FEATURE ARTICLE

Portrait of a Crack: Rapid Fracture Mechanics Using Parallel Molecular Dynamics

FARID F. ABRAHAM
IBM Almaden Research Center

How do materials fracture? The molecular dynamics methods used to model this important problem parallelize well, allowing bigger and more realistic computational experiments. Simulations of how materials crack at the atomic level are yielding surprising results that sometimes contradict existing theory, but that may explain recent physical experiments.

The classical physics of the continuum has historically provided most of the theoretical and computational tools for the engineer and the technologist: in elasticity, fluid dynamics, thermodynamics, and other fields, this classical picture has worked well. More and more now, however, technology is encountering nanoscale behavior where the continuum picture is no longer valid. The nanophysics of materials presently belongs to the domain of condensed-matter physics, statistical mechanics, and quantum chemistry, and, in the broadest sense, to the nonlinear science of complex behavior. This dichotomy of theoretical approaches may best be illustrated in the field of fracture dynamics by examining recent advanced texts by Herrmann and Roux[1] and Freund.[2]

With the advent of high-performance computers, computational materials science is demonstrating the potential for providing very powerful tools and solutions of interest to current technology. The group of investigators I work with at the IBM Almaden Research Center in California, assisted by the resources and personnel of the Cornell Theory Center in New York state, has been developing parallel computing methods that allow us to reach new insights into the physics of materials at the atomic level. Our recent efforts have focused on the dynamics of rapid fracture. We are studying the fracture of two- and three-dimensional notched solids under tension, using simulations of up to 100 million and more atoms. In our parallel simulations of the rapid brittle fracture of rare-gas films and solid slabs, one of the most intriguing phenomena is the dynamic instability of the crack tip as it approaches a certain fraction of the Rayleigh sound speed.

Are cracks smooth?

Continuum fracture theory typically assumes that cracks are smooth. For dynamic cracks, it predicts that, as they propagate, they accelerate to a limiting velocity equal to the Rayleigh speed of the material.[2] In contrast, experiment tells us that, in a common fracture sequence in some polymers, glasses, and ceramics, an initially smooth and mirrorlike fracture surface begins to appear misty and then evolves into a rough, hackled region with a limiting velocity of about six-tenths the Rayleigh speed. In some brittle materials, the crack pattern can also exhibit a wiggle with a characteristic wavelength. Recent experiments have clearly shown that violent crack velocity oscillations occur beyond a speed of about one-third the Rayleigh speed and are correlated with the roughness of the crack surface.[3–5]

1070-9924/97/$10.00 © 1997 IEEE

Authorized licensed use limited to: IEEE Xplore. Downloaded on October 23, 2008 at 15:38 from IEEE Xplore. Restrictions apply.

It has been believed that these features are a consequence of imperfections in the materials. Others argue that the fracture dynamics may be universal, or structure-independent, and that a dynamical instability of the crack tip governs the crack velocity behavior and the morphology sequence of "mirror, mist, and hackle." All of these features are unexplained using continuum theory, though recent theoretical advances (for example, by Langer[6] and Marder and Liu[7]), are providing very important insights into this difficult problem. This suggests that a fundamental understanding may require a microscopic picture of the fracturing process.

Computational molecular dynamics, coded to run on scalable parallel computers, is becoming a very powerful tool for providing immediate insights into the nature of fracture dynamics. As with the experiments published elsewhere,[3–5] our initial interest was to study two-dimensional "mode one" loading.[8] We were able to follow the crack propagation over sufficient time and distance intervals so that a comparison with experiment became feasible. We compared laboratory and two-dimensional computer experiments, and identified microscopic processes. In particular, we arrived at an explanation for the limiting velocity of the crack being significantly less than the theoretical limit. When going to three dimensions, we again discovered the dynamical instability, but found that now the instability is immediate, followed by an unexpected brittle-to-ductile transition giving rise to a proliferation of loop dislocations and the arrest of the crack's propagation.

Methodology of molecular dynamics

The molecular dynamics simulation technique is based on the motion of a given number of atoms, governed by their mutual interatomic interactions described by continuous interatomic potentials. It requires the numerical integration of Hamilton's equations of motion.[9–11] For the sake of simplicity, we assume that the particles are identical, have three degrees of freedom each, and obey classical mechanics defined by a Hamiltonian $\mathcal{H}$. Denoting the atomic coordinates by $\mathbf{r} = \mathbf{r}_1, \ldots, \mathbf{r}_N$ and momenta by $\mathbf{p} = \mathbf{p}_1, \ldots, \mathbf{p}_N$, the Hamiltonian of the N-body system can be written as

$$\mathcal{H}(\mathbf{r},\mathbf{p}) = \mathcal{K}(\mathbf{p}) + \Phi(\mathbf{r})$$

where $\mathcal{K}$ is the kinetic energy, Φ is the potential energy, and

$$\mathcal{K}(r) = \frac{\sum_i \mathbf{p}_i \mathbf{p}_i}{2m}$$

(where m is the mass of one atom). The dynamical equations are

$$\frac{d\mathbf{r}_i}{dt} = \frac{\partial \mathcal{H}}{\partial \mathbf{p}_i} = \frac{\mathbf{p}_i}{m},$$

$$\frac{d\mathbf{p}_i}{dt} = \frac{\partial \mathcal{H}}{\partial \mathbf{r}_i} = \frac{-\partial \Phi(\mathbf{r})}{\partial \mathbf{r}_i}.$$

We use the fifth-order Nordsieck algorithm to solve these equations numerically.[12] This method gives high-order accuracy while requiring explicit evaluation only of the forces (not derivatives of the forces).

Equations of motion are needed, but a simulation study is defined by a model created to incorporate the important relevant features of the physical system of interest. These features are defined by the interatomic potentials, the initial conditions, and the boundary conditions. The interatomic forces are treated as central forces, modeled as a combination of a Lennard–Jones 12:6 interaction with a spline cutoff.[13] The total potential is written as

$$\Phi = \sum_{i=1}^{N} \frac{1}{2} \sum_{i \neq j} \phi(r_{ij}),$$

where ϕ is the potential of one atom pair and r_{ij} is the center-to-center separation distance. The LJ 12:6 potential has the form

$$\phi_{\mathrm{LJ}}(r) = 4\varepsilon((\sigma/r)^{12} - (\sigma/r)^{6}).$$

We express quantities in terms of reduced units. Lengths are scaled by the parameter σ, the value of the interatomic separation for which the LJ potential is zero, and energies are scaled by the parameter ε, the depth of the minimum of the LJ potential. Reduced temperature is therefore kT/ε.

Our choice of these simple interatomic force laws is dictated by our desire to investigate the qualitative features of a particular many-body problem common to a large class of real physical systems and not governed by the particular complexities of a unique molecular interaction. The van der Waals bonding, giving the cohesion of the rare-gas solids we investigated, can be modeled accurately by the two-body Lennard–Jones potential, which has served as a

Authorized licensed use limited to: IEEE Xplore. Downloaded on October 23, 2008 at 15:38 from IEEE Xplore. Restrictions apply.

