1} K^{-1} at T =800, 1200, and 1600 K, respectively.

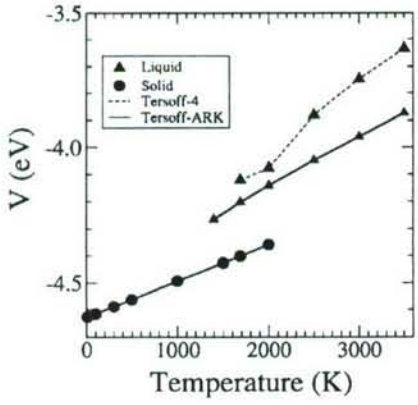


FIG. 9. The configurational energy values of solid (●) and liquid (▲) as functions of temperature: solid curves for the Tersoff-ARK potential and dotted curves for the Terosoff-4 potential.

As shown in Fig. 9, in the case of the liquid, we note a large difference between the configurational energy values given by the two models. We also note significant variation in the slopes with temperature. With respect to the average specific heat in the range 1687 to 2000 K, we obtain C_P =31.0 J mol^{-1} K^{-1} for V_V, that given by V_{IV} in the temperature range 2500–3000 K is 38.5 J mol^{-1} K^{-1}, against the experimental value of 29.2 J mol^{-1} K^{-1} at T=1687 K (see Table III).

From the information given in Fig. 9, we can determine the heat of fusion as 19.3 kJ mol^{-1} for the Tersoff-ARK potential, which is in poor agreement with the experimental value,[53] 50.6 kJ mol^{-1} (see Table III). This disagreement suggests the need for further improvement in the potential.

Using MD simulations, the configurational energy (V) as well as the density for a liquid and a solid at various temperatures have also been computed using the Tersoff-ARK potential on the periodic system having 216 atoms in the simulation box. The computed values so obtained are found to be in excellent agreement with the corresponding values reported in Figs. 7 and 9 for the periodic system having 1000 atoms. Regarding the size dependence of the computed melting point, Miranda and Antonelli[60] recently reported an agreement between the values of the melting point of silicon given by the periodic systems having 216 and 512 atoms in the simulation box. From the results presented here, together with the other results available in the literature, it appears that the values of the melting point, density, and configurational energy determined in the present studies are size independent, at least in the range of atoms considered; although, the limits of this generalization have yet to be established. Whether or not the degree of overheating of the periodic system with an extremely large number of atoms (say on the order of a billion or more) would change due to the change in the lattice vibrations remains to be seen. Admittedly, simulations of such large systems require very fast, highly powerful computers with adequate memory. The availability of such systems (single systems or with parallel processing capability) in the not too distant future can answer this question more definitely.

d. Point defect and surface energies. In Table IV, we have reported the computed values of the point defect and surface energy values given by the Tersoff-4 and Tersoff-ARK potentials for vacancy, hexagonal interstitial, and tetrahedral interstitial defects, and (100) and (111) surfaces. For determining the point defect energy values, we have considered a

TABLE III. Comparison of various potentials for Si (all simulation values are reported at the melting point of the respective potential model).

	MEAM[a]	SW[b]	Tersoff-4[c]	Tersoff-ARK[c]	Expt.
Melting point (K)	1475	1691	2509	1711	1687[d]
Density of liquid ($g\ ml^{-1}$)	2.302	2.459	2.225	2.589	2.583[e]
Heat of fusion at melting ($kJ\ mol^{-1}$)	35.0	30.9	39.7	19.3	50.6[b]
C_P of liquid ($J\ mol^{-1}\ K^{-1}$)	32.3	35.3	~38.5	~31.0	29.2[b]
Liquid coordination number	6.0–8.2	5.5–6.2	~4.7	~5.4	~6.4[b]

[a]Data taken from Ref. 23.
[b]Data taken from Ref. 53.
[c]Present work.
[d]Data taken from Ref. 49.
[e]Data taken from Ref. 50.

system of 216 atoms with the periodic boundary conditions and have added or subtracted one atom to make it an interstitial defect or vacancy. The system is allowed to relax before computing the energy. For a comparison, the point defect energy values computed by Tersoff[3] on the Tersoff-3 potential and the recent density functional theory (DFT) results given by Goedecker *et al.*[54] and Leung *et al.*[55] are also listed in the table.

From the data listed in Table IV, we note that the values of the vacancy energy given by Tersoff-3, Tersoff-4, and Tersoff-ARK are equal and are comparable to that obtained by Goedecker *et al.*[54] under the local density approximation (LDA). We also note that the tetrahedral interstitial energy value given by the Tersoff-4 potential is ~14% higher and that by the Tersoff-ARK potential is ~36% lower than the corresponding LDA value determined by Leung *et al.*[55] As regards the hexagonal interstitial energy, the value given by the Tersoff-4 potential is ~39% higher and that by the Tersoff-ARK potential is ~15% lower than the corresponding LDA value given by Leung *et al.* It may also be noted that the value of the hexagonal interstitial energy given by the Tersoff-ARK potential is in excellent agreement with the corresponding LDA value determined by Goedecker *et al.*[54]

TABLE IV. Point defect and surface energy values.

	Tersoff-3	Tersoff-4	Tersoff-ARK	Other
Point defect energy (eV)				
Vacancy	3.7[a]	3.7	3.7	3.17,[b] 3.56[c]
Interstitial (tetrahedral)	3.8[a]	3.9	2.2	3.43[d]
Interstitial (hexagonal)	4.7[a]	4.6	2.8	3.31[b,d] 2.87[c]
Surface energy ($eV/Å^2$)				
Si(100)	0.144[e]	0.144	0.142	
Si(111)	0.080[e]	0.080	0.078	

[a]Reference 3.
[b]Reference 54; DFT calculation using general gradient approximation (GGA).
[c]Reference 54; DFT calculation using LDA.
[d]Reference 55.
[e]Reference 39.

To compare the surface energy values with the present results for the Tersoff-4 and Tersoff-ARK potentials, we have included the results of Balamane *et al.*[39] for the Tersoff-3 potential for Si(100) and Si(111) surfaces in Table IV. We note that the surface energy values given by the Tersoff-4 potential are equal to the corresponding values given by the Tersoff-3 potential, and the values given by the Tersoff-ARK potential are ~2% lower. It may be noted that all six values listed in the table for surface energy correspond to the ideal unrelaxed configuration.

The agreement between the Tersoff-4 and Tersoff-ARK potentials in predicting surface energy as well as vacancy energy, as listed in Table IV, again shows that the Tersoff-4 and Tersoff-ARK potentials give nearly the same results when the deformation in the bond angles and bond lengths is small. The higher values of hexagonal interstitial energy as well as tetrahedral interstitial energy given by the Tersoff-4 potential show that as compared to the Tersoff-ARK potential, the Tersoff-4 potential makes the deformation more difficult; qualitatively, this observation is consistent with the fact that the melting point given by the Tersoff-4 potential is higher than that given by the Tersoff-ARK potential.

e. Other points. Figure 10 compares the radial distribution function (RDF) of liquid Si at 3000 K given by V_V (solid curve) and V_{IV} (dotted curve). Table V compares the locations of the first and next peaks, the height of the first peak, and the full width at half maximum (FWHM) of the first peak. For comparison, the values given in Ref. 3 for V_{III} and those given by the experiment[56,57] are also reported in the table. We note that there is a good agreement among all four sets of numbers.

The coordination number for the solid crystal, given by V_{IV} as well as V_V, is found to be 4.0, which is in agreement with the tetrahedral structure. For the liquid, however, we

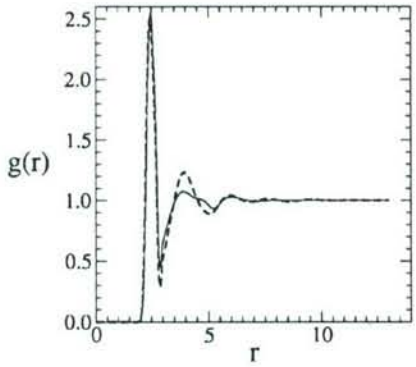


FIG. 10. The RDF for liquid Si at 3000 K: solid curve for the Tersoff-ARK potential and dotted curve for the Tersoff-4 potential.

note that the coordination number ~5.4, given by V_V, which is higher than the value ~4.7 given by V_{IV} but is lower than the experimental value ~6.4 (see Table III).

