Comput Mech (2007) 40:531–546
DOI 10.1007/s00466-006-0124-z

ORIGINAL PAPER

Search algorithm, and simulation of elastodynamic crack propagation by modified smoothed particle hydrodynamics (MSPH) method

R. C. Batra · G. M. Zhang

Received: 5 April 2006 / Accepted: 11 September 2006 / Published online: 24 November 2006

Abstract We first present a nonuniform box search algorithm with length of each side of the square box dependent on the local smoothing length, and show that it can save up to 70% CPU time as compared to the uniform box search algorithm. This is especially relevant for transient problems in which, if we enlarge the sides of boxes, we can apply the search algorithm fewer times during the solution process, and improve the computational efficiency of a numerical scheme. We illustrate the application of the search algorithm and the modified smoothed particle hydrodynamics (MSPH) method by studying the propagation of cracks in elastostatic and elastodynamic problems. The dynamic stress intensity factor computed with the MSPH method either from the stress field near the crack tip or from the *J*-integral agrees very well with that computed by using the finite element method. Three problems are analyzed. One of these involves a plate with a centrally located crack, and the other with three cracks on plates's horizontal centroidal axis. In each case the plate edges parallel to the crack are loaded in a direction perpendicular to the crack surface. It is found that, at low strain rates, the presence of other cracks will delay the propagation of the central crack. However, at high strain rates, the speed of propagation of the central crack is unaffected by the presence of the other two cracks. In the third problem dealing with the simulation of crack propagation in a functionally graded plate, the crack speed is found to be close to the experimental one.

R. C. Batra · G. M. Zhang (✉)
Department of Engineering Science and Mechanics,
M/C 0219, Virginia Polytechnic Institute and State University,
Blacksburg, VA 24061, USA
e-mail: gazhang1@vt.edu

Keywords Crack propagation · Meshless method · Search algorithm · Dynamic stress intensity factor · Functionally graded materials

1 Introduction

Modeling crack propagation during numerical and analytical solutions of a transient problem is very challenging especially if the crack path is unknown a priori. Following strategies are often employed to delineate fracture in the solution of a problem by the finite element method (FEM): (i) introduce cohesive elements along inter-element boundaries that are weak in shear and tension but very strong in compression; (ii) use nodal release technique by placing two nearly coincident but unconnected nodes and setting tractions on the newly created surfaces to zero, and (iii) reduce values of elastic constants and stresses developed in the failed regions to zero and essentially delete these elements from the analysis. Wang and Nakamura [1] have discussed four techniques, including the above three, for simulating material failure. Numerical techniques for modeling crack propagation include the finite difference method [2,3], the FEM [4,5], the boundary element method [6,7], and meshless methods [8–11]; only a few references are cited in each category to keep the list short, there is no way one can include here all papers dealing with fracture. Belytschko and Black [4] enriched the FE basis functions by adding to them four basis functions representative of the singular solution in linear elastic problems. Ching and Batra [12] and Batra and Ching [8] added these four basis functions to the monomials used to derive basis functions by the moving least squares method and determined

the stress intensity factor (SIF) near a crack tip with the meshless local Petrov–Galerkin (MLPG) method [13]. They accounted for the discontinuity in the trial solution by using the visibility [14] and the diffraction criteria [15].

In a meshless method employing not even a background mesh, such as the collocation method with radial basis functions [16], smoothed particle hydrodynamics (SPH) [17], modified smoothed particle hydrodynamics (MSPH) [18], the reproducing kernel particle method (RKPM) [19,20], and the MLPG [13], techniques (ii) and (iii) of modeling material failure are more viable than technique (i). Here we use the MSPH method and the nodal release technique to simulate crack initiation and propagation in elastodynamic problems. Whereas for static problems, one often uses either the critical SIF, or the critical crack opening displacement, or the critical value of the energy release rate as the criterion for crack initiation, for a dynamic problem the choice of the criterion is rather fuzzy. For example, Batra and Love [21] found that in mode-I transient deformations of a tungsten plate the J-integral increases essentially monotonically with the crack length when the nominal axial strain rate is 200/s but exhibits oscillations at the higher nominal axial strain rate of 2,000/s. Here we adopt the same criterion as used by Batra and Love [21], namely, that a crack initiates at a point when the maximum tensile principal stress there reaches a material-dependent critical value. Thus the criterion is local, does not incorporate any length scale, and does not account for deformations in the neighborhood of a point. We also use the analog of the nodal release technique to simulate crack opening. The presently computed dynamic stress intensity factor is found to match well with that of Nishioka and Atluri [22], who used the FEM and included moving singular elements embedded with analytical asymptotic solutions to account for the singularity near the crack tip.

Many numerical methods require a search algorithm to identify particles or nodes in the neighborhood of a given particle or node. In a transient problem particles are continuously displaced and one may need to find after every time step particles in the neighborhood of a given particle which can be computationally expensive. Even in a static problem where nodes are densely placed in one region to accurately compute deformations, a search algorithm can take considerable CPU time to ascertain particles in the neighborhood of every particle.

The search algorithm usually includes two steps. The first step is to separate the domain into different smaller domains, and the second step is to determine all particles in the neighborhood of a given particle. During the second step, only particles in certain smaller domains are considered instead of the whole domain. So the efficiency of the search algorithm is mainly determined by the first step, i.e., dividing of the whole domain into smaller domains. The direct search algorithm, which is also the easiest one, calculates the distance of every particle in the domain from the given particle i to check whether or not it is in the neighborhood of the particle i. Obviously, the direct search algorithm does not include the first step, and is CPU intensive requiring N^N operations where N equals the total number of particles. Thus the CPU time of the direct search algorithm increases dramatically with an increase in the number of particles.

There are two types of methods that first divide the domain. The first category is tree-like method, such as hierarchical tree [23], binary tree and Barnes–Hut tree [24] and the tree-code [25]. In the tree-code a cube of side l enclosing all particles is subdivided into eight cubic boxes with each of them further subdivided into eight children boxes. The process is continued till the smallest boxes, called the terminal boxes, have only one particle in them. Capuzzo-Dolcetta and Miocchi [25] have proposed the fast multipole algorithm that is similar to the tree algorithm except that the smallest box contains more than one particle and reduces the CPU time.

Another commonly used method is the box algorithm [26,27]. The uniform box search algorithm divides the computational domain into uniform square boxes with each side of length equal to twice the maximum smoothing length, $h_{\max}$. This requires searching particles in boxes that are in the neighborhood of the box containing the particle i, and needs N operations thereby reducing the CPU time as compared to that for the direct search algorithm.

The order of the CPU time required for the tree algorithms is about $N \log N$, while that for the box algorithm is proportional to N. The CPU time for the tree-like algorithm decreases to $O(N^2)$ when the tree is severely unbalanced. For the box algorithm [26,27], when the distances between adjacent particles are of similar sizes, so are sizes of boxes. When the distances between adjacent particles vary noticeably as, for example, in the SPH formulation, the smoothing lengths of some particles are much larger than those of other particles. For such problems, the use of uniform boxes will greatly increase the CPU time.

The analysis of contact problems also uses either body-based or space-based search algorithms [28–32] both of which require CPU time proportional to $N \ln(N)$ where N equals the total number of discrete elements. The performance of coordinate hashing algorithms is influenced by the number of overlapping boxes. Munjiza

and Andrews [27] have proposed a no binary search contact detection algorithm for problems involving a large number of bodies whose CPU time is directly proportional to the number of particles.

In order to improve upon the uniform box search algorithm at least for some class of problems, we propose here a nonuniform box search algorithm with the length of the side of a square box determined by the local smoothing length of the kernel function used to find kernel estimates of the unknown function. For a transient problem, we also give an empirical rule for estimating the time interval after which the nonuniform box search algorithm should be applied. For one example problem involving unevenly distributed nodes/particles, we show that the proposed nonuniform search algorithm requires nearly 70% less CPU time than the uniform box search algorithm, which requires 90% less CPU time than the direct search algorithm. Also the CPU time for the nonuniform search algorithm decreases back to $O(N)$. The proposed modification can be seen as an extension of the current box algorithm [26,27].

2 Brief review of the MSPH method

The MSPH method, proposed by Zhang and Batra [18], overcomes the two weaknesses, namely the tensile instability and the inconsistency, of the classical SPH method. In the MSPH method, the concept of the kernel estimate is applied to the Taylor series expansion, terms up to and including second-order derivatives are retained and a set of simultaneous equations is solved for the kernel estimates of the function $f(\mathbf{x})$ and its first and second order derivatives. For example, the Taylor series expansion of the function $f(\mathbf{x})$ up to its second-order derivatives is

$$f(\boldsymbol{\xi}) = f\big(\mathbf{x}^{(i)}\big) + \frac{\partial f}{\partial x_\alpha^{(i)}}\Big(\xi_\alpha - x_\alpha^{(i)}\Big) + \frac{1}{2}\frac{\partial^2 f}{\partial x_\alpha^{(i)}\partial x_\beta^{(i)}}\Big(\xi_\alpha - x_\alpha^{(i)}\Big)\Big(\xi_\beta - x_\beta^{(i)}\Big), \tag{2.1}$$

where a repeated Greek index implies summation over the range of the index, but no summation is implied on repeated Latin indices enclosed in parentheses. Let h be the smoothing length, and $2h$ the compact support of the non-negative kernel function $W(\mathbf{x}-\boldsymbol{\xi}, h)$. Multiplying both sides of Eq. (2.1) by $W(\mathbf{x}-\boldsymbol{\xi}, h)$, its first derivative $W_\gamma = \partial W/\partial x_\gamma$, and its second derivative $W_{\gamma\delta} = \partial^2 W/\partial x_\gamma \partial x_\delta$, and integrating the resulting equations over the domain Ω, we obtain the following system (2.2) of equations for the determination of the kernel estimate of $f(\mathbf{x})$, and of its first and second order derivatives at the point $\mathbf{x}^{(i)}$ of domain Ω .