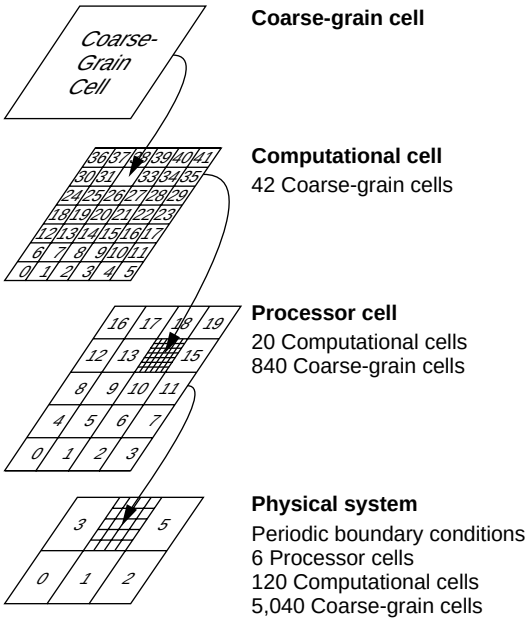

Figure 1. Domain decomposition allows fast parallel computation of molecular dynamics in materials fracture. Coarse-grain cells are grouped into rectangular blocks called computational cells, which in turn are grouped into processor cells. The processor cells, in aggregate, constitute the total physical system.

paradigm for studying classical many-body phenomena of atomistic systems in computational physics. The Lennard–Jones potential can be used to represent a "brittle" material in two dimensions[13] (though in three dimensions it is questionable). The longitudinal sound speed c_0 at zero temperature and pressure is $c_0 = (\kappa\varepsilon/m)^{1/2}$, where the constant κ is 81 for the 2D LJ solid and 72 for the 3D LJ solid. The transverse sound speed is $c_t = c_0/\sqrt{3}$. The Rayleigh speed c_R is approximately equal to the transverse sound speed.

Doing parallel MD computation

Molecular dynamics is easy to implement efficiently in a message-passing environment. We explicate our method here for completeness and to assist those starting new codes. The key to the efficient algorithm is an efficient technique for handling the sums over particles which appear in the dynamical equations.

If the summations over all N interacting particles had to be carried out for each of the N particles, we would have a problem with execution time scaling as N^2, a computational disaster for large systems. However, if the forces are short range the problem may be reduced so that the execution time is proportional only to N. To do so, we restrict the summation to those particles with an interatomic separation that is less than the range of interaction. Two techniques are used to efficiently find the interacting pairs of atoms: the use of coarse-grain tables[14] and the use of neighbor tables,[9] both implemented with infrequent update. The computational space is divided up into coarse-grained cells which are chosen so that in searching for the neighbors of a given atom, only the coarse-grained cell in which that atom is located and the nearest-neighbor and next-nearest-neighbor cells need to be considered. Since placing the particles in the proper cells can be done in linear time, we have reduced the problem from one which scales as N^2 to one which scales with N. By building explicit tables of the neighbors of a given atom (and updating infrequently) we can gain another factor of six or greater in speed.

So, with a parallel computer that has a number of processors increasing with the number of coarse-grain cells (the number of atoms per cell remaining constant), the computational burden remains constant. The task for computing the forces and updating the positions of the particles in a given cell is given to a processor of the parallel computer. To perform this task, the processor will first have to learn the positions of neighboring particles. Thus we are led to divide the parallel algorithm into a communication phase followed by a computation phase. Efficiency demands that the communication bandwidth grow linearly with the number of processors. This can be achieved by connecting the processors with a network that allows the communication to be done in parallel also. Each processor sends coordinates of its particles to its neighbors, say to the north, and simultaneously receives coordinates from its neighbor to the south. For a two-dimensional problem, eight such permutations are required, while for three dimensions, 26 are required.

The perfectly efficient parallel computer would exhibit linear speedup: the computation rate of P processors would be P times that of a single processor. However, communication and other global demands may dominate if a sufficiently fine-grained parallel structure is defined. If we identify the single-processor operation as an overhead step, we see that speedup depends in a sensitive way on the fraction of work done in parallel; if a large proportion of the effort

Authorized licensed use limited to: IEEE Xplore. Downloaded on October 23, 2008 at 15:38 from IEEE Xplore. Restrictions apply.

goes into overhead, the efficiently of the parallel computer drops dramatically.

Our discussion of the scheme we have used to run molecular dynamics on parallel computers will be for two-dimensional systems, but we point out that it can expanded in a straightforward, albeit tedious, manner to three dimensions. The scheme is based on a message-passing, SPMD (single program, multiple data) paradigm. Message passing means that memory shared by processors is not assumed or used. SPMD means that there is a single program, identical copies of which run on each processor.

Decomposition of the problem

The coarse-grain cells are grouped together in larger rectangular blocks which we call computational cells (Figure 1). These, in turn, are grouped into larger rectangular "processor cells." Finally, the aggregate of all the processor cells constitutes the total physical system under study. It is traditional for coarse-grain cells to be approximately square, and there are arguments for communication efficiency calling for the other cells to be approximately square, but our implementation does not require this. In fact, in some systems it may be useful to have computational cells that are strips traversing the entire width of the system.

As the names imply, if there are N processors available, then there will be N processor cells, with one assigned to each processor. Each processor is responsible for the particles in its own cell; and except for near neighbors as described below, has no knowledge of particles belonging to any other node. In this sense, the scheme is totally scalable. (The exceptions are global quantities such as total energy and temperature, which cause sequential bottlenecks in the computation.)

The reason for subdividing the processor cells into computational cells may not be immediately apparent. Historically, it was done during program development to allow the entire system to be run on a single processor ($N = 1$). It has been retained because it leads to virtually no code complication or performance overhead, and continues to be a useful tool by giving additional flexibility in the subdividing of the system. Finally, in the limit of large systems with many computational cells per processor, there is a possibility of achieving better load balancing by moving computational cells dynamically from one processor cell to another during the computation. The rest of this discussion will concern primarily computational cells, not processor cells. The reader may find it easier to think of the special case of one computational cell per processor cell; in this case the computational and processor cells are one and the same.

As already mentioned for the processor cells, each computational cell is responsible for only the particles assigned to it. For molecular dynamics using a force field with limited range, the only additional information the computational cell needs is the coordinates of particles in other computational cells in a limited region just outside the cell. In fact, since the cutoff distance for the assumed short-range potential is less than the size of the coarse-grain cells, the additional coordinates needed are precisely those in a monolayer of coarse-grain cells surrounding the computational cell. The computational cell has storage for all its own particles; these are stored in the lower part of an array. There is room in the top of the array for coordinates of particles from other cells. This way the computational cell's own particles are grouped together, but when it is useful, all the particles can be accessed in a single block.

