

Pathfinder: A parallel search algorithm for concerted atomistic events

Aiichiro Nakano

*Collaboratory for Advanced Computing and Simulations, Department of Computer Science,
Department of Physics & Astronomy, Department of Chemical Engineering & Materials Science,
University of Southern California, Los Angeles, CA 90089-0242, USA*

Received 26 October 2006; received in revised form 5 November 2006; accepted 7 November 2006

Available online 1 December 2006

Abstract

An algorithm has been designed to search for the escape paths with the lowest activation barriers when starting from a local minimum-energy configuration of a many-atom system. The pathfinder algorithm combines: (1) a steered eigenvector-following method that guides a constrained escape from the convex region and subsequently climbs to a transition state tangentially to the eigenvector corresponding to the lowest negative Hessian eigenvalue; (2) discrete abstraction of the atomic configuration to systematically enumerate concerted events as linear combinations of atomistic events; (3) evolutionary control of the population dynamics of low activation-barrier events; and (4) hybrid task + spatial decompositions to implement massive search for complex events on parallel computers. The program exhibits good scalability on parallel computers and has been used to study concerted bond-breaking events in the fracture of alumina.

© 2006 Elsevier B.V. All rights reserved.

PACS: 02.70.-c; 02.70.Ns; 82.20.Db

Keywords: Transition state theory; Molecular dynamics; Parallel computing

1. Introduction

Many important material processes occur through a sequence of infrequent events [1,2]. An example is slow crack growth, such as stress corrosion cracking, in which a sequence of bond-breaking events over years leads to a catastrophic failure of a structure [3]. Enumeration of events with low activation barriers and accurate estimation of their barrier energies are essential for understanding microscopic mechanisms of the long-time dynamics as well as for predicting the lifetime of the structure.

Various computational methods have been proposed for carrying out an exhaustive search of activated events in many-atom systems [4,5], including the activation–relaxation technique [6], the dimer method [7], and a variety of eigenvector-following methods [8–11] especially those using the Lanczos algorithm to obtain the lowest eigenvalue of the Hessian matrix and the corresponding eigenvector [12]. In materials with complex microstructures, however, the search for activated events remains

a hard computational problem [13,14], since the events with the lowest activation barriers often involve unexpected combinations of elementary atomistic events [15]. It is thus of great importance to design an efficient algorithm with tractable computational complexity to systematically search for such concerted events.

Discrete abstraction [16,17] of atomic configurations enables the use of combinatorial techniques to systematically enumerate concerted events. For example, an atomic configuration can be abstracted as a graph $G = (S_v, S_e)$, in which atoms constitute the set of vertices S_v , and the edge set S_e consists of chemical bonds [18]. Graph-based topological analysis (e.g., shortest-path circuit analysis) of million-to-billion node chemical bond networks has been used successfully to discover complex atomistic events underlying impact-damage [19] and hardening [20] mechanisms of materials.

Another computational technique that can significantly accelerate the combinatorial search for concerted events is evolutionary computation [21,22]. In evolutionary algorithms, a population of candidate solutions in the search space is maintained, and its dynamics is controlled with various techniques (e.g.,

E-mail address: anakano@usc.edu (A. Nakano).

recombination and mutation) to obtain approximate solutions while avoiding the combinatorial complexity of the search.

Advanced parallel and distributed computing technologies are also expected to facilitate massive searches for concerted events. Event-search algorithms are often implemented as loosely-coupled parallel applications, in which multiple search tasks are executed concurrently on distributed computers [23–25]. When each search task becomes computationally demanding, a hybrid task + spatial decomposition approach [26,27] can be implemented using the communicator construct in the message passing interface (MPI) language [28], which is a natural migration path to hybrid Grid remote procedure call (GridRPC) + MPI programming on a Grid of geographically distributed parallel computers [29].

This paper presents the design of a search algorithm for activated events with low barrier energies, starting from a local minimum-energy configuration of a many-atom system. The pathfinder algorithm combines: (1) a steered eigenvector-following (SEF) method that guides a constrained escape from the convex region of the minimum and subsequently climbs to a transition state tangentially to the eigenvector corresponding to the lowest negative Hessian eigenvalue; (2) discrete abstraction of the atomic configuration to systematically enumerate concerted events as linear combinations of atomistic events (LCAE); (3) elitist control of the population dynamics of low activation-barrier events; and (4) hybrid task + spatial decompositions (HTSD) to implement massive searches on parallel computers. The program exhibits good scalability on parallel computers and has been used to study concerted bond-breaking events in the fracture of aluminum oxide.

This paper is organized as follows. The next section describes the pathfinder algorithm for systematic event search, and its parallelization is discussed in Section 3. Numerical results are presented in Section 4, and Section 5 contains summary.

2. Pathfinder algorithm

Consider a system of N atoms with its state specified by a $3N$ -dimensional vector $\mathbf{R} = [r_{1x}, r_{1y}, r_{1z}, \dots, r_{Nx}, r_{Ny}, r_{Nz}]^T \in \mathbb{R}^{3N}$, where $\mathbf{r}_i = [r_{ix}, r_{iy}, r_{iz}]^T \in \mathbb{R}^3$ is the position of the i th atom (\mathbb{R} is a set of real numbers, and the superscript T denotes a transpose). The forces \mathbf{F} on the atoms are computed from the potential energy function $V(\mathbf{R})$ as

$$\mathbf{F} = \begin{bmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_N \end{bmatrix} = \begin{bmatrix} -\partial V / \partial \mathbf{r}_1 \\ \vdots \\ -\partial V / \partial \mathbf{r}_N \end{bmatrix} = -\frac{\partial V}{\partial \mathbf{R}}. \quad (1)$$

Let \mathbf{R}^{init} be an initial state, which is a local energy-minimum such that $\mathbf{F}(\mathbf{R}^{\text{init}}) = 0$ and such that all the eigenvalues of the Hessian matrix,

$$\mathbf{H} = \partial^2 V / \partial \mathbf{R}^2 \in \mathbb{R}^{3N \times 3N}, \quad (2)$$

are positive at \mathbf{R}^{init} . (For systems with periodic boundary conditions, we filter out the zero-eigenvalue translational motions [30].)