$$B_{IJ}F_J = T_I \quad I = 1,2,\ldots,10, \tag{2.2}$$

where the repeated index J is summed, and

$$\begin{aligned}
&B_{IJ} = \sum_{j=1}^{N} \Phi(I)\Theta(J)\frac{m_j}{\rho_j},\\
&\Phi(1) = W_{ij}, \quad \Phi(2) = W_{ij,1}, \quad \Phi(3) = W_{ij,2},\\
&\Phi(4) = W_{ij,3},\\
&\Phi(5) = W_{ij,11}, \quad \Phi(6) = W_{ij,22}, \quad \Phi(7) = W_{ij,33},\\
&\Phi(8) = W_{ij,12}, \quad \Phi(9) = W_{ij,23}, \quad \Phi(10) = W_{ij,31};\\
&\Theta(1) = 1, \quad \Theta(2) = \xi_1^{(j)} - x_1^{(i)}, \quad \Theta(3) = \xi_2^{(j)} - x_2^{(i)},\\
&\Theta(4) = \xi_3^{(j)} - x_3^{(i)},\\
&\Theta(5) = \frac{1}{2}(\xi_1^{(j)} - x_1^{(i)})^2, \quad \Theta(6) = \frac{1}{2}(\xi_2^{(j)} - x_2^{(i)})^2,\\
&\Theta(7) = \frac{1}{2}(\xi_3^{(j)} - x_3^{(i)})^2,\\
&\Theta(8) = (\xi_1^{(j)} - x_1^{(i)})(\xi_2^{(j)} - x_2^{(i)}),\\
&\Theta(9) = (\xi_2^{(j)} - x_2^{(i)})(\xi_3^{(j)} - x_3^{(i)}),\\
&\Theta(10) = (\xi_3^{(j)} - x_3^{(i)})(\xi_1^{(j)} - x_1^{(i)});\\
&F = \{f_i, f_{1i}, f_{2i}, f_{3i}, f_{11i}, f_{22i}, f_{33i}, f_{12i}, f_{23i}, f_{31i}\}^{\mathrm{T}},\\
&T_I = \sum_{j=1}^{N} f_j \Phi(I)\frac{m_j}{\rho_j},\\
&W_{ij} = W\Big(\mathbf{x}^{(i)} - \boldsymbol{\xi}^{(j)}, h\Big), \quad W_{ij,\alpha} = \left.\frac{\partial W}{\partial x_\alpha}\right|_{\mathbf{x}=\mathbf{x}^{(i)},\boldsymbol{\xi}=\boldsymbol{\xi}^{(j)}},\\
&W_{ij,\alpha\beta} = \left.\frac{\partial^2 W}{\partial x_\alpha \partial x_\beta}\right|_{\mathbf{x}=\mathbf{x}^{(i)},\boldsymbol{\xi}=\boldsymbol{\xi}^{(j)}},\\
&f_i = f(\mathbf{x}^{(i)}), \quad f_{\alpha i} = \frac{\partial f}{\partial x_\alpha^{(i)}}, \quad f_{\alpha\beta i} = \frac{\partial^2 f}{\partial x_\alpha^{(i)}\partial x_\beta^{(i)}}.
\end{aligned} \tag{2.3}$$

The integer N equals the number of particles in the compact support of particle i, and m_i and ρ_i denote, respectively, the mass and the mass density of particle i located at $\mathbf{x}^{(i)}$. It is clear that the kernel estimate of the function so obtained is second-order consistent, and kernel estimates of the first and the second derivatives are, respectively, first-order and zero-order consistent.

For a two-dimensional (2-D) problem, we set

$$W(\mathbf{x}-\boldsymbol{\xi}, h) = \begin{cases} \frac{1.10081}{(h\sqrt{\pi})^2}\left(\mathrm{e}^{-|\mathbf{x}-\boldsymbol{\xi}|^2/h^2} - \mathrm{e}^{-4}\right) & |\mathbf{x}-\boldsymbol{\xi}| \leqslant 2h \\ 0 & |\mathbf{x}-\boldsymbol{\xi}| > 2h \end{cases} \tag{2.4}$$

Thus the compact support of the kernel function is a circle of radius $2h$ with center at the point $\mathbf{x}$, and the integral of $W(\mathbf{x}-\xi,h)$ over the support equals 1.

3 Governing equations and discretization

Governing equations for a 2-D linear elastodynamic problem are

$$\frac{\mathrm{d}U_\alpha}{\mathrm{d}t}=\frac{1}{\rho}\sigma_{\alpha\beta,\beta}+b_\alpha, \tag{3.1}$$

$$\sigma_{\alpha\beta}=\lambda\varepsilon_{kk}\delta_{\alpha\beta}+2\mu\varepsilon_{\alpha\beta}, \tag{3.2}$$

$$\varepsilon_{\alpha\beta}=\frac{1}{2}\left(u_{\alpha,\beta}+u_{\beta,\alpha}\right), \tag{3.3}$$

$$U_\alpha=\frac{\mathrm{d}u_\alpha}{\mathrm{d}t}, \tag{3.4}$$

where u_α is the displacement, U_α the velocity, ρ the mass density, t the time, $\sigma_{\alpha\beta}$ the Cauchy stress tensor, $\varepsilon_{\alpha\beta}$ the strain tensor, b_α the body force per unit mass, and λ and μ the Lame' constants. Equation (3.1) expresses the conservation of linear momentum, Eq. (3.2) the constitutive relation for a linear elastic isotropic material, Eq. (3.3) the relation between the infinitesimal strain tensor and the displacement gradients, and Eq. (3.4) defines the velocity of a material point. A repeated index in Eq. (3.1) implies summation over the range of the index.

If the function f in Eq. (2.2) is replaced by the displacement u_α, the first derivatives of the displacement $u_{\alpha,\beta}$ at the particle i can be evaluated in terms of displacements of particles that are in the neighborhood of the particle i. From derivatives of the displacement, the strain tensor can be computed by using Eq. (3.3) and the Cauchy stress tensor by Eq. (3.2). Replacing the function f in Eq. (2.2) by the stress component $\sigma_{\alpha\beta}$, we can compute the first derivatives, $\sigma_{\alpha\beta,\gamma}$, of the stress tensor at the particle i in terms of stresses at particles in the neighborhood of particle i. Once values of variables on the right-hand side of Eq. (3.1) have been found, we use the central-difference scheme to integrate with respect to time Eqs. (3.1) and (3.4) and compute the velocity and the displacement field at the new time. The time step, Δt_i, of particle i is given by the Courant–Frederich–Levy (CFL) condition,

$$\Delta t_i=\alpha\frac{h_i}{C_{Si}+|U_i|}, \tag{3.5}$$

where C_{Si} is the speed of an elastic wave at particle i, and the constant α is less than 1.0. We set $\alpha=0.5$ in our computations, and the time step, ΔT, equal to the minimum of Δt_i for all particles, i.e.,

$$\Delta T=\min(\Delta t_i, i=1,2,\ldots,N). \tag{3.6}$$

We note that there is no artificial viscosity introduced, thus the computed solution especially near the crack tip is likely to oscillate.

4 Search algorithm

4.1 Static problem

A meshless method generally requires the determination of all particles that are in the neighborhood of a given particle and are also in the compact support of the kernel or the weight function associated with the particle. The compact support of particle i usually equals a circle of radius $2h_i$ centered at the particle i. This search is usually computationally expensive.

The box search algorithm described in Sect. 1 may become inefficient when particles are unevenly distributed and the smoothing lengths of particles in one zone are much larger than those of particles in another zone. So in problems requiring closely spaced particles in some regions and widely spaced in others such as those having a propagating crack or a region of intense plastic deformations, the uniform box search algorithm is inefficient. The smoothing length in the coarsely spaced region is likely to be several times, say n_α, that in the densely spaced region. In the uniform box search algorithm, the length of the boxes' side is $2h_{\max}$. Thus the box in the finely discretized zone will include about $(n_\alpha)^\beta$ (β is the dimensionality of space) times as many particles as will a box in the coarsely discretized zone, and the CPU time for boxes in the finely discretized region will be quite large. In order to overcome this, we propose using nonuniform boxes with the length of boxes' sides varying with the local smoothing length so that no one box will include a large number of particles. This nonuniform box search algorithm for a 2-D problem can be implemented as follows:

1. First determine the maximum and the minimum values, $x_{\max}$, $y_{\max}$, $x_{\min}$, $y_{\min}$, among coordinates of all particles, and also the maximum smoothing length, $h_{\max}$. If the condition $y_{\max}-y_{\min}\geqslant x_{\max}-x_{\min}$ is satisfied, divide the domain from the y-direction, otherwise divide it from the x-direction. Below, we assume that the algorithm is carried out from the y-direction, and set $h_0=0$.
2. In the zone $[y_{\max}-4h_{\max}, y_{\max}]$ determine the maximum and the minimum x- coordinates $x^i_{\max}, x^i_{\min}$, and the maximum smoothing length $h^i_{\max}$. Let $h=\max(h^i_{\max}, h_0)$.

3. In the zone $[y_{\max} - 2\bar{h}, y_{\max}]$ assign square boxes from $x^i_{\min}$ to $x^i_{\max}$ with the length of the side of a box $= 2\bar{h}$.
4. Determine the box number of every particle in the zone $[y_{\max} - 2\bar{h}, y_{\max}]$. Let $h_0 = h^i_{\max}$ and $y_{\max} = y_{\max} - 2\bar{h}$.
5. If $y_{\max} > y_{\min}$, go to step 2, otherwise go to step 6.
6. Find particles in the neighborhood of a particle by first locating its box number, and then looking at particles that are in boxes neighboring the particle's box.

In order to ascertain the computational efficiency of the three algorithms, namely the direct search algorithm, the uniform box search algorithm, and the proposed nonuniform box search algorithm, we consider the 12 mm × 15 mm rectangular domain and employ different discretizations in zones A, B, C and D depicted in Fig. 1. (We have not experimented with either the tree algorithm or any of the others described in Sect. 1.) In each zone, particles are uniformly spaced, and the smoothing length equals 1.5 times the particle distance in that zone; particle distances in these zones are related as $\Delta_B = 2\Delta_A$, $\Delta_C = 2\Delta_B$, and $\Delta_D = 2\Delta_C$.