Keblinski *et al.*[58] remark that with the exception of the Stillinger and Weber (SW) potential[5] and a similar potential developed by Mistriotis *et al.*,[21] among more than 30 empirical potentials for silicon, no potential other than the environmental-dependent interatomic potential (EDIP) developed by Justo *et al.*[16] predicts a reasonable melting point. Recently, there has been a number of studies on the computation of the melting point of Si using the potential of Justo *et al.* The melting point given by this potential is found to be 1572 K by Kaczmarski *et al.*,[59] 1582±25 K by Miranda and Antonelli,[60] and 1520±30 K by Keblinski *et al.*,[58] all within 10% of the experimental value, 1687 K. For comparison, the value of the melting point given by the Tersoff-ARK potential is 1711 K. With respect to the heat of fusion and density, the EDIP potential gives[59] a heat of fusion equal to 36.0 kJ mol^{-1} and, contrary to the experiment, it predicts[59] a decrease in density of Si as it undergoes the phase transition from the solid phase to the liquid phase.

It would be interesting to compare various properties of liquid silicon as predicted by the Tersoff-4,[4] Tersoff-ARK, modified-embedded-atom-method (MEAM),[8,23] and Stillinger and Weber (SW) potentials[5] (see Table III for details). First, we note that none of these models, including Tersoff-ARK, provides very good agreement of all the properties listed with the experimental values. Second, compared to the Tersoff-4 potential, Tersoff-ARK yields better agreement with experiment of the melting point, density of the liquid, and specific heat of the liquid. Third, presently, Tersoff-ARK's shortcomings are the prediction of lower values of heat of fusion and liquid coordination number. Both MEAM and SW models give better agreement of the heat of fusion and liquid coordination number with the experiment compared to Tersoff-ARK, but at the expense of the density of liquid (and the melting point in the case of MEAM). Also, MEAM and SW models predict specific heat of the liquid nearly 10% and 20% higher than the experiment, respectively. With respect to the solid, the computed results are expected to be nearly the same for the Tersoff-4 and Tersoff-ARK potentials when the temperature is not high and the structure remains tetrahedral. Future studies will focus on further modification of the parameters in the Tersoff-ARK potential, such that these two properties as well as other properties, such as elastic constants and phonon frequencies, agree with the values reported in the literature. This may involve modification of the numerical values of some of the parameters in the Tersoff's potential and/or functional modification of the parameters such that a majority of experimental values agree with predictions.

TABLE V. A comparison of RDF results given by different potentials with the experiment.

	V_V[a]	V_{IV}[b]	V_{III}[c]	Experiment
Location of the first peak (Å)	2.48	2.44	2.45	2.5,[d] 2.41–2.46[e]
Location of the next peak (Å)	3.88	3.95	3.90	3.8,[d] 3.28–3.45[e]
FWHM of first peak (Å)	0.48	0.48	0.52	0.6[d]
Height of the first peak	2.55	2.50	2.4	2.5[d]

[a]Present modification.
[b]Tersoff-4 potential as given in Ref. 4.
[c]Reference 3.
[d]Reference 56.
[e]Reference 57.

IV. SUMMARY AND CONCLUSIONS

The void-nucleated melting method has been employed to compute the melting point of Si using the Tersoff-4 potential. The good agreement of the computed melting point to that determined by Yoo *et al.*[24] by the method of equilibration of coexisting solid and liquid phases using the same potential supports the appropriateness of the void-nucleated method, which is at present empirical.

The computed value of the ratio of the melting point and mechanical melting point f, given by Eq. (1), is found to be 0.82. This value of f is comparable to those computed for van der Waals and molecular crystals of Ar, Ne, CH_3NO_2, RDX, TNAZ, ammonium nitrate, and ammonium dinitramide as well as those given by the theoretical model of Lu and Li[33] for a large number of metallic crystals of Al, Au, Co, Cu, Fe, Mn, Ni, Pb, Pd, Pt, Sb, and Sn; for all these crystals, f is found to be in the range 0.77 to 0.85.

The validity of the void-nucleated melting method and the fact that $f \sim 0.8$ for a large number of systems suggests the need for a more rigorous investigation on whether the existence of the plateau region in the void size versus melting temperature is universal, whether the melting point in the plateau region corresponds to the thermodynamic melting point, and whether the value of $f \sim 0.8$ is also a universal result.

In agreement with the previous computed results, the present studies show that the Tersoff-4 potential overestimates the melting point by more than 50% and gives the density of the liquid ~14% lower than the experiment. It is also noted that an addition of a long-range Lennard-Jones (12-6) interaction to the Terosoff-4 potential does not significantly change the melting point.

The present studies also show that a simple modification in 3 of the 12 parameters of the Tersoff-4 potential can bring

the melting point as well as the density of the liquid into good agreement with experiment without altering the density and energy values of the solid crystal at room temperature.

A comparison of density over a wide range of temperatures shows that the density of the solid predicted by the Tersoff-ARK potential (Tersoff potential with the modified parameters) is larger than that given by the Tersoff-4 potential. This difference is not appreciable at room temperature, but increases with temperature and is ~1% at the melting point. The computed cohesive energy of the solid, the point defect energy corresponding to a vacancy, and the surface energy values for (100) and (111) surfaces are also found to be almost equal for the modified as well as unmodified parameters. The point defect energy values corresponding to the hexagonal and tetrahedral interstitials, however, are found to be ~40% lower when the modified parameters are used. The hexagonal interstitial energy given by the modified parameters is found to be in excellent agreement with the corresponding LDA value determined by Goedecker *et al.*[54] The coordination number of the liquid and specific heat are also found to be in better agreement with experiment when the modified set of parameters is used. It may be of interest to investigate further to see if the Tersoff-ARK potential is a better choice for other properties, such as elastic constants and phonon frequencies.

ACKNOWLEDGMENTS

This project is funded by Grant Nos. DMI-0200327 and DMI-0457663 from the National Science Foundation. We thank Dr. W. DeVries, Dr. G. Hazelrigg, Dr. J. Cao, and Dr. D. Durham of the Division of Design, Manufacturing, and Industrial Innovation; Dr. B. M. Kramer, Engineering Centers Division; and Dr. J. Larsen Basse, Tribology and Surface Engineering program for their interest and support of this work. This project was also funded by a DEPSCoR grant on the Multiscale Modeling and Simulation of Material Processing (Grant No. F49620-03-1-0281). The authors thank Dr. Craig S. Hartley of the AFOSR for his interest in and support of this work. One of the authors (R.K.) also thanks A. H. Nelson, Jr., Endowed Chair in Engineering, for additional support for this project.

*Corresponding author. Fax: (405) 744-7873; Email address: ranga@ceat.okstate.edu