Passing particle coordinates and forces

At the beginning of a molecular dynamics time step, a computational cell must collect neighboring coordinates to calculate the forces on its particles. Coordinates from eight neighboring computational cells are needed (or 26, for three dimensions). This works, but the forces for the particles near the surface will be

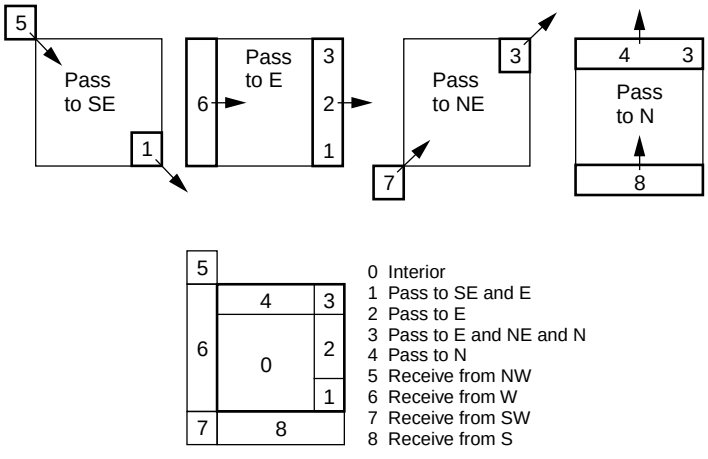

Figure 2. Scheme for collection and passing of particle coordinates in border areas. To fully calculate the forces on its own particles, a computational cell (bold outline in lower diagram) needs coordinates from the border area of neighboring computational cells.

Authorized licensed use limited to: IEEE Xplore. Downloaded on October 23, 2008 at 15:38 from IEEE Xplore. Restrictions apply.

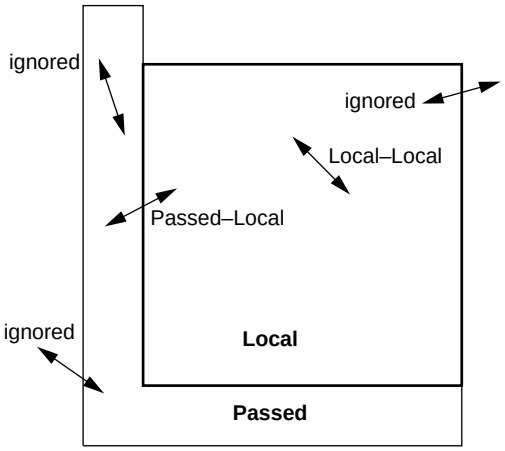

Figure 3. Interactions among particles in the interior of the computational cell are "local–local"; for particles near the surface, interactions are "passed–local." (The Verlet table is divided accordingly.) The remaining interactions are ignored by this local cell.

calculated twice, and those in the corner four times. It suffices to pass the coordinates from the coarse-grain cell monolayer (surrounding the computational cell in question) in just half the directions, and receive coordinates from the other half. Thus the appropriate coordinates are sent to the southeast, east, northeast, and north; coordinates are received from the northwest, west, southwest, and south (Figure 2). A careful analysis shows that coordinates in the coarse-grain cells near the corners of the computational cell in question are passed to or received from several neighbors. For example, in Figure 2, the coordinates of particles in region 3 (a single coarse-grain cell) are passed in three directions (E, NE, and N) while the coordinates in region 2 (a column of coarse-grain cells) are passed in only one direction (E). The interior coordinates in region 0 are not passed at all. If the computational cell is made up of a sufficiently large number of coarse-grain cells, most of the coordinates need not be passed.

Once the particle coordinates have been sent and received, each computational cell calculates the partial forces on all the particles known to it; the fact that the force on A due to B is equal and opposite to the force on B due to A is of course used. For the particles in the interior of the cell, the forces are complete. These are due to the interactions called "local–local" in Figure 3. The partial forces for particles near the surface, called "passed–local" in Figure 3, are now passed in the opposite direction and accumulated by the receiving cells. Following the example for the coordinates, partial forces are sent to the northwest, west, southwest, and south; partial forces are received from the southeast, east, northeast, and north. The amount of communication is the same, but the number of force calculations is reduced by communicating half the coordinates and half the forces. The Verlet table is set up from link lists in the normal way, but is divided into two parts: local–local interactions, followed by passed–local interactions. This division is convenient because the two types of interactions are handled somewhat differently. The other interactions depicted in Figure 3 are called "ignored"; they are the concern of various neighboring computational cells but are of no concern to this one, the local computational cell.

Each computational cell has its own local coordinate system in which its lower left corner is the origin. There are several advantages to having these local origins and using relative coordinates for all calculations. The most important is that it simplifies the handling of periodic boundary conditions. No computational cell needs to be concerned whether or not it is at the edge of the physical system; it knows only that there are other cells to the north, south, east, and west. Whenever coordinates are passed from one neighboring computational cell to another, they are adjusted so that the coordinates received are relative to the receiver's origin. No "wrap-arounds" are possible in subsequent calculations, so conditional code branches are avoided and the code is easier to read.

A second advantage is that lower-precision floating numbers may suffice to represent the coordinates, reducing the storage required or perhaps allowing the calculation to proceed with the precision available on a particular machine. This is because the force is a function of (among other things) the distance between two particles. If the particles are both far from the origin, precision is lost in the difference of two very large numbers. With a local origin, coordinates are smaller and the differences (distances between particles) are correspondingly more precise. For 8-byte floating numbers this may not be significant, but for 4-byte floating numbers and a large system, it can be shown to be important. Further thought shows that as particles move from one computational cell to another, and even across the entire physical system, they always see their near neighbors precisely and ad-

Authorized licensed use limited to: IEEE Xplore. Downloaded on October 23, 2008 at 15:38 from IEEE Xplore. Restrictions apply.

just their motion appropriately. Their positions relative to distant neighbors do not need to be known as precisely.

The remainder of the task requires explicit development of tables and bookkeeping details (as yet unpublished). The reason for the tables is that most of the geometric relationships can be worked out in advance. Then during the calculation there are very few decisions to be made; much is done by table lookup, thereby minimizing the number of conditional branches in the program. The importance of this depends on the particular computer architecture.

For another description of fast parallel algorithms for short-range molecular dynamics, we refer the reader to Plimpton's *Journal of Computational Physics* article.[15]

Computer experiments for two dimensions

Our early interest was to study two-dimensional "mode one" loading[8] by following the crack propagation over sufficient time and distance intervals so that a comparison with physical experiment became feasible. Our 2D system is a simulated rectangular solid slab of "rare-gas" atoms such as argon, krypton, or xenon. The slab has L atoms on a side, where $L = 712$ for a half-million-atom system and 1,424 for a two-million-atom system. The slab is initialized at a reduced temperature of 0.00001. A triangular notch of 10 lattice spacings is cut midway along the lower horizontal slab boundary, and an outward strain rate $\dot{\varepsilon}_x$ is imposed on the outermost columns of atoms defining the opposing vertical faces of the slab. In other words, we grasp opposite edges of the slab and try to pull it apart, assuming a crack will start at the notch we have cut. We found that a strain rate of $\dot{\varepsilon}_x = 0.0001$ is sufficiently small for systems of our size to prevent multiple fractures from forming in addition to the desired fracture at the notch. With this choice, the crack begins at the notch tip when the solid has been stretched by approximately 3 percent. We have also used a strain rate of $\dot{\varepsilon}_x = 0.00001$, lower by an order of magnitude, and longer notch lengths. At the onset of crack motion, the imposed strain rate remains constant (experiment 1) or is set to zero (experiment 2), and the simulation is continued until the growing crack has traversed the total length of the slab.