For eight values of the total number of particles, Table 1 gives, respectively, the CPU times G_d, G_u and G_n of the direct search, the uniform box search, and the nonuniform box search algorithms. For each value of the total number of particles, the distance Δ_A between two adjacent particles in zone A is also listed in the table. It is clear that with an increase in the total number of particles, the ratio of the CPU time taken by the nonuniform box search algorithm to that for either the direct or the uniform box search algorithm decreases rapidly. For each case studied, the nonuniform box search algorithm takes the least CPU time and the direct search algorithm the most. For small number of particles, the uniform box search algorithm takes more CPU time than the direct search algorithm because the CPU time needed to compute boxes and determine every particle's box number exceeds that saved by searching particles in boxes neighboring particle's own box. Figure 2a–c exhibits the CPU time for the three algorithms versus the number of particles. It is clear that the CPU time for the direct search algorithm increases exponentially with an increase in the total number of particles, but that of the nonuniform box search algorithm increases almost linearly. It can be seen that the CPU time for nonuniform box search algorithm is of $O(N)$, the same as that for the uniform box search algorithm often used to solve problems of similar smoothing lengths.

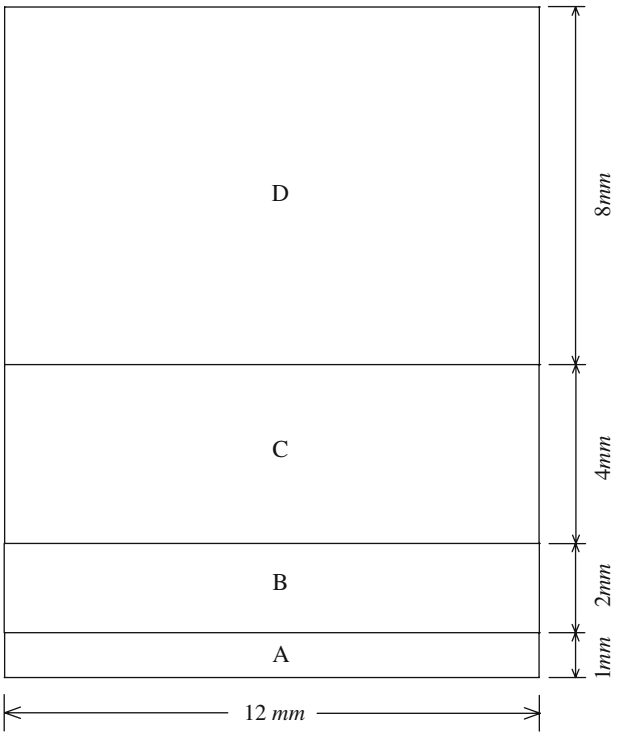

Fig. 1 Discretization of a domain

Table 1 Comparison of the CPU time required by three different search algorithms on a single processor PC

Particle distance in zone A, Δ_A(mm)	Total number of particles, N	CPU time (s)		
		Direct search algorithm, G_d	Uniform box search algorithm, G_u	Nonuniform box search algorithm, G_n
1/10	2,411	0.31	0.35	0.29
1/20	9,321	1.94	0.75	0.38
1/30	20,731	11.82	1.66	0.56
1/40	36,641	42.00	3.64	0.87
1/50	57,051	100.94	5.92	1.47
1/60	81,961	208.02	13.10	2.27
1/70	111,371	386.46	19.63	2.98
1/80	145,281	655.08	23.83	3.95

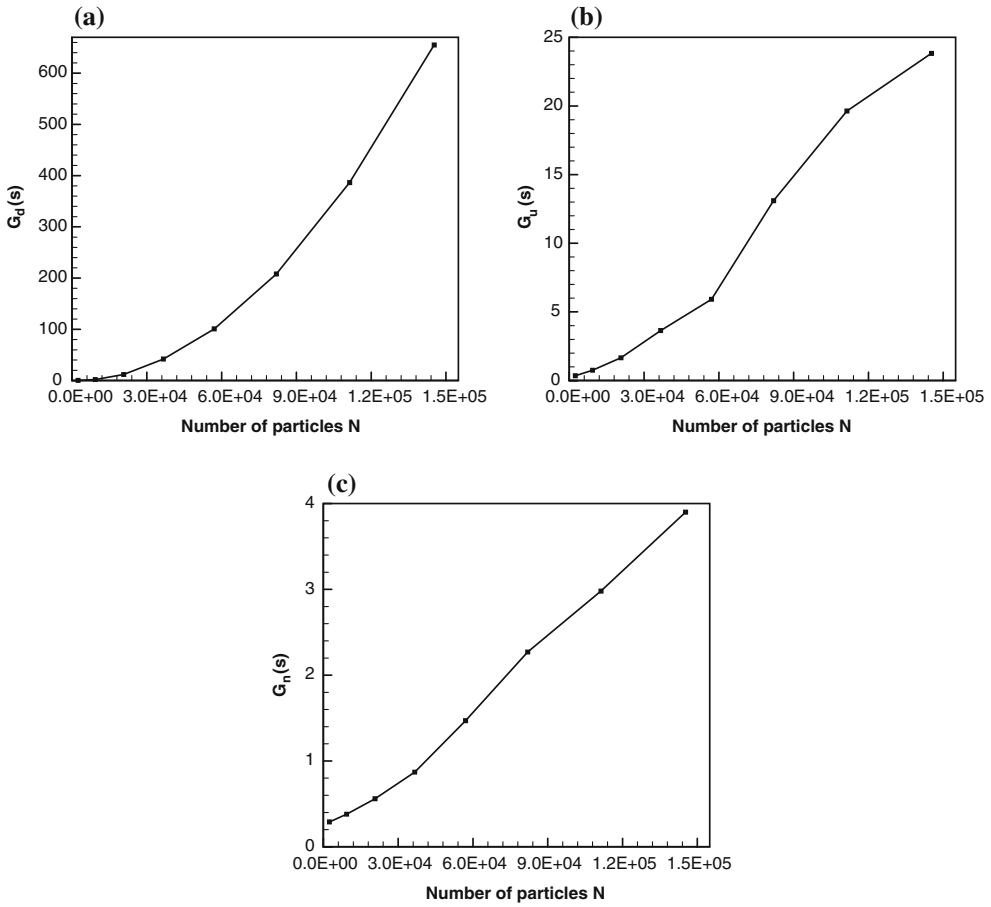

Fig. 2 The CPU time versus the number of particles for **a** the direct search algorithm, **b** the uniform box search algorithm, and **c** the nonuniform box search algorithm

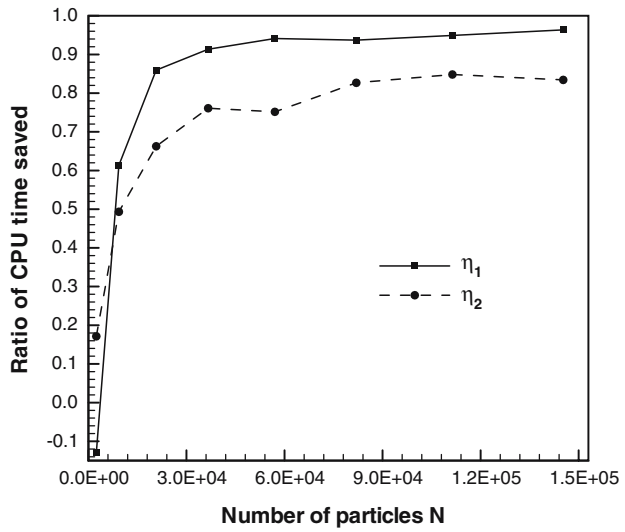

Fig. 3 The ratio of CPU time saved versus the number, N, of particles

Figure 3 gives the ratio $\eta_1 = \frac{G_d - G_u}{G_d}$ of the CPU time saved by using the uniform box search algorithm relative to that for the direct search algorithm, and the ratio $\eta_2 = \frac{G_u - G_n}{G_u}$ of the CPU time saved by using the nonuniform box search algorithm relative to that for the uniform box search algorithm. For 30,000 particles, the uniform box search algorithm can save more than 90% CPU time as compared to that for the direct search algorithm, while the nonuniform box search algorithm can save 70% CPU time as compared to that for the uniform box search algorithm.

For $N = 9321$, Fig. 4a, b depicts distributions of particles and boxes for the nonuniform and the uniform box search algorithms. The numbers of boxes for the uniform and the nonuniform box search algorithms equal 130 (13 layers) and 740 (23 layers), respectively. Whereas in the coarsely discretized zone distributions of boxes are the same for the two algorithms, the length of a side of a box decreases in the nonuniform box search algorithm due to the smaller smoothing length for it. The uniform box search algorithm can be viewed as a special case of the nonuniform one when the smoothing length is same throughout the computational domain.

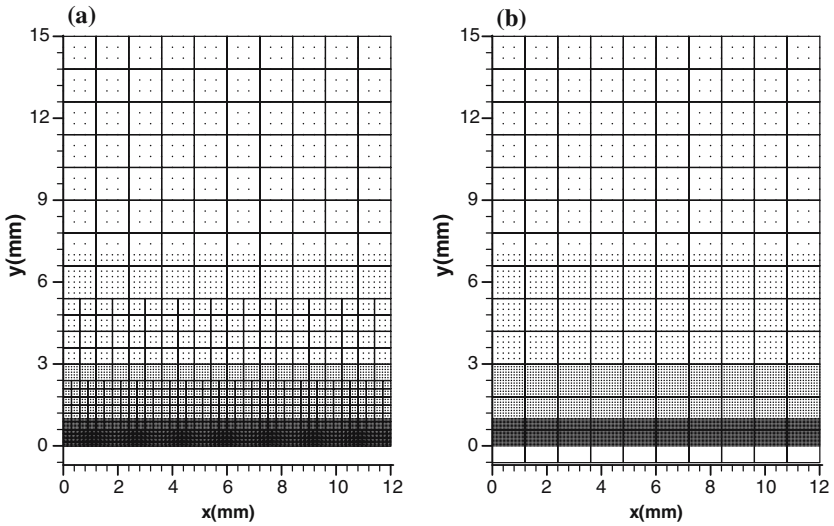

Fig. 4 Comparison of distributions of particles, and assigned boxes for **a** the nonuniform box search algorithm, and **b** the uniform box search algorithm

4.2 Dynamic problem

In a dynamic problem particles move and some may leave one particle's compact support while others may enter it. Thus, it may be necessary to determine neighbors of every particle after every time step. A dynamic problem often includes thousands of time steps, so for a large number of particles the total CPU time required to determine neighbors of particles may become a significant part of the total CPU time. We propose that in the search algorithm the length of the side of a box during the time interval, $\delta t = \frac{\psi h}{2|U_{\max}|}$, be taken as $(2+\psi)h$ since the relative movement between two particles in time δt will not exceed ψh. Here, $U_{\max}$ denotes the maximum speed of a particle. Thus, we can use the search algorithm after every δt rather than after every time step ΔT. Of course during every time step, additional CPU time is needed to determine the neighbor particles from the list of all possible neighbor particles. With an increase in the value of ψ, the additional CPU time needed to determine neighbor particles from the list of all possible neighbors will increase. So the coefficient ψ cannot be large. In our computations we take $\psi = 0.1$. Since $U_{\max} \ll C_{Si}$, therefore $\delta t \gg \Delta T$.