[1] J. Tersoff, Phys. Rev. Lett. **56**, 632 (1986).
[2] J. Tersoff, Phys. Rev. B **37**, 6991 (1988).
[3] J. Tersoff, Phys. Rev. B **38**, 9902 (1988).
[4] J. Tersoff, Phys. Rev. B **39**, 5566 (1989).
[5] F. H. Stillinger and T. A. Weber, Phys. Rev. B **31**, 5262 (1985); Phys. Rev. B **33**, 1451(E) (1986).
[6] G. Ackland, Phys. Rev. B **40**, 10351 (1989).
[7] C. F. Abrams and D. B. Graves, J. Appl. Phys. **86**, 5938 (1999).
[8] M. I. Baskes, J. S. Nelson, and A. F. Wright, Phys. Rev. B **40**, 6085 (1989).
[9] D. W. Brenner and B. J. Garrison, Phys. Rev. B **34**, 1304 (1986); P. M. Agrawal, L. M. Raff, and D. L. Thompson, Surf. Sci. **188**, 402 (1987).
[10] R. Biswas and D. R. Hamann, Phys. Rev. Lett. **55**, 2001 (1985); Phys. Rev. B **36**, 6434 (1987).
[11] B. C. Bolding and H. C. Andersen, Phys. Rev. B **41**, 10568 (1990).
[12] A. E. Carlsson, P. A. Fedders, and C. W. Myles, Phys. Rev. B **41**, 1247 (1990).
[13] J. R. Chelikowsky, J. C. Phillips, M. Kamal, and M. Strauss, Phys. Rev. Lett. **62**, 292 (1989).
[14] B. W. Dodson, Phys. Rev. B **35**, 2795 (1987).
[15] D. Humbird and D. B. Graves, J. Chem. Phys. **120**, 2405 (2004).
[16] J. F. Justo, M. Z. Bazant, E. Kaxiras, V. V. Bulatov, and S. Yip, Phys. Rev. B **58**, 2539 (1998).
[17] E. Kaxiras and K. C. Pandey, Phys. Rev. B **38**, 12736 (1988).
[18] P. Keblinski, M. Z. Bazant, R. K. Dash, and M. M. Treacy, Phys. Rev. B **66**, 064104 (2002).
[19] K. E. Kho and S. Das Sarma, Phys. Rev. B **38**, 3318 (1988).
[20] P. N. Keating, Phys. Rev. **145**, 637 (1966); W. Weber, Phys. Rev. B **15**, 4789 (1977).
[21] A. D. Mistriotis, N. Flytzanis, and S. C. Farantos, Phys. Rev. B **39**, 1212 (1989).
[22] E. Pearson, T. Takai, T. Halicioglu, and W. A. Tiller, J. Cryst. Growth **70**, 33 (1984).
[23] S. J. Cook and P. Clancy, Phys. Rev. B **47**, 7686 (1993).
[24] S. Yoo, X. C. Zeng, and J. R. Morris, J. Chem. Phys. **120**, 1654 (2004).
[25] J. Solca, A. J. Dyson, G. Steinebrunner, and B. Kirchner, Chem. Phys. **224**, 253 (1997).
[26] P. M. Agrawal, B. M. Rice, and D. L. Thompson, J. Chem. Phys. **118**, 9680 (2003).
[27] Z. H. Jin, P. Gumbsch, K. Lu, and E. Ma, Phys. Rev. Lett. **87**, 055703 (2001).
[28] J. Solca, A. J. Dyson, G. Steinebrunner, B. Kirchner, and H. Huber, J. Chem. Phys. **108**, 4107 (1998).
[29] F. Lutsko, D. Wolf, S. R. Phillpot, and S. Yip, Phys. Rev. B **40**, 2841 (1989).
[30] P. M. Agrawal, B. M. Rice, and D. L. Thompson, J. Chem. Phys. **119**, 9617 (2003).
[31] G. F. Velardez, S. Alavi, and D. L. Thompson, J. Chem. Phys. **120**, 9151 (2004).
[32] G. F. Velardez, S. Alavi, and D. L. Thompson, J. Chem. Phys. **119**, 6698 (2003).
[33] K. Lu and Y. Li, Phys. Rev. Lett. **80**, 4474 (1998).
[34] S. R. Phillpot, J. F. Lutsko, D. Wolf, and S. Yip, Phys. Rev. B **40**, 2831 (1989).
[35] C-S. Zha, R. Boehler, D. A. Young, and M. Ross, J. Chem. Phys. **85**, 1034 (1986).
[36] J. R. Morris and X. J. Song, Chem. Phys. **116**, 9352 (2002); J. R. Morris, C. Z. Wang, K. M. Ho, and C. T. Chan, Phys. Rev. B **49**, 3109 (1994).
[37] A. Gavezzotti, J. Mol. Struct. **485-486**, 485 (1999).
[38] P. M. Agrawal, B. M. Rice, and D. L. Thompson (unpublished).
[39] H. Balamane, T. Halicioglu, and W. A. Tiller, Phys. Rev. B **46**, 2250 (1992).

[40] M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids* (Oxford Science Publications, Oxford, UK, 1993).

[41] S. Melchionna, G. Ciccotti, and B. L. Holian, Mol. Phys. **78**, 533 (1993).

[42] F. A. Lindemann, Phys. Z. **11**, 609 (1910); J. J. Gilvarry, Phys. Rev. **102**, 308 (1956).

[43] M. Born, J. Chem. Phys. **7**, 591 (1939); Proc. Cambridge Philos. Soc. **36**, 160 (1940).

[44] L. Burakovsky, D. L. Preston, and R. R. Silbar, J. Appl. Phys. **88**, 6294 (2000).

[45] L. Burakovsky, D. L. Preston, and R. R. Silbar, Phys. Rev. B **61**, 15011 (2000).

[46] L. Burakovsky and D. L. Preston, Solid State Commun. **115**, 341 (2000).

[47] L. Gómez, A. Dobry, and H. T. Diep, Phys. Rev. B **63**, 224103 (2000).

[48] A. K. Rappe, C. J. Casewit, K. S. Colwell, W. A. Goddard III, and W. M. Skiff, J. Am. Chem. Soc. **114**, 10024 (1992).

[49] D. R. Lide, editor-in-chief, *CRC Handbook of Chemistry and Physics* (CRC Press, Boca Raton, FL, 2004–2005).

[50] Z. Zhou, S. Mukherjee, and W. Rhim, J. Cryst. Growth **257**, 350 (2003).

[51] I. Barin and O. Knacke, *Thermodynamic Properties of Inorganic Substances* (Springer-Verlag, Berlin, 1973).

[52] M. W. Chase, Jr., C. A. Davies, J. R. Downey, Jr., D. J. Frurip, R. A. McDonald, and A. N. Syverud, J. Phys. Chem. Ref. Data Suppl. **14**, 1 (1985).

[53] J. Q. Broughton and X. P. Li, Phys. Rev. B **35**, 9120 (1987).

[54] S. Goedecker, T. Deutsch, and L. Billard, Phys. Rev. Lett. **88**, 235501 (2002).

[55] W. K. Leung, R. J. Needs, G. Rajagopal, S. Itoh, and S. Ihara, Phys. Rev. Lett. **83**, 2351 (1999).

[56] Y. Waseda and K. Suzuki, Z. Phys. B **20**, 339 (1975).

[57] S. Ansell, S. Krishnan, J. J. Felten, and D. L. Price, J. Phys.: Condens. Matter **10**, L73 (1998).

[58] P. Keblinski, M. Z. Bazant, R. K. Dash, and M. M. Treacy, Phys. Rev. B **66**, 064104 (2002).

[59] M. Kaczmarski, R. Rurali, and E. Hernandez, Phys. Rev. B **69**, 214105 (2004).

[60] C. R. Miranda and A. Antonelli, J. Chem. Phys. **120**, 11672 (2004).

THE JOURNAL OF CHEMICAL PHYSICS **124**, 134306 (2006)

Molecular dynamics investigations of the dissociation of SiO_2 on an *ab initio* potential energy surface obtained using neural network methods

Paras M. Agrawal
Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, Oklahoma 74078

Lionel M. Raff[a)]
Chemistry Department, Oklahoma State University, Stillwater, Oklahoma 74078

Martin T. Hagan
Electrical and Computer Engineering, Oklahoma State University, Stillwater, Oklahoma 74078

Ranga Komanduri
Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, Oklahoma 74078

(Received 2 November 2005; accepted 16 February 2006; published online 7 April 2006)

The neural network (NN) procedure to interpolate *ab initio* data for the purpose of molecular dynamics (MD) simulations has been tested on the SiO_2 system. Unlike other similar NN studies, here, we studied the dissociation of SiO_2 without the initial use of any empirical potential. During the dissociation of SiO_2 into Si+O or $Si+O_2$, the spin multiplicity of the system changes from singlet to triplet in the first reaction and from singlet to pentet in the second. This paper employs four potential surfaces. The first is a NN fit [NN(STP)] to a database comprising the lowest of the singlet, triplet, and pentet energies obtained from density functional calculations in 6673 nuclear configurations. The other three potential surfaces are obtained from NN fits to the singlet, triplet, and pentet-state energies. The dissociation dynamics on the singlet-state and NN(STP) surfaces are reported. The results obtained using the singlet surface correspond to those expected if the reaction were to occur adiabatically. The dynamics on the NN(STP) surface represent those expected if the reaction follows a minimum-energy pathway. This study on a small system demonstrates the application of NNs for MD studies using *ab initio* data when the spin multiplicity of the system changes during the dissociation process. © *2006 American Institute of Physics*.
[DOI: 10.1063/1.2185638]

I. INTRODUCTION

The interpolation of *ab initio* potential energy surface (PES) data using neural networks[1–5] (NNs) and other methods[6–24] for the purpose of executing molecular dynamics simulations has been a subject of considerable interest. Recently, Raff *et al.*[1] have investigated in detail a method involving the use of NNs to affect the interpolation of *ab initio* PES data. The results of such studies[1] performed for Si_5 and vinyl bromide systems show that the NN method is very promising in terms of accuracy, convenience, and the requirement of CPU time. Our recent study[2] on the vinyl bromide dissociation dynamics demonstrates the success of the NN method for molecular dynamics (MD) simulations of a six-atom system having six open reaction channels.