Figures 4 and 5 graphically summarize our nonzero-strain-rate simulation (experiment 1) for $L = 712$. Figure 4 shows (a) the crack-tip position (in units of reduced length) and (b) the crack-tip speed (in units of the Rayleigh sound speed), both as a function of reduced time. Figure 4c is an expanded view of the crack-tip speed for early time. From Figure 4a, we find that the crack tip achieves a limiting average speed equal to 0.57 of the Rayleigh speed c_R. However, the instantaneous tip velocity is very erratic (Figure 4b) after reaching a speed of $0.32c_R$. Before a time of about 70 and a speed less than $0.32c_R$, the acceleration of the crack tip is quite smooth (Figure 4c); but with the onset of the erratic fluctuations of the tip speed, the acceleration of the propagating crack lessens significantly. We also obtained data of the types shown in Figure 4 for experiment 2 and for the larger system (L = 1,424). The data for both experiments at both sizes show agreement with the physical experi-

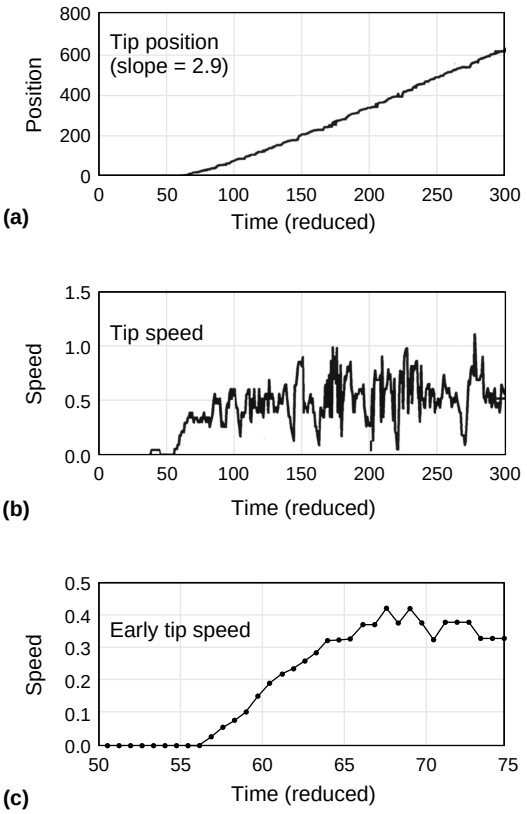

Figure 4. Fracture dynamics in a simulated 2D slab containing a half million rare-gas atoms. (a) The crack-tip position (in units of reduced length) as a function of reduced time. The slope is the limiting speed in reduced units, which corresponds to 0.57 of the Rayleigh speed c_R. (b) The crack-tip speed (in units of the Rayleigh sound speed) as a function of reduced time. (c) An expanded view of the crack-tip speed for early time.

Authorized licensed use limited to: IEEE Xplore. Downloaded on October 23, 2008 at 15:38 from IEEE Xplore. Restrictions apply.

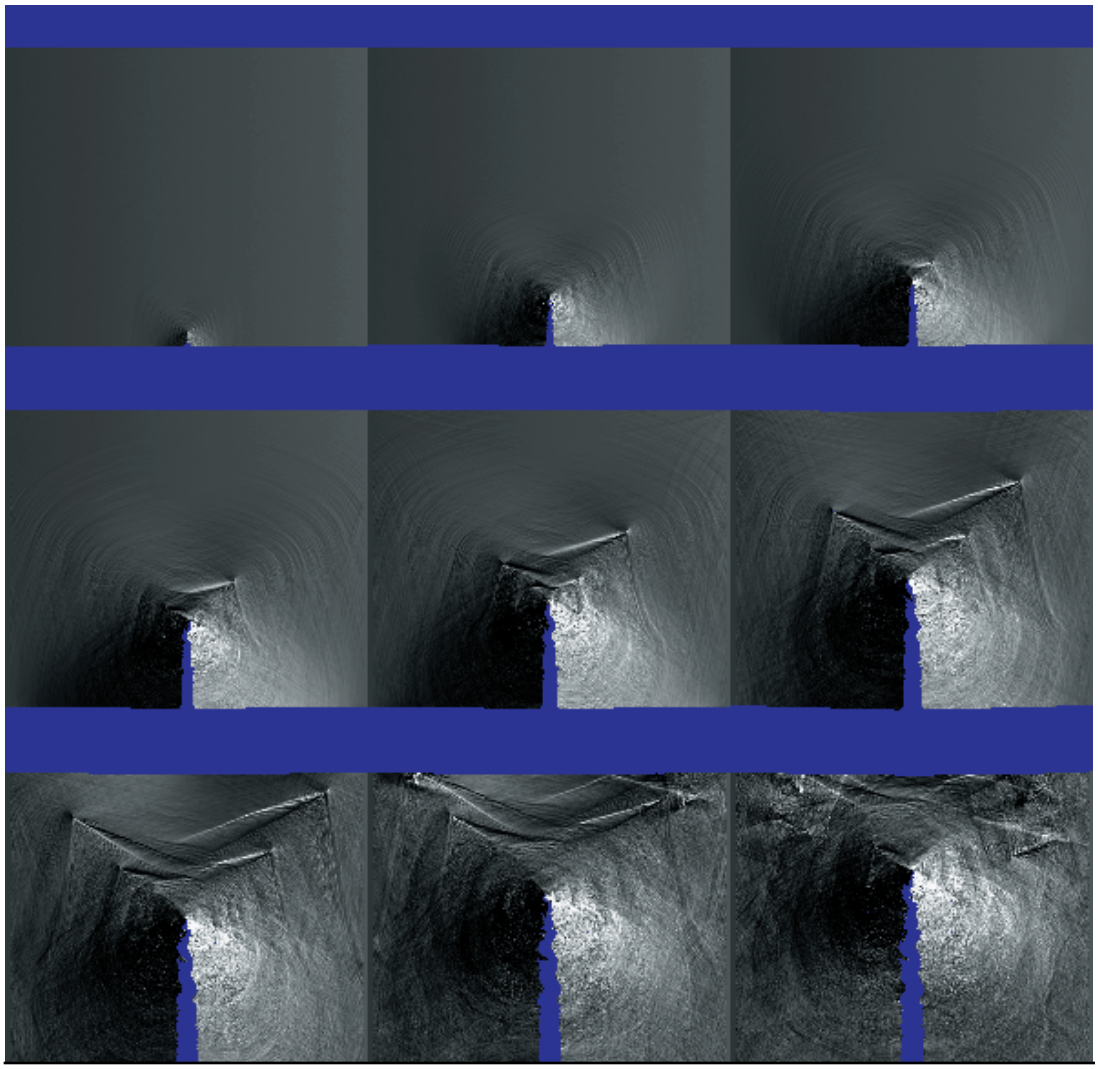

Figure 5. The time evolution of the propagating Lennard–Jones crack at constant strain rate of 0.0001 using a gray-scale rendering of the instantaneous local transverse velocity v_x, which goes from dark gray for the most negative v_x to light gray for the most positive v_x. The time sequence goes from left to right and top to bottom. The frames are for reduced times 100, 150, 175, 200, 225, 250, 275, 300, and 325.