5 Applications

5.1 Dynamic stress intensity factor (DSIF)

We use the aforestated problem formulation and the search algorithm to find time history of the DSIF in a rectangular plate having a through-the-thickness crack on its horizontal centroidal plane with the crack centroid coinciding with the plate centroid, and the top and

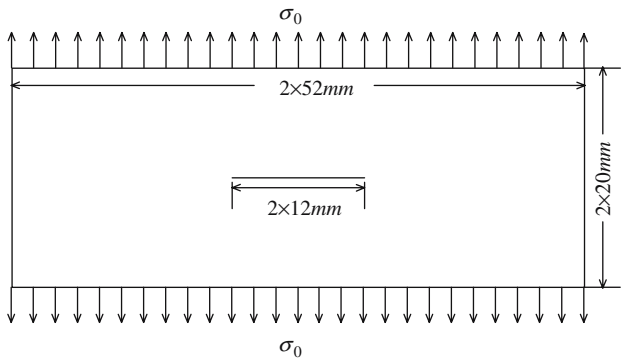

Fig. 5 Schematic sketch of a plate containing a through-the-thickness crack and loaded in tension

the bottom surfaces of the plate loaded by uniformly distributed surface tractions, as shown in Fig. 5. Dimensions of the plate are given in the figure, and we set $\sigma_0 = 0.4H(t)$ GPa, where $H(t)$ is the Heaviside function, shear modulus $\mu = 29.4$ GPa, Poisson's ratio $\nu = 0.286$, and mass density $\rho = 2,450\,\text{kg/m}^3$. Due to the symmetry of the problem about the horizontal and the vertical centroidal axes, plane strain deformations of only the North–East quarter of the plate are analyzed.

The discretization of the domain is similar to that shown in Fig. 1 except that it is divided only into three different zones, A, B and C. Zone A starts from the crack surface and extends 2 mm in the vertical direction, has particles separated by $\Delta = 0.1$ mm; Zone B is also 2 mm in the vertical direction and the particle distance in it is 0.2 mm; while the width of the top zone C is 16 mm and $\Delta = 0.4$ mm in it. The total number of particles equals 18,791, and the smoothing length in each zone equals 1.5Δ. In our computations, as shown in Fig. 6, the two traction free crack surfaces represented by lines 1-2-3-4-5-$\cdots$ and 1′-2′-3′-4′-5′-$\cdots$, are taken to

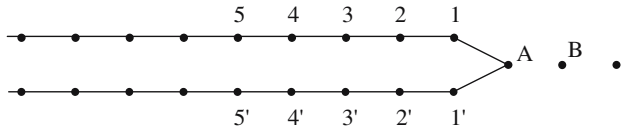

Fig. 6 Location of particles near the crack tip

be $\delta = 2 \times 10^{-3}$ mm apart (The figure is exaggerated in the direction perpendicular to the crack surface), particle A is the crack tip and discontinuous fields near the crack tip are modeled by the visibility criterion [14] and the diffraction criterion [15]. The traction free boundary conditions at the crack surface, except at the crack tip, are imposed by first transforming the stress tensor into local coordinates with axes aligned along and perpendicular to the crack surface, and then setting the normal and the tangential tractions equal to zero. The stress tensor is then transformed back to the global coordinate axes.

For mode-I loading, components of the stress field near the crack tip can be expressed as

$$\begin{aligned}
\sigma_{11}(r,\theta) &= \frac{K_I}{\sqrt{2\pi r}} \cos\frac{\theta}{2}\left(1 - \sin\frac{\theta}{2}\sin\frac{3\theta}{2}\right),\\
\sigma_{22}(r,\theta) &= \frac{K_I}{\sqrt{2\pi r}} \cos\frac{\theta}{2}\left(1 + \sin\frac{\theta}{2}\sin\frac{3\theta}{2}\right),\\
\sigma_{12}(r,\theta) &= \frac{K_I}{\sqrt{2\pi r}} \sin\frac{\theta}{2}\cos\frac{\theta}{2}\cos\frac{3\theta}{2},
\end{aligned} \tag{5.1}$$

where (r,θ) are polar coordinates of a point with the origin at the crack tip, and K_I is the dynamic stress intensity factor (DSIF). For $\theta = 0$ and 90°, Eq. $(5.1)_1$ gives,

$$K_I = \sqrt{2\pi r}\,\sigma_{22}(r, 0^\circ), \tag{5.2}$$

$$K_I = \frac{4}{3}\sqrt{\pi r}\,\sigma_{22}(r, 90^\circ). \tag{5.3}$$

We nondimensionalize the DSIF by $\sigma_0\sqrt{\pi a}$ and obtain

$$\bar{K}_I = \sqrt{2r/a}\,\frac{\sigma_{22}(r,0^\circ)}{\sigma_0}, \tag{5.4}$$

or

$$\bar{K}_I = \frac{4}{3}\sqrt{r/a}\,\frac{\sigma_{22}(r,90^\circ)}{\sigma_0}, \tag{5.5}$$

where a is the crack length.

Figure 7 depicts time histories of the non-dimensional DSIF computed from Eqs. (5.4) and (5.5) at the two particles closest to the crack tip by using both the diffraction and the visibility criteria to account for the discontinuity of displacements across the crack surfaces. The DSIF equals zero until the dilatational wave reaches the crack tip at about $t = 2.5\,\mu$s. It is apparent that the DSIF computed from Eq. (5.4) agrees well with that found by Nishioka and Atluri [22] by the FEM. But for $\theta = 90^\circ$, the difference between the DSIFs computed at the first and the second particles closest to the crack tip is large. Also there is large difference between the DSIFs computed with the diffraction and the visibility criteria. For both the diffraction and the visibility criteria, the DSIF computed from the stress field at the first particle closest to the crack tip is less than that found by Nishioka and Atluri, while the DSIF computed from the stress field at the second particle closest to the crack tip is greater than Nishioka and Atluri's value.

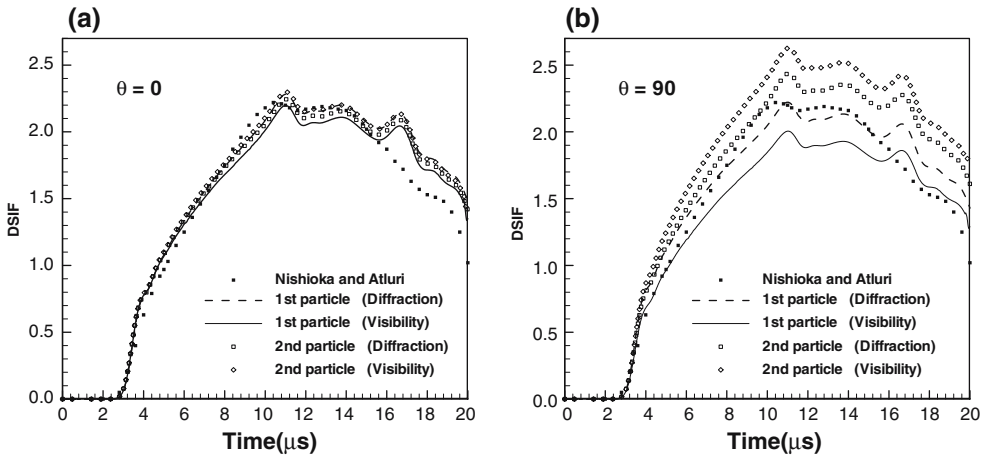

Fig. 7 Time histories of the nondimensional dynamic stress intensity factor (*DSIF*) computed by taking **a** $\theta = 0$ and **b** $\theta = 90^\circ$ using the diffraction and the visibility criteria

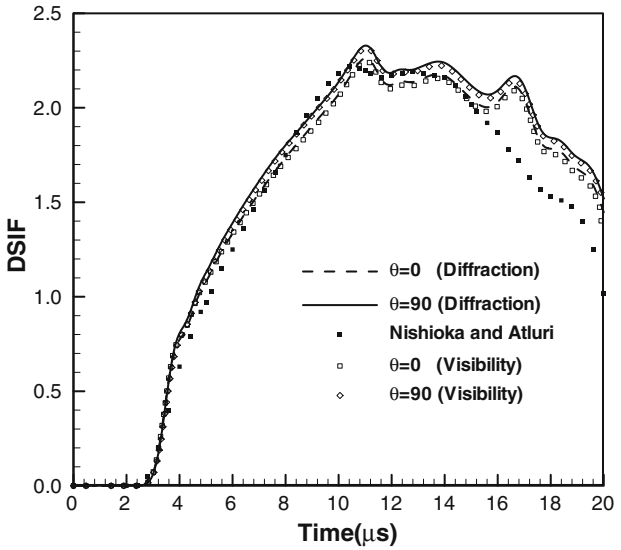

Fig. 8 Time histories of the nondimensional dynamic stress intensity factor (*DSIF*) computed by taking the average of its values for the first and the second particles closest to the crack tip

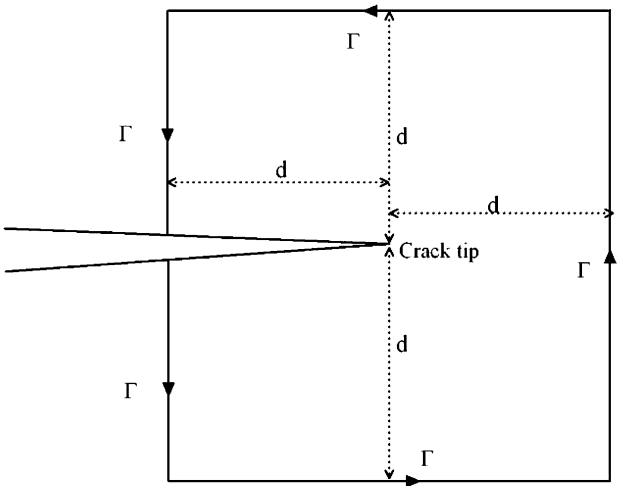

Fig. 9 Sketch of the contour for computing the *J*-integral

We have plotted in Fig. 8 the average of the DSIFs at the first and the second particles. It can be seen clearly from Fig. 8 that the DSIF computed with $\theta = 90°$ and that with $\theta = 0$ agree very well with that found by Nishioka and Atluri [22] by the FEM except when the time is greater than 15 μs. The maximum difference in values of the DSIFs found from Eqs. (5.4) and (5.5) is about 3%. Both the diffraction criterion and the visibility criterion can simulate the discontinuity very well and yield essentially the same value of the DSIF. Henceforth we use average of the results for the first and the second particles closest to the crack tip to be the DSIF and give results computed with Eq. (5.4).