The problem is to find a set of activated events with the lowest barrier energies, starting from \mathbf{R}^{init} . Within the framework of the transition state theory [2,31], we define an event as a triplet of states, $e = (\mathbf{R}^{\text{init}}, \mathbf{R}^{\text{tst}}, \mathbf{R}^{\text{fin}})$, that are interconnected by a continuous escape path $\mathbf{R}(\tau)$ ($\mathbb{R} \rightarrow \mathbb{R}^{3N}$; τ is a real-valued parameter such that $\mathbf{R}^{\text{init}} = \mathbf{R}(\tau_{\text{init}})$, $\mathbf{R}^{\text{tst}} = \mathbf{R}(\tau_{\text{tst}})$, and $\mathbf{R}^{\text{fin}} = \mathbf{R}(\tau_{\text{fin}})$ with $\tau_{\text{init}} < \tau_{\text{tst}} < \tau_{\text{fin}}$). The ascent path $\mathbf{R}(\tau_{\text{init}} \leq \tau \leq \tau_{\text{tst}})$ connects \mathbf{R}^{init} to a transition state, taken here to be a saddle point \mathbf{R}^{tst} , at which $\mathbf{F}(\mathbf{R}^{\text{tst}}) = 0$, and at which only the lowest eigenvalue λ_1 of the Hessian matrix is negative. The final state \mathbf{R}^{fin} is another local energy-minimum that is reached along a steepest-descent path $\mathbf{R}(\tau_{\text{tst}} \leq \tau \leq \tau_{\text{fin}})$, starting from $\mathbf{R} = \mathbf{R}^{\text{tst}}$ pushed slightly away from \mathbf{R}^{init} . The barrier energy of event e is defined as $b(e) = V(\mathbf{R}^{\text{tst}}) - V(\mathbf{R}^{\text{init}})$.

The pathfinder algorithm generates a set of events with low barrier energies in such a way that concerted events are systematically constructed from elementary events. Each event, in turn, is generated from an event seed based on a steered eigenvector-following algorithm. Section 2.1 first defines the event seed and then describes the generation of a single event by the steered eigenvector-following algorithm. Systematic construction of concerted events through the control of event-population dynamics in the pathfinder algorithm is described in Section 2.2.

2.1. Steered eigenvector-following (SEF) event generator

In order to initiate an ascent path $\mathbf{R}(\tau_{\text{init}} \leq \tau \leq \tau_{\text{tst}})$ from the initial state, $\mathbf{R}(\tau_{\text{init}}) = \mathbf{R}^{\text{init}}$, to a transition state, $\mathbf{R}(\tau_{\text{tst}}) = \mathbf{R}^{\text{tst}}$, we first define an event seed σ as a parameterized sequence of $(3N - 1)$ -dimensional surfaces $S(\tau)$, in which the atoms' moves are constrained. A specific example for the slow crack-growth problem is a bond-length constraint imposed on a given atomic pair (i, j) ,

$$\sigma = \{S(\tau)\} = \{\|\mathbf{r}_{ij}\| = r_{ij}(\tau) = r_{ij}^0 + \dot{r}_{ij}(\tau - \tau_{\text{init}})\}, \quad (3)$$

where $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$, r_{ij}^0 is their bond length in the initial state, and \dot{r}_{ij} is the bond-stretching rate along the path.

The steered eigenvector-following event generator algorithm consists of three algorithmic phases (see Table 1): (1) steered centrifugal escape from the convex region (in which the Hessian matrix is positive definite) of the initial energy-minimum; (2) eigenvector-following climb to a transition state; and (3) steepest descent to reach a final energy-minimum [32].

The steered centrifugal escape phase starts from the initial state \mathbf{R}^{init} , and performs a sequence of steepest-descent steps,

$$\mathbf{R} \leftarrow \mathbf{R} + \frac{\delta\tau^2}{2\langle m \rangle} \mathbf{F}, \quad (4)$$

where $\delta\tau$ (~ 1 fs) is a time-discretization unit, and $\langle m \rangle$ is the average mass of the atoms. (Various energy-minimization methods can be used in this step, such as variable-step steepest-descent [12], conjugate-gradient [11] and quasi-Newton [4] methods.) Each steepest-descent step is followed by the projection of state \mathbf{R} onto the constrained surface,

$$\mathbf{R} \leftarrow P(S(\tau))\mathbf{R}, \quad (5)$$

Table 1
Steered eigenvector-following event generation algorithm

Algorithm event_generator

Input:

$\mathbf{R}^{\text{init}} \in \mathbb{R}^{3N}$: an initial local minimum-energy state
 $\sigma = \{S(\tau)\}$: an event seed, i.e. a parameterized sequence of $(3N - 1)$ -dimensional constraint surfaces

Output:

$e = (\mathbf{R}^{\text{init}}, \mathbf{R}^{\text{tst}}, \mathbf{R}^{\text{fin}})$: an event, i.e. a triplet of initial, transition, and final states

Steps:

1. *Steered centrifugal escape*

$\tau \leftarrow 0$
 $\mathbf{R} \leftarrow \mathbf{R}^{\text{init}}$
do
 $\tau \leftarrow \tau + \delta\tau$
 $\mathbf{R} \leftarrow \mathbf{R} + (\delta\tau^2/2(m))\mathbf{F}$ // steepest-descent step
 $\mathbf{R} \leftarrow P(S(\tau))\mathbf{R}$ // projection onto the constraint surface
while $\lambda_1 \geq -\Delta\lambda_1$