ments of Fineberg et al.[3,4] As with the laboratory experiment, the influence of physical boundaries is a concern when sound and dynamical defects reflect from them. It should be noted that the onset of the instability in tip motion (t = 70) occurs significantly earlier than it takes sound to travel from the tip to a lateral boundary and return (t = 235). Hence, the transition to chaos seems to be an intrinsic instability.

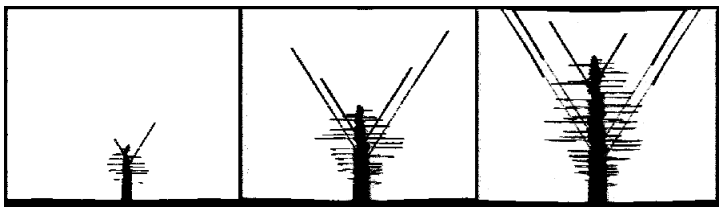

Figure 6. Snapshots of the dislocation trajectories for the LJ fracture simulation in Figure 5. The time sequence goes from left to right and is for reduced times 130, 235, and 360.

Figure 5 shows the time evolution of the propagating crack using a gray-scale rendering of the instantaneous local transverse velocity v_x. (By *transverse velocity*, we mean the velocity of the atoms in the direction of the pull caused by the imposed strain—that is, *perpendicular* to the forward motion of the crack.) The scale goes from dark gray for the most negative v_x to light gray for the most positive v_x. Initially, the brittle crack propagates in a straight line and leaves "mirror" cleaved surfaces. Periodic stress waves immediately appear with motion. Correspond-

Authorized licensed use limited to: IEEE Xplore. Downloaded on October 23, 2008 at 15:38 from IEEE Xplore. Restrictions apply.

ing to the onset of the erratic oscillations of the tip speed, at a tip speed of approximately $0.32c_R$, the crack first begins to roughen and then to oscillate back and forth at approximate angles of ≤30 degrees from the vertical (the original direction of crack motion), which are symmetry lines of the crystal. (The first stages of the roughening cannot be detected in Figure 5, but we will see them in a later figure at atomic detail.) Accompanying the oscillating zigzag excursions of the growing crack is significant relaxation in the regions immediately next to the newly created surfaces. Prominent "zigs" or "zags" in the crack direction are accompanied by a propagating atomic displacement along two adjacent rows of atoms (a slip plane) that are ±30 degrees to the vertical; it is initiated at the vertex of the change of direction and travels at about the longitudinal sound speed c_l. These dislocations appear as slanted, inverted *V*'s, which were surprising and mysterious when we first encountered them in our simulations. The *V*'s are being emitted from the moving crack tip, first to the right and then to the left. They may be traced back to their origin by constructing an imaginary line 30 degrees from the vertical and passing through the *V*. The vertical separation between neighboring dislocations equals the wavelength of the oscillating crack and is about 115 reduced length units. The explanation of the *V* is that it is simply an acoustical wake created by the moving dislocation. When the dislocation hits the top free boundary, an atomic step is formed. These off-angle slips provide an excellent signature for crack oscillation.

Another class of dislocations exists that are not apparent in the gray-scale pictures of the transverse velocity fields as presented in Figure 5. In Figure 6, the dislocation trajectories are shown at various times during fracture-tip motion. The off-angle slips are quite evident as lines emitted from the crack tip at plus and minus 30 degrees from the direction of motion. What is surprising is the appearance of an abundance of dislocations being emitted near the crack tip at right angles to the forward motion! By examination, we observe that these transverse dislocations go out some distance, then return to the fracture surface where they disappear. The spacing between these dislocations is quite regular. We can understand their origin as a transverse slippage between two neighboring rows of atoms arising from a growing shear stress at the crack tip as the ever-increasing cascade of broken bonds in the forward direction causes the increasing numbers of severed rows of atoms to want to relax laterally. This buildup of severed atomic rows will eventually be large enough to overcome the barrier to slippage. The slippage front will manifest itself as a dislocation. Of course, this is a repeating process, continually creating transverse dislocations. Also, the return of dislocations to the crack surface and the healing of the surface is a consequence of neighboring bands of matter bounded by slip planes relaxing to equilibrium.

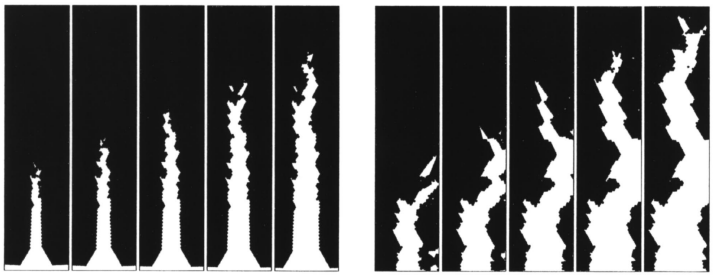

Figure 7. ***Left:*** **the onset of crack instability, in reduced time intervals of 7, beginning at time 85.** ***Right:*** **Late zigzag crack propagation, in reduced time intervals of 7, beginning at time 220. These simulation results appear to explain why a crack propagates significantly more slowly than theory predicts.**

To highlight the microscopic features of the failure dynamics, Figure 7 shows a short-time-interval sequence of close-up views of the crack tip at an early time and at a late time. We see the onset of the crack instability beginning as a roughening of the created surfaces, which eventually results in the pronounced zigzag tip motion; that is, "smooth → rough → zigzag" corresponds to "mirror → mist → hackle." The roughening occurs before the emission of dislocations. Also, the onset of roughening corresponds to a point in the crack-tip dynamics where the time it takes the tip to traverse one lattice constant (that is, one atomic separation distance between two neighboring rows of atoms) approximately equals the period for one atomic vibration. Hence, the bond-breaking process no longer sees a symmetric environment due to thermal averaging, but begins to see local atom configurations "instantaneously" distorted from perfect lattice symmetry. This gives rise to small-scale (atomic) fluctuations in the bond-breaking path, and hence atomic roughening. This roughening could trigger larger-scale deviations if the dynamics is intrinsically unstable. In the hackle region, the growth of the fracture is not simply a sharp one-dimensional cleavage progressing in a

Authorized licensed use limited to: IEEE Xplore. Downloaded on October 23, 2008 at 15:38 from IEEE Xplore. Restrictions apply.