The vinyl bromide system so investigated is somewhat simple, at least in the sense that the electronic state of the system for various dissociation channels studied does not change during the dissociation process. That is, the reactions are all expected to occur adiabatically on the ground-state singlet potential surface. However, it is well known that many chemical reactions for various systems proceed in such a manner that the system changes its spin state during and after the reaction. For example, for the dissociation,

$$SiO_2 \rightarrow SiO + O, \tag{1}$$

the system is initially in the singlet state but it is in the triplet state after the dissociation. Similarly, the dissociation given by the following reaction:

$$SiO_2 \rightarrow Si + O_2, \tag{2}$$

corresponds to an example in which the system is initially in the singlet state, whereas the products are in the pentet state. Therefore, in pursuit of studying the applicability of the NN procedure[1–5] to other systems, it may be interesting to study the reactions given by Eqs. (1) and (2).

One special feature of our previous study on vinyl bromide is the application of a method based on the use of an empirical potential energy surface to sample the configurations of the molecule in a 12-dimensional hyperspace. There, we generate configurations of vinyl bromide by running a large number of dissociating trajectories using an empirical potential. Out of the configurations so generated, a few thousands are first selected. These configurations are then used to compute an *ab initio* potential database which is employed to generate a NN potential surface on which additional trajec-

[a)]Electronic mail: lionelraff@hotmail.co

 © 2006 American Institute of Physics

Downloaded 19 Jul 2006 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

tories are computed to generate more configurations whose energies are then calculated to increase the size of the *ab initio* database. By repeated application of such additions using a novelty sampling (NS) technique,[1,2] we obtain a sufficient number of data points to reach a converged PES. This (MD/NS/NN) method thus selects the important regions of phase space that are involved in the reactions under consideration. These techniques reduce the requirement of computing *ab initio* data at millions of configurations to a few thousands. However, the method has a limitation that it is not self-starting. One needs to start with an empirical potential. Therefore, it is of interest to see if a system such as SiO_2 with only three internal degrees of freedom can be investigated without the initial use of an empirical potential.

For the SiO_2 system, where only three-dimensional hyperspace corresponding to three coordinates R_1, R_2, and R_3 is needed, one can attempt to investigate the reactions by sampling configuration space using chemical intuition and a narrow grid of points within an appropriate range of R_1, R_2, and R_3. R_1, R_2, and R_3 refer to the $Si-O^{(1)}$, $Si-O^{(2)}$, and $O^{(1)}-O^{(2)}$ distances, respectively. Here, superscripts (1) and (2) have been used to distinguish the two oxygen atoms.

In this paper, we shall present the results of such studies of the SiO_2 system. In the next section we have described the computational procedure. The results are then presented and discussed in Sec. III. Section IV presents a discussion of possible adiabatic versus nonadiabatic processes for this system. The results are summarized in the final section.

II. COMPUTATIONAL PROCEDURE

A. *Ab initio* data

Ab initio calculations have been performed using GAUSSIAN 03 (Ref. 25) at B3LYP/6-31G* level at 3753 configurations. For each configuration, three Gaussian calculations have been run by varying the spin multiplicity (singlet, triplet, and pentet). The 3753 configurations have been obtained from a nonuniform grid in R_1-R_2-θ space, where θ is the angle between R_1 and R_2 vectors. The grid spacing in the R_1 or R_2 dimension is varied from 0.01 to 1.0 Å. The minimum spacing value is used near the equilibrium value of the Si–O bond length and the maximum value is used for large values of R_1 or R_2. The bond distances are varied from 1 to 10 Å and θ is varied from 0° to 180°. In the θ dimension, the spacing is varied from 3° to 10°.

Configurations that are very different from those associated with reactions (1) and (2) have been ignored. For example, configurations corresponding to Si+O+O have not been considered. When R_1 and R_2 are large, configurations having $R_3 > 2.5$ Å have been ignored.

To obtain the final singlet, triplet, and pentet databases, we have discarded all those points which have potential higher than V_O, where V_O corresponds to the potential energy of the Si(triplet)+O(triplet)+O(triplet) system. For convenience, we have scaled the potential values taking V_O as zero. Finally, symmetry considerations were employed to increase the size of the database. That is, for a configuration with $R_1=a$, $R_2=b$, and $R_3=c$, if $V=V_{abc}$ we added a point having $V=V_{abc}$ for the configuration with $R_1=b$, $R_2=a$, and $R_3=c$. A fourth database has been obtained by comparing singlet, triplet, and pentet energies at each configuration and selecting the minimum of these three.

TABLE I. The maximum and minimum values of input and output parameters.

K	K_{min}	K_{max}
R_1 (Å)	1.2	10.0
R_2 (Å)	1.2	10.0
R_3 (Å)	0.942 26	10.0
V (eV) (STP surface)	−11.840 5	−0.0169
V (eV) (S surface)	−11.840 5	−0.0169

B. Neural networks

As mentioned earlier, we have N data having R_1, R_2, and R_3 as inputs and the *ab initio* potential, V(*ab initio*), as output. The N data points in each database are used to train and test a two-layer (3-40-1) neural network.[26] In the following discussion, the network fits to the singlet, triplet, pentet, and combined, minimum-energy databases are denoted as NN(S), NN(T), NN(P), and NN(STP), respectively.

For training the neural networks, the database of 6673 energies was partitioned into training and testing sets containing 5500 and 1173 data points, respectively. In each case, the fitting to the training set was accomplished iteratively using 3000 cycles with the Levenberg-Marquardt algorithm.[27] To obtain the best fit, we chose the number of neurons (n) in the hidden layer to be 40.

From such training, we determined $\mathbf{W}_1$, $\mathbf{b}_1$, $\mathbf{W}_2$, and $\mathbf{b}_2$ matrices of dimensions $(n\times 3)$, $(n\times 1)$, $(1\times n)$, and (1×1), respectively. These matrices can be used[26] to compute the NN potential $V=V(\boldsymbol{\alpha};\mathrm{NN})$ for any desired value of input matrix $\boldsymbol{\alpha}=\boldsymbol{\alpha}(R_1,R_2,R_3)$ using the following equations:

$$V(\boldsymbol{\alpha};\mathrm{NN}) = \mathbf{W_2 a_1} + \mathbf{b_2}, \tag{3}$$

where

$$\mathbf{a_1} = 1/\{1+\exp[-\mathbf{n_1}]\}, \tag{4}$$

with

$$\mathbf{n_1} = \mathbf{W_1}\boldsymbol{\alpha} + \mathbf{b_1}. \tag{5}$$

Here, $\mathbf{n}_1$, $\mathbf{a}_1$, and $\boldsymbol{\alpha}$ represent the matrices of dimensions $(n\times 1)$, $(n\times 1)$ and (3×1), respectively. The elements of matrix $\boldsymbol{\alpha}$ are the input parameters R_1, R_2, and R_3.

To obtain better interpolation accuracy, we have linearly scaled the input and output parameters in the range of −1 to 1 using the following relation:

$$K_s = 2(K-K_{min})/(K_{max}-K_{min}) - 1, \tag{6}$$

where K_s is the scaled value of K, and K_{min} and K_{max} are the minimum and maximum values of the parameter. After computing $V(\boldsymbol{\alpha};\mathrm{NN})$ using Eq. (3), we perform the reverse scaling procedure to obtain the actual $V(\boldsymbol{\alpha};\mathrm{NN})$. The values of K_{min} and K_{max} for all parameters are given in Table I.

Downloaded 19 Jul 2006 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

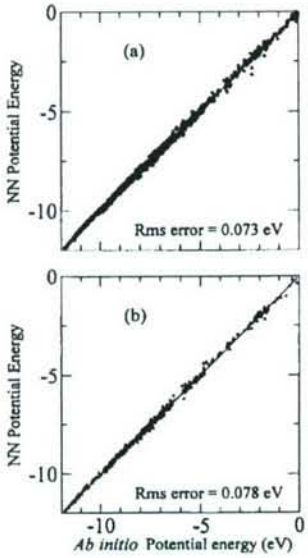


FIG. 1. The potential energy in eV given by the NN(STP) neural network vs the corresponding *ab initio* potential energy for the data used for (a) training and (b) testing.