2. *Eigenvector-following climb*

do
 $\mathbf{R} \leftarrow \mathbf{R} - \frac{\delta\tau^2}{2(m)}(\mathbf{V}^1\mathbf{V}^{1T})\mathbf{F} + \frac{\delta\tau^2}{2(m)}(\mathbf{I} - \mathbf{V}^1\mathbf{V}^{1T})\mathbf{F}$ // eigenvector-following step
while $\max_{i\alpha}\{|f_{i\alpha}| \mid i = 1, \dots, N; \alpha = x, y, z\} > \Delta f$
 $\mathbf{R}^{\text{tst}} \leftarrow \mathbf{R}$

3. *Steepest descent*

$\mathbf{R} \leftarrow \mathbf{R}^{\text{tst}} + \delta_{\text{os}}(\mathbf{R}^{\text{tst}} - \mathbf{R}^{\text{init}})$ // push the state over the transition state away from the initial state
do
 $\mathbf{R} \leftarrow \mathbf{R} + (\delta\tau^2/2(m))\mathbf{F}$ // steepest-descent step
while $\max_{i\alpha}\{|f_{i\alpha}| \mid i = 1, \dots, N; \alpha = x, y, z\} > \Delta f$
 $\mathbf{R}^{\text{fin}} \leftarrow \mathbf{R}$

corresponding to the current time τ , where $P(S(\tau))$ is the projection operator [33]. For the bond-length constraint in Eq. (3), the projection operator is expressed as [34]

$$P(S(\tau))\mathbf{r}_k = \mathbf{r}_k + \frac{\delta_{ki} - \delta_{kj}}{2} \left(\frac{r_{ij}(\tau)}{\|\mathbf{r}_{ij}\|} - 1 \right) \mathbf{r}_{ij} \quad (k = 1, \dots, N), \quad (6)$$

where $\delta_{ki} = 1$ (if $k = i$) and 0 (else).

After each constrained steepest-descent step, the minimum eigenvalue λ_1 of the Hessian matrix is computed iteratively using the Lanczos algorithm [4,12] in Appendix A. We use a finite-difference method to evaluate the product of the Hessian matrix \mathbf{H} and a vector $\mathbf{Q} \in \mathbb{R}^{3N}$,

$$\mathbf{H}(\mathbf{R})\mathbf{Q} = c_{\text{fd}}[-\mathbf{F}(\mathbf{R} + \mathbf{Q}/c_{\text{fd}}) + \mathbf{F}(\mathbf{R})], \quad (7)$$

so that only the forces but not the Hessian matrix need to be computed. We use various divide-and-conquer algorithms to compute the forces in Eq. (7) in $O(N)$ time. For example, a space-time multiresolution molecular dynamics (MRMD) algorithm [35] and a fast reactive force-field (F-ReaxFF) algorithm [36] are used in cases of classical interatomic potentials and semi-classical reactive force fields, respectively. To compute the forces quantum-mechanically from the Hellmann–Feynman theorem, we use an embedded divide-and-conquer density-functional-theory (EDC-DFT) algorithm [37]. Consequently, the computational complexity of the pathfinder algorithm is $O(N)$. In Eq. (7), $c_{\text{fd}} = \max_{i\alpha}\{|q_{i\alpha}| \mid i = 1, \dots, N; \alpha = x, y, z\}/\delta_{\text{fd}}$ and δ_{fd} ($\sim 10^{-2}$ Å) is a discretization unit for finite differencing. It typically requires 4–8 force evaluations for

λ_1 to converge within a convergence criterion Δ_{eigen} ($\sim 10^{-3}$). The steered centrifugal escape steps are terminated when λ_1 becomes negative. For systems with a large number of small Hessian eigenvalues (due to floppy oscillations of dangling bonds) such as amorphous solids, we alternatively introduce a control parameter, $-\Delta\lambda_1$ (~ -10 eV/Å²), to terminate the escape steps when $\lambda_1 < -\Delta\lambda_1$.

Once the minimum Hessian eigenvalue becomes sufficiently negative, the eigenvector-following climb phase performs steepest ascent parallel to the Hessian eigenvector,

$$\mathbf{V}^1 = \begin{bmatrix} \mathbf{v}_1^1 \\ \vdots \\ \mathbf{v}_N^1 \end{bmatrix} \in \mathbb{R}^{3N}, \quad (8)$$

corresponding to λ_1 and steepest descent perpendicular to it [4,11,12]:

$$\mathbf{R} \leftarrow \mathbf{R} - \frac{\delta\tau^2}{2(m)}(\mathbf{V}^1\mathbf{V}^{1T})\mathbf{F} + \frac{\delta\tau^2}{2(m)}(\mathbf{I} - \mathbf{V}^1\mathbf{V}^{1T})\mathbf{F}, \quad (9)$$

where \mathbf{I} is the $3N$ by $3N$ identity matrix, and $\mathbf{V}^{(1)}$ is normalized as

$$\|\mathbf{V}^1\| = \left(\sum_{i=1}^N \|\mathbf{v}_i^1\|^2 \right)^{1/2} = 1. \quad (10)$$

At a transition state, the forces are zero, whereas the energy takes a minimum value for all directions except for \mathbf{V}^1 , along which the energy is instead maximum. Thus the eigenvector-following climb, through steepest ascent parallel to \mathbf{V}^1 and

Table 2
Pathfinder algorithm to search for concerted events with low activation barriers

Algorithm pathfinder

Input:

$\mathbf{R}^{\text{init}} \in \mathbb{R}^{3N}$: an initial local minimum-energy state
 $\{\sigma(k) \mid k = 1, \dots, N_{\text{seed}}\}$: a set of N_{seed} elementary event seeds

Output:

$\{e(k) \mid k = 1, \dots, N_{\text{elite}}\}$: a set of N_{elite} events with the lowest activation barriers

Steps:

1. *Elementary (singly-excited) event generation*

for $k = 1$ to N_{seed}
 call event_generator: $e(k) \leftarrow G(\sigma(k))$
 $N_{\text{event}} \leftarrow N_{\text{seed}}$

2. *Multiply-excited event generation*

for excitation = 2 to $Max_excitation$
 $N_{\text{combination}} \leftarrow 0$
 for $\forall(\sigma(k), \sigma(l))(k, l \in [1, N_{\text{event}}]; k < l)$
 $\sigma \leftarrow \sigma(k) \cup \sigma(l)$ // composite event seed as a union
 if $m(\sigma) = excitation$
 $N_{\text{combination}} \leftarrow N_{\text{combination}} + 1$
 $\sigma(N_{\text{event}} + N_{\text{combination}}) \leftarrow \sigma$
 $b_{\text{estimate}}(\sigma(N_{\text{event}} + N_{\text{combination}})) \leftarrow b(e(k)) + b(e(l))$ // estimated barrier energy
 sort $\sigma(N_{\text{event}} + 1 : N_{\text{event}} + N_{\text{combination}})$ in ascending order of b_{estimate}
 for $k = N_{\text{event}} + 1$ to $N_{\text{event}} + \min(N_{\text{combination}}, N_{\text{add_event}})$ // generate only $N_{\text{add_event}}$ new events
 call event_generator: $e(k) \leftarrow G(\sigma(k))$
 $b(e(k)) \leftarrow V(\mathbf{R}^{\text{tst}}) - V(\mathbf{R}^{\text{init}})$ // actual barrier energy
 $N_{\text{event}} \leftarrow N_{\text{event}} + \min(N_{\text{combination}}, N_{\text{add_event}})$
 sort $e(1 : N_{\text{event}})$ in ascending order of b
 $N_{\text{event}} \leftarrow \min(N_{\text{event}}, N_{\text{elite}})$ // retain only N_{elite} new events

steepest descent perpendicular to it, converges to a transition state. The eigenvector-following climb steps are terminated, when the maximum force component of every atom falls below a prescribed threshold value: $\max_{i\alpha} \{|f_{i\alpha}| \mid i = 1, \dots, N; \alpha = x, y, z\} < \Delta f$ (~ 0.1 eV/Å).

Once the eigenvector-following climb converges to a transition state \mathbf{R}^{tst} , the state is pushed slightly away from \mathbf{R}^{init} [6],

$$\mathbf{R} \leftarrow \mathbf{R}^{\text{tst}} + \delta_{\text{os}}(\mathbf{R}^{\text{tst}} - \mathbf{R}^{\text{init}}), \quad (11)$$

where the dimensionless overshoot parameter δ_{os} (~ 0.1) is an input parameter to the algorithm. The algorithm then performs steepest-descent steps, Eq. (4), until the maximum force component becomes less than Δf , signifying the convergence to a final local energy-minimum \mathbf{R}^{fin} .

2.2. Concerted event generation by discrete linear combination of atomistic events (LCAE)

The event generator in Section 2.1 defines a mapping, $e \leftarrow G(\sigma)$, from seed σ to event e . To systematically search for events with low barrier energies, we introduce a discrete indexing scheme, which allows the use of combinatorial search techniques. For a specific example of the bond-length constraint in Eq. (3), we first define a composite seed σ as a set of distinct atomic pairs, $l(\sigma) = \{p_1, \dots, p_{m(\sigma)}\}$, along with the bond-length constraints, Eq. (3), on the pairs. Here, the excitation level $m(\sigma)$ of seed σ is defined as the number of atomic pairs, p_i ($i = 1, \dots, m(\sigma)$), which constitute the seed. An event seed σ is thus indexed uniquely by a set l of distinct atomic pairs. For

example, $\{(15, 783), (47, 875), (175, 811)\}$ is a seed of excitation level 3 consisting of atomic pairs (15, 783), (47, 875) and (175, 811), where the atoms are indexed by positive integers. Similarly, an event $e = G(\sigma)$ is indexed according to its seed σ , from which it is generated. A population of events is stored as an array of the event data type that consists of the atomic-pair list of its seed, the triplet of its initial-, transition-, and final-state energies, and other attributes such as the estimated and actual barrier energies. In addition, the atomic configurations of the transition and final states are stored in files.

The pathfinder algorithm in Table 2 generates progressively more complex composite events, starting from a set of elementary event seeds, $\{\sigma(k) \mid k = 1, \dots, N_{\text{seed}}\}$, which is an input to the algorithm. An example of elementary event seeds for the slow crack-growth problem is a set of bond-stretching event seeds for all pairs of atoms that are within a cut-off radius from a crack tip. The algorithm first generates N_{seed} elementary events from the elementary seeds by calling algorithm event_generator in Table 1: $e(k) \leftarrow G(\sigma(k))$ ($k = 1, \dots, N_{\text{seed}}$).