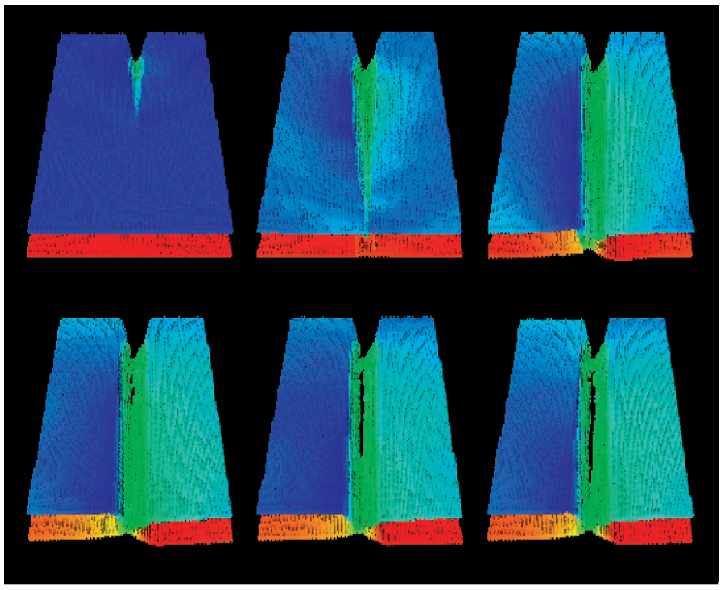

Figure 8. Time series of the mechanical grooving of a very thin 3D film, 12 atomic layers thick; a total of 5.5 million atoms. The coloring is according to depth; blue is top and red is bottom. The interior bulk atoms are not shown.

zigzag manner at 30 degrees from the mean crack direction. Instead, we see a stair-step growth of connected "ideal 30-degree segments," resulting in a net forward angle of less than 30 degrees from the vertical before changing the local direction by about 60 degrees.

Recall that continuum theory predicts a crack velocity about equal to the Rayleigh sound speed, but that physical experiment shows the velocity limited to around 0.6 c_R. The origin of the erratic velocity oscillations shown in Figure 4b is associated with the stair-step branching and connecting of regions of failure at and preceding the crack tip. While the "ideal 30-degree crack segments" open up at a velocity approximately equal to the Rayleigh speed, as predicted by theory, the oscillating zigzag motion of the crack tip and the segmented stair-step growth contribute to the effective "forward" crack speed being less than the theoretical prediction.

Thus, many of the recent laboratory findings occur in our simulation experiments, one of the most intriguing being the dynamic instability of the crack tip and its associated properties. Microscopic processes have been identified, and explanations of certain features have been suggested; in particular, the reason for the limiting velocity being significantly less than the theoretical limit. We conclude that the fracture behavior appears to be universal, or independent of materials structure, simply because we observe the laboratory phenomenon in our simulation of a two-dimensional simple atomic system.

Multimedia versions of our 2D atomistic simulation studies of fracture are available via the World Wide Web at http://www.almaden.ibm.com/vis/fracture/prl.html.

We now consider fracture in three dimensions.

Computer experiments for three dimensions

Theory suggests the inherent *ductility* of the rare-gas solids and, indeed, of most face-centered cubic solids at low temperature.[16–18] However, this is in sharp contrast to the *brittle* fracture of two-dimensional rare-gas films observed in our simulation studies.[8] What causes this discrepancy? Perhaps the brittle nature of 2D rare-gas films is a consequence of dimensionality of the modeled solid slab, and in three dimensions plasticity prevails as theory predicts.

Here we describe a simulation of 100 million atoms recently carried out at the Cornell Theory Center on the IBM RS/6000 SP parallel computer.[19] We now include the third dimension in our fracture simulations of rare-gas solids at very low temperature in order to address the issue mentioned above. We find that the rare-gas solid initially cleaves as a brittle solid, but soon undergoes a dynamical brittle-to-ductile transition; the propagation of the crack is arrested and dislocations proliferate.

In a thin-film study prior to modeling a full 3D thick slab, we considered a three-dimensional slab with $\mathbf{L}_x$ = 672 atoms by $\mathbf{L}_y$ = 672 atoms by $\mathbf{L}_z$ = 12 atoms for the three orthogonal sides. The notch is midway along $\mathbf{L}_x$ for y = 0, having a y extension of length $\mathbf{l}_t$ = 40 which extends through the entire thickness $\mathbf{L}_z$. Using common crystallographic terminology, the exposed notch faces are in the y-z planes with (110) faces, and the notch is pointed in the <1$\bar{1}$0> direction. Periodic boundary conditions are *not* imposed between the x-y faces at z = 0 and z = $\mathbf{L}_z$. This notched thin-slab geometry has approximately 5.5 million atoms. A uniform strain of 1.5 percent is imposed in the x-direction.

Figure 8 shows how the failure in the system evolves with time. The coloring denotes depth, blue being top and red being bottom. We show only those atoms that have a potential energy greater than –6.1, where the ideal bulk value is –6.3. The visible atoms are associated with faces of the slab and initial notch, surfaces created by crack motion, local interplanar separation asso-

Authorized licensed use limited to: IEEE Xplore. Downloaded on October 23, 2008 at 15:38 from IEEE Xplore. Restrictions apply.