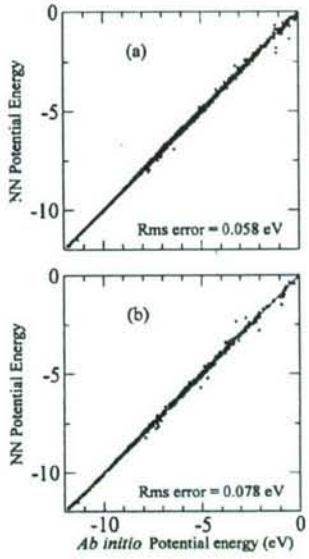


FIG. 2. The potential energy in eV given by the NN(S) neural network vs the corresponding *ab initio* potential energy for the data used for (a) training and (b) testing.

The required weight and bias matrices for the NN((STP) and NN(S) potential surfaces have been placed on deposit with the Electronic Physics Auxiliary Publication Service (EPAPS).[28]

To obtain the forces for the MD simulations, derivatives of the potential have been determined numerically. It may be noted that, unless otherwise specified, we determine potential energy and/or its derivatives for a given configuration using the procedure described in this subsection.

The quality of fit of the NN(STP) surface to the training set is shown in Fig. 1(a). It shows the potential, $V(\boldsymbol{\alpha};\mathrm{NN(STP)})$, given by Eq. (3), versus the $V(ab\ initio)$ potential for all 5500 configurations used to train the networks. A similar plot for the remaining 1173 data in the testing set is shown in Fig. 1(b). Ideally, the points in such plots would lie on a 45° line. The standard deviation of the points from the 45° line is a measure of the accuracy of the network. We find that a rms error in the training is 0.073 eV and that for the testing set is 0.078 eV, which is within the range generally regarded as "chemical accuracy" for the purposes of electronic structure calculations.

Figures 2(a) and 2(b) show analogous plots for the NN(S) adiabatic potential surface. In this case, the rms interpolation errors for the training and testing sets are 0.058 and 0.078 eV, respectively.

C. Molecular dynamics simulations

The Runge-Kutta method is used to integrate Hamilton's equations of motion.[29,30] An integration time step of 0.05 t.u. is used (1 t.u.$=1.018\times10^{-14}$ s). With this time step size, the energy is conserved to an accuracy of better than 10^{-3} eV. To study dissociation of SiO_2 at an internal energy E, a set of 500 trajectories is run. The initial configuration is obtained by randomly distributing this energy E into three normal modes of vibration. For the initial position coordinates, we have taken the equilibrium configuration of SiO_2, and the momenta are assigned to three atoms according to the method based on normal mode vectors, as described in Ref. 31. The normal mode vectors for this purpose have been computed using GAUSSIAN 03. The use of this method ensures that the initial states for the study of the decomposition reactions have a microcanonical distribution of the internal energy and contain no linear or angular momentum. A trajectory is allowed to run for 40 000 time steps; if dissociation occurs earlier, the trajectory is terminated.

III. RESULTS AND DISCUSSION

A. Energetics and geometry

Unless otherwise specified, the results reported in this paper refer to the NN(STP) surface.

It is well known that the geometry of the SiO_2 molecule in the gas phase, such as CO_2, is linear.[32–34] Here, we noted the same and found that the equilibrium Si–O bond length is 1.52 Å. Chu *et al.*[32] performed *ab initio* calculations on this system at the density functional theory (DFT) level using a 3-21G basis set for silicon and 6-31G* basis set for the oxygen atoms. They found this value to be 1.54 Å. Using a molecular orbital approach and the generalized gradient approximation in the density functional theory, Nayak *et al.*[33] determined that the equilibrium value of Si–O bond length in SiO_2 is 1.53 Å. Further, in the present studies, the Si–O bond length in SiO is found to be equal to 1.52 Å.

To obtain the barrier height or dissociation energy of the reaction given by Eq. (1), we checked and noted that the minimum-energy path follows the linear geometry, i.e., for a given value of R_1 and R_2, the energy is minimum when $R_3=R_1+R_2$. Therefore, in the linear geometry for a fixed value of R_2, we have varied R_1 to obtain the minimum energy. The variation of this minimum energy with R_2 on the NN(STP)

Downloaded 19 Jul 2006 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

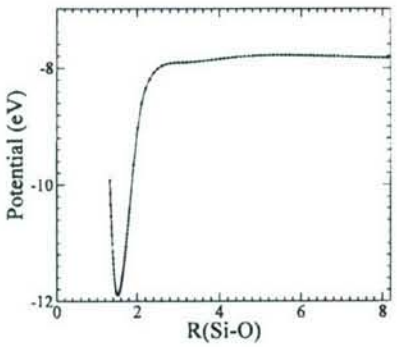


FIG. 3. The potential energy of SiO_2 as a function of R_2 on the NN(STP) surface corresponding to the minimum reaction path, as described in the text for the reaction given by Eq. (7).

potential surface is shown in Fig. 3. From this curve, we find that the barrier height for this reaction is 4.1 eV.

The reaction given by Eq. (2) has been studied in two steps. In step I, for a fixed value of θ, we varied R_1, keeping R_2 equal to R_1, to obtain the minimum-energy configuration. The value of minimum energy so obtained as a function of θ is shown in Fig. 4. Here, we note an existence of a shallow minimum at $\theta=55°$. R_1, R_2, and R_3 values at this minimum are 1.67, 1.67, and 1.54 Å, respectively. Thus, we infer that though SiO_2 is most stable in the linear geometry, it also has a triangular metastable state which is 2.2 eV higher than the most stable linear state. The barrier height for the transition out of this metastable state is about 0.6 eV.

In step II, we monitor the minimum-energy path on the NN(STP) surface for the transition from the triangular SiO_2 state to the products $Si+O_2$. To do so, in the nonlinear geometry for a given value of $R_1(=R_2)$, we vary R_3 to get a minimum in the energy versus R_1 curve. The minimum energy so obtained as a function of R_1 is shown in Fig. 5. From this curve, we note that the minima at $R_1=R_2=1.67$ Å correspond to the metastable state obtained in step I. The energy difference between this minimum and the state with large $R_1(=R_2)$ given by Fig. 5 is equal to 4.2 eV. It has been found that the value of R_3 corresponding to such a large R_1 is 1.22 Å. As expected, this value of R_3 is in agreement with the experimental value (=1.207 Å) (Ref. 35) for the equilibrium bond length of O_2. Some of the above results can be expressed succinctly as follows:

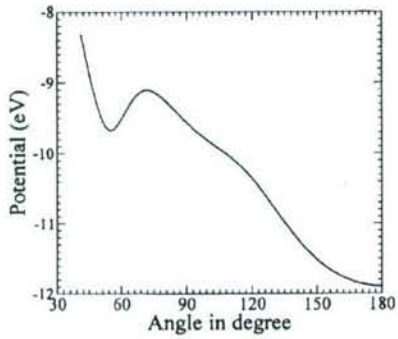


FIG. 4. The potential energy of SiO_2 as a function of θ corresponding to the minimum reaction path on the NN(STP) surface, as described in the text for the reaction given by Eq. (8).

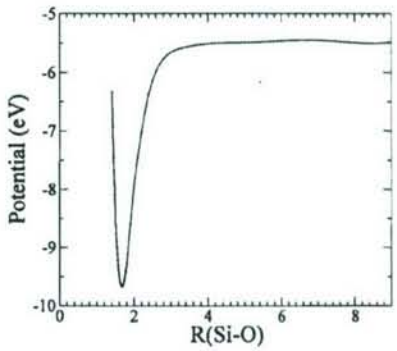


FIG. 5. The potential energy of SiO_2 as a function of R_1 corresponding to the minimum reaction path on the NN(STP) surface, as described in the text for the reaction given by Eq. (9).

$$SiO_2 + E_1 = SiO + O, \tag{7}$$

$$SiO_2(\text{linear}) + E_2 = SiO_2\ (\text{triangular}), \tag{8}$$

$$SiO_2(\text{triangular}) + E_3 = Si + O_2, \tag{9}$$

where $E_1=4.1$ eV, $E_2=2.2$ eV, and $E_3=4.2$ eV. Combining Eqs. (8) and (9), we obtain

$$SiO_2(\text{linear}) + E_4 = Si + O_2, \tag{10}$$

where $E_4=6.4$ eV. The above barrier heights and energies have been obtained without any consideration for the small corrections due to zero-point energy.