In order to construct concerted events from these elementary events, we construct composite event seeds as unions of simpler seeds. Here, a union, $\sigma = \sigma(k) \cup \sigma(l)$, of a seed-pair $(\sigma(k), \sigma(l))$ is defined as the union of their corresponding atomic-pair sets, $l(\sigma(k)) \cup l(\sigma(l))$, along with the bond-length constraints, Eq. (3), on all constituent atomic pairs. The corresponding composite event is generated as

$$e = G(\sigma) = G(\sigma(k) \cup \sigma(l)). \quad (12)$$

The pathfinder algorithm maintains a population of events, $S = \{e(1), \dots, e(N_{\text{event}})\}$, where $N_{\text{event}} = |S|$ is the number of events in the population. At the beginning of the algorithm, $N_{\text{event}} = N_{\text{seed}}$ and all events are singly excited, i.e. $m(\sigma(k)) = 1$ ($k = 1, \dots, N_{\text{event}}$). The algorithm then loops over excitation levels from 2 to $Max_excitation$, where the control parameter $Max_excitation$ specifies the maximum excitation level considered by the algorithm. At each excitation level, all pairs of the events (or their seeds) in S are considered as candidates for creating new composite events by the union operation. A composite event (or its seed σ) is counted as a new event, only if its number of pairs $m(\sigma)$ is equal to the excitation level under consideration and its atomic-pair set $l(\sigma)$ is distinct from those of all the events in S .

In order to prune the combinatorial search space, we first define the estimated barrier energy of a composite event seed, $\sigma = \sigma(k) \cup \sigma(l)$, as $b_{\text{estimate}}(\sigma) = b(e(k)) + b(e(l))$. After enumerating all new composite events (let the number of which be $N_{\text{combination}}$), we sort them in ascending order of b_{estimate} . To avoid combinatorial explosion of the number of events to be considered, we retain only the $N_{\text{add_event}}$ lowest (estimated) barrier-energy events out of $N_{\text{combination}}$, where $N_{\text{add_event}}$ is one of the control parameters of the algorithm. The pathfinder algorithm calls Algorithm event_generator in Table 1 to generate events for the $N_{\text{add_event}}$ new seeds and to estimate their actual barrier energies. We then increment the number of event N_{event} by $N_{\text{add_event}}$ and sort all the events in ascending order of the actual barrier energy b . We retain only the N_{elite} lowest (actual) barrier-energy events for the next excitation level, where N_{elite} is another control parameter.

3. Parallelization by hybrid task + spatial decompositions (HTSD)

The pathfinder algorithm has been implemented on parallel computers by first assigning different events to separate processors (task decomposition) and then using spatial decomposition within each task for further parallelization. The parallel program is written in Fortran 90 and message passing interface (MPI) [28] languages, in which all processors constitute an overall MPI communicator, `MPI_COMM_WORLD`, and processors are grouped into different event groups by defining multiple MPI communicators as subsets of `MPI_COMM_WORLD`. (The MPI communicator construct combines a processor group and a context, in such a way that messages with different contexts are not intermixed.) In our program, each event calculation is assigned a dedicated communicator. One advantage of the hybrid task + spatial decomposition approach [26,27] implemented with MPI communicators is that the program can be easily converted to a hybrid Grid remote procedure call (GridRPC) + MPI program to be run on a Grid of distributed parallel computers, in which the number of processors change dynamically on demand and resources are allocated and tasks are migrated adaptively in response to unexpected faults [29].

The total number of processors is given by $P = M_c \times P_c$, where M_c is the number of communicators and P_c is the num-

ber of processors in each communicator. The number of events to be generated at each algorithmic step is typically larger than the number of communicators M_c , and thus communicator $c \in [0, M_c - 1]$ is assigned a set of events $\{k \mid (k - 1) \bmod M_c = c\}$. In spatial decomposition within each task [35], the total volume of the system is divided into P_c subsystems of equal volume, and each subsystem is assigned to a processor in an array of P_c processors. To calculate the force on an atom in a subsystem, the coordinates of the atoms in the boundaries of neighbor subsystems are “cached” from the corresponding processors. After updating the atomic positions due to a steepest-descent/ascent procedure, some atoms may have moved out of its subsystem. These atoms are “migrated” to the proper neighbor processors. With the spatial decomposition, the computation scales as N/P_c , while communication scales in proportion to $(N/P_c)^{2/3}$. Tree-based algorithms such as the fast multipole method (FMM) [38] incur an $O(\log P_c)$ overhead, which is negligible for coarse-grained ($N/P_c \gg P_c$) applications [39].

4. Numerical results

Scalability of the parallel pathfinder algorithm has been tested on a cluster of dual-core, dual-processor AMD Opteron (at clock speed 2 GHz) nodes with Myrinet interconnect, with 4 GB of memory per 4-core node. We define the speed of a program as a product of the total number of atoms and search steps executed per second. The speedup is the ratio between the speed of P processors and that of one processor. The parallel efficiency is the speedup divided by P .

First, we have performed a strong-scaling (or fixed problem-size) test to measure the efficiency of task decomposition parallelism. Here, the system is a cracked Al_2O_3 crystal consisting of 1920 atoms, and multiple communicators ($M_c = 32, \dots, 512$) of size $P_c = 1$ explore a large search space. We choose $Max_excitation = 2$ and $N_{\text{add_event}} = N_{\text{elite}} = 512$. The test uses all four cores per node. Fig. 1 shows the speedup of the parallel pathfinder program over that on 32 processors (we normalize the speedup on 32 processors as 32). The measured speedup

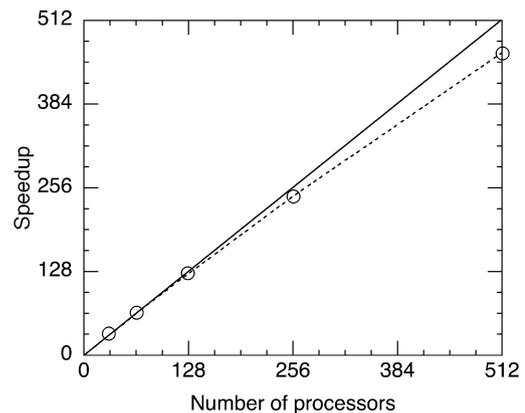


Fig. 1. Strong-scaling (fixed problem-size) speedup of the parallel pathfinder algorithm over 32 processors (normalized so that the speedup is 32 for $P = 32$) as a function of the number of processors P for a 1920-atom cracked Al_2O_3 system on dual-core, dual-processor AMD Opteron nodes. The circles are measured speedups, whereas the solid line denotes the perfect speedup.

on 512 processors is 463.0, and thus the parallel efficiency is 0.904. Although multiple events are generated independently on multiple processors, the parallel algorithm involves sequential bottlenecks such as the sorting of events, and accordingly the parallel efficiency degrades for a larger number of processors.