Another method to find the DSIF is to first calculate the *J*-integral. For a 2-D problem, the *J*-integral is given by

$$J = \int_\Gamma \left[(w + I)\mathrm{d}x_2 - \sigma_{ij} n_j \frac{\partial u_i}{\partial x_1} \mathrm{d}s \right]$$

$$w = \int_0^{\varepsilon_{ij}} \sigma_{ij} \mathrm{d}\varepsilon_{ij} = \int_0^t \sigma_{ij} D_{ij} \mathrm{d}t,$$

$$D_{ij} = \frac{1}{2}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right), \quad I = \frac{1}{2}\rho U_i U_i. \tag{5.6}$$

Here, w is the strain energy density, I the kinetic energy density, Γ an arbitrary contour around the crack tip, and $\mathrm{d}s$ the element of arc length along the path Γ. We take Γ as the rectangular path with distance d of each side from the crack tip, as shown in Fig. 9.

For a static plane strain mode-I crack problem, the SIF is related to the *J*-integral by [6]

$$K = \sqrt{\frac{2\mu J}{1-\nu}}, \tag{5.7}$$

and the nondimensional SIF $\bar{K}$ is computed from

$$\bar{K} = \frac{1}{\sigma_0}\sqrt{\frac{2\mu J}{\pi a(1-\nu)}}. \tag{5.8}$$

Here we presume that Eqs. (5.7) and (5.8) hold for the present transient problem. Figure 10 exhibits time histories of the nondimensional DSIF for four values of the distance d, namely, 2Δ, 4Δ, 6Δ and 8Δ, where Δ is the distance between any two particles near the crack tip. For both the diffraction and the visibility criteria, the *J*-integral is found to be path independent, and the computed DSIF agrees with that given by Nishioka and Atluri [22]. Since there is very little difference between results computed with the diffraction and the visibility criteria, we discuss below results computed with the diffraction criterion, and also set $d = 4\Delta$.

5.1.1 Effect of the smallest distance between two particles

We now investigate the effect of the number of particles on the DSIF. Two cases are studied; first the total number of particles is increased so that the smallest distance between any two particles is decreased to one half of its previous value, and the particle distance Δ near the crack tip now equals 0.05 mm. In the second case the smallest distance between any two particles is further decreased by a factor of 2 and it now equals 0.025 mm near the crack tip. Results depicted in Fig. 11a show that

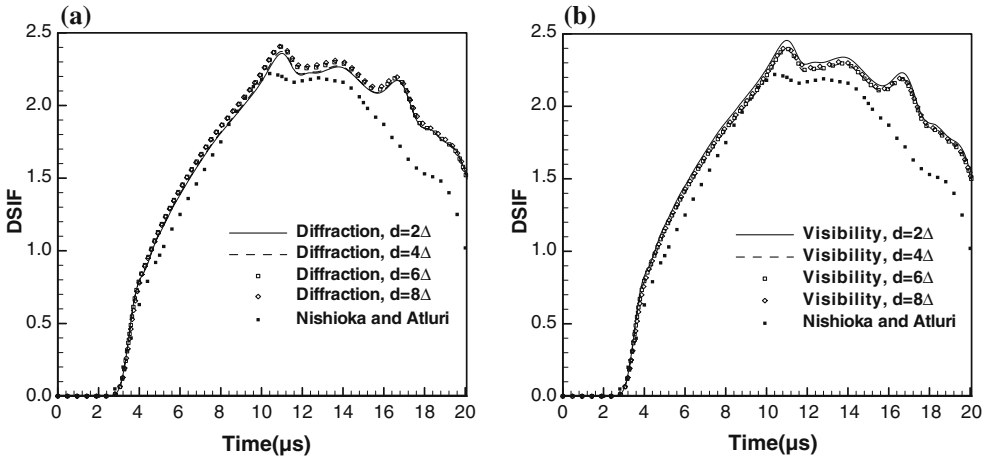

Fig. 10 Time histories of the nondimensional dynamic stress intensity factor (*DSIF*) computed from the *J*-integral with **a** the diffraction criterion, and **b** the visibility criterion

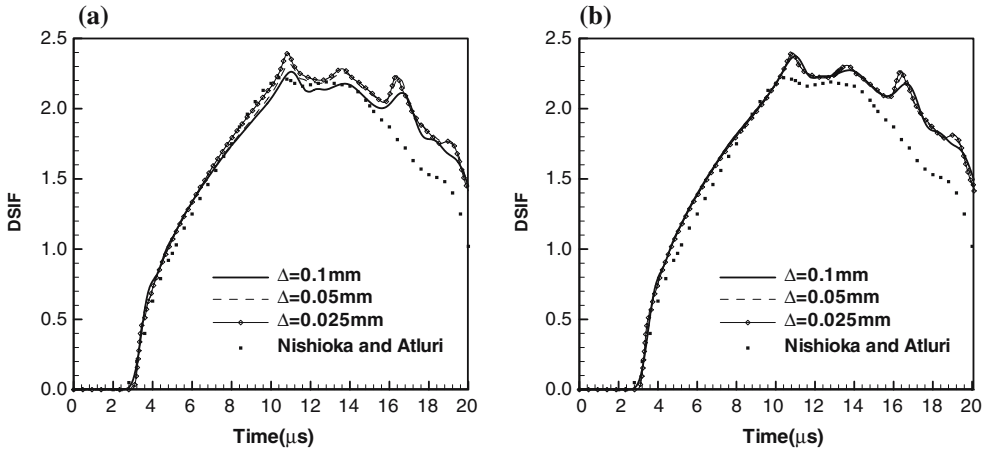

Fig. 11 Time histories of the nondimensional dynamic stress intensity factor (DSIF) computed by using **a** Eq. (5.4), and **b** the *J*-integral, for different values of the distance between adjacent particles near the crack tip

the total number of particles does not influence much the DSIF computed with Eq. (5.4). The DSIF increases with a decrease in the particle distance. The effect of the particle distance on the DSIF computed with the *J*-integral is much smaller. If we compare results of Fig. 11(a) with those of Fig. 11(b), we find that the DSIF computed with the *J*-integral is closer to the results computed by using Eq. (5.4) when the minimum distance between two particles is 0.025 mm.

5.1.2 Effect of the smoothing length

Time histories of the DSIF computed from Eq. (5.4) and with four values,1.2Δ, 1.5Δ, 2.0Δ and 2.5Δ, of the smoothing length, and with the *J*-integral are depicted in Fig. 12a, b. It is obvious that the DSIF decreases slightly with an increase in the smoothing length, and the DSIFs computed from the *J*-integral are close to each other for different values of the smoothing length.

5.1.3 Effect of the time step size

Figure 13 exhibits the effect of the time step size on time histories of the DSIF; the size of the time step is varied by setting the coefficient α in Eq. (3.5) equal to 0.1, 0.3, 0.5, 0.7 and 0.9. For both the DSIF computed by Eq. (5.4) and by the *J*-integral, the time step size seems to have virtually no effect on the results.

5.2 Simulation of Crack propagation

5.2.1 One central crack in a plate

We now study the propagation of a through-the-thickness crack centrally located in a square plate pulled

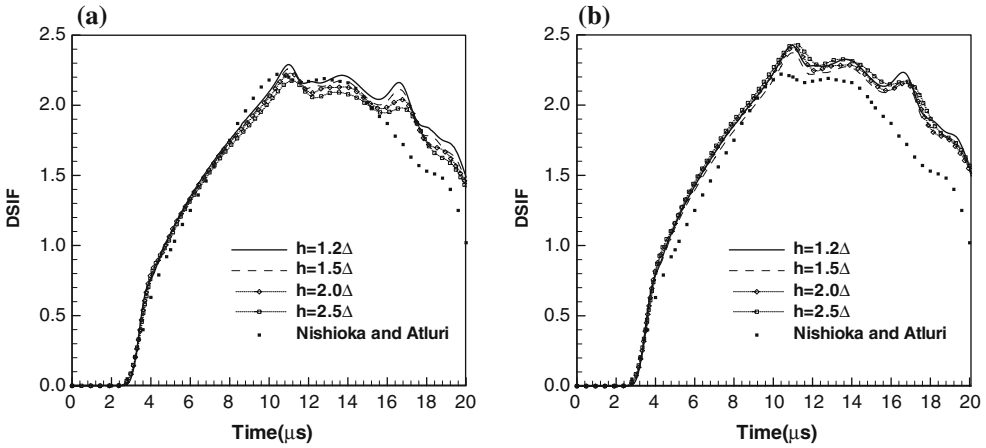

Fig. 12 Time histories of the nondimensional dynamic stress intensity factor (*DSIF*) computed by using **a** Eq. (5.4), and **b** the *J*-integral, for different values of the smoothing length