It may be noted that the value of $E_1=4.1$ eV given by the NN(STP) and *ab initio* data in our present studies is very close to 4.013 eV given directly by our *ab initio* data without the use of NN fitting. Lu *et al.*[34] also performed similar B3LYP/6-31G(*d*) *ab initio* calculations using the GAMESS (Ref. 36) code and found the value of $E_1=3.953$ eV. Similarly, against $E_4=6.4$ eV given by the NN(STP) surface in the present studies, we obtained $E_4=6.432$ eV directly from the *ab initio* data and Lu *et al.*[34] got $E_4=6.310$ eV using the GAMESS software. These results show that the error associated with the NN fitting is comparable to the inaccuracy associated with the *ab initio* data. The differences in the *ab initio* results may be due to small differences in the numerical procedures in the GAMESS and GAUSSIAN codes.

As expected, the value of $E_2=2.2$ eV given by the present studies is also in good agreement with 2.108 eV computed by Lu *et al.*[34] using the GAMESS code at B3LYP/6-31G(*d*) level. The Si–O equilibrium bond length of 1.67 Å noted in the present studies for SiO_2 in the triangular geometry is comparable with the Si–O bond lengths computed by Chu *et al.*[32] for various large clusters of oxides of silicon: Chu *et al.*[32] noted that the large clusters of oxides of silicon are composed of two buckled rings where the Si–O bond length ranges from 1.67 to 1.96 Å; they also noted that the value of the Si–O bond length in various Si_2O_4 rings in the Si_nO_m clusters with $m \geq n+2$ is 1.66 Å.

In regard to comparison with the experiment, the values of E_4 given by the present studies as well as by Lu *et al.*[34] differ from the experimental value[37] of 6.769 eV by ~5%. For the binding energy of SiO, we note a difference of ~5%

Downloaded 19 Jul 2006 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

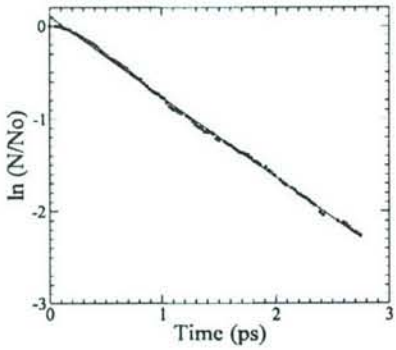


FIG. 6. The variation of $\ln(N/N_o)$ as a function of time (t) in picoseconds at E=4.5 eV computed on the NN(STP) surface.

from the experiment. We obtain a value of 7.828 eV for this quantity in good agreement with 7.738 eV obtained by Lu *et al.*[34] using the GAMESS code at B3LYP/6-31G(d) level. The reported experimental value[37] is 8.237 eV.

It may be interesting to note the spin states of the systems studied here. We see that in Eq. (7), SiO_2 is in a singlet state but the products of this reaction are in the triplet state. In Eq. (8), SiO_2 on both sides is in a singlet state, but in Eq. (9) or (10), the $Si+O_2$ system is in a pentet state. If we confine our studies to an adiabatic reaction on the NN(S) surface, the dissociation energy for the reaction given by Eq. (1) comes out to be 6.8 eV instead of 4.1 eV.

B. Trajectories

Trajectories have been computed on both the NN(STP) and NN(S) surfaces using the procedure described in Sec. II C to study the dissociation reaction to SiO+O products. A typical decay plot for dissociation on the NN(STP) surface at vibrational energy E=4.5 eV is shown in Fig. 6. The near linearity of the result shows that the dissociation process follows first-order kinetics, i.e.,

$$N = N_o \exp(-kt), \tag{11}$$

where N is the number of undissociated trajectories at time t and N_o is the value of N at t=0. As mentioned earlier, we have taken N_o=500. It may be mentioned that a typical trajectory running for 1500 time steps takes about 1 s of CPU time on our 1.4 GHz machine.

From the slope of the decay plot, we obtain the dissociation rate coefficient, k. The rate coefficients for a few values of E in the range of 4.3–5.0 eV have been computed with the NN(STP) surface. These coefficients lie in the range from 0.304 to 4.274 ps^{-1}. If the reaction is assumed to occur adiabatically on the singlet NN(S) surface, the corresponding rate coefficients are much smaller due to the reaction endothermicity being 2.7 eV larger for the process. All results are given in Table II.

The dependences of dissociation rate coefficients on energy E for both the NN(STP) and NN(S) surfaces are shown in Fig. 7. This plot shows that the dissociation rates follow the Rice-Ramsperger-Kassel (RRK) equation,[38]

TABLE II. Reaction rate coefficients (k) for the reaction SiO_2=SiO+O.

Energy (eV)	k (ps^{-1}) NN(S)	k (ps^{-1}) NN(STP)
4.3	···	0.304
4.4	···	0.548
4.5	···	0.867
4.6	···	1.123
4.8	···	2.266
5.0	···	4.274
7.0	0.0314	···
7.2	0.225	···
7.4	0.467	···
7.5	0.618	···
8.0	1.456	···
8.5	2.397	···

$$k = f(1 - E/E_o)^{s-1}, \tag{12}$$

where E_o is the critical energy that can be taken equal to the barrier height of the dissociation reaction, s is a parameter that gives the number of vibrational degrees of freedom participating in the reaction, and f is the frequency factor or high-energy limit of the rate coefficient. For this plot, we have taken E_o=4.1 and 6.8 eV for the NN(STP) and NN(S) surfaces, respectively. A least-squares fit of the data shown in Fig. 7 yields corresponding s values of 2.90 and 3.19 and f values of 92.4 ps^{-1} (3080 cm^{-1}) and 97.6 ps^{-1} (3253 cm^{-1}). These values of s are in good agreement with the number of vibrational degrees of freedom participating in the reaction, which is equal to three due to a rapid energy transfer among all the three modes. The computed frequency factors f are nearly twice the asymmetric stretching frequency (1449.8 cm^{-1}) of SiO_2 given by the Gaussian calculation at the B3LYP/6-31G* level. The factor of 2 is easy to understand from the fact that SiO_2 has two oxygen atoms and the reaction given by Eq. (1) can take place in two ways.

To investigate the dissociation of SiO_2 into $Si+O_2$, the vibrational energy must be in excess of 6.4 eV [see Eq. (10)]. Therefore, we inserted a total vibrational energy E as high as 8.0 eV randomly in all modes and computed more than 2000 trajectories but all trajectories are found to lead to dissociation to SiO+O. We also put the same total energy in a single

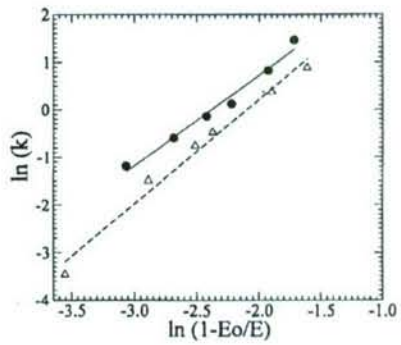


FIG. 7. RRK plot for the dissociation rate k in ps^{-1} for the dissociation of SiO_2 into SiO+O. The circles and solid line correspond to the data obtained using the NN(STP) potential with E_o=4.1 eV. The triangles and the dotted line correspond to the data obtained using the NN(S) potential surface with E_o=6.8 eV.

Downloaded 19 Jul 2006 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

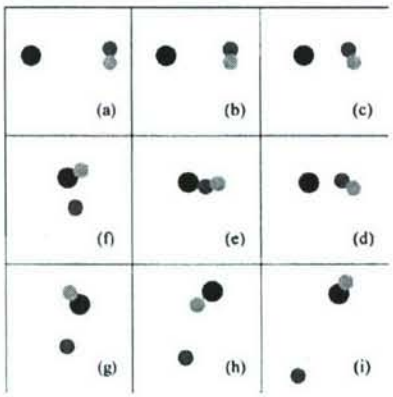


FIG. 8. The location of the atoms during the combination-dissociation reaction given by Eq. (14) for a typical trajectory at time steps: (a) 1, (b) 200, (c) 400, (d) 500, (e) 600, (f) 800, (g) 1000, (h) 1200, and (i) 1600; 1 time step=0.509 fs. Each of the nine boxes shown here is of size 12×12 Å in the plane in which the reaction occurs.

bending mode but we could not get $Si+O_2$. This may be due to the fact that the SiO+O channel has a very low barrier height relative to that for the $Si+O_2$ channel. Therefore, the dissociation into SiO+O occurs with very high probability, precluding the formation of $Si+O_2$ products.