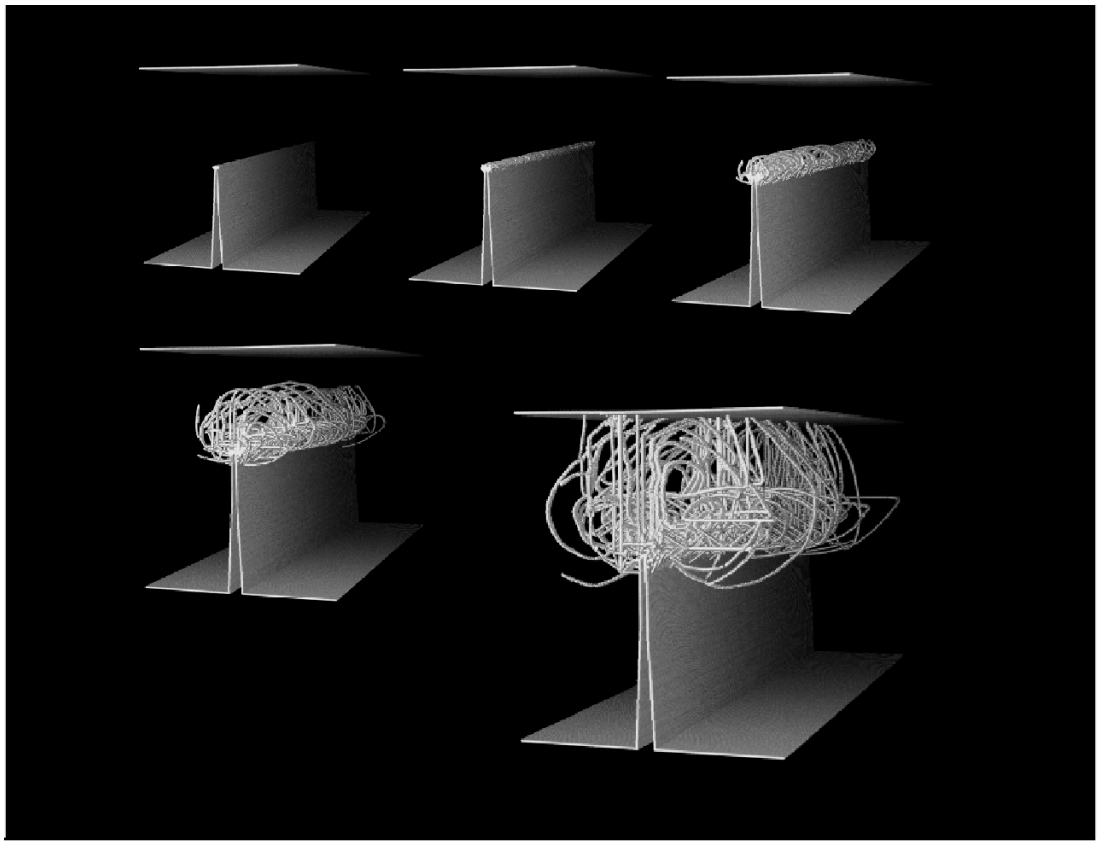

Figure 9. Time sequence of the crack and associated dislocations propagating through the 100-million-atom 3D slab for times 34, 43, 52, 68, and 90. Only atoms with a potential energy less than 97 percent of the bulk value are displayed, resulting in the selected visualization of atoms near surfaces and dislocations. The slab's top and bottom exterior surfaces are included, but the other exterior surfaces are omitted.

ciated with the material's dynamic failure at the tip, and topological defects created in the otherwise perfect crystal. As time progresses, grooves appear at the free surfaces in the region of the crack tip and grow outward from the notch as well as into the slab's depth. The notch grows in length as the depth of the grooves reaches half of the slab's thickness, this occurring farther away from the original tip position as time increases. This mechanical grooving is a consequence of loop dislocations being emitted from the crack edge, meeting the free surfaces, and forming steps that move away from the stationary tip as the dislocations move downstream. The original notch is not of sufficient length (for the imposed strain) to fail by breaking bonds; however, the strained slab still fails by dislocation grooving.

For our study of a full three-dimensional slab, the system had $\mathbf{L}_x = 336$ atoms by $\mathbf{L}_y = 336$ atoms by $\mathbf{L}_z = 896$ atoms on its three orthogonal sides. The notch has a y extension of length $\mathbf{l}_t = 120$. The exposed notch faces are (110) faces, and the notch is pointed in the $<1\bar{1}0>$ direction. Periodic boundary conditions are now imposed between the x-y faces at $z = 0$ and $z = \mathbf{L}_z$. This notched slab geometry has a total of 100,509,696 atoms, and the total simulation time for this study is 30,000 time steps or 135 in reduced units. It takes 12 CPU seconds per time step for a 306-node spatial decomposition simulation on the IBM RS/6000 SP. This timing compares very favorably with other timing studies reported for parallel molecular dynamics programs for comparable system sizes (for example, see the paper by Plimpton[15]). The slab is initialized at zero reduced temperature, and an outward strain rate $\dot{\varepsilon}_x$ is imposed on the outermost columns of atoms defining the opposing vertical faces of the slab. A linear velocity gradient is established across the slab, and an increasing lateral strain with time occurs in the solid slab with an applied strain rate of $\dot{\varepsilon}_x = 0.00033$. With this choice, the solid fails at the notch tip after 10,000 time steps when the solid has been stretched by about 1.5 percent. The imposed strain is set to zero at 13,500 time steps,

Authorized licensed use limited to: IEEE Xplore. Downloaded on October 23, 2008 at 15:38 from IEEE Xplore. Restrictions apply.

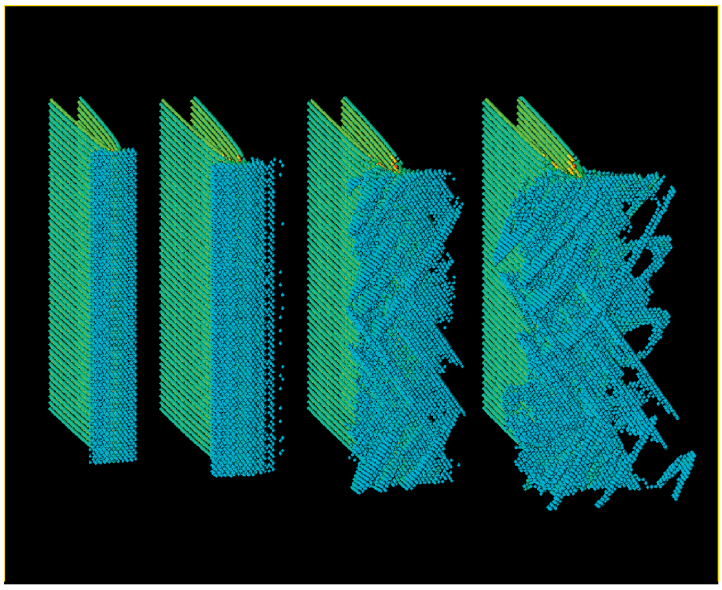

Figure 10. Magnified off-diagonal perspective views of the process zone during the time period of the brittle-to-ductile transition in the 100-million-atom 3D slab (at times 36, 43, 49, and 54). The coloring is according to the potential energy of the atom.

and the simulation is continued a further 20,000 time steps. We now adopt the initiation of crack-tip motion as zero time.

Figure 9 shows snapshots of the dynamics of the notched solid. Interior atoms are not shown, reducing the number of atoms seen by about two orders of magnitude. Because of the periodic boundary condition, there are no exterior front and back surfaces. The sequence shown is for times 34, 43, 52, 68, and 90 (in reduced time units).

In the time interval 0 to 42, the crack motion is representative of brittle fracture, with planar cleavage of the bonds between the two (110) neighboring atomic sheets defined by the initial notch. The small bud at the crack tip represents local expansion, and we will refer to this region as the "process zone." During brittle fracture, this process zone represents no plasticity, but, at a time of 43, the bud begins to blossom into a "flower of loop dislocations." This is apparent from the snapshot picture at time 52. Also, the crack slows to a stop and continues to dissipate elastic energy through the continued creation and motion of dislocations. Hence, a "dynamic" brittle-to-ductile (B–D) transition occurs in the fracturing of this rare-gas nanocrystal. We see the time evolution of the loop dislocations emitted from the arrested crack surface neighboring the vertical crack edge and propagating in the (111) family of slip planes through the solid slab. The dislocations move at approximately a quarter of the longitudinal sound speed (c_s = 8.5) and eventually terminate as steps (not shown) on the top surface of the slab.