Next, we have performed a weak-scaling (or isogranular) test to measure the efficiency of spatial decomposition parallelism. In addition to exploring a large number of events for a relatively small number of atoms, the pathfinder program often uses a single communicator to evaluate the barrier energies of a few well-defined events for a larger system. This is the case in multimillion-atom simulations of fracture [40], impact [19], and indentation [20] of materials on a large number of processors P_c . In the weak-scaling test, the number of atoms is scaled linearly with the number of processors. Specifically, we choose $N = 14400P_c$, whereas the number of communicators is fixed as $M_c = 1$. Here, we choose $Max_excitation = 1$ and $N_{add_event} = N_{elite} = 1$. Fig. 2 shows the total execution and communication times of the parallel pathfinder program on the

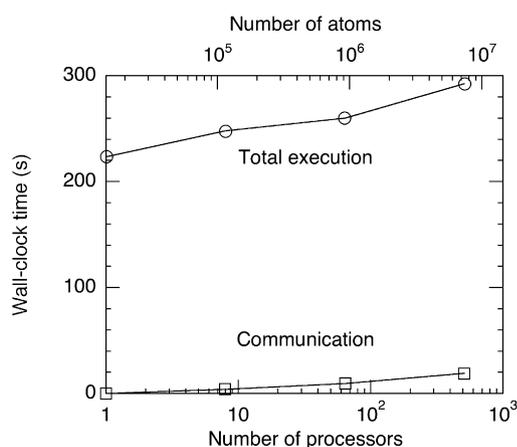


Fig. 2. Weak-scaling (isogranular) test of the parallel pathfinder algorithm on dual-core, dual-processor AMD Opteron nodes. The total execution (circles) and communication (squares) times are plotted as a function of the number of processors P for $14400P$ -atom Al_2O_3 systems.

Opteron cluster for the number of processors $P = 1, \dots, 512$. (The largest number of atoms is 7372800 for $P = 512$.) All four cores per dual-processor, dual-core node are used for the test, except for $P = 1$, where only one core is used. The execution time increases slightly for large P , and the parallel efficiency is 0.764 on 512 processors.

The isogranular parallel efficiency is typically used for very large simulations that are performed for a small number of steps. The large granularity, N/P , in such applications makes the parallel efficiency nearly perfect (~ 1). For example, we have recently performed benchmark tests including 134 billion-atom space–time multiresolution molecular dynamics (MD) [35], 1.06 billion-atom reactive force-field MD [36], and 11.8 million-atom (1.04 trillion grid points) quantum-mechanical MD in the framework of the divide-and-conquer density functional theory on adaptive multigrids [37], with the parallel efficiency as high as 0.998 on 65536 dual-processor BlueGene/L processors [41]. We expect the isogranular parallel efficiency of the parallel pathfinder algorithm to become similarly high for such large-scale applications.

To illustrate the use of pathfinder, we simulate a 1920-atom α -crystalline Al_2O_3 with a crack propagating in the $\langle 2\bar{1}\bar{1}0 \rangle$ direction in the $\{01\bar{1}0\}$ plane (Fig. 3). The initial state is prepared by first imposing displacements to the atoms according to a linear elastic crack solution corresponding to the stress intensity factor of $1.25 \text{ MPa}\sqrt{\text{m}}$ [3], and then relaxing the atomic configuration to the local energy-minimum, while fixing the positions of the two outer atomic layers in the $\langle 2\bar{1}\bar{1}0 \rangle$ and $\langle 01\bar{1}0 \rangle$ directions. The periodic boundary condition is applied in the $\langle 0001 \rangle$ direction. The simulation uses an interatomic potential consisting of two- and three-body terms, which is similar to those used in previous simulations [19,20,40]. The set of elementary event seeds consists of 43 bonds that are within 2.5 \AA from the crack tip. We choose $Max_excitation = 4$ and $N_{add_event} = N_{elite} = 128$.

Fig. 3 shows the resulting events with 60 lowest barrier energies, which are a mixture of singly- to quadruply-excited events. Such multiplicity of low activation-barrier events is common in crack growth, which often involves complex events