To test the ability of the NN(STP) potential energy surface to properly represent the reaction channel leading to $Si+O_2$ products, we investigated the reaction given by Eq. (2) in the back direction, i.e.,

$$Si + O_2 \rightarrow SiO_2. \tag{13}$$

We have found that although SiO_2 is formed in the collision of Si with O_2, in all cases SiO_2 immediately dissociates into SiO+O. Figure 8 depicts a typical trajectory undergoing such a combination and dissociation process, i.e.,

$$Si + O_2 \rightarrow SiO_2 \rightarrow SiO + O. \tag{14}$$

In this trajectory, we started with the initial configuration as shown in Fig. 8(a) and imparted a small relative velocity to Si atom in the direction of the center of mass of O_2. We also initially inserted small vibrational and rotational energies into the O_2 molecule. Figure 9 shows the temporal behavior of the interatomic distances in the trajectory. Initially, R_1 and

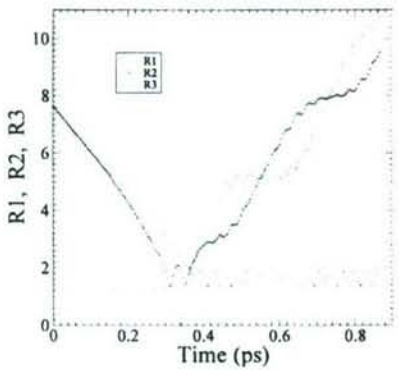


FIG. 9. The variation of R_1, R_2, and R_3 (in Å) as a function of time t for a typical trajectory undergoing a combination-dissociation reaction given by Eq. (14).

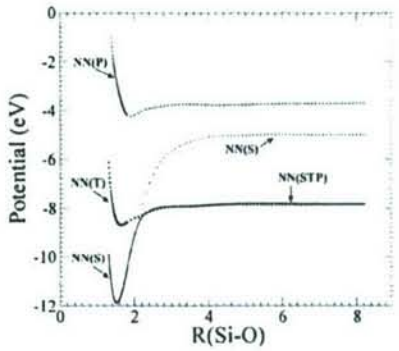


FIG. 10. Variation of the four potentials along the $SiO_2 \rightarrow SiO+O$ reaction coordinate. The results for the singlet, triplet, pentet, and minimum-energy reaction pathways are labeled in the figure as NN(S), NN(T), NN(P), and NN(STP), respectively. The crossing point of the NN(S) and NN(STP) potentials is seen at a Si–O distance of 2.1 Å. The energy at the crossing point relative to that for equilibrium SiO_2 is 3.5 eV.

R_2 are large while R_3 is small. At $t\sim0.35$ ps, R_1 and R_2 both become small and R_3 starts increasing, i.e., two Si–O bonds tend to be formed and the O–O bond begins to break. Furthermore, we note that very soon the system appears as SiO+O (demonstrated by a small value of R_1 and large values of R_2 and R_3). This trajectory thus reveals that the potential is capable of representing scission of the O–O bond and formation of Si–O bonds. However, the large exothermicity of the recombination process causes the newly formed SiO_2 molecule to undergo rapid dissociation to SiO+O.

IV. ADIABATIC/NONADIABATIC PROCESSES

The investigations of the SiO_2 dynamics reported here represent two limiting cases. If dissociation to SiO+O occurs adiabatically on the singlet surface represented by NN(S), it will be a high-energy process with a threshold slightly below 6.8 eV when zero-point energy effects are considered. If, however, the dissociation follows the lowest-energy pathway leading from SiO_2 to SiO+O, dissociation can occur at significantly lower internal energies with a threshold around 4.0 eV. In addition to these two limiting cases, there exist a very large number of other possible pathways that involve transitions between the singlet and triplet surfaces at points other than the crossing points. Such pathways might be investigated using a surface hopping procedure such as that suggested by Tully.[39]

A simple thermodynamic argument suggests that the minimum pathway mechanism represented by the NN(STP) surface would be favored. However, such a view ignores the fact that the system must be able to undergo a change in spin state in the time available when the singlet-triplet seam is reached. If the spin-orbit and nonadiabatic coupling terms are not sufficiently strong to induce such a multiplicity change, the dissociation process might pass through the seam without a change of spin state and continue on the adiabatic singlet surface. In the absence of a careful computation of these coupling terms in a surface hopping investigation,[39] we cannot definitively answer these questions related to the SiO_2 dissociation mechanism.

Downloaded 19 Jul 2006 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

One factor that favors a near minimum-energy pathway mechanism is the location of the singlet-triplet crossing point along the reaction coordinate for dissociation to SiO+O. This is shown in Fig. 10. This figure shows the computed *ab initio* energies for the singlet, triplet, and pentet states along the dissociation coordinate. The NN fits, NN(S), NN(T), and NN(P), are shown as points. The minimum-energy path on the NN(STP) surface is the solid curve. The crossing point between the NN(T) and NN(S) surfaces occurs at an energy of 3.5 eV above the SiO_2 minimum. Therefore, dissociating trajectories with a total internal energy of 4.3 eV will have only 0.8 eV of internal vibrational energy remaining when the seam is reached. This energy will be distributed throughout three vibrational degrees of freedom. Consequently, we would expect the energy remaining in the reaction coordinate to be about 0.3 eV. The relatively heavy silicon and oxygen atoms will not be moving very fast at such energies. Under these conditions, we might expect the system to remain in the region of the crossing point for a sufficient time to permit spin-orbit and other adiabatic coupling terms to induce a spin transition to the triplet state.

A more quantitative evaluation of the extent of surface hopping in this system will require a careful computation of the spin-orbit coupling and transition probabilities.

V. SUMMARY AND CONCLUSIONS

Ab initio potential energy values for SiO_2 have been computed in 3753 configurations using GAUSSIAN 03 at the B3LYP/6-31G* level of theory. At each configuration, the energy values are calculated for singlet, triplet, and pentet spin states. For further use, out of these three values, we picked that value of energy which is minimum to create a fourth database. By discarding highly repulsive configurations and by adding some configurations based on the symmetry of the molecule, we obtained *ab initio* data points to interpolate for each of the four databases.

Two-layer (3-40-1) neural networks are used to interpolate between the points in each database. The NN surfaces thus obtained are denoted NN(S), NN(T), NN(P), and NN(STP) for the singlet, triplet, pentet, and minimum-energy databases, respectively. Trajectories have been computed using the interpolated data provided by the NN(S) and NN(STP) surfaces. Thus, the method is a self-starting one which does not require the initial use of an empirical potential. The following results obtained in this study show that the NN interpolation process is very good.

(1) The rms interpolation error on the NN(STP) surface associated with the training set is 0.073 eV and that with the testing set is 0.078 eV. This error is within the range generally regarded as "chemical accuracy" for the purposes of electronic structure calculations. For the NN(S) surface, the corresponding rms errors are 0.058 and 0.078 eV.

(2) The difference in the binding energy or dissociation energy of SiO as well as SiO_2 between the direct *ab initio* results and those given by the NNs is less than 0.1 eV. This difference is comparable with that given by the differences in the values given by GAUSSIAN 03 and GAMESS.

(3) The dissociation rate coefficients for the reaction, $SiO_2 \rightarrow SiO+O$, follow the RRK equation [see Eq. (12)]. The values of s and f parameters of the RRK equation for this reaction are found to be close to the expected values based on the number of degrees of freedom participating in the dissociation process, the vibrational frequency of the asymmetric stretching mode, and the number of possible ways in which the molecule can dissociate. This is the case whether the reaction occurs adiabatically on the NN(S) surface or follows a minimum-energy path on the NN(STP) surface.

This study demonstrates that the spin multiplicity can be appropriately and easily considered while using a NN to interpolate the *ab initio* data to run MD simulations.