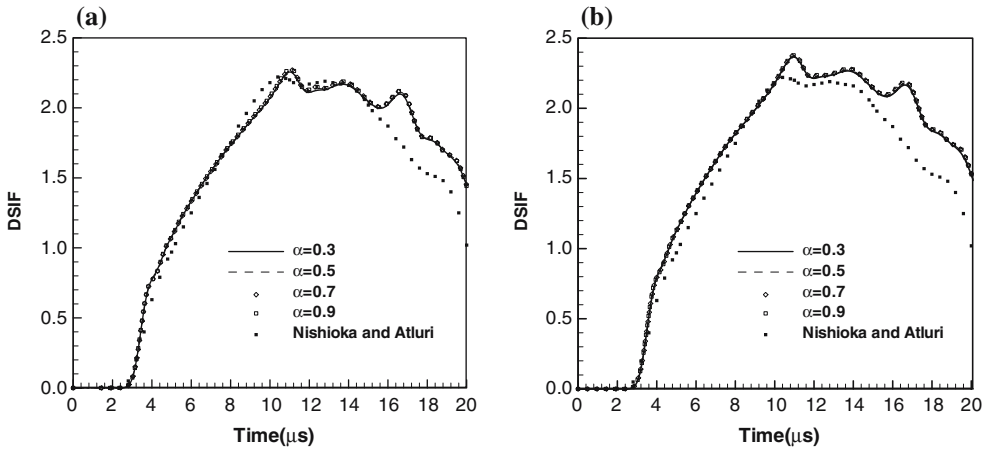

Fig. 13 Time histories of the nondimensional dynamic stress intensity factor (*DSIF*) computed by using **a** Eq. (5.4), and **b** the *J*-integral, for different values of the time step size

axially on opposite edges that are parallel to the crack surfaces so as to induce an average axial strain rate of either 200/s or 1,000/s. A schematic sketch of the problem studied, dimensions of the tungsten plate and the starter or the initial crack are shown in Fig. 14. Values assigned to material parameters are: Young's modulus $E = 400\,\text{GPa}$, Poisson's ratio $\nu = 0.29$, mass density $\rho = 19,300\,\text{kg/m}^3$, and the maximum principal tensile stress at crack opening = 4.5 GPa. Because of the symmetry of the problem and the boundary conditions about the two centroidal planes, we analyze deformations of a quarter of the plate, set normal velocities and tangential tractions equal to zero on the planes of symmetry, and assume that the crack propagates horizontally. The locations of particles in the domain are similar to those shown in Fig. 1. Zone A with the smallest distance, $\Delta = 0.025\,\text{mm}$, between two particles abuts the crack surface and extends 0.3 mm in the vertical direction; Zone B is 0.5 mm in the vertical direction and the smallest distance between any two particles in it is 0.05 mm. The widths of zones C and D equal, respectively, 1 and 8.2 mm, and Δ in them equals 0.1 and 0.2 mm, respectively. The total number of particles equals 10,324, and the smoothing length in each zone equals 1.5Δ. The Rayleigh wave speed, C_R, in the material can be obtained by first finding the largest root, η_max, of the cubic equation [34]

$$f(\eta) = \eta^3 + (4\gamma - 3)\eta^2 - \eta - (1 - 2\gamma)^2 = 0, \tag{5.9}$$

and then using

$$C_\text{R} = C_2\sqrt{\frac{\eta_\text{max}^2 - 1}{\eta_\text{max}^2 - \gamma}} \tag{5.10}$$

where $C_2 = \sqrt{\mu/\rho}$ and $\gamma = \mu/(\lambda + 2\mu)$. For the tungsten plate, $C_\text{R} = 2624.1\,\text{m/s}$.

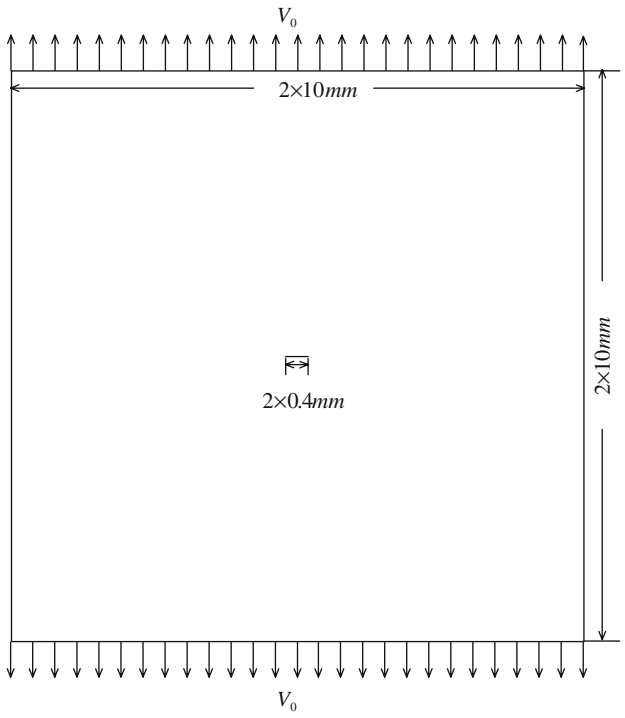

Fig. 14 Schematic sketch of a plate with a centrally located crack

Figure 15 shows schematically the propagation/growth of the crack. When the crack initiation criterion is satisfied at a particle, the crack is assumed to open there. The crack opening is handled differently depending upon whether or not the crack initiation criterion is satisfied next to the crack-tip or elsewhere. If particle B where the crack initiation criterion is met is next to the crack tip as shown in Fig. 15a, and $\cdots$-5-4-3-2-1 -A-1′-2′-3′-4′-5′-$\cdots$ is the crack surface, the crack is assumed to propagate from particle A to particle B which will become the new crack tip. The initiation of crack at particle A is simulated by placing two particles A″ and A′ at A with a small vertical distance $\delta = 2 \times 10^{-3}$ mm between them. Thus the new crack surface becomes $\cdots$-5-4-3-2-1 -A″-B-A′-1′-2′- 3′-4′-5′-$\cdots$. However, if the particle B is not next to the crack tip, as shown in Fig. 15b, a new crack is generated with two crack tips – one at particle 2 and the other at particle 3. If one of the two particles, say particle 2, happens to be the tip of another crack (see Fig. 15c), then the new crack tip is located at particle 3, and new crack surface is 1′-2′-B′-3-B″- 2″-1″, see Fig. 15d. Thus the crack has advanced by two particles rather than by one particle. In our numerical simulations we did not encounter a situation when the crack initiation criterion was met at a point whose distance from the crack tip was more than 2Δ.

Time histories of evolution of the crack length for average axial strain rates of 200/s and 1,000/s are depicted in Fig. 16 till the crack length becomes 5 mm. The crack begins to propagate at about 9.6 μs when the average axial strain rate is 200/s, but at about 2 μs for the nominal axial strain rate of 1,000/s. The crack speed at any time is taken to equal the slope of the least square line fit through 11 points with 5 points immediately preceding the crack tip and 5 points immediately following it. Figure 17 exhibits time histories of the crack propagation speed at the two nominal axial strain rates; the horizontal dashed line is the Rayleigh wave speed for the plate material. The crack accelerates faster when the plate is deformed at a nominal axial strain rate of 1,000/s than when it is deformed at the nominal axial strain rate of 200/s; however, the crack speed stays below the Rayleigh wave speed which agrees with Eischen's [35] result.

Figure 18 shows the crack propagation speed versus the crack length for average axial strain rates of 200/s and 1,000/s. In the beginning the crack propagation speed increases rapidly as it elongates by about 1 mm; subsequently, its speed increases rather slowly. The crack

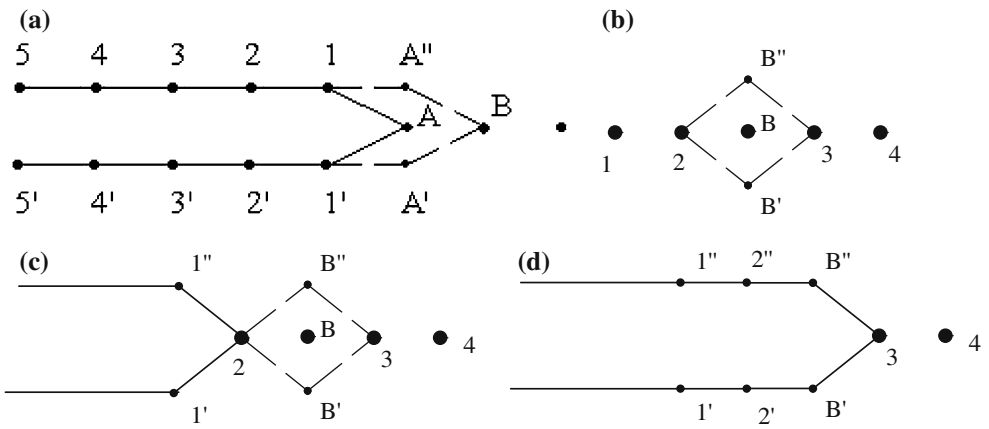

Fig. 15 Schematic sketch of crack propagation

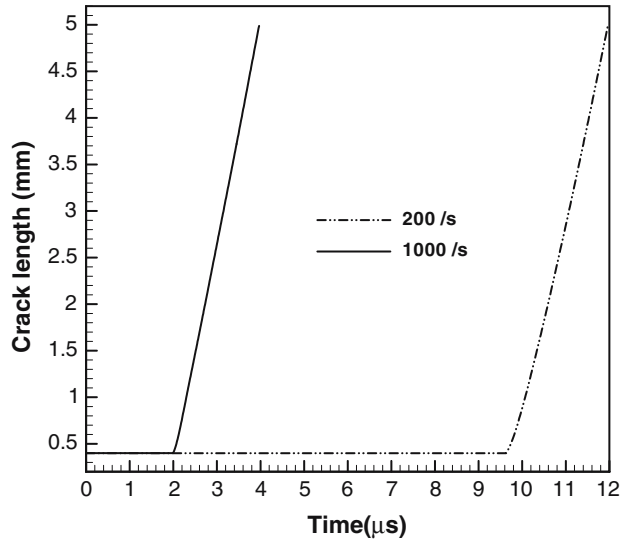

Fig. 16 Time histories of the crack length at nominal axial strain rates of 200/s and 1,000/s