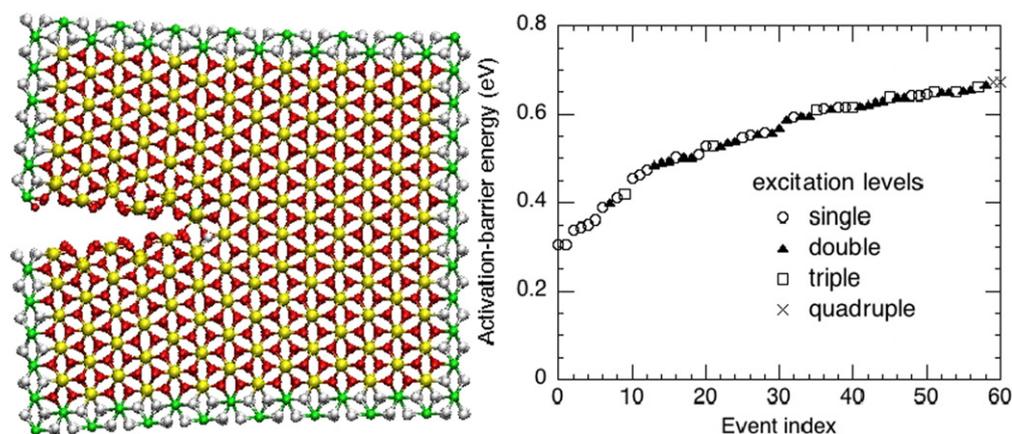


Fig. 3. (Left) The initial state of the 1920-atom cracked Al_2O_3 system, where yellow and red spheres are Al and O atoms, respectively. The positions of the Al (green) and O (grey) atoms at the outer layers are fixed according to a linear-elastic crack solution. (Right) Events with the lowest barrier energies and their excitation levels.

Algorithm Lanczos**Input:** $\mathbf{R} \in \mathbb{R}^{3N}$: a statelogical *initialize*: TRUE for the first call in each event generation; FALSE otherwise**Output:** λ_1 : the minimum eigenvalue of the Hessian matrix, $\mathbf{H}(\mathbf{R}) = \partial^2 V / \partial \mathbf{R}^2$ $\mathbf{V}^1 \in \mathbb{R}^{3N}$: the Hessian eigenvector corresponding to λ_1 **Steps:**if *initialize*randomize $\Delta \in \mathbb{R}^{3N}$, such that it contains no translational motion $s \leftarrow 0$ $\beta^s \leftarrow \|\Delta\|$ $\mathbf{Q}^s (\in \mathbb{R}^{3N}) \leftarrow 0$

do

 $s \leftarrow s + 1$ $\mathbf{Q}^s \leftarrow \Delta / \beta^{s-1}$ $c_{\text{fd}} \leftarrow \max_{i\alpha} \{ |q_{i\alpha}^s| \mid i = 1, \dots, N; \alpha = x, y, z \} / \delta_{\text{fd}}$ $\Delta \leftarrow c_{\text{fd}} [-\mathbf{F}(\mathbf{R} + \mathbf{Q}^s / c_{\text{fd}}) + \mathbf{F}(\mathbf{R})] - \beta^{s-1} \mathbf{Q}^{s-1}$ $\alpha^s \leftarrow \mathbf{Q}^{sT} \Delta$ $\Delta \leftarrow \Delta - \alpha^s \mathbf{Q}^s$ $\beta^s \leftarrow \|\Delta\|$

$$\text{diagonalize } \mathbf{T}_s = \begin{bmatrix} \alpha_1 & \beta_1 & & & & \\ \beta_1 & \alpha_2 & \beta_2 & & & \\ & & \ddots & \ddots & \ddots & \\ & & & \beta_{s-2} & \alpha_{s-1} & \beta_{s-1} \\ & & & & \beta_{s-1} & \alpha_s \end{bmatrix}, \quad \text{so that } \tilde{\mathbf{Q}}_s^T \mathbf{T}_s \tilde{\mathbf{Q}}_s = \text{diag}(\tilde{\lambda}_1^s, \dots, \tilde{\lambda}_s^s)^*$$

while $|(\tilde{\lambda}_1^s - \tilde{\lambda}_1^{s-1}) / \tilde{\lambda}_1^{s-1}| > \Delta_{\text{eigen}}$ $\lambda_1 \leftarrow \tilde{\lambda}_1^s$ $\mathbf{V}^1 \leftarrow \sum_{k=1}^s \mathbf{Q}^k \tilde{q}_k^1$ $\mathbf{V}^1 \leftarrow \mathbf{V}^1 / \|\mathbf{V}^1\|$

* $\text{diag}(\tilde{\lambda}_1^s, \dots, \tilde{\lambda}_s^s)$ is an s by s diagonal matrix, with its diagonal elements given by $\tilde{\lambda}_1^s, \dots, \tilde{\lambda}_s^s$. $\tilde{\mathbf{Q}}^s = [\tilde{q}^1, \dots, \tilde{q}^s]$ is an s by s orthogonal matrix, with $\tilde{q}^m \in \mathbb{R}^s$ is the m th eigenvector of \mathbf{T}_s .

other than individual bond breakings at the crack tip. An example is nanovoid nucleation ahead of the crack tip in glasses, which results from collective atomic motions and long-range stress relaxation [40].

5. Summary

We have designed a search algorithm for escape paths with low activation barriers starting from a local energy minimum configuration of a many-atom system. The pathfinder algorithm combines: (1) a steered eigenvector-following method to generate an escape path tangentially to the eigenvector corresponding to the lowest negative Hessian eigenvalue; (2) systematic combinatorial generation of concerted events as linear combinations of atomistic events; (3) control of population dynamics of low activation-barrier events; and (4) hybrid task + spatial decompositions to implement the algorithm on parallel computers. We have observed reasonable constant problem-size and isogranular parallel efficiencies. The program has been used to study concerted bond-breaking events in the fracture of alumina crystal. The pathfinder algorithm could be combined with other event-population control schemes such as genetic algorithms [21], which could then be used in kinetic Monte Carlo simulations [42,43] that feature on-demand construction of event lists during runtime to explore atomistic mechanisms underlying long-time behavior of materials [44].

Acknowledgements

This work was partially supported by AFOSR—DURINT, ARO—MURI, Chevron—CiSoft, DOE, and NSF. Numerical tests were performed at the University of Southern California using the 5384-processor Linux cluster at the Research Computing Facility and the 2048-processor Linux cluster at the Collaboratory for Advanced Computing and Simulations.