As regards geometry of SiO_2, we have found that the most stable state is linear with a Si–O bond length of 1.52 Å. This value is in good agreement with that computed by Chu *et al.*[32] as well as Nayak *et al.*[33] Further, for a Si–O bond length in SiO, we get the same value (1.52 Å). For the dissociation energy of SiO, and SiO_2 (into $Si+O_2$), we found that the present results are in very good agreement with those obtained by Lu *et al.*[34] using GAMESS. However, these energy values are found to be about 5% lower than the experimental data.[37] In agreement with the previous results of Lu *et al.*,[34] we have seen that SiO_2 exists in a triangular state which is 2.1 eV higher than the most stable linear configuration. We further note that the barrier height to exit from this triangular state to the linear state is 0.6 eV. The Si–O bond length in this triangular SiO_2 state is 1.67 Å, which is in good accord with the value of 1.66 Å noted by Chu *et al.*[32] for the Si–O bond length in various Si_2O_4 rings in the Si_nO_m clusters with $m \geq n+2$.

Dissociation processes on the NN(S) and NN(STP) surfaces represent two limiting cases. The former is a high-energy dissociation, whereas the latter describes dissociation along a minimum-energy pathway between SiO_2 and SiO +O that requires a spin-state transition at the crossing points between singlet and triplet surfaces. In addition to these two limiting cases, there exist a very large number of other possible pathways that involve transitions between the singlet and triplet surfaces at points other than the crossing points.

It is suggested that both thermodynamic considerations and the energies at the singlet-triplet crossing points favor reaction along the minimum-energy pathway. The crossing points occur far along the reaction coordinate near the product valley of the surface. This means the energy remaining in motion along the reaction coordinate will be small. This point and the fact that the dissociation coordinate involves the relative motion of heavy Si and O atoms suggest that the system will remain in the region of the crossing seam for a sufficient time to permit spin-orbit and adiabatic couplings to induce a spin transition.

Downloaded 19 Jul 2006 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp

ACKNOWLEDGMENTS

This project is funded by grants from the National Science Foundation (DMI-0200327 and DMI-0457663). We thank Dr. W. DeVries, Dr. G. Hazelrigg, Dr. J. Cao, and Dr. D. Durham of the Division of Design, Manufacturing, and Industrial Innovation, Dr. B. M. Kramer, Engineering Centers Division, and Dr. J. Larsen Basse, Tribology and Surface Engineering program for their interest and support of this work. This project was also sponsored in part by a DEPS-CoR grant on the Multiscale Modeling and Simulation of Material Processing (F49620-03-1-0281). The authors thank Dr. Craig S. Hartley and Dr. J. Tiley, Program Managers of the Metallic Materials Program, of the Air Force Office of Scientific Research (AFOSR) for their interest in and support of this work. One of the authors (R.K.) also thanks A. H. Nelson, Jr., endowed chair in engineering for additional support.

[1] L. M. Raff, M. Malshe, M. Hagan, D. I. Doughan, M. G. Rockley, and R. Komanduri, J. Chem. Phys. **122**, 084104 (2005).
[2] D. I. Doughan, L. M. Raff, M. G. Rockley, M. Hagan, P. M. Agrawal, and R. Komanduri, J. Chem. Phys. **124**, 054321 (2006).
[3] S. Lorenz, A. Groß, and M. Scheffler, Chem. Phys. Lett. **395**, 210 (2004).
[4] D. F. R. Brown, M. N. Gibbs, and D. C. Clary, J. Chem. Phys. **105**, 7597 (1996).
[5] T. B. Blank, S. D. Brown, A. W. Calhoun, and D. J. Doren, J. Chem. Phys. **103**, 4129 (1995).
[6] J. Ischtwan and M. A. Collins, J. Chem. Phys. **100**, 8080 (1994).
[7] K. A. Nguyen, I. Rossi, and D. G. Truhlar, J. Chem. Phys. **103**, 5522 (1995).
[8] T. Ishida and G. C. Schatz, J. Chem. Phys. **107**, 3558 (1997).
[9] K. C. Thompson, M. J. T. Jordan, and M. A. Collins, J. Chem. Phys. **108**, 8302 (1998).
[10] L. A. Pederson, G. C. Schatz, T.-S. Ho, T. Hollebeek, H. Rabitz, L. B. Harding, and G. Lendvay, J. Chem. Phys. **110**, 9091 (1999).
[11] R. P. A. Bettens and M. A. Collins, J. Chem. Phys. **111**, 816 (1999).
[12] R. P. A. Bettens, M. J. T. Jordan, D. H. Zhang, and M. A. Collins, J. Chem. Phys. **112**, 10162 (2000).
[13] D. H. Zhang, M. A. Collins, and S.-Y. Lee, Science **290**, 961 (2000).
[14] L. A. Pederson, G. C. Schatz, T. Hollebeek, T.-S. Ho, H. Rabitz, and L. B. Harding, J. Phys. Chem. A **104**, 2301 (2000).
[15] K. Song and M. A. Collins, Chem. Phys. Lett. **335**, 481 (2001).
[16] G. E. Moyano and M. A. Collins, J. Chem. Phys. **119**, 5510 (2003).
[17] M. A. Collins and L. Radom, J. Chem. Phys. **118**, 6222 (2003).
[18] R. Z. Pascual, G. C. Schatz, G. Lendvay, and D. Troya, J. Phys. Chem. A **106**, 4125 (2002).
[19] M. A. Collins, Theor. Chem. Acc. **108**, 313 (2002).
[20] C. Crespos, M. A. Collins, E. Pijper, and G. J. Kroes, Chem. Phys. Lett. **376**, 566 (2003).
[21] C. Crespos, M. A. Collins, E. Pijper, and G. J. Kroes, J. Chem. Phys. **120**, 2392 (2004).
[22] G. G. Maisuradze, D. L. Thompson, A. F. Wagner, and M. Minkoff, J. Chem. Phys. **119**, 10002 (2003).
[23] A. Kawano, Y. Guo, D. L. Thompson, A. F. Wagner, and M. Minkoff, J. Chem. Phys. **120**, 6414 (2004).
[24] Y. Guo, A. Kawano, D. L. Thompson, A. F. Wagner, and M. Minkoff, J. Chem. Phys. **121**, 5091 (2004).
[25] M. J. Frisch, G. W. Trucks, H. B. Schlegel *et al.*, GAUSSIAN 03, Revision C.02, Gaussian, Inc., Wallingford, CT, 2004.
[26] M. T. Hagan, H. B. Demuth, and M. Beale, *Neural Network Design* (PWS, Boston, MA, 1996).
[27] M. T. Hagan and M. Menhaj, IEEE Trans. Neural Netw. **5**, 989 (1994).
[28] See EPAPS Document No E-JCPSA6-124-008613. This document can be reached via direct link in the online article's HTML reference section or via the EPAPS homepage (http://www.aip.org/pubservs/epaps.html).
[29] M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids* (Clarendon, Oxford, 2002).
[30] L. M. Raff and D. L. Thompson, in *Theory of Chemical Reaction Dynamics*, edited by M. Baer (CRC, Boca Raton, FL, 1985), Vol. III.
[31] E. B. Wilson, Jr., J. C. Decius, and P. C. Cross, *Molecular Vibrations* (Dover, New York, 1955).
[32] T. S. Chu, R. Q. Zhang, and H. F. Cheung, J. Phys. Chem. B **105**, 1705 (2001).
[33] S. K. Nayak, B. K. Rao, S. N. Khanna, and P. Jena, J. Chem. Phys. **109**, 1245 (1998).
[34] W. C. Lu, C. Z. Wang, V. Nguyen, M. W. Schmidt, M. S. Gordon, and K. M. Ho, J. Phys. Chem. A **107**, 6936 (2003).
[35] *CRC Handbook of Chemistry and Physics*, edited by D. R. Lide (CRC, Boca Raton, FL, 2004).
[36] M. W. Schmidt, K. K. Baldridge, J. A. Boatz *et al.*, J. Comput. Chem. **14**, 1347 (1993).
[37] *NIST-JANAF Thermochemical Tables*, 4th ed., edited by M. W. Chase, Jr., J. Phys. Chem. Ref. Data, Monograph No. 9, (National Institute of Standards and Technology, Gaithersburg, MD, 1998).
[38] K. J. Laidler, *Theories of Chemical Reaction Rates* (McGraw-Hill, New York, 1969), p. 119.
[39] J. C. Tully, J. Chem. Phys. **93**, 1061 (1990).

Downloaded 19 Jul 2006 to 139.78.79.49. Redistribution subject to AIP license or copyright, see http://jcp.aip.org/jcp/copyright.jsp