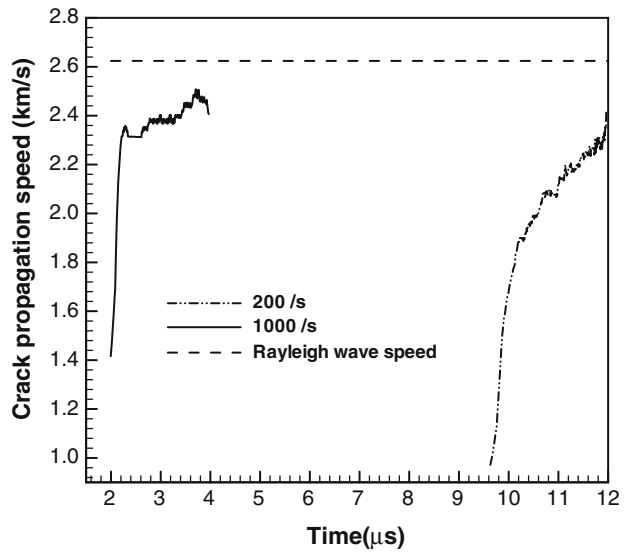

Fig. 17 Time histories of the crack propagation speed at nominal axial strain rates of 200/s and 1,000/s

propagation speed is higher for an axial strain rate of 1,000/s than that for the axial strain rate of 200/s; these results are in qualitative agreement with those of Batra and Love [21] who employed the FEM, the nodal release technique, and simulated crack propagation at nominal strain rates of 200/s and 2,000/s. For a crack length of 2 mm and the nominal strain rate of 200/s, both analyses give crack propagation speed of 1.9 km/s.

5.2.2 Three cracks on a centroidal axis of a plate

We now investigate the problem when the plate, as shown in Fig. 19, has three 0.8 mm long cracks on its

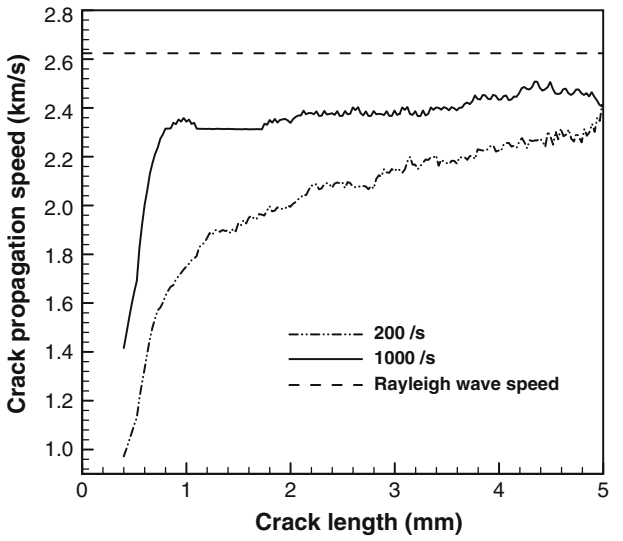

Fig. 18 Crack propagation speed versus the crack length

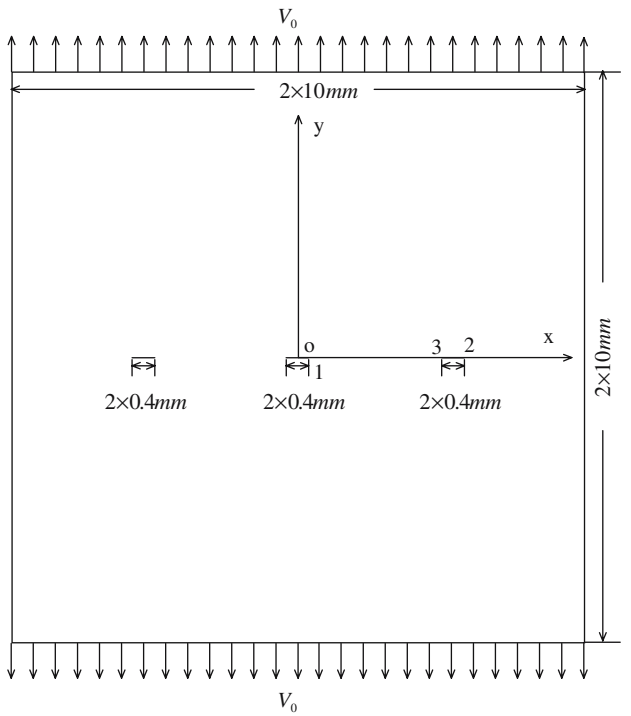

Fig. 19 Schematic sketch of a plate with three cracks

horizontal centroidal axis and the two opposite plate edges parallel to the crack are pulled vertically at a prescribed speed. Particles are located in a way similar to that in the plate having only one crack. Because of the symmetry of the problem about the horizontal and the vertical centroidal axes, deformations of a quarter of the plate are analyzed. Note that the loading wave arrives simultaneously at all crack tips/surfaces. Figures 20 and 21 exhibit time histories of the propagation speed

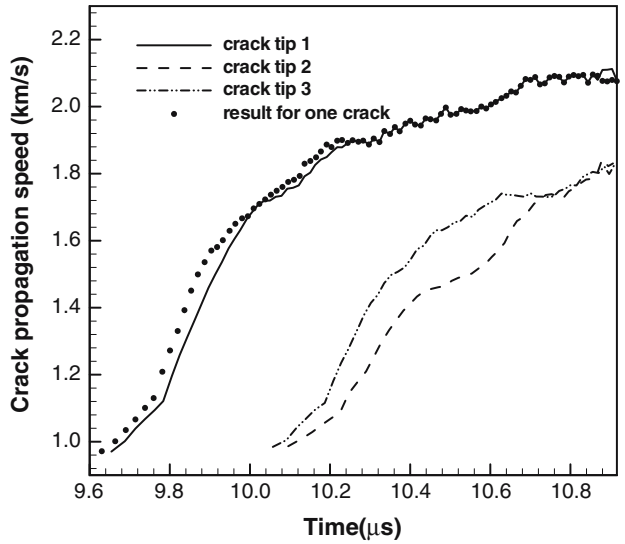

Fig. 20 Time histories of the crack propagation speed for average axial strain rate of 200/s

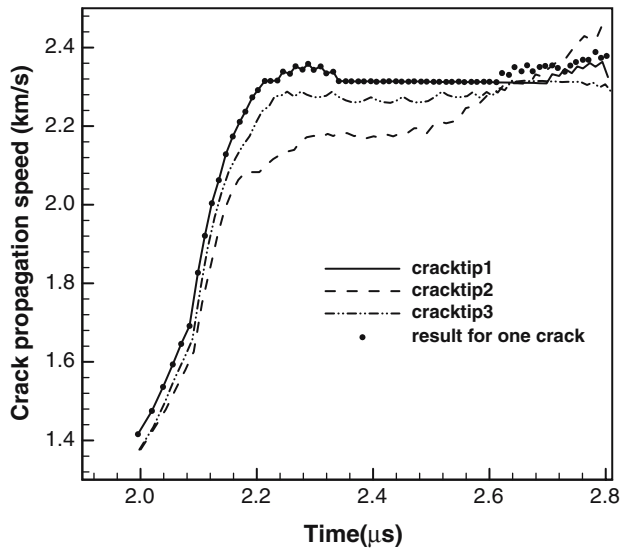

Fig. 21 Time histories of the crack propagation speed for nominal axial strain rate of 1,000/s

of crack tips 1, 2 and 3 for axial strain rates of 200/s and 1,000/s; the crack tip speed of only one centrally located crack studied in Sect. 5.2.1 is also plotted in this figure for comparison. For an axial strain rate of 200/s, it can be seen from Fig. 20 that the presence of the other two cracks delays the propagation of the central crack. The crack tip 3 begins to propagate $\sim 0.4\,\mu$s later than crack tip 1, and crack tip 2 starts to propagate a little later. When the nominal axial strain rate is 1,000/s, the three crack tips begin to propagate essentially simultaneously, and the presence of the other two cracks seems not to affect the propagation of the central crack.

5.2.3 Crack propagation in a functionally graded (FG) plate

Shukla and Jain [36,37] have studied experimentally the crack propagation in a FG elastic plate with material properties varying only in the anticipated horizontal direction of crack propagation as shown in Fig. 22. The single edge notch specimen with an edge crack at the left edge of initial length $a = 0.125\,W$ (W being the plate width) is pulled by applying equal and opposite axial velocities of 2.1 m/s at the top and the bottom surfaces of the plate resulting in mode-I deformations of the plate. Due to the symmetry of the problem about the horizontal centroidal plane, we analyze deformations of one-half of the plate, and use 360 particles in the x-direction near the crack tip.

Variations of the mass density, the shear modulus, and the static SIF (SSIF), K, along the x-direction obtained by fitting curves by the least squares method to the test data given in Jain and Shukla's paper [37] are given by

$$\rho(x) = 1,200\mathrm{e}^{-1.52x}(\mathrm{Kg/m^3})$$
$$\mu(x) = 1.345\mathrm{e}^{1.76x}(\mathrm{GPa})$$
$$K(x) = 2.18x + 0.71(\mathrm{MPam^{1/2}})$$

The Poisson ratio $\nu = 0.34$ is taken to be a constant.

In our simulations, a particle is assumed to have failed when the DSIF there equals twice the SSIF for that particle, and the crack is extended.

Figure 23, giving the time history of the crack length, shows that the crack begins to propagate at $\sim$90 μs. The crack speed equaling the slope of the crack length versus time curve is 822 m/s, which is close to the experimental speed of 600 and 700 m/s reported by Shukla and Jain [36].