Appendix A. Lanczos algorithm to obtain the minimal Hessian eigenpair

The Lanczos algorithm is used to compute the minimum eigenvalue λ_1 and the corresponding eigenvector \mathbf{V}^1 of the Hessian matrix, to be used in the steered eigenvector-following event generator algorithm in Table 1.

References

- [1] A.F. Voter, F. Montalenti, T.C. Germann, Annual Review of Materials Research 32 (2002) 321.
- [2] D.G. Truhlar, B.C. Garrett, S.J. Klippenstein, Journal of Physical Chemistry 100 (1996) 12771.
- [3] B. Lawn, Fracture of Brittle Solids, Cambridge University Press, Cambridge, UK, 1993.
- [4] R.A. Olsen, G.J. Kroes, G. Henkelman, et al., Journal of Chemical Physics 121 (2004) 9776.

- [5] H.B. Schlegel, *Journal of Computational Chemistry* 24 (2003) 1514.
- [6] G.T. Barkema, N. Mousseau, *Physical Review Letters* 77 (1996) 4358.
- [7] G. Henkelman, H. Jonsson, *Journal of Chemical Physics* 111 (1999) 7010.
- [8] A. Banerjee, N. Adams, J. Simons, et al., *Journal of Physical Chemistry* 89 (1985) 52.
- [9] J. Baker, *Journal of Computational Chemistry* 7 (1986) 385.
- [10] P. Culot, G. Dive, V.H. Nguyen, et al., *Theoretica Chimica Acta* 82 (1992) 189.
- [11] L.J. Munro, D.J. Wales, *Physical Review B* 59 (1999) 3969.
- [12] N. Mousseau, P. Derreumaux, G.T. Barkema, et al., *Journal of Molecular Graphics & Modelling* 19 (2001) 78.
- [13] F.H. Stillinger, *Physical Review E* 59 (1999) 48.
- [14] J.P.K. Doye, D.J. Wales, *Journal of Chemical Physics* 116 (2002) 3777.
- [15] P.J. Feibelman, *Physical Review Letters* 65 (1990) 729.
- [16] G.T. Barkema, N. Mousseau, *Physical Review Letters* 81 (1998) 1865.
- [17] O. Trushin, A. Karim, A. Kara, et al., *Physical Review B* 72 (2005) 115401.
- [18] C. Zhang, B. Bansal, P.S. Branicio, et al., *Computer Physics Communications* 175 (2006) 339.
- [19] P.S. Branicio, R.K. Kalia, A. Nakano, et al., *Physical Review Letters* 96 (2006) 065502.
- [20] I. Szlufarska, A. Nakano, P. Vashishta, *Science* 309 (2005) 911.
- [21] M. Mitchell, *An Introduction to Genetic Algorithms*, MIT Press, Cambridge, MA, 1997.
- [22] K.A. DeJong, *Evolutionary Computation*, MIT Press, Cambridge, MA, 2002.
- [23] A.F. Voter, *Physical Review B* 57 (1998) R13985.
- [24] M. Shirts, V.S. Pande, *Science* 290 (2000) 1903.
- [25] G. Henkelman, H. Jonsson, *Physical Review Letters* 90 (2003) 116101.
- [26] H. Kikuchi, R.K. Kalia, A. Nakano, et al., in: *Proceedings of Supercomputing 2002*, IEEE/ACM, 2002.
- [27] S. Ogata, E. Lidorikis, F. Shimojo, et al., *Computer Physics Communications* 138 (2001) 143.
- [28] W. Gropp, E. Lusk, A. Skjellum, *Using MPI*, second ed., MIT Press, Cambridge, MA, 1999.
- [29] H. Takemiya, Y. Tanaka, S. Sekiguchi, et al., in: *Proceedings of Supercomputing 2006*, IEEE/ACM, 2006.
- [30] D. Wales, *Energy Landscapes: Applications to Clusters, Biomolecules and Glasses*, Cambridge University Press, Cambridge, UK, 2004.
- [31] G.A. Voth, D. Chandler, W.H. Miller, *Journal of Chemical Physics* 91 (1989) 7749.
- [32] We compute the minimum Hessian eigenvalue of the final state to check if it is a local energy minimum.
- [33] D.H. Lu, M. Zhao, D.G. Truhlar, *Journal of Computational Chemistry* 12 (1991) 376.
- [34] D.H. Lu, D.G. Truhlar, *Journal of Chemical Physics* 99 (1993) 2723.
- [35] A. Nakano, R.K. Kalia, P. Vashishta, et al., *Scientific Programming* 10 (2002) 263.
- [36] A. Nakano, R.K. Kalia, K. Nomura, et al., *Computational Materials Science* (2006), in press.
- [37] F. Shimojo, R.K. Kalia, A. Nakano, et al., *Computer Physics Communications* 167 (2005) 151.
- [38] L. Greengard, V. Rokhlin, *Journal of Computational Physics* 73 (1987) 325.
- [39] S. Ogata, T.J. Campbell, R.K. Kalia, et al., *Computer Physics Communications* 153 (2003) 445.
- [40] Z. Lu, K. Nomura, A. Sharma, et al., *Physical Review Letters* 95 (2005) 135501.
- [41] A. Nakano, R.K. Kalia, K. Nomura, et al., *International Journal of High Performance Computing Applications* (2007), accepted for publication.
- [42] K.A. Fichthorn, W.H. Weinberg, *Journal of Chemical Physics* 95 (1991) 1090.
- [43] A.F. Voter, in: K.E. Sickafus, E.A. Kotomin, B.P. Uberuaga (Eds.), *Radiation Effects in Solids*, Springer, Dordrecht, The Netherlands, 2006.
- [44] G. Henkelman, H. Jonsson, *Journal of Chemical Physics* 115 (2001) 9657.