Figure 10 shows magnified off-diagonal perspective views of the process zone in the 100-million-atom 3D simulation, during the time period of the brittle-to-ductile transition (at times 36, 43, 49, and 54). The color denotes potential energy of an atom. We note that just prior to the B–D transition (time 36) the process zone is symmetric. The zone then becomes asymmetric with the onset of a roughening as a quasi-one-dimensional strip (time 43) and rapidly broadens into a wall of embryonic dislocations (time 54). The crack surface has begun to roughen prior to the emission of dislocations. In our 2D fracture simulations,[8] we saw a similar roughening, which was identified as the onset of an intrinsic dynamical instability of the brittle fracture process. For the 3D simulation, this roughening extends in depth as a one-dimensional step which is a few atom diameters wide. However, the three-dimensional B–D fracture transition immediately follows, in sharp contrast to the fracture of the two-dimensional rare-gas films, where brittle fracture prevails. Furthermore, the surface roughening (that is, the brittle instability dynamics) occurs when the crack-tip velocity equals about 0.33 of the Rayleigh sound speed for the rare-gas solid, the same as we found in the two-dimensional systems.

A multimedia version of the 3D atomistic simulation of fracture is available via the World Wide Web at http://www.tc.cornell.edu/~farid/fracture/100million/.

A deeper understanding of how materials crack is of immense technical and financial importance in the modern world. We are now addressing the issue of reaching time and space scales of practical usefulness to the engineer and technologist, by developing techniques that bring together the continuum, atomistic, and quantum descriptions of materials in a seamless marriage for the study of fracture. The continuum description, used successfully by the applied mechanics community for decades, is proper except in the region of failure, where an atomistic description is required. The atomistic level is modeled by classical dynamics and empirical interatomic force laws, except where

Authorized licensed use limited to: IEEE Xplore. Downloaded on October 23, 2008 at 15:38 from IEEE Xplore. Restrictions apply.

bonds are breaking. At the point of bond breakage, in our ongoing work, a quantum description is being applied. ♦

Acknowledgments

I am grateful for discussions and collaborative efforts with W.E. Rudge, D. Brodbeck, and X. Xu at IBM Almaden, J.Q. Broughton at the Naval Research Laboratory, and D. Schneider, B. Land, D. Lifka, J. Gerner, and M. Rosenkrantz at the Cornell Theory Center. Much of this research was conducted using the resources of the Cornell Theory Center, which receives major funding from the National Science Foundation and New York State with additional support from the Advanced Research Projects Agency, the National Center for Research Resources at the National Institutes of Health, IBM Corporation, and members of the Corporate Research Institute.

References

1. H.J. Herrmann and S. Roux, eds., *Statistical Models for the Fracture of Disordered Media,* North-Holland, Amsterdam, 1990.
2. L.B. Freund, *Dynamical Fracture Mechanics,* Cambridge Univ. Press, New York, 1990.
3. J. Fineberg et al., "Instability in Dynamic Fracture," *Physical Review Letters,* Vol. 67, 1991, pp. 457–460.
4. J. Fineberg et al., "Instability in the Propagation of Fast Cracks," *Physical Review B,* Vol. 45, 1992, pp. 5146–5154.
5. S.P. Gross et al., "Acoustic Emissions from Rapidly Moving Cracks," *Physical Review Letters,* Vol. 71, 1993, pp. 3162–3165.
6. J.S. Langer, "Dynamic Model of Onset and Propagation of Fracture," *Physical Review Letters,* Vol. 70, 1993, pp. 3592–3594.
7. M. Marder and X. Liu, "Instability in Lattice Fracture," *Physical Review Letters,* Vol. 71, 1993, pp. 2417–2420.
8. F.F. Abraham et al., "Instability Dynamics of Fracture: A Computer Simulation Investigation," *Physical Review Letters,* Vol. 73, 1994, pp. 272–275.
9. M.P. Allen and D.J. Tildesley, *Computer Simulation of Liquids,* Clarendon Press, Oxford, UK, 1987.
10. W.G. Hoover, *Computational Statistical Mechanics,* Elsevier, Amsterdam, 1991.
11. F.F. Abraham, "Computational Statistical Mechanics Methodology, Applications and Supercomputing," *Advances In Physics,* Vol. 35, 1986, pp. 1–111.
12. A. Nordsieck, *Mathematics of Computation,* Vol. 16, 1962, p. 22.
13. B.L. Holian et al., "Effects of Pairwise Versus Many-Body Forces on High-Stress Plastic Deformation," *Physical Review A,* Vol. 43, 1991, pp. 2655–2661.
14. R.W. Hockney and J. Eastwood, *Computer Simulation Using Particles,* McGraw-Hill, New York, 1981.
15. S. Plimpton, "Fast Parallel Algorithms for Short-Range Molecular Dynamics," *J. Computational Physics,* Vol. 117, 1995, pp. 1–19.
16. A. Kelly and N.H. Macmillan, *Strong Solids,* Clarendon Press, Oxford, UK, 1986, p. 10.
17. A. Kelly, W.R. Tyson, and A.H. Cottrell, "Ductile and Brittle Crystals," *Philosophical Magazine,* Vol. 15, 1967, pp. 567–581.
18. J.R. Rice and R. Thomson, "Ductile vs. Brittle Behaviour of Crystals," *Philosophical Magazine,* Vol. 29, 1974, pp. 73–97.
19. F.F. Abraham et al., "Instability Dynamics in Three Dimensions: Atomistic Simulation Using 100 Million Atoms," to be published in *J. Mechanics and Physics of Solids;* also in *Statistical Mechanics in Physics and Biology,* D. Wirtz and T. Halsey, eds., MRS Symposium Proc. Vol. 463, Materials Research Soc., Pittsburgh, 1997.

Farid F. Abraham is a computational physicist at IBM's Almaden Research Center, San Jose, California. He has previously held positions at Stanford, the University of California at Santa Barbara, and the Universidad Autónoma Metropolitana in Mexico City. He received his BS and PhD in physics from the University of Arizona. Abraham pioneered the use of atomistic simulation methods in the study of microscopic liquid droplets, the liquid-vapor surface, and the liquid-solid interface. Later, his molecular dynamics simulations produced a number of previously unknown or unconfirmed results that in some cases contradicted theoretical wisdom, among them that 2D and 3D solids melt the same way, that spinodal decomposition really exists, and that solid membranes in natural form are not crumpled. Since 1993, he has investigated problems in computational materials science spanning length scales from microscopic to macroscopic, focusing on studies in friction, wear, lubrication, and materials failure. He is a fellow of the American Physical Society.

Abraham can be reached at IBM Research Division, Almaden Research Center, K18/D1, 650 Harry Road, San Jose, CA 95120-6099, USA; e-mail, farid@almaden.ibm.com.

Authorized licensed use limited to: IEEE Xplore. Downloaded on October 23, 2008 at 15:38 from IEEE Xplore. Restrictions apply.