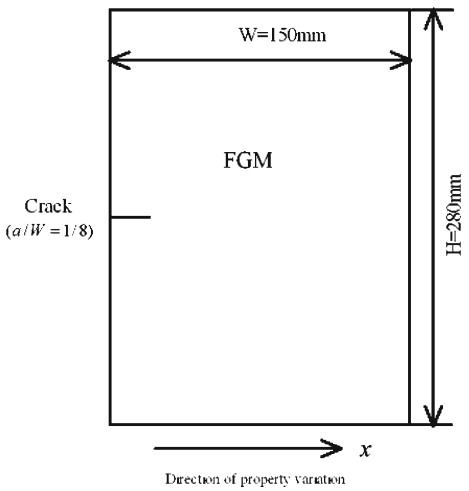

Fig. 22 The crack propagation in FG plate

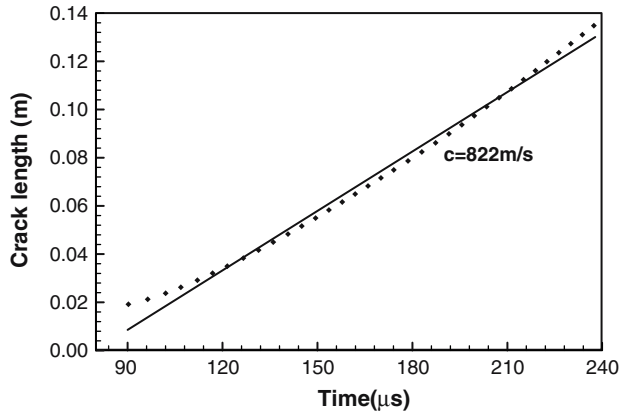

Fig. 23 Time history of the crack length in the FG plate (*Dots* are computed values, and the *solid line* is the best least squares fit)

We note that Jin and Batra [38,39] have shown that the crack-tip fields in a FG material are similar to those in a homogeneous material. Also, Batra and Zhang [40] have used the MSPH method to study wave propagation in a FG plate.

6 Conclusions

We have proposed a nonuniform box search algorithm in which the length of the side of a square box depends upon the local smoothing length. For a class of problems involving nodes placed densely in one part of the domain and coarsely in the other, it saves nearly 70% CPU time over the uniform box search algorithm which in turn equals 10% of the CPU time taken by a direct search algorithm. We have neither compared the performance of the proposed algorithm with that of other available algorithms nor tried it on several problems with randomly distributed particles having uneven spacing. Furthermore, for a dynamic problem, we have suggested and successfully tested a time increment after which the search algorithm should be applied.

We have employed the modified smoothed particle hydrodynamics (MSPH) method and the diffraction and the visibility criteria to analyze three linear elastodynamic crack problems. The computed dynamic stress intensity factor (DSIF) for mode-I loading of a cracked rectangular plate is found to match well with that obtained by Nishioka and Atluri using the finite element method. The DSIF computed from stresses at a point near the crack tip whose polar angle is 0° is better than that from stresses at a point with the polar angle of 90°. We also computed the J-integral, established its path independence, and used it to find the DSIF. Effects of the smallest distance between two particles, the smoothing length, and the time step size on the DSIF have also been examined.

During the analysis of transient deformations of a rectangular elastic plate containing a crack parallel to the two opposite edges that are pulled at a prescribed speed, it is found that the crack propagates sooner and accelerates faster at the nominal axial strain rate of 1,000/s than that at the average axial strain rate of 200/s. However, the highest computed crack propagation speed is lower than the Rayleigh wave speed of the material. In a rectangular plate containing one centrally located crack and two other cracks symmetrically located around it on the horizontal centroidal axis, the elongation of the central crack is delayed by the presence of the other two cracks at the nominal axial strain rate of 200/s but is virtually unaffected at the higher nominal axial strain rate of 1,000/s.

We have also simulated crack propagation in an edge cracked functionally graded plate loaded in tension at the two opposing faces parallel to the crack surface so as to induce mode-I deformation near the crack tip. Material parameters are assumed to vary only in the direction of the crack. The computed crack speed is found to match well with that found experimentally.

Acknowledgments This work was partially supported by the ONR grants N00014-1-98-0300 and N00014-1-06-0567 to Virginia Polytechnic Institute and State University (VPI&SU) with Dr. Y. D. S. Rajapakse as the program manager, and by the AFOSR MURI to Georgia Institute of Technology with a subcontract to VPI&SU. Views expressed herein are those of the authors and neither of the funding agencies nor of VPI&SU.

References

1. Wang Z, Nakamura T (2004) Simulations of crack propagation in elastic-plastic graded materials. Mech Mater 36: 601–622
2. Chen YM (1975) Numerical computation of dynamic stress intensity factors by a Lagrangian finite-difference method (the HEMP code). Eng Fract Mech 7:653–660
3. Lin X, Ballmann J (1993) Re-consideration of Chen's problem by finite difference method. Eng Fract Mech 44: 735–739
4. Belytschko T, Black T (1999) Elastic crack growth in finite elements with minimal remeshing. Int J Numer Method Eng 45:601–620
5. Aoki S, Kishimoto K (1978) Elastodynamic analysis of crack by finite element method using singular element. Int J Fracture 14:59–68
6. Fedelinski P, Aliabadi MH (1994) The dual boundary element method: J-integral for dynamic stress intensity factors. Int J Fracture 65:369–381
7. Albuquerque EL, Sollero P, Fedelinski P (2003) Dual reciprocity boundary element method in Laplace domain applied to anisotropic dynamic crack problems. Comput Struct 81:1703–1713

8. Batra RC, Ching H-K (2002) Analysis of elastodynamic deformations near a crack/notch tip by the Meshless Local Petrov-Galerkin (MLPG) method. CMES: Comp Model Eng Sci 3:717–730
9. Lee S-H, Yoon Y-C (2004) Numerical prediction of crack propagation by an enhanced element-free Galerkin method. Nucl Eng Des 227:257–271
10. Nairn JA (2003) Material point method calculations with explicit cracks. CMES: Comp Model Eng Sci 4:649–663
11. Atluri SN, 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: Comp Model Eng Sci 3:11–51
12. Ching H -K, Batra RC (2001) Determination of crack tip fields in linear elastostatics by the Meshless Local Petrov-Galerkin (MLPG) method. CMES: Comp Model Eng Sci 2:273–289
13. Atluri SN, Zhu T (1998) A new Meshless Local Petrov–Galerkin (MLPG) approach in computational mechanics. Comput Mech 22:117–127
14. Belytschko T, Lu YY, Gu L (1994) Element-free Galerkin methods. Int J Num Methods Eng 37:229–256
15. Organ D,Fleming M,Terry T, Belytschko T (1996) Continuous meshless approximations for nonconvex bodies by diffraction and transparency. Comput Mech 18:225–235
16. Franke C, Schaback R (1998) Convergence order estimates of meshless collocation methods using radial basis functions. Adv Comput Math 8:381–399
17. Lucy LB (1977) A numerical approach to the testing of the fission hypothesis. Astron J 82:1013–1024
18. Zhang GM, Batra RC (2004) Modified smoothed particle hydrodynamics method and its application to transient problems. Computational Mechanics 34:137–146
19. Liu WK (1995) An introduction to wavelet reproducing kernel particle methods. USACM Bull 8(1):3–16
20. Liu WK, Jun S, Zhang YF (1995) Reproducing kernel particle methods. Int J Numer Method Fluids 20:1081–1106
21. Batra RC, Love BM (2005) Crack propagation due to brittle and ductile failures in microporous thermoelastoviscoplastic functionally graded materials. Eng Fract Mech 72:1954–1979
22. Nishioka T, Atluri SN (1980) Numerical modeling of dynamic crack propagation in finite bodies, by moving singular elements, Part 2: results. J Appl Mech 47:577–582
23. Hernquist L, Katz N (1989) Treesph: a unification of SPH with the hierarchical tree method. Astrophys J Suppl Ser 70: 419–446
24. Waltz J, Page GL, Milder SD, Wallin J, Antunes A (2002) A performance comparison of tree data structures for N-body simulation. J Comput Phys 178:1–14
25. Capuzzo-Dolcetta R, Miocchi P (1998) A comparison between the fast multipole algorithm and the tree-code to evaluate gravitational forces in 3-D. J Comput Phys 143:29–48
26. Libersky LD, Petschek AG (1993) High strain Lagrangian hydrodynamics: a three-dimensional SPH code for dynamic material response. J Comput Phys 109:67–75
27. Munjiza A, Andrews KRF (1998) NBS contact detection algorithm for bodies of similar size. Int J Numerical Methods Eng 43:131–149
28. Oldenburg M, Nilsson L (1994) The position code algorithm for contact searching. Int J Numer Meth Eng 37:359–386
29. Bonet J, Peraire J (1991) An alternating digital tree (ADT) algorithm for 3D geometric searching and intersection problems. Int J Numer Meth Eng 31:1–17
30. O'Connor R, Gill J, Williams JR (1993) A linear complexity contact detection algorithm for multy-body simulation. In: Proceeding of 2nd U.S. Conference on Discrete Element Methods, MIT, MA
31. Preece DS, Burchell SL (1993) Variation of spherical element packing angle and its infuence on computer simulations of blasting induced rock motion. In: Proceeding of 2nd U.S. Conference on Discrete Element Methods, MIT, MA
32. Mirtich B (1988) Impulse-based dynamic simulation of rigid body systems, Ph.D. Thesis, Berkeley, California
33. Batra RC, Zhang GM (2004) Analysis of adiabatic shear bands in elasto-thermo-viscoplastic materials by modified smoothed-particle hydrodynamics (MSPH) method. J Comput Phys 201:172–190
34. Vinh PC, Ogden RW (2004) On formulas for the Rayleigh wave speed. Wave Motion 39:191–197
35. Eischen JW (1987) Fracture of nonhomogeneous materials. Int J Fract 34:3–22
36. Shukla A, Jain N (2004) Dynamic damage growth in particle reinforced graded materials. Int J Impact Eng 30: 777–803
37. Jain N, Shukla A (2006) Mixed mode dynamic fracture in particulate reinforced functionally graded materials. Exp Mech 46:137–154
38. Jin Z-H, Batra RC (2006) Crack tip fields in functionally graded materials with temperature-dependent properties. AIAA J 44:2129–2130
39. Jin Z-H, Batra RC (1996) Some basic fracture mechanics concepts in functionally graded materials. J Mechs Phys Solids 44:1221–1235
40. Zhang GM, Batra RC (2007) Wave propagation in functionally graded materials by modified smoothed particle hydrodynamics (MSPH) method. J Comput Phys, DOI 10.1016/j.jcp.2006.07.028 (